BULETINUL ŞTIINȚIFIC

al

Universității „POLITEHNICA” din Timişoara, România

Seria ELECTRONICĂ ŞI TELECOMUNICAȚII

Număr special dedicat Simpozionului
de “Electronică şi Telecomunicații, ETc 2006”

Timişoara, 21-22 septembrie, 2006

SCIENTIFIC BULLETIN

of

the „POLITEHNICA” University of Timişoara, Romania

Transactions on ELECTRONICS AND COMMUNICATIONS

Tomul 51 (65), Fascicola 2, 2006
ISSN 1583-3380

EDITURA POLITEHNICA

International Scientific Committee

Honorary Chairman:

Prof. Dr. Jaakko Astola, Tampere University of Technology, Finland

Chairman:

Prof. Dr. Eng. Marius Otesteanu, "Politehnica" University of Timisoara, dean of the Faculty of Electronics and Telecommunications, Romania

Members:

Prof. Dr. Kyoki Imamura, Kyushu Institute of Technology, Japan

Prof. Dr. Emil Petriu, SITE, University of Ottawa, Canada

Prof. Dr. Andre Quinquis, ENSIETA Bretagne, Brest, France

Prof. Dr. Maria Victoria Rodellar Biarge, Politecnic University of Madrid, Spain

Prof. Nasir Memon, Department of Computer and Information Science, New York Polytechnic University, USA

Prof. Dr. Eng. Axel Graeser, Institut für Automatisierungstechnik (IAT), Universitty Bremen

Prof. Dr. Vilem Srovnal, VSB Technical University of Ostrava, Czeck Republic

Prof. Dr. Frantisek Zezulka, Brno University of Technology, Czech Republic

Prof. Dr. Axel Sikora, BerufsakademieLoerrach, Germany

M. C. Dr. Sorin Moga, ENST Bretagne, Brest, France

Prof. Dr. Eng.Dr.h.c.mult. Rolf Dietert Schraft, Director Fraunhofer Institute for Manufacturing Engineering and Automation Stuttgart, Germany

Prof. Dr. Eng. Werner Braatz, Fachhochschule Roseneheim, Germany

Prof. Dr. Eng. Werner Neddermeyer, Fachhochschule Gelsenkirchen, Germany

Prof. Dr. Ladislau Matekovits, Politecnico di Torino, Italy

Prof. Dr. Kuba Attila, University of Szeged, Institute of Informatics, Hungary

Prof. Dr. Palagyi Kalman, University of Szeged, Institute of Informatics, Hungary

Prof. Dr. Eng. Teodor Petrescu, "Politehnica" University of Bucharest, Romania

Prof. Dr. Eng. Adelaida Mateescu, "Politehnica" University of Bucharest, Romania

Acad. Adrian Rusu, "Politehnica" University of Bucharest

Prof. Dr. Eng. Silviu Ciochina, "Politehnica" University of Bucharest, Romania

Prof. Dr. Eng. Lucian Stanciu, "Politehnica" University of Bucharest, Romania

Prof. Dr. Eng. Brandusa Pantelimon, "Politehnica" University of Bucharest, Romania

Prof. Dr. Eng. Dimitrie Alexa, Technical University "Gheorghe Asachi", Iasi, Romania

Prof. Dr. Eng. Aurel Vlaicu, Technical University of Cluj-Napoca, Romania

Prof. Dr. Eng. Monica Borda, Technical University of Cluj-Napoca, Romania

Prof. Dr. Eng. Virgil Dobrota, Technical University of Cluj-Napoca, Romania

Prof. Dr. Eng. Serban Lungu, Technical University of Cluj-Napoca, Romania

Prof. Dr. Eng. Vladimir Rasvan, University of Craiova, Romania

Prof. Dr. Eng. Iuliu Szekely, "Transilvania" University of Brasov, Romania

Prof. Dr. Eng. Cornelia Gordan, University of Oradea, Romania

Dr. Eng. Stefan Victor Nicolaescu, National Communications Research Institute, Bucharest, Romania

Prof. Dr. Eng. Ioan Nafornita, "Politehnica" University of Timisoara

Prof. Dr. Eng. Corneliu Toma, "Politehnica" University of Timisoara

Prof. Dr. Eng. Miranda Nafornita, "Politehnica" University of Timisoara

Prof. Dr. Eng. Radu Vasiu, "Politehnica" University of Timisoara

Prof. Dr. Eng. Alexandru Isar, "Politehnica" University of Timisoara

Prof. Dr. Eng. Viorel Popescu, "Politehnica" University of Timisoara

Prof. Dr. Eng. Virgil Tiponut, "Politehnica" University of Timisoara

Prof. Dr. Eng. Horia Carstea, "Politehnica" University of Timisoara

Prof. Dr. Eng. Alimpie Ignea, "Politehnica" University of Timisoara

Prof. Dusan Popov, "Politehnica" University of Timisoara

Prof. Dr. Eng. Andrei Campeanu, "Politehnica" University of Timisoara

Prof. Dr. Eng. Liviu Toma, "Politehnica" University of Timisoara

Prof. Dr. Eng. Aurel Gontean, "Politehnica" University of Timisoara

DEAN'S WELCOME SPEECH

Dear colleagues,

*I wish you a warm welcome to the seventh edition of the "**Symposium of Electronics and Telecommunications – ETc 2006**".*
We started the first edition in 1994, as a national event, with Proceedings printed in Romanian. With each edition, every second year, we saw a continuous increase in both quantity and quality of papers, submitted by well known researchers from universities and industry, from Romania and abroad. The Symposium became quickly an international event, having papers published in English, in our Scientific Bulletin of the Politehnica University.
During the last years, the Bulletin and the Symposium have been supported by an international reviewers committee, with increased expertise and exigency. As a result, for the present edition, the committee accepted only 89 papers for presentation and publication, out of 122 submitted papers, resulting a rejection ratio of 27 %. I wish to express my gratitude to the members of the International Scientific Committee for the hard work and the time they dedicated to the revision of manuscripts.
Our Organizing Committee is proud to have the opportunity to publish such highly ranked papers in two dedicated volumes of the Scientific Bulletin, and congratulates the authors for their success.
Besides the scientific and informative value of the published volumes, the next purpose of our Symposium is to be, for all participants, a forum for exchange of ideas and socialization, for emphasizing the feeling of belonging to a highly specialized and expert community.
*In order to offer such a frame for the most dynamic category of researchers – the PhD students, we launched, in 2005, the "**Doctor ETc**" conference, dedicated to PhD Students in Electronics and Telecommunications, as a national event, with selected papers published in the Scientific Bulletin. This conference will be organized every second year, alternatively with the ETc Symposium.*
The present edition of the Symposium, in addition to the accepted papers, is honored by three keynote speeches, delivered by well known personalities from the research and economic world: Dan Bedros, CEO of Alcatel Network Systems, Romania, Dr. Christian Baier-Welt form Siemens VDO, and Prof. Dr. Rolf Dieter Schraft, from IPA Stuttgart, Fraunhofer Institute, Germany.
This symposium would not have been possible without the gracious help of our sponsors, to whom we express our gratitude.
I am also honored to express my thanks to you, all the participants, for attending the Symposium, to wish you a successful and profitable audience through the sessions, and a nice stay in Timisoara. I am looking forward of meeting you again in 2007 (Doctor ETc '07) and in 2008 (Symposium ETc '08).

Chairman of Symposium ETc '06,
Dean, Prof. Dr. Eng. Marius Otesteanu

Timisoara, September 21st, 2006

Buletinul Ştiinţific al Universităţii "Politehnica" din Timişoara

Seria ELECTRONICĂ şi TELECOMUNICAŢII
TRANSACTIONS on ELECTRONICS and COMMUNICATIONS

Tom 51(65), Fascicola 2, 2006

Table of Contents

Buletinul Ştiinţific al Universităţii "Politehnica" din Timişoara

Seria ELECTRONICĂ şi TELECOMUNICAŢII
TRANSACTIONS on ELECTRONICS and COMMUNICATIONS

Tom 51(65), Fascicola 2, 2006

A New Algorithm for Determining the Coefficients in B-spline Interpolation

Liliana Stoica[1]

Abstract – This algorithm is one of the methods that use spline functions for interpolation. In the context of general interpolation the coefficients are calculated using the values of the function and function's derivatives in the knots. Compared with another known algorithm, in this case is not necessary to perform the signal extension. But appear another problem: how to calculate the values for the derived function. Three methods are presented to resolve this. All the methods were applied for several input signals. From the practical results were made some conclusions.
Keywords: interpolation, B-spline functions, divided differences

I. INTRODUCTION

In this world of speed and high performances the interpolation problem remains on actuality. The traditional methods are improved and always are searched new ways to obtain better results with minimum costs. In this paper is presented a new algorithm for determine the B-spline coefficients in the generalized interpolation approach. All started with an algorithm presented in the specialty literature that has some disadvantages.

In Section II are presented the concepts of general and traditional interpolation and an algorithm for B-spline interpolation that use a modern technique. This algorithm was implemented and several observations were made [4]. To eliminate some disadvantages were searched another improved algorithms. In Section III are calculated the initial coefficients for interpolation using the properties of the spline functions: polynomial on short intervals, continuous and differentiable. The coefficients are determined from the input samples and from the derived function values in the knots. This method eliminates the signal extension necessary in the other algorithm. Section IV presents the new algorithm for calculating all the coefficients. This algorithm is based also on the derived function in the knots. The problem is to calculate those values. For that are presented three methods. The practical results of implementing all tree methods are discussed in Section V. There is made also a comparison with the Unser's algorithm results.

II. INTERPOLATION

A. Traditional Interpolation

Consider $y = \{y(k)\}$, k = 0,N-1 a set of discrete data, regularly sampled. To find the interpolated value $f(x)$ it is necessary to calculate:

$$f(x) = \sum_{k \in Z} y(k)\varphi_i(x-k) \qquad (1)$$

This is the traditional method to perform the interpolation: using the input data and the basis function values $\varphi_i(x-k)$ that give the sample weights.

B. Generalized Interpolation

Another way to perform the interpolation is to use the generalized formulation [5]:

$$f(x) = \sum_{k \in Z} c(k)\varphi(x-k) \qquad (2)$$

In this case the interpolated values are obtained from the coefficients $c(k)$ and not directly from the sample values $y(k)$. This method requires two different steps: determining the coefficients from the input data and calculate the interpolated values with those coefficients. It can be considered that the traditional interpolation is a particular case for $c(k) = y(k)$.

C. Unser's B-spline Interpolation Algorithm

The spline functions were used from a long time in problems of traditional interpolation. These are polynomial functions of degree n on adjacent intervals connected in the knots. The function and his derived up to n-1 order are continuous. These properties make the spline functions easy and convenient to use. For performing high-quality interpolation are often used the cubic spline function (n=3). In the traditional

[1] "Politehnica" University of Timisoara, Faculty of Electronics and Telecomunication, Bd. V. Pârvan No. 2, 300223 Timişoara, e-mail: liliana.stoica@etc.upt.ro

manner the spline interpolation is performed by matrix algebra methods and there is necessary a great amount of operations.

Another approach is to use digital filtering techniques. Michael Unser and his team developed an algorithm that uses digital filters for interpolation [6], [7], [8]. For the cubic B-spline function β^3 (x) in (3) it is defined the discrete B-spline function b_1^3(k) and the direct B-spline filter (4).

$$\beta^3(x) = \begin{cases} 2/3 - |x|^2 + |x|^3/2, & 0 \le |x| < 1 \\ (2-|x|)^3/6, & 1 \le |x| < 2 \\ 0, & 2 \le |x| \end{cases} \quad (3)$$

$$(b_1^3)^{-1}(k) \quad \leftrightarrow \quad [B_1^3(z)]^{-1} = \frac{6}{z+4+z^{-1}} \quad (4)$$

Applying this filter to the input signal are obtained the spline coefficients $c(k)$. The operation is called "direct B-spline transform". The interpolated function f(x/m) by a factor m, denoted $f_m^n(x)$ will be:

$$f_m^n(x) = \sum_{k \in Z} c(k) b_m^n(x - km) \quad (5)$$

This operation is called "indirect B-spline transform" and it is implemented also by digital filtering [6], [8].

For calculating the coefficients the direct B-spline filter is implemented by 2 filters: first a causal filter and the second anti-causal. The recursive algorithm demands some initial conditions. Is performed the signal extension by mirroring and they are taken a finite number of samples. The initial conditions introduce some side errors for the coefficients [4]. Those errors are transmitted in the interpolated signal and could have great importance especially if the input signal contains a small number of samples.

III. NEW INITIAL B-SPLINE COEFFICIENTS

To perform the spline interpolation in the traditional manner are used the known input samples and some values of the derived function. From this idea, to determine the new initial coefficients there were evaluated also the derivatives for the input function.

Consider f(x) an approximation for the cubic spline function that pass trough all the input values: f(k) = y(k), k = 0, N-1. In the knots f(k) represent the convolution between the coefficients' string and the cubic B-spline function (2). The relation involving the function and the coefficients c(k) can be write:

$$6f(k) = 4\,c(k) + c(k-1) + c(k+1) \quad (6)$$

The cubic B-spline function derivatives of first and second order are analyzed. From these ones are determined the relations between the f(k) derivatives and the coefficients:

$$f'(k) = 0\,c(k) - \tfrac{1}{2}\,c(k-1) + \tfrac{1}{2}\,c(k+1) \quad (7)$$

$$f''(k) = -2\,c(k) + c(k-1) + c(k+1) \quad (8)$$

The formulas (6), (7) and (8) are evaluated for k=2 to determine the initial values. The first 3 coefficients can be obtained by:

$$c(2) = f(2) - f''(2)/6 \quad (9)$$

$$c(0) = c(2) - 2f'(1) \quad (10)$$

$$c(1) = \frac{6f(1) - c(0) - c(2)}{4} \quad (11)$$

Compared with the Unser's algorithm, in this new approach is not necessary to perform the signal extension. But it has to establish a way to determine the values for the function derivatives of order one and two. These values must be obtained by numerical methods only from the input samples. The problem is to calculate $f'(1)$ and $f''(2)$ from the known signal values. The interpolation function is a B-spline (piecewise polynomial), so we can approximate f (k) by a polynomial function on short intervals. With this polynomial and his derivatives we calculate the values for the first 3 coefficients.

IV. A NEW ALGORITM BASED ON NUMERICAL DIFERENTIATION

With 3 initial values calculated it can be established an algorithm to determine the other coefficients. From the relation (7) it can be established a general formulation in every knot:

$$c(k+1) - c(k-1) = 2f'(k) \quad (12)$$

The algorithm supposes to use the function derivatives and to impose their values. This type of interpolation is called Hermite interpolation.

In the knots the values of the function f (k) must be equal to the input data samples:

$$f(k) = y(k) \text{ for } k = 0, 1, \ldots, N-1 \quad (13)$$

Dealing with discrete dates, now the problem it is to perform the numerical differentiation. There are discussed 3 methods for calculating those.

A. The First Method

It is used the classical definition for the divided differences [1],[2]:

$$f'(k) = \frac{f(k+1) - f(k)}{(k+1) - k} \quad (14)$$

The divided differences of order 2:

$$f''(k) = \frac{f(k+2) - 2f(k+1) + f(k)}{(k+2) - k} \quad (15)$$

In this case the algorithm for calculating the coefficients for the input signal y(k) became:

$$c(k+1) - c(k-1) = 2[y(k+1) - y(k)] \quad (16)$$

B. The Second Method

Stanasila [3] defines the next divided differences:
- the divided differences at left:

$$f'(k) \cong \frac{f(k) - f(k-h)}{h} \quad (17)$$

- the divided differences at right:

$$f'(k) \cong \frac{f(k+h) - f(k)}{h} \quad (18)$$

- by averaging it is obtained:

$$f'(k) \cong \frac{f(k+h) - f(k-h)}{2h} \quad (19)$$

$$f''(k) \cong \frac{f(k+h) - 2f(k) + f(k-h)}{h^2} \quad (20)$$

The same relation can be found by calculating the central derivative for a polynomial function that goes through 3 points.
In this case the initial values are:

$$c(2) = y(2) - (y(3) - 2y(2) + y(1))/6$$
$$c(0) = c(2) - y(2) + y(0)$$
$$c(1) = \frac{6y(1) - c(0) - c(2)}{4}$$

For any k value the iterative relation for calculating the coefficients became:

$$c(k+1) - c(k-1) = y(k+1) - y(k-1) \quad (21)$$

As it can be observed any differences between 2 coefficients $c(k+h)$ and $c(k)$ depends of the samples values in k+h and k points only.

C. The Third Method

The convergence properties can be improved by stronger conditions of continuity. It means that the interpolation function is continuous and his derivatives up to the fourth order are continuous $f(x) \in C^4$. This is demonstrated by a theorem in [1]. So we consider $f(x)$ a polynomial function of 4 degree:

$$f(x) = a + b\,x + d\,x^2 + e\,x^3 + g\,x^4 \quad (22)$$

The function is piecewise polynomial, so it can be analyzed on short intervals. The function and the function derivatives of order 1 and 2 have been evaluated on the interval [0;4] and are obtained the next relations:

$$f'(1) = \frac{-3f(0) - 10f(1) + 18f(2) + f(4)}{12}$$
$$f''(2) = \frac{-(f(0) + f(4)) + 16(f(1) + f(3)) - 30f(2)}{12}$$

The general formulation for the firs derivative is:

$$f'(k) = \frac{f(k-2) - 8f(k-1) + 8f(k+1) - f(k+2)}{12} \quad (23)$$

For $f(k) = y(k)$, the algorithm for calculating the coefficients became:

$$c(k+1) - c(k-1) = \frac{y(k-2) - 8y(k-1) + 8y(k+1) - y(k+2)}{6} \quad (24)$$

It has to demonstrate the algorithm convergence. For that are take into consideration a finite number of successive iterations:

$$c(3) - c(1) = [y(0) - 8y(1) + 8y(3) - y(4)]/6$$
$$c(5) - c(3) = [y(2) - 8y(3) + 8y(5) - y(6)]/6$$
$$c(7) - c(5) = [y(4) - 8y(5) + 8y(7) - y(8)]/6$$
$$\ldots\ldots$$
$$c(k) - c(k-2) = [y(k-3) - 8y(k-2) + 8y(k) - y(k+1)]/6$$
$$\Rightarrow \quad c(k) - c(1) = [8y(k) - y(k-1) - y(k+1) + 8y(1) + y(0) + y(2)]/6$$
$$\Rightarrow \quad c(k) - c(1) = \{y(k) - [y(k+1) - 2y(k) + y(k-1)]/6\} - \{y(1) - [y(2) - 2y(1) + y(0)]/6\} \quad (25)$$

Any differences $c(k) - c(1)$ does not depending on intermediary values, but only the ones related to $y(k)$ and $y(1)$. It can be observed that the expressions in square brackets in (25) represent the divided differences of second order by Stanasila's definition determined in the k and 1 points [3]. The values for $c(k)$ and $c(1)$ are not bounded by intermediary samples of the function, so this function can be arbitrary between k and 1. In this case the method could be generalized and used also for discontinuous signals.

V. COMPARATIVE RESULTS

All tree methods were implemented to determine the coefficients and the algorithms were applied for several known signals. Some significant results are gone be presented along. The input signal were $y(k) = \sin(2\pi k/M)$ or $y(k) = \cos(2\pi k/M)$, k = 0, N-1 for different values of M and N. Were analyzed situations for diverse sampling frequencies (different values for M). For the periodic signals the input string has a small number of samples (M=12 and N=13) corresponding to one period or an increased number of samples equivalent to more than two periods.
For the same input string were calculated the coefficients $c(k)$ using each of the three methods and it was performed the interpolation in every case. The interpolated values are obtained by the same method

like in the Unser's algorithm using the equation (5). It was performed the interpolation by factor m=2.

The results are comparative for the sine and cosine signals. If M=12 the input signal has a small number of samples. Applying the *A* method for calculating the coefficients the interpolation errors are of 10^{-1} order. Almost all the interpolated values are influenced by errors of this range. For the same input string the interpolation errors are 10^{-2} in case of *B* and 10^{-3} for *C* methods.

For a signal with a greater number of samples per period (M=120) the differences between the tree methods are significant. The interpolation errors are 10^{-3} with the classical definition for the divided differences. Using the *B* method these errors became 10^{-4}. If the derived function values are calculated by the polynomial of degree 4 then the interpolation errors are decreasing to 10^{-7}. By increasing the sampling frequency for the input signal are reduced the interpolation errors.

For $y(k)=\cos(2\pi k/M)$ being the input signal, some results are presented in Table 1. In two cases: M=12 and M=120 are given the interpolation errors for some distinctive points α on the function characteristic.

Table1. Interpolation errors for $y(k)=\cos(2\pi k/M)$.

α	M	*A* method	*B* method	*C* method
$\pi/3$	12	-0.05502116	-0.00817301	0.00024929
	120	-0.00068398	-0.00022684	-0.00000005
$\pi/2$	12	-0.03867513	-0.03867513	-0.00099717
	120	-0.00091239	-0.00045525	-0.00000015
π	12	-0.12200846	-0.06630823	-0.00274223
	120	-0.00136921	-0.00091207	-0.00000040

As it can be seen the *A* method has results that are not too good. The improved method *B* can offer acceptable errors for some applications. It has the advantage of simplicity and requires a relative small number of operations. The algorithm has better results by using the polynomial function of degree 4. But in this case are necessary additional operations. Decreasing errors is possible by increasing the computational costs. The operations are not complicated and they don't take much time for calculating in applications that require better results.

The results can be compared with the ones obtained with the Unser's algorithm where for determining the coefficients it is applied the direct B-spline transform. The errors in this case are greater at the beginning and the end of the data string compared with the data in the middle [4]. This is due to the finite number of samples used at the initialization procedure for determining the coefficients. For M=12 the side values present errors of 10^{-1} order and the others have interpolation errors of 10^{-3}. For M=120 the interpolation errors are 10^{-8} up to 10^{-6} at the beginning and at the end.

The new algorithm has the advantage that the errors introduced by the method of determining the coefficients are the same for all the interpolated samples.

In all studied cases the firsts and lasts 2 interpolated values are influenced by greater errors. These are introduced by the interpolation method. Every interpolated value is obtained from the coefficient corresponding to the current point and some anterior and posterior coefficients (convolution in (5)). Some of these (c(-1) and c(N) for example) are not known and considered zero when calculate the interpolated values on the sides of the string. This problem appears also at the Unser's algorithm. It can be resolved and it will be discussed in to a further paper.

VI. CONCLUSIONS

The algorithm use known techniques combined in new manner. The main advantage of this one compared with the Unser's algorithm is that the input signal don't have to be extended to establishes the initial conditions. The coefficients are calculated using the input samples and the values for the first derivative of the input function in the sampling points. The problem was to determine these values only from the input data. One of the presented methods (the *C* method) offers very good results for the interpolation. The function is approximated by a polynomial of forth order. From this is established the recursion formula for calculate the coefficients.

The algorithm offers simplicity of implementation. It was applied on input signals that are continuous for different sampling frequencies. The presented methods will be tested on other types of signals to observe if the results are as good as the presented ones.

ACKNOWLEDGEMENTS

The author would like to address special thanks to Professor Eugen Pop for his patience, guidance and support.

REFERENCES

[1] Gh. Micula, *Functii Spline si aplicatii*, Editura Tehnica Publishing House, Bucuresti, 1978

[2] I. Gh. Sabac, *Matematici Speciale*, vol. 2, Editura Didactica si Pedagogica Publishing House, Bucuresti, 1965

[3] O. Stanasila, *Analiza Matematica*, Editura Didactica si Pedagogica Publishing House, Bucuresti, 1981

[4] L. Stoica., "Contributions in Recursive Filtering for B-Splines Interpolation in Signal Processing", *Proceedings of the International Symposium on Electronics and Telecommunications ETC 2006*, Timisoara, September 21-23, 2006

[5] P. Thevenaz, T. Blu, M. Unser, "Interpolation Revisited", *IEEE Transactions on Medical Imaging*, Vol. 19, No. 7, pp. 739-758, July 2000.

[6] M. Unser, "Splines: A Perfect Fit for Signal and Image Processing", *IEEE Signal Processing Magazine*, Vol. 16, No. 6, pp. 22-38, Nov. 1999.

[7] M. Unser, A. Aldroubi, M. Eden, "B-Spline Signal Processing: Part I - Theory", *IEEE Transactions on Signal Processing*, Vol. 41, No. 2, pp. 821-833, Feb. 1993.

[8] M. Unser, A. Aldroubi, M. Eden, "B-Spline Signal Processing: Part II – Efficient Design and Applications", *IEEE Transactions on Signal Processing*, Vol. 41, No. 2, pp. 834-848, Feb. 1993.

Buletinul Ştiinţific al Universităţii "Politehnica" din Timişoara

Seria ELECTRONICĂ şi TELECOMUNICAŢII
TRANSACTIONS on ELECTRONICS and COMMUNICATIONS

Tom 51(65), Fascicola 2, 2006

A New Linear Systolic Array for the VLSI Implementation of 2-D IDST

Doru Florin Chiper[1]

Abstract - In this paper a new linear VLSI array architecture for the VLSI implementation of the 2-D IDST based on a new systolic array algorithm is proposed. This new design approach uses a new efficient VLSI algorithm. It employs a new formulation of the inverse DST that is mapped on a linear systolic array. Using the proposed systolic array high computing speed is obtained with a low I/O cost. The proposed architecture is characterized by a small number of I/O channels located at the two extreme ends of the array together with a low I/O bandwidth that is independent of the transform length N. The topology of the proposed VLSI architecture is highly modular and regular and uses only local connections. Thus, it is well suited for a VLSI implementation

Keywords: Inverse discrete sine transform, systolic algorithms, systolic architectures

I. INTRODUCTION

The 2-D forward and inverse discrete sine transforms are important transform functions that are widely used in many signal and image processing applications. They are especially employed in image compression due to the fact that they behave very much like the statistically optimal Karhunen-Loeve transform (KLT). Thus, the forward and inverse 2-D DST and DCT represent the critical part in the implementation of JPEG compression [2].

The forward and inverse DST are computational intensive. So, in order to use them in real-time applications the development of application specific hardware is demanded.

In the literature there are presented several 2-D VLSI architectures [4-10]. Most of them use the row-column decomposition method. Some of them are using a direct method to compute forward or inverse 2-D DST or DCT [7-9].

Systolic arrays [11] are a good architectural paradigm to be used in real-time applications. They are also well suited for the VLSI implementation. The VLSI algorithms for forward and inverse DST have to be derived specifically. The way of data moving is very important in determination of the efficiency of a VLSI algorithm and of its implementation. Thus, the use of regular and modular computational structures with local data communications can lead to efficient VLSI implementation [12, 13] using the systolic array architectural paradigm. Thus, an efficient way to convert the inverse DCT into such structures can lead to optimal VLSI implementations

II. TWO DIMENSIONAL IDST ARCHITECTURE

The 2-D inverse DST (IDST) for a NxN pixel block can be defined as follows:

$$x(k,l)=\sum_{i=0}^{N-1}\sum_{j=0}^{N-1}X(i,j)\cdot\sin\left[(2i+1)\alpha\right]\cdot\sin\left[(2j+1)l\cdot\alpha\right] \quad (1)$$

where:

$$\alpha=\frac{\pi}{2N} \quad (2)$$

$x(k,l)\ \ (k,l=0,1,...,N-1)$ is the pixel data, $X(i,j)\ (i,j=0,1,...,N-1)$ is the transform coefficient .

In the literature there are presented several 2-D VLSI architectures for IDST. Most of them use the row-column decomposition method. Some of them are using a direct method to compute forward or inverse 2-D DCT or DST.

The row-column approach can de expressed in a matrix form as:

$$[x_N]\ =[S_N][X_N][S_N]^T \quad (3)$$

where $[S_N]$ is the 1-D N-point IDCT, with:

$$[S_N]_{i,j}=\begin{cases}1 & for\ i=0\\ \sin[(2i+1)j\cdot\alpha] & otherwise\end{cases} \quad (4)$$

Equation (4) can be computed by N N-point IDST along the rows of the input $[X_N]$, obtaining $[Y_N]=[X_N][Y_N]^T$, and followed by N N-point IDSTs along the columns of the matrix obtained from the row transformed $[x_N]=[S_N]\ [Y_N]$. It can be observed

[1] Facultatea de Electronică şi Telecomunicaţii, Bd. Carol I Nr. 11, 6600, Iasi.

that using the row-column decomposition method we have to compute two 1-D IDSTs one after the other.

This simple decomposition method reduces the computation complexity with a factor of 4.

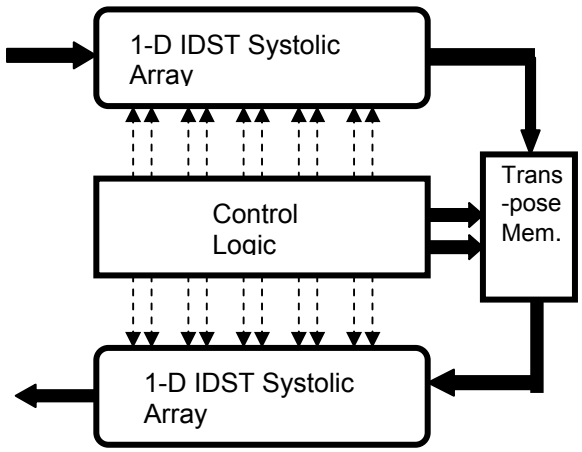


Fig.1. The linear systolic array for 2-D IDST computation

III. 1-D N-POINT INVERSE DST ARCHITECTURE

A. Systolic Algorithm for 1-D Inverse DST

The 1-D N-point inverse discrete sine transform IDST is defined as follows:

$$x(k)=\sum_{i=0}^{N-1} Y(i)\cdot \sin[(2k+1)i\cdot\alpha]; \qquad k=1,2,...,N \tag{5}$$

with $\alpha=\dfrac{\pi}{2N}$ (6)

In order to reformulate relation (5) as a circular correlation form we introduce some auxiliary sequences and use the proprieties of the Galois Field of indexes to appropriate permute the input and output sequences.

The output auxiliary sequence $\{T(k): k=1,2,...,N-1\}$ can be computed as follows:

$$T(k)=2T'(k) \tag{7}$$

The new auxiliary output sequence $\{T'(k): k=1,2,...,N-1\}$ can be computed as a circular correlation, if the transform length N is a prime number, as following:

$$T'(<g^k>_N)=\sum_{i=1}^{(N-1)/2}[(-1)^{\psi(k,i)}\cdot Y_C(<g^i>_N)+ +(-1)^{\psi(k,i+(N-1)/2)}\cdot Y_C(<g^{i+(N-1)/2}>)]\times \times\sin(<g^{i+k}>_N\times 2\alpha) \tag{8}$$

where $<x>_N$ denotes the result of x modulo N and

$$\psi(k,i)=\left\lfloor\frac{<g^k>_N\times<g^i>-<g^{i+k}>_N}{N}\right\rfloor \tag{9}$$

with $\lfloor x\rfloor$ the greater integer smaller the x and is called the floor function.

We have used the properties of the Galois Field of indexes to convert the computation of the auxiliary output sequence $\{T'(k): k=1,2,...,N-1\}$ as a circular correlation.

The auxiliary input sequence $\{x_C(i): i=1,2,...,N-1\}$ is defined as following:

$$Y_C(i)=Y(i)\cdot\cos(i\alpha) \tag{10}$$

Finally, the output sequence can be recursively computed using the auxiliary output sequence $\{T(k): k=1,2,...,N-1\}$ as:

$$x(k)=T(k)-x(k-1); \quad k=1,2,...,N-1 \tag{11}$$

$$x(0)=\sum_{i=1}^{N} Y_S(i) \tag{12}$$

with

$$Y_S(i)=Y(i)\cdot\sin(i\alpha) \tag{13}$$

B. An Example

To illustrate our approach, we will consider an example for 1-D IDST with the length N=11 and the primitive root g=2.

We can write (8) in matrix-vector product form as:

$$\begin{bmatrix}T'(2)\\T'(4)\\T'(8)\\T'(5)\\T'(10)\\T'(9)\\T'(7)\\T'(3)\\T'(6)\\T'(1)\end{bmatrix}=\begin{bmatrix}s(4)&s(8)&s(5)&s(10)&s(9)\\s(8)&s(5)&s(10)&s(9)&s(4)\\s(5)&s(10)&s(9)&s(4)&s(8)\\s(10)&s(9)&s(4)&s(8)&s(5)\\s(9)&s(4)&s(8)&s(5)&s(10)\\s(4)&s(8)&s(5)&s(10)&s(9)\\s(8)&s(5)&s10)&s(9)&s(4)\\s(5)&s(10)&s(9)&s(4)&s(8)\\s(10)&s(9)&s(4)&s(8)&s(5)\\s(9)&s(4)&s(8)&s(5)&s(10)\end{bmatrix}\times \times\begin{bmatrix}\pm[x_C(2)\pm x_C(9)]\\\pm[x_C(4)\pm x_C(7)]\\\pm[x_C(8)\pm x_C(3)]\\\pm[x_C(5)\pm x_C(6)]\\\pm[x_C(10)\pm x_C(1)]\end{bmatrix} \tag{13}$$

where we noted by $s(k)$ as $\sin(2k\alpha)$ and the sign of the items in relation (9) is given by the following matrix:

$$SIGN = \begin{bmatrix} 01 & 01 & 11 & 01 & 11 \\ 01 & 11 & 01 & 11 & 11 \\ 11 & 01 & 11 & 11 & 11 \\ 00 & 10 & 10 & 00 & 00 \\ 11 & 11 & 11 & 01 & 11 \\ 10 & 10 & 00 & 00 & 00 \\ 10 & 00 & 10 & 10 & 00 \\ 00 & 10 & 00 & 10 & 00 \\ 11 & 01 & 01 & 01 & 11 \\ 00 & 00 & 00 & 00 & 00 \end{bmatrix}$$

where:

- The first bit designates the sign before the brackets
- The second bit denotes the sign inside the brackets

where the "1" bit indicates the minus sign (the first bit) and the subtraction operation (the second one)

III. THE LINEAR SYSTOLIC ARRAY FOR 1-D IDST

Using the dependence-graph of equation (13) and the dependence-graph based synthesis procedure [14] we have obtained a linear systolic array. The hardware-core of this array is presented in figure 2. The function of the processing elements Pes is presented in figure 2b. In order to deal with the sign differences in equation (13) we have used the tag-control technique presented in [15].

Using the tag-control mechanism we can keep the I/O channels at the two extreme ends of the linear array, where the tag sequences t_c controls the loading of the input data into the array as shown in fig.2b. Using this mechanism we can control the content of the internal registers using only channels placed at one of the two ends of the array.

The pre-processing and post-processing stages realize the appropriate reorder of the auxiliary input and output sequences.

In the preprocessing stage we also compute the auxiliary input sequence $\{Y_C(i): i = 1,2,..., N-1\}$ and $\{Y_S(i): i = 1,2,..., N-1\}$. In the post-processing stage we also compute the auxiliary output sequences $\{T(k): k = 1,2,..., N-1\}$ and finally the output sequence using the equations (11), (12) respectively.

IV. PERFORMANCES AND COMPARISON

The average computation time is $(N-1)T_{cycle.}$ The number of multipliers is (N-1)/2+1 and the number of adders is (N-1)/2+2. Thus, low hardware and I/O costs can be obtained. We can easily obtain a high throughput using a two-level pipelining mechanism with low hardware and I/O costs.

In [16] a time-recursive structure is proposed. As compared with [16] the throughput is significantly increased using a two-level pipelining. The structure proposed in [16] did not allow a two-level pipelining due to its recursive nature.

In [17] and [18] the throughput can be also substantially increased using the two-level pipelining. These structures do not allow two level pipelining due to the data-path feedback.

As compared with [19] the throughput is also much increased when using a two-level pipelining. This is explained due to the presence of the feedback in RACs.

The proposed structured has also a low I/O cost. As compared with [20] the I/O cost is significantly lower. The I/O cost can significantly limit the speed performances due to so called I/O bottleneck.

V. CONCLUSION

In this paper a new VLSI architecture for the VLSI implementation of 2-D inverse discrete sine transform is presented. It has some appealing features as a low I/O cost and high speed performances. It employs a new VLSI algorithm that efficiently uses the advantages of the circular correlation computational structure as high degree of parallelism, small computational complexity and local data communications. The 2-D IDST VLSI architecture is obtained using two linear systolic arrays connected in a serial manner. The proposed VLSI architecture is highly regular and modular and has local interconnections. It has also a small number of I/O channels placed at the two extreme ends of the array with a reduced I/O bandwidth. Thus it is well suited for a VLSI implementation.

REFERENCES

[1] N. Ahmad, T.Natarajan, and K.R. Rao, "Discrete Cosine transform," IEEE Transactions on Computers, vol.C-23, pp.90-94, 1974.

[2] W. Pennebaker, J/ Michell. JPEG Still Image Data Compression Standard. Van Nostrand Reinhold, USA, 1992.

[3] M.Kovac, N. Ranganathan, "JAGUAR: A Fully Pipelined VLSI Architecture for JPEG Image Compression Standard," Proc. of IEEE, vol.83, No.2, 1995, pp.247-258.

[4] A. Madisseti, A. Wilson Jr., "A 100 Mhz 2-D 8x8 DCT/IDCT Processor for HDTV Applications," IEEE Transaction on Circuits and Systems for Video Technology, vol.5, no.2, pp. 158-165, Apr. 1995.

[5] S. Uramoto, et al. , “A 100 Mhz 2-D discrete cosine transform processor,” IEEE Solid-State Circuits, 1992, vol.27, No.4, pp.492-498.

[6] M.T. Sun, T.C. Chen, and A.M. Gottlieb, “VLSI Implementation of 16x16 discrete cosine transform,” IEEE Trans. on Circuits and Systems, 1989, vol.36, no.4, pp.610-617.

[7] C. Wang, C. Chen, “High-Throughput VLSI Architectures for the 1-D and 2-D Discrete Cosine Transform,” IEEE Transaction on Circuits and Systems for Video Technology, vol.5, no.1, pp. 31-40, Febr. 1995

[8] Y. Lee, T. Chen, I. Chen, M. Chen, C.Ku, ”A Cost-Effective Architecture for 8x8 Two-Dimensional DCT/IDCT Using Direct Method,” IEEE Transaction on Circuits and Systems for Video Technology, vol.7, no.3, pp. 459-467, June 1997

[9] H.Lim, V.Piuri, E.E.Swartzlander, ”A Serial-Parallel Architecture for Two-Dimensional Discrete Cosine and Inverse Cosine Transforms,” IEEE Transactions on Computers, vol.49,No.12, pp.1297-1309, Dec. 2000.

[10] S. Bique, “New characterizations of 2D Discrete Cosine transform,” IEEE Trans. on Computers, vol.54, no.9, Sept. 2005.

[11] H.T. Kung, “Why systolic architectures?,” Computer Magazine, 1982, vol.15, no.1, pp.37-45.

[12] C.M. Rader, “Discrete Fourier transform when the number of data samples is prime,” Proc. IEEE, vol.56, pp.1107-1108, June 1968.

[13] J.I. Guo, C.M. Liu and C.W. Jen, “A New Array Architecture for Prime-Length Discrete Cosine Transform,” IEEE Transactions on Signal Processing, vol. SP-41,no.1, Jan. 1993.

[14] S.Y. Kung, VLSI Array Processors. NJ. Prentice Hall, 1988.

[15] C.W. Jen and H.Y. Hsu, “The design of systolic arrays with tag input,” Proc. IEEE Int. Symp. on Circuits and Systems, 1988.

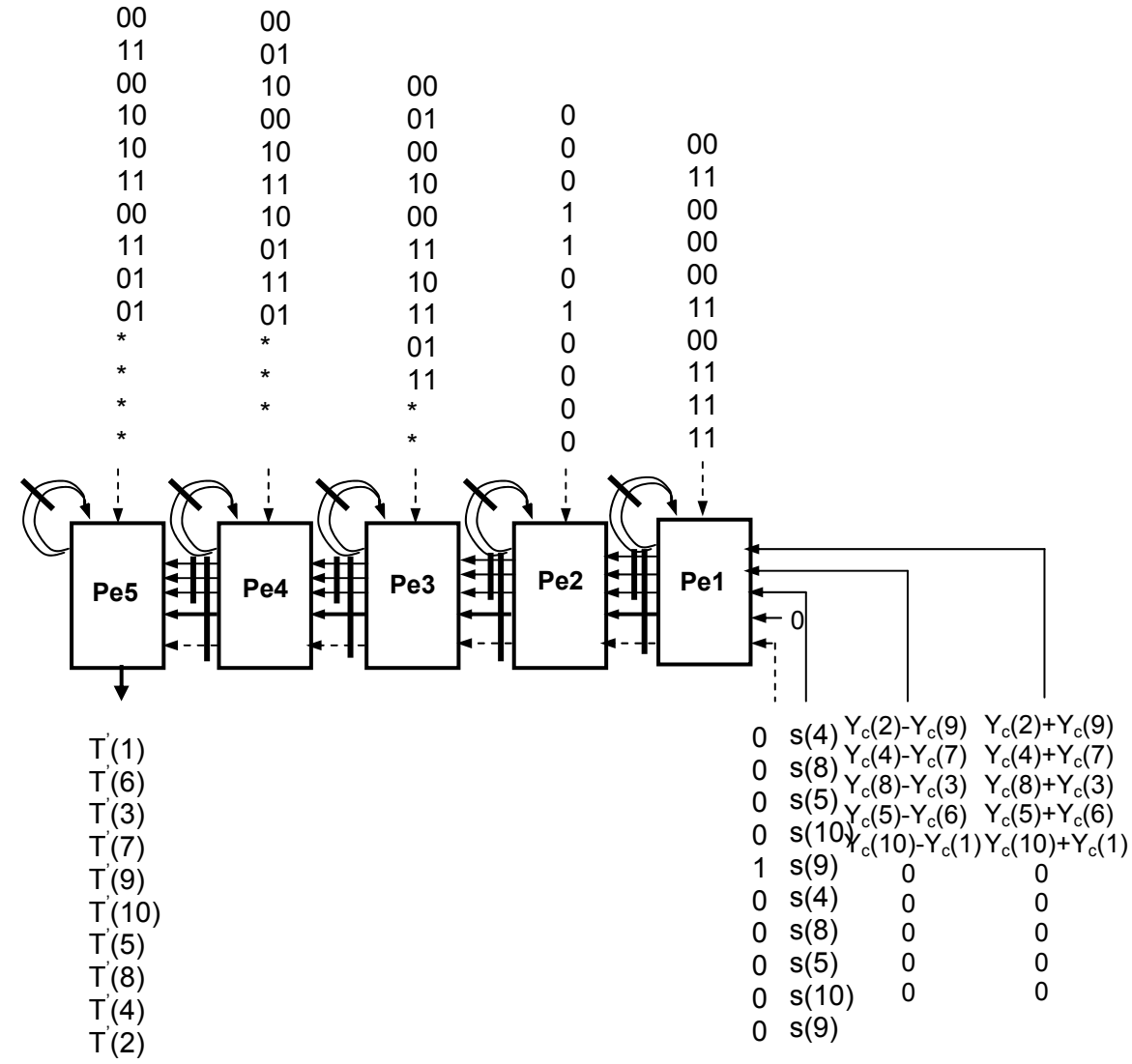
[16] J. F. Yang and C-P. Fang, "Compact recursive structures for discrete cosine transform," *IEEE Trans. on Circuits and Systems-II,* vol. 47, pp. 314-321, Apr. 2000.

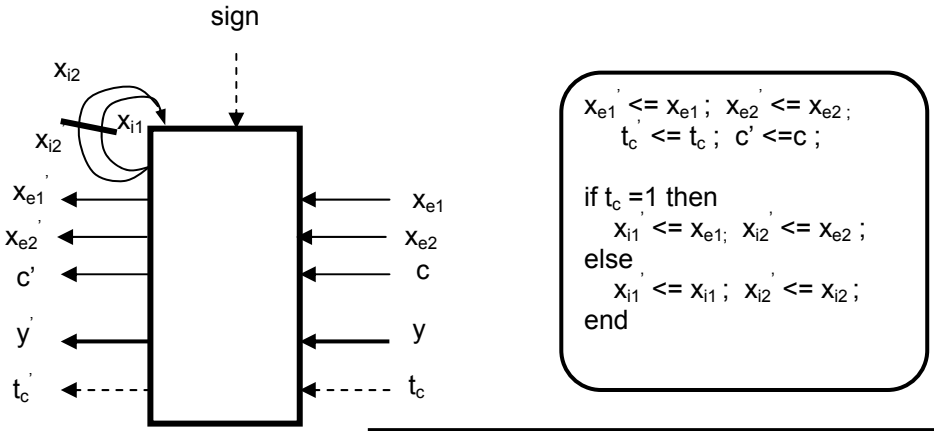
[17] W. H. Fang and M. L. Wu, "An efficient unified systolic architecture for the computation of discrete trigonometric transforms," in *Proc. IEEE Symp. on Circuits and Systems*, vol. 3, 1997, pp. 2092-2095.

[18] W. H. Fang and M-L. Wu, "Unified fully-pipelined implementations of one- and two-dimensional real discrete trigonometric transforms*," IEICE Trans. on Fund. Electron. Commun. Comput. Sci*., vol. E82-A, no. 10, pp. 2219-2230, Oct. 1999.

[19] J. Guo, C. Chen, and C-W. Jen, "Unified array architecture for DCT/DST and their inverses," *Electron. Letters*, vol. 31, no. 21, pp. 1811-1812, 1995.

[20] S.B.Pan and R-H. Park, "Unified systolic arrays for computation of DCT/DST/DHT*," IEEE Trans. on Circuits and Systems for Video Technology*, vol. 7, no. 2, pp. 413-419, April 1997.






```
xe1' <= xe1 ; xe2' <= xe2 ;
    tc' <= tc ; c' <=c ;

if tc =1 then
    xi1' <= xe1; xi2' <= xe2 ;
else
    xi1' <= xi1 ; xi2' <= xi2 ;
end
```

t_c \ sign	00	01	10	11
0	$y+x_{i1}*c$	$y+x_{i2}*c$	$y-x_{i1}*c$	$y-x_{i2}*c$
1	$y+x_{e1}*c$	$y+x_{e2}*c$	$y-x_{e1}*c$	$y-x_{e2}*c$

Fig.2. (a) The VLSI array architecture of the hardware-core of 1D-IDST
(b) The function of the processing elements PEs

Buletinul Ştiinţific al Universităţii "Politehnica" din Timişoara

Seria ELECTRONICĂ şi TELECOMUNICAŢII
TRANSACTIONS on ELECTRONICS and COMMUNICATIONS

Tom 51(65), Fascicola 2, 2006

A New Quasi Shift Invariant Non-Redundant Complex Wavelet Transform

Ioana Adam[1], Marius Oltean[2], Mircea Bora[1]

Abstract

The property of shift-invariance associated with the property of good directional selectivity are important for the application of a wavelet transform in many fields of image processing. Unfortunately, the classical discrete wavelet transform is shift-variant. All modified algorithms proposed in the literature for the computation of a shift invariant transform are less or more redundant and difficult to implement, and consequently thorny to use in signal processing applications. In this paper, we propose a new, quasi shift-invariant wavelet transform, without redundancy and easy to implement.

1. INTRODUCTION

A wavelet transform (WT), is shift-sensitive if an input signal shift causes an unpredictable change of the transform coefficients.

Shift-sensitivity is an undesirable property because it implies the impossibility to distinguish between wavelet transform coefficients corresponding to input signal shifts.

The shift-sensitivity of the Discrete Wavelet Transform (DWT) is generated by the down-samplers used for its computation.

The property of shift-invariance associated with the property of good directional selectivity are important for the application of a wavelet transform in many fields of image processing including denoising, de-blurring, super-resolution, watermarking, segmentation and classification.

In the next section, several quasi shift-invariant WTs, proposed in the literature are presented. Our transform is introduced and explained in section 3. Simulation results are presented in section 4, in order to illustrate the degree of shift invariance of the proposed transform. In the final section, a few conclusions are exposed and future possible research directions on the subject are indicated.

2. TYPES OF WAVELET TRANSFORMS

There are in the literature some wavelet transforms which are shift-invariant or quasi shift-invariant. In the following, some of them are presented.

A. UDWT

Since down-samplers in the DWT implementation create shift-sensitivity, Mallat [1], Beylkin [2], Coifman and Donoho [3] and Guo [4], devised the un-decimated DWT (UDWT), which is a wavelet transform without down-samplers. Although the UDWT is shift-insensitive, it has high redundancy, caused by the absence of down-samplers. Unfortunately, the high redundancy incurs a massive storage requirement that makes the UDWT inappropriate for most signal processing applications. Another disadvantage of the UDWT comes from the fact that it requires the implementation of a large number of different filters.

B. Shift Invariant Discrete Wavelet Transform

Lang, Guo, Odegard, Burrus and Welles [4] have proposed a new shift-invariant but very redundant wavelet transform, named Shift Invariant Discrete Wavelet Transform, SIDWT. Their proposition is based on a translation invariant algorithm proposed by Coifman and Donoho [3]. The computation of this transform implies the consideration of all circular shifts of the input signal. After the computation of the DWT of every shifted version of the signal, this method requires the shifting back (or unshifting) and averaging over all results obtained.

C. Cycle Spinning

The method introduced by Coifman and Donoho in [3] and called Cycle Spinning (CS) was conceived to suppress the artefacts in the neighbourhood of

[1] Ph. D Students, [2] Teaching Assistant, Faculty of Electronics and Telecommunications, Communications Departement, Bd. V. Pârvan Nr. 2, 300223 Timişoara, e-mail ioana.adam@etc.upt.ro

discontinuities introduced by the classical DWT, and it implies the rejection of the translation dependence. For a range of shifts, data (time samples of a signal) is shifted (right or left as the case may be), the DWT of shifted data is computed, and than the result is un-shifted. Doing this for a range of shifts, and averaging the several results so obtained, a quasi shift-invariant discrete wavelet transform is obtained. The degree of redundancy of this transform is proportional to the number of shifts of the input signal produced. Cycle spinning over the range of all circular shifts of the input signal is equivalent to SIDWT.

D. Dual Tree Complex Wavelet Transform

Abry [5], first demonstrated that approximate shiftability is possible for the DWT with a small, fixed amount of transform redundancy. He designed a pair of real wavelets such that one is approximately the Hilbert transform of the other. This wavelet pair defines a complex wavelet transform (CWT). For explaining that such a transform is complex, consider the pair of DWT trees associated with the wavelet pair already mentioned. A complex wavelet coefficient is obtained by interpreting the wavelet coefficient from one DWT tree as being its real part, whereas the corresponding coefficient from the other tree is interpreted as its imaginary part. This transform is represented in figure 1.

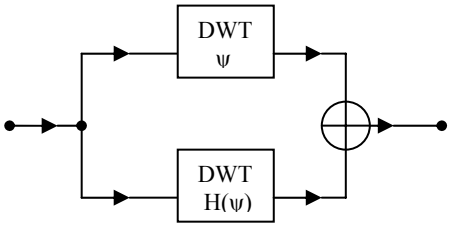


Figure 1. Abry's CWT.

Kingsbury [6] developed the dual tree complex wavelet transform (DTCWT), which is a quadrature pair of DWT trees, similar to Abry's wavelet transform (see figure 1). The DTCWT coefficients may be interpreted as arising from the DWT associated with a quasi-analytic wavelet. Both DTCWT and Abry's transform are invertible and quasi shift-invariant; however the design of these quadrature wavelet pairs is quite complicated and it can be done only through approximations.

E. Mapping-based Complex Wavelet Transform

Fernandes, van Spaendonck and Burrus have introduced, in [7], a two-stage mapping-based complex wavelet transform (MBCWT) that consists of a mapping onto a complex function space followed by a DWT of the complex mapping computation. The authors of this article have observed that the DTCWT coefficients admit also another interpretation: they may be interpreted as the coefficients of a DWT applied to a complex signal associated with the input signal. The complex signal is defined as the Hardy-space image of the input signal. As the Hardy-space mapping of a signal is impossible to compute, they have defined a new function space called the Softy-space, which is an approximation to Hardy-space.

The advantages of this method are:

- controllable redundancy of the mapping stage that offers a balance between the degree of shift sensitivity and the transform redundancy;
- the possibility to use any mother wavelet for the computation of the DWT in the transform implementation, which provides flexibility to this transform.

3. ANALYTIC DISCRETE WAVELET TRANSFORM

In this paper, we propose a new complex wavelet transform, similar to the DTCWT but easier to implement. It involves computing a single DWT but, instead of applying it to the original signal we apply it to the analytical signal associated with our input signal. The analytical signal associated with the signal x is defined as $x_a=x+iH\{x\}$, where $H\{x\}$ denotes the Hilbert transform of the input signal.

In the following, this transform will be called analytic discrete wavelet transform, ADWT. The equivalence between the DTCWT and the ADWT is illustrated in figure 2.

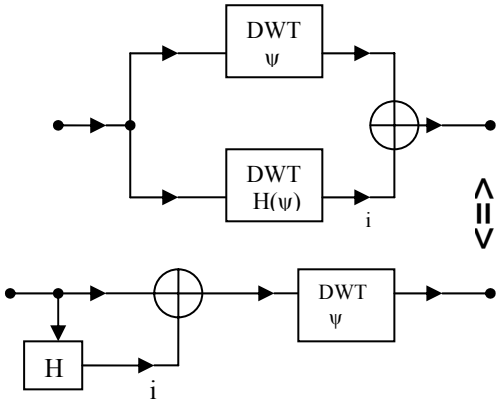


Figure 2. The equivalence between the DTCWT (top) and the ADWT (bottom).

In [8], Simoncelli has defined a new measure of the shift-invariance, called "shiftability". According to their definition, a transform is shiftable if and only if any subband energy of the transform is invariant under input-signal shifts. Although weaker than shift invariance, shiftability is important for applications because it is equivalent to interpolability, which is a property ensuring the preservation of transform-subband energy under input-signal shifts.

4. SIMULATIONS

In order to evaluate the shift-invariance performance of our transform, we introduced a new criterion: the degree of shift invariance. In order to calculate this measure, we calculate the energies of every set of detail coefficients (at different decomposition levels) and of the approximation coefficients, corresponding to a certain delay (shift) of the input signal samples. This way, we obtain a sequence of energies at each decomposition level, each sample of this sequence corresponding to a different shift. Then the mean *m* and the standard deviation *d* of every energy sequence are computed. Our degree of invariance is defined as:

$$Grad = 1 - d/m \qquad (1)$$

We perform the normalization with respect to the mean of the energy sequence because we want the values of the degree of invariance to be within the interval [0, 1], for better interpreting it.

If the transform is shift-invariant, then the value of its degree of invariance is 1 because the standard deviation of the energy sequence is zero in this case. The reciprocity is not guaranteed. There are quasi shift invariant wavelet transforms with the degree of shift-invariance equal to 1 that are not perfectly shift-invariant. However, generally, when the transform is not shift-invariant the value of this degree of invariance is smaller than 1. This observation is also sustained by experimental work.

We consider that the degree of shift invariance is an objective way of analysing the shift invariance of a transform.

In the simulations purpose, we used as input signal a unitary step, like in [6]. In fact, 16 different unitary steps were used. They were generated one from another by delaying with a sample. Each unitary step is composed of 1024 samples. The number of iterations used for the computation of the DWT was 3. We repeated the simulations for several mother wavelets commonly used in the literature (Daubechies, Symmlet and Coiflet).

In the first set of simulations we have compared the degree of shift invariance of our transform with the degree of shift invariance of the DWT.

In the second set of simulations we have compared the degree of shift invariance of our transform with the degree of shift invariance of the CS with a various number of cycle spins and for a variety of spinning steps (a spinning step is the number of samples the signal is shifted once).

In table 1 we present a comparison between our transform and the DWT. This comparison is based on the values of the degree of shift invariance calculated for the approximation coefficients obtained after the 3rd iteration of the DWT computation algorithm (Scaling fn., level 3), for the detail coefficients obtained after the 3rd iteration (Wavelets level 3), for the detail coefficients obtained after the 2nd iteration

	The degree of shift invariance	
Decomposition level	ADWT	Classical DWT
Scaling fn. level 3	0.8594	0.7552
Wavelets level 3	0.9981	0.7878
Wavelets level 2	0.9982	0.8265
Wavelets level 1	0.9992	0.9236

Table 1. A comparison between the proposed WT and the DWT with respect to the degree of shift-invariance

(Wavelets level 2) and for the detail coefficients obtained after the 1st iteration (Wavelets level 1). By recomposing all these signals, the initial step signal should be obtained. Mother wavelet used was Daubechies-10 (with five vanishing moments). In order to isolate the coefficients corresponding to each level, after the computation of the DWT, we put all the complex coefficients corresponding to the other levels to zero, by applying a "mask" on the sequence obtained after DWT computation. For a better understanding of this procedure, we illustrate in figure 3 the system used for the analysis of the shift-invariance at the 3rd decomposition level of the ADWT.

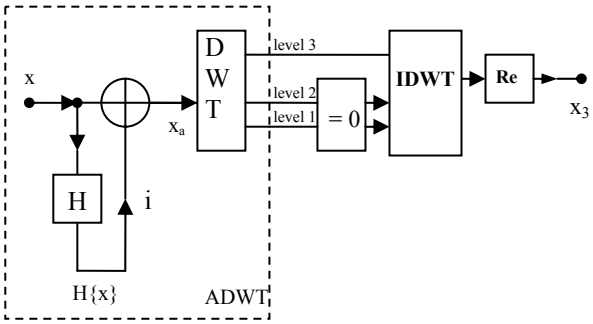


Figure 3. The system used for the shift-invariance analysis of the third level of the wavelet decomposition.

The first experiment already described is illustrated in figure 4. The results obtained using the proposed WT are presented in figure 4 a) and the results obtained using the classical decimated DWT in figure 4 b). It can be observed that the DWT is not shift-invariant. The ADWT is quasi shift-invariant. It can be observed that the ADWT is quasi shift-invariant. That is, for shifted version of the same signal applied to the transform's input, we obtained shifted-like versions of the signal reconstructed following the steps indicated in figure 3.

In fig. 5 we show the dependency of the degree of shift invariance of the proposed WT with respect to the regularity of the mother wavelet used for its computation. We investigated the Daubechies family, each element being indexed by its number of vanishing moments. As the curve illustrated in figure

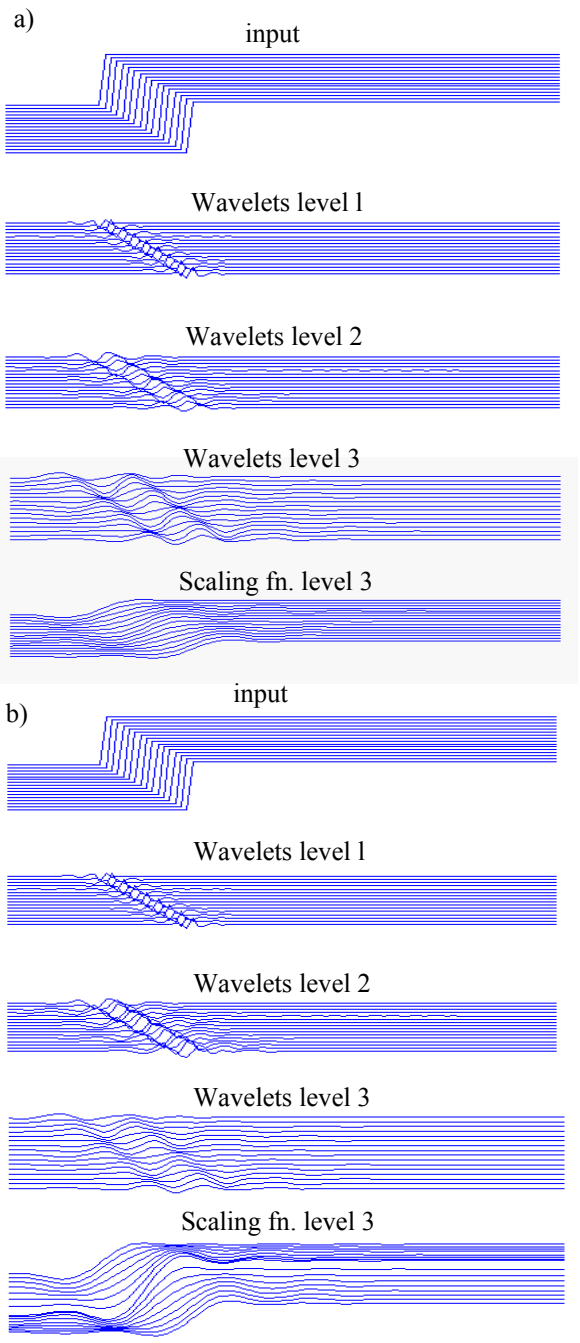


Figure 4. A comparison between the ADWT (a) and the DWT (b).

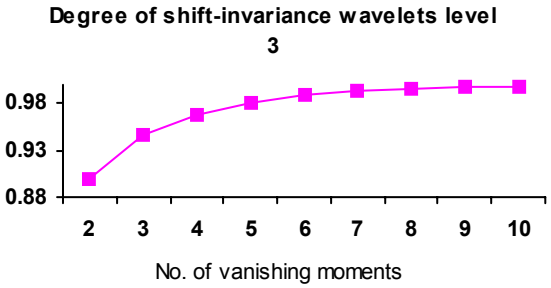


Figure 5. The dependency of the degree of shift-invariance of ADWT on the regularity of the mother wavelet used. for its computation.

Symmlet, 10	ADWT	CS step=1 64 delays	CS step=1 512 delays
Redundancy	Non redundant	64	512
Scaling fn. level 3	0,8594		0,7551
Wavelets level 3	0,9962	0,9962	0,9995
Wavelets level 2	0,9963	0,9965	0,9996
Wavelets level 1	0,9992	0,9985	0,9998
Daubechies, 10	ADWT	CS step=1 64 delays	CS step=1 512 delays
Scaling fn. level 3	0,8594	0,7551	0,7551
Wavelets level 3	0,9981	0,9965	0,9996
Wavelets level 2	0,9982	0,9968	0,9996
Wavelets level 1	0,9992	0,9985	0,9998

Table 2. A comparison between two quasi shift-invariant WTs, the ADWT and the CS.

5 indicates, the degree shift-invariance increases with the regularity of the mother wavelets used. In table 2 we present a comparison between our transform and the CS. It can be observed, analyzing this table, that the ADWT is equivalent to the CS with redundancy 64, from the degree of shift-invariance point of view. This in an excellent result, given that our transform is non-redundant, since for L samples to the input of ADWT, we still get L complex samples in the wavelet domain.

5. CONCLUSION

In this paper we propose a new complex non-redundant quasi shift-invariant WT. A new measure of the degree of shift-invariance of a WT is introduced. The degree of shift-invariance of the proposed transform is studied using this new measure. We show, on an illustrative example chosen, that the ADWT is equivalent from the degree of shift-invariance point of view with the CS with redundancy 64, when both WTs are applied to a signal having a duration of 1024 samples. This research will be continued on the following directions:

- a comparison of the degree of shift-invariance obtained applying the proposed WT with the degree of shift-invariance obtained applying other WTs like the DTCWT or the MBCWT.
- The generalization of ADWT in 2D.
- The construction and the study of a new 2D ADWT with improved directional selectivity, 2D ADWTIDS.
- The implementation and the study of a new 2D ADWTIDS with enhanced diversity, 2D ADWTIDSED.
- The construction and the study of a wavelet packets transform inspired by the 2D ADWTIDSED.

- The utilization of the 2D AWFTIDSED for the de-blurring and denoising of SONAR images.

ACKNOWLEDGEMENT

The authors want to thank Professor Jean-Marc Boucher and Associated Professor Sorin Moga from ENST-Bretagne, for the fruitful discussions on the topic of this paper, discussions developed during the conferences sustained in our university. The results reported here were obtained in the framework of two Romanian research programs granted by CNCSIS and directed by Associated Professor Dorina Isar and Professor Alexandru Isar. The authors would also like to express a special word of gratitude for the later, who carefully guided us during our research work.

REFERENCES

[1] S.Mallat, "Zero-crossings of a wavelet transform", *IEEE Trans. Information Theory*, vol. 37, pp. 1019 - 1033, July 1991.

[2] G. Beylkin, "On the representation of operators in bases of compactly supported wavelets", *SIAM J. Numer. Anal*., vol. 29, no. 6, pp. 1716 - 1740, 1992.

[3] R. Coifman and D. Donoho, "Translation-invariant de-noising", *Wavelets and Statistics*, A. Antoniadis and G. Oppenheim Eds, Springer-Verlag, pp. 125-150, New York, 1995.

[4] M. Lang, H. Guo, J. E. Odegard, C. S. Burrus and R. O. Wells Jr., "Noise reduction using an undecimated discrete wavelet transform", *IEEE Signal Processing Lett*., vol.3, no.1, pp. 10-12, Jan. 1996.

[5] P. Abry, "Transformées en ondelettes-Analyzes multirésolution et signaux de pression en turbulence", Ph.D.dissertation, Université Claude Bernard, Lyon, France, 1994.

[6] Nick Kingsbury, "Complex Wavelets for Shift Invariant Analysis and Filtering of Signals", *Applied and Computational Harmonic Analysis*, 10, 234-253, 2001.

[7] Felix C. A. Fernandes, Rutter L.C. van Spaendonck and C. Sindey Burrus, "A New Framework for Complex Wavelet Transforms", *IEEE Transactions on Signal Processing*, vol. 51, no. 7, pp.1825–1837, July, 2000.

[8] E.P.Simoncelli, W.T. Freeman, E.H.Adelson and D.J.Heeger, "Shiftable multi-scale transforms", *IEEE Trans. on Inform. Theory*, vol. 38, pp. 587 – 607, March 1992.

Buletinul Ştiinţific al Universităţii "Politehnica" din Timişoara

Seria ELECTRONICĂ şi TELECOMUNICAŢII
TRANSACTIONS on ELECTRONICS and COMMUNICATIONS

Tom 51(65), Fascicola 2, 2006

Acoustic Noise Reduction using an Improved Power Spectral Subtraction Method Based on Hartley Transform

Radu M. Udrea, Silviu Ciochină, Dragoş N. Vizireanu[1]

Abstract – We propose an improved spectral subtraction method for reducing acoustic noise added to speech in noisy environments like helicopter cockpit or car engine. Basic power spectral subtraction is modified using Discrete Hartley Transform to estimate cross-terms that are usually neglected. A large amount of memory storage and computational volume is saved using a real data transform. Experiments with speech affected by Gaussian and engine noise showed a better estimation of clean speech with the proposed method.
Keywords: speech enhancement, spectral subtraction

I. INTRODUCTION

There are many situations when speech has to be processed in the presence of undesirable background noise that degrades speech quality and intelligibility. A variety of speech enhancement methods capable to reduce background noise were studied in the literature [1]. Many of them are adaptive techniques that use a second microphone for noise-only capture [2]. But multiple input may not be always available because of environment or cost reasons.

Spectral subtraction method was extensively studied [3], [4] because it can suppress noise effectively from speech corrupted signal only. The approach used was to estimate the power frequency spectrum of the clean speech by subtracting the noise power spectrum from the noisy power spectrum. An estimate of the current noise spectrum is approximated using the average noise square-magnitude measured during non-speech activity.

Major disadvantages of implementing spectral subtraction method consist of the large amount of computations involved in this algorithm. Obtaining noise speech spectrum, subtracting noise spectrum components and returning in time domain are operations that require many memory and processing time.

This paper presents an optimized algorithm of spectral subtraction using Discrete Hartley Transform for computing signal and noise spectrum. Also, the noise reduction algorithm was modified for Hartley spectral domain. We first identify an accurate estimation of power spectrum for clean speech. Cross terms that usually are neglected when computing power spectrum can be estimated from the spectrum of input speech and estimated noise. This terms can be effective computed using relations between Discrete Fourier Transform and Discrete Hartley Transform [6]. Algorithm optimization resulted using a real transform instead of complex Fourier Transform. Second, the equivalence between the two spectral domains and algorithm modifications are established. Finally, algorithm implementation and experimental results are presented.

II. SPECTRAL SUBTRACTION

Spectral subtraction needs only noisy speech as input [1]. The standard algorithm consists in obtaining an estimate of the noise-free signal spectrum by subtracting an estimate of the noise spectrum from the input noisy signal spectrum.

The noise spectrum is obtained from the measured signal during non-speech activity. Several assumptions are necessary for developing the algorithm. The background noise is acoustically added to the speech. The background noise remains locally stationary to the degree that its spectral magnitude expected value prior to speech activity equals its expected value during speech activity. The algorithm requires a speech detector to determine presence of speech in noisy signal.

Assume that a speech signal $s(n)$ has been degraded by the uncorrelated additive noise signal $v(n)$:

$$x(n) = s(n) + v(n) . \tag{1}$$

Taking the Fourier Transform of $x(n)$ gives:

$$X(\omega) = S(\omega) + V(\omega) . \tag{2}$$

Power spectral relation resulted from above equation is:

$$\begin{aligned} |X(\omega)|^2 = |S(\omega)|^2 + |V(\omega)|^2 + \\ + S(\omega)V^*(\omega) + S^*(\omega)V(\omega) \end{aligned} \tag{3}$$

[1] Politehnica University of Bucharest, Faculty of Electronics, Telecommunications and Information Technology
1-3 Iuliu Maniu Bd., 061071 Bucharest, e-mail mihnea@comm.pub.ro, silviu@comm.pub.ro, nae@comm.pub.ro.

where $S^*(\omega)$ and $V^*(\omega)$ are complex conjugates of $S(\omega)$ and $V(\omega)$ respectively.

Because in our system only the power of the input noisy signal $|X(\omega)|^2$ can be evaluated, the rest of terms are approximated by their average during non-speech activity period.

If $v(n)$ is uncorrelated with $s(n)$ then:

$$E\{S(\omega)V^*(\omega)\}=0 \text{ and } E\{S^*(\omega)V(\omega)\}=0. \quad (4)$$

The power spectral subtraction estimator results by replacing noise square-magnitude $|V(\omega)|^2$ with its average value taken during non-speech activity period:

$$|\hat{S}(\omega)|^2=|X(\omega)|^2-|\hat{V}(\omega)|^2. \quad (5)$$

where $|\hat{V}(\omega)|^2=E\{|V(\omega)|^2\}$.

Based on the fact that human ear does not perceive phase modifications [5], the phase $\theta_X(\omega)$ of the input signal is used for reconstruction of the estimated signal spectrum:

$$\hat{S}(\omega)=[|X(\omega)|^2-|\hat{V}(\omega)|^2]^{1/2}e^{j\theta_X(\omega)}. \quad (6)$$

The block diagram of spectral subtraction algorithm is represented in Fig. 1. Input signal spectrum is obtained using Discrete Fourier Transform (DFT) over the windowed half-overlapped input data buffer. The magnitude spectra of the windowed data are calculated and subtracted by the noise spectra calculated during non-speech activity.

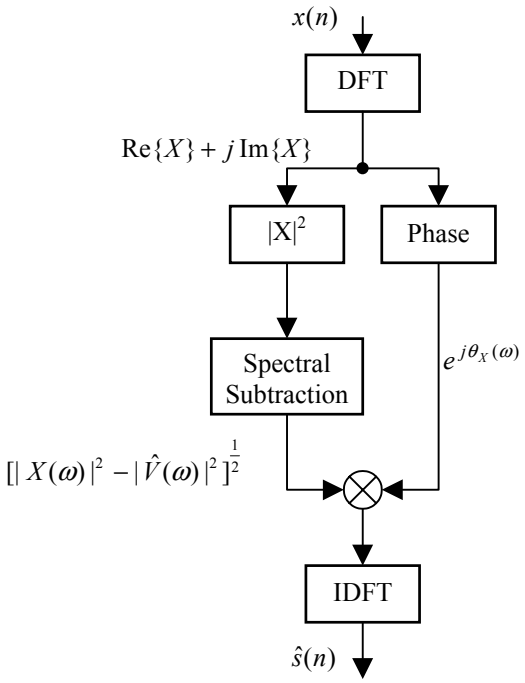


Fig. 1. Block diagram of spectral subtraction.

Then, time domain signal is obtained using Inverse Discrete Fourier Transform (IDFT) from the modified magnitude. Based on the relation (5) a number of secondary modifications can be implemented to reduce the spectral error resulted from the difference between real noise spectrum and its estimated average [1]. These include: magnitude averaging, half-wave rectification, nonlinear residual noise reduction and spectral over-subtraction.

One observation that results from the algorithm described above is that DFT returns the complex spectrum as real and imaginary parts. From this we have to compute magnitude and phase that involves a large number of computations for every spectral component. Notice that after spectral subtraction another number of computations are needed to obtain real and imaginary parts in order to perform IDFT.

III. THE IMPROVED ALGORITHM

In the literature, the cross-terms $E\{S(\omega)V^*(\omega)\}$ and $E\{S^*(\omega)V(\omega)\}$ are neglected based on the assumption that the additive noise $v(n)$ is uncorrelated with the speech signal $s(n)$. However, although these assumptions hold true in the statistical sense based on long term averaging, the assumption is not necessarily true for short-time estimates, as is the case with all the subtractive type algorithms, which are processed on a frame-by-frame basis. Figure 2 plots the values of the power spectrums of speech and noise with dashed lines and with solid line the cross terms $S(\omega)V^*(\omega)+S^*(\omega)V(\omega)=2\,\mathrm{Re}\left[S(\omega)V^*(\omega)\right]$.

It can be clearly seen that the cross-terms are not negligible compared to the values of the power spectrum amplitudes of speech and noise.

By neglecting the cross-terms, will result an underestimate of the clean speech and thereby we will not completely suppress the noise. On accounting for the cross-terms it is possible to reduce the residual noise in the enhanced speech and thereby provide a better estimate of the clean speech.

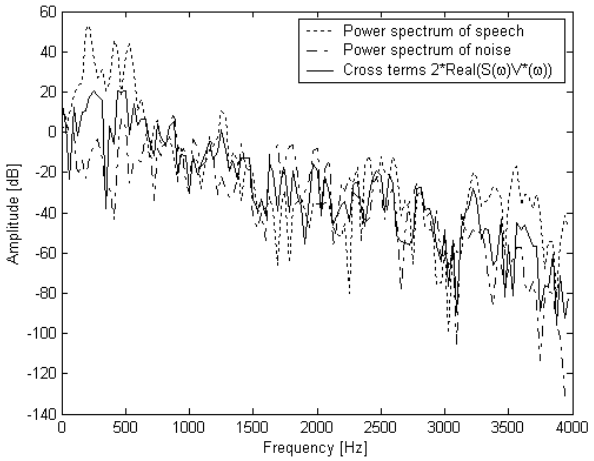


Fig. 2. Plots for power spectra of speech and noise along with the cross-term of speech and noise, over a 20 ms window.

Unfortunately, we do not have access to the clean speech and we can estimate only an average power spectrum of the noise. Therefore, in an attempt to approximate the cross-terms, using equation (2) results:

$$\begin{aligned} &X(\omega)V^*(\omega)+X^*(\omega)V(\omega)= \\ &=\left[S(\omega)+V(\omega)\right]V^*(\omega)+\left[S^*(\omega)+V^*(\omega)\right]V(\omega)= \\ &=S(\omega)V^*(\omega)+S^*(\omega)V(\omega)+2\left|V(\omega)\right|^2 \end{aligned} \quad (7)$$

If we replace the noise spectrum power $|V(\omega)|^2$ with its average estimate $|\hat{V}(\omega)|^2$ equation (3) becomes:

$$\begin{aligned} &|X(\omega)|^2=|S(\omega)|^2+|\hat{V}(\omega)|^2+ \\ &+X(\omega)V^*(\omega)+X^*(\omega)V(\omega)-2|\hat{V}(\omega)|^2 \end{aligned} \quad (8)$$

Still we cannot estimate the complex terms $V^*(\omega)$ and $V(\omega)$.

The idea of the proposed algorithm is to use for computations of short term spectrum, the Discrete Hartley Transform (DHT) which is a real transform defined by:

$$X_H(k)=\mathrm{DHT}\{x(n)\}(k)=\sum_{n=0}^{N-1}x(n)\,\mathrm{cas}\left(kn\frac{2\pi}{N}\right). \quad (9)$$

for $k=0$ to N,

and where $\mathrm{cas}(a)=\cos(a)+\sin(a)$.

The relation that can be determined between Discrete Hartley Transform and Discrete Fourier Transform (DFT) is:

$$X_H(k)=\mathrm{Re}\{X_F(k)\}-\mathrm{Im}\{X_F(k)\}. \quad (10)$$

where subscript H indicates DHT computed spectrum and subscript F indicates DFT computed spectrum

Also, because the input signal is a real sequence, symmetry of Fourier Transform gives:

$$\begin{aligned} &\mathrm{Re}\{X_F(k)\}=\mathrm{Re}\{X_F(N-k)\} \\ &\mathrm{Im}\{X_F(k)\}=-\mathrm{Im}\{X_F(N-k)\} \end{aligned} \quad (11)$$

These relations lead us to the following power spectral component relations:

$$\begin{aligned} &X_H^2(k)+X_H^2(N-k)= \\ &=2\left(\mathrm{Re}\{X_F(k)\}^2+\mathrm{Im}\{X_F(k)\}^2\right)=2|X_F(k)|^2 \end{aligned} \quad (12)$$

for $k=0$ to $\frac{N}{2}-1$.

Expressed in terms of DFT equation (8) becomes:

$$\begin{aligned} &|X_F(k)|^2=|S_F(k)|^2-|V_F(k)|^2+ \\ &+X_F(k)V_F^*(k)+X_F^*(k)V_F(k) \end{aligned} \quad (13)$$

Based on relations:

$$\begin{aligned} &X_H(k)=\mathrm{Re}\{X_F(k)\}-\mathrm{Im}\{X_F(k)\} \\ &X_H(N-k)=\mathrm{Re}\{X_F(k)\}+\mathrm{Im}\{X_F(k)\} \end{aligned} \quad (14)$$

last terms from equation (13) can be expressed as:

$$\begin{aligned} &X_F(k)V_F^*(k)+X_F^*(k)V_F(k)= \\ &=X_H(k)V_H(k)+X_H(N-k)V_H(N-k) \end{aligned} \quad (15)$$

We can estimate Hartley transform of the noise from the current frame by its absolute value average taken from frames when there is no voice activity and use the sign of the DHT from the current frame.

$$\hat{V}_H(k)=\mathrm{sgn}\{X_H(k)\}\cdot E\{|V_H(k)|\} \quad (16)$$

Results the power spectral subtraction estimator:

$$\begin{aligned} &|\hat{S}_F(k)|^2=|X_F(k)|^2+|\hat{V}_F(k)|^2- \\ &-X_H(k)\hat{V}_H(k)-X_H(N-k)\hat{V}_H(N-k) \end{aligned} \quad (17)$$

for $k=0$ to $\frac{N}{2}-1$.

By replacing the DFT power spectrums with DHT like in relation (12) results an DHT based estimator:

$$\hat{S}_H(k)^2=X_H(k)^2+\hat{V}_H(k)^2-2X_H(k)\hat{V}_H(k) \quad (18)$$

A reduced arithmetic complexity results for spectral subtraction algorithm using Discrete Hartley Transform because, instead of complex number computations like phase and absolute value we use the real data DHT.

Time domain estimate of noise-free signal is obtained with inverse transform:

$$\hat{s}(n)=\mathrm{IDHT}\{\hat{S}_H(k)\cdot\mathrm{sgn}(X_H(k)\}(n). \quad (19)$$

where Inverse Discrete Hartley Transform (IDHT) is defined by:

$$\mathrm{IDHT}\{X_H(k)\}(n)=\frac{1}{N}\sum_{k=0}^{N-1}X_H(k)\,\mathrm{cas}\left(kn\frac{2\pi}{N}\right) \quad (20)$$

Since IDHT has the same formula like DHT we can use the same subroutine for both transforms, changing input data vectors accordingly.

IV. ALGORITHM IMPLEMENTATION AND EXPERIMENTAL RESULTS

The proposed power spectral subtraction algorithm was implemented on a 16bit fixed-point Freescale SC140 DSP. Two types of noise were used for experiments: Gaussian white noise low-pass filtered

and engine recorded noise. The noise and speech signals are added and used as input. Input samples are stored into a buffer designed for overlap-add method. The input buffer is then windowed using Hanning window and the Discrete Hartley Transform is computed with a fast algorithm similar to FFT.

Since our algorithm uses a real transform, a large amount of DSP memory is saved in data storage because there is no need for imaginary part of data and spectrum like in the Fourier Transform based algorithm and because the IDHT is the same subroutine like the DHT. Also, the proposed algorithm reduces by 10% arithmetic complexity of spectral subtraction method because no phase and other complex calculation are needed.

Objective measurements, in terms of signal to noise ratio improvement, can be evaluated using results above. Output noise power is computed using norm of the difference between enhanced speech and originally input clean speech.

$$SNR_{out}(dB) = 10\log_{10}\frac{\sum_{n=0}^{n\max}|s(n)|^2}{\sum_{n=0}^{n\max}|\hat{s}(n)-s(n)|^2} \qquad (21)$$

In Table 1 is shown signal to noise ratio improvement in case of standard Power Spectral Subtraction and in case of the modified method that estimates cross-terms using Hartley Transform.

Table 1. Output signal SNR for standard algorithm and for modified method with different input SNRs.

SNR_{in}	SNR_{out} (basic method)	SNR_{out} (modified method)
0 dB	10.98 dB	11.73 dB
-5 dB	3.6 dB	3.93 dB

Fig. 3 shows the spectrograms of the speech signal, the noise-speech signal and the noise-cleared speech signal obtained using our algorithm. Due to the low signal-to-noise ratio of the input signal (about -5dB) the reconstructed signal has still residual noise that can be perceived like musical tones with random frequency.

V. CONCLUSIONS

In this paper we presented a more accurate estimation of power spectrum for clean speech when affected by additive noise. Cross terms that usually are neglected when computing power spectrum can be estimated from the spectrum of input speech and estimated noise. These terms can be effective computed using relations between Discrete Fourier Transform and Discrete Hartley Transform. The noise reduction algorithm was modified for Hartley spectral domain.

Experiments were effectuated with white noise or engine noise. Performances of the noise reduction algorithm were compared with the standard spectral subtraction algorithm.

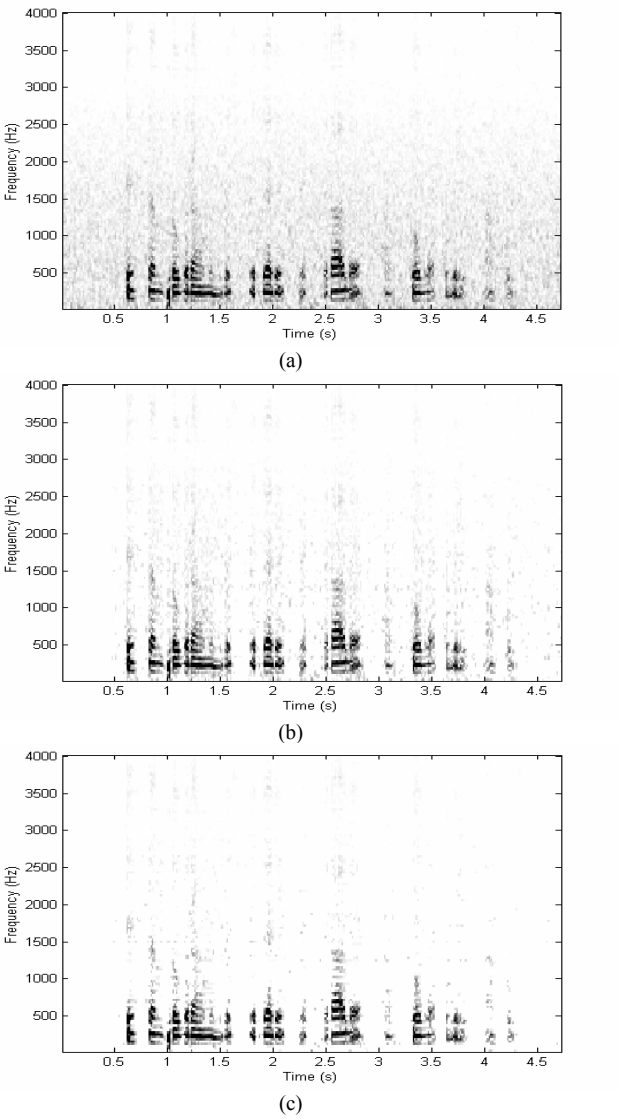


Fig. 3. (a) Spectrogram of the speech corrupted by low-pass filtered white Gaussian noise, (b) Spectrogram of the enhanced speech using basic power spectral subtraction (c) Spectrogram of the enhanced speech using DHT modified spectral subtraction algorithm.

REFERENCES

[1] M. Berouti, R. Schwartz, J. Makhoul, “Enhancement of speech corrupted by acoustic noise”, Proc. IEEE Int. Conf. Acoust., Speech and Signal Proc., pag. 208-211, Apr. 1979.

[2] L. Arslan, A. McCree, V. Viswanathan, “New methods for adaptive noise suppression,” ICASSP, vol.1, May 1995, pp. 812-815.

[3] Chin-Teng Lin, “Single-channel speech enhancement in variable noise-level environment”, IEEE Trans on Systems, Man and Cybernetics, Part A, Vol. 33, Jan. 2003, pp. 137–143.

[4] S. Kamath, P. Loizou, “A multi-band spectral subtraction method for enhancing speech corrupted by colored noise”, Proceedings of ICASSP-2002, Orlando, FL, May 2002.

[5] D.L. Wang and J.S. Lim, "The Unimportance of Phase in Speech Enhancement", IEEE Trans. On Acoustics, Speech, and Signal Processing, vol. 30, no.4, Aug. 1982, pp. 679-681.

[6] R. M. Udrea, S. Ciochină, “Implementation of an Optimized Algorithm for Acoustic Noise Reduction Based on Hartley Transform”, Proc. of the Symposium on Electronics and Telecommunications “Etc. 2000”, vol. I, Timişoara, Romania, Nov. 2000, pp. 223-227.

Buletinul Ştiinţific al Universităţii "Politehnica" din Timişoara

Seria ELECTRONICĂ şi TELECOMUNICAŢII
TRANSACTIONS on ELECTRONICS and COMMUNICATIONS

Tom 51(65), Fascicola 2, 2006

Adapting a Normalized Gradient Subspace Algorithm to Real-Valued Data Model

Stefan Slavnicu, Silviu Ciochină[1]

Abstract – A new gradient approach to adaptive subspace-based frequency estimation of multiple real valued sine waves is considered in this paper. The new approach proposed here combines the normalized gradient subspace tracking technique based on Oja learning rule - NOOja (for the signal subspace update) with the ESPRIT-like frequency estimation of real-valued sinusoids (for frequency values retrieval). Consequently, a new adaptive subspace-tracking algorithm for frequency estimation is proposed. The method proposed brings a significant reduction in arithmetical complexity at the same level of accuracy. The algorithm is tested in numerical simulations and compared to complex-valued NOja method.

Keywords: subspace tracking, frequency estimation, real-valued data, R-ESPRIT, NOja

I. INTRODUCTION

Adaptive subspace tracking for determining time-varying frequencies of sine wave carriers is a research field still under study. Not only old techniques have been optimized [6], [7], but also new algorithms have been developed in order to improve the accuracy of the methods or to decrease the computational burden [2], [8].

However, traditional methods present the major drawback of assuming that the data are complex-valued and this implies additional computational effort. All super-resolution subspace block methods (MUSIC, ESPRIT etc.) are based on a complex-valued signal model, as they have initially been designed for array processing [4]. Only recently Mahata and Söderström developed an ESPRIT-like method to estimate the real-valued sinusoidal frequencies [1], [9]. This new non-iterative method, called R-ESPRIT by the authors, is based on a real-valued signal model and brings a spectacular reduction in the number of operations required to compute the frequency estimates.

It is then natural to think about adaptive methods able to take advantage of this much lower complexity. In the present paper we have made a further step to the work presented in [2] in the context of projection approximation subspace tracking and adapted the well-known normalized orthogonal gradient subspace-tracking algorithm based on Oja learning rule NOOja to the real-valued signal model.

Similar to the method presented in [2] for PAST algorithm, the subspace tracking-type NOOja method is modified for applying R-ESPRIT for real sinusoids retrieval. We will name the new algorithm R-NOOja. From the author's knowledge, the new method has never been published before.

We will compare the performances and the complexity of the newly derived algorithm with the NOOja method based on the complex-valued data model.

II. SIGNAL MODEL

The signal model is presented in [2]. We will briefly review it here for present paper consistency.

The input signal consists in a number r of sinusoidal signals that may well be sine wave carriers, embedded in white Gaussian noise:

$$x(t) = \sum_{k=1}^{r} s_k \sin(t\omega_k + \phi_k) + n(t)\,, \qquad (1)$$

where s_k is the amplitude, ω_k is the angular frequency of the of the k^{th} sinusoid and $n(t)$ represents the corrupting additive zero-mean white noise. The phases $\{\phi_k\}_{k=1}^{r}$ are random variables uniformly distributed in the $[-\pi, \pi]$ interval.

The compact subspace representation dedicated for real valued sinusoids differs from the classical complex-valued signal model [1]. We have to obtain an alternative snapshot vector so that its noise-free part lies in a subspace of dimension r. To that aim, we will introduce the following input vectors:

$$\mathbf{x}_c(t) \overset{\Delta}{=} [x(t) \quad \dots \quad x(t+n-1)]^T \qquad (2)$$

$$\mathbf{x}_b(t) \overset{\Delta}{=} [x(t-1) \quad \dots \quad x(t-n)]^T \qquad (3)$$

$$\mathbf{x}_r(t) \overset{\Delta}{=} \frac{1}{2}\{\mathbf{x}_c(t) + \mathbf{x}_b(t)\} \qquad (4)$$

where the snapshot vector dimension $n > 2r$. From the above definitions we obtain

$$\mathbf{x}_r(t) = \mathbf{A}_r \mathbf{s}_r(t) + \mathbf{n}_r(t) \qquad (5)$$

[1] POLITEHNICA University of Bucharest, Splaiul Independenţei, 313,Bucharest, ROMANIA,
e-mail: slavnicu@ieee.org; silviu@comm.pub.ro

where $\mathbf{s}_r(t)$ is an $r\times 1$ vector given by

$$\mathbf{s}_r(t)=\begin{bmatrix} a_1\cos\{\omega_1 t+\phi_1^+\} \\ \vdots \\ a_r\cos\{\omega_r t+\phi_r^+\}\end{bmatrix} \quad (6)$$

where $\phi_k^+=\phi_k-(1/2)\omega_k$ for $1\le k\le r$. $\mathbf{A}_r$ is an $n\times r$ matrix given by

$$\mathbf{A}_r=\begin{bmatrix} \cos\left(\frac{\omega_1}{2}\right) & \cdots & \cos\left(\frac{\omega_r}{2}\right) \\ \cos\left(\frac{3\omega_1}{2}\right) & \cdots & \cos\left(\frac{3\omega_r}{2}\right) \\ \vdots & \ddots & \vdots \\ \cos\left\{\left(n-\frac{1}{2}\right)\omega_1\right\} & \cdots & \cos\left\{\left(n-\frac{1}{2}\right)\omega_r\right\}\end{bmatrix}. \quad (7)$$

The noise snapshot vector $\mathbf{n}_r(t)$ in this modified model is given by

$$\mathbf{n}_r(t)=\frac{1}{2}\{\mathbf{n}_c(t)+\mathbf{n}_b(t)\} \quad (8)$$

where

$$\mathbf{n}_c(t)\overset{\Delta}{=}[n(t) \quad \dots \quad n(t+n-1)]^T \quad (9)$$

$$\mathbf{n}_b(t)\overset{\Delta}{=}[n(t-1) \quad \dots \quad n(t-n)]^T \quad (10)$$

One can show [1] that $\mathbf{A}_r$ is a full column rank matrix. The important fact here is that the noise-free part of $\mathbf{x}_r(t)$ lies in an *r*-dimensional subspace that is different from the complex-valued data model, where the dimension of the signal subspace is 2*r*.

Further on, let us introduce

$$\mathbf{P}_r\overset{\Delta}{=}E\{\mathbf{s}_r(t)\mathbf{s}_r^T(t)\}. \quad (11)$$

The noise vectors $\mathbf{n}_c(t)$ and $\mathbf{n}_b(t)$ are random vectors, mutually independent, with $E\{\mathbf{n}_r(t)\mathbf{n}_r^T(t)\}=(\sigma^2/2)\mathbf{I}_n$ where σ^2 is the noise variance. We obviously have that

$$\mathbf{R}_r\overset{\Delta}{=}E\{\mathbf{x}_r(t)\mathbf{x}_r^T(t)\}=\mathbf{A}_r\mathbf{P}_r\mathbf{A}_r^T+\frac{\sigma^2}{2}\mathbf{I}_n. \quad (12)$$

We may then consider the eigenvalue decomposition

$$\mathbf{R}_r=\mathbf{S}_r\mathbf{\Lambda}_r\mathbf{S}_r^T+\mathbf{G}_r\mathbf{\Sigma}_r\mathbf{G}_r^T \quad (13)$$

where $\mathbf{\Lambda}_r$ is an $r\times r$ diagonal matrix containing the *r* dominant eigenvalues of $\mathbf{R}_r$ on the diagonal. The $n\times r$ matrix $\mathbf{S}_r$ is composed of the corresponding left eigenvectors. In the same perspective, $\mathbf{\Sigma}_r$ is a diagonal matrix containing the remaining $n-r$ eigenvalues of $\mathbf{R}_r$. The $n\times(n-r)$ matrix $\mathbf{G}_r$ is composed of the corresponding left eigenvectors. The columns of $\mathbf{G}_r$ are orthogonal to those of $\mathbf{S}_r$.

One can show (see [1]) that

$$\mathbf{S}_r=\mathbf{A}_r\mathbf{C}_r \quad (14)$$

where

$$\mathbf{C}_r=\mathbf{P}\mathbf{A}_r^T\mathbf{S}_r\left\{\mathbf{\Lambda}_r-\frac{\sigma^2}{2}\mathbf{I}_n\right\}^{-1}. \quad (15)$$

The columns of $\mathbf{S}_r$ form an orthonormal basis of the column space of $\mathbf{A}_r$. The idea is to adaptively obtain an estimate of $\mathbf{S}_r$ from the data via the normalized orthogonal gradient adaptive subspace tracking method NOOja, which will then be processed to obtain the frequency estimates.

III. ALGORITHMS

A. *R-Esprit*

The R-ESPRIT algorithm is an ESPRIT-like estimation method of real-valued sinusoidal frequencies. The algorithm has been proposed and has been presented in detail in [1] and [9]. We will resume as in [2] the main aspects of this method as it represents a key factor in developing our new adaptive method. R-ESPRIT relies on the signal model presented in Section 2 of the present paper.

The basic idea is to make use of two $(n-2)\times n$ Toeplitz matrices

$$\mathbf{T}_r^{(1)}=\begin{bmatrix} 0 & 1 & 0 & 0 & \cdots & 0 \\ 0 & 0 & 1 & 0 & \cdots & 0 \\ \vdots & \ddots & \ddots & \ddots & \ddots & \vdots \\ 0 & \cdots & 0 & 0 & 1 & 0\end{bmatrix} \quad (16)$$

$$\mathbf{T}_r^{(2)}=\frac{1}{2}\begin{bmatrix} 1 & 0 & 1 & 0 & \cdots & 0 \\ 0 & 1 & 0 & 1 & \cdots & 0 \\ \vdots & \ddots & \ddots & \ddots & \ddots & \vdots \\ 0 & \cdots & 0 & 1 & 0 & 1\end{bmatrix}. \quad (17)$$

From the definition of matrix (see relation (7)) the following identity holds:

$$\mathbf{T}_r^{(2)}\mathbf{A}_r=\mathbf{T}_r^{(1)}\mathbf{A}_r D_r \quad (18)$$

where D_r is the following diagonal matrix:

$$D_r=\mathrm{diag}\{\cos(\omega_1), \quad \dots \quad ,\cos(\omega_n)\}. \quad (19)$$

Let us also introduce the following matrix

$$\mathbf{\Phi}_r=\mathbf{C}_r^{-1}D_r\mathbf{C}_r \quad . \quad (20)$$

Then, the algorithm may be derived as follows:

a) It is first required to estimate $\hat{\mathbf{S}}_r$ from the input data.

b) This estimate will be used in estimating $\hat{\mathbf{\Phi}}_r$, from (14) and (18), as

$$\hat{\mathbf{\Phi}}_r=(\mathbf{T}_r^{(1)}\hat{\mathbf{S}}_r)^{-1}\mathbf{T}_r^{(2)}\hat{\mathbf{S}}_r. \quad (21)$$

c) The eigendecomposition of $\hat{\mathbf{\Phi}}_r$ will lead us to $\hat{D}_r$, following equation (20).

d) Knowing $\hat{D}_r$, frequency values easily result from:

$$\omega_k=\cos^{-1}\{\hat{D}_r(k,k)\} \qquad k=1,\dots,r \quad (22)$$

This finally gives the frequency estimates. The dimension of the signal subspace is reduced to *r* from 2*r* i.e. the case of traditional ESPRIT method for *r* real-valued sine waves.

B. *The R-NOOja algorithm*

In this chapter we will derive a novel adaptive method for estimating the signal subspace $\hat{\mathbf{S}}_r$ from the input data. This algorithm is based on the NOOja adaptive method proposed in [7] and modified for the real data model presented in Section 2 of this paper. We will therefore refer to this algorithm as R-NOOja. From the authors' knowledge, this method has never been published in this form before.

Let $\mathbf{x}_r \in \boldsymbol{R}^n$ be the input data vector at time t defined as in relation (5), with the correlation matrix $\mathbf{R}_r = E\{\mathbf{x}_r \mathbf{x}_r^H\}$.

Note here that r represents the number of real sinusoids, which is half the number of complex sinusoids used in traditional adaptive methods for tracking frequencies of real sine waves. This is the first level of reduction in arithmetical complexity. The second level comes from the use of R-ESPRIT technique, a much simpler method adapted for real sinusoids environments, instead of ESPRIT, traditionally proposed for complex-valued signal environments.

We are interested to recursively estimate the signal subspace $\hat{\mathbf{S}}_r$, therefore to compute the signal subspace estimate at the time instant t from the subspace estimate at $t-1$ and the new arriving sample vector $\mathbf{x}_r$.

As in [7] we know that if we consider the following cost function:

$$\begin{aligned} J(\mathbf{W}_r(t)) &= E\{\left\|\mathbf{x}_r - \mathbf{W}_r(t)\mathbf{W}_r^T(t)\mathbf{x}_r\right\|^2\} \\ &= \mathrm{tr}\left[\mathbf{R}_r(t)\right] - 2\,\mathrm{tr}\left[\mathbf{W}_r^T(t)\mathbf{R}_r(t)\mathbf{W}_r^T(t)\right] \\ &\quad + \mathrm{tr}\left[\mathbf{W}_r^T(t)\mathbf{R}_r(t)\mathbf{W}_r(t)\mathbf{W}_r^T(t)\mathbf{W}_r(t)\right] \end{aligned} \quad (23)$$

where $\mathbf{W}_r(t)$ is a real-valued $n \times r$ matrix, then following the theory in [3] and [5], we can prove that the matrix $\mathbf{W}_r(t) \in \boldsymbol{R}^{n \times r}\ (r < n)$ minimizing $\mathrm{J}(\mathbf{W}_r(t))$ is a good estimate for the signal subspace $\hat{\mathbf{S}}_r(t)$ of the correlation matrix $\mathbf{R}_r(t)$.

We can compute the gradient of the cost function

$$\begin{aligned} \nabla J(\mathbf{W}_r(t)) = [&-2\mathbf{R}_r(t) + \mathbf{R}_r(t)\mathbf{W}_r(t)\mathbf{W}_r^T(t) \\ &+ \mathbf{W}_r(t)\mathbf{W}_r^T(t)\mathbf{R}_r(t)]\mathbf{W}_r(t) \end{aligned} \quad (24)$$

and we can write the signal subspace update as

$$\begin{aligned} \mathbf{W}_r(t) = \mathbf{W}_r(t-1) - \mu[&-2\hat{\mathbf{R}}_r(t) + \hat{\mathbf{R}}_r(t)\mathbf{W}_r(t-1)\mathbf{W}_r^T(t-1) \\ &+ \mathbf{W}_r(t-1)\mathbf{W}_r^T(t-1)\hat{\mathbf{R}}_r(t)]\mathbf{W}_r(t-1) \end{aligned} \quad (25)$$

where $\mu > 0$ represents the adaptation step and $\hat{\mathbf{R}}(t)$ represents the estimate of the correlation matrix $\mathbf{R}$ at time instant t.

The simplest method to estimate matrix $\mathbf{R}(t)$ is to consider the instantaneous estimate $\hat{\mathbf{R}}(t) = \mathbf{x}(t)\mathbf{x}^H(t)$ according to LMS method from adaptive filtering. The following recursive formulas for updating the signal subspace result:

$$\mathbf{y}_r(t) = \mathbf{W}_r^T(t-1)\mathbf{x}_r(t) \quad (26)$$

$$\begin{aligned} \mathbf{W}_r(t) = \mathbf{W}_r(t-1) + \mu[&2\mathbf{x}_r(t)\mathbf{y}_r^T(t) \\ &- \mathbf{x}_r(t)\mathbf{y}_r^T(t)\mathbf{W}_r^T(t-1)\mathbf{W}_r(t-1) \\ &- \mathbf{W}_r(t-1)\mathbf{y}_r(t)\mathbf{y}_r^T(t)] \end{aligned} \quad (27)$$

Further on, we see that we may approximate

$$\mathbf{W}_r^T(t-1)\mathbf{W}_r(t-1) \cong \mathbf{I}_r \quad (28)$$

where $\mathbf{I}_r$ is the $r \times r$ identity matrix. Thus, we obtain a simplified version of the gradient method that represents, in fact, the Oja learning rule:

$$\mathbf{W}_r(t) = \mathbf{W}_r(t-1) + \mu[\mathbf{x}_r(t) - \mathbf{W}_r(t-1)\mathbf{y}_r(t)]\mathbf{y}_r^T(t). \quad (29)$$

Approximation (28) is based on the observation that, for stationary signals, $\mathbf{W}_r(t)$ will converge towards a matrix with orthonormal columns (if $\mu = \mu(t) \overset{t\to\infty}{\to} 0$), or almost orthonormal (if μ is small and constant).

In order to obtain the normalized orthogonal version of Oja algorithm we will add an orthonormalization step of the real-valued matrix $\mathbf{W}_r(t)$ and we will write

$$\mathbf{W}_r(t) = \mathbf{W}_r(t)\left(\mathbf{W}_r^T(t)\mathbf{W}_r(t)\right)^{-1/2} \quad (30)$$

where $\left(\mathbf{W}_r^T(t)\mathbf{W}_r(t)\right)^{-1/2}$ denotes the inverse of the square root for the matrix $\left(\mathbf{W}_r^T(t)\mathbf{W}_r(t)\right)$.

As in [7], one can easily show that

$$(\mathbf{W}_r^T(t)\mathbf{W}_r(t))^{-1/2} = \mathbf{I} + \tau(t)\mathbf{y}_r(t)\mathbf{y}_r^T(t) \quad (31)$$

where

$$\tau(t) \overset{\Delta}{=} \frac{1}{\left\|\mathbf{y}_r(t)\right\|^2}\left(\frac{1}{\sqrt{1+\mu_{opt}^2(t)\left\|\mathbf{e}(t)\right\|^2\left\|\mathbf{y}_r(t)\right\|^2}} - 1\right) \quad (32)$$

and

$$\mathbf{e}(t) = \mathbf{x}_r(t) - \mathbf{W}_r(t-1)\mathbf{y}_r(t). \quad (33)$$

Taking into account that $\mathbf{W}_r(t)$ is now orthogonal at each iteration, i.e. $\mathbf{W}_r^T(t)\mathbf{W}_r(t) = \mathbf{I}$, we may write

$$\hat{\nabla} J(\mathbf{W}_r(t)) = -[\mathbf{x}_r(t) - \mathbf{W}_r(t-1)\mathbf{y}_r(t)]\mathbf{y}_r^T(t). \quad (34)$$

The optimal variable stepsize at time instant t becomes

$$\hat{\mu}_{opt}(t) = \frac{1}{\left\|\mathbf{y}_r(t)\right\|^2 - \left\|\mathbf{x}_r(t)\right\|^2} \quad (35)$$

where one can show that $\left\|\mathbf{y}_r(t)\right\|^2 - \left\|\mathbf{x}_r(t)\right\|^2 \le 0$.

Following the relations (1) to (35) and adapting the method in [7] we can easily derive a new normalized orthogonal subspace tracking gradient algorithm based on Oja learning rule and adapted to real sinusoidal carrier environments. We name this method R-NOOja. We will prove in the next section that the newly derived method performs as well as complex-valued NOOja algorithm at a much less computational burden.

Table 1 briefly presents the subspace tracking R-NOOja algorithm adapted for sinusoidal carriers frequency identification. Here $\mathbf{x}_r(t)$ represents the input vector at time t.

Table 1
R-NOOJA ALGORITHM FOR REAL-VALUED FREQUENCY ESTIMATION

r = number of real sinusoids

n = dimension of $\mathbf{x}_r(t)$

$$\mathbf{W}(0) = \begin{bmatrix} \mathbf{I}_r \\ \mathbf{0}_{n-r} \end{bmatrix}$$

FOR $t = 1,2, \ldots$ DO

$$\mathbf{x}_r(t) \overset{\Delta}{=} \frac{1}{2}\{\mathbf{x}_c(t) + \mathbf{x}_b(t)\}$$

$$\mathbf{y}_r(t) = \mathbf{W}_r^T(t-1)\mathbf{x}_r(t)$$

$$\mathbf{z}(t) = \mathbf{W}_r(t-1)\mathbf{y}_r(t)$$

$$\mathbf{e}(t) = \mathbf{x}_r(t) - \mathbf{z}(t)$$

$$\hat{\mu}_{opt}(t) = \frac{-\mu}{\|\mathbf{y}_r(t)\|^2 - \|\mathbf{x}_r(t)\|^2 + \gamma}$$

$$\phi(t) = \frac{1}{\sqrt{1 + \hat{\mu}_{opt}^2(t)\|\mathbf{e}(t)\|^2\|\mathbf{y}_r(t)\|^2}}$$

$$\tau(t) = \frac{\phi(t) - 1}{\|\mathbf{y}_r(t)\|^2}$$

$$\mathbf{p}(t) = -\frac{1}{\hat{\mu}_{opt}(t)}\tau(t)\mathbf{z}(t) + \phi(t)\mathbf{e}(t)$$

$$\mathbf{u}(t) = \frac{\mathbf{p}(t)}{\|\mathbf{p}(t)\|}$$

$$\mathbf{v}(t) = \mathbf{W}_r^T(t-1)\mathbf{u}(t)$$

$$\mathbf{W}_r(t) = \mathbf{W}_r(t-1) - 2\mathbf{u}(t)\mathbf{v}^T(t)$$

$$\hat{\mathbf{S}}_r(t) = \mathbf{W}_r(t)$$

$$\mathbf{f}(t) = \text{R}-\text{ESPRIT}(\hat{\mathbf{S}}_r(t))$$

END FOR

Estimated frequencies vector $\mathbf{f}$ is obtained by applying R-ESPRIT method (see section 3.1.) to the orthonormal basis $\mathbf{W}_r$ of signal subspace.

Here μ and γ represent two positive constants ($0 < \mu < 1$) that help in improving the numerical stability of the algorithm [7].

IV. SIMULATION RESULTS

A. *Evaluation of arithmetical complexity*

We evaluate the computational effort for the main loop of each algorithm in order to better compare the two methods R-NOOja and NOOja in the context of real-valued sinusoidal carriers frequency identification. Performance of subspace tracking-type algorithms depends not only on the number r of sinusoids, but also on the dimension n of the input vector $\mathbf{x}_r(t)$.

We obtain the following estimations for the arithmetical complexity of the main loop (where operation means real numbers addition or multiplication):

$$NOOja: \ 18nr + 19n + 24r + 16 \ \ operations/iteration$$

$$R-NOOja: \ 9nr + 21n + 12r + 16 \ \ operations/iteration$$

Even if both algorithms are $O(nr)$, we can clearly see that R-NOOja algorithm requires fewer operations than NOOja method for computing the update of the signal subspace estimate $\mathbf{W}_r$. Further gain in computational burden comes from the use of R-ESPRIT instead of ESPRIT for the values of the frequency estimates. A detailed comparison of these two block methods from complexity point of view may be found in [9].

From extensive simulations, we may state that the overall computational effort for R-NOOja is only about 40% as compared to complex-valued NOOja for the same input vector dimension. We have checked the results with MATLAB `flops` routine.

B. *Algorithms behavior in stationary environments*

We study the statistical properties of both R-NOOja and NOOja algorithms in stationary environments. We are interested to see if the reduction in arithmetical complexity affects the algorithm performance. We present the results obtained for the two algorithms when retrieving two sinusoids of normalized frequencies $f_1 = 0.1, f_2 = 0.2$, embedded in background white noise. We have considered $\mu = 0.5$ and $\gamma = 10$ for both methods in order to achieve best numerical stability.

We calculate the bias and variance of the estimated frequencies for various signal lengths N and for various signal-to-noise ratios. In each case, we run 100 independent simulations. Each time we compute the Cramer-Rao bound (CRB) to verify the accuracy of the estimates.

Table 2
STATISTICAL RESULTS FOR R-NOOJA (n=2r+5)

N	SNR (dB)	Bias f_1	Var f_1	Bias f_2	Var f_2	CRB
		x10^{-4}	**x10^{-4}**	**x10^{-4}**	**x10^{-4}**	**x10^{-4}**
100	0	7.22	61.15	8.16	42.17	3.96
	10	-1.26	15.32	2.05	15.17	1.25
	20	-0.55	4.76	0.46	4.91	0.40
	30	-0.20	1.50	0.12	1.56	0.13
		x10^{4}	**x10^{-4}**	**x10^{-4}**	**x10^{-4}**	**x10^{-5}**
200	0	1.79	51.92	2.62	46.67	13.90
	10	-1.46	15.40	0.59	13.45	4.39
	20	-0.62	4.78	-0.030	4.36	1.39
	30	-0.21	1.50	-0.008	1.38	0.44
		x10^{-5}	**x10^{-4}**	**x10^{-5}**	**x10^{-4}**	**x10^{-5}**
500	0	116.78	52.68	95.45	49.09	3.50
	10	25.54	14.27	10.59	13.57	1.11
	20	6.84	4.48	1.08	4.24	0.35
	30	2.00	1.42	0.12	1.34	0.11

Table 3
STATISTICAL RESULTS FOR NOOJA (n=2r+5)

N	SNR (dB)	Bias f_1	Var f_1	Bias f_2	Var f_2	CRB
		x10^{-4}	**x10^{-4}**	**x10^{-4}**	**x10^{-4}**	**x10^{-4}**
100	0	-8.98	157.03	16.52	90.50	3.96
	10	0.38	20.02	1.71	17.26	1.25
	20	0.60	6.77	0.33	5.66	0.40
	30	0.76	2.17	-0.06	1.81	0.13
		x10^{-4}	**x10^{-4}**	**x10^{-4}**	**x10^{-4}**	**x10^{-5}**
200	0	-7.56	64.34	2.06	59.10	13.9
	10	1.57	22.14	-1.16	18.16	4.39
	20	-0.61	7.35	-0.39	5.86	1.39
	30	-0.21	2.33	-0.13	1.84	0.44
		x10^{-5}	**x10^{-4}**	**x10^{-5}**	**x10^{-4}**	**x10^{-5}**
500	0	-97.99	65.63	164.5	63.89	3.50
	10	-5.54	22.00	-8.37	20.90	1.11
	20	2.47	6.86	5.11	5.92	0.35
	30	0.92	2.19	1.26	1.90	0.11

Tables 2 and 3 present the statistical performances R-NOOja and NOOja algorithms, respectively. We see that R-NOOja overall performs about the same as NOOja at a much lower arithmetical complexity. We also see that both algorithms converge in less than 100 iterations.

IV. CONCLUSIONS

In the present paper we have moved forward to the work presented in [2] and adapted the normalized gradient subspace tracking technique based on Oja learning rule - NOOja [3], [7] to the real-valued signal model. Thus, we derive another novel gradient subspace method, optimized for tracking real sinusoidal carriers in noise. We name this method R-NOOja. The new algorithm uses the real data model. We compare its performances to the complex-valued NOOja algorithm. We conclude that R-NOOja has about the same performances as NOOja in stationary environments, but at much lower computational effort.

It seems that we can further mitigate the major drawback in the use of subspace tracking-type algorithms, their high arithmetical complexity.

This paper follows-up the authors work in [2] in the field of optimizing adaptive subspace tracking methods like [6] for estimating frequencies of real valued sinusoids in noise. This is also the perspective of our future studies.

REFERENCES

[1] K. Mahata, T. Söderström, "ESPRIT-like Estimation of Real-Valued Sinu-soidal Frequencies", *IEEE Trans. on Signal Processing*, vol. 52, No.5, May 2004, pp. 1161-1170

[2] S. Slavnicu, S. Ciochină, "Subspace Method Optimized for Tracking Real-Valued Sinusoids in Noise", *Proc. Signals, Circuits and Systems, 2005. ISSCS 2005. International Symposium on*, Volume 2, 14-15 July 2005, pp. 697-700

[3] E. Oja, "Neural networks, principal components and subspaces," Int. J. Neural Syst., vol. 1, no. 1, pp. 61–68, 1989.

[4] P. Stoica, R.L. Moses, *Introduction to Spectral Analysis*, Prentice Hall, 1997, ISBN: 0-13-258419-0

[5] B. Yang, "Projection Approximation Subspace Tracking", *IEEE Trans. on Signal Processing*, vol. 43, No.1, January 1995, pp. 95-107

[6] K. Abed-Meraim, A. Chkeif, Y. Hua, "Fast Orthonormal PAST Algorithm", *IEEE Signal Proc. Letters*, vol. 7, No. 3, March 2000, pp. 60-62

[7] S. Attalah, K. Abed-Meraim, "Fast Algorithms for Subspace Tracking", *IEEE Trans. on Signal Processing*, vol. 8, No.7, July 2001, pp. 203-206

[8] R. Badeau, G. Richard, B. David, "Sliding Window Adaptive SVD Algorithms", *IEEE Trans. on Signal Processing*, vol. 52, No.1, January 2004, pp. 1-10

[9] K. Mahata, T. Söderström, "Subspace Estimation of Real-Valued Sinusoidal Frequencies", Dept. Inform. Technol., Uppsala Univ., Tech. Rep., Uppsala, Sweden, Jan. 2003

Buletinul Ştiinţific al Universităţii "Politehnica" din Timişoara

Seria ELECTRONICĂ şi TELECOMUNICAŢII
TRANSACTIONS on ELECTRONICS and COMMUNICATIONS

Tom 51(65), Fascicola 2, 2006

Algorithms for Fast Full Nearest Neighbour Search on Unstructured Codebooks: A Comparative Study

Spiridon Florin Beldianu[1]

Abstract – This paper presents several fast nearest neighbor search algorithms for vector quantization on unstructured codebooks of arbitrary size and vector dimension that uses linear projections and variance of a vector. Several new inequalities based on orthonormal Tchebichef moments and projections on the first vectors of the DCT and PCA transformations of an image block are introduced to reject those codewords that are impossible to be the nearest codevector and cannot be rejected by inequalities based on Hadamard Transform, sum and variance, thereby saving a great deal of computational time, while introducing no extra distortion compared to the conventional full search algorithm.

Keywords: vector quantization, fast full nearest neighbor search, image vector quantization, linear projections

I. INTRODUCTION

Vector Quantization (VQ) [1], [2] is an efficient technique for data compression which has been successfully used in various applications involving VQ-based encoding and VQ-based recognition. The response time of encoding and recognition is a very important factor to be considered for real-time applications. The k-dimensional, N-level vector quantizer is defined as a mapping from a k-dimensional Euclidean space into a certain finite set $C=\{C_1,C_2,...,C_N\}$. The subset C is called a codebook and its elements are called codewords. The codeword searching problem in VQ is to assign one codeword to the input test vector thus the distortion between this codeword and the test vector is the smallest among all codewords. Given one codeword $C_j=(c_{j1},c_{j2},...,c_{jk})$ and the test vector $\mathbf{x}=(x_1,x_2,...,x_k)$, the squared Euclidean distortion measure can be expressed as follows:

$$D(C_j,\mathbf{x})=\sum_{i=1}^{k}(c_{ji}-x_i)^2 . \qquad (1)$$

From the above equation, each distortion calculation requires multiplications and $2k-1$ additions. For an exhaustive full search algorithm, encoding each input vector requires N distortion computations and $N-1$ comparisons. Therefore, it is necessary to perform kN multiplications, $(2k-1)N$ additions and $N-1$ comparisons to encode each input vector. The need for a larger codebook size and higher dimension for high performance in VQ encoding system results in increased computation load during the codeword search.

Many researchers have looked for fast encoding algorithms to accelerate the VQ process. These works can be classified into two groups. The first group rely on the use of data structures that facilitate fast search of the codebook such as TSVQ or K-d tree [3] ,[4]. The second group addresses an exact solution of the nearest-neighbor encoding problem on unstructured codebooks. A very simple but effective method is the partial distortion search (PDS) method reported by Bei and Gray [5], which allows early termination of the distortion calculation between a test vector and a codeword by introducing a premature exit condition in the searching process. The equal-average nearest neighbor search (ENNS) algorithm uses the mean value of an input vector to reject impossible codewords [6]. The improved algorithm, i.e., the equal-average equal-variance nearest neighbor search (EENNS) algorithm, uses the variance as well as the mean value of an input vector to reject more codewords [7]. This algorithm reduces computational time further with $2N$ additional memory cells. The improved algorithm termed IEENNS uses the mean and the variance of an input vector like EENNS but develops a new inequality between these features and the distance [8],[9]. The DHSS3 [10] method uses an inequality based on projections on the firsts three axis of ordered Walsh-Hadamard transformation to reject impossible codewords. In [11] is presented a new algorithm based on projections on Tchebichef Moments (also named as Discreete Tchebichef Transform—DTT) vector basis (DTTS), which proves to have a lower search complexity than IEENNS.

In this paper, we will examine the kernel and the complexity search for IEENNS, DHSS3, DTTS algorithms and two new ones based on projections on

[1] Faculty of Electronics and Telecommunications,Telecommunications Department, Bd. Carol 11, 700506, Iasi, fbeldianu@etc.tuiasi.ro

three vector basis of DCT and Karhunen Loeve (KLT) transformations.

II. THE ALGORITHM

A. *IEENNS and DHSS3 algorithms*

The IEENNS algorithm [8] uses two characteristics of a vector, sum and the variance simultaneously. Let $\mathbf{x}=[x_1,x_2,...,x_k]$ be a k-dimensional vector. The sum of vector components can be expressed as $S_{\mathbf{x}}=\sum_{i=1}^{k}x_i$ and the variance as $V_{\mathbf{x}}=\sqrt{\sum_{i=1}^{k}(x_i-S_{\mathbf{x}}/k)}$. The basic inequalities for IEENNS method are as follows: if $\mathbf{y}$ is a codeword and $\mathbf{x}$ is an input vector, the following important inequalities are true:

$$\begin{aligned}&\left(S_{\mathbf{x}}-S_{\mathbf{y}}\right)^2\le kD(\mathbf{x},\mathbf{y})\\&\left(S_{\mathbf{x}}-S_{\mathbf{y}}\right)^2+k\left(V_{\mathbf{x}}-V_{\mathbf{y}}\right)^2\le kD(\mathbf{x},\mathbf{y})\end{aligned}\tag{2}$$

Assuming that the current minimum distortion is $D_{\min}$, the main spirit of the IEENNS algorithm can be stated as follows:
If $(S_{\mathbf{x}}-S_{C_j})^2\ge kD_{\min}$ **then** $D(\mathbf{x},C_j)\ge D_{\min}$ and C_j will not be the nearest neighbor to $\mathbf{x}$; **ElseIf** $(V_{\mathbf{x}}-V_{C_j})^2\ge D_{\min}$ **then** $D(\mathbf{x},C_j)\ge D_{\min}$ and C_j will be rejected; **ElseIf** $(S_{\mathbf{x}}-S_{C_j})^2+k(V_{\mathbf{x}}-V_{C_j})^2\ge kD_{\min}$ **then** $D(\mathbf{x},C_j)\ge D_{\min}$ and C_j will be rejected; **Else** compute $D(\mathbf{x},C_j)$ and if $D(\mathbf{x},C_j)<D_{\min}$ update $D_{\min}=D(\mathbf{x},C_j)$. To perform the IEENNS algorithm, 2*N* values should be computed off-line and stored.
The DHSS3 algorithm [10] utilizes the compactness property of signal energy on transform domain and the geometrical relations between the input vector and every codevector to eliminate those codevectors that have no chance to be the closest codeword of the input vector. It achieves a full search equivalent performance. Let $\mathbf{h}_1$, $\mathbf{h}_2$ and $\mathbf{h}_3$ be the first three orthonormal vectors of ordered Walsh-Hadamard transform. For example if $k=16$, we have:

$$\begin{aligned}&\mathbf{h}_1=[1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1]/4;\\&\mathbf{h}_2=[1,1,1,1,1,1,1,1,-1,-1,-1,-1,-1,-1,-1,-1]/4;\\&\mathbf{h}_3=[1,1,-1,-1,1,1,-1,-1,1,1,-1,-1,1,1,-1,-1]/4;\end{aligned}$$

Denote the axis in the direction of $\mathbf{h}_i(i=1,2,3)$ as the i-th axis. Let $H_i(\mathbf{x})$ be the projection value of an input vector $\mathbf{x}$ on the i-th axis. That is, $H_i(\mathbf{x})$ is the inner product of $\mathbf{x}$ and $\mathbf{h}_i$, and can be calculated as follows: $H_i(\mathbf{x})=\langle\mathbf{x},\mathbf{h}_i\rangle$.
It can been shown that for an input vector $\mathbf{x}$ and for a codeword C_j the following inequality is true:

$$D(\mathbf{x},C_j)\ge\sum_{i=1}^{3}\left|H_i(\mathbf{x})-H_i(C_j)\right|^2\tag{3}$$

To speed up the searching process, all codewords are sorted in ascending order of their projections on the first axis. The elimination process of the DHSS3 algorithm consists of four steps. The firsts three steps are as follows:
If $\left|H_i(\mathbf{x})-H_i(C_j)\right|\ge\sqrt{D_{\min}}$ $(i=1,2,3)$ then C_j will be rejected. Last step is: **If** $\sum_{i=1}^{3}\left|H_i(\mathbf{x})-H_i(C_j)\right|^2\ge\sqrt{D_{\min}}$ then C_j will be rejected; **ElseIf** $D(\mathbf{x},C_j)<D_{\min}$ update $D_{\min}=D(\mathbf{x},C_j)$. To perform the DHSS3 algorithm, 3*N* values should be computed off-line and stored.

B. *Tchebichef Polynomials and Orthonormal Moments*

For a given positive integer (usually the image size), and a value x in the range $[0,M-1]$, the scaled Tchebichef polynomials $t_n(x)$, $n=0.1,...,M-1$, are defined using the following recurrence:

$$t_n(x)=\frac{(2n-1)t_1(x)t_{n-1}(x)-(n-1)\left(1-\frac{(n-1)^2}{M^2}\right)t_{n-2}(x)}{n}\qquad n=2,3,...M-1\tag{4}$$

where $t_0(x)=1$ and $t_1(x)=(2x+1-M)/M$. The above definition uses the following scaled factor [12] for the polynomial of degree n:

$$\beta(n,M)=M^n\tag{5}$$

The set $\{t_n\}$ has a squared-norm given by:

$$\begin{aligned}&\rho(n,M)=\sum_{x=0}^{M-1}\{t_n(x)\}^2=\\&=\frac{M(1-1/M^2)(1-2^2/M^2)...(1-n^2/M^2)}{2n+1}\end{aligned}\tag{6}$$

These polynomials are orthogonal, and by modifying the scale factor $\beta(n,M)$ in (5) as in [13]:

$$\beta(n,M)=\sqrt{\frac{M(M^2-1)(M^2-2^2)...(M^2-n^2)}{2n+1}}\tag{7}$$

we obtain a set of orthonormal polynomials that can be used to define a set of orthonormal moments in (8).

$$T_{m,n}(f)=\sum_{x=0}^{M-1}\sum_{y=0}^{M-1}\hat{t}_m(x)\hat{t}_n(x)f(x,y)\qquad m,n=0,1,...M-1\tag{8}$$

$f(x,y)$ denotes the intensity value of the pixel position (x,y) in the image. It can be easily seen that the recurrence relations given in (4) now change to the following:

$$\hat{t}_n(x)=\alpha_1 x\hat{t}_{n-1}(x)+\alpha_2\hat{t}_{n-1}(x)+\alpha_3\hat{t}_{n-2}(x)\qquad n=2,3,...M-1;\quad x=0,1,2,...M-1\tag{9}$$

where:

$$\alpha_1 = \frac{2}{n}\sqrt{\frac{4n^2-1}{M^2-n^2}},\ \alpha_2 = \frac{(1-M)}{n}\sqrt{\frac{4n^2-1}{M^2-n^2}},$$
$$\alpha_3 = \frac{(n-1)}{n}\sqrt{\frac{2n+1}{2n-3}}\sqrt{\frac{M^2-(n-1)^2}{M^2-n^2}}. \quad (10)$$

The starting values for the above recursion can be obtained from the following equations:

$$\hat{t}_0(x) = \frac{1}{\sqrt{M}}$$
$$\hat{t}_1(x) = (2x+1-M)\sqrt{\frac{3}{M(M^2-1)}}. \quad (11)$$

The squared norm is now $\rho(n,M) = \sum_{i=0}^{M-1}\{\hat{t}_n(i)\}^2 = 1.$

Since the new moment set is orthonormal we can introduce the following theorem which is an inequality between Euclidian distance of two images and sum of squared differences of orthonormal Tchebichef moments of those images.

Theorem: Let f and g be two images with $M \times M$ resolution. Then:

$$\sum_{m=0}^{p\le M-1}\sum_{n=0}^{q\le M-1}\left|T_{mn}(f)-T_{mn}(g)\right|^2 \le D(f,g) \quad (12)$$

where $D(f,g)$ is the squared Euclidian distance between images f and g, and can be defined similar as in (1).

Proof: Since $m,n = 0,1,2...M-1$, the set $\{T_{mn}\}$ is composed by M^2 orthonormal moments. So, $T_{mn}(f)$ can be assimilated with a linear orthonormal transformation of an image f which has M^2 vector basis. A linear orthonormal transformation is a bijective map between two metric spaces which preserves the distances. This property is called isometry, and in this case we can write:

$$\sum_{m=0}^{M-1}\sum_{n=0}^{M-1}\left[T_{mn}(f)-T_{mn}(g)\right]^2 = D(f,g). \quad (13)$$

The left side of (13) is the squared Euclidian distance computed in the output space of the transformation given by Tchebichef moments. Having this equality is obviously that the inequality in (12) always holds.

For example in Fig. 1 are presented the firsts four vector basis of this linear transform for $M = 4$. $T_{00}(f), T_{01}(f), T_{10}(f), T_{11}(f)$ can be computed using the dot product between those vector basis and input image f.

C. *DTTS Algorithm*

For the proposed algorithm we use only firsts three moments [3], namely, T_{pq}, where $(p,q) \in \{(0,0),(0,1),(1,0)\}$. The inequality in (12) becomes now:

$$\frac{1}{4}\begin{bmatrix}1&1&1&1\\1&1&1&1\\1&1&1&1\\1&1&1&1\end{bmatrix} \quad \frac{1}{\sqrt{80}}\begin{bmatrix}-3&-1&1&3\\-3&-1&1&3\\-3&-1&1&3\\-3&-1&1&3\end{bmatrix}$$
$$\frac{1}{\sqrt{80}}\begin{bmatrix}-3&-3&-3&-3\\-1&-1&-1&-1\\1&1&1&1\\3&3&3&3\end{bmatrix} \quad \frac{1}{20}\begin{bmatrix}9&3&-3&-9\\3&1&-1&-3\\-3&-1&1&3\\-9&-3&3&9\end{bmatrix}$$

Figure 1. From left to right and from top to bottom: firsts four vector basis used to compute orthonormal Tchebichef moments T_{00}, T_{01}, T_{10} and T_{11}.

$$(T_{00}(f)-T_{00}(g))^2 + (T_{01}(f)-T_{01}(g))^2 + \\ +(T_{10}(f)-T_{10}(g))^2 \le D(f,g) \quad (14)$$

The proposed searching sequence for a given input image $f \in \mathbf{F} \equiv \{f_1, f_2, \ldots, f_L\}$ can be described as follows:

*

Step 0: For every image codeword C_j, $j = \overline{1,N}$, $T_{00}(C_j), T_{01}(C_j), T_{10}(C_j)$ are computed. The codewords are sorted in the ascending order of $T_{00}(C_j)$. This step is operated off-line. In the following steps the memory for $T_{00}(C_j), T_{01}(C_j), T_{10}(C_j)$ $j = 1,2,...,N$ are ready; Go to step **1**;

For every input image vector $f \in \mathbf{F}$ find the nearest neighbor codevector as follows:

Step 1: $T_{00}(f), T_{01}(f), T_{10}(f)$ are computed; go to step **2**

Step 2: Obtain the tentative matching codeword C_p whose index is calculated by $p = \arg\min_j \left|T_{00}(f)-T_{00}(C_j)\right|$. Calculate the squared Euclidian distortion $D_{\min} = D(f,C_p)$ and set $i = 1$; go to step **3**;

Step 3: ***If*** $p+i > N$ or codeword C_{p+i} to C_N have been rejected go to step **4**; ***Else*** go to step **3.1**;

Step 3.1: ***If*** $\left|T_{00}(f)-T_{00}(C_{p+i})\right| \ge \sqrt{D_{\min}}$ reject the codewords C_{p+i} to C_N and go to step **4**; ***Else*** go to step **3.2**;

Step 3.2: ***If*** $\left|T_{01}(f)-T_{01}(C_{p+i})\right| \ge \sqrt{D_{\min}}$ reject the codeword C_{p+i} and go to step **4**; ***Else*** go to step **3.3**;

Step 3.3: ***If*** $\left|T_{10}(f)-T_{01}(C_{p+i})\right| \ge \sqrt{D_{\min}}$ reject the codeword C_{p+i} and go to step **4**; ***Else*** go to step **3.4**;

Step 3.4: ***If***

$$\left|T_{00}(f)-T_{00}(C_{p+i})\right|^2 + \left|T_{01}(f)-T_{01}(C_{p+i})\right|^2 + \\ +\left|T_{10}(f)-T_{10}(C_{p+i})\right|^2 \ge D_{\min}$$

reject the codeword C_{p+i} and go to step 4; ***Else*** use PDS to find minimum distortion, update $D_{\min} = \min(D_{\min}, D(f, C_{p+i}))$ and go to step **4**;

Step 4: ***If*** $p-i<1$ or codeword C_{p-i} to C_1 have been rejected go to step **5**; ***Else*** go to step **4.1**

Step 4.1: ***If*** $\left|T_{00}(f)-T_{00}(C_{p-i})\right| \geq \sqrt{D_{\min}}$ reject the codewords C_{p-i} to C_1 and go to step **5**; ***Else*** go to step **4.2**;

Step 4.2: ***If*** $\left|T_{01}(f)-T_{01}(C_{p-i})\right| \geq \sqrt{D_{\min}}$ reject the codeword C_{p-i} and go to step **5**; ***Else*** go to step **4.3**;

Step 4.3: ***If*** $\left|T_{10}(f)-T_{01}(C_{p-i})\right| \geq \sqrt{D_{\min}}$ reject the codeword C_{p-i} and go to step **5**; ***Else*** go to step **4.4**;

Step 4.4: ***If***

$$\left|T_{00}(f)-T_{00}(C_{p-i})\right|^2 + \left|T_{01}(f)-T_{01}(C_{p-i})\right|^2 + \\ + \left|T_{10}(f)-T_{10}(C_{p-i})\right|^2 \geq D_{\min}$$

reject the codeword C_{p-i} and go to step **5**; ***Else*** use PDS to find minimum distortion, update $D_{\min} = \min(D_{\min}, D(f, C_{p-i}))$ and go to step **5**;

Step 5: Set $i = i+1$; ***If*** $p+i > N$ and $p-i<1$ or all codewords have been deleted, terminate the algorithm and return the closest codeword for input image vector f; ***Else*** go to step **3.**

*

The complexity reduction is caused to reduction in number of addition and multiplications needed to compute the left side of (11) instead to compute $D(f, C_i)$ in (1). By choosing this searching sequence, experimental results shows that this proposed algorithm is faster than IEENNS and DHSS3 algorithms, in terms of computational complexity.

D. *DCT and PCA based algorithms*

Similar as in C section, we can develop two new algorithms which uses instead of several projections on DTT, three projections on firsts DCT or PCA vector basis. We have to note that in PCA based approach we must previously compute the first three eigen vectors corresponding to the greatest eigenvalues of the covariance matrix of the codebook. Being the fact that the DCT and PCA are orthonormal transformations, the *Theorem* is also true for this two approaches. The new methods are the same as DTT-S except that we replace in (12), (13) and (14) the Tchebichef moments $T_{mn}(f)$ with the projections on the first three basis vectors of the DCT and PCA transformation Also note that in some practical applications additional computation of the eigen vectors for PCA based method, and for some codebooks, can be sometimes prohibitive.

Table 1. Comparison of average Number of Distortion Calculations per Image (4×4) Block

Codebook size	Method	Encoded image	
		Peppers	**Baboon**
128	Full Search	128	128
	PDS	55.65	89.32
	DHSS3	3.97	16.84
	IEENNS	3.59	14.96
	DTTS	**2.34**	**11.85**
	DCT based	**2.32**	**12.01**
	PCA based	**2.28**	**11.43**
512	Full Search	512	512
	PDS	174.34	302.23
	DHSS3	13.09	64.16
	IEENNS	12.30	53.97
	DTTS	**7.01**	**46.17**
	DCT based	**6.96**	**46.23**
	PCA based	**6.81**	**43.82**
1024	Full Search	1024	1024
	PDS	486.23	743.21
	DHSS3	24.65	114.60
	IEENNS	22.95	89.66
	DTTS	**12.92**	**82.01**
	DCT based	**12.85**	**82.08**
	PCA based	**12.08**	**79.30**

III. EXPERIMENTAL RESULTS

The images used in this experiment are 512×512 monochrome with 256 gray levels. An image is partitioned in 4×4 image blocks and the codebook is design using the Linde-Buzo-Gray (LBG) algorithm with Lena image as a training set. The Peppers and Baboon images are used as the test images. The proposed algorithms are compared to the Full Search, PDS [5] , IEENNS [8,9] and DHSS3 [10] algorithms. Table I and II show the average number of distortion computations and the number of operations (multiplications, additions and comparisons) per pixel for various codebook sizes. For the DCT based algorithm the projections are choosen as the first three elements from the matrix of the DCT-2D coefficients, namely 00, 01 and 10, and for PCA based method are choosen the eigenvectors corresponding with the first three eigenvalues in decreasing order.

From Table I, we can see that our methods have the best performance of rejecting unlikely codewords. Compared with IEENNS method, proposed algorithms can reduce the number of distortion calculations by 10% to 44% and the average reduction of operations per pixel needed to encode an image block is 39% for Peppers and 11% for Baboon. Compared with DHSS3, our approaches also reduce the number of distortion calculations by 13% to 50% and the average reduction of operations is 43% for Peppers and 15% for Baboon. Compared with DHSS3 and IEENNS, DTT, DCT-2D and PCA based methods can extract much better the information about spatial orientation of image blocks in k-dimensional space. So, they can better discriminate between images with different features, which will determine an increased number of rejected codewords.

Also note that: **(i)** The complexity search for Peppers is approximatively 20%, 17% and 16% for 128, 512

Table 2. Comparison of average Number of Operations per Pixel

Codebook size	Search Method	Encoded image					
		Peppers			**Baboon**		
		Mult.	Add.	Comp.	Mult.	Add.	Comp.
128	Full Search	128	248	8	128	248	8
	PDS	19.44	52.16	4.67	40.66	96.48	8.52
	DHSS3	5.01	9.99	2.1975	20.9462	40.3312	7.3831
	IEENNS	5.40	11.34	1.73	19.35	36.64	5.72
	DTTS	**2.98**	**6.54**	**1.94**	**15.08**	**30.05**	**6.74**
	DCT based	**2.972**	**6.52**	**1.938**	**15.16**	**31.03**	**6.88**
	PCA based	**2.91**	**6.48**	**1.83**	**14.84**	**29.59**	**6.55**
512	Full Search	512	992	32	512	992	32
	PDS	57.60	147.23	16.28	143.38	339.71	32.98
	DHSS3	16.64	33.42	7.69	79.80	153.47	27.94
	IEENNS	16.05	32.01	6.07	67.12	125.19	21.13
	DTTS	**9.11**	**20.58**	**6.73**	**58.75**	**116.37**	**25.62**
	DCT based	**9.08**	**20.32**	**6.69**	**58.92**	**116.97**	**25.60**
	PCA based	**8.97**	**19.92**	**6.67**	**57.01**	**110.99**	**24.10**
1024	Full Search	1024	1984	64	1024	1984	64
	PDS	104.45	262.21	29.77	263.87	574.32	59.52
	DHSS3	31.49	63.06	14.55	142.69	274.31	49.99
	IEENNS	29.10	57.28	11.36	111.06	207.81	36.81
	DTTS	**16.91**	**38.25**	**12.68**	**98.56**	**197.81**	**39.87**
	DCT based	**16.38**	**38.17**	**12.69**	**99.87**	**198.23**	**39.90**
	PCA based	**16.32**	**37.89**	**11.93**	**94.76**	**189.8**	**37.06**

and 1024 codebook size from the complexity search of the Baboon; **(ii)** PCA based approach seems to be slightly better than DTT and DCT based approaches especially for large codebooks. This is an expected result because PCA is the optimal transform regarding the compaction of the energy. But if we consider the fact that we have to use supplementary computation to obtain the eigen vectors, the performance of the overall PCA based method may have a drawback; **(iii)** The complexity difference between DTT and DCT based algorithms is reduced. An explanation is that the kernels of the DCT and DTT transformation are both derived from orthogonal Tchebichef polynomials. From table I and II we observe that for Peppers image, DCT based method outperforms the DTTS and for Baboon is the opposite case; **(iv)** The average time needed for encoding a specific image also depends on two factors: how complex is the image, which refers to how larger is the entropy of that image (it is clear that Baboon has larger entropy than Peppers) and the machine which implements the encoding algorithm. There are several machines in which a multiplication requires much more time than an addition or a comparison, and are others where the difference is not so significant. Also, the floating or integer point implementation can cause reordering of the performance of the presented methods; **(v)** At last but not at least, the accessing time for the precomputed values can be different on several types of implementations.

In conclusion, the trade-off between those factors may produce a system which spent significant less time than in the exhaustive search.

IV. CONCLUSIONS AND FUTURE WORK

In this paper, some fast-encoding algorithms are presented and new ones are introduced. We have presented a new inequality between Euclidian distance of two image blocks and sum of squared differences of orthonormal Tchebichef Moments (also first three projections on the DCT and KLT transforms). This algorithm uses projections of an image block to eliminate many of the unlikely codewords, which cannot be rejected by other available algorithms. Compared with other available approaches, our algorithm has the best performance in terms of number of distortion calculations and the number of operations per pixel needed to encode a certain image.

Future work will focus on using image blocks with 8×8 resolution and will include higher order Tchebichef Moments in (14), which will reject more codewords that cannot be rejected by presented methods.

REFERENCES

[1] Y. Linde, A. Buzo, and R. M. Gray, "An algorithm for vector quantizer design", *IEEE Trans. Commun.*, vol COM-28, no. 1, pp.84-95, 1980

[2] A. Gersho and R. M. Gray, *Vector quantization and signal compression*, Kluwer Academic Press, Massachusetts, 1990

[3] N. Moayeri, D. L. Neuhoff, and W. E. Stark, "Fine-coarse vectorquantization," *IEEE Trans. Signal Processing*, vol. 39, pp. 1503–1515, July 1991.

[4] V. Ramasubramanian and K. K. Paliwal, "Fast k-dimensional tree algorithms for nearest neighbor search with application to vector quantization encoding," *IEEE Trans. Signal Processing*, vol. 40, pp. 518–531, Mar. 1992

[5] C. D. Bei and R. M. Gray, "An improvement of the minimum distortion encoding algorithms for vector quantization and pattern matching," *IEEE Trans. Commun.*, vol. COMM-33, pp. 1132–1133, Oct. 1985.

[6] S. W. Ra and J. K. Kim, "A Fast Mean-Distance-Ordered Partial Codebook Search Algorithm for Image Vector Quantization," *IEEE Trans. Circuits Syst. II*, vol. 40, no. 9, pp. 576–579, 1993

[7] C. H. Lee and L. H. Chen, “Fast closest codeword search algorithm for vector quantization,” *Proc. Inst. Elect. Eng.*, vol. 141, no. 3, pp. 143–148, 1994.

[8] S. J. Baek, B. K. Jeon, and K. M. Sung, “A fast encoding algorithm for vector quantization,” *IEEE Signal Processing Lett.*, vol. 4, pp. 325–327, Feb. 1997.

[9] J.-S. Pan, Z.-M. L. Lu, and S.-H. Sun, “An Efficient Encoding Algorithm for Vector Quantization Based on Subvector Technique”, *IEEE Trans. Image Processing*, vol. 12, no. 3, March 2003

[10] S.C. Tai, C.C. Lai, and Y.C. Lin, “Two fast nearest neighbor searching algorithms for image vector quantization”, *IEEE Trans. Commun*., vol. 40, pp.1623-1628, Dec.1996

[11] S.F. Beldianu, “A Fast Vector Quantization image encoding using Tchebichef Moments”, *Proceedings of the 4-th International Conference on “Microelectronics and Computer Science”*, vol. 2, september 2005, pp.40-43

[12] R. Mukundan, S.H. Ong, P.A. Lee, “Image Analysis by Tchebichef Moments”, *IEEE Transaction on Image Processing*, vol. 10, No. 9, pp.1357-1364, September 2001

[13] R. Mukundan, “Some Computational Aspects of Discrete Orthonormal Moments”, *IEEE Trans. on Image Processing*, vol. 13, No. 8, August 2004

Buletinul Ştiinţific al Universităţii "Politehnica" din Timişoara

Seria ELECTRONICĂ şi TELECOMUNICAŢII
TRANSACTIONS on ELECTRONICS and COMMUNICATIONS

Tom 51(65), Fascicola 2, 2006

An IP design of the idea cryptographic algorithm

M. A. Ajo[1], G. Fericean[2], M. Borda[2] and V. Rodellar[1]

Abstract – In this paper we introduce a library component implementation of the IDEA cryptographic algorithm that may be used embedded in security applications. The model allows scalability in the number of bits of the plaintext and ciphertext and in the number of keys. The hardware design has been modeled in VHDL portable code resulting in a technology independent soft-core.

Keywords: Reusability, IP core, IDEA algorithm, Cryptography.

I. IDEA ALGORITHM

The IDEA (International Data Encryption Algorithm) block cipher is a symmetric-key algorithm, which encrypts 64-bits plaintext blocks to 64-bit cipher text blocks using a 128-bit key K. The same algorithm is used for both encryption and decryption as it is a symmetric-key encryption system [1], [2], [3], [4]. IDEA has been patented in the U.S. and in several European countries, but the non-commercial use of IDEA is free everywhere. The cryptographic strength of IDEA is summarized by the following characteristics:

- Block length: The block length should be long enough to avoid preferences in the block appearance. The use of a block size of 64 bits is recognized as sufficiently strong.
- Key length: The key length should prevent exhaustive key searches. IDEA uses 128 bits.
- Confusion: The ciphertext should depend on the plaintext and key in a complicated and involved way. The objective is to complicate the determination of how the statistics of the ciphertext depend on the statistics of the plaintext. This goal is obtained by applying the operations of exclusive-OR, addition of integers modulus 2^{16} (65536) and multiplication of integers modulus $2^{16} + 1$ (65537) over two inputs of 16 bits. These three operations are incompatible in the sense that no pair of these three operations satisfies a distributive or an associative law.
- Diffusion: Each plaintext bit and each key bit should influence every ciphertext bit. The spreading out of a single plaintext bit over many ciphertext bits hides the statistical structure of the plaintext. The diffusion is provided by the basic building block of the algorithm denoted as MA (multiplication and addition). This block takes as inputs two 16-bit values derived from the plaintext and two 16-bit subkeys derived from the key and produces two 16-bit outputs. This particular structure is repeated eight times in the algorithm, providing very effective diffusion.

II. COMPUTATIONAL STRUCTURE

IDEA consists of 8 computationally identical rounds followed by a final transformation, as can be seen in Fig. 1. The 64-bit data block is divided into four 16-bit sub-blocks: X1, X2, X3 and X4. These four sub-blocks become the input to the first round of the algorithm. In each round the four sub-blocks are XOR-ed, added and multiplied with each other and with six 16-bit sub-keys ($K_1^{(1)}$… $K_6^{(8)}$). Between rounds, the second and third sub-blocks are swapped. Finally, the four sub-blocks are combined with four sub-keys ($K_1^{(9)}$… $K_4^{(9)}$) in a final transformation block.
In the next sub-sections the structure of a single round and final transformation stages will be described. And also sub-keys generation from the main key will be presented. The structure is described in terms of the basic operations involved in the algorithm and mentioned in the confusion characteristic, that is:

- $\oplus$ Exclusive OR
- $\boxplus$ Modulus addition
- $\odot$ Modulus multiplication

A. *Single round stage*

The basic structure for a single round is illustrated in Fig. 2. Specifically, it shows the structures for the first round. Next rounds have the same structure but with different sub-keys and ciphertext-derived inputs. The

[1] Departamento de Arquitectura y Tecnología de Sistemas Informáticos. Facultad de Informática. Universidad Politécnica de Madrid. Campus de Montegancedo s/n. Boadilla del Monte (28660 Madrid – SPAIN) email: ajo@adtech.es, victoria@pino.datsi.fi.upm.es
[2] Faculty of Electronics and Telecommunications. Technical University of Cluj-Napoca. C. Daicoviciu No. 15 (400020 Cluj Napoca – ROMANIA) email: Gabriel.Fericean@com.utcluj.ro, Monica.Borda@com.utcluj.ro

round, begins with an initial transformation that combines the four inputs sub-blocks (Y1, Y2, Y3, Y4) with four sub-keys (DK_1, DK_2, DK_3, DK_4), by using the addition and multiplication operations. The four outputs of this transformation (D1, D2, D3, D4) are then combined using the XOR operation to form two 16-bit blocks that are the input (D5 and D6) to the MA structure. The MA structure also takes two sub-keys (DK_5 and DK_6) as input and combines these inputs to produce two 16 bits-outputs. Finally, the four output blocks (D1, D2, D3, D4) from the upper transformation are XOR-operated with the obtained outputs (D9, D10) from the MA structure producing the four outputs blocks for this round. After this process, the output blocks Y1-2, Y1-3 are exchanged, so that Y1-1, Y1-3, Y1-2 and Y1-4 are used as input to the next round (in that order) along with the next 6 sub-keys. This procedure is followed for the eight rounds in total giving four output blocks: Y8-1, Y8-3, Y8-2 and Y8-4.

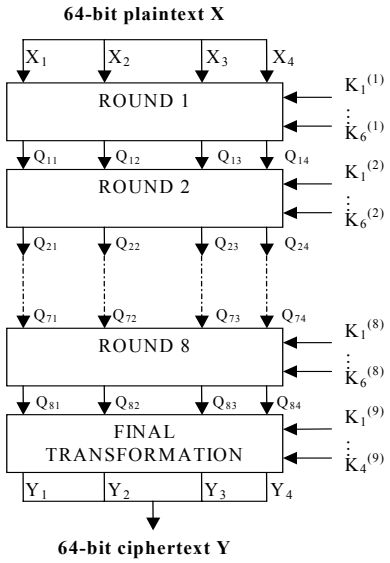


Figure 1. IDEA general structure

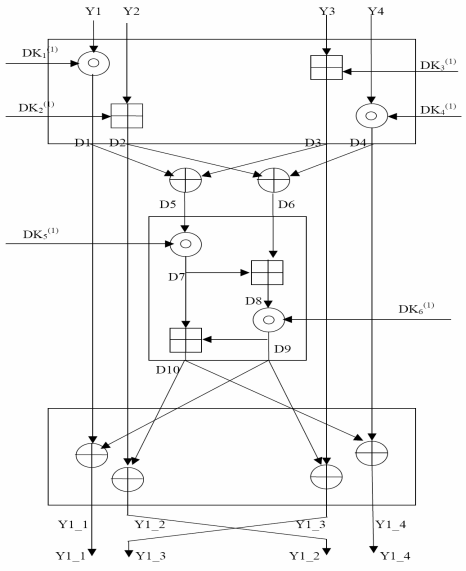


Figure 2. Structure for the first round

B. *Final transformation stage*

The final transformation stage has the same computational structure as the first transformation step of the structure for the first round, as can be seen in Figu. 3. The only difference is that the second and third inputs are interchanged before being applied to the operational units. This has the effect of undoing the interchange at the end of the eight rounds. The reason for this extra interchange is so that decryption has the same structure as encryption. This stage requires only four sub-key inputs, compared to six sub-key inputs for each of the first eight stages. The final four blocks, X1 to X4 are re-attached to form a 64-bit block of plain text. The whole process is repeated for each successive 64-block of ciphertext until all of the ciphertext has been decrypted.

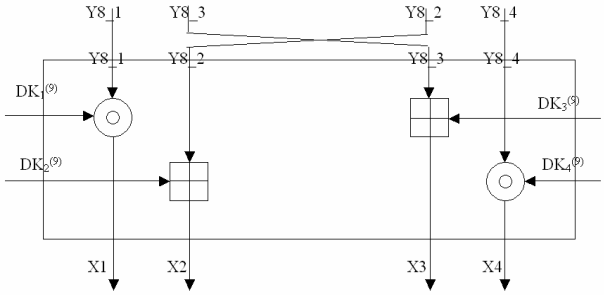


Figure 3. Final transformation structure

C. *Sub-key generation*

As mentioned earlier the algorithm works exactly the same in encryption and decryption modes, with the only difference being that sub-keys used for addition and multiplication are different. Encryption uses sub-keys derived directly from the main key. For the purpose of decryption, inverse sub-keys derived from the encrypted ones, are used.

D. *Encryption*

All 52 sub-keys used in encryption are obtained from the input key K. The scheme for generation is as follows. The first eight sub-keys are taken directly from the key. Then a circular left shift of 25 positions is applied to the key, and the next eight keys are extracted. This procedure is repeated until all 52 keys are generated.

E. *Decryption*

The derivation of the decryption sub-keys from the encryption is as follows: The first four sub-keys of decryption round r are derived from the first four sub-keys of encryption round (10 – r), where the final transformation is counted as round 9. The first and fourth decryption sub-keys are equal to the multiplicative inverse modulus (2^{16} + 1) of the corresponding first and fourth encryption sub-keys. For rounds 2 through 8, the second and third decryption sub-key share is equal to the additive inverse modulus (2^{16}) of the corresponding third and second encryption sub-keys. For rounds 1 and 9, the

second and third decryption sub-keys are equal to the additive inverse modulus (2^{16}) of the corresponding second and third encryption sub-keys. And finally for the first eight rounds, the last two sub-keys of decryption round r are equal to the last two sub-keys of the encryption round (9 - r).

III. IMPLEMENTATION ISSUES AND RESULTS

The general structure described above was modeled in VHDL, according to the restrictions and recommendations for high level behavioral synthesis [5]. The data format size was defined as a generic parameter taking value 16, as default. The design strategy adopted was bottom up, developing in the first instance the basic blocks: XOR, modulus adder and multiplier. The complex blocks of the structure were constructed by using the basic blocks, and they were designed using the available techniques for circuit technology-independent power reduction at the RTL level. These techniques mainly focus on better management of switching activity of the dynamic power consumption. Thus, pipelining and path balancing techniques have been applied to avoid glitch propagation and to balance the delay among basic blocks [6]. The basic blocks, the arithmetic operators, one round structure and the complete system with 8 rounds were simulated, synthesized and tested using the EDA tool Quartus II from Altera. The designs have been implemented on the NIOS development board EP2S60F672C5ES. This provides a hardware platform for developing embedded systems based on Altera Stratix II devices. The results are measured in terms of the resource utilization and delays. Concerning the resource demand, the number of adaptive look-up tables (ALUT) and DSP blocks and registers are given. The measured delays are the delays between an input and an output (I/O t_{pd}) and the key changes to output (K/O t_{pd}).

A. *Arithmetic operators*

The addition operations, both modulus 2 and modulus 2^{16} are implemented by using the XOR and '+' addition defined in the IEEE.STD_LOGIC_1164 and IEEE.NUMERIC_STD libraries. The multiplication modulus $2^{16}+1$ is a slightly more complex operation than the others. It has been implemented following an algorithmic approach that includes unsigned multiplication, addition, subtraction and comparison operations:

1. if operand1 = 0 then operand1 := 2^{16}
2. if operand2 = 0 then operand2 := 2^{16}
3. multiplication := operand1*operand2
4. div := multiplication/2^{16}
5. rem := multiplication mod 2^{16}
6. if (rem>div) then result := rem − div
   else result := rem − div + 2^{16} + 1

The implementation of the multiplication modulus $2^{16}+1$ uses the modulus 2^{16} multiplication synthesized by the development tool. The division and modulus is calculated by taking the upper 16 bits and lower 16 bits of the multiplication result. The addition and subtraction are the operations defined in the IEEE.NUMERIC_STD library for the UNSIGNED data type. The comparisons used in the implementation are UNSIGNED '>' and '=0'.

The key distribution block is implemented using a simple structural design, producing every key for the different stages of the algorithm from the 128 bit encryption key. The following table shows the synthesis results for the basic operators involved in the algorithm. The adders demand a similar amount of physical resources but the adder modulus 2^{16} is around a 1.3 times slower that the adder modulus 2. The multiplier is 1.9 times slower than the adder modulus 2^{16} and 2.5 times slower than the adder modulus 2. These delay factor relations will be of interest when balancing the delays among blocks, as we will see later on.

Table 1. Synthesis results for the basic arithmetic operators

Operation	**ALUs**	**DSP**	**I/O tpd**
Multiplication mod $2^{16}+1$	73	2	24.9ns
Addition mod 2^{16}	16	0	12.9ns
Addition mod 2	16	0	9.8ns

B. *Rounds*

We have implemented a direct version of the IDEA algorithm as it is shown in the dependencies graph in Fig. 2, by using the operators described in the last section. The synthesis results obtained for the INITIAL, MA and XOR blocks, a single round, the output transformation, and the complete IDEA algorithm composed of eight rounds are shown in Table 2. The INITIAL, the MA and final transformation blocks involve the same type and number of operations. They all demand 4 DSP units as they have two multipliers embedded. The number of ALUT's is similar for both the MA and Final transformation blocks and it is higher in the INITIAL because the two adders modulo 2 have been enclosed on this block. Concerning the delay, the MA block is the slowest of all three. This is due to the way in which the adders and multipliers are connected. In this case, the realization of a multiplication is followed by realization of an addition and vice versa. While in the other two blocks the four operations are independent and can be done in parallel.

Table 2. Synthesis results for the direct implementation.

Block	**ALUTs**	**DSP**	**K/O t_{pd}**	**I/O t_{pd}**
INITIAL	210	4	< I/O	27.3ns
MA	164	4	< I/O	42.5ns
XOR	64	0	<I/O	10ns
Single round	407	8	59.2ns	58.3ns
Final transformation	178	4	25.9ns	25.5ns
IDEA	3135	68	418ns	414ns

The delay results obtained for the complete IDEA algorithm implies a processing capacity of:

2.41 Mwords/s * 64bits/word = 154 Mbps
= 19.28 MBytes/s

A global view of the RTL synthesis results for the complete system is shown in Fig. 4.

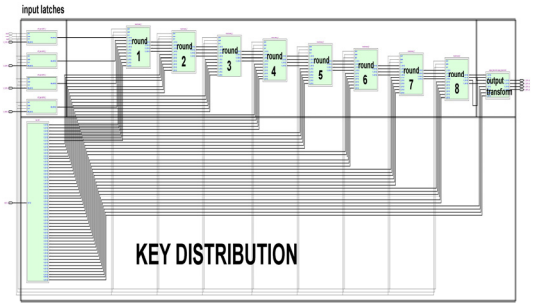


Figure 4. RTL synthesis results

In order to balance the delay among the blocks the action of each round inside the algorithm is decomposed into three pipeline stages, as shown in Fig. 5. The first stage holds the result of the first block and the first two addition operations: D1, D2, D5, D6, D3 and D4, from Fig. 2. The second stage holds the result of the MA block: D10 and D9, and again D1, D2, D3 and D4. The third stage holds the result of the output block: Y1_1, Y1_3, Y1_2, Y1_4. Each pipeline stage is controlled by a clock signal, enable signal, and reset signal, allowing for the control by an external asynchronous state machine.

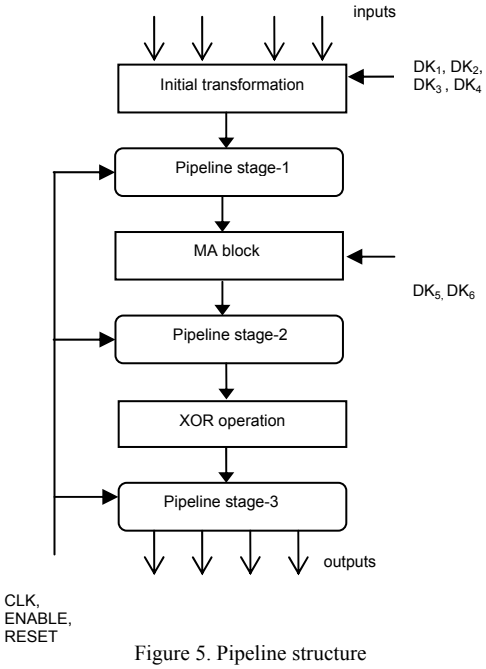


Figure 5. Pipeline structure

Finally, the eight rounds and the final transformation stage are connected together with the key distribution block to build up a complete IDEA encryption or decryption system. That makes a total of 25 pipeline stages (8*3 + 1 for the input latch), which means that 25 cycles are needed to get 64 bits of data from the input to the output. Table 3 shows the synthesis results for the pipelined structure shown in Fig. 5.

Table 3. Synthesis results for the pipeline implementation

Block	ALUT	Registers	DSP	F.max (MHz.)
INITIAL	226	96	4	N/A
MA	180	96	4	N/A
XOR	64	64	0	N/A
Single round	470	256	8	39.37
Final transfor	178	0	4	N/A
IDEA	3.945	2.112	68	37.13

The results in terms of ALUTs and DSP resources are similar to the ones obtained for the direct implementation case in Table 2. But now, additionally, a 2.112 bit register is needed. The max frequency for the complete IDEA algorithm is 37.13 MHz, which means:

37.13 Mwords/s * 64bits/word = 2.376.32 Mbps
= 297.04 MBytes/s.

IV. SUMMARY

In this paper, the design of the IDEA encryption and decryption algorithm oriented toward a high level synthesis for FPGA's implementation has been described. Two different structural approximations have been proposed. The first is based on the direct mapping to natural operations sequence and the second is an improved version introducing three pipelining stages. Both versions demand the same number of ALUT's and DSP blocks. The pipeline version demands the additional amount of 2.112 registers but this allows for a speed improvement of 15.37 times. The pipelined version of this design is suitable for high speed communication, video or audio encryption. The non pipelined version could be useful in secure controllers, and accessing encrypted memory program or data.

REFERENCES

[1] Stallings W. "Cryptography and Network Security: Principles and Practice Second Edition", Prentice Hall, New Jersey, 1999.

[2] Borko Furth, Darko Kirovski, "Multimedia Security Handbook", February 17, 2004.

[3] W. Mao, *Modern Cryptography*, Prentice-Hal, 2004.

[4] A. J. Menezes, V. Oorschot and S. A. Vanstone, *Handbook of Applied Cryptography*, CRC Press, New York 1997.

[5] Michel Keating and Pierre Bricand, *Reuse Methodology Manual: For System-on-a-Chip Designs*. Third Edition. Kluwer Academic Publishers, 2002.

[6] Christian Piguet, Low-Power CMOS Circuits. Technology, Logic Design and CAD Tools. CRC Press 2006.

Buletinul Ştiinţific al Universităţii "Politehnica" din Timişoara

Seria ELECTRONICĂ şi TELECOMUNICAŢII
TRANSACTIONS on ELECTRONICS and COMMUNICATIONS

Tom 51(65), Fascicola 2, 2006

Application for Frequent Pattern Recognition in Telecommunication Alarm Logs

Petru Serafin[1], Alimpie Ignea[2]

Abstract – Based on an algorithm for frequent pattern recognition, this paper presents the implementation of a software application and its respective results in analyzing real-time telecommunication alarm logs. The software application was developed in OMNeT++ (Objective Modular Network Testbed in C++) simulation environment using ACE (Adaptive Communication Environment) toolkit. Different working scenarios are presented in order to simulate extensions of the frequent pattern recognition algorithm: the introduction of time-constraints between alarms and the construction of a Petri net whose transitions are labeled by recognized frequent patterns of alarms.

Keywords: pattern recognition, OMNeT++ simulation environment, ACE toolkit.

I. INTRODUCTION

The volume of information transported by telecommunication networks increases and also the number of specific alarms in telecommunication networks increases. Therefore it became necessary to study different alarm correlation techniques in order to guarantee that all alarms are treated accordingly to the telecommunication networks supervision policy [3]. One of the alarm correlation techniques is to use data-mining into alarm logs to search for possible patterns (chronological sequences of alarms, also called chronicles) that repeat themselves with a certain frequency and therefore may indicate a correlation between the respective alarms. Frequent pattern recognition (chronicle recognition [4]) is used to determine possible alarm correlations but does not determine the relevance of these alarms. It is in the scope of work of the network operator or of the expert-system for network supervision to further analyze alarm correlations and to establish relevance for the recognized patterns.

For the purpose of analyzing real-time alarm logs, such as telecommunication alarm logs, we developed the theoretical aspects for a frequent pattern recognition algorithm, presented in a previous paper [8], and now we present the practical aspects following a software application that implements the given algorithm and its extensions. We also present in this paper different working simulation scenarios that were used for the purpose of assessing some performance aspects of the algorithm [7] and of its extensions by the introduction of time-constraints [2], and Petri net analysis [1].

To implement the software application we used the *OMNeT++* (Objective Modular Network Testbed in C++) simulation environment [10], previously presented in paper [9]. For the real-time communication modules we used *ACE* (Adaptive Communication Environment) toolkit [5], [6], [11]. *ACE* is an open-source software of approximately 135.000 SLOC (Source Lines Of Code).

II. SIMULATION ENVIRONMENT

In the field of telecommunication network analysis there are different simulation environments with specific facilities for addressing different simulation needs. For example, commercial simulation environments such as *COMNET*, *OPNET*, *Hyperformix Workbench*, *Mesquite CSIM* and *Simscript* address industrial simulation needs, while academic simulation environments such as *Smurph*, *NetSim++*, *OMNeT++* address laboratory and non-commercial needs. For our analysis we have chosen the open-source distribution of *OMNeT++* (latest binary *3.2p1* released on January 2006), which is well supported and documented on the respective community web site [10]. We mention though that since last year *OMNeT++* community offers also a commercial version (called *OMNEST*) which addresses industrial simulation needs. *OMNeT++* is a simulation environment based on object-oriented technology and adapted for discrete event systems.

The main advantages of *OMNeT++* are the following:

- It is not necessary to study new specific programming languages for simulation, since it integrates C++ programming code,
- It offers a complete *GUI* (Graphical User Interface) to implement and supervise processes and verify software functionality,
- Simulation is platform-independent and portable on various operating systems, including win32-based and unix-based distributions,

[1] Alcatel Romania, IT S&D Department, 9 Gh.Lazăr, 300081, Timişoara, petru.serafin@alcatel.ro
[2] "Politehnica" University of Timişoara, 2 V.Pârvan, 300223, Timişoara, alimpie.ignea@etc.upt.ro

- Structures can be quickly modified using multiple parameterization facilities, without code impact,
- Predefined classes and libraries are under continuous development and improvement in open-source software development.

Examples of simulations already implemented in *OMNeT++* include queuing systems, communication protocols and other discrete event dynamic systems simulations (*INET Framework*, *Mobility Framework*, *IPv6Suite* etc.).

OMNeT++ offers a modular architecture where components are developed in C++ programming language and then assembled into higher level components using *NED* (Network Description Language). *NED* is implemented as part of the simulation environment and contains many programming facilities and graphical definitions for implementing network topology and parameterization of processes.

The main components of *OMNeT++* are the following:

- Central simulation library,
- *NED* language compiler (*nedc*),
- *GUI* for network topology (*GNED*),
- Simulation interface (*Tkenv*),
- Command-line interface for simulation execution (*Cmdenv*),
- Graphical application for simulation results (*Plove*),
- Supporting toolkits for simulation development.

The modules can be dynamically modified during a simulation in order to take into consideration the evolution of the network topology.

The modules can have an arbitrary number of connections that are developed based on input-output ports. The usage of input-output ports allows further reusability of modules in more complex connections.

The input ports detect the presence of messages and following the validation of some execution conditions other messages are presented at the output ports.

In our simulation, messages that are transported in the network are in fact alarms or alarm patterns that will be transiting the application as tokens.

III. APPLICATION DESIGN

The general architecture of the software application for analyzing telecommunication alarm logs consists of the following specialized modules:

- *Collector* module – with the purpose of reading alarm logs using a specific collector interface with the network elements,
- *Pattern Recognition* module – with the purpose of generating candidate patterns, calculating pattern frequency and retaining frequent patterns,
- *Pattern Analysis* module – to analyze collected and generated data in order to consolidate results.

Frequent patterns of alarms that are discovered in the recognition process are presented individually to the operator to further analyze alarm correlation.

Fig. 1 presents the general architecture of the software application for frequent pattern recognition:

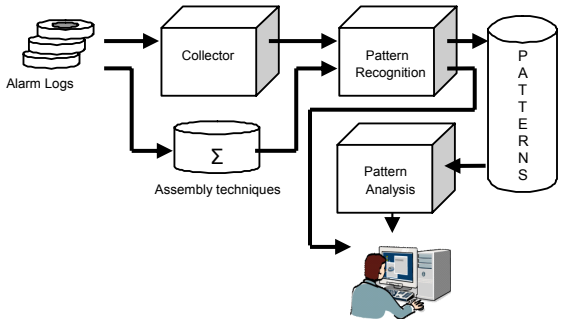


Fig. 1. General architecture for frequent pattern recognition

The detailed architecture of the software application contains the functional components and sub-modules. Our functional implementation of the pattern recognition process is presented in *Fig. 2*:

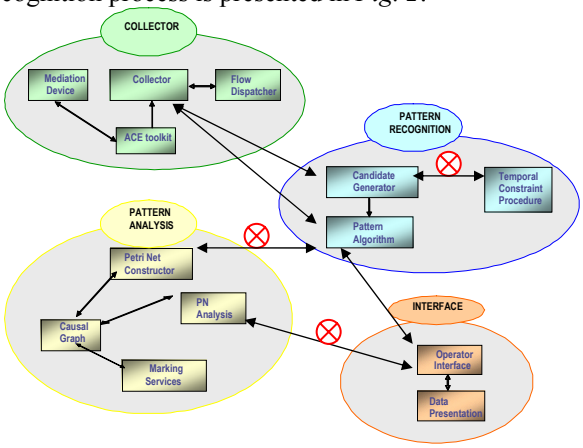


Fig. 2. Detailed architecture for frequent pattern recognition

Breakpoints are represented in *Fig. 2* by the means of the ⊗ symbol and will be used in determining the different simulation scenarios, which will be detailed in the next paragraph.

A brief description of the modules and their functionality is necessary in order to understand the simulation scenarios we will later use.

The *Collector* module is located at the entry point of the application and has the main role of mediation and flow dispatcher between the network elements and the *Pattern Recognition* module. The software implementation of the *Collector* module can be distributed in alarm concentrator units or centralized in a network supervision unit. In our *OMNeT++* implementation we centralized alarm collection in a central unit. The internal architecture of the *Collector* module uses the *ACE* toolkit as a library of platform-independent adaptive communication functions. *ACE* toolkit has the advantage of being portable on different operating systems, contributing to the overall portability of the software application. Different network supervision systems are based on different operating systems and therefore using a common library is important for reusability aspects.

Alarm data is collected by pull transfer mode which is a synchronized extraction of data block piloted by the *Collector* module. Generally the communication protocol between network elements and *Collector* module is constructor dependent. Our simulation uses *FTP* (File Transfer Protocol) to retrieve buffered alarm logs. Flow dispatcher further adapts and negotiates alarm blocks transfer through the upper level of the application. Alarm messages are transmitted and consumed by the software modules under the form of tokens.

Pattern Recognition module realizes the algorithm in its initial description: based on some assembly techniques it generates candidate patterns and then applies a formula for frequency calculation and retain only the frequent patterns to be presented to the operator and/or to the *Pattern Analysis* module.

Different assembly techniques may be used to determine patterns, depending on the prerequisite relations between alarms.

Serial assembly may be used if there is no ordering between alarms, neither by priority nor by chronology. This generates sequences of unordered alarms. For example, a sequence of alarms *(a,b,c)* serial assembled with a repeating alarm *b* results in the sequence *(a,b,c,b)*.

Parallel assembly takes into consideration a certain priority between alarms, dictated by network supervision policy. This generates sequences of ordered alarms. For example, a sequence of alarms *(a,b,c)* parallel assembled with a repeating alarm *b* results in the sequence *(a,b,b,c)*. This presumes that network supervision policy considered that *a* alarm has priority over *b* alarm, and *b* alarm has priority over *c* alarm.

Once the candidate patterns are generated, the frequent pattern recognition algorithm calculates the occurrence frequency of the candidate pattern using expression (1):

$$f_{\min}(p) = \left[\min_{p \in L} \left(\frac{a_i \in L}{a_i \in p} \right) \right] \quad (1)$$

where p is the candidate pattern (included in the L alarm log) and a_i is the generic term for alarm occurrences included in this pattern (i being the alarm index in the pattern).

The algorithm then selects and retains only frequent patterns ($f \geq f_{\min}$) to be further analyzed.

To explain the pattern recognition algorithm, we consider the alarm log given by expression (2):

$$L(a,b,c) = \left\{ aca \begin{matrix} b \\ c \end{matrix} abcc \right\} \quad (2)$$

This considered alarm log *L(a,b,c)* contains occurrences of alarms *a*, *b* and *c* (observe that at a certain time alarms *b* and *c* occur simultaneously, which is represented by a superposition of those alarms).

At each step, the algorithm generates candidate patterns of superior order, starting from the elementary order (see *Fig. 3* and *Fig. 4*). Then there is a frequency calculation based on expression (1). To explain the expression (1), we may calculate the occurrence frequency for pattern *(a,c,c)* in the alarm log *L* given by expression (2):

$$f_{\min}(a,c,c) = \left[\min_{a,c \in L} \left(\frac{a \in L}{1}, \frac{c \in L}{2} \right) \right] = 2 \quad (3)$$

Serial assembly over *L* alarm log with a given minimal frequency $f_{\min}=2$ has the following results, represented in *Fig. 3*:

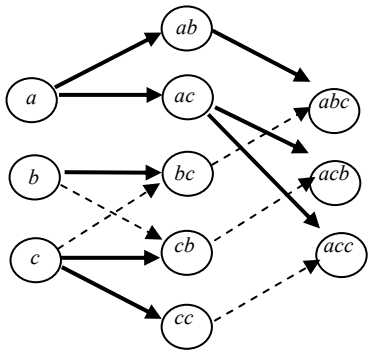


Fig. 3. Frequent patterns recognized in serial assembly

Given the same alarm log and frequency, parallel assembly results in the following frequent patterns, represented in *Fig. 4*:

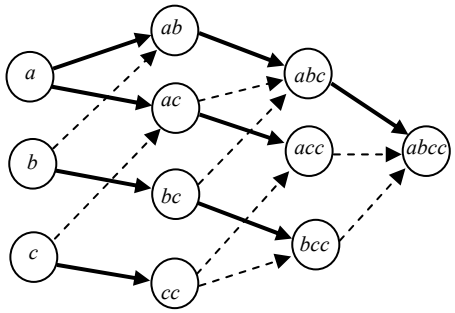


Fig. 4. Frequent patterns recognized in parallel assembly

The results of the frequent pattern recognition algorithm presented previously in *Fig. 3* and *Fig. 4* are based on simple hypothesis regarding assembly techniques and do not take into consideration temporal constraints between alarms. Therefore, the first major extension of the algorithm consists in defining and using temporal constraints in the pattern recognition process.

Temporal constraints between alarms are introduces by the following notions: given an alarm log that contains alarms *a* and *b*, the temporal constraint

between *a* and *b* is the superior limit of the temporal distance between *a* and *b* with respect to their occurrences in the pattern. For example, if we consider the occurrence $[(a,t_a)(b,t_b)]$ the temporal distance would be $T(a,b)=t_b-t_a$.
We define a temporal constraint parameter *c(T)* that is the ratio between *a* and *b* occurrences that verify expression *T* over the total number of *a* and *b* occurrences in the pattern:

$$c(T)=\frac{\left|\left\{\left[(a,t_a)(b,t_b)\right]\in p_{ab}, t_b-t_a=T\right\}\right|}{\left|p_{ab}\right|} \quad (4)$$

Following the introduction of the temporal constraints and the parameter in expression (4), we construct the temporal constraint procedure as follows:

Procedure *Tconstraint* $(a,b,c_{\min}(T))$;
/* Temporal constraint calculations */
Input
$p_{ab}=\left\{\left[(a,t_a^i)(b,t_b^i), i=1..n\right]\right\}$ /* pattern */
$c_{\min}(T)$ /* given minimal constraint */
Output
$C(T)$ /* constr. set verifying $c_{\min}(T)$ */
{
/* Initialize constraint set */
$C(T)=NULL;$
/* Initialize constraint set space */
$S=\left\{t_b^i-t_a^i \middle| i=1..n\right\};$
/* Calculate temporal constant k */
$k=\left[c_{\min}(T)\cdot n\right];$
/* Sorting temporal space */
Sorting $S=\left\{x_1\le\ldots\le x_n\right\};$
/* Composing and verifying $c_{\min}(T)$ */
For $i=1; i\le n-k+1; i++$
For $j=i+k-1; j\le n; j++$
$C(T)=C(T)\cup\left\{x_i,x_j\right\};$
/* Return constraint set */
Return $C(T)$;
}

The introduction of the temporal constraint procedure in the *Pattern Recognition* module further refines the recognized patterns. A situation where temporal constrains show their necessity is if the alarm log contains occurrences of a pattern in a relatively closed time frame and then also contains the same pattern detected with a very large time frame. To speed up the algorithm we define a minimal time frame during which patterns may be recognized and so we will not need to memorize a pattern once it was already recognized. This scenario will constitute one of the performance tests of the algorithm; detailed results are presented in the following paragraph.
Once frequent patterns are recognized we want to perform a first analysis of these patterns, related essentially to finding consequent patterns or patterns that include each other. One of the possible approaches for this analysis is the construction of a Petri net which transitions are labeled with the previously recognized frequent patterns, and then we want to analyze the marking situations in this Petri net. Mixing pattern recognition with Petri net assembly is a first step toward *Pattern Analysis* and it provides important information about the recognized patterns.
One of the main advantages of the *Pattern Analysis* module is that it operates almost independently from the *Pattern Recognition* module. Almost independently because it takes inputs from the recognition algorithm during the assembly of the Petri net and then helps operate on the recognition algorithm. Petri net simply provides results of the eventually consequent patterns and therefore simplifies some calculations of higher level candidate patterns during the algorithm. The theoretical bases of mixing Petri net assembly and pattern recognition are detailed in [7].

IV. PERFORMANCE RESULTS

With the previous considerations, we constitute a list of scenarios activating or deactivating functional sub-modules of the software application.
The first scenario (further referred as *Scenario 1*), consists of a simple execution of the pattern recognition algorithm, without temporal constraint procedure and without activating *Pattern Analysis* module. This provides primary results that can be compared with next scenarios.
The second scenario we use (further referred as *Scenario 2*), consist of the activation of temporal constraint procedure during the pattern recognition algorithm. Referring to *Fig. 2*, *Scenario 2* is obtained by activating the ⊗ symbol between the *Candidate Generator* procedure and the *Temporal Constraint* procedure. As we expected, the introduction of the algorithm does filtrate some patterns that are recognized rather late with respect to a given time frame. This leads to better performance of the overall software application.
The third scenario (noted *Scenario 3*), activates the independent module of Petri net assembly and analyses possible inconsistencies between the recognized frequent patterns. Therefore some frequent patterns will not be presented to the operator since they are included in other frequent patterns. Referring to *Fig. 2*, *Scenario 3* is obtained by activating the two ⊗ symbols that connect the *Pattern Analysis* module to the software application. As expected, this scenario leads to better performance of the pattern recognition.

All scenarios were simulated over the same input alarm log, in order to preserve the possible comparative arguments between the scenarios.
Considering a recorder telecommunication alarm log of 3000 occurrences of 25 types of alarms, we start by executing simulations at given minimal frequencies. For example, we chose 25, 50, 100, 250 and 500 as minimal frequencies for our calculations.
For each considered frequency we then execute the simulation and memorize or calculate following data:

- Frequent alarms,
- Generated candidate patterns,
- Frequent patterns,
- Simulation execution time.

For example, *Table 1* contains results for the simulation execution of *Scenario 1*:

Table 1

Patterns Frequency	*Frequent Alarms*	*Candidate Patterns*	*Frequent Patterns*
25	24	5817	366
50	17	4905	108
100	9	2892	28
250	3	838	11
500	1	78	0

As expected, by increasing the frequency we obtain less frequent patterns and frequent patterns results are refined by the simulation scenarios.
The synthesis graph showing frequent patterns evolution in relation to given minimal frequencies is presented in *Fig. 5*:

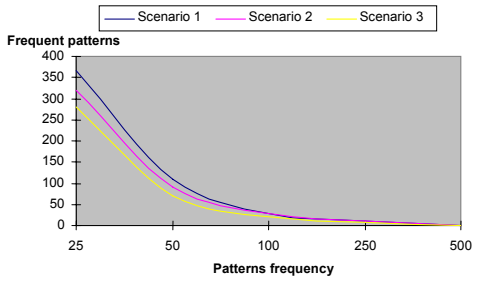


Fig. 5. Frequent patterns in simulation scenarios

Simulation time decreases by increasing minimal patterns frequency, which is explained by the fact that fewer candidate patterns are generated and calculated as they do not verify minimal frequency condition. At the extreme cases, if the desired patterns frequency is too high then the algorithm may stop at the first step of calculating single alarm frequencies. Vice-versa, by selecting a low frequency more and more candidate patterns verifies the minimum frequency and therefore the calculations become time-consuming and the simulation time increases.
Concerning the simulation scenarios' execution time, we collected the following data that is presented in *Table 2*:

Table 2

Patterns Frequency	*Scenario 1*	*Scenario 2*	*Scenario 3*
25	13:27	12:51	11:34
50	03:10	02:50	02:29
100	00:44	00:42	00:35
250	00:28	00:25	00:22
500	00:04	00:04	00:03

The synthesis graph showing simulation scenarios execution time in relation to given minimal frequencies are presented in *Fig. 6*:

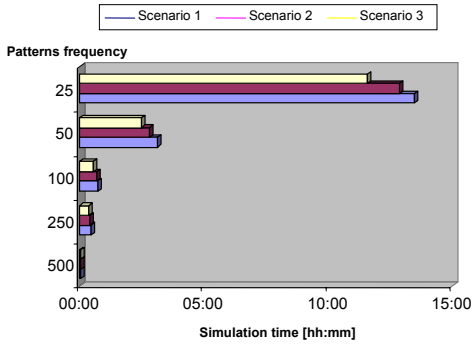


Fig. 6. Simulation scenarios execution time

Some comments of the performance results are necessary before concluding on the overall software application. First we notice that the implementation of the algorithm's extensions is improving the simulation time and also improving the quality of the solution (less frequent patterns are recognized but the relative relevance of these frequent patterns is greater, either because a time frame for the recognition process was defined or because inconsistent patterns were eliminated from the final solution).
Performance improvements presented above demonstrate the refinement of the final solution by the means of eliminating some intermediate solutions to reach a better final solution to be presented to the network operator.
Further simulations on different alarms logs produced equivalent results, depending on the topology of the alarm logs. For example, the recognition algorithm produces faster final solutions when applied over a simpler alarm log with fewer occurrences of alarms. On the contrary, when applied over a more complex alarm log that contains more occurrences of alarms, the recognition algorithm takes longer to produce both intermediate (candidate patterns) and final (frequent patterns) solutions.
Another aspect of these performance results is that it proves that the recognition algorithm itself can be extended with the help of theoretical contributions and mixing other data analysis techniques to the recognition process. Theoretical improvements include the consideration of a certain time frame limitation when recognizing patterns. This is expressed mathematically by the introduction of time

constraints between alarms and the physical application of these constraints is to filter out late occurrences of alarms in the considered alarm log.
Data analysis techniques that may help obtain a better final solution include the Petri net assembly. As demonstrated in our software application implementation, a dedicated module that constructs Petri net and then provides a short analysis of the resulting topologies increases the performance of the overall application.
It is important to mention that only some preliminary analysis was done with the help of the Petri net assembly, only for the purpose of demonstrating the possible application of this method for the scope of pattern recognition. Based on this support we may consider other methods for the scope of obtaining a better final solution of the presented algorithm.

V. CONCLUSIONS

The main outcome of the software application developed for frequent pattern recognition in alarm logs is that it proved a feasible implementation of the theoretical aspects of the recognition algorithm and some of its extensions.
Using generic project management techniques we developed the software application to support various possible simulation scenarios for the purpose of demonstrating value-added possible extensions of the recognition algorithm.
Beside the performance aspects presented in the previous paragraph, there are some interesting results about the frequent patterns themselves. For example, in a real-time situation analysis, we managed to detect a pattern that was not taken into consideration by the network operators, since it was collateral to the telecommunication network supervision policy: it was detected that an auxiliary power supply interruption caused a sequence of alarms, starting from an over-heating alarm and leading to a pattern of telecommunication-related alarms. The explanation was simple: the auxiliary power supply connected the cooling system and therefore its interruption caused cooling system malfunction and finally lead to telecommunication equipment alarms. Generally this kind of sequences of alarms demonstrates the interest in pattern recognition for the telecommunication alarm logs: it proves that some of the recognized pattern may be useful in network maintenance and supervision.
The most important factor in the analysis of frequent patterns is to focus on the initial alarm in the sequence of alarms. In most cases, the initial alarm represents the primary cause of the defect that is being signalized to the network operator. However, in real-time telecommunication systems, alarms do not always appear to the supervision network in the order in which they were produces in the network. This is caused mainly by alarm propagation delays that occur in telecommunication networks.
One of the possible solutions to the problem of considering propagation delays is to register original occurrence time in the alarm logs (and to sort the alarm log in the chronological order of appearance) or to accept larger time slots which will induce the possibility of alarms that appear to be simultaneous in the mathematical representation prior to the application of the pattern recognition algorithm.
Experts in telecommunication network supervision systems that consulted our software application concluded that frequent pattern recognition is useful in networks supervision and has potential towards further development of expert-systems applied to this field of expertise. Also, it was observed that the pattern recognition algorithm and its proposed extensions are theoretically applicable to other fields of expertise such as electrical energy network supervision or other event correlation systems analysis.

REFERENCES

[1] A. Aghasaryan, C. Dousson, "Mixing Chronicle and Petri Net Approaches in Evolution Monitoring Problems", *Proceeedings of the 12th WPD (Workshop Principles of Diagnosis)*, pp.1-7, San Sicario, March 2001
[2] F. Fessant, C. Dousson, F. Clérot, "Mining on a telecommunication alarm log to improve discovery of frequent patterns", *Industrial Conference on Data Mining (ICDM)*, Leipzig, July 2003
[3] G. Fiche, G. Hébuterne, "Trafic et performances des réseaux de télécoms", *Ed.Hermes-Science, Groupe des Ecoles de Telecommunication & Lavoisier,* Paris, 2003
[4] B. Guerraz, C. Dousson, "Chronicle Construction Starting from the Fault Model of the System to Diagnose", *International Workshop on Principles of Diagnosis,* pp.51-56, Carcassonne, 2004
[5] S. Hudson, J. Johnson, U. Syyid, "The ACE Progammer's Guide", *Ed.Addison-Westley*, October 2003
[6] D. Schmidt, S. Hudson, "C++ Network Programming : Mastering Complexity with ACE and Patterns", Volume 1, *C++ In-Depth Series, Bjarne Stroustrup, Ed. Addison-Westley*, December 2001
[7] P. Serafin, "Contribuţii la analiza alarmelor în reţelele de telecomunicaţii", *Ph.D. Thesis, "Politehnica" University of Timişoara,* pp.91-134, December 9, 2005
[8] P. Serafin, "Algorithm for Frequent Pattern Recognition in Telecommunication Alarm Logs", *Scientific Bulletin "Politehnica" University of Timişoara, Transactions on Electronics and Communications*, Tom 50 (64), Fascicola 1, pp.30-33, September 22, 2005
[9] P. Serafin, "Network Simulation Using OMNeT++ Environment", *Scientific Bulletin "Politehnica" University of Timişoara, Transactions on Electronics and Communications*, Tom 49 (63), Fascicola 1, pp.407-411, Symposium of Electronics and Telecommunications, Timişoara, October 22-23, 2004
[10] http://www.omnetpp.org, OMNeT++ (Objective Modular Network Testbed in C++) community web site
[11] http://www.riverace.com, ACE (Adaptive Communication Environment) web site

Buletinul Ştiinţific al Universităţii "Politehnica" din Timişoara

Seria ELECTRONICĂ şi TELECOMUNICAŢII
TRANSACTIONS on ELECTRONICS and COMMUNICATIONS

Tom 51(65), Fascicola 2, 2006

Contributions in Recursive Filtering for B-spline Interpolation in Signal Processing

Liliana Stoica[1]

Abstract –The problem of interpolation a set of data is an old one, but the demanding of flexibility and high speed in operating on-line and in real time processing need to find new methods and improve the old ones [1]. The main properties of B-spline functions offer the possibility to implement algorithms of interpolation in a faster and optimal manner. A function can be represented by B-spline functions with a set of coefficients. For interpolative signal reconstruction it is necessary to calculate those coefficients. In this paper, for cubic spline interpolation it is analyzed a known algorithm and some of his deficiencies. Also there are relieved some possibilities for developing new algorithms that could eliminate those problems. It is presented another way to determine the initial coefficients by using the polynomial representation on short intervals of the spline function and his derivatives. Based on this results are made several observations for further use in improving the algorithm.
Keywords: interpolation, B-spline functions

I. INTRODUCTION

In many applications are given some samples of a signal and it is necessary to estimate the values between them. The interpolation problem it is an actual matter although great mathematicians with hundred years ago approached it. This interpolation process supposes to find a function that can approximate the given data. Usually, the signals in digital signal processing are represented by equally spaced samples. For interpolation can be used the spline functions with uniform knots and unit spacing. Polynomial spline functions have been used in many applications because of their main properties: continuous piecewise polynomial of degree n with derivatives up to order *n*-1. Another advantage is that a spline function can be obtained as a linear combination of shifted B-spline functions and a set of coefficients. The B-splines are maybe the simplest functions and the most used of them are cubic splines. The section II presents some properties of the B-spline functions and the problem of spline interpolation.

Having a set of data, to determine the interpolation function it is necessary to calculate the spline coefficients. In their work Michael Unser and his team give an algorithm to find those coefficients, algorithm that need some boundary conditions for calculating initial coefficients [3], [4], [5]. In section III is presented their algorithm that use simple digital filter techniques.

By section IV are analyzed some results obtained for known signals. Samples from sine and cosine functions, from a straight line are used to calculate the B-spline coefficients and then the interpolated values for a specific interpolation factor.

In section V it is presented a new approach for calculating initial B-spline coefficients. The solution does not need to perform an extension on *Z* for the input signal function. There are presented some results in comparison with the other algorithm.

II. B-SPLINE FUNCTIONS

The spline functions are piecewise polynomials of degree n with continuity of the spline and its derivatives up to order (*n*-1) at the knots [2]. In this work are used only functions with uniform knots and unit spacing.

A spline function $f^n(x)$ is uniquely characterized by the B-spline coefficients c(k) , where:

$$f^n(x) = \sum_{k \in Z} c(k) \beta^n(x-k) \tag{1}$$

$\beta^n(x)$ is the B-spline function of degree n constructed from the (*n*+1) -fold convolution of a rectangular pulse β^0:

$$\beta^0(x) = \begin{cases} 1, & -1/2 < x < 1/2 \\ 1/2, & |x| = 1/2 \\ 0, & otherwise \end{cases} \tag{2}$$

For $n = 3$ we have the cubic B-spline function which is often used for performing high-quality interpolation [3]:

[1] "Politehnica" University of Timisoara, Faculty of Electronics and Telecomunication, Bd. V. Pârvan No. 2, 300223 Timişoara, e-mail: liliana.stoica@etc.upt.ro

$$\beta^3(x)=\begin{cases}2/3-|x|^2+|x|^3/2, & 0\le|x|<1\\ (2-|x|)^3/6, & 1\le|x|<2\\ 0, & 2\le|x|\end{cases} \quad (3)$$

The B-spline interpolation problem traditionally it is resolved by using matrices and standard numerical techniques. The algorithms are long and are necessary many operations to determine the solutions. A faster way to resolve the problem is to use simpler digital filter technique. Unser use the discrete B-spline functions and define the indirect and direct B-spline transforms [3], [4]. The B-spline coefficients can be obtained by linear filtering. Having a set of data the spline coefficients are calculated such that the function goes through the data points exactly. The interpolated values of the signal are obtained also by digital filtering.

III. SPLINE INTERPOLATION ALGORITHM

To perform an interpolation process for a set of N samples it is necessary to find the interpolation function in (1).The input data are lettering by {$s(0)$, $s(1)$, $s(2)$,…, $s(N-1)$}. For that we have to calculate the B-spline coefficients. A solution proposed by M. Unser [3], [4] is to apply for the input signal a digital filter, called direct B-spline filter:

$$(b_1^3)^{-1}(k) \quad \leftrightarrow \quad [B_1^3(z)]^{-1}=\frac{6}{z+4+z^{-1}} \quad (4)$$

$b_1^3(k)$ is the discrete B-spline function of degree n=3 (cubic). It is said that the coefficients can be calculated by "direct B-spline transform".
This filter is implemented by 2 filters: first a causal filter and the second anti-causal. A recursive algorithm it is given to calculate the B-spline coefficients:

$$c^+(k) = s(k) + z_1\, c^+(k-1), \quad k = 1, \dots, N-1 \quad (5)$$

$$c^-(k) = z_1\,(c^-(k+1) - c^+(k)), \quad k = N-2, \dots, 0 \quad (6)$$

where: $z_1=-2+\sqrt{3}$ and $c(k) = 6c^-(k)$.
For that it is necessary to establish some initial conditions. To recover exactly the initial samples by convolving $c(k)$ with $b_1^3(k)$ are used mirror-symmetric boundary conditions: $s(k) = s(l)$ for (k+l)mod (2N-2) = 0. The resulting signal is periodic with period 2N-2. For the first recursion we have the next initialization:

$$c^+(0)=\sum_{k=0}^{+\infty}s(k)z_1^k \quad (7)$$

Practically is not efficient to calculate this. For the periodic signal resulted by mirroring the input data the relation became:

$$c^+(0)=\frac{1}{1-z_1^{2N-2}}\sum_{k=0}^{2N-3}s(k)z_1^k \quad (8)$$

The author propose for using in practice the formula (9), with $k_0 > \log \varepsilon / \log |z_1|$, where ε is the desired level of precision [3]. In the literature it is not pointed out the exact signification and how ε have an effect on the calculating the coefficients process. By taking different values for ε it was observed that only a few coefficients are affected at the beginning and the end of the data string. For a given signal the intermediate values are the same in cases of different values for ε.

$$c^+(0)=\sum_{k=0}^{k_0}s(k)z_1^k \quad (9)$$

For the second recursion it is used:

$$c^-(N-1)=\frac{z_1}{(z_1^2-1)}(c^+(N-1)+z_1c^+(N-2)) \quad (10)$$

Having the B-spline coefficients, the signal interpolation by an integral factor m it is completed by $f^n(x/m)$, denoted by $f_m^n(x)$. From (1) we obtain:

$$f_m^n(x)=\sum_{k\in Z}c(k)b_m^n(x-km) \quad (11)$$

This operation is called "indirect B-spline transform" and it is implemented by digital filtering [5], [6].

IV. THE ANALYZIS OF THE UNSER'S ALGORITHM

The presented algorithm was implemented and there were calculated the B-spline coefficients and the recovered signal in some particular cases. There are used more arrays of initial data with N samples of sine or cosine functions.
The samples $s(k)$ are defined by $s(k)=\sin(2\pi k/M)$ or $s(k)=\cos(2\pi k/M)$ with k=0, …, N-1, were M=12, 50, 100 or 120. The N samples represent 1, 3 or 5 periods of the functions for different sampling frequencies. It was analyzed also the case of a straight line. The value for k_0 in (9) was selected to k_0=7. From the results there were done several observations.
The B-spline coefficients follow the signal variation and are close to the samples values. In figure 1 are presented the differences between the coefficients and the input data values for $s(k)=\sin(2\pi k/120)$ with N=361. In all studied cases the values of the B-spline coefficients are almost equal with the samples values. In the case of cardinal spline functions the coefficients are exactly the input data. But those functions are no longer compactly supported. There is lost the advantage of calculating any value of the interpolation polynomial using maximum (n+1) B-spline functions

(for the sampling points are necessary only n functions).

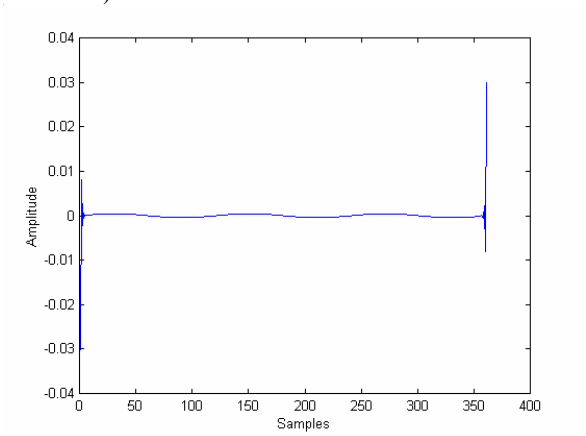


Fig.1Diferences between B-spline coefficients and input data

For sine and cosine functions have been calculated the B-spline coefficients for different sampling frequencies. There were taken one period of the signal and then three periods. In each case were obtained different values for the coefficients. Generally $c(k)$ depends on the signal sampling frequency (sampling unit). In Table 1 are compared the coefficients $c(k)$ for different sampling frequencies in the case of sine function: $\sin(2\pi k/M)$. The values are obtained on the same points α of the input function characteristic.

Table 1. Coefficients for $s(k)= \sin(2\pi k/M)$, $k_0=7$

α	M	First period		Second period		Third period	
		k	$c(k)$	k	$c(k)$	k	$c(k)$
0	12	0	-0.3021395140	(M+0)	-0.0000000413	(2M+0)	0.0000000413
	120	0	-0.0302443872	(M+0)	0	(2M+0)	0
π/6	12	1	0.6043309293	(M+1)	0.5233729016	(2M+1)	0.5233727360
	120	10	0.5002284575	(M+10)	0.5002285152	(2M+10)	0.5002285152
π/3	12	2	0.8848157966	(M+2)	0.9065084347	(2M+2)	0.9065090142
	120	20	0.8664212038	(M+20)	0.8664212038	(2M+20)	0.8664212038

Table 2 present the same coefficients, but for the cosine function: $\cos(2\pi k/M)$. The values depend not only of the shape of the function, but also of the sampling frequency.

Studying the case of sine and cosine functions it is clearly that for great sampling frequencies the values of $c(k)$ and $s(k)$ are very close (differences of maximum 10^{-4} order for M=120). At smaller sampling frequency the differences are increasing (10^{-2} for cosine and 10^{-1} for sine functions when M=12).

In the case of the odd functions by performing the mirror extension [5] for the input signal, the functions derivatives are no longer continues.

The coefficients obtained for one or more periods are not the same. For the second period, the coefficients are different in comparison with the first period, but are almost equals with the ones in the next periods. In Table 1 and Table 2 are observed the differences between $c(0)$ from the first period and his correspondents in second and third periods. Those last ones are almost equal.

For the intermediary periods the values are the same and some differences are present at the beginning of the first period and at the end of the last. The firsts and lasts few coefficients have values close to the others, but not equal.

Between firsts and lasts coefficients appear a sort of symmetry. It can be said that we have a "side effect". This side effect can be due to the initial conditions.

Those coefficients introduce some errors at the beginning and the end of the sequence in the signal reconstruction.

In the case of the sine function we observe that the difference between $c(0)$ and $c(M)$ is greater than in the case of cosine function. For the odd functions the mirror-symmetric conditions introduce bigger errors.

For the signal reconstruction and interpolation, in all cases studied, were observed those side effects. At the beginning and the end of the data sequence the errors are higher then the others.

Considering the input signal $s(k)= \sin(2\pi k/120)$ we calculated the B-spline coefficients and completed the interpolation by factor 2. For the first and last few points the errors are $e=10^{-3}$ and decrease fast to $e=10^{-8}$

Table 2. Coefficients for $s(k)= \cos(2\pi k/M)$, $k_0=7$

α	M	First period		Second period		Third period	
		k	$c(k)$	k	$c(k)$	k	$c(k)$
0	12	0	1.0467677172	(M+0)	1.0467457811	(2M+0)	1.0467457811
	120	0	1.0004237020	(M+0)	1.0004570305	(2M+0)	1.0004570305
π/6	12	1	0.9065025600	(M+1)	0.9065084377	(2M+1)	0.9065084377
	120	10	0.8664212037	(M+10)	0.8664212038	(2M+10)	0.8664212038
π/3	12	2	0.5233744655	(M+2)	0.5233728905	(2M+2)	0.5233728905
	120	20	0.5002285152	(M+20)	0.5002285152	(2M+20)	0.5002285152

in the interpolated points. The algorithm has excellent properties of convergence. At the input data points the values obtained are exactly excepting the 2 extremely points. For calculating every interpolated value are necessary the coefficient for the current point $c(k)$ and some anterior and posterior coefficients. For the first and last 2 points of the string some of those coefficients are not known (example $c(-1)$) and considered zero. This is one of the causes for the side errors that appear in every cases when perform interpolation.

For $s(k) = \cos(2\pi k/120)$ the results are comparative. It was done the interpolation by 2 and the errors at the beginning and the end of the results array are smaller, starting from 10^{-6} and decay faster then for the above function. Similar results have been obtained in the cases of the studied signals with other sampling unit (M=50, 100). The errors for the B-spline coefficients have a real significance in the interpolation process.

For initializing $c^+(0)$ are used a finite number k_0 of samples. Extending the signal by mirroring [3],[5] it was calculated $c^+(0)$ in the relation (8). The results for coefficients, signal reconstruction and interpolation by 2 are better than in the case of using (9) for initialization, with $k_0=7$. For the cosine signal $\cos(2\pi k/120)$ the interpolation errors at the beginning and the end of the string are much smaller then for $k_0=7$. Compared with the same case of initialization for $\sin(2\pi k/120)$ the results are better too. But the errors decay slower compared with the cosine function case. It can be said that the mirror extension present disadvantages for the odd functions.

The coefficients for the extended signals are similar with the values obtained in the second and third period of the signals for $k_0=7$. It can be said that the values for $c(k)$ corresponding to the second period in Table 1 and Table 2 (where $k_0=7$) in fact represents the coefficients for $k_0= \infty$. If we can take the coefficients values from the intermediary period then the initialization is not necessary to be very precise. But in this case are necessary an increased amount of numerical operations.

If the input samples correspond to a straight line $s(k)= 1$, N=50, the B-spline coefficients $c(k)$ are calculated and presented in Table 3. At the beginning and at the end the coefficients have values different from 1. But they converge fast to 1. We perform the interpolation of the signal by the integral factor 2 and obtain the values $y(k/2)$ presented in the third column of Table 3.

Excepting the first and last values, in data points the signal is exactly recovered. Between those points, the interpolated values have errors for some positions. There is shown again that the algorithm has a good convergence due to polynomial spline properties. But for some oscillations in coefficients sequence are obtained oscillations in the interpolated signal. This could be an inconvenient if the input signal has a small number of samples.

For the errors that appear in the process of determination the B-spline coefficients a possible cause is that on principle the input signal is considered of infinite duration (extended to $k \in Z$) [3], [4]. In practice, for the recursive algorithm, the initialization is done in k = 0 and is limited to a k_0. If it is done a signal extension on both sides and performed the interpolation, in the center area the results are very good. In this way the middle area is the interval were the input samples are defined and the side errors present no more importance.

Table 3. Input signal $s(k)= 1$, $k_0=7$

k	$c(k)$	$y(k/2)$
0	0.999637023	0.8333107558
		0.9791538800
1	1.0000097259	1
		1.0000026699
2	0.9999973939	1
		0.9999992845
3	1.0000006982	1
		1.0000001916
4	0.9999998128	1
…	…	…
30	1	1
		1
31	1	1
…	…	…
49	1	1
		0.9791666666
50	1	0.8333333333

In the case of the straight line we took for input much more samples. For calculating $c^+(0)$ it is used the relation (7):

$$c^+(0) = \sum_{k=0}^{+\infty} s(k) z_1^k = \sum_{k=0}^{+\infty} z_1^k = \frac{1}{1-z_1} \qquad (12)$$

All the values for B-spline coefficients are equal to 1. The samples obtained for the interpolated signal by the integral factor 2 are presented in Table 4, along with the coefficients.

Table 4. Input signal $s(k)= 1$, $k_0=\infty$

k	$c(k)$	$y(k/2)$
0	1	0.8333333333
		0.9791666666
1	1	1
		1
2	1	1
		1
3	1	1
		1
4	1	1
…	…	…
30	1	1
		1
31	1	1
…	…	…
49	1	1
		0.9791666666
50	1	0.8333333333

We see that still exist a side effect due to the 2 coefficients that are not known at the beginning and the end of the data string.
In conclusion, to reduce the side effect it is necessary to perform a signal extension on both sides, but it is essentially to establish the right way to do it. The presented examples show that the extension of a signal by using its mirror image it is not an optimal solution in all the cases. The extension can be performed for the coefficients string too. We searched another approach for calculate the B-spline coefficients.

V. NEW METHOD FOR CALCULATING THE INITIAL B-SPLINE COEFFICIENTS

It was searched a way to initialize the spline coefficients without performing the signal extension on Z. For the new coefficients it is used the notation: $c_n(k)$. For the given set of data $\{s(0), s(1), s(2),\ldots, s(N-1)\}$ we consider $f(x)$ the interpolation function. We seek for that one to be a cubic spline function. An important property of those functions is that they are piecewise polynomial functions.
From the convolution of the coefficients with the cubic B-spline function in formula (1) it can be write the next relation:

$$6f(k) = 4\,c_n(k) + c_n(k-1) + c_n(k+1) \qquad (13)$$

This is happening in the function knots. Also in the sample points, the relations between the function derivatives and the B-spline coefficients are:

$$f'(k) = 0\,c_n(k) - \tfrac{1}{2}\,c_n(k-1) + \tfrac{1}{2}\,c_n(k+1) \qquad (14)$$
$$f''(k) = -2\,c_n(k) + c_n(k-1) + c_n(k+1) \qquad (15)$$

Those relations help us to evaluate the properties and the values for the B-spline coefficients. For k=2 it can be deduced:

$$c_n(2) = f(2) - f''(2)/6 \qquad (16)$$
$$c_n(0) = c_n(2) - 2f'(1) \qquad (17)$$
$$c_n(1) = \frac{6f(1) - c_n(0) - c_n(2)}{4} \qquad (18)$$

The problem is how to calculate $f'(1)$ and $f''(2)$ from the known samples. The interpolation function is a B-spline (piecewise polynomial), so we can approximate $f(k)$ by a polynomial function on short intervals. With this polynomial and his derivatives we calculate the values for the first 3 coefficients.
Consider f a polynomial function of 4-th order (pass trough 5 points):

$$f(x) = a + b\,x + d\,x^2 + e\,x^3 + g\,x^4 \qquad (19)$$

The function and the function derivatives of order 1 and 2 have been evaluated on the interval [0;4] and are obtained the next relations:

$$f'(1) = \frac{-3f(0) - 10f(1) + 18f(2) + f(4)}{12} \qquad (20)$$
$$f''(2) = \frac{-(f(0) + f(4)) + 16(f(1) + f(3)) - 30f(2)}{12} \qquad (21)$$

The main condition is that the interpolation function to pass through the input samples: $f(k)=s(k)$ for k=0,1…, N-1. By using (20) and (21) in relations (16), (17) and (18) are calculated the initial coefficients $c_n(0)$, $c_n(1)$ and $c_n(2)$ without performing any signal extension.
Those 3 coefficients have been calculated for the previous studied signals. Some results are presented in Table 5 for sine functions and Table 6 for cosine functions. The coefficients calculated with the new relations (in the last column) are compared with the corresponding coefficients in Unser's algorithm, for $k_0=7$ and $k_0=\infty$.
The coefficients for $s(k)=\sin(2\pi k/M)$ are presented in Table 5 for two situations: M=12 and M=120. As it can be seen the new values are much closer to the ideal values that the ones for $k_0=7$. The differences between $c_n(k)$ and $c(k)$ for $k_0=\infty$ are in order of 10^{-3} for M=12. For a higher sampling frequency the differences decrease to 10^{-7} (M-120).

Table 5. Coefficients for $s(k)=\sin(2\pi k/M)$

M	k	$c(k)$		$c_n(k)$
12	0	$k_0=7$	-0.3021395140	-0.0035163260
		$k_0=\infty$	-0.0000000413	
	1	$k_0=7$	0.6043309293	0.5244880516
		$k_0=\infty$	0.5233729016	
	2	$k_0=7$	0.8848157966	0.9055641193
		$k_0=\infty$	0.9065084347	
120	0	$k_0=7$	-0.0302443872	-0.0000000544
		$k_0=\infty$	0	
	1	$k_0=7$	0.0604638345	0.0523598917
		$k_0=\infty$	0.0523598753	
	2	$k_0=7$	0.1024047866	0.1045762250
		$k_0=\infty$	0.1045762359	

In case of $s(k)=\cos(2\pi k/M)$ the values are presented in Table 6. The results are similar: differences of order 10^{-7} for M=120.

Table 6. Coefficients for $s(k)=\cos(2\pi k/M)$

M	k	$c(k)$		$c_n(k)$
12	0	$k_0=7$	1.0467677172	1.0495366943
		$k_0=\infty$	1.0467457811	
	1	$k_0=7$	0.9065025600	0.9059470100
		$k_0=\infty$	0.9065084377	
	2	$k_0=7$	0.5233744655	0.5228276880
		$k_0=\infty$	0.5233728905	
120	0	$k_0=7$	1.0004237020	1.0004569306
		$k_0=\infty$	1.0004570305	
	1	$k_0=7$	0.9990948692	0.9990859898
		$k_0=\infty$	0.9990859389	
	2	$k_0=7$	0.9949740293	0.9949763183
		$k_0=\infty$	0.9949764222	

Having calculated the initial B-spline coefficients the next step is to establish the algorithm for calculating

the others. It has to perform the interpolation with those new coefficients and compare the results with the existing ones.

VI. CONCLUSIONS

In the studied articles this cubic spline interpolation was used for image processing [3], [5], [6]. All the results referred to techniques used in this area. We took the algorithm and applied it for some usually digital signals. The observations and conclusions regarding the coefficients were used for finding an improved method to perform cubic spline interpolation. The B-spline coefficients depend of the sampling frequency, of the input samples values, are close to those and follow the signal variation. The algorithm has excellent properties of convergence due to the spline function nature. It presents some side errors that have a great importance in the interpolation process. Those errors are due to the finite length of the input signal and to the extension by mirroring for some functions. It was searched a way to eliminate the oscillations in interpolated signal by reducing the oscillations in coefficients series.

By the process presented in section V is given an alternative to calculate the initial terms with minimum errors. There were elaborated and tested a few algorithms for eliminating ones of the deficiencies in the presented one and reducing the interpolation errors. The B-spline coefficients are calculated in a simple manner and the side effect can be negligible in the interpolated signal. Those algorithms must be finished and then published in to a further work.

ACKNOWLEDGEMENTS

The author would like to address special thanks to Professor Eugen Pop for his patience, guidance and support.

REFERENCES

[1] T. Blu, P. Thevenaz, M. Unser, "Linear Interpolation Revitalised", *IEEE Transactions on Image Processing*, Vol. 13, No. 5, pp.710-719, May 2004

[2] Gh. Micula, *Functii Spline si aplicatii*, Editura Tehnica Publishing House, Bucuresti, 1978

[3] M. Unser, "Splines: A Perfect Fit for Signal and Image Processing", *IEEE Signal Processing Magazine*, Vol. 16, No. 6, pp. 22-38, Nov. 1999.

[4] M. Unser, A. Aldroubi, M. Eden, "B-Spline Signal Processing: Part I - Theory", *IEEE Transactions on Signal Processing*, Vol. 41, No. 2, pp. 821-833, Feb. 1993.

[5] M. Unser, A. Aldroubi, M. Eden, "B-Spline Signal Processing: Part II – Efficient Design and Applications", *IEEE Transactions on Signal Processing*, Vol. 41, No. 2, pp. 834-848, Feb. 1993.

[6] B. Vrcelj, P.P. Vaidyanathan, "Efficient Implementation of All-Digital Interpolation", *IEEE Transactions on Image Processing*, Vol. 10, No. 11, pp. 1639-1646, Nov. 2001

Buletinul Ştiinţific al Universităţii "Politehnica" din Timişoara

Seria ELECTRONICĂ şi TELECOMUNICAŢII
TRANSACTIONS on ELECTRONICS and COMMUNICATIONS

Tom 51(65), Fascicola 2, 2006

Cryptographical System For Secure Client–Server Communication

Mircea-Radu Campean[1] and Monica Borda[2]

ABSTRACT

The aim of this paper was the research of a way to implement a cryptographical system for secure Client-Server communication, designed to satisfy the specific needs of the health care domain. A real IP based Client-Server application was created, that assures confidential message transfer using standardized cryptographic algorithms and components. A particular PGP (Pretty Good Privacy) like architecture was designed to ensure the communication security. Low costs, along with an easy to use implementation, represent decisive advantages when trying to implement the system in the medical area,which has limited budget for informatization.

Keywords: Cryptography, Encryption, Decryption, Authentication, Confidentiality, Client-Server

I. INTRODUCTION

Internet network development created the need for new types of services. The health care domain is one area, which is starting to benefit from the advantages Internet offers, providing different kind of services to help patients when they need medical care.
In our research we wanted to create a system, which allows patients to communicate with theirs doctors, using a PC connected to the Internet. For this purpose we developed a Client – Server application with a series of key features:

i. limited access to registered users
ii. ensures confidentiality and authentication
iii. intuitive interface for the Client
iv. limited hardware resources

Borland Delphi 7 was chosen as the programming language to create the Client – Server application after carefully considering its advantages over other solutions (i.e. object oriented programming language; high quality debugging support; easy to create user friendly interface; provides good error handling).

II. CLIENT-SERVER APPLICATION

Before developing an application we had to establish what were the requirements of the service we wanted to offer through our project. We intended to create a simple system, which could be used by patients to easily contact their doctor, using special software installed on a PC connected to Internet network. As shown in Figure1, we imagined that all communication, between doctor and patients would be filtered through the Server component, which runs on a computer situated at the Medical Care Centre.

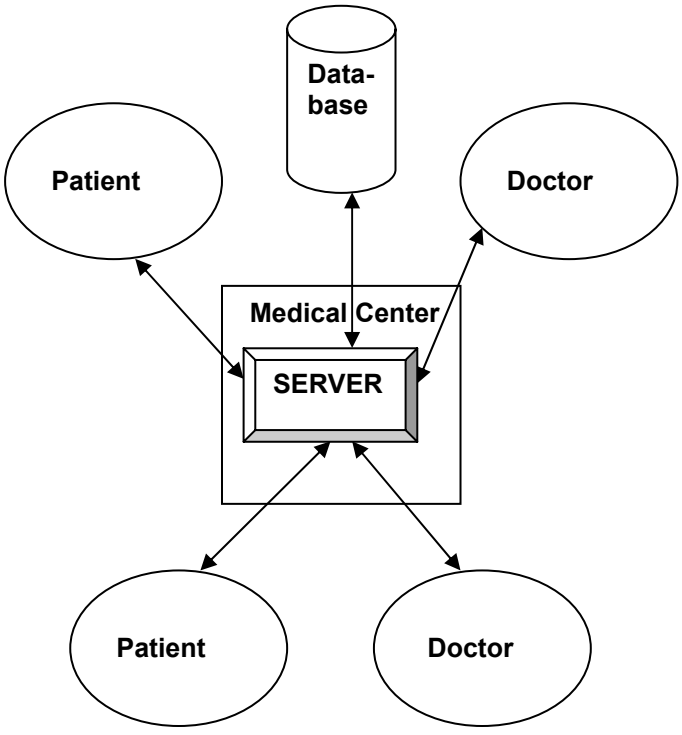


Fig 1: System structure for application

Due to the fact that both doctors and patients use the same software (the Client application) to gain access

[1] Facultatea de Electronică şi Telecomunicaţii, Departamentul Comunicaţii Str. Constantin Daicoviciu, No. 15, 400020, Cluj-Napoca, e-mail Radu.Campean@com.utcluj.ro
[2] Facultatea de Electronică şi Telecomunicaţii, Departamentul Comunicaţii Str. Constantin Daicoviciu, No. 15, 400020, Cluj-Napoca, e-mail Monica.Borda@com.utcluj.ro

into the system, the Server has to activate different features of the Client component according to the specific user rights, either doctor or patient. Also a simple database is used in order to store messages sent to/from users. The messages are stored in their original encrypted form, and this is done only if the recipient is not logged into the system when the messages are sent.
For the Client – Server part of the project, which solves the communication problems, we have used Indy (Internet Direct). Indy is an open source Internet component suite composed of Internet protocols written in Delphi and it is included in Delphi 6 and the later versions. There are more than 60 components especially designed to support network programming with Delphi, all grouped in four different categories:

- Indy Clients
- Indy Servers
- Indy I/O Handlers
- Indy Misc

II.1. THE SERVER APPLICATION

The Server is designed in such a way to implement a few basic functions like: connecting clients, authenticating users (verifying their *UserID* and *Password*), basic message processing, administrating the database, which contains user information and stored messages.
The Server part of the project is built around *TIdTCPServer* component, which implements a multi-threaded TCP (Transmission Control Protocol) server. This component uses one or more threads to listen for clients connections, and in conjunction with *TIdThreadMgr*, allocates a separate thread to handle each client connection to the server.
To successfully start a TCP server we need to specify the IP address and Port where the Server listens for clients. This is done using the Bindings property of the *TIdTCPServer* component. Also an OnExecute event handler had to be written, enabling the Server to reply to commands sent from the client.
As mentioned before the Server has to manage information found in a database. For this purpose we had used components found on the ADO (ActiveX Data Object) page. ADO is a set of COM components (DLLs) that allows you to access databases as well as well as e-mail and system files. Three data aware components were used in this project from the ADO page:

- *DBGrid:* - it is used to browse through the records retrieved from a table or by query
- *DataSource*: -used to provide a link between a dataset and DBGrid component on a form that enables display, navigation and editing of the data underlying the dataset
- *ADTtable:* -represents a table retrieved from an ADO data store

II.2. THE CLIENT APPLICATION

The Client component is based upon *TIdTCPClient*, which encapsulates a complete TCP (Transmission Control Protocol) client. As a first step when connecting to a TCP server, the Host and Port properties (the IP address and port where the Server awaits client connections) of the TIdTCPClient, have to be set.
In general a server cannot, by default, send a command or data to a client without having the client specifically asking for something. In our case we wanted the server to be able to initiate a scenario, in case it has a message to deliver. Because a Client (TIdTCPClient) does not implement a standard listen event, a TTimer component was added for handling the eventual command from the Server in an OnTimer event handler.
As mentioned before, doctors and patients will use the same Client application in order to connect to the Server, and gain access into the system. Each user will have a unique UserID and Password. A user-friendly interface was designed for the Client component, which has the following features:

- "*Login*" button - used when a client is trying to get access into the system
- "*Logout*" button – used for exiting the system
- "*Send*" button – used for sending messages to other users
- "*Clear*" button – cleans the message box
- "*ChangePassword*" – allows users to change their passwords
- "*AddPatient*" – allows doctors to add a new patient to their own list of *CONTACTS*

Note: Every user, patient or doctor, has one "Contact List" that contains other users, and with whom they can communicate.

III. SECURITY ISSUES

When designing a communication system one of the main concerns is the protection of the transmitted data, to ensure the confidentiality of system users. This is done using different cryptographic algorithms.
In order to ensure the communication security, a particular PGP like architecture was designed.(Note: we did not use the PGP system, we only used the principles of PGP security). This solution was chosen because PGP can be used to protect data in storage, in contrast to security protocols like SSL, which only protects data in transit over a network. PGP uses symmetric and asymmetric – key cryptography. The asymmetric cryptography part assumes that the recipient of a message has previously generated a key pair, a public key and a private key. The destination's public key is used by the sender to encrypt a secret key (session key) that is then used to encrypt the message (plaintext). The recipient of a PGP encrypted message decrypts the session key using its private key and then decrypts the message using the decrypted

key. Using two ciphers makes sense since there is a considerable difference in operating speed between public key and symmetric key cryptography, the latter being much faster. As an addition to basic PGP is the possibility to detect whether a message has been altered, and whether it was actually sent by the person who claims to be the sender. To solve this, the sender creates a digital signature of the message using RSA or a DSA signature algorithm, which is then compared with a computed message digest at the recipient.

The algorithms chosen in our system were: Triple DES using 128 bits keys; RSA with 512 bits keys; SHA-1 with a digital signature of 20 bytes encrypted with a 512 bits RSA key.

For this part of our project we used LockBox, a cross-platform library that can be used in Borland Delphi and C++ Builder applications under Windows. LockBox provides services that enable programmers to add cryptography to their own projects. There is also the possibility to digitally sign documents, using components that encapsulate the required functionality.

LockBox also contains a component hierarchy to offer an easy to use and but still powerful encryption possibility. Figure 2 shows this class hierarchy. AT its base is an encryption engine class, TlbCipher. This class has virtual methods to encrypt and decrypt an arbitrary buffer of data into another, a file into another, a stream or a string into another.

Two simple descendant classes act as roots for the two types of ciphers, symmetric and asymmetric. These classes provide extra functionality by dealing with keys, a single private key in the symmetric case and the pair of public and private keys in the asymmetric case. Finally there are the descendent components that perform the actual encryption and decryption using specific ciphers.

In our Client – Server application we have used three of the components presented in the component hierarchy (Figure2):

- *TLb3DES* – used for implementing the symmetric key algorithm
- *TLbRSASSA* –creating and verifying digital signatures
- *TLbRSA* –used for asymmetric key cryptography

In the encryption process, except for the Triple DES keys, no key variables are assigned, those being set up in the user authentication phase. The encryption is done on strings instead of files, the main reason being the implementation of the client server communication part. The encryption has three important steps. First the digital signature (encrypted hash) is generated using the plaintext and the sender's private key. As mentioned before SHA-1 with a digital signature of 20 bytes encrypted with a 512 bits RSA key has been chosen in this project. Then using Triple DES algorithm the plaintext and digital signature are encrypted using a session key. The last step involves the encryption of the session key using the recipient public key. As I previously said the application works with strings as plaintext and ciphertext. The decryption process acts as the reverse function of the encryption. Firstly, the recipient decrypts the session key using its own private key, and then using the session key decrypts the plaintext and digital signature.

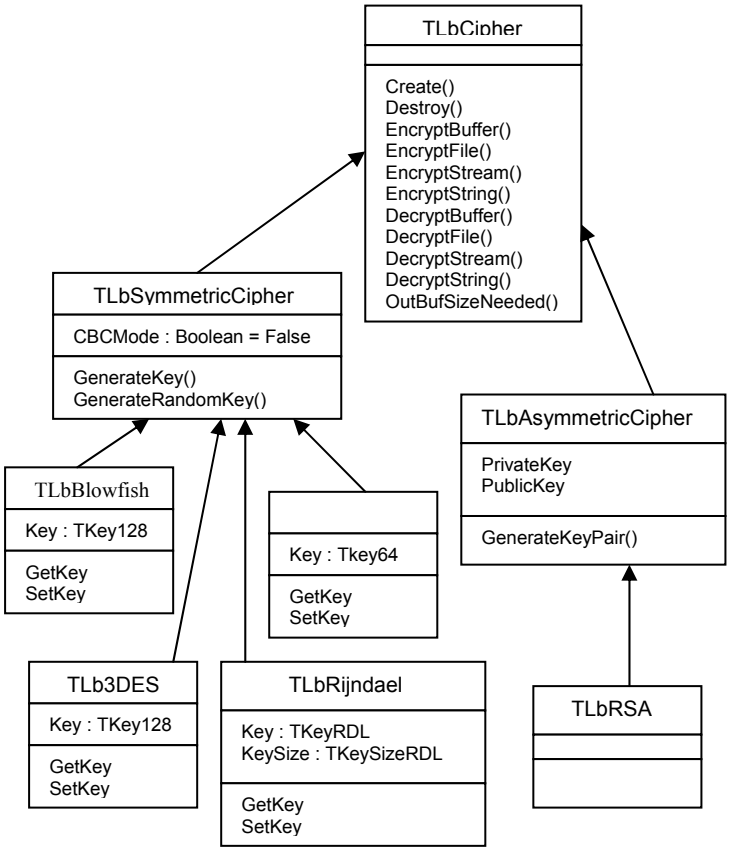


Fig*2*: Component hierarchy

IV. CONCLUSIONS AND FUTURE WORK

The system we have created has been tested in laboratory conditions on a limited number of computers thus on limited hardware configurations. We've had good results when we tested the application on computers with different versions of Windows OS (operating system) without having any kind of conflicts stating incompatibilities. The versions on which the application was tested were: Win 98, Win 98SE, Win 2000, Win XP (Home and Professional editions).

Another aspect we were interested in was the hardware resource needed on a PC (Personal Computer), which runs the Client or the Server application. The lowest hardware configuration that we tested had a 450 MHz processor and 64 MB of RAM. The Client application had no problem running on this system and both the processor and the memory usage were at low levels. The server on the other had would need better resources (at least 128 MB of RAM and a processor at around 1GHz) when there is a large number of clients connected in the system.

As said before the entire system was tested using a limited number of computers. The maximum Clients that we had connected to the Server at one moment were 20 and the system performed well. There should be no problem when a larger number of Clients connect to the Server since both implementations, Client and Server, are optimised.

In the future we want to develop the system so that it can be used in hospitals to create and maintain a database containing patients' medical charts and when needed, the patient's medical history could be sent to another hospital using the Client-Server application in a very short period of time. Access to the files containing the medical charts will be granted only to the patients' doctors ensuring in this way the patients confidentiality.

V. REFERENCES

[1]Bruce Schneier, Applied Cryptography Second Edition: Protocols, Algorothms, and Source Code in C, John Wiley & Sons, New York, 1996

[2] Alfred J. Menezes, Paul C. Van Oorschot, Scott A. Vanstone, Handbook of Applied Cryptography, CRC Press, Boca Raton, 1997

[3] William Stallings, Cryptography and Network Security – Principles and Practice. Second Edition, Prentice Hall, Upper Saddle River, New Jersey, 1999

[4]Titu Bajenescu, Monica Borda, Securitatea în informatică şi telecomunicaţii, Dacia, Cluj-Napoca, 2001

[5] Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest , Clifford Stein , Introduction to Algorithms, Second Edition; MIT Press and McGraw-Hill, 2001

[6] Marco Cantu, Mastering Delphi 7, Publisher: Sybex Inc..,2003 (ISBN: 0-7821-2874-2)

[7] TurboPower LockBox2 – Manual [pdf] , TurboPower Software Company, 2000

Buletinul Ştiinţific al Universităţii "Politehnica" din Timişoara

Seria ELECTRONICĂ şi TELECOMUNICAŢII
TRANSACTIONS on ELECTRONICS and COMMUNICATIONS

Tom 51(65), Fascicola 2, 2006

Designing an Audio Application for Bluetooth Enabled Devices

Andrei Maiorescu, Adriana Sîrbu, Ioan Cleju and Ion Bogdan[1]

Abstract – Bluetooth wireless technology has opened a new perspective over the services available in personal area networks. Implementation of audio based profiles is a challenge taking in account the diversity of involved devices and circumstances. The paper focuses on the implementation issues concerning the establishment of an audio link between two devices with built in Bluetooth radio chips. The particularities imposed by the protocol specifications are commented. Finally the test application is presented.
Keywords: Bluetooth, audio application

I. INTRODUCTION

Bluetooth technology has become a global wireless specification for a large variety of communications between portable and/or fixed devices. A broad base of vendors representing almost all segments of computer and communications market have chosen to support the development and promotion of this technology in the market place. It is estimated that until 2008, 922 millions of Bluetooth enabled products will be produced [1].

The Bluetooth radio transceivers operate in the globally available unlicensed ISM radio band of 2.4 GHz [2]. The use of a generally available frequency band ensures the fact that this wireless technology can be used virtually anywhere in the world to link up for ad hoc networking within a specified area.

In this context it is expected that Bluetooth technology will be embedded in a whole range of electronic devices in the next few years. This is the reason why it is envisaged that, soon enough, personal area networks will be able to provide surveillance and control of home appliances, materializing the concept of intelligent home.

In [3] the main aspects of the architecture, functions and performances of such a network have been commented. In Fig. 1 we present a possible structure of a home environment personal area network.

This paper focuses on the implementation issues concerning the establishment of an audio link between two devices with built in Bluetooth radio chips. Section II describes the particularities imposed on the audio links by the protocol specification. In section III we present the design of the audio test application. Finally, conclusions are drawn.

II. VOICE OVER BLUETOOTH

The Bluetooth specifications define two types of links in support of voice and data applications: an asynchronous connectionless (ACL) and synchronous connection-oriented (SCO) link. ACL link support data traffic; the information carried can be user data or control data. SCO links support real-time voice and multimedia traffic using reserved bandwidth. Both data and voice are carried in the form of packets and there can be ACL and SCO links at the same time, [4].

SCO links provide symmetrical, circuit-switched, point-to-point connections, which are typically used for voice. Three synchronous channels of 64 Kbps each are available.

But Bluetooth does not just define a radio system, it also defines a software stack to enable applications to find other Bluetooth devices in the area, discover what services they can offer and use those services. Fig 2 depicts the Bluetooth stack. The specification is broken up into several parts: the Core Specification and the profiles. The Core Specification includes the radio base-band and the software layers which make up the protocol stack. The Bluetooth stack is defined as a series of layers, though there are some features which cross several layers. The profiles give guidelines on how to use the protocol stack to implement different end-user applications.

The first version of the profiles document provides three different profiles covering audio applications: the Headset profile, the Cordless Telephony profile and the Intercom profile. Within the Bluetooth SIG, there are working groups that are producing profiles to support further audio applications[5].

[1] Facultatea de Electronică şi Telecomunicaţii,
Bd. Carol I Nr. 11, 700506, Iaşi, e-mail mandrei@etc.tuiasi.ro

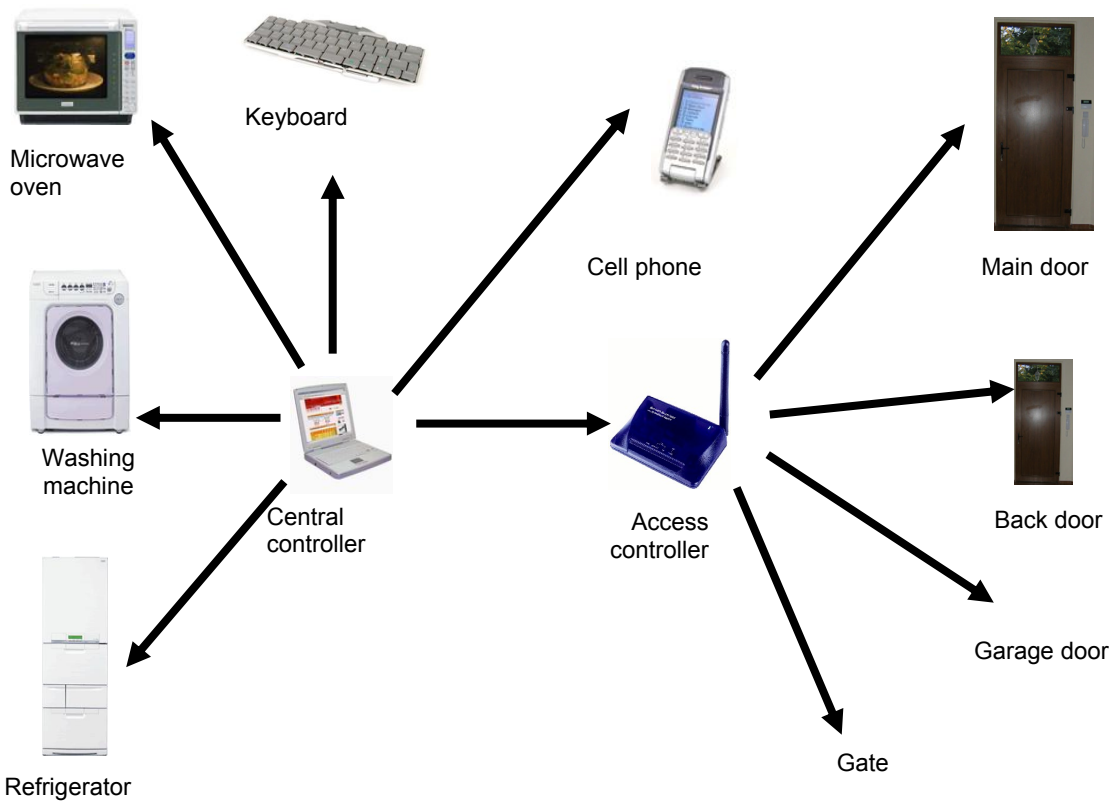


Fig. 1. Arhitecture of the personal area network for intelligent homes

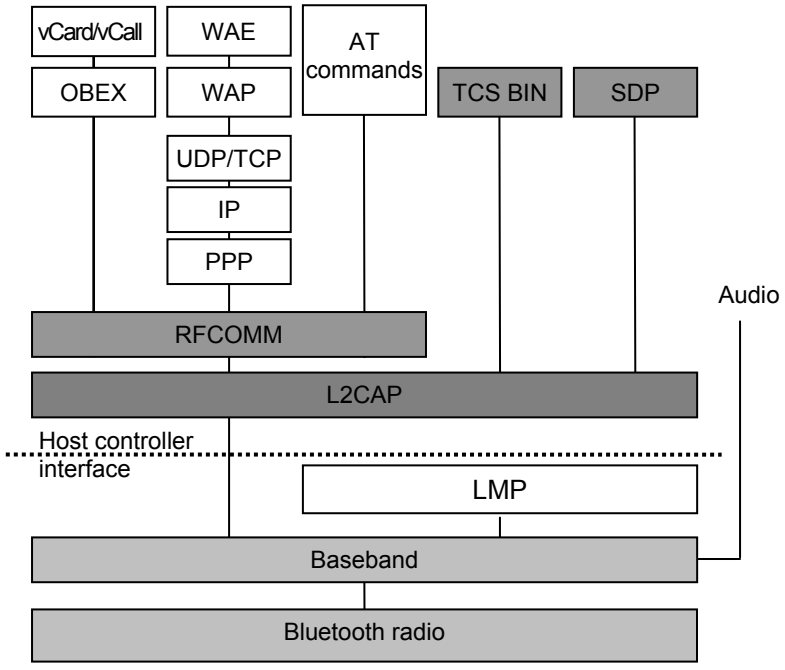


Fig. 2. The Bluetooth stack

LMP - Link Manager Protocol;
HCI – Host Controller Interface;
L2CAP – Logical Link Control and Adaptation Protocol;
SDP – Service Discovery Protocol;
RFCOMM – serial port emulation protocol;
TCS BIN – Telephony Control Protocol (bit-oriented protocol);
AT Commands – Telephony Control Protocol (command-oriented protocol over serial port);

IP, PPP, UDP, TCP – Network Protocols;
WAP, OBEX – interfaces for higher layer Communication Protocols;

The literature mentions that if an application provides a service which is covered by the existing Bluetooth profiles, then one should implement the relevant profile. However, at the moment, there are many possible audio applications which are not covered by profiles. In this case, one should design a complete proprietary application. As we intend to provide several audio services in a home environment personal area network, our endeavor fits in the second class of applications.

Beyond profiles though, the main core of all envisaged applications is the appropriate design of the audio link. That is why in the next section we present the details of implementation for a possible audio connection between two Bluetooth enabled devices.

III. DESIGN OF THE AUDIO TRANSFER

In order to test the quality of the audio link we have devised a test application called "Talking 2/1", that is transferring audio links between two audio cards connected on the same computer. Implementing such a test is useful taking into account that both Bluetooth devices and modem are recognized by the system as multimedia devices. This will allow easy integration of the test application in the final product.

The connection diagram for the test application is depicted in the Fig. 3.

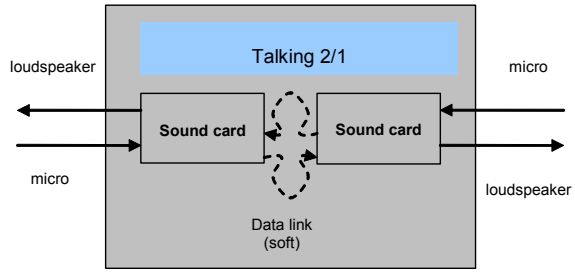


Fig. 3. Connection diagram for the test application "Talking 2/1"

One can see the interconnections between the two audio cards by means of the application. The link is bidirectional. Each card works independently. The record process is performed upon the input channel, data, coming from an A/D converter, are stored in a buffer.

The play process is running on the output channel where data, previously recorded in a buffer, are played using a D/A converter. One can not appropriately synchronize the record and play processes so that the same buffer to be used. This is the reason why, one has to use two buffers: one for recording and one for playing. Considering the same frequency for both processes, one can conclude that the recording and playing of one buffer last for the same time duration. So, one can use two buffers which interchange their roles.

In Fig. 4 we present the status of the buffers at a given moment of time. The input buffer is partially full and is filled, while the playing buffer is partially full and is emptied. Obviously, the two channels work independently and have their own buffers,

Testing this solution on a high-quality acquisition card, it proved to be functional. For the case of sound cards, due to the hardware implementation issues and taking into account the driver operation, the results were unsatisfactory. One supplementary reason of this behavior could be due to the delay introduced by the buffer interchange duration.

One observed that during usual functioning, a small but disturbing delay appears. This delay disappears if data are placed in a waiting queue. We have preferred a four buffer queue structure for each channel, so that at any instance those queues are not empty.

In Fig. 5 one can notice the placement of an empty buffer and a full buffer respectively in the waiting queue. This solution proved to be reliable during the tests we have performed.

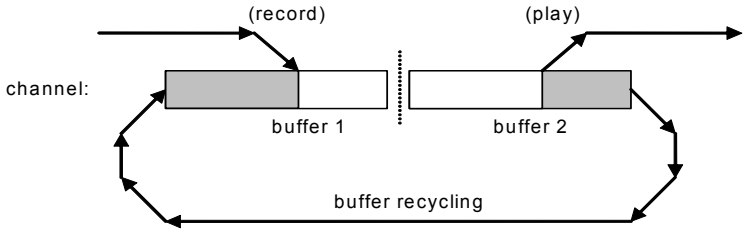


Fig. 4. Double - buffer functioning

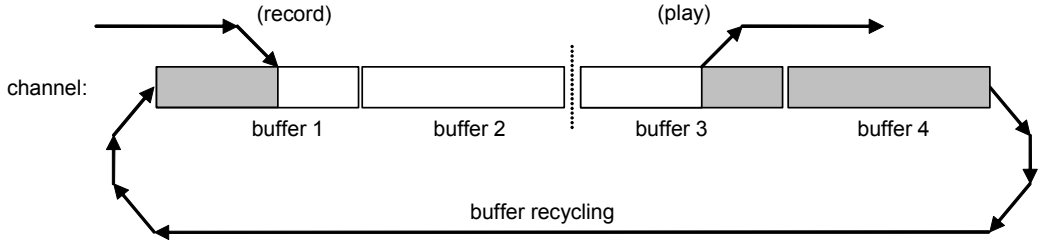


Fig. 5. Four - buffer functioning

Configuring the audio transfer at each location, that is the local sound card "**NVIDIA® nForce™ Audio**" and the virtual sound card "**Bluetooth Audio**" is presented in Fig. 6. The transfer is made using 4 data buffers as explained before. The audio channel quality as well as the volume of transferred data are selectable: 22 kHz sampling rate and 16 bits quantization. The buffer length is specified in kbytes and directly influences the total delay.

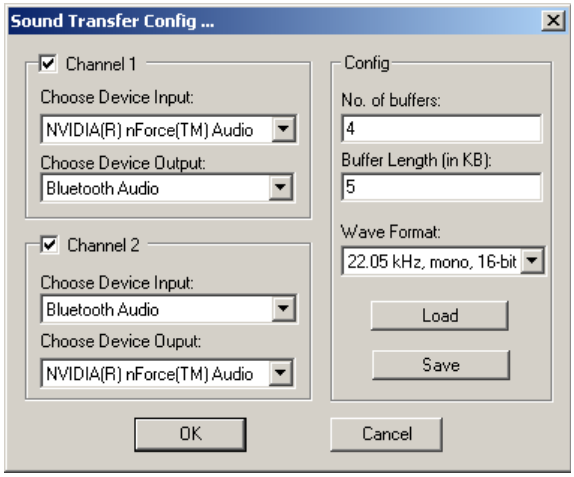


Fig. 6. Sound application interface

The final application, implemented on two separate computers, is called "Talking 2/2" is presented in Fig. 7. It is symmetric implemented and the transfer parameters have to be identical, that is number of buffers, sampling frequency and quantization. The total delay Δt introduced by the application can be calculated using the following relation:

$$\Delta t = 2 \cdot \frac{B}{2} \cdot \frac{DIM}{N} \cdot \frac{1}{f_{es}} + \Delta t_{BT}$$

where:

- B is the total number of buffers;
- DIM is the size, in Kbytes, of each buffer;
- N is the number of *bytes* per sample (equal to 1 for 8 bits/sample or 2 for 16 bits/sample);
- f_{es} is the sampling rate in *kHz*;
- Δt_{BT} is the delay on the Bluetooth audio channel.

The sampling frequency f_{es} and the type of quantization decisively influence the quality of the audio signal and the delay Δt_{BT}. A small size of the buffer determines the reduction of the channel delay, but a small size of the buffer is not recommended from the multi buffer functioning.

IV. CONCLUSIONS

In order to design a complete proprietary application to provide several audio services in a home environment personal area network, a test application for an audio link between two Bluetooth enabled devices was designed. The implementation details were commented. Based on this test a new profile for Bluetooth audio application can be developed.

REFERENCES

[1] http://www.bluetooth.com/products/.
[2] Jeniffer Bray, Charles Sturman – Bluetooth Connect Without Cables, Prentice Hall, 2001.
[3] A. Sîrbu, C. Comsa, I. Bogdan – Retele personale de monitorizare si control la domiciliu in tehnologie Bluetooth: arhitectura, functii, performante – Telecomunicatii, anul XXXII, Nr. 2, 2005 (in Romanian).
[4] N. Muller – Bluetooth Demistified, Mgraw-Hill TELECOM.
[5] D. Kammer,G. McNutt,B. Senese,J. Bray (Technical Editor) - Bluetooth Application Developer's Guide: The Short Range Interconnect Solution, Syngress.com.

ACKNOLEDGEMENTS

This work was partly supported by INFOSOC National Program under Grant INF 134/2004.

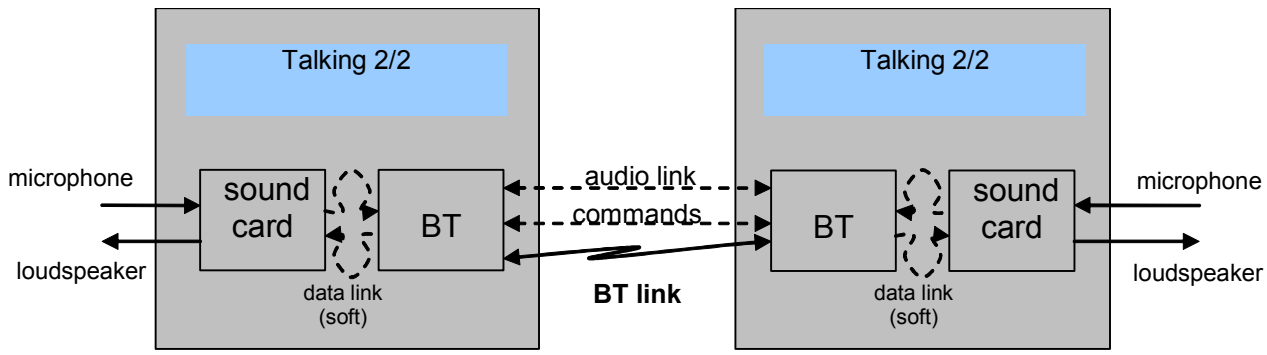


Fig. 7. Connection diagram for the application "Talking 2/2"

Buletinul Ştiinţific al Universităţii "Politehnica" din Timişoara

Seria ELECTRONICĂ şi TELECOMUNICAŢII
TRANSACTIONS on ELECTRONICS and COMMUNICATIONS

Tom 51(65), Fascicola 2, 2006

Digital Watermarking for Image Copyright Protection in the Wavelet Domain, robust against Geometric Attacks

Radu O. Preda[1], Dragoş N. Vizireanu[2], Radu M. Udrea[3]

Abstract – This paper proposes a digital watermark embedding method based on a multiresolution wavelet decomposition. The robustness against geometric distortions is based on image normalization. The watermark embedding and extraction are carried out with respect to an image normalized to meet a set of predefined moment criteria. We embed the watermark into the higher level detail wavelet coefficients from different wavelet subbands with the use of a key. The resulting watermarking scheme can be used for public watermarking applications, where the original image is not available for watermark extraction.

Keywords: Digital watermarking, Copyright Protection, Discrete Wavelet Transform, geometric attacks, image normalization.

I. INTRODUCTION

Digital watermarking is a process by which a user-specified signal (watermark) is hidden or embedded into another signal (cover data), for example digital content such as electronic documents, images, sounds and videos. There is urgent demand for techniques to protect the original digital data and to prevent unauthorized duplication or tampering.

The application that attracts the most attention is copyright protection. In this context, a watermark is permanently embedded in the work to identify its original owner. In order to be efficient, the embedded mark has to be robust, that is, it has to be detectable as long as the host carries its information, hence, the name of robust watermarking. Among these problems is the resilience of watermarking to geometric attacks. Such attacks are easy to implement, but can make many of the existing watermarking algorithms ineffective. Examples of geometric attacks include rotation, scaling, translation, shearing, random bending, and change of aspect ratio. Such attacks are effective in that they can destroy the synchronization in a watermarked bit stream, which is vital for most of the watermarking techniques. This is problematic, especially in applications where the original image is not available for watermark extraction.

In the literature, several approaches have been proposed to combat geometric attacks. Ruanaidh and Pun [1] proposed a scheme based on the invariant properties of Fourier–Mellin transform (FMT) to deal with attacks such as rotation, scaling, and translation (RST). This approach was effective in theory, but difficult to implement. Aimed to alleviate the implementation difficulty of this approach, Lin et al. [2] proposed to embed the watermark in a one-dimensional (1-D) signal obtained by projecting the Fourier–Mellin transformed image onto the log-radius axis. This approach was intended to embed only one bit of information, i.e., presence or absence of the watermark.

In [3], Pereira and Pun proposed another approach in which an additional template, known as a "pilot" signal in traditional communication systems, besides the watermark was embedded in the DFT domain of the image. This embedded template was used to estimate the affine geometric attacks in the image. The image first corrected with the estimated distortion, and the detection of the watermark was performed afterwards. A theoretical analysis was provided in [4] on the bit error rate for this pilot-based approach under a number of geometric attacks. This approach requires the detection of both the synchronization pattern and the watermark. A potential problem arises when a common template is used for different watermarked images, making it susceptible
to collusion-type detection of the template [5].

In [6], Bas et al. proposed a watermarking approach that is adaptive to the image content. In this approach salient feature points, extracted from the image, were used to define a number of triangular regions. A 1-bit watermark was then embedded inside each triangle using an additive spread spectrum scheme. This approach requires robust detection of the salient points in the image in order to retrieve the watermark.

In [8], a watermarking scheme was proposed using moment based image normalization, a well-known

[1] Electronics and Telecommunications Faculty, Department of Communications, Bd. Iuliu Maniu Nr. 1-3, 061071 Bucharest, e-mail radu@comm.pub.ro
[2] Electronics and Telecommunications Faculty, Department of Communications, Bd. Iuliu Maniu Nr. 1-3, 061071 Bucharest, e-mail nae@comm.pub.ro
[3] Electronics and Telecommunications Faculty, Department of Communications, Bd. Iuliu Maniu Nr. 1-3, 061071 Bucharest, e-mail mihnea@comm.pub.ro

technique in computer vision and pattern recognition applications [7]. In this approach, both watermark embedding and extraction were performed using a normalized image having a standard size and orientation. Thus, it is suitable for public watermarking where the original image is not available. The approach in [8] was used to embed a 1-bit watermark.

In this paper, we propose a watermarking technique to alleviate the problem of geometric distortions. The method is a multibit public watermarking scheme based on image normalization, aimed to be robust to general affine geometric attacks. Our scheme is different from the one in [8] because we address more general affine distortions, where shearing in the x and y directions are allowed rather than simple scaling and rotation attacks.

II. IMAGE NORMALIZATION

The key idea of this watermarking scheme is to use a normalized image for both watermark embedding and detection. The normalized image is obtained from a geometric transformation procedure that is invariant to any affine distortions of the image. This will ensure the integrity of the watermark in the normalized image even when the image undergoes affine geometric attacks. A functional diagram of this watermarking scheme is illustrated in Fig. 1. It is noted that the cover image is not needed for the watermark extraction. Thus, this scheme is suitable for public watermarking applications.

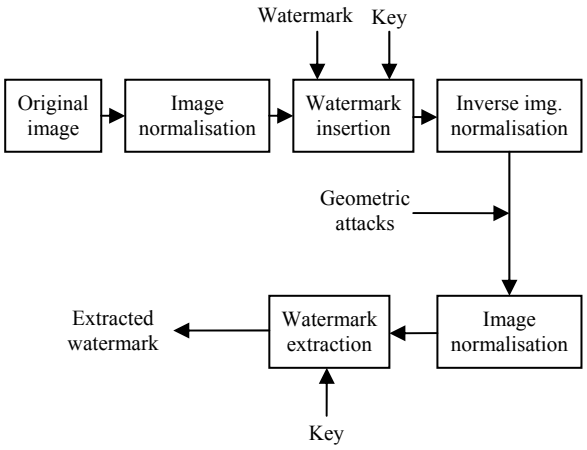


Fig. 1. Watermarking system based on image normalisation

We describe the components that define this scheme in detail. We begin with some background on image moments and geometric affine transforms, which are the necessary tools for image normalization.

a) Image Moments and Affine Transforms

Let $f(x,y)$ denote a digital image of size $M \times N$. Its *geometric moments* m_{pq} and *central moments* μ_{pq}, $p,q = 0,1,2,...$ are defined, respectively, as

$$m_{pq} = \sum_{x=0}^{M-1}\sum_{y=0}^{N-1} x^p y^q f(x,y) \tag{1}$$

and

$$\mu_{pq} = \sum_{x=0}^{M-1}\sum_{y=0}^{N-1} (x-\bar{x})^p (y-\bar{y})^q f(x,y) \tag{2}$$

where

$$\bar{x} = \frac{m_{10}}{m_{00}}, \bar{y} = \frac{m_{01}}{m_{00}} \tag{3}$$

An image $g(x,y)$ is said to be an *affine transform* of $f(x,y)$ if there is a matrix $A = \begin{pmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{pmatrix}$ and vector $d = \begin{pmatrix} d_1 \\ d_2 \end{pmatrix}$ such that $g(x,y) = f(x_a, y_b)$, where

$$\begin{pmatrix} x_a \\ y_b \end{pmatrix} = \mathbf{A} \cdot \begin{pmatrix} x \\ y \end{pmatrix} - \mathbf{d} \tag{4}$$

It is readily seen that RST are all special cases of affine transforms. Other examples of affine transforms include: *shearing in the x direction*, which corresponds to $A = \begin{pmatrix} 1 & \beta \\ 0 & 1 \end{pmatrix} = A_x$ in (4); *shearing in the y direction*, which corresponds to $A = \begin{pmatrix} 1 & 0 \\ \gamma & 1 \end{pmatrix} = A_y$; and *scaling in both x and y directions*, which corresponds to $A = \begin{pmatrix} \alpha & 0 \\ 0 & \delta \end{pmatrix} = A_S$.

Moreover, it is straightforward to show that any affine transform can be decomposed as a composition of the aforementioned three transforms, e.g., $A = A_S \cdot A_y \cdot A_x$, provided that $a_{11} \neq 0$ and $\det A \neq 0$.

b) Image Normalization

In this section, we describe a normalization procedure that achieves invariance under affine geometric distortions.

The general concept of image normalization using moments is well-known in pattern recognition problems (e.g., see [9], [10], where the idea is to extract image features that are invariant to affine transforms). In this application, we apply a normalization procedure to the image so that it meets a set of predefined moment criteria.

The normalization procedure consists of the following steps for a given image $f(x,y)$.

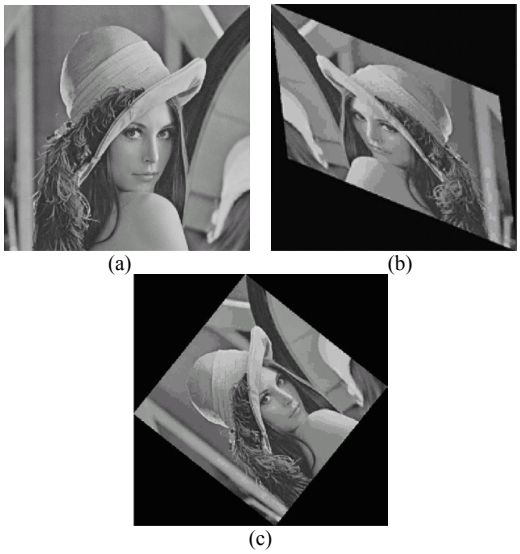


Fig. 2. (a) Original Lena image. (b) Lena image in (a) after distortion. (c) Normalized image from both (a) and (b).

1) Center the image $f(x,y)$; this is achieved by setting in (4) the matrix $A=\begin{pmatrix}1 & 0\\ 0 & 1\end{pmatrix}$ and the vector $d=\begin{pmatrix}d_1\\ d_2\end{pmatrix}$ with

$$d_1=\frac{m_{10}}{m_{00}},\ d_2=\frac{m_{01}}{m_{00}} \tag{5}$$

This step is aimed to achieve translation invariance. Let $f_1(x,y)$ denote the resulting centered image.

2) Apply a shearing transform to $f_1(x,y)$ in the x direction with matrix $A=\begin{pmatrix}1 & \beta\\ 0 & 1\end{pmatrix}$ so that the resulting image, denoted by $f_2(x,y)=A_x[f_1(x,y)]$, achieves $\mu_{30}^{(2)}=0$, where the superscript is used to denote $f_2(x,y)$.

3) Apply a shearing transform to $f_2(x,y)$ in the y direction with matrix $A=\begin{pmatrix}1 & 0\\ \gamma & 1\end{pmatrix}$ so that the resulting image, denoted by $f_3(x,y)=A_y[f_2(x,y)]$, achieves $\mu_{11}^{(3)}=0$, where the superscript is used to denote $f_3(x,y)$.

4) Scale $f_3(x,y)$ in both x and y directions with $A=\begin{pmatrix}\alpha & 0\\ 0 & \delta\end{pmatrix}$, so that the resulting image, denoted by $f_4(x,y)=A_S[f_3(x,y)]$, achieves a prescribed standard size and $\mu_{50}^{(4)}>0$ and $\mu_{50}^{(4)}>0$.

The final image $f_4(x,y)$ is the normalized image, based on which subsequent watermark embedding or extraction is performed. Intuitively, the above normalization procedure can also be explained as follows: The discussion following (4) points to the fact that a general affine transformation attack can be decomposed as a composition of translation, shearing in both x and y directions, and scaling in both x and y directions. The four steps in the normalization procedure are designed to eliminate each of these distortion components. More specifically, step 1) eliminates the translation of the affine attack by setting the center of the normalized image at the density center of the affine attacked image, steps 2) and 3) eliminate shearing in the x and y directions, and, finally, step 4) eliminates scaling distortion by forcing the normalized image to a standard size. It is important to note that each step in the normalization procedure is readily invertible. This will allow us to convert the normalized image back to its original size and orientation once the watermark is inserted.

Of course, we need to determine in the above procedure the parameters associated with the transforms A_x, A_y and A_S. We will address this issue in the next subsection. In the following theorem we present the invariant property of the normalized image to affine transforms.

Theorem 1: An image and its affine transforms have the same normalized image.

To demonstrate this normalization procedure, we show in Fig. 2(a) an original image "Lena". In Fig. 2(b), we show this image after an affine distortion; both of these images yield the same normalized image, shown in Fig. 2(c), when the above normalization procedure is applied.

c) Determination of the Transform parameters

- Shearing matrix $A_x=\begin{pmatrix}1 & \beta\\ 0 & 1\end{pmatrix}$

The paramater β can be obtained solving the equation:

$$\mu_{30}^{(1)}+3\beta\mu_{21}^{(1)}+3\beta^2\mu_{12}^{(1)}+\beta^3\mu_{03}^{(1)}=0 \tag{6}$$

Equation (6) can have up to three roots in the case that $\mu_{03}^{(1)}\neq 0$. If one of the roots is real and the other two are complex, we use the real root. If all the roots are real, we set β to the median of the three real roots. this choice of β ensures the uniqueness of the resulting normalized image.

- Shearing matrix $A_y=\begin{pmatrix}1 & 0\\ \gamma & 1\end{pmatrix}$

We have

$$\mu_{11}^{(3)}=\gamma\mu_{20}^{(2)}+\mu_{11}^{(2)} \tag{7}$$

Setting $\mu_{11}^{(3)}=0$, we obtain

$$\gamma=-\frac{\mu_{11}^{(2)}}{\mu_{20}^{(2)}} \tag{8}$$

- Scaling matrix $A_S = \begin{pmatrix} \alpha & 0 \\ 0 & \delta \end{pmatrix}$

Parameters α and δ are determined by resizing the image to a prescribed standard size in both horizontal and vertical directions. Their signs are determined so that both $\mu_{50}^{(4)}$ and $\mu_{05}^{(4)}$ are positive (which can be changed by flipping either horizontally or vertically).

III. THE WATERMARKING SCHEME

Our goal is to hide the Copyright information in the normalized original image using the Wavelet Transform domain for the watermark embedding. We use a unique (secret) binary identification key of 128 bits to allow the recovery of the mark. The main steps of our embedding technique are presented in the following.

a) First the owner's identification key of 128 bits is stored and kept secret by the owner, it is not transmitted. 128 bits are enough to grant uniqueness of the key and protect the owner.

b) The first 8 bits in the secret key are used to select the wavelet decomposition scheme (the Wavelet functions used and the number of decomposition levels). The multitude of basis functions available increases the security of our scheme. The Wavelet families used are Daubechies, Coiflets, Symlets and Biorthogonal and the maximum level of decomposition is L.

c) Using the specification extracted from the secret key the Wavelet decomposition of the original image is performed. The multidimensional decomposition is done using successive filter banks.

d) The next 16 bits of the secret author's key indicate the size of the binary image used as the mark. The other bits of the key are used to identify the location of coefficients, where the mark will be embedded. For every bit of the mark a Wavelet coefficient is identified. These coefficients are evenly distributed in the bands of decomposition levels between 2 and L-1, where L is the maximum decomposition level of the original image.

e) Each bit of the binary watermark image is inserted in the corresponding coefficient by rounding the value of the coefficient to an even or odd quantization level. Rounding to an even quantization level embeds a zero, while rounding the to an odd quantization level embeds a one.

f) Finally, we apply the Inverse Discrete wavelet Transform on the available coefficients – some modified and some not – to create the watermarked image. As shown in Fig. 3, the image produced is visually identical to the original unmarked image.

At the other end of the communication channel or after the image has been stored, the watermark has to be extracted. The first four steps of the decoding procedure are identical to the embedding ones. The secret key is used to decompose the normalized image in levels of detail according to specific parameters. Then the Wavelet coefficients are selected and the watermark is extracted.

a) b)

Fig. 3. a) Original "Lena" image and b) watermarked image

IV. EXPERIMENTAL RESULTS

This section describes the experimental results, which verify the capabilities of our watermarking scheme. For this purpose we used real square-size images of resolution 512x512 pixels. We notice that our system is more suitable for photographic-like gray-scale images since they have more detail in which to hide a watermark. The binary images used as watermarks containe a text message and are of size 80x20 pixels.

First we have made sure that our embedding system does not introduce visual artifacts in images. We have first measured the visual quality of the marked images by qualitative observations. However, to produce more objective results, we have also used the Peak Signal to Noise Ratio (PSNR). The results obtained for 20 different original images are shown in Table 1. We have obtained an average PSNR of 51.01 dB. This is high above the usually tolerated degradation level of 40 dB.

Table 1 PSNR results for different Wavelet families

Wavelet function	**Average PSNR [dB]**
Haar	50.21
Daubechies 12	50.36
Daubechies 16	52.92
Coiflets 1	50.18
Coiflets 5	50.40
Symlets 4	50.64
Symlets 12	50.76
Biorthogonal 2.2 (order for reconstruction.order for recomposition)	52.59
Overall	**51.01**

The best results are obtained for the Daubechies Wavelet family of higher order and for the Biorthogonal Wavelet family (compactly supported biorthogonal spline wavelets for which symmetry and exact reconstruction are possible with FIR filters).

The first goal of our project was to develop a watermarking scheme for Copyright protection, that can withstand a certain degree of image compression and resist a series of image processing an geometric attacks. Generally speaking, JPEG recommends a quality factor between 75 and 95 for a compressed image to be visually indistinguishable from the original one, and between 50 and 75 to be merely

acceptable. First we tested our watermarking scheme against JPEG compression and some image processing attacks. Tables 2 and 3 show the robustness of our Wavelet Packets-based digital watermarking system to some of this attacks.

Table 2 Resistence of the watermarking scheme to JPEG Compression

JPEG Quality Factor	JPEG Compression	PSNR after compression	Extracted Watermark
100	1.6	47.12	
90	3.1	42.05	
80	4.9	39.2	
70	6.4	38.13	
60	8	37.24	
40	10.2	36.3	
20	12.6	35.04	

Table 3. Resistence of the watermarking scheme to blurring, sharpening amd mixed attacks

Attack type	PSNR after attack (dB)	Extraxted Watermark
Blur	37.63	
Sharpen	34.07	
Blur+JPEG compression with Q=40	35.22	
Sharpen+JPEG compression with Q=40	30.86	

We we tested the robustness agianst geometric attacks. The following is a list of attacks used to distort the images in the experiments (note that not all of them are affine transforms).

1. Scaling by different factors: (a) 0.5, (b) 0.75, (c) 0.9, (d) 1.1, (e) 1.5, and (f) 2.
2. Aspect ratio change: (a) (0.8, 1.0), (b) (0.9, 1.0), (c) (1.1,1.0), (d) (1.2, 1.0), (e) (1.0, 0.8), (f) (1.0, 0.9), (g) (1.0, 1.1), and (h) (1.0, 1.2), where each pair of numbers indicate the amount of scaling in the x and y directions, respectively.
3. Rotation with different angles: (a) , (b) , (c) 5 , (d) 25 , (e) 35 , (f) 45 , and (g) 80.
4. Shearing: (a) (0, 1%), (b) (0, 5%), (c) (1%, 0), (d) (5%, 0), (e) (1%, 1%) and (f) (5%, 5%), where each pair of numbers indicate the amount of shearing in the x and y directions, respectively.
5. General geometric affine transformation with matrix: (a) $\begin{pmatrix} 1.1 & 0.2 \\ -0.1 & 0.9 \end{pmatrix}$, (b) $\begin{pmatrix} 0.9 & -0.2 \\ 0.1 & 1.2 \end{pmatrix}$, and (c) $\begin{pmatrix} -1.01 & -0.2 \\ -0.2 & 0.8 \end{pmatrix}$.
6. Flipping: (a) horizontal and (b) vertical.

The test results from the first experiment are summarized in Table 4. We see from these results that the proposed algorithm achieves very low decoding BER (Bit Error Rate) for all the geometric attacks.

Table 4 Robustness against geometric attacks (BER)

	(a)	(b)	(c)	(d)	(e)	(f)	(g)	(h)
Scaling	0	0.001	0.001	0.002	0.003	0.06		
Aspect Ratio	0	0	0	0	0	0	0	0
Rotation	0	0	0	0	0	0	0	
Shearing	0	0	0	0.001	0.001	0.001		
Affine Transform	0	0.001	0.002					
Flipping	0	0						

V. CONCLUSIONS

In paper, we proposed a public watermarking algorithm that is robust to general affine geometric transformation attacks. The proposed algorithm achieves its robustness by both embedding and detecting the watermark message in the normalized images. By numerical experiments we demonstrate that the proposed algorithm can achieve very high robustness when used with multibit watermarks under various image processing and affine attacks.

REFERENCES

[1] J. O'Ruanaidh and T. Pun, "Rotation, scale and translation invariant spread spectrum digital image watermarking," Signal Process., vol. 66, no. 3, pp. 303–317, 1998.

[2] C. Y. Lin, M.Wu, J. A. Bloom, I. J. Cox, M. Miller, and Y. M. Lui, "Rotation, scale, and translation resilient public watermarking for images," IEEE Trans. Image Process., vol. 10, no. 5, pp. 767–782, May 2001.

[3] S. Pereira and T. Pun, "Robust template matching for affine resistant image watermarks," IEEE Trans. Image Process., vol. 9, no. 6, pp. 1123–1129, Jun. 2000.

[4] M. Alvarez-Rodríguez and F. Pérez-González, "Analysis of pilot-based synchronization algorithms for watermarking of still images," Signal Process.: Image Commun., vol. 17, no. 8, pp. 661–633, Sep. 2002.

[5] I. J. Cox, M. L. Miller, and J. A. Bloom, Digital Watermarking. San Mateo, CA: Morgan Kaufmann, 2001.

[6] P. Bas, J.-M. Chassery, and B. Macq, "Geometrically invariant watermarking using feature points," IEEE Trans. Image Process., vol. 11, no. 9, pp. 1014–1028, Sep. 2002.

[7] J. Wood, "Invariant pattern recognition: A review," Pattern Recognit., vol. 29, no. 1, pp. 1–17, 1996.

[8] M. Alghoniemy and A. H. Tewfik, "Geometric distortion correction through image normalization," presented at the ICME Multimedia Expo, 2000.

[9] I. Rothe, H. Susse, and K. Voss, "The method of normalization to determine invariants," IEEE Trans. Pattern Anal. Mach. Intell., vol. 18, no. 4, pp. 366–376, Apr. 1996.

[10] D. Shen and H. S. Ip, "Generalized affine invariant image normalization," IEEE Trans. Pattern Anal. Mach. Intell., vol. 19, no. 5, pp. 431–440, May 1997.

[11] D. Stinson, Cryptography: Theory and Practice, 2nd ed. Boca Raton, FL: CRC, 2002.

[12] A. van Leest, M. van der Veen, and A. Bruekers, "Reversible watermarking for images," Proc. SPIE, vol. 5306, Jan. 2004.

[13] T. Z. Chen, G. Horng, and S. H. Wang, "A Robust Wavelet Based Watermarking Scheme using Quantization and Human Visual System Model," Proceedings of the Pakistan Journal of Information and Technology, vol. 2, pp. 212-230, 2003.

[14] X. Kang, J. Huang, Y. Q. Shi, and Y. Lin, "A DWT-DFT Composite Watermarking Scheme Robust to Both Affine Transform and JPEG Compression," in IEEE Transactions on Circuits and Systems for Video Technology, vol. 13, pp. 776-786, 2003.

Buletinul Ştiinţific al Universităţii "Politehnica" din Timişoara

Seria ELECTRONICĂ şi TELECOMUNICAŢII
TRANSACTIONS on ELECTRONICS and COMMUNICATIONS

Tom 51(65), Fascicola 2, 2006

ECG Signal Denoising in the Diversity Enhanced Wavelet Domain

Marius Oltean[1], Victor Adafinoaiei [2]

Abstract – The paper presents a denoising algorithm using which is particularly suited to ECG signals. The main stage of this algorithm consists in a MAP filtering in wavelet domain. Its effectiveness relies on the diversity enhancement of the signal to be processed and on realistic a-priori assumptions regarding statistical properties of the wavelet coefficients. Tests made on a big number ECG signals, in realistic conditions, showed very promising results. The noise is removed, while the useful waveforms are preserved.
Keywords: denoising, wavelet, ECG , MAP

I. INTRODUCTION

The clinical electrocardiogram (ECG) records the changing potentials of the electrical field generated by the heart. Electrocardiography can be used, within limits, to identify anatomical, metabolic, ionic and hemodynamic changes. Automatic ECG signal processing aims the detection and even the prevention of cardiac illness and can be very helpful for the cardiologists. Unfortunately, ECG signal acquisition process is subjected to various disturbing perturbations like power-line interferences, electromyogram noise caused by muscle activity, motion artifacts and baseline drift due to the respiration mechanism. All these unwanted phenomena make from the automatic interpretation of the signal a difficult and sometimes even an impossible task. In these conditions, a pre-treatment of the signal is highly desirable for removing such interferences. This procedure will be next referred to as denoising.

The term was introduced by Donoho [1] in relation with the wavelet transform (WT). WT has been extensively used in the signal processing community in order to highlight informative representations of non-stationary signals. WT is able to simultaneously provide time and frequency information and offers good temporal localization for high frequencies and high–frequency resolution for low frequencies. ECG record is a non-stationary signal, so WT-based denoising particularly matches to it. The architecture of a wavelet-based denoising system relies on WT ability to concentrate the useful signal energy into a small number of wavelet coefficients. The algorithm introduced by Donoho uses discrete wavelet transform (DWT) and it has three steps:

1. DWT is applied on the noisy signal;
2. Wavelet coefficients are filtered (procedure which is sometimes referred to as “shrinkage”, or “thresholding”). In general, some of these coefficients are put to 0, since they don’t contain useful information;
3. Remaining coefficients are back-converted in time domain to estimate the useful signal.

Generally, the results are highly dependent on the wavelet mother used (stages 1 and 3) and on the filtering procedure chosen (stage 2). Some modern wavelet denoising techniques implement a MAP filtering in the stage 2 of the algorithm, taking into account the statistical properties of the wavelet coefficients. Such a method, which is used for processing the ECG signal in noisy conditions [2], adapts the wavelet domain empirical Wiener filter presented in [3] to the particular case of ECG signals. The statistical properties of the wavelet coefficients are estimated through a pilot signal. The pilot is obtained by applying the classical Donoho’s algorithm on the input noisy signal. Next, a MAP filtering in wavelet domain is performed, using the properties estimated through the pilot. The wavelet basis used in the two stages (pilot estimation and MAP filtering) are different. Using a wavelet basis function with compact temporal support in the first stage allows for an accurate preservation of the areas around the QRS complex [2]. On the other hand, the use of wavelets with good frequency localization in the second stage of the algorithm refines the shapes of P and T waves. Note that Wiener filter could be regarded as a particular case of a MAP filter. The analytical solution required for implementing this kind of filter uses the hypothesis that both useful and noise samples (wavelet coefficients when the filter is applied in WT domain) have Gaussian probability density function (pdf). The two most important features of such a filtering technique are: realistic a-priori assumptions regarding statistical properties of both signal and noise components and a good

[1] Teach Assistant,[2] Student: Facultatea de Electronică şi Telecomunicaţii, Departamentul Comunicaţii Bd. V. Pârvan Nr. 2, 300223 Timişoara, e-mail marius.oltean@etc.upt.ro

estimation of the parameters that describe these properties.

An improved method to estimate the statistical parameters of the wavelet coefficients is proposed in this paper. This method relies on the diversity enhancement of the signal to be processed. On the other hand, realistic a-priori assumptions were made regarding the statistical properties of the wavelet coefficients. These assumptions are well adapted to the characteristic shape of ECG signal.

In ECG denoising, exact preservation of the useful waveforms is critical. However, distortions are sometimes introduced in the useful signal by DWT based denoising. In our algorithm, these distortions are mitigated by the use of a redundant WT, which provides translation invariance. On the other hand, the distortions can be controlled by a proper selection of the mother wavelet function and its corresponding scaling function. In the present study, after averaging ten results that we got by using different wavelet mothers, SNR improvement was obtained. In the same time, we show that there isn't a wavelet mother that offers the best results in all situations.

The theoretical background of our algorithm especially considers the suppression of wide band EMG noise, but good practical results are provided for the power-line interference too.

In section II, the proposed denoising algorithm is presented. Next, simulation results are shown. Section IV contains a few concluding ideas and draws future possible directions to continue our work on this subject.

II. METHOD

The architecture of the proposed denoising system is presented in fig. 1.

To the input we get the useful signal (s) additively perturbed by a Gaussian colored noise (p):

$$x = s + p \tag{1}$$

The denoising procedure is composed of two stages, presented below.

Stage1: Pilot signal and noise estimation

The goal of this stage is to provide a reliable estimation of both "clean" signal and noise statistical parameters.

In this purpose, the classical denoising method proposed by Donoho [1] is applied in the wavelet domain W1. Thus, the signal is converted in the wavelet domain, the resulting wavelet coefficients are shrinked ("Sh" block, fig. 1) and then back-converted in time domain.

As illustrated in [3], the use of a wavelet mother with compact temporal support is recommended in this stage. This choice mitigates pseudo-Gibbs phenomenon effects (ripples around discontinuities), usually associated with the shrinkage of the DWT coefficients. Thus, a good preservation of the zones around QRS is provided in this stage.

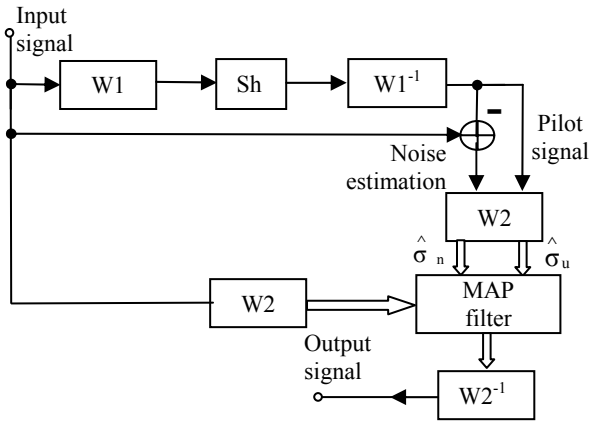


Fig.1 : Architecture of the denoising system.

The estimation of the pilot signal plays a double role. Besides the estimation of the "clean" signal, an estimation of the noise can be obtained as the difference between the noisy "observed" signal and the pilot signal. This operation takes into account the additive nature of the noise, illustrated by equation 1.

Thus, the first stage of the algorithm provides two time-domain signals: a pilot signal (estimating the useful signal) and a "purely" noise signal estimation respectively.

Stage 2: MAP filtering in the diversity-enhanced wavelet domain W2

In this stage, an empirical MAP filtering in the wavelet domain W2 is implemented. In order to provide robustness and superior performance to our algorithm, we made realistic a-priori assumptions regarding pdf of the useful and noise coefficients. In the same time, the statistical parameters estimation is improved by the diversity-enhancement of the signal to be processed. The diversity enhancement is obtained in the wavelet domain, by combining two redundant wavelet transforms, each of them providing several versions of the signal to be processed. The sources of diversity are the type of wavelet mother used in the computation of the discrete wavelet transform (DWT) [4] and the circular translation of the signal samples respectively [5]. In the first case we consider L_1 different wavelet mothers. In the second one, L_2 circular translations of the signal samples are used, but only one wavelet mother (chosen from the L_1 options). The two transforms are known as diversity-enhanced DWT (DEDWT) [4] and translation invariant DWT (TIDWT) [5] respectively. These transforms are combined in the following manner: L_1 versions of TIDWT are performed, each of them corresponding to a different wavelet mother. A new transform is obtained, called TIDWTED (Translation Invariant Wavelet Transform with Enhanced Diversity). Its redundancy is $L=L_1xL_2$. In our denoising system (see fig. 1), this transform is denoted by W2. Thus, to the output of the W2 block, we get L sequences of discrete wavelet coefficients, as follows:

$$^{l}w = {}^{l}u + {}^{l}n, \quad l = 1,..., L \tag{2}$$

${}^{l}u$ and ${}^{l}n$ denoting the useful and the noise coefficients respectively, for the l-th set of wavelet coefficients.

Using Bayesian rules, the MAP estimation of ${}^{l}u$ can be computed as:

$$ {}^{l}\hat{u}\,({}^{l}w) = \arg\max_{{}^{l}u}\left(\log(p_{w/u}({}^{l}w/{}^{l}u)\cdot p_u({}^{l}u))\right) = $$
$$ = \arg\max_{{}^{l}u}\left(\log(p_n({}^{l}w-{}^{l}u)) + \log(p_u({}^{l}u))\right) \quad (3) $$

In the following, without loss of generality, we can consider for the noise coefficients a Gaussian distribution, with zero mean and variance σ_n^2:

$$ p_n(n) = \frac{1}{\sqrt{2\pi}\sigma_n}\exp\left(-\frac{n^2}{2\sigma_n^2}\right) \quad (4) $$

For the useful signal coefficients pdf (p_u), a Laplacian distribution seems to be well suited to the characteristic shape of the ECG signal. This supposition is supported by empirical work on large ECG databases [6]. In fact, the wavelet transform of an ECG signal consists into a small number of high value wavelet coefficients (especially marking the limits of the electrical activity zones) and a large number of small value coefficients (for the slow-evolution portions of the ECG). A heavy-tailed distribution for these coefficients seems therefore far more realistic than a Gaussian-one, and the particular case of a Laplacian probability density function (pdf) becomes attractive by its computational tractability. Consequently, we take:

$$ p_u(u) = \frac{1}{\sqrt{2}\sigma_u}\exp\left(-\frac{\sqrt{2}|u|}{\sigma_u}\right) \quad (5) $$

Under the considered assumptions, the solution of (3) is [6,7]:

$$ {}^{l}\hat{u} = \begin{cases} {}^{l}w - {}^{l}T, & \text{if } |{}^{l}w| > {}^{l}T \\ 0, & \text{otherwise} \end{cases}, \quad l = 1,\ldots,L \quad (6) $$

This solution represents a softthresholding filtering of each of l sequences of noisy observations with the optimal threshold value ${}^{l}T$. This value is computed using the estimated standard deviations of the pilot and noise coefficients (see stage 1):

$$ {}^{l}T(j,k) = \frac{\sqrt{2}\,{}^{l}\hat{\sigma}_n^2(j)}{{}^{l}\hat{\sigma}_u(j,k)} \quad (7) $$

Note that the threshold value is individually estimated for each coefficient w(j,k), positioned on the j-th decomposition scale and having the index k within the scale. This is highly recommendable, since $\hat{\sigma}_u$ (estimated standard deviation of the useful coefficients) must be performed locally, in order to accurately track the ruptures that exist in the signal (e.g. the QRS complex). This parameter is separately estimated for each coefficient, using a sliding window:

$$ {}^{l}\hat{\sigma}_u(j,k) = \frac{\sqrt{\sum_i |\xi(j,i)|^2}}{\upsilon}, \quad i = k - \frac{\upsilon-1}{2},\ldots,k+\frac{\upsilon-1}{2} \quad (8) $$

where $\xi(j,i)$ represents the wavelet coefficient of the pilot signal, j standing for the decomposition scale and i for the position within the scale. υ is the length of the sliding window. Experimental work showed that the value υ=1 provides comparative results with higher window lengths (υ=3 or 5), so this value is chosen. On the other hand, the noise variance is separately estimated at each decomposition level j, using the wavelet coefficients of the purely noise signal at that level. This approach takes into account the fact that, generally, the noise that affects an ECG signal is not white, so its variance changes within scales (different frequency subbands).

Finally, useful signal is estimated by averaging all L versions of the estimated signal. This implies L_2 un-shifting operations, and then an averaging-over-shifts, performed by the Inverse Translation Invariant DWT (ITIDWT) [5]. Remember that we applied this transform for L_1 different wavelet mothers. The final result is obtained by averaging the L_2 variants of the denoised signal. As observed, the wavelet transform used (TIDWTED) is double redundant. The translation invariance is offered by averaging over the circular shifts. This mitigates the problem of peudo-Gibbs oscillations around fast transition portions of the signal (QRS area). The system performance is not sensitive to the wavelet mother chosen, since several basis functions are simultaneously used. Both transforms improve the SNR performance, by the averaging operation.

III. RESULTS

Several simulation sets were performed on real ECG signals, in order to demonstrate the performance of the proposed method.

3.1 General simulation parameters

ECG test signals were chosen from CHU Brest database. The sampling frequency of these signals is of 1000 Hz, with a resolution of 16 bits/sample. In order to obtain the pilot estimation (stage I), we shrinked the Haar coefficients of the noisy signal, with the threshold value $T(j) = s(j)\sqrt{2\log M}$ [8], where s(j) represents the standard deviation of the noisy wavelet coefficients at the decomposition level j and M is the length of the data block, namely M=4096

samples. For the second stage of the algorithm, we have chosen for DEDWT implementation L_1=10 different wavelet mothers with good frequency localization, from Daubechies, Coiflet and Symmlet families. In the case of TIDWT, we used the "fully" TIDWT [5], which averages over all circular shifts of the signal. That is, in this case, we get L_2=4096. Note that this transform can be calculated rapidly, in M logM time, despite appearances. This way, the redundancy factor is $L=L_1xL_2$=40960.

3.2 Simulation sets

In order to correctly evaluate the method's performance, several types of simulations were performed. Thus, SNR improvement was estimated. On the other hand, we evaluated the denoising effects on the next stage of the automatic processing chain, namely signal segmentation.

3.2.1 SNR improvement

SNR improvement represents a classical measure of denoising quality. In order to compute this measure, the clean signal must be a-priori known. In this context, we chose 5 "clean" ECG test signals of 60 seconds each. Artificially generated noise was added to this signal, resulting in SNR ratios between 10 and 20 dB. For the noise generation, a second-order AR-process was used, generating a colored Gaussian noise. This simulates the physical EMG noise, which is a wide-band colored signal, whose dominant energy spans in the 50 – 150 Hz range.

Output SNR is calculated for the entire ECG signal as well as for the fragments delimiting the P wave, which is the most sensitive to noise (this last measure is denoted by PwSNR). For each input SNR the experience was repeated 10 times and the results were averaged. The output SNR is computed for each of ten wavelet mothers used in TIDWT, as well as for the signal resulted by averaging this ten versions of the denoised signal (the signal to the TIDWTED output). A selection of the results is shown in table 1.

For P wave region, an averaged PwSNR improvement factor was computed at each "overall" SNR (fig. 2). This factor represents the difference between the output and the input PwSNR. Note that in this case neither an input or output averaged PwSNR can be calculated, since for the same overall input SNR there is an important variation of the input PwSNR between different signals. The reason is that the ratio: energy of the P wave / energy of the whole beat is strongly dependent on the physical characteristics of the patient, so it's different for each patient in particular.

	Type	INPUT SNR					
		10	**12**	**14**	**16**	**18**	**20**
Output SNR	Coiflet 1	22.50	24.18	25.81	27.44	29.04	30.54
	Coiflet 2	22.49	24.21	25.84	27.52	29.16	30.64
	Coiflet 3	22.38	24.05	25.71	27.37	29.01	30.53
	Daubechies 4	22.42	24.05	25.72	27.34	28.98	30.49
	Daubechies 6	22.56	24.11	25.80	27.44	29.03	30.53
	Daubechies 8	22.45	24.10	25.74	27.40	29.02	30.54
	Daubechies 10	22.32	23.94	25.63	27.28	28.86	30.38
	Daubechies 12	22.23	23.84	25.43	27.07	28.72	30.27
	Symmlet 4	22.52	24.22	25.88	27.56	29.18	30.69
	Symmlet 6	22.41	24.08	25.73	27.41	28.71	30.57
	TIDWTED	22.58	24.27	25.96	27.59	29.18	30.70

Table 1: SNR improvement results.

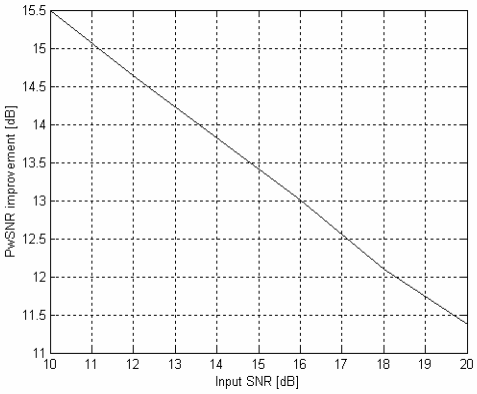


Fig. 2: PwSNR improvement factor versus overall input SNR.

The results shown in table 1 prove the effectiveness of the proposed method in terms of SNR improvement. This improvement is in all cases more than 10 dB. The performance is better compared to other ECG denoising results reported in literature [2,9,10], with spectacular differences for low SNRs. Yet, this comparison must be regarded with circumspection, since the work databases are different. The main gain in the SNR is brought by the use of a translation invariant wavelet transform (more than 1 dB better than DEDWT [6]). Note that in all cases, TIDWTED performs better than TIDWT with the best wavelet mother. Table 1 shows that there isn't a wavelet basis that can be classified as being "the best" for use in ECG denoising, since for different signals (and even for the same signal, but different SNRs) the best basis is different. In conclusion, even if the diversity enhancement obtained by the use of several different wavelet mothers does not significantly improve the results over TIDWT, it eliminates the performance's dependency on the choice of the wavelet mother.

The results in fig.2 illustrate excellent performance for the P wave denoising (more than 11dB PwSNR improvement in all cases). Note that for higher SNRs, PwSNR improvement is less spectacular than in low SNR conditions. This tendency (observable, but less important for the overall SNR) can be caused by the reduced energy of the P wave, comparing with the overall beat energy. Thus, the shrinkage of the wavelet coefficients, even if adapted to different portions of the signal, could affect the useful P coefficients.

3.2.2 Denoising influence on signal segmentation

This set of simulations evaluates the denoising influence on the signal segmentation. This approach takes into account the fact that signal denoising is only a pre-treatment step in ECG processing, being followed by segmentation, relevant parameters computation and patients classification. It is strongly desirable that segmentation results do not be dependent on the pre-treatment applied. When the signals are strongly affected by noise and their automatic segmentation is virtually impossible, a good denoising method should accurately estimate the useful ECG trace, allowing for a reasonable segmentation. The segmentation method used in this paper was presented and implemented by the authors in [11]. The method captures the dependencies that exist between the wavelet coefficients situated at different decomposition levels in the form of a probabilistic Markov tree with hidden states. The hidden state is represented by the coefficient's energy.

In this respect, we applied our method as a pre-treatment step for 30 relatively clean signals from CHU Brest database, that were next segmented using the procedure in [11] (only the first 20 beats were considered). The segmentation results for P wave were compared with the case where another denoising procedure [12] is applied (a SURE filtering [13], followed by a Wiener filtering with the protection of the QRS coefficients) (see table 2). The segmentation results, using the denoising proposed in [6] (the same algorithm, but comparing two different transforms-TIDWT with one wavelet mother and DEDWT) are also illustrated in the table.

	Onset error	End error	Segmentation Error Rate
Method in [12]	11.16 ms	11.37 ms	15.96 %
DEDWT	11.01 ms	8.87 ms	15.2 %
TIDWT	10.22 ms	7.99 ms	13.46 %
TIDWTED	10.64 ms	8.09 ms	13.54%

Table 2 : Denoising effects on the automatic segmentation of the P wave.

The results are quasi similar to those obtained by using TIDWT. The improvement is significant with respect to [12], showing a 2.5% reduction of the segmentation error rate. Note that a segmentation is considered erroneous if at least one of the three conditions are met: onset error>25ms, offset error>25 ms, more than 10 P wave with segmentation error for one single patient. The reference segmentation was provided by cardiologists from CHU Brest. Note that, unlike in the SNR improvement case, diversity enhancement in TIDWT does not improve the segmentation results. This could indicate that there are certain wavelet basis that are better for the segmentation than others.

In order to provide a deeper analysis of the denoising influence on the segmentation in various SNR conditions, another set of tests was performed. The test procedure has three steps: artificially generated noise is added on five signals with reduced segmentation error, the denoising procedure is applied and the segmentation is repeated, this time on the denoised signal. This way, a comparison of the segmentation results for the original and denoised signals can be done. In fig. 3, two extreme cases are shown. The worst case (test signal number 1) corresponds to a low-energy P wave, (input PwSNR= -6.69dB, for an overall input SNR of 10 dB). In this case, the denoising assures acceptable signal segmentation errors from input SNRs superior to 12 dB (input PwSNR<-4dB), which is a promising result. In the best case (prominent P wave), the denoising has little effect on the segmentation error, since the amount of noise is not sufficiently large to perturb the segmentation. In this situation, segmentation shift is reduced from the beginning.

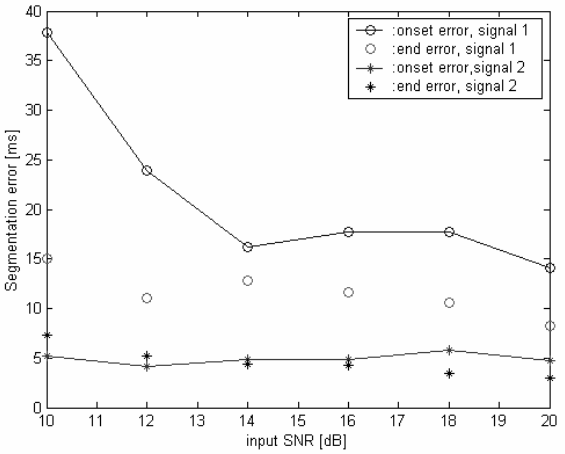


Fig. 3: Segmentation errors in various SNR conditions: two illustrative examples.

3.2.3 Denoising of signals affected by real noise

The algorithm was conceived for the denoising of ECG signals affected by real noise. This pre-treatment should allow correct signal segmentation. For testing our algorithm's effectiveness in real conditions, we applied it on a high number of ECG signals strongly perturbed by noise. The signals are raw data, provided by Task Force Monitor 3040i, from CNS Systems. An example is shown in figure 4.

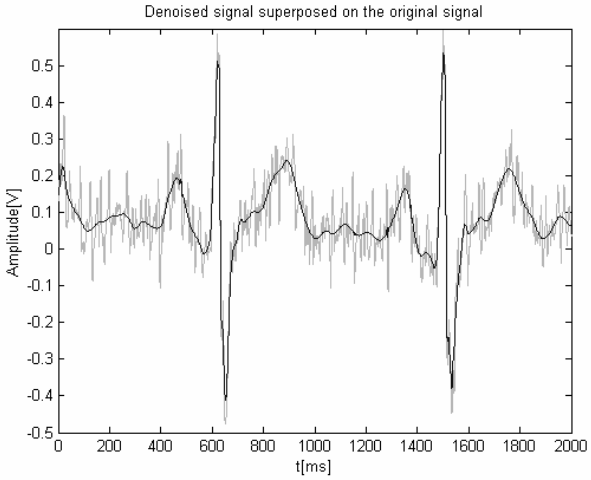


Fig. 4: Denoising applied on signal affected by real noise.

In the background, the original signal is strongly affected by noise. The denoised signal (in black) allows for a simple visual identification of the elementary waveforms (P, QRS,T). The noise is eliminated. A loss in amplitude of the QRS complex can be noticed, but this loss is maintained in a tolerable range (<10 % is acceptable, accordingly to cardiologists).

In conclusion, tests made on signals affected by real noise showed promising results. The noise that affects the signal in fig. 4 is a wide-band colored noise, which fits the theoretical background of the algorithm (see section II). On some test signals, the parasite component of 50 Hz can be clearly highlighted. Our algorithm has good practical results in these situations too.

IV. CONCLUSIONS

A new ECG denoising algorithm is presented in this paper. This algorithm relies on a modification of the empirical Wiener filtering in wavelet domain proposed in [3]. Superior performance is provided by a diversity enhancement of the signal to be processed. Realistic assumptions on the statistical properties of the useful wavelet coefficients are made. Several set of tests were performed, in order to demonstrate our algorithm's effectiveness. These tests highlight a good behavior of our method: an important SNR improvement (computed for signal affected by artificial noise), positive effect on signal segmentation and removal of noise (evaluated by a visual inspection of the denoised signal in real noise conditions).

Further improvements are still possible. In the future, we will focus on more elaborated methods for obtaining the pilot signal and in a deeper statistical study of the ECG wavelet coefficients.

ACKNOWLEDGEMENT

An important part of the research work that led to the redaction of this paper was supported and encouraged by Mister J. M. Boucher, professor at ENST Bretagne. Furthermore, ECG database, collected at CHU Brest, is obtained by his gentleness. That's why the authors would like to gratefully thank Mister Boucher for his help and support.

REFERENCES

[1] L. Donoho, **"**Denoising by Softthresholding", *IEEE Trans. Inform. Theory,* 41:613-627,1995.

[2] N. Nikolaev and A. Gotchev, "ECG signal denoising using wavelet domain Wiener filtering", *Proceedings of the European Signal Processing Conf. EUSIPCO-2000*, Tampere, Finland, pp. 51-54, September 2000.

[3] S. Ghael, A. Sayeed and R. Baraniuk, "Improved wavelet denoising via empirical Wiener filtering", *Proceedings of SPIE 3169*, San Diego, U.S.A., July 1997.

[4] A. and D. Isar, "Adaptive denoising of low SNR signals", *ARI, The Bulletin of the Istanbul Technical University*, Volume 53, Number 2, pp. 31-37, September 2003.

[5] R. Coifman and D. Donoho, "Transaltion-invariant de-noising", *Wavelets and Statistics*, A. Antoniadis and G. Oppenheim Eds, Springer-Verlag, pp. 125-150, New York, 1995.

[6] M. Oltean, J.M. Boucher and A. Isar, "MAP filtering in wavelet domain applied to ECG signal denoising", accepted to *ICASSP 2006, Toulouse*, France.

[7] L. Sendur and I. Selesnick, "Bivariate shrinkage functions for wavelet based denoising exploiting interscale dependency", *IEEE Trans. Signal Proc.*, vol. 50, pp. 2744-2756, Nov. 2002.

[8] I. Jonstone et B. Silverman, "Wavelet threshold estimators for data with correlated noise", *Journal of Royal Statistical Society*, Volume B 59, number 2, pp. 319-351, 1997.

[9] A. Gotchev, N. Nikolaev and K. Egiazarian, "Improving the transform domain ECG denoising performance by applying inter-beat and intra-beat decorrelating transforms", *Proceedings of the IEEE International Symposium on Circuits and Systems, ISCAS 2001*, Volume II, pp. 17-20, Sydney, Australia, May 2001.

[10] N. Nikolaev, A. Gotchev, "De-noising of ECG signals using wavelet shrinkage with time-frequency dependant threshold", *Proceedings of the European Signal Processing Conf. EUSIPCO-98*, Island of Rhodes, Greece, September 1998, pp. 2449-2453.

[11] S. Graja and J.M. Boucher, "Multiscale Hidden Markov Model Applied to ECG Segmentation", *Intelligent Signal Processing IEEE International Symposium* , pp 105-109, Sept. 2003.

[12] R. Le Page, *Detection and analysis of the P wave of an ECG signal: application to the atrial fibrillation detection,* Ph. D.Thesis, Université de Bretagne Occidentale, Brest, France, February 2003.

[13] D. Donoho and I. Johnstone, "Adapting to unknown smoothness via wavelet shrinkage", *Journal of American Statistical Assoc.*, vol. 90, pp. 1200-1224, December 1995.

Buletinul Ştiinţific al Universităţii "Politehnica" din Timişoara

Seria ELECTRONICĂ şi TELECOMUNICAŢII
TRANSACTIONS on ELECTRONICS and COMMUNICATIONS

Tom 51(65), Fascicola 2, 2006

Estimation of Noisy Sinusoids Instantaneous Frequency by Kalman Filtering

János Gal, Andrei Campeanu, Ioan Nafornita[1]

Abstract – The paper adresses the problem of estimating the instantaneous frequency of discrete time sinusoids imbedded in Gaussian noise. The proposed method is based on a model of the signal phase as a polynomial. This approach offers the opportunity to represent these signals by an adequate state space model and to apply standard Kalman filtering procedures in view to estimate the parameters of the phase polynomial. Procedure simulations were made on linear chirp sinusoids and are consistent with the theoretical approach. The paper presents the most important results.

Keywords: instantaneous frequency, polynomial phase, chirp signal, Kalman filter

I. INTRODUCTION

In order to estimate the instantaneous frequency of sinusoids corrupted by noise, we use an adequate model of the signal with emphasis on its instantaneous phase. This section introduces the model of a polynomial phase sinusoid and the use of state space description in Kalman estimation of its parameters.

Take a non-stationary continuous-time signal, $s(t)$, given by:

$$s(t) = A\cos\Phi(t) \tag{1}$$

where A is constant. For non-stationary signals, (signals whose spectral contents vary with time) the frequency at a particular instant of time is described by the concept of instantaneous frequency. The instantaneous frequency $f_i(t)$ of this signal is [1] ÷ [6]:

$$f_i(t) = \frac{1}{2\pi}\frac{d\Phi(t)}{dt} \tag{2}$$

We limit the discussion at the signals, whose phase $\Phi(t)$ is an M -order polynomial,

$$\Phi(t) = \sum_{k=0}^{M} a_k t^k \tag{3}$$

The instantaneous frequency for these signals becomes:

$$f_i(t) = \frac{1}{2\pi}\sum_{k=1}^{M} k a_k t^{k-1} \tag{4}$$

The discrete-time signals, $s[n]$, where n is the normalized time, are given by:

$$s[n] = A\cos\Phi[n] \tag{5}$$

where $\Phi[n]$ is a polynomial defined as:

$$\Phi[n] = \sum_{k=0}^{M} a_k n^k \tag{6}$$

At the normalized moment of time n, the instantaneous frequency is given by:

$$f_i[n] = \frac{1}{2\pi}\sum_{k=1}^{M} k a_k n^{k-1} \tag{7}$$

It can be seen that if the polynomial's coefficients $a_1 \div a_M$, which describe the phase, are known it is possible to achieve the measurement of the instantaneous frequency for this class of signals.

There are two classes of methods to establish the polynomial's coefficients or the instantaneous frequency:

i) nonparametric methods which resort to time-frequency representation [1], [5] and

ii) parametric methods [2], [3], [6], [9], based on credible model for the signal in which the parameter's values are determining it.

In this paper we will find a states space model for the signal so as to be able to resort to Kalman filtering for the parameters determination.

As a rule of thumb, if the state vector $\mathbf{X}[n]$ describes the state of the system, which is determined by measured values summed in $\mathbf{Y}[n]$, we have [7]:

$$\mathbf{X}[n+1] = \mathbf{A}\mathbf{X}[n] + \mathbf{N}[n] \tag{8}$$

$$\mathbf{Y}[n] = \mathbf{B}\mathbf{X}[n] + \mathbf{W}[n] \tag{9}$$

In these relations $\mathbf{N}[n]$ is playing the role of the excitation, but it can represent also only a noise. $\mathbf{W}[n]$ is the vector of noise in the measured signal. $\mathbf{A}$ is the transition matrix and $\mathbf{B}$ is the measurement matrix.

[1] Facultatea de Electronică şi Telecomunicaţii, Departamentul Comunicaţii Bd. V. Pârvan Nr. 2, 300223 Timişoara, e-mail janos.gal@etc.utt.ro, andrei.campeanu@etc.utt.ro and ioan.nafornita@etc.utt.ro

Based on eq. (8) and (9), we will find a model for polynomial phase signals imbedded in additive white Gaussian noise with zero-mean value.

II. THE MODEL OF THE NOISY SINUSOID

If $s[n]$ is a signal given by (5), the associated analytical signal is $A\exp\{j\Phi[n]\}$. In our measurement scheme, the signal values are corrupted by additive white Gaussian noise $w[n]$, zero-mean and variance σ^2. Let's consider the associated analytical noise $w[n]$ given by:

$$w[n] = w_R[n] + jw_I[n] \tag{10}$$

with $w_R[n]$ and $w_I[n]$, the real part and the imaginary part of the analytical noise. If both parts are not correlated between them, having the same variance, we can write:

$$E\{w_R[n]w_R[n+k]\} = \frac{\sigma^2}{2}\delta[k] \tag{11}$$

$$E\{w_I[n]w_I[n+k]\} = \frac{\sigma^2}{2}\delta[k] \tag{12}$$

$$E\{w_R[n]w_I[n+k]\} = 0, \quad \forall k \in Z \tag{13}$$

where $E\{\cdot\}$ is the expectation operator. An analytical signal having these properties is called "cyclic" noise [8]. The measured signal $y[n]$ is obtained by the addition:

$$y[n] = A\exp\{j\Phi[n]\} + w[n] \tag{14}$$

We consider the last vectorial addition in terms of polar components of complex signal in (14):

$$\mathbf{Y}[n] = \begin{bmatrix} |y[n]| \\ Arg\{y[n]\} \end{bmatrix} \tag{15}$$

As is shown in [7], if the signal-to-noise ratio (SNR) in the measured signal $y[n]$ exceeds 13dB, the noise real part affects only the amplitude A, whereas the phase $\Phi[n]$ is affected by the imaginary part of the "cyclic" noise. Eq. (14) can be written now in terms of amplitude and phase as:

$$\begin{bmatrix} |y[n]| \\ Arg\{y[n]\} \end{bmatrix} = \begin{bmatrix} A \\ \Phi[n] \end{bmatrix} + \begin{bmatrix} v_R[n] \\ v_I[n]/A \end{bmatrix} \tag{16}$$

III. STATE VECTOR AND TRANSITION EQUATIONS

The values of an M-order polynomial $P(x)$, can be expressed by the Taylor series expansion [7]:

$$P(x_0 + \Delta x) = \sum_{k=0}^{M} \frac{(\Delta x)^k}{k!} P^{(k)}(x_0); \forall x_0, \forall \Delta x \in R \tag{17}$$

viewing that all derivatives having the order higher than M are zero. For the l-order derivative of the polynomial $P^{(l)}(x)$ can be used the following series expansion:

$$P^{(l)}(x_0 + \Delta x) = \sum_{k=l}^{M} \frac{(\Delta x)^k}{(k-l)!} P^{(k)}(x_0); \quad \forall x_0, \forall \Delta x \in R, l = \overline{1, M} \tag{18}$$

Replacing $P(x)$ by $\Phi[n]$, x_0 with n and Δx by 1, we have:

$$\Phi[n+1] = \sum_{k=0}^{M} \frac{1}{k!} \Phi^{(k)}[n] \tag{19}$$

$$\Phi^{(l)}[n+1] = \sum_{k=l}^{M} \frac{1}{(k-l)!} \Phi^{(k)}[n] \quad l = \overline{1, M} \tag{20}$$

The state vector $\mathbf{X}[n]$ is given by the amplitude of the sinusoid, the phase and the first M derivatives of the phase:

$$\mathbf{X}[n] = \left[A \quad \Phi[n] \quad \Phi^{(')}[n] \quad \dots \quad \Phi^{(M)}[n] \right]^T \tag{21}$$

having an $(M+2)\times 1$ size. The state at the moment $n+1$ can be written as:

$$\mathbf{X}[n+1] = \left[A \quad \Phi[n+1] \quad \Phi^{(')}[n+1] \quad \Phi^{('')}[n+1] \quad \dots \quad \Phi^{(M)}[n+1] \right]^T \tag{22}$$

For two consecutive moments, the relation between the two states is derived from (19) and (20):

$$\begin{bmatrix} A \\ \Phi[n+1] \\ \Phi^{(')}[n+1] \\ \Phi^{('')}[n+1] \\ \vdots \\ \Phi^{(M)}[n+1] \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 & \dots 0 \\ 0 & 1 & \frac{1}{1!} & \frac{1}{2!} & \dots \frac{1}{M!} \\ 0 & 0 & 1 & \frac{1}{1!} & \dots \frac{1}{(M-1)!} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & \dots 1 \end{bmatrix} \cdot \begin{bmatrix} A \\ \Phi[n] \\ \Phi^{(')}[n] \\ \Phi^{('')}[n] \\ \vdots \\ \Phi^{(M)}[n] \end{bmatrix} \tag{23}$$

We obtain a transition equation. Comparing (23) with (8), we find that the $(M+2)\times(M+2)$-size transition matrix is:

$$\mathbf{A} = \begin{bmatrix} 1 & 0 & 0 & 0 & \dots & 0 \\ 0 & 1 & \frac{1}{1!} & \frac{1}{2!} & \dots & \frac{1}{M!} \\ 0 & 0 & 1 & \frac{1}{1!} & \dots & \frac{1}{(M-1)!} \\ 0 & 0 & 0 & 1 & \dots & \frac{1}{(M-2)!} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & \dots & 1 \end{bmatrix} \tag{24}$$

and $\mathbf{N}[n]$ is the null vector:

$$\mathbf{N}[n]=\mathbf{0} \tag{25}$$

IV. THE MEASUREMENT EQUATION

The measurement vector has two components having, therefore a (2×1) size. Because the state vector has a $(M+2)\times1$ size, the measurement matrix $\mathbf{B}$ is $2\times(M+2)$ size. As shows [10], the measured output variables $\mathbf{Y}[n]$ can be obtained from (16) by:

$$\mathbf{Y}[n]=\begin{bmatrix}|y[n]|\\ Arg\{y[n]\}\end{bmatrix}= \\ =\begin{bmatrix}1&0&0&0&\dots&0\\0&1&0&0&\dots&0\end{bmatrix}\begin{bmatrix}A\\ \Phi[n]\\ \Phi^{(')}[n]\\ \Phi^{(")}[n]\\ \vdots\\ \Phi^{(M)}[n]\end{bmatrix}+\begin{bmatrix}v_R[n]\\ \frac{v_I[n]}{A}\end{bmatrix} \tag{26}$$

This is the measurement equation. Comparing (26) and (9) we find that the measurement matrix $2\times(M+2)$ size is:

$$\mathbf{B}=\begin{bmatrix}1&0&0&0&\dots&0\\0&1&0&0&\dots&0\end{bmatrix} \tag{27}$$

The vector $\mathbf{W}[n]$, must be considered as:

$$\mathbf{W}[n]=\begin{bmatrix}v_R[n]\\ \frac{v_I[n]}{A}\end{bmatrix} \tag{28}$$

V. CONNECTIONS BETWEEN STATE MATRIX AND PHASE POLYNOMIAL COEFFICIENTS

We determined a model of the signal (5) in the states space by (23) and (26). We can determine the state vector $\mathbf{X}[n]$ using the Kalman filtering. We want to obtain the way of getting the coefficients $a_1 \div a_M$ or even $a_0 \div a_M$.

To know the states means to know the following: A, $\Phi[n]$ and $\Phi^{(l)}[n]$, $l=\overline{1,M}$. If we know $\Phi[n]$, $\Phi^{(l)}[n]$ and replacing the value for stationary regime, for a given n in (19) and (20), we get an equation like (6):

$$\sum_{k=l}^{M}\sum_{m=k}^{M}\frac{(m-k)!}{(k-l)!}a_m n^{m-k}=\Phi^{(l)}[n],\ n \text{ given},\ l=\overline{0,M} \tag{29}$$

Solving the M linear equations system (29) we determine the $a_1 \div a_M$ coefficients. For a given n, the solution of (29) is succesively calculated, starting with the last coefficient a_M:

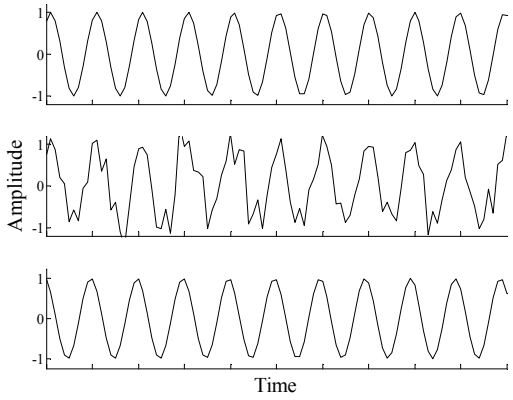


Fig. 1 The use of the polynomial phase model in view to apply a Kalman filter procedure to a noisy sinusoid.

$$a_M=\frac{1}{M!}\Phi[n]$$

$$a_{M-l}=\frac{1}{(M-l)!}\left[\Phi^{(M-k)}[n]-\sum_{k=0}^{l-1}\frac{(M-k)!}{k!}n^k a_{M-k}\right],\ l=\overline{1,M} \tag{30}$$

The instantaneous frequency, is computed directly from $\Phi^{(')}[n]$, as:

$$f_i[n]=\frac{1}{2\pi}\Phi^{(')}[n] \tag{31}$$

VI. EXPERIMENTAL RESULTS

In order to implement the states space model of a noisy constant amplitude sinusoid introduced before we used linear chirp sinusoids corrupted by noise. Hilbert transformation followed by modulus and phase calculation is applied to noisy sinusoid to obtain the Cartesian coordinates decomposition of eq. (15). These data represent the measured input vector for a Kalman filtering algorithm based on one-step prediction, which is implemented in MATLAB.

The reference signal, as was mentioned before is a constant amplitude linear frequency modulated sinusoid. The phase of this signal is described by a second-order polynomial, $M=2$ in eq. (3). The sampling frequency is 5000Hz. During one second, the instantaneous frequency of chirp sinusoid changes linearly between 100Hz and 900Hz. A part from this signal is represented in the first graph of Fig. 1. The second graph in Fig 1 shows the measured signal obtained by the addition of a zero-mean and variable variance white Gaussian noise to the reference signal. The signal-to-noise ratio (SNR) for this example is 10dB. Finally, the last image in Fig. 1 shows the result of Kalman filtering algorithm application to second image signal.

To give a better understanding of Kalman filter action on polynomial phase signals, Fig. 2 shows the results of instantaneous frequency $f_i(t)$ measurements performed on signals represented in Fig.1. In this case $\mathrm{SNR}=23\,\mathrm{dB}$. If, for the reference and measurement

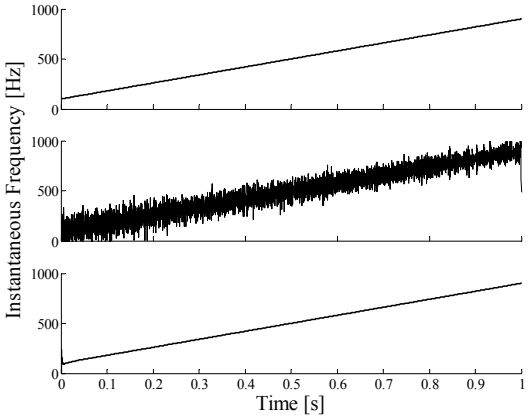


Fig. 2 The measurement of the instantaneous frequency of a polynomial phase noisy sinusoid by Kalman filtering.

signals, $f_i(t)$ established by (2) are represented in the first two images, the Kalman filter output shown in the last image, gives the estimation of $f_i(t)$ computed in (31), where $\mathbf{\Phi}^{(\cdot)}[n]$ is given by the third element of the state vector $\mathbf{X}[n]$. Fig. 2 is persuasive on the efficiency of Kalman filtering of polynomial phase noisy sinusoids in order to establish the instantaneous frequency.

In order to give an objective measure of the quality of the instantaneous frequency estimation by our method we define the *RMS Frequency Error*, $\Delta f_i^{RMS}[n]$ as

$$\Delta f_i^{RMS}[n] = \sqrt{E\left\{\left(f_i^{ref}[n] - f_i^{est}[n]\right)^2\right\}} \qquad (32)$$

where $f_i^{ref}[n]$ is the instantaneous frequency of the reference, and $f_i^{est}[n]$, the intantaneous value estimated by Kalman filtering. $E\{\cdot\}$ denotes the statistical expectation operator.

Fig. 3 and Fig. 4 use $\Delta f_i^{RMS}[n]$ to evaluate the performances of Kalman filter in instantaneous frequency estimation. The first experiment in Fig. 3 is made with SNR=13dB and shows that the maximum performances are obtained when the state vector

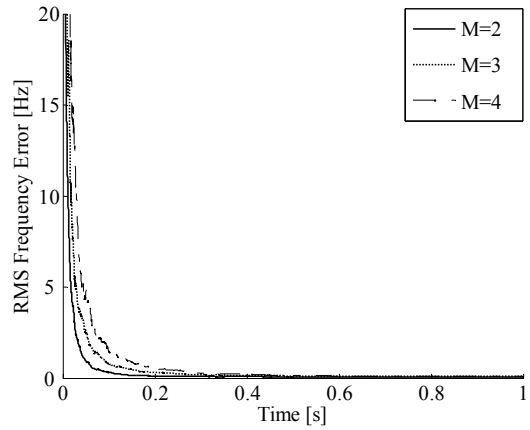


Fig. 3 The dependency of RMS Frequency Error on the order M+2 of the state vector **X**[n] used in Kalman filter.

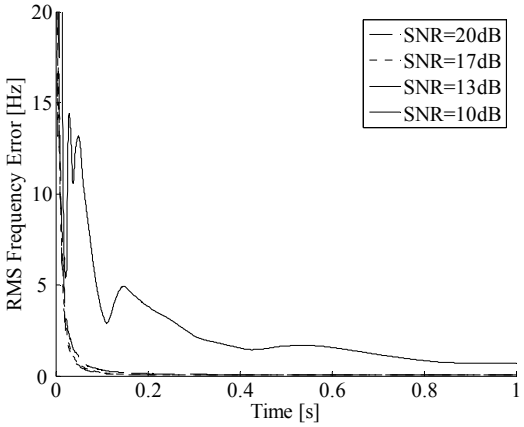


Fig. 4 The dependency of RMS Frequency Error on the SNR of the input noisy sinusoid.

$\mathbf{X}[n]$ is of minimum order: M+2=4. As about the Fig. 4, it was drawn for M=2 case and confirms that the method performs well as long as SNR exceeds 13dB.

VII. CONCLUSIONS

The paper gives the state space model of polynomial phase signals with good opportunities in instantaneous frequency estimation. The Kalman filter implemented on this model presents very encouraging results in the case of noisy linear frequency variation sinusoids.

REFERENCES

[1] Gal J., Sălăgean M., Bianu M., Naforniță I., "The Instantaneous Frequency Determination for Signals with Polynomial Phase using Kalman Filtering", *Buletinul Stiintific al Universitatii "Politehnica" din Timisoara, Proceedings of the Symposium on Electronics and Telecommunications*, Fifth edition, September 19-20, 2002, vol. I, pp. 186-189

[2] Gordan C., "Studiul reprezentărilor timp-frecvență şi aplicarea lor la estimarea frecvenței instantanee" – *Teză de doctorat*, 1999.

[3] Boashash B., "Estimating and Interpreting the Instantaneous Frequency of a Signal – Part 1: Fundamentals", *Proceedings of the IEEE*, Vol. 80, No. 4, April 1992, pp 520-538.

[4] Boashash B., "Estimating and Interpreting the Instantaneous Frequency of a Signal – Part 2: Algorithms and Applications", *Proceedings of the IEEE*, Vol. 80, No. 4, April 1992, pp 540-568.

[5] Boashash B., O'Shea P., Arnold M. I., "Algorithms for Instantaneous Frequency Estimation: a Comparative Study", *Proceedings of SPIE*, Vol. 1348, 10-12 July 1990, San Diego California, pp. 126-148.

[6] Bianu M., Gordan C., Naforniță I.. "A States Space Model of Nonstationary Signals with Phase Variation as a Polynomial", *4'th International Conference on Renewable Sources and Environmental Electro-Technologies, Oradea*, 6 – 9 iunie 2002

[7] Tretter S.A., "Estimating the Frequency of a Noisy Sinusoid by Linear Regression", *IEEE Trans. On Information Theory*, vol. II-31, No. 6, Nov. 1985, pp. 832-835

[8] Moon, T.K., Stirling W.C., "*Mathematical Methods and Algorithms for Signal Processing*", Prentice-Hall, 2000.

[9] Boashash B., Powers E., Zoubir A.M., (Edited by) "*Higher Order Statistical Signal Processing*", pp.27-111, Longman&Woley, 1995.

[10] Vincent I., Doncarli C., Le Carpentier E., "Classification des signaux non-stationnaires: comparaison d'une approche parametrique et d'une approche non-parametrique", *Quinzieme colloque Gretsi-Juan-les-Paris*, 18-21 Septembre 1995, pp. 161-164.

Buletinul Ştiinţific al Universităţii "Politehnica" din Timişoara

Seria ELECTRONICĂ şi TELECOMUNICAŢII
TRANSACTIONS on ELECTRONICS and COMMUNICATIONS

Tom 51(65), Fascicola 2, 2006

Evaluation of Information Capacity for a Class of MIMO Channels

Rodica Stoian, Lucian Andrei Perişoară[1]

Abstract - The Multiple Input Multiple Output (MIMO) Channels are usually used in wireless communications, by the use of spatial diversity at both sides of the link. The MIMO concept is more general and embraces many other scenarios such as wireline networks (LANs).
This paper summarizes the state of art in MIMO channels, presenting MIMO channel models, summarizing the computing of the information capacity for some particular MIMO channels and making a comparative analysis for different channel modeling parameters.
Keywords: Information Theory, Wireless Networks, LANs, MIMO systems, multipath propagation, information capacity.

I. INTRODUCTION

Based on generalization of Shannon's fundamental problem of communication, "that of reproducing at one point either exactly or approximately a message selected at another point", [1], the network information theory provides the theoretic basis to build up the best architecture for information transport.

Determining the appropriate architecture for information transfer between the nodes of a wireless network and the computation of amount of information that can be transported, i.e. the information capacity of the wireless network are fundamental problems for which the solutions is given by the network information theory.

In connection with this approach, in the field of communication systems, Multiple Input Multiple Output (MIMO) channels have recently become a popular means to increase the spectral efficiency and quality of wireless communications by the use of spatial diversity at both sides of the link [2, 3]. In fact, the MIMO concept is much more general and embraces many other scenarios such as wireline networks (LAN's) and single antenna frequency-selective channels [4, 5].

This paper summarizes the state of the art in MIMO channels. We begin by reviewing some well-known definitions and results on MIMO channel models, then we present a summary of the information capacity computing in MIMO systems. The core of this paper is dedicated to determine the information capacity for some particular MIMO channels, which can be the theoretical basis for a MIMO system implementation.

II. THE MIMO CHANNEL MODEL

The model of a MIMO channel is used in applications like the transmission between two nodes, where are many propagation paths, i.e. wireless and wireline (LAN) networks. We will consider a communication system where N sample signals x_i are transmitted from N input nodes simultaneously. Each transmitted signal x_i goes through the channel to arrive at each of the M output nodes.

In a MIMO communication channel with N input nodes and M output nodes, each output of the channel is a linear superposition of the faded versions of the inputs. Each pair of transmit and receive nodes provides a signal path from the transmitter i to the receiver j, described by the path gain h_{ij}. Based on this model, the signal y_j, which is received at node j, is given by:

$$y_j = \sum_i h_{ij} x_i + n_j , \qquad (1)$$

where n_j is the AWGN noise sample, with variance σ_n^2, of the receive antenna j.

We represent the signals that are transmitted from N transmit nodes as $\mathbf{x} = (x_1, x_2, ..., x_N)$, the signals received from M receive nodes as $\mathbf{y} = (y_1, y_2, ..., y_M)$, the path gains in a channel matrix $\mathbf{H} = \left[h_{ij}\right]_{N \times M}$, resulting the following matrix form of (1):

$$\mathbf{y} = \mathbf{H}\mathbf{x} + \mathbf{n} , \qquad (2)$$

where $\mathbf{n} = (n_1, n_2, ..., n_M)$ is the AWGN noise matrix.

[1] Faculty of Electronics,Telecommunication and Information Technology, Applied Electronics and Information Enginnering Departament, University Politehnica of Bucharest, ROMANIA, rodicastoian2004@yahoo.com, lperisoara@yahoo.com, www.orfeu.pub.ro

For wireless systems, different path gains may be independent from each other, that is h_{ij} is independent from $h_{i'j'}$ for $i \neq i'$ or $j \neq j'$. The path gain h_{ij} represents the attenuation of a signal from antenna *i* to antenna *j* and can be written as:

$$h_{ij} = \frac{ce^{-\gamma d_{ij}}}{d_{ij}^{\delta}}, \quad (3)$$

where $d_H(i, j) = d_{ij}$ is the Euclidian distance between two specific nodes i, j, δ is the path loss exponent and γ is the medium absorption constant [7]. Each node is also assumed to have a power constraint *P*.

Until now, we used the term of distance in two ways: the distances between nodes and the attenuation as a function of distance. The third way, distance will explicitly enter our model is through the choice of the performance measure "the information capacity".

For a network with *N* input nodes and *M* output nodes, having *NM* possible source-destination pairs, we use the rate vector denoted by the matrix $\mathbf{R} = [R_{ij}]$, with $i = \overline{1, N}$, $j = \overline{1, M}$, [6, Cap. 14]. Let $\Re$ be the set of feasible rate vectors. We define the information capacity as the supremely distance weighted sum of rates, described by the relation:

$$C = \max_{\mathbf{R} \in \Re} \sum_i \sum_j R_{ij} d_{ij}, \text{ [bps/Hz]}. \quad (4)$$

III. THE INFORMATION CAPACITY OF MIMO CHANNELS

For the information capacity of a particular MIMO channel, we assume that the receiver knows the realization of the channel, i.e. it knows both **y** and **H** or it has a perfect channel state information. For the transmitter, we study two cases:

A) the transmitter does not know the realization of the channel; however, it knows the distribution of **H**.

B) the transmitter knows the realization of the channel.

The resulting capacity of the channel is a random variable because the capacity is a function of the channel matrix **H** and has the distribution of the channel matrix **H**. For the input signal **x**, we assume that it is a zero-mean circularly symmetric complex Gaussian vector with covariance matrix $\mathbf{C}_X$, [2, 3].

A. *Capacity of a deterministic MIMO Channel*

We use a quasi-static block fading model. Under such a model, the channel path gains are fixed during a large enough block such that information theoretical results are valid. The values of path gains change from block to block based on Rayleigh fading channel model. First, we assume that the realization of the channel **H** is fixed, i.e. the channel matrix is deterministic and its value is known at the receiver.

The capacity is defined as the maximum of the mutual information between the input and output given a power constraint *P* on the total transmission power of the input, that is $\mathrm{Tr}(\mathbf{C}_X) \leq P$ [2].

The mutual information *I*, between input and output for the given realization is:

$$I = \log_2 \det\left(\mathbf{I}_M + \frac{1}{\sigma_n^2}\mathbf{H}\mathbf{C}_X\mathbf{H}^H\right), \text{ [bps/Hz]}, \quad (5)$$

Then, the information capacity is the maximum of the mutual information over all inputs satisfying $\mathrm{Tr}(\mathbf{C}_X) \leq P$. In other words,

$$C = \max_{\mathrm{Tr}(\mathbf{C}_X) \leq P} \log_2 \det\left(\mathbf{I}_M + \frac{1}{\sigma_n^2}\mathbf{H}\mathbf{C}_X\mathbf{H}^H\right), \text{ [bps/Hz]}, \quad (6)$$

The unit bps/Hz represents the fact that for a bandwidth of *W*, the maximum possible rate for a reliable communication is *CW* bps.

The capacity from (6) can be computed in terms of the positive eigenvalues λ_i of $\mathbf{H}\mathbf{H}^H$, using a water-filling algorithm [6] for the power allocation of the nodes, as:

$$C = \max_{\sum_i \gamma_i = \gamma} \sum_i \log_2(1 + \gamma_i \lambda_i), \text{ [bps/Hz]}, \quad (7)$$

where $i = \overline{1, rank(\mathbf{H})}$.

B. *Capacity of a random MIMO Channel*

Since the channel matrix **H** is random in nature, the capacity in (6) is also a random variable. Let us assume an equal distribution of the input power that transforms matrix $\mathbf{C}_X$ to a multiple of identity matrix $\mathbf{I}_N$ and with the constraint $\mathrm{Tr}(\mathbf{C}_X) \leq P$, we have $\mathbf{C}_X = \mathbf{I}_N$ and:

$$C = \log_2 \det\left(\mathbf{I}_M + \frac{1}{\sigma_n^2}\mathbf{H}\mathbf{H}^H\right), \text{ [bps/Hz]}. \quad (7)$$

IV. INFORMATION CAPACITY OF MIMO PARTICULAR CHANNELS

In this section, we study the capacity of a particular MIMO channels for different values of *M* and *N*.

• The graphical model of the particular MIMO channel with $M = 2,\ N = 2$ is illustrated in Fig. 1.

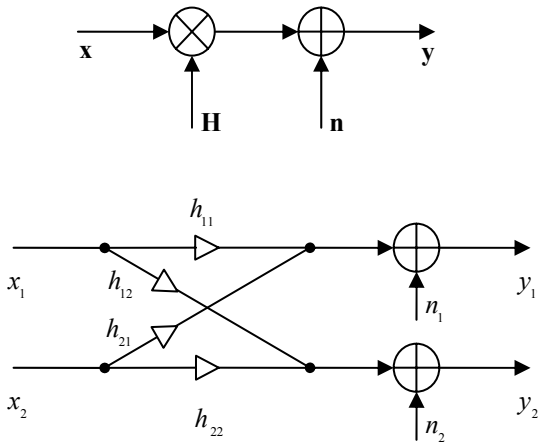


Fig. 1. The model of a MIMO Channel for $M = 2, N = 2$

The signal model can be written as:

$$\mathbf{y} = \mathbf{H}\mathbf{x} + \mathbf{n}\,, \tag{8}$$

where $\mathbf{x} = \begin{bmatrix} x_1 & x_2 \end{bmatrix}^T \in \mathbb{C}^{2\times 1}$ are the input data samples with the covariance matrix $\mathbf{C}_X = E\left[\mathbf{x}\mathbf{x}^H\right]$, $\mathbf{y} = \begin{bmatrix} y_1 & y_2 \end{bmatrix}^T \in \mathbb{C}^{2\times 1}$ are the output data samples, $\mathbf{n} = \begin{bmatrix} n_1 & n_2 \end{bmatrix}^T \in \mathbb{C}^{2\times 1}$ are the Gaussian noise samples and $\mathbf{H} = \left[h_{ij}\right]_{M\times N} = \left[\,|\,h_{ij}\,|\,e^{j\alpha_{ij}}\right]_{M\times N}$ is the fading channel matrix.

The mutual information is defined as ($\mathbf{I}_2$ is the unitary matrix):

$$I = \log_2 \det\left(\mathbf{I}_2 + \frac{1}{\sigma_n^2}\mathbf{H}\mathbf{C}_X\mathbf{H}^H\right), \text{[bps/Hz]}. \tag{9}$$

The information capacity of the MIMO channel, from Fig. 1, is:

$$C = \max_{\mathbf{C}_x} \log_2 \det\left(\mathbf{I}_2 + \frac{1}{\sigma_n^2}\mathbf{H}\mathbf{C}_X\mathbf{H}^H\right), \text{[bps/Hz]}. \tag{10}$$

For a flat fading channel, characterized by the coefficients $h_{ij} = 1$, the information capacity for a signal to noise ratio $SNR = \sigma_X^2 / \sigma_n^2$ can be determined for the two cases of an unknown or known channel at the transmitter side (see Table I, column 2). For a fading channel, if the attenuations are omitted, $|\,h_{ij}\,| = 1$, the matrix channel is $\mathbf{H} = \left[e^{j\alpha_{ij}}\right]_{M\times N}$. For an unknown channel, the coefficients α_{ij} are calculated so that $\mathbf{H} = 2\mathbf{I}_2$. The expression of information capacity for these cases is also shown in Table I.

• For the MISO channel with $M = 2$ and $N = 1$, the signal model can be written as:

$$y = \mathbf{H}\mathbf{x} + n\,, \tag{11}$$

where $\mathbf{x} = \begin{bmatrix} x_1 & x_2 \end{bmatrix}^T \in \mathbb{C}^{2\times 1}$ is the input data samples with the covariance matrix $\mathbf{C}_X$, $y \in \mathbb{C}$ is the output data sample, $n \in \mathbb{C}$ is the Gaussian noise samples and the fading channel matrix is $\mathbf{H} = \begin{bmatrix} h_1 & h_2 \end{bmatrix}$.

The information capacity of a MISO channel is calculated in the same manner as for MIMO channel and can be written as in Table I, column 3, for a flat fading channel with matrix $\mathbf{H} = \begin{bmatrix} 1 & 1 \end{bmatrix}$ and for a fading channel, in the case of a known or unknown channel at the transmitter.

• For the SIMO channel with $M = 1$ and $N = 2$, the signal model is:

$$\mathbf{y} = \mathbf{H}x + \mathbf{n}\,, \tag{12}$$

where $x \in \mathbb{C}$ is the input data sample with the covariance σ_X^2, $\mathbf{y} = \begin{bmatrix} y_1 & y_2 \end{bmatrix}^T \in \mathbb{C}^{2\times 1}$ are the output data samples, $\mathbf{n} = \begin{bmatrix} n_1 & n_2 \end{bmatrix}^T \in \mathbb{C}^{2\times 1}$ are the Gaussian noise samples and $\mathbf{H} = \begin{bmatrix} h_1 & h_2 \end{bmatrix}^T$ is the fading channel matrix. The information capacity of this channel is shown in Table I, column 4.

Table I. The particular expressions of information capacity for flat fading / fading MIMO channels determined for a known / unknown channel at the transmitter.

Information Capacity, [bps/Hz]	**MIMO**	**MISO**	**SIMO**	**SISO**
flat fading, unknown	$\log_2(1+4\cdot SNR)$	$\log_2(1+2\cdot SNR)$	$\log_2(1+2\cdot SNR)$	$\log_2(1+SNR)$
flat fading, known	$\log_2(1+8\cdot SNR)$	$\log_2(1+4\cdot SNR)$	$\log_2(1+2\cdot SNR)$	$\log_2(1+SNR)$
fading, unknown	$2\log_2(1+2\cdot SNR)$	$\log_2(1+2\cdot SNR)$	$\log_2(1+2\cdot SNR)$	$\log_2(1+SNR)$
fading, known	$2\log_2(1+2\cdot SNR)$	$2\log_2(1+SNR)$	$\log_2(1+2\cdot SNR)$	$\log_2(1+SNR)$

- For the SISO channel with $M = 1$ and $N = 1$, the signal model can be written as:

$$y = hx + n, \tag{13}$$

where $x \in \mathbb{C}$ is the input data sample with the covariance σ_x^2, $y \in \mathbb{C}$ is the output data sample, $n \in \mathbb{C}$ is the Gaussian noise sample and h is the fading channel coefficient.

For a flat fading channel, the capacity is shown in Table I, column 5 and it coincides with the standard Shannon capacity of a Gaussian channel for a given value of SNR.

V. CONCLUSIONS

In the previous section we determined the particular expressions of information capacity of MIMO channels for different channel model parameters: number of input / output nodes, flat fading / fading and for a known / unknown channel at the transmitter.

Based on the results from Table I, we make the following dependences analysis:

- **The information capacity increases as the number of input or output channel nodes increases** at the same SNR, in the following order SISO, SIMO, MISO, MIMO. If the number of input and output nodes are the same, the capacity increases at least linearly as a function of number of antennas. This dependence is not influenced if the channel is with or without fading or if it is known at the input nodes. Normally, the SISO channel has the lowest capacity and the MIMO channel with two input and output nodes has the highest capacity.

Fig. 2 illustrates the dependence of information capacity of SNR for different numbers of input and output nodes and for flat fading channel, known at the transmitter.

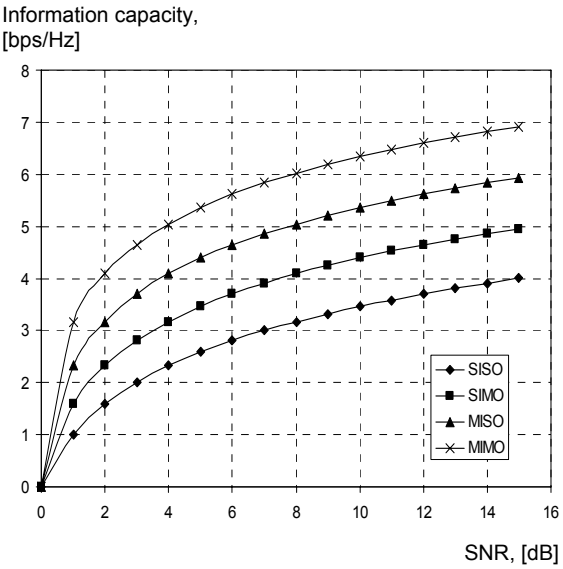


Fig. 2. The dependence of information capacity of SNR for different channel models.

At high SNR, the curves tends to be flat, because the information capacity for a MIMO channel without errors is limited and depends on various parameters, i.e. the system structure, channel parameters.

In this paper, the information capacity of a MIMO channel with different number of input and output nodes is analyzed. These results confirm the work from [8] and are the base for choosing the strategy by which nodes cooperates and for choosing the best architecture for information transport. These will be generalized, in further works, for MIMO channels with $M, N \geq 3$, in applications like wireless networks and LANs communications.

REFERENCES

[1] C. E. Shannon, "A Mathematical Theory of Communication", Bell Syst. Tech. Journal, vol. 27, pp. 379-423, 1948.

[2] I. E. Telatar, "Capacity of multi-antenna Gaussian channels," Eur. Trans. Telecomm.., vol. 10, no. 6, pp. 585–595, Nov-Dec. 1999.

[3] G. Foschini, M. Gans, "On limits of wireless communications in a fading environment when using multiple antennas," Wireless Pers. Commun., vol. 6, pp. 311–335, 1998.

[4] M. L. Honig, K. Steiglitz and B. Gopinath, "Multichannel signal processing for data communications in the presence of crosstalk," IEEE Trans. Commun., vol. 38, no. 4, pp. 551–558, Apr. 1990.

[5] A. Scaglione, G. B. Giannakis and S. Barbarossa, "Redundant filterbank precoders and equalizers. Part I: Unification and optimal designs," IEEE Trans. Signal Process., vol. 47, no. 7, pp. 1988–2006, Jul. 1999.

[6] T. M. Cover, J. A. Thomas, "Elements of Information Theory", John-Wiley, 1991.

[7] E. A. Jorswieck, H. Boche, "Performance Analysis of Capacity of MIMO Systems under Multiuser Interference Based on Worst-Case Noise Behavior", *EURASIP Journal on Wireless Comm. and Net*., no. 2, pp. 273-285, Dec. 2004.

[8] V. Tarokh, H. Jafarkhani and A. R. Calderbank, "Space-time block codes for high data rate wireless communication: performance results", *IEEE Journal on Selected Areas in Communications*, vol. 17(3), pp. 451–460, Mar. 1999.

Buletinul Ştiinţific al Universităţii "Politehnica" din Timişoara

Seria ELECTRONICĂ şi TELECOMUNICAŢII
TRANSACTIONS on ELECTRONICS and COMMUNICATIONS

Tom 51(65), Fascicola 2, 2006

Feature Based 2D Image Registration using Mean Shift Parameter Estimation

Daniela Fuiorea[1], Dan Pescaru[2], Vasile Gui[1], Corneliu I. Toma[1]

Abstract – A new method of feature based 2D image robust registration is proposed. The image distortion is modeled as a similarity transform with four parameters, estimated sequentially by 1D transforms, resulting in an increased sample density as compared to 4D space processing. By adopting a mean shift estimator, advantages of RANSAC and M-estimators can be combined within a single and sound theoretical framework. Experimental results confirm the validity of the proposed approach.

Keywords: image registration, robust estimation, mean shift, similarity transform

I. INTRODUCTION

Image registration is one of the basic image processing operations in many computer vision applications, like remote sensing, biomedical imaging, surveillance, robotics, multimedia etc [1]. The goal is to overlay two or more images of the same scene taken at different times, from different viewpoints, and/or by different sensors. To register two images, a transformation must be found so that each point in one image (reference image) can be mapped to a point in the second (sensed image). In other words, the transform geometrically "optimally" aligns two images. Due to the diversity of images to be registered and to various type of degradations it is impossible to design a universal method applicable to all registration tasks. Every method should take into account not only the assumed type of geometric deformation between images but also radiometric deformations and noise corruption, required registration accuracy and application-dependent data characteristics.

Registration methods consist of the following four steps:

- Feature detection
- Feature matching
- Transform model estimation
- Image resampling and transformation

Features can be specific image points are image areas. The present study concentrates on the first approach.

Suppose the feature detection and matching problems have been solved by an appropriate automatic method. It is well known that the correspondence problem is difficult in the general case and prone to errors. Even a single gross correspondence error can drive the solution far away from the real one. Therefore robust estimation methods are needed to cope with point correspondence errors. One of the first robust estimators proposed for image registration was the RANSAC estimator [2]. Recently M estimators and related kernel based estimators received much attention in the community of researchers looking for robust solutions in computer vision [3]. The two methods have complementary merits. The M-estimators find good solutions but require a good initial estimate to converge correctly. RANSAC does not need to start from an initial estimate [4], but the solution does not take into account all the available data, thus its precision is not maximized. In the present work, a mean shift [5] based solution is proposed for robust parameter estimation in image registration. Like RANSAC, the mean shift estimator does not require an initial estimate. At the same time, as the (related) M-estimators, the mean shift estimator makes a better use of the available inlier samples.

II. BRIEF REVIEW OF THE MEAN SHIFT

Given a sample of N d-dimensional data points, $\mathbf{x}_i$, drawn from a distribution with multivariate probability density function $p(\mathbf{x})$, an estimate of this density at $\mathbf{x}$ can be written as [4]:

$$\hat{p}_{\mathbf{H}}(\mathbf{x}) = \frac{1}{N}\sum_{i=1}^{N} K_{\mathbf{H}}(\mathbf{x}-\mathbf{x}_i) \qquad (1)$$

where

[1] Facultatea de Electronică şi Telecomunicaţii, Departamentul Comunicaţii Bd. V. Pârvan Nr. 2, 300223 Timişoara, e-mail daniela.fuiorea@etc.upt.ro
[2] Facultatea de Automatizări şi Calculatoare, Bd. V. Pârvan Nr. 2, 300223 Timişoara, e-mail dan@cs.utt.ro

$$K_{\mathbf{H}}(\mathbf{x}) = |\mathbf{H}|^{-1/2} K_{\mathbf{H}}(\mathbf{H}^{-1/2}\mathbf{x}) \quad (2)$$

is the kernel function depending on the symmetric positive definite $d \times d$ matrix $\mathbf{H}$, called bandwidth matrix. Frequently $\mathbf{H}$ has a diagonal form or even the form $\mathbf{H} = h^2\mathbf{I}$, assuming the same scale h for all dimensions, i.e. a single scale parameter and an isotropic estimator, K_h. A radially symmetric estimator can be generated starting from a 1D kernel function K_1 as:

$$K^R(\mathbf{x}) = \alpha K_1(\| \mathbf{x} \|), \quad (3)$$

with α is a strictly pozitive constant chosen such that the kernel function integrates strictly to 1. The profile of the radially symmetric kernel is defined as:

$$K^R(\mathbf{x}) = c_{k,d} k(\| \mathbf{x} \|^2), \quad (4)$$

with $c_{k,d}$ a normalization constant.

Starting from any location $\mathbf{y}$, a gradient ascent *mean shift* algorithm can be used to find the location of the maxima of the estimated PDF closest to the starting location. This can be simply done by iterating the equation

$$\mathbf{y}_{j+1} = \frac{\sum_{i=1}^{n} \mathbf{x}_i g\left(\left\| \frac{\mathbf{y}_j - \mathbf{x}_i}{h} \right\|^2\right)}{\sum_{i=1}^{n} g\left(\left\| \frac{\mathbf{y}_j - \mathbf{x}_i}{h} \right\|^2\right)}, \quad j = 1,2,... \quad (5)$$

where

$$g(x) = -k'(x) \quad (6)$$

until convergence. The proof of the convergence can be found in [4]. More, in practice the convergence is very fast, typically only two or three iterations being needed.

III. ROBUST IMAGE REGISTRATION

A widely used 2D geometric transformation in image registration is the similarity transform, consisting of rotation, translation and scaling. The model is defined by the equations:

$$\begin{bmatrix} u \\ v \end{bmatrix} = \begin{bmatrix} s & 1 \\ 1 & s \end{bmatrix} \begin{bmatrix} \cos(\varphi) & -\sin(\varphi) \\ \sin(\varphi) & \cos(\varphi) \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} t_x \\ t_y \end{bmatrix}, \quad (7)$$

relating the old pixel coordinates (x,y) to the new ones. As this transform preserves the angles and curvatures, it has been named "shape-preserving mapping". The four parameters of the transformation can be unambiguously determined from the correspondence of two pairs of points. However, in most of the cases, the number of the points available for estimating the transformation parameters s, φ, t_x and t_y is higher. By denoting the vector of parameters as

$$\mathbf{p} = \begin{bmatrix} s \\ \varphi \\ t_x \\ t_y \end{bmatrix}, \quad (8)$$

82

the problem of estimating the geometric transformation can be formulated as the problem of minimizing a measure of the matching error of the available data points:

$$\mathbf{p} = \underbrace{\arg\min}_{\mathbf{p}} \sum_i \rho(r_i), \quad (9)$$

where the residuals r_i represent estimations of the matching error between a pair of corresponding features after registration. By choosing:

$$\rho(r) = r^2, \quad (10)$$

a least-squares matching is obtained. This problem has been extensively studied in early work on point matching. See for example the frequently cited work [6] with the improvements from [7]. Because of the squaring up in equation (10), least squares fitting is notoriously sensitive to the presence of the outliers, data samples deviating widely from "typical" samples. The problem can be alleviated by using a different shape of the function in equation (10), in order to reduce the influence of the outlier samples. M-estimators [3] are one of the most notorious examples from this category. In the present work, we use a density estimation approach, based on the mean shift to obtain robust estimates of the similarity transform. The approach is closely related to the M-estimator, as pointed out in [5]. However, the interpretation of the density estimator is different. Links can be found between mean density estimators and the RANSAC as well, but this is a subject beyond the scope of the present paper.

The number of corresponding points available in different applications varies widely. We concentrate on the case where this number is relatively small and obtaining reliable estimates in the presence of outliers is more difficult. A useful step in order to obtain higher sample densities is to reduce the dimension of the search space. In this paper we propose a solution based on search in 1D spaces, as opposed to the general approach of simultaneous estimation of all parameters in a 4D space. We start from the observation that angles between line segments are not changed by translation or rescaling. Therefore, the rotation parameter, φ can be estimated based on such

angles prior to estimating the translation or rescaling parameters. Rescaling parameter estimation can also be done prior to translation or rotation estimation, based on distances between pairs of points. On the other hand, rotation or rescaling strongly affect translation parameters, as illustrated in the example in figure 1.

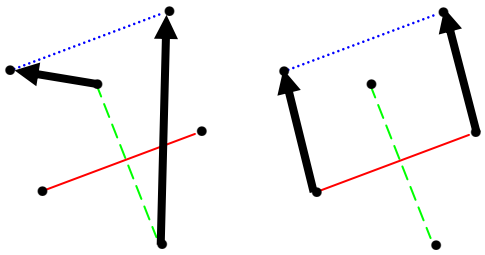


Fig. 1. Translation vectors before (left) and after (right) rotation compensation

From the example above, it is clear that to robustly estimate the translation vector components it has to be done *after* rotation and rescaling have been estimated and compensated.

Denote by $\{V_i\}$, $i = 1,2,\ldots, N$ a set of points from the reference image and $\{Q_i\}$, $i = 1,2,\ldots, N$ the corresponding points from the registered image. In the spirit of the RANSAC estimator, we form minimal sets of points, to estimate transform parameters. For rotation angle estimation, minimal set means pairs of points (Q_i, Q_j) and (V_i, V_j), with the corresponding vectors $\mathbf{q}_{ij}$ and $\mathbf{v}_{ij}$. The angle between the lines (Q_i, Q_j) and (V_i, V_j) is then given by the equation:

$$\cos(\varphi) = \frac{\mathbf{q}_{ij}^T \mathbf{v}_{ij}}{\left\|\mathbf{q}_{ij}^T\right\| \left\|\mathbf{v}_{ij}^T\right\|} \tag{11}$$

Denote by $\{\varphi_i\}$, $i = 1,2,\ldots,M$ the set of M angles obtained by pairs of points. The rotation angle estimate is defined as the highest density location obtained by the mean shift algorithm starting from all data samples, φ_i. This is the mean shift filtered data set. Notice that the denominator in equation (5) is - up to a factor - a measure of the sample probability density estimated with the shadow kernel of $K()$. Therefore, no additional processing is needed to compute and compare probability densities. Moreover, very fast mean shift implementations can be obtained by marking all locations visited by the algorithm through iterations and associating corresponding intervals to the location of convergence.

In a similar manner, scale factor estimates can be obtained from the sets of pairs of points using the equation:

$$s = \|\mathbf{v}_{ij}\| / \|\mathbf{q}_{ij}\|. \tag{12}$$

After performing the inverse geometrical transform to compensate for scale and rotation angle, robust translation vector component estimation is performed by point correspondences. Given a pair of points with position vectors $\mathbf{v}_i$ and $\mathbf{q}_i$, we form the translation data samples

$$\mathbf{t}_x = \mathbf{v}_{xi} - \mathbf{q}_{xi}, \tag{13}$$
$$\mathbf{t}_y = \mathbf{v}_{yi} - \mathbf{q}_{yi}, \tag{14}$$

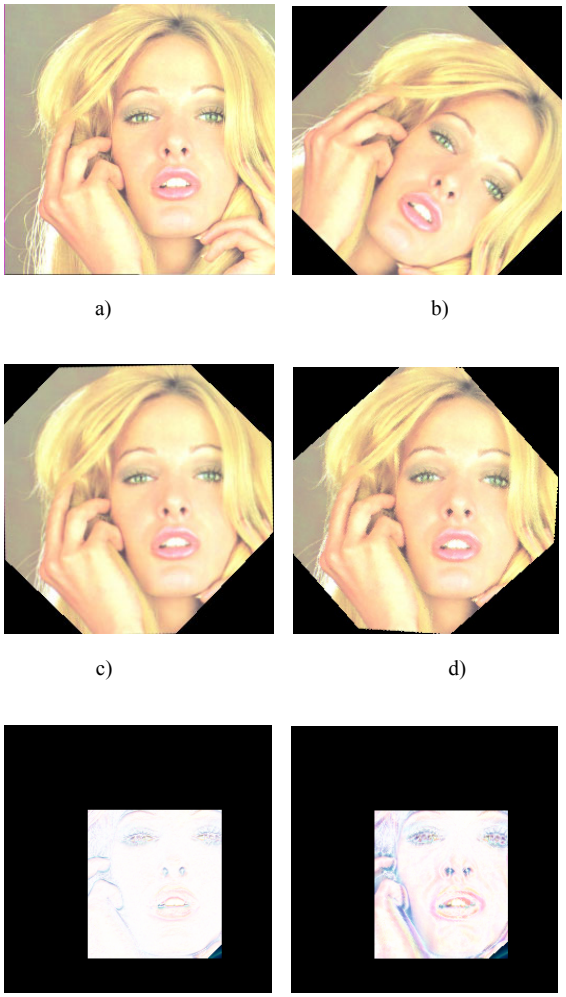
then proceed to translation parameter estimation using mean shift.

IV. EXPERIMENTAL RESULTS

In order to obtain qualitative assessment of the proposed registration method, artificial image pairs have been generated with known geometrical transformation parameters. Feature points have been selected interactively in both images. For reference, a least squares estimator [6][7], was also used, mostly to validate the performances of the proposed approach for small errors, where the lest squares estimator works at its best. Results for the case of a similarity transform consisting of a translation and a 45^o rotation are shown in figure 2. The original image is shown in figure 2a, while the similarity transformed image is shown in figure 2b. In figure 2c, the results of the robust registration method proposed in this paper for image pairs from figure 2a and figure 2b are illustrated. The same results for the least-squares registration are illustrated in figure 2d. In figure 2e and figure 2f, the matching errors for the robust registration and for the lest square registration methods are displayed. Note that the errors were evaluated only within the minimum area rectangle enclosing the feature points used for registration and that the error images are displayed in negative contrast, for better visibility. A careful examination of the error images in figure 2e and figure 2f reveal the presence of significantly higher errors for the least squares estimator as compared with the robust mean shift based estimator, both in terms of translation and rotation parameters.

In order to obtain quantitative evaluation of the performances of the proposed image registration technique, in a second series of experiments, we generated 1D data sets with controlled percentage of outliers. Both inlier and outlier samples were generated as uniformly distributed random sequences. The outlier samples were generated with a standard deviation 10, while the inlier samples were generated with standard deviation 1. Comparative results for the mean shift and least square estimated parameters are given in figures 3 and 4, for 10% and respectively 33% outlier percentage. The mean shift estimator was implemented with the Epanechnikov kernel and scale parameter $h = 2$. As theoretically expected, the mean shift estimator errors are virtually unchanged and low,

while the lest squares estimator errors are increasing with the outlier percentage.



a) b)

c) d)

e) f)

Fig. 2. a) Original image; b) Similarity transformed image; c) Mean shift registered image; d) Lest squares registered image; e) Matching error for the mean shift estimator; f) Matching error for the lest squares estimator.

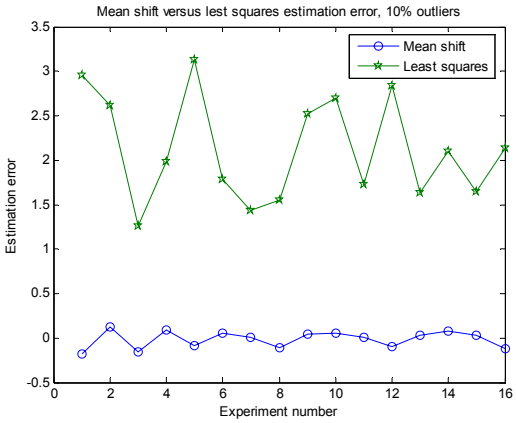


Fig.3. Mean shift versus least squares estimation errors, with outlier percentage 10%

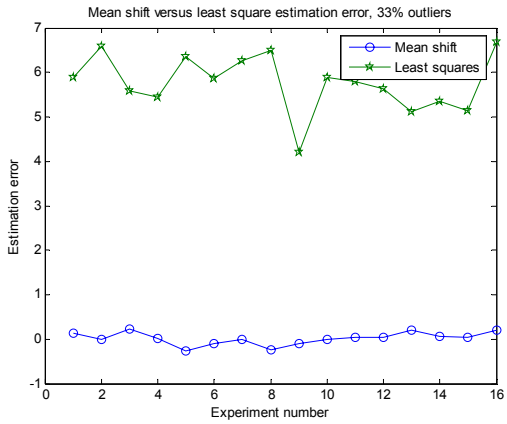


Fig.4. Mean shift versus least squares estimation errors, with outlier percentage 33%

V. CONCLUSION

The mean shift based 2D image registration method proposed proved to be reliable and computationally efficient in our work. It can safely tolerate a high percentage of spurious data. Unlike for the RANSAC type robust estimators, the feature space is searched in a systematic and computationally efficient manner. No initial guess solution is needed as in the case of M-estimators. By a careful analysis, the search in a 4D space has been replaced by four 1D searches. This technique results in an increased sample density in the lower dimensional space, making kernel density estimation performances potentially less dependent on bandwidth selection.

REFERENCES

[1] B. Zitova, J. Flusser, "Image registration methods: a survey", *Image Vision and Computing* 21*",* Elsevier, 2003, pp. 977-1000.

[2] M. A. Fischler and R. C. Bolles, "Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography". *Comm. Assoc. Comp.Mach*, 24(6):381–395, 1981.

[3] H. Chen, P. L. Meer, and D. E. Tyler, „Robust regression for data with multiple structures". In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1069–1075, 2001.

[4] V. Lepetit and P. Fua, "Monocular Model-Based 3D Tracking of Rigid Objects: A Survey". In *Foundations and Trends in Computer Graphics and Vision* Vol. 1, No 1 (2005) 1–89.

[5] D.Comaniciu, P.Meer, "Mean Shift: A Robust Approach toward Feature Space Analysis", *IEEE Trans. PAMI*, Vol.24, No.5, pp.603-619, 2002.

[6] K.S. Arun, T.S. Huang, S.D. Blostein: Least-squares fitting of two 3-D point sets. IEEE Transactions on Pattern Analysis and Machine Intelligence 9(5), 698-700, 1987.

[7] S. Umeyama: Least-squares estimation of transformation parameters between two point patterns. IEEE Transactions on Pattern Analysis and Machine Intelligence 13(4), 376-380, 1991.

Buletinul Ştiinţific al Universităţii "Politehnica" din Timişoara

Seria ELECTRONICĂ şi TELECOMUNICAŢII
TRANSACTIONS on ELECTRONICS and COMMUNICATIONS

Tom 51(65), Fascicola 2, 2006

Features Extraction from Romanian Vowels Using Matlab

Alina Nica[1], Alexandru Căruntu[1], Gavril Toderean[1], Ovidiu Buza[1]

Abstract –In this paper we developed in MATLAB a software environment in order to extract the main features from the Romanian vowels, which we intend to use in the synthesis step. We estimated speech parameters such as: energy, zero crossing rate (ZCR), fundamental frequency, formants. The used methods for obtaining the parameters are time domain analysis, cepstral analysis and Linear Predictive Coding (LPC). The analyzed vowels were uttered by several speakers and some experimental results are presented.
Keywords: speech analysis, vowels, features extraction, cepstrum, LPC

I. INTRODUCTION

In most applications of speech processing (e.g., synthesis, recognition), the speech analysis is the first step, and it involves the features extraction from the speech signal. By analyzing the speech signal, we want to obtain a useful representation of the speech waveform, in terms of parameters, that contain relevant information for speech synthesis. We used methods of speech analysis both in time domain (in this case the analysis is performed directly on the speech waveform), and in frequency domain (after a spectral transformation of the speech).
The representation of the speech waveform in terms of time-domain measurements include average zero-crossing rate, energy. These representations are very simple to implement and on their basis it is possible to estimate important features of the speech signal [1].
A very important speech analysis technique is the method of linear predictive analysis. This method provides an accurate representation of the basic speech parameters and is relatively efficient for computation [2].
The cepstral analysis involves the process of separating two convolutionally related proprieties by transforming the relationship into a summation. The importance of the cepstrum stems from the fact that it allows for the separate representation of the spectral envelope and fine structure.
We used the analysis methods mentioned above in order to extract the romanian vowels parameters, and the software environment, to perform it, was developed in MATLAB.
The Romanian language has seven vowels: *a ,e ,i, o, u, ă, î* and accordingly to the phonetically researches they appear with a frequency of 45,16%[3]. Generally romanian vowels are oral sounds, excepting the situations when they are neighboring with a nasal consonant; in this case they become also nasal sounds[4].
Vowels are voiced sounds and they are produced with the vibration of the vocal folds. The frequency of the vocal-folds oscillation is the fundamental frequency of the speech signal [5], [6]. The mean value for the fundamental frequency for male voices is 125 Hz and for female voices is 250 Hz [2].
The principal resonant structure, particularly for the vowels, is known as the vocal tract, and the resonance frequencies are called formants and by convention they are numbered from the low-frequency end and are usually referred to as F_1, F_2, F_3, etc. The most significant formants in determining the phonetic properties of speech sounds are generally F_1 and F_2 (ranged between 250 Hz and 3 kHz); but for certain phonemes some higher-frequency formants can also be important.
The the paper is organized as follows: in Section II we review briefly the analysis methods used to extract the speech parameters and some examples on different vowels are presented, too; in Section III are given a few of the experimental results; finally, in Section IV are the conclusions.

II. FEATURES EXTRACTION

Many analysis techniques have been developed in time. In our experiments we have chosen the most used of them. This section presents shortly a description and some details of their implementation.

A. *Sound spectrogram analysis*

An important tool for speech analysis is the *spectrogram*, which converts a two-dimensional speech waveform (amplitude-time) into a three-dimensional pattern (time-frequency-amplitude). Thus, on a spectrogram, time and frequency are displayed in its horizontal and vertical axes, respectively, and amplitude is noted by the darkness of the display [7], [8]. Peaks in the spectrum, corresponding to the formants, appear as dark

[1] Technical University of Cluj-Napoca,
e-mail: Alina.Nica@com.utcluj.ro

horizontal bands and the vertical stripes correspond to the fundamental frequency. To compute a spectrogram the short-time Fourier analysis is used. The speech signal is time-varying, but speech analysis assumes that the signal properties change relatively slowly on short periods of time (10 to 30 ms), so that the signal characteristics can be considered uniform in that regions. Consequently, the speech signal is decomposed into a sequence of short segments, referred to as *analysis frames* and each one is analyzed independently. This technique is called *short-time analysis*.

For a given signal *s[m]*, the short-time signal $s_n[m]$ of frame *n* is defined as:

$$s_n[m] = s[m]w_n[m]. \quad (1)$$

the product of *s[m]* by a *window function* $w_n[m]$, which is zero everywhere except in a small region.

Prior to frequency analysis, the frames are multiplied by a tapered window, in order to reduce any discontinuities at the edges of the selected region; otherwise it could appear some spurios high-frequency components into the spectrum. The most used windows are Hamming and Hanning windows. The length of the analysis window must give an adequate time and frequency resolution. A common compromise is to use a 20-30 ms window applied at 10 ms intervals.

There are two types of spectrograms: *wide-band spectrograms*, which use short windows (<10 ms) and *narrow-band spectrograms*, which use long windows (>20 ms). Wide-band spectrograms are useful in viewing vocal tract parameters (formant frequencies), while narrow-band spectrograms are good for fundamental frequency estimation.

B. *Time domain analysis*

By analyzing the speech signal in time domain, some important features can be estimated: maximum and medium amplitude, energy, zero-crossing rate, fundamental frequency.

Short-time energy of the speech wave is defined with the equation:

$$E_n = \frac{1}{N}\sum_m [s(m)w(n-m)]^2 . \quad (2)$$

where *N* is the number of samples, *w* is a window used for analysis and *s* is the speech signal. It provides a convenient represantation of the amplitude variation over time. Energy emphasizes high amplitudes (the signal is squared in calculating the energy). Voiced segments have high energy and unvoiced segments have much lower energy. In order to reflect accurately the variations of the signal amplitude, the choise of window duration is very important: if it is too short, the energy will depend exactly of the waveform and if it is too large, the variations of the signal amplitude will not be reflected very corectly.

Zero-crossing rate (ZCR) is a simple measurement and provides adequate spectral information at a low cost. The short-time average zero-crossing rate is defined as:

$$ZCR_n = \frac{1}{2}\sum_m \left|\mathrm{sgn}[s(m)] - \mathrm{sgn}[s(m-1)]\right| w(n-m). \quad (3)$$

where *s* is the signal, sgn is the *signum* function and *w* is a window. This parameter is a simple measure of the dominant frequency of a signal. It is useful in differentiating between voiced and unvoiced signals, because unvoiced speech have much higher ZCR values than voiced speech. A suggested boundary is 2500 crossing/s, since unvoiced and voiced speech average about 4900 and 1400 crossing/s, respectively [7]. The zero-crossing rate can be used in the phone segmentation when preparing a database for concatenative Text-to-Speech synthesis [9].

C. *LPC analysis*

Linear Predictive Coding is a very important tool in speech analysis. A parametric model is computed based on least mean squared error theory, this technique being known as linear prediction (LP). LPC has been used to estimate fundamental frequency, vocal tract area functions, but it primarily provides a small set of speech parameters (called LPC coefficients) that represent the configuration of the vocal tract (the formants) [7], [10], [11].

LPC estimates the speech signal based on a linear combination of its *p* previous samples [1]. A larger *predictor order p* enables a more accurate model. In Figure 1 is an example of LPC spectra using different values of *p*, for vowel *a* (a1.wav).

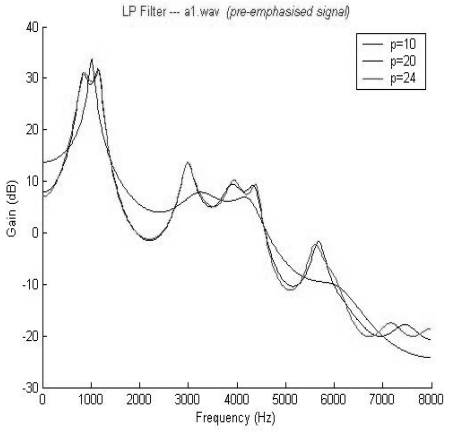


Fig. 1. LPC spectrum for vowel *a* (a1.wav) for different values of predictor order *p*

By removing the formant effects from the speech signal, the residual signal is obtained. Prior to speech analysis is recommended to pre-emphasis the signal, in order to emphasize the low frequencies in the speech spectrum.

D. *Cepstral analysis*

Cepstral analysis provides a method for separating

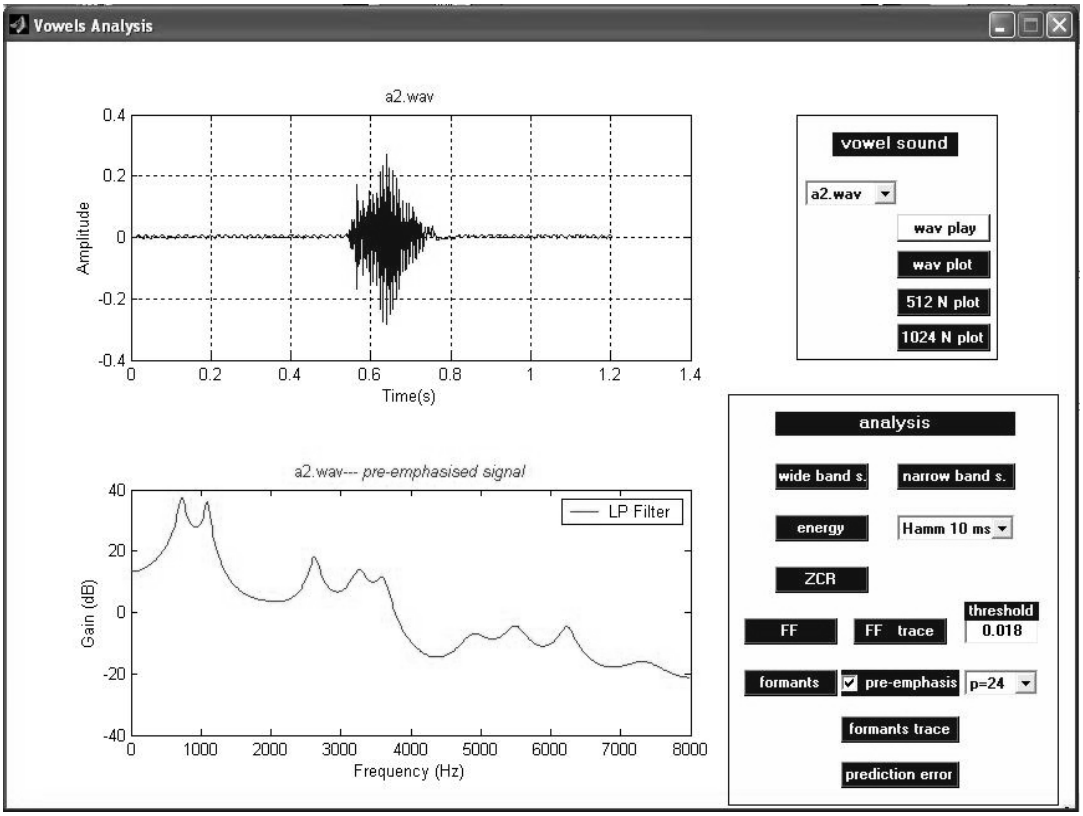


Fig. 2. The Vowels Analysis interface

the vocal tract information from excitation.

The process of passing an excitation signal through a vocal-tract filter to generate a speech signal can be represented as a process of convolution in time domain, which is equivalent to multiplying the spectral magnitudes of the source and filter components. If the spectrum is represented logarithmically, these components are additive and it is much easier to separate them using filtering techniques. *Cepstrum* is defined as the inverse Fourier transform of the short-time logarithmic amplitude spectrum [12]. The cepstrum can be used to determine the fundamental frequency of voiced speech, because the part of the cepstrum corresponding to the source is often manifested as a single pike.

In Figure 3 is presented an example of fundamental frequency estimation for vowel *e* (e1.wav).

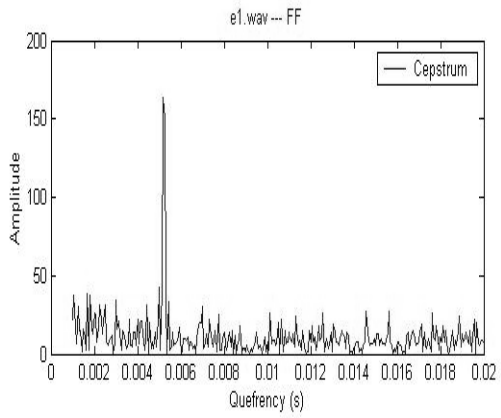


Fig. 3. Fundamental frequency estimation for vowel *e* (e1.wav) using cepstral method

The location of this peak gives the measurement for the frequency of the source signal.

III. IMPLEMENTATION AND EXPERIMENTAL RESULTS

Our application was developed in MATLAB [13], [14]. We named it *Vowels Analysis* and its interface is presented in Figure 2. It has two panels: *vowel sound* and *analysis*, and on them, there are some buttons, which perform different tasks. Thus, it can be displayed the entire waveform of the speech signal (or only a few samples of it) and the corresponding narrow-band and wide-band spectrograms.

Also, the user can visualize some of the speech signal features: energy, zero-crossing rate, fundamental frequency, formants, formants trace, prediction error.

The sounds used in our experiments were recorded in wave format, at a sampling rate of 16 kHz, and they were uttered by several speakers (men and women).

We obtained one of the most important feature, which characterizes the vowels, the formants. In order to obtain the formants, we have done experiments using different values for the prediction order *p*, and varying the degree of pre-emphasis.

In Figure 2 it is illustrated the estimation of the formants for vowel *a* (a2.wav) using pre-emphasized signal and a value of the prediction order *p* of 24.

For example, according to [15], the romanian vowel *a* has the average value for the first three formants as follows: F_1=700 Hz, F_2=1300 Hz and F_3=2600 Hz.

The mean values obtained by us, for the first three formants of vowel *a* are: F_1=744 Hz, F_2=1172 Hz and F_3=2744 Hz.

Also, we extracted the fundamental frequency of the speech waves and the fundamental frequency trace can be visualized. In Figure 4 is presented an example of fundamental frequency trace for vowel *e* (e1.wav)

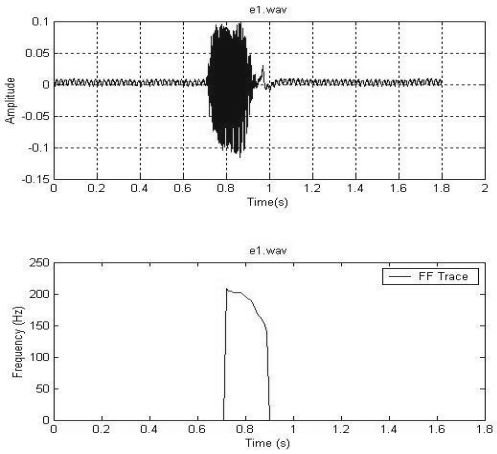


Fig. 4. Waveform and fundamental frequency trace for vowel *e* (e1.wav)

As well, we have measured the energy (we used Hamming window of different lengths) and the zero-crossing rate.

IV. CONCLUSIONS

In this paper, a few analysis techniques, which we have used to extract the main features of the speech signal, were briefly described. Next, we presented the interface of the software tool which we implemented in MATLAB, in order to extract the speech parameters. This application provided the means of some of the aspects of speech processing theory in a graphical manner. As well, we obtained the most important features for the vowels, which will be necessary in the synthesis stage.

We intend to develop this software environment, in order to obtain a database with the speech signal parameters. The goal of our future work is to experiment the concatenative synthesis method and to perform some prosody modifications on the synthesized speech signal.

REFERENCES

[1] L.R. Rabiner and R.W. Schafer, *Digital Processing of Speech Signals*, Prentice-Hall, Englewood Cliffs, 1978.

[2] S. Furui, *Digital Speech Processing, Synthesis, and Recognition,* Second Edition, Revised and Expanded, Marcel Dekker, Inc., 2001.

[3] S. Puşcariu, *Limba română-Rostire*, vol. I, Editura Minerva, Bucureşti, 1976.

[4] V. Şerban, *Fonetica*, Editura Augusta, Timişoara, 1997.

[5] J. Holmes and W. Holmes, *Speech Synthesis and Recognition*, Second Edition, Taylor&Francis, 2001.

[6] B. Yegnanarayana and N.J. Velduis, "Extraction of Vocal-Tract System Characteristics from Speech Signal", *IEEE Transaction on Speech and Audio Processing,* Vol. 6, No. 4, pp. 313-327, July 1998.

[7] D. O'Shaghnessy, *Speech Communications, Human and Machine*, Second Edition, IEEE Press, Inc., New York, 2000.

[8] T. Dutoit, *Introduction au traitement automatique de la parole*, *Notes de cours*, Faculte Polytechnique de Mons, Belgium, 2000.

[9] T. Zang and C.-C. Jay Kuo, "Audio Content Analysis for Online Audiovisual Data Segmentation and Classification",", *IEEE Transaction on Speech and Audio Processing,* Vol. 9, No. 4, pp. 441-457, May 2001.

[10] X. Huang, A. Acero and H.W. Hon, *Spoken Language Processing: A Guide to Theory, Algoritm and System Development*, First Edition, Prentice-Hall, 2001.

[11] G. K. Vallabha and B. Tuller, "Systematic errors in the formant analysis of steady-state vowels", *Speech Communication,* Vol. 38, pp. 141-160, 2002.

[12] http://www.utdallas.edu/~loizou/ee6362/lec6.pdf

[13] http://www.ee.ic.ac.uk/hp/staff/dmb/voicebox/voicebox.html

[14] http://svr-ww.eng.cam.ac.uk/~ajr/SA95/SpeechAnalysis.html

[15] I.T. Stan, *Fonetica*, Editura Presa Universitară Clujeană, Cluj-Napoca, 1996.

Buletinul Ştiinţific al Universităţii "Politehnica" din Timişoara

Seria ELECTRONICĂ şi TELECOMUNICAŢII
TRANSACTIONS on ELECTRONICS and COMMUNICATIONS

Tom 51(65), Fascicola 2, 2006

Image filtering and enhancement using directional and anisotropic diffusion techniques

Romulus Terebes[1], Monica Borda[2], Ioan Nafornita[3]

Abstract – A novel diffusion filter for low-level image processing is proposed. Analyzing the drawbacks of Gaussian convolution based regularization of partial derivatives equations, we propose an alternate method that employs anisotropic diffusion techniques to pre-smooth an image. The new technique is developed within the framework of previously proposed directional diffusion processes. Through a statistical interpretation we prove that the new filter produces consistently better results than the original version, especially when dealing with oriented textures having different spatial frequencies. Application samples are also provided in the final part of the paper.

Keywords: diffusion, anisotropic, orientation

I. DIFFUSION FILTERS FOR IMAGE SMOOTHING AND ENHANCEMENT

In recent years a lot of research was done for proposing various diffusion based image filtering and enhancement techniques. The simplest diffusion equation is the isotropic filter that relies on the classical heat equation to smooth an image. Let *U(x,y,t)* denote the gray level of a pixel of coordinates *(x,y)* at some instant *t*. The partial derivatives equation (PDE) that drives the diffusion process is:

$$\frac{\partial U}{\partial t} = div(\nabla U) = \Delta U \quad . \qquad (1)$$

(1) is usually solved by iterative means and, as time advances, smoothed versions of the original image $U(x,y,0) = U_0(x,y)$ are produced**.** As pointed out by Koenderink [4], the solution of the isotropic diffusion equation at time $t = \frac{\sigma^2}{2}$ is equivalent with a convolution between the original image and a Gaussian kernel of standard deviation σ:

$$U_\sigma = U(x,y,\frac{\sigma^2}{2}) = G_\sigma * U(x,y,0) \quad . \qquad (2)$$

Perona and Malik were the firsts to consider anisotropic behavior for diffusion processes. They proposed in [5] an anisotropic diffusion equation that is driven by a non constant diffusivity *c(|∇U(x,y,t)|)*. *c(.)* plays the role of an edge detector that penalizes the intensity of the smoothing process in regions where gradient norms $|\nabla U|$ are large (e.g. edges):

$$c(|\nabla U(x,y,t)|) = g(|\nabla U|) = \frac{1}{1 + (|\nabla U| / K)^2} \qquad (3)$$

The behavior of their anisotropic diffusion equation:

$$\frac{\partial U}{\partial t} = div[c(|\nabla U(x,y,t)|)\nabla U] \quad , \qquad (4)$$

can be more easily understood if its directional interpretation [6] is considered:

$$\frac{\partial U}{\partial t} = g(|\nabla U|)U_{\xi\xi} + [g(|\nabla U|) + |\nabla U| g'(|\nabla U|)]U_{\eta\eta} \, . \qquad (5)$$

For the type of diffusivity functions proposed by Perona and Malik, in the edge directions - $\vec{\xi} = (-\frac{U_y}{|\nabla U|}, \frac{U_x}{|\nabla U|})$ - the diffusion process will always have a smoothing action *(g(.)>0)* , whereas in the direction of gradient vectors - $\vec{\eta} = (\frac{U_x}{|\nabla U|}, \frac{U_y}{|\nabla U|})$ - smoothing can take place *(g'(.)>0)* or, for gradient norms greater than the diffusion threshold *K,* the equation can behave like an inverse diffusion filter that enhances edges *(g'(.)<0)*.

The results obtained by the authors are impressive, edges are kept better and noise is eliminated.

Even if edge enhancement is desired in the original model, Catte et al. pointed out [1] that negative diffusion coefficients can make the diffusion equation instable. They also argued the fact that if noise is important, edge enhancement can amplify also the

[1],[2] Universitatea Tehnică din Cluj-Napoca, Facultatea de Electronică şi Telecomunicaţii, Catedra de Comunicaţii, Str. Bariţiu, nr.26-29, 400027, Cluj-Napoca, e-mail Romulus.Terebes@com.utcluj.ro, Monica.Borda@com.utcluj.ro
[3] Facultatea de Electronică şi Telecomunicaţii, Departamentul Comunicaţii Bd. V. Pârvan Nr. 2, 300223 Timişoara, e-mail ioan.nafornita@etc.upt.ro

noise level to theoretically unbounded levels. The solution proposed by Catte et. al consists in pre-smoothing the image prior to the estimation of the diffusivities:

$$c(|\nabla U(x,y,t)|) = g(|\nabla(G_\sigma * U)|) \qquad (6)$$

The benefit of the Gaussian convolution is twofold: from a practical point of view influence of noise is diminished and, from a theoretical point of view, the diffusion equation becomes well posed and admits a unique solution. The same idea is encountered in more elaborate diffusion filters that were proposed since: the edge [8] or coherence enhancing diffusion [9] filters proposed by Weickert, the flow coherence diffusion filter we proposed in [7] etc.

II. PROPOSED METHOD

Addressing specific problems appearing when filtering, restoring or enhancing images composed of oriented patterns, we proposed in [6] an efficient method for low level processing of this type of images. The PDE model for our filter was:

$$\frac{\partial U}{\partial t} = \frac{\partial}{\partial \xi}[g^{\xi}(U_{\xi})U_{\xi}], \qquad (7)$$

where $\vec{\xi}$ denotes the eigenvector corresponding to the smallest eigenvalue of the gradient autocorrelation matrix:

$$M = \frac{1}{N}\begin{pmatrix} \sum_{i=1}^{N}(U_i)_x^2 & \sum_{i=1}^{N}(U_i)_x(U_i)_y \\ \sum_{i=1}^{N}(U_i)_x(U_i)_y & \sum_{i=1}^{N}(U_i)_y^2 \end{pmatrix}. \qquad (8)$$

$\vec{\xi}$ points to a direction orthogonal to the mean direction of the gradient vectors and its orientation *(θ)* represents the mean orientation of a structure passing through the pixel under study. As shown theoretically in [3], orientation estimation using (8) is highly robust against additive Gaussian like noise.

When additive noise is considered, provided that the orientation of the underlying oriented textures can be correctly estimated, the values of the directional derivatives U_ξ are depending only on the noise level and not on the local signal to noise ratio. Local maxima of $|U_\xi|$ are characterizing abrupt orientation changes (e.g. corners and junctions) whereas on region like areas $|U_\xi|$ depends only on the noise level. The directional interpretation of.(7):

$$\frac{\partial U}{\partial t} = [g^{\xi}(U_{\xi}) + g^{\xi}{}'(U_{\xi})U_{\xi}]U_{\xi\xi} = c_{\xi}U_{\xi\xi}, \qquad (9)$$

shows that, in each pixel, the equation acts as a one-dimensional diffusion process that smoothes *($c_\xi>0$)* oriented patterns with energy independent speed and can enhance corners and junctions for negative diffusion coefficients *($c_\xi<0$)*. The above discussion and described behavior of our filter are of course valid only if noise has low values. Only under this assumption maxima of $|U_\xi|$ can be directly associated to corners and junctions.

For dealing with images composed both of regions and oriented textures we proposed also a 2D version for our filter:

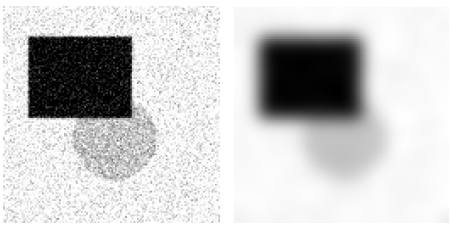
$$\frac{\partial U}{\partial t} = \frac{\partial}{\partial \xi}[g^{\xi}(U_{\xi})U_{\xi}] + \frac{\partial}{\partial \eta}[g^{\eta}(U_{\eta})U_{\eta}] \qquad (10)$$

In (10) $\vec{\eta}$ denotes the eigenvector associated to the biggest eigenvalue of (8) i.e. the mean direction of the gradient vectors. In contrast to (7) the new equation allows smoothing of regions like areas and is capable of enhancing edges. (10) is essentially a superposition of 1D diffusion processes that, unlike divergence equations as (4), allows a complete control of its behavior. Different thresholds can be chosen on the two directions, different functions can be employed for $\vec{\eta}$ and $\vec{\xi}$ etc.

Influence of heavy tailored noise on the results obtained by diffusion processes can be diminished using Gaussian convolution (or equivalently an isotropic diffusion) in a pre-processing step. This was the solution we employed in [6] when proposing a regularized version for the equation:

$$\frac{\partial U}{\partial t} = \frac{\partial}{\partial \xi}[g^{\xi}(\frac{\partial}{\partial \xi}U_{\sigma_\xi})U_{\xi}] + \frac{\partial}{\partial \eta}[g^{\eta}(\frac{\partial}{\partial \xi}U_{\sigma_\xi})U_{\xi}] \qquad (11)$$

Convolution with a Gaussian kernel is essentially a low pass filter and when embedding it in anisotropic diffusion processes some precautions have to be taken. Larger kernel sizes are efficiently filtering out spurious noise (Fig.1), but on the same time they are eliminating objects with spatial dimensions inferior to the standard deviation of the associated Gaussian function. Another drawback of Gaussian convolution is the fact that it produces inherent edge displacement (Fig.2) and, due to the edge enhancing term, a diffusion model based on (11) could produce artifacts such as false edges that can be further enhanced as time advances. If diffusion thresholds are chosen to have large values, by diminishing the gradient norms, Gaussian convolution forbids any edge enhancement process.



a) b)

Fig.1 Isotropic diffusion a) Original image b) Smoothed image

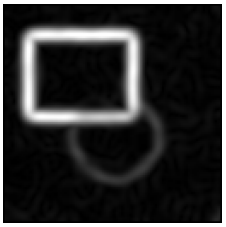


Fig.3 Edge displacement due to Gaussian convolution

For avoiding the above-mentioned effects we propose a method that employs a different pre-smoothing technique, based on a Perona and Malik filter. By denoting the solution of (4) at some instant t with $U_{PM,\,t}$ we consider the following evolution equation:

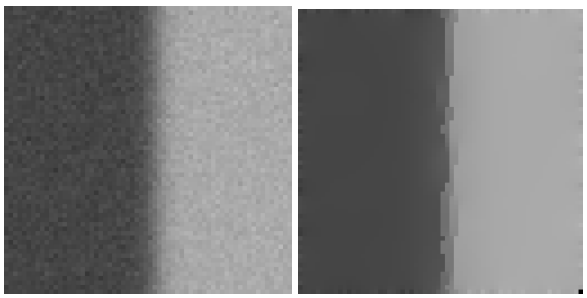
$$\frac{\partial U}{\partial t} = \frac{\partial}{\partial \xi}[g^{\xi}(U_{PM,t_\xi})U_{\xi}] + \frac{\partial}{\partial \eta}[g^{\eta}(U_{PM,t_\eta})U_{\eta}] \quad (12)$$

Since an anisotropic diffusion process outperforms an isotropic one, we expect better results when filtering an image with the modified equation (12).

However, the influence of the two extra parameters for the Perona and Malik process – the scale t and the threshold K – is still to be discussed.

We showed in [6], [7] that our original filter produced better results than classical filters. We found experimentally out that the optimal results were obtained for limited sizes of the Gaussian kernel: $\sigma = 0.75 \div 1$. The pre-smoothing scale t for (12) can be thus chosen according to Koenderink's observation: $t = \sigma^2/2$. When implementing diffusion equations with explicit discrete schemes, a time step $dt = 0.2$ satisfies the stability constraints for the 2D case; this leads to a number of about 5 iterations that will introduce the same amount of smoothing but in an anisotropic way.

The choice of the threshold K can also influence strongly the results. An undesired effect that might appear in the so-called staircase effect, well documented for the anisotropic diffusion equation [11]. For particular choices of K, due to edge enhancement term, some contours might get irregular when processing the image with (4) (Fig.3).



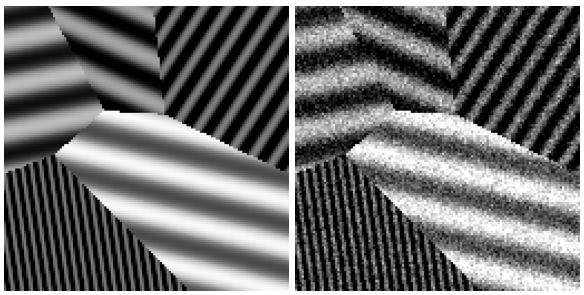
a) b)

Fig.3 Anisotropic diffusion. a) Original image
b) Irregular contours (staircase effect)

For the diffusion function (4) the effect appears only for gradient norms inferior to $\sqrt{3}K$ [11] and, thus, it can be avoided for sufficiently large K's. Following the technique indicated by Perona and Malik we choose to set K equal to some percentage (60 % in all our experiments) of the integral value of the gradient norms histogram [5].

III. EXPERIMENTAL RESULTS

PDE based models have a large number of parameters and comparisons between them are not always straightforward. A particular choice of parameters may suit well an image and could be less optimal for others. For solving this problem we took an experimental approach to solve this problem: we considered 15 randomly generated images, composed of oriented patterns, affected by Gaussian white noises and, for a given method and for each image, we searched for a best filtered result by allowing all parameters to vary. To quantify objectively the results we used the classical *PSNR* measure. Results from [10] are indicating that a 0.5dB improvement in terms of PSNR should be visible on the processed images. As processing methods we consider the new approach (12), its previous version (11) and the classical Perona and Malik filter (PM). The nature of the image we are interested in is shown on Fig.4.



a) b)

Fig.4 Image composed of oriented patterns a) Noise free image
b) Degraded image (*PSNR=15dB)*

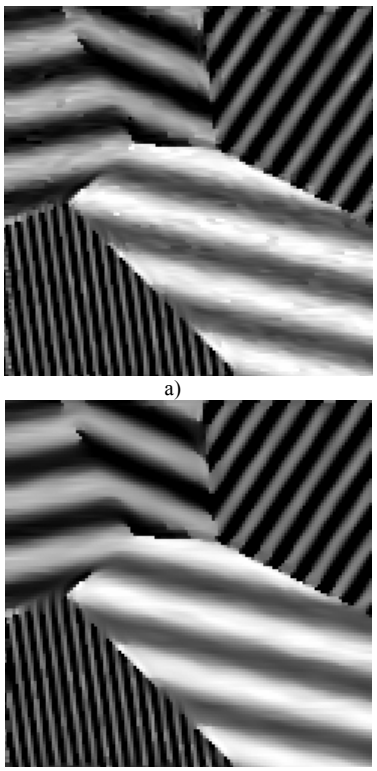
The noise levels and the *PSNR's* corresponding to the best filtered results for each considered method are shown on Table 1.

Table 1 PSNR values obtained for the 15 images under study

Image	Noise levels [dB]	Best filtered results - [dB]		
		PM	Original [6]	Proposed
1	16.66	25,3617	30,4611	30,8752
2	14.07	21,6275	24,7432	25,234
3	14.67	24,2511	25,8878	26,1256
4	15.60	23,6651	26,7267	26,9073
5	15.16	24,4292	26,3235	26,53745
6	13.68	24,4584	25,3846	26,0875
7	15.00	25,2881	27,8959	28,852
8	14.95	24,2299	26,3201	27,2197
9	14.64	26,1899	27,689	28,256
10	14.39	23,8604	26,0956	26,6131
11	14.85	24,6417	27,0079	27,5345
12	16.10	24,5689	26,4134	26,93394
13	13.27	22,0977	25,0166	25,485
14	16.65	26,2491	27,4969	28,0894
15	14.14	25,5648	25,7799	26,7116

Both our filters are outperforming the classical anisotropic diffusion equation. In terms of relative performances between our approaches, we obtain

systematically better results with the new filter. Quantitatively, the improvements in terms of PSNR are ranging from 0.18dB (for the fourth image) to 0.95 dB (for the seventh image). The following PSNR's are obtained for the set of images: 24.43dB for the Perona and Malik filter, 26.61dB for the original approach and 27.16dB for the new equation. In terms of visual results, we are showing bellow the best filtered images corresponding to both our methods.



a) b)

Fig.5 Best filtered results for the image from Fig.4.
a) Result obtained using (11): PSNR=27.89dB;
b) Result obtained using (12): PSNR=28.85dB

The improvement of almost 1dB is clearly visible and can be explained theoretically. The high frequency region placed on the bottom left hand side of the image strongly limits the Gaussian kernel's size. Consequently, on lower frequency regions Gaussian of the image pre-smoothing is unable to diminish the noise influence and, besides corners and junctions, local maxima of directional derivatives are appearing also on the oriented part of the image. Anisotropic diffusion based pre-smoothing does not suffer from the above mentioned effects and better results are obtained.

A question that may arise is related to the variability of the results from Table 1. A non-parametric two-way rank analysis of variance (ANOVA) [2] (Table 2) allows us to isolate the two sources of variability: the nature of the images and the different behavior of each method.

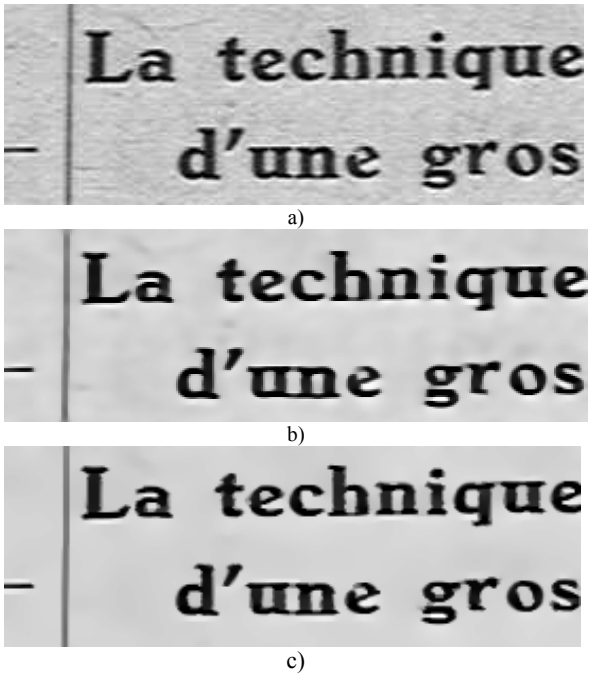
As the results from Table 2 are showing, more than 93 % of the variability between the obtained results is due to both the choice of the processing method and to the nature of the images. The two-way ANOVA allows us to isolate and investigate only the method effect. The extremely low probability $(p=4.9*10^{-13})$ associated to a Fisher-Snédécour test (F) allows us to conclude that the processing method has a very significant influence over the quality of the processed result.

Table 2 Two way non-parametric ANOVA [2]

Source of variance	Sum of squares	Degrees of freedom	Mean squares	F	P
Total	7890.00	44	172.5		
Between images	3076.00	14	219.7		
Between methods	3917.73	2	1959	91,99	$4.9*10^{-13}$
Residual	596.27	28	21.30		

We are also interested in building a hierarchy for the analyzed methods. The mean ranks, computed for each method over the 45 measurements, are: 10.06 for the Perona and Malik filter, 27.2 for our original method and 31.73 for the improved one. The three ranks can be compared using a classical Student-Newman-Keuls post-hoc test (SNK)[2]. Its critical values, computed for a 5% risk, are: 3.45 for comparing two consecutive ranks and 4.16 for comparing two values spanning three ranks. Using the SNK test we can that conclude that the new method is better that the original one and that both our methods are significantly better than the original anisotropic diffusion equation.

Some results obtained for a real gray scale image are shown in Fig.6. Starting from the original image (Fig. 6.a), containing both oriented patterns and region like areas, we first processed it with the improved filter (Fig. 6.c). We then considered the filtered result as an original noise free image and searched for the choice of parameters for the original version that produces the closest result in terms of PSNR (Fig. 6.b).



a) b) c)

Fig.6 Results for a real image. a) Original image
b) Result obtained using (11) c) Result obtained using (12)

A different effect is observable. The Gaussian regularization employed by our original formulation smoothes edges of the processed image and leads to a slightly blurred result. The new formulation does not suffer from this effect since edges are kept better when using anisotropic diffusion pre-smoothing. The filtered result is not blurred and background is also more efficiently filtered.

A third experiment deals with a color image shown in Fig.7. The degradations are much more severe than for the image in Fig.6. and they are consisting in moiré effects and blocking artifacts, dues to the insufficient resolution of the scanning device and to the presence of high frequency details.

Fig.7 Color image

For processing the image we implemented a straightforward extension of the algorithm presented in section II. The image is first decomposed on the constituent red, green and blue channels; each channel is then processed with the same set of parameters and the filtered results are then recomposed.

In Fig. 8, Fig. 9.a and Fig. 9.c we are presenting respectively the original red channel image and the filtered results corresponding to both our approaches. We used the same approach for establishing the parameters: first we computed a result with the proposed method that we judged the best and then we searched for those parameters of the Gaussian formulation of our filter that are producing the closest results in terms of PSNR values.

Fig.8 Green channel of the image in Fig. 7

Once again the new approach proves to be more efficient than its Gaussian formulation. Even if edges are more regular when preprocessing with the Gaussian filter, the anisotropic diffusion formulation allows true edge enhancing and efficient background restoration.

a)

b)

Fig.9 Results for the green channel a) Original image Result obtained using (11) c) Result obtained using (12)

a)

b)

Fig.10 Results for a color image a) Result obtained using (11) b) Result obtained using (12)

The combined effect of processing all the color channels is illustrated in Fig.10. The same behavior is observable; the anisotropic diffusion based preprocessing allows a better restoration of both regions like areas and of important edges in the image.

IV. CONCLUSIONS

We proposed a technique that employs anisotropic pre-smoothing of an image prior to the estimation of the diffusivity function of outer diffusion processes. Considering this technique in the framework of directional diffusion we showed that better results can be obtained when compared to classical Gaussian regularization. The described technique can be employed to any diffusion equation that requires a pre-smoothing step.

ACKNOWLEDGEMENT

The authors wish to express their thanks to the society *I2S* from Pessac, France for providing the real images.

REFERENCES

[1] F. Catté, P.L. Lions, J. M. Morel and T. Coll, "Image selective smoothing and edge detection by non-linear diffusion", *SIAM J Numerical Analysis*,29, 1992,.pp. 182-193.

[2] W. J Connover., R. L Imam, "Rank transformations as a bridge between parametric and nonparametric statistics", *The American Statistician*, 35, (1981), pp. 124-129.

[3] B. Jahne, "Performance characteristics of low –level motion estimators in spatiotemporal images", *DAGM Workshop Performance Characteristics and Quality of Computer Vision Algorithms*, Braunschweig, Germany, 1997

[4] J. Koenderink, "The structure of images", *Biological Cybernetics,* Vol.50, 1984, pp. 363-370.

[5]P. Perona, J. Malik, " Scale-space and edge detection using anisotropic diffusion*" IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol.12, no.7, 1990, pp.629-639.

[6]R. Terebes, M. Borda , Y. Baozong, O. Lavialle, P. Baylou, "A new PDE based approach for image restoration and enhancement using robust diffusion directions and directional derivatives based diffusivities", *The 7'th International Conference on Signal Processing,* ICSP'04, Beijing-China, 2004, pp.707-717.

[7] R. Terebeş, O.Lavialle, M. Borda, P. Baylou, "Flow Coherence Diffusion. Linear and Nonlinear Case*"*, *Lecture Notes in Computer Science*, Springer-Verlag, vol. 3708/2005, pp. 316-322.

[8] J. Weickert , "Conservative image transforms with restoration and scale-space properties", *In: Proc. IEEE International Conference on Image Processing, ICIP'96,* Laussane, Switzerland, Vol.1, 1996, pp. 465-468.

[9] J. Weickert , "Coherence enhancing diffusion", *International Journal of Computer Vision ,* no.31, 1999, pp. 111-127.

[10] "Final report from the video quality experts group on the validation of objective models of video quality assessment", http://www.its.bldrdoc.gov/vqeg/, 2000.

[11] R. Whitaker, S.M. Pizer, "A multi-scale approach to nonuniform diffusion", *Graphical Model and Image Processing: Image Understanding,* vol.57, pp.111-120, 1993.

Buletinul Ştiinţific al Universităţii "Politehnica" din Timişoara

Seria ELECTRONICĂ şi TELECOMUNICAŢII
TRANSACTIONS on ELECTRONICS and COMMUNICATIONS

Tom 51(65), Fascicola 2, 2006

Image Processing Facilities for Echographic Measurements

Mircea-Florin Vaida[1], Valeriu Todica[2]

Abstract – The aim of the paper is to present a flexible dedicated application, HealthImag, able to integrate different medical facilities. Representative quantitative parameters are used for echographic measurements. Important facilities concerning the preprocessing, specific quantitative parameter measurements, data visualization are considered for medical investigations.
Keywords: HealthImag, echographic, fractal, texture.

I. INTRODUCTION

The HealthImag application was developed as a dedicated medical image processing application based on previous medical applications [1], based on VC++ 6.0 platform. The aim was to offer more flexibility and facilities to the HealthImag application and to use new technologies offered by Microsoft .NET Platform and C# language.
The processing of medical images can be divided in two categories [2]: pre-processing facilities for corrections and enhancements (because of contrast problems or imperfections in the image acquisition system) and image analysis facilities for determining some parameters, descriptors or statistical results. The application takes into account both categories presented above by implementing general functions like rotations, contrast adjustments, histogram adjustments, RGB adjustments, HSL adjustments, some kinds of spatial filters and implementing some analysis like automatic or semi-automatic objects detections (resulting some parameters), determining textural descriptors or implementing some specific analysis used for echographic images assisted diagnosis.
HealthImag is a modular application, composed by DLLs (Dinamic Link Libraries) modules. In this way, new functionality can be added very easily by creating new DLLs.

II. HEALTHIMAG MAIN FACILITIES

HealthImag offer some general facilities like:
- Opening an image from the disk or clipboard.
- Saving an image under different formats (jpeg, bmp, png, gif, tiff).
- Computing the image histogram (grayscale histogram or color histograms).
- Geometric operations: resize, zoom, flip, mirror, rotate, crop or clone an image.
- *Color* filters: grayscale filter, RGB adjustment, HSL adjustment, invert filter, filters for extracting components colors, contrast adjustment, gamma adjustment, image binarization.
- Spatial filters: Smooth filter, Sharpen filter, Mean filter, Blur filter or Custom filter (the spatial matrix can be defined by the user).
- Gradient edge detection filters: Sobel and Prewitt filters.
- Two source filters: adding two images, removing an image from another image, the mean of two images.

In Fig. 1 it can be viewed an example for using the image binarization operation. The threshold for binarization is automatically computed but the user can modify it.

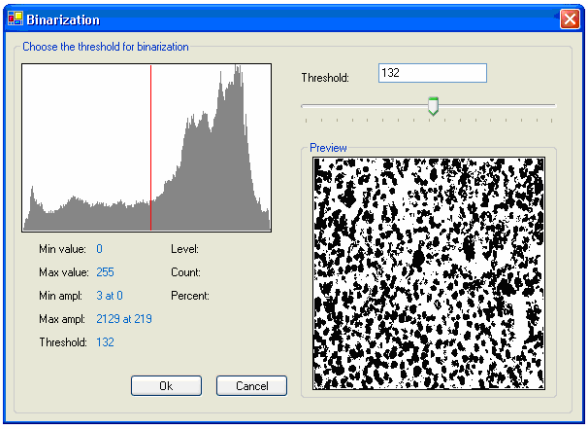


Fig. 1. Image binarization

III. HEALTHIMAG DEDICATED MEASUREMENTS

In this section are described some dedicated measurements offered by HealthImag like objects detection, fractal dimension computing, or Haralick descriptors computing.

[1] Technical University of Cluj Napoca, Baritiu Street no. 26, 400027, Cluj Napoca, Romania, e-mail Mircea.Vaida@com.utcluj.ro
[2] Technical University of Cluj Napoca, Baritiu Street no. 26, 400027, Cluj Napoca, Romania, e-mail todicav23@yahoo.com

A. *Objects detection*

Objects detection operation can be used for discovering objects in an image. Considering an input image this operation can discover objects according to a threshold. This threshold can be automatically computed or it can be selected by the user. Along with this threshold an Otsu algorithm is used for detecting objects in the image.
For every object discovered the following parameters are computed [4]:

- **Feret X Diameter** – defined by the X dimension of the rectangle containing the object.
- **Feret Y Diameter** – defined by the Y dimension of the rectangle containing the object.
- **Form Factor (ff)** – is defined by the following relation:

$$ff = \frac{P^2}{2\pi A} \quad (1)$$

,where P is the perimeter of the object and A is the area of the object.

- **Mass Center** coordinates:

$$xc = \frac{\sum_k x(k)}{P}, yc = \frac{\sum_k y(k)}{P} \quad (2)$$

,where (x(k), y(k)) k=0,..,P are points from the object contour and P is the number of points considered.

- **Regularity (reg)** – defined by the following relation:

$$reg = \frac{d_{med}}{\text{var}} \quad (3)$$

,where d_{med} is the mean distance from the object contour to the object mass center and *var* is the variance of these distances.

- **Ondulation factor (fond)** – obtained considering the relation:

$$fond = \frac{2P}{\pi(\max d^2 + \min d^2)} \quad (4)$$

,where P is the perimeter of the object and d is the distance from the object mass center to the object contour.

- **Excentricity (exc)** – obtained considering relation :

$$exc = \frac{\max d}{\min d} \quad (5)$$

,where *min d* is the minim distance from the object mass center to the object contour and *max d* is the maxim distance from the object mass center to the object contour.
In Fig. 2 it can be seen an image with some objects in it. By pointing with the mouse an object, it's becoming highlighted and the result can be seen in the object properties window. In this window are listed the number of objects, the current object number and all parameters described above.

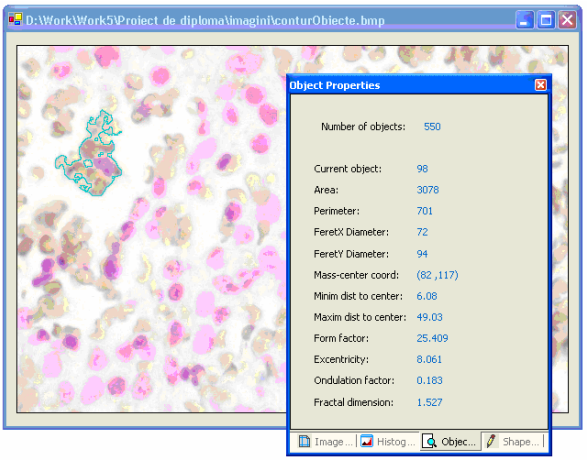


Fig. 2. Object detection results

Also, the fractal dimension of the object is computed. Fractal dimension is a statistical quantity used to measure the roughness of an object and can be used for comparing objects or for statistical measurements. Fractal dimension computing will be presented in the next section.
There is the possibility to save the results in a text file.

B. *Fractal dimension computing*

Fractal dimension is related to the objects detection operation presented in the previous section. For every object the fractal distance can be computed and listed in the Object Properties Window. Fractal dimension (also called fractal distance) is a property of a fractal object.
Fractals can be considered as self-similar patterns that repeat themselves. Smaller part of the fractal can be reflected in the entire fractal (this property is called self-similarity). Fractals can be very useful tools because the fractal geometry is the real geometry of the nature (like Mandelbrot said) . Most objects in nature aren't formed of squares or triangles, but of more complicated geometric figures. Objects like ferns or coastlines are shaped like fractals. Their

appearance led to the development of concepts like fractal dimensions [5].
Fractals proved to be very useful tools in many science domains. Fractal geometry achieved popularity through the efforts of B. Mandelbrot with his book *The Fractal Geometry of Nature.*
Self-similarity is a very important property of a fractal.
A shape S *is* called *exactly (linearly) self-similar* [6] *if* the whole *S* can be splitted into parts *Si* : *S* = *S*1 U *S*2 U. . .U *Sn* and the parts satisfy two restrictions. First restriction is that each part *Si* is a copy of the whole *S* scaled by a linear contraction factor *ri* , and the second restriction is that there is no intersections between parts in the sense of dimension.
The fractal dimension can be defined in many ways. We will consider here two important dimensions [6]:

1. Similarity Dimension
The definition of similarity dimension is related to the fact that the cube unit in *D*-dimensional Euclidean space is self-similar. This means that for any positive integer *b* the cube can be decomposed into *N =pow(b,D)* number of cubes, each cube scaled by the similarity ratio *r* =1/*b*, and overlapping (at most) along (*D* −1)- dimensional cubes.
The *similarity dimension dsim* is defined by the following relation*:*

$$d_{sim} = \frac{\log(N)}{\log\left(\frac{1}{d}\right)} \qquad (6)$$

2. Box Dimension
The similarity dimension can be used only for exactly self-similar shapes. For more general sets, the similarity dimension is replaced by the box dimension. For an set *A* in Euclidean space of dimension E, and for any *δ* >0, a *δ-cover* of *A* is a collection of sets of diameter *δ* whose union contains *A*. *Nδ*(*A*) is defined as the smallest number of sets in a *δ*-cover of *A*. Then the *box dimension dbox* of *A* can be computed using the relation:

$$d_{box} = \lim_{\delta \to \infty} \frac{\log(N_\delta(A))}{\log\left(\frac{1}{\delta}\right)} \qquad (7)$$

When the limit does not exist, the above relation can be modified by replacing the *lim* with *lim sup* and *lim inf*, defining in this way the *upper* and *lower box dimensions*:

$$\overline{d_{box}} = \lim_{\delta \to \infty} \sup \frac{\log(N_\delta(A))}{\log\left(\frac{1}{\delta}\right)} \qquad (8)$$

$$\underline{d_{box}} = \lim_{\delta \to \infty} \inf \frac{\log(N_\delta(A))}{\log\left(\frac{1}{\delta}\right)} \qquad (9)$$

Besides these two dimensions it also can be considered other types of dimensions like: mass dimension, Minkowski–Bouligand dimension or Hausdorff–Besicovitch dimension.
The algorithm used for computing the fractal distance in HealthImag is the Box-counting algorithm. The Box-counting dimension is much more used than the self-similarity dimension because the box-counting dimension can be used on images that are not exactly self-similar (most real-life images are not self-similar).
Considering [7] a well known fractal (Sierpinsky gasket), the image can be covered with a grid of square cells with cell size *r* (Fig. 3). For digital images the cell size can be considered as the number of pixels. In this example the number of grids containing a part of the structure is N = 51 out of 90 boxes.

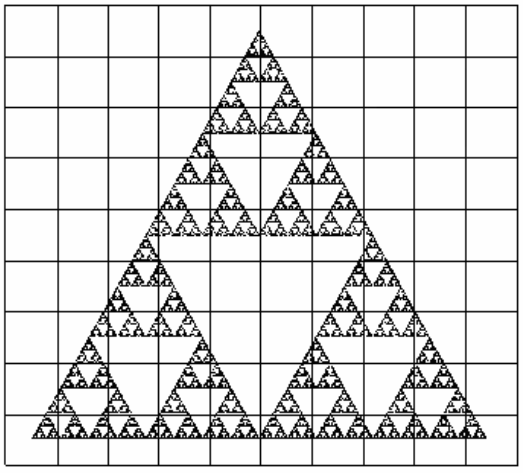


Fig. 3. Image overlaid with a grid of squares in order to compute the fractal distance [7]

The plot (called Richardson-Mandelbrot) containing different box sizes *r* can be viewed in the Fig. 4.

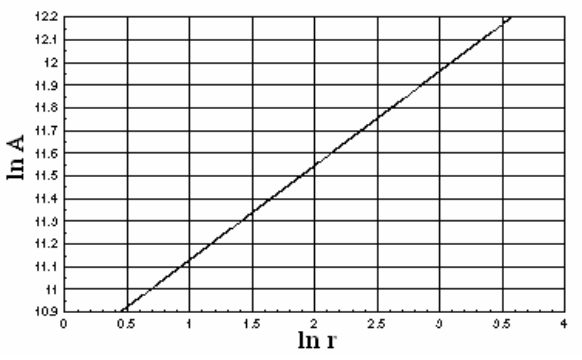


Fig. 4. Richardson-Mandelbrot plot for different box sizes in pixels [7]

For this example the slope of the resulting line is 0.414 and the fractal dimension is computed using the relation:

$$d_f = 2 - s = 2.0 - 0.414 = 1.586 \quad (10)$$

This is a simple example and the plot is a straight line but for real-life images the fractal dimension computing is not as simple as in this case. Note that the box counting procedure can be extended to three dimensions if cubes are used instead of squares.
HealthImag allows choosing the box dimensions for computing the fractal distance (Fig. 5).

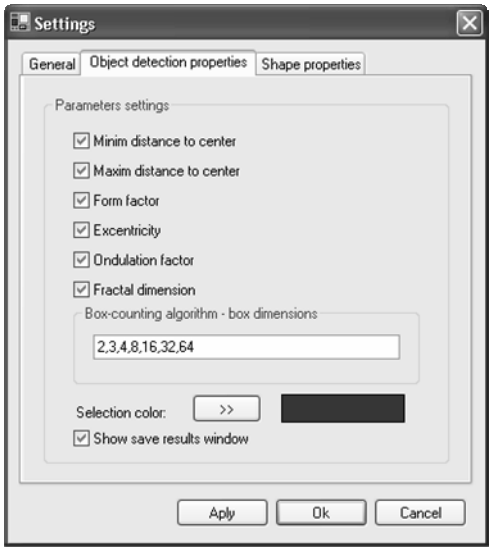


Fig. 5. The settings window in HealthImag

C. *Textural Analysis*

Generally, in some pictures, it can be easily to see the specific regions, having a similar grey or colour value, repeated over the image. This kind of pattern is called a texture.
While there isn't a universally accepted formal definition of a texture, a major characteristic is the *repetition of a pattern or patterns over a region*. The pattern may be repeated exactly or, usually, with small variations on the theme, possibly in function of the position. There are also random aspects related to the patterns repetition (that cannot be ignored) like: the size, shape, color and orientation of the elements of the patterns can vary over the region. It can be considered that the difference between two textures is contained in the degree of variation alone, or in the statistical distribution related to the patterns.

Haralick Descriptors

The Haralick descriptors are usually used for image classification. These descriptors contain information about the patterns that can be founded in textures.
Statistical characteristics can be extracted from textural images as nine Haralick descriptors [8]. These descriptors are computed using the *co-occurrence matrix*. The *co-occurrence matrix* is a matrix of relative frequencies Cij with which two neighborhood pixels separated by distance vector d occur on the image, one with gray level i and the other with gray level j [9]. The *co-occurrence matrix* is computed using a m × n window. In order to compute the Haralick descriptors we need some initials parameters:

$$\mu_i = \sum_{i=1}^{n}\sum_{j=1}^{m} iC_{ij} \quad \mu_j = \sum_{i=1}^{n}\sum_{j=1}^{m} jC_{ij} \quad (11)$$

$$C_i = \sum_{j=1}^{m} C_{ij} \quad (12)$$

$$\mathrm{var}(i) = \sum_{i=1}^{n}\sum_{j=1}^{m} (i-\mu_i)^2 C_{ij} \quad (13)$$

$$\mathrm{var}(j) = \sum_{i=1}^{n}\sum_{j=1}^{m} (j-\mu_j)^2 C_{ij} \quad (14)$$

Because the *co-occurrence matrix* can be computed considering four directions we have four sets of Haralick descriptors.
Relations for Haralick descriptors are given below, where C is the co-occurrence matrix and Cij is the element contained at the row i and at the column j in the *co-occurrence matrix* C [8].

- **Texture cluster tendency**

$$H_1 = \sum_{ij} \left(i - \mu_i + j - \mu_j\right)^2 C_{ij} \quad (15)$$

- **Texture entropy**

$$H_2 = -\sum_{ij} C_{ij} \log C_{ij} \quad (16)$$

- **Texture contrast**

$$H_3 = \sum_{ij} |i-j| \; C_{ij} \quad (17)$$

- **Texture correlation**

$$H_4 = \frac{\sum_{ij}(i-\mu_i)(j-\mu_j)C_{ij}}{\sqrt{\mathrm{var}(i)\,\mathrm{var}(j)}} \quad (18)$$

- **Texture homogeneity**

$$H_5 = \sum_{ij} \frac{C_{ij}}{1+|i-j|} \quad (19)$$

- **Texture inverse difference moment**

$$H_6 = \sum_{ij, i \neq j} \frac{C_{ij}}{|i-j|} \quad (20)$$

- **Maximum texture probability**

$$H_7 = \max_{ij} C_{ij} \quad (21)$$

- **Texture probability of run length of 2**

$$H_8 = \sum_{i} \frac{(C_i - C_{ii})^2 C_{ii}}{C_i^{\,2}} \quad (22)$$

- **Uniformity of texture energy**

$$H_9 = \sum_{ij} C_{ij}^{\,2} \quad (23)$$

In the Fig. 6 it can be seen the result for computing Haralick descriptors. There is the possibility to choose the direction used for the *co-occurrence matrix* (horizontal direction, vertical direction, etc). Also, the results can be saved as text files on the disk.

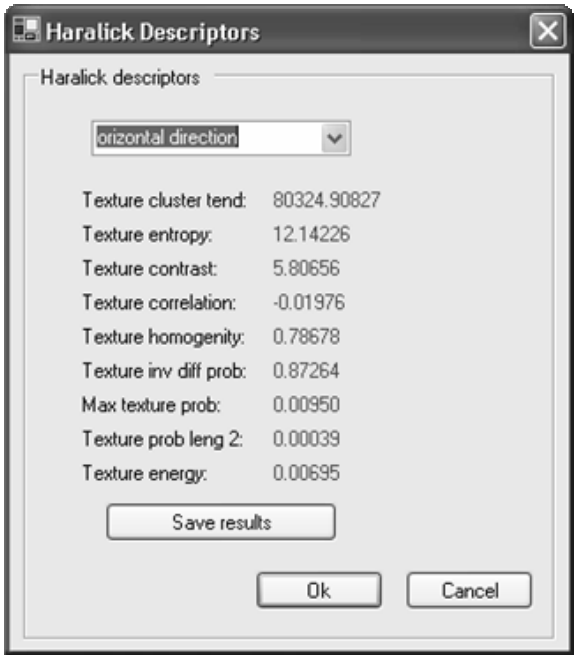


Fig. 6. Haralick descriptors values for a given image

IV. ECHOGRAPHIC MEASUREMENTS

These kinds of measurements can be used for echographic images analysis. If a simple inspection of an echographic image is not very relevant, by using some modern methods in image analysis the precision of the diagnostic can be drastically improved.

In the analysis of an echographic image the following steps are required:
- setting the initial state of the image acquisition system.
- obtaining the image using the image acquisition camera.
- applying the considered echographic measurement.
- showing the results like numerical data or using some graphical plots.

The results for different images can be compared (of course, by using the same settings for the image acquisition camera).
By using enough images a domain of values can be defined and used for comparing affections or severity degrees for the same affection.
HealthImag offer the possibility to make an image correction by extracting from an image another image considered a reference image. In this way, existing defects in the acquisition system can be eliminated.

A. *Pixel neighborhood analysis*

This analysis is used for obtaining information from the pixels neighborhood. Using the mouse cursor, the user can see the information related to the greyscale levels for pixels contained in the proximity of the pixel pointed. For this kind of analysis a 3 x 3 or 5 x 5 matrix can be considered. Along with the greyscale values also the mean and the medial values are computed.
For a given image, the results are showed in the pixel neighborhood information window (Fig. 7).

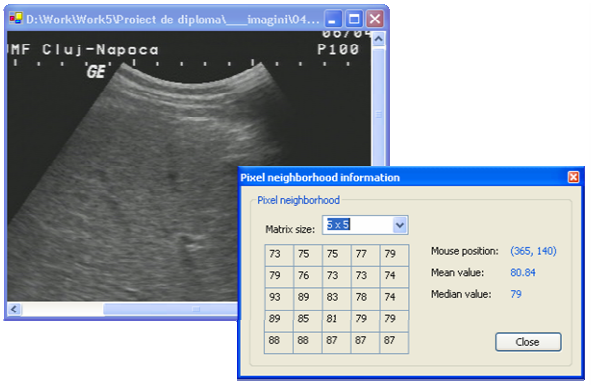


Fig. 7. Pixel neighborhood analysis

The same analysis can be used automatically (Fig. 8).

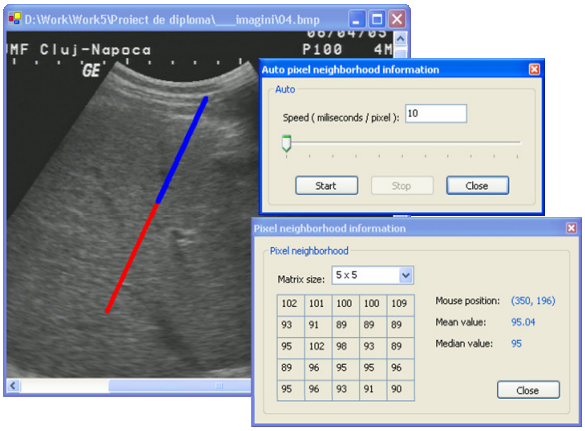


Fig. 8. Automatic pixel neighborhood analysis

The user select a straight line and using a timer (the timer frequency can be modified by the user) the pixels from the line are analyzed in the same way as the previous analysis.

For every point contained in the line greyscale levels, mean values and median values are listed. The process of scanning the considered line can be stopped or resumed by the user.

At the end, the results are plotted (Fig. 9). There are 3 plots: grayscale plot, mean plot and median plot.

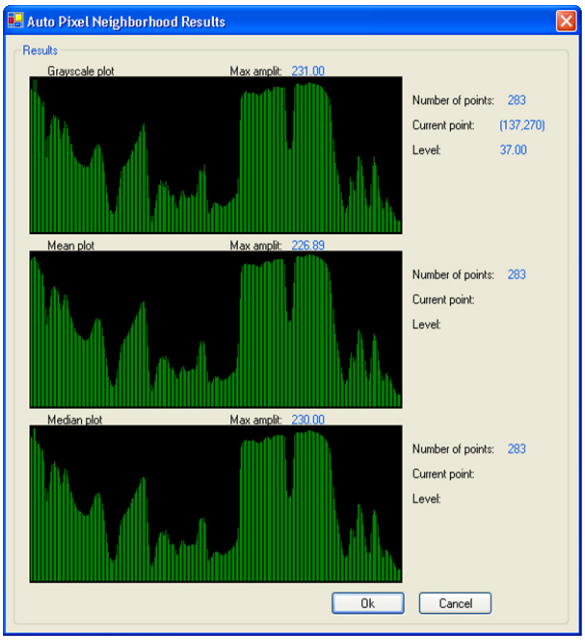


Fig. 9. Plotting the results of the *automatic pixel neighborhood* analysis

V. CONCLUSIONS

The HealthImag application is a powerful instrument that can be used for image corrections and enhancements or for dedicated medical analysis. HealthImag has an easy-to-use GUI (graphical user interface) and can be a good tool for a medical staff. Implementations for some analysis in an automatic way can significantly reduce the time for analyzing medical images.

Implemented using some new technologies like .NET, HealthImag offer some modern possibilities to process images, to analyze images or to show results. Also, new functionality can be added very easily by implementing new DLLs modules containing the desired functionality.

The application is used in a radiology department at the University of Medicine "Iuliu Hatieganu" Cluj-Napoca. Relevant medical results are expected considering an assisted diagnosis process. Statistics facilities will be implemented to generate different medical reports.

REFERENCES

[1] Mircea-Florin Vaida, Essential Quantitative and Statistics Parameters in a Dedicated Assisted Medical Image Processing Application, ITI 2004, 7-10 June 2004, Cavtat, Croatia, pp. 259-264

[2] Gonzalez R. C. and Woods R. E., Digital Image Processing, Addison-Wesley Publishing, 1992

[3] William K. Pratt, Digital image processing, John Wiley & Sons, 2001

[4] Mircea Vaida, A. Suciu, T. Moldovan, Dedicated Assisted Diagnosis Using An Image Analysis Application, rev. Acta Tehnica Napocensis Electronics and Telecommunications, Cluj-Napoca, Vol. 40, No.2, 2000, pp.60-67

[5] Oliver Dick, FractalVision: Put Fractals to Work for You, Sam Publishing, 1992

[6] Mandelbrot B., Frame M., Fractals, Encyclopedia of Physical Science and Technology Volume 6, 2002

[7] http://www.wzw.tum.de/ane/algorithms/ subsection3_4_1.html [12.04.2006]

[8] Martin Svec, Analysis of Sonographic Images of Thyroid Gland Based on Texture Classification, Czech Technical University, 2001

[9] Eizan Miyamoto and Thomas Merryman, Fast calculation of Haralick texture features, Carnegie Mellon University

Buletinul Ştiinţific al Universităţii "Politehnica" din Timişoara

Seria ELECTRONICĂ şi TELECOMUNICAŢII
TRANSACTIONS on ELECTRONICS and COMMUNICATIONS

Tom 51(65), Fascicola 2, 2006

Lattice MMSE Single User Receiver for Asynchronous DS-CDMA Systems

Constantin Paleologu, Călin Vlădeanu, Andrei A. Enescu[1]

Abstract – This paper considers a lattice Minimum Mean-Squared Error (MMSE) single user adaptive receiver for the asynchronous Direct Sequence – Code Division Multiple Access (DS-CDMA) system. It is based on the Gradient Adaptive Lattice (GAL) algorithm. Since the lattice predictor orthogonalizes the input signals this algorithm achieves a faster convergence rate than the transversal counterpart, the Least Mean Square (LMS) adaptive algorithm, paying with an increased computational complexity. Superior performances are obtained by adapting the tap weights several times during each bit interval.
Keywords: DS-CDMA, adaptive filter, GAL, LMS.

I. INTRODUCTION

There are a lot of mobile communications systems that employ the CDMA (Code Division Multiple Access) technique, where the users transmit simultaneously within the same bandwidth by means of different code sequences. CDMA technique has been found to be attractive because of such characteristics as potential capacity increases over competing multiple access methods, anti-multipath capabilities, soft capacity, narrow-bandwidth anti-jamming, and soft handoff.

In Direct Sequence CDMA (DS-CDMA) systems [1], the conventional matched filter receiver distinguishes each user's signal by correlating the received multi-user signal with the corresponding signature waveform. The data symbol decision for each user is affected by Multiple-Access Interference (MAI) from other users and by channel distortions. Hence, the conventional matched filter receiver performances are limited by its original purpose. It was designed to be optimum only for a single user channel where no MAI is present, and to be optimum for a perfect power control, so it suffers from the near-far problem.

Multi-user receivers have been proposed to overcome the inherent limitations of the conventional matched filter receiver. The use of these multi-user receivers has shown to improve system's performance, and enhance its capacity relative to the conventional matched filter detection. Unfortunately, most of these multi-user detectors require complete system information on all users [1].

Implementations of adaptive Minimum Mean-Squared Error (MMSE) receivers in DS-CDMA systems have been analyzed in [2] and [3]. The principle of the adaptive MMSE receivers consists of a single user detector that works only with the bit sequence of that user. In this case the detection process is done in a bit by bit manner, and the final decision is taken for a single bit interval from the received signal. The complexity of an adaptive MMSE receiver is slightly higher than that of a conventional receiver, but with superior performance [2]-[5]. Besides its facile implementation the adaptive MMSE receiver has the advantage that it needs no supplementary information during the detection process, as compared to the conventional matched-filter receiver.

The adaptive algorithms used for MMSE receivers can be divided into two major categories [6], [7]. The first one contains the algorithms based on mean square error minimization, whose representative member is the Least Mean Square (LMS) algorithm. The second category of algorithms uses an optimization procedure in the least squares (LS) sense, and its representative is the Recursive Least Squares (RLS) algorithm. The transversal LMS algorithm with its simple implementation suffers from slow convergence, which implies long training overhead with low system throughput. On the other hand, LS algorithms such as RLS offer faster convergence rate and tracking capability than the LMS algorithm. This performance improvement of the RLS over the LMS is achieved at the expense of the large computational complexity.

Lattice structures have also been considered for this type of applications [8], [9]. Since the lattice predictor orthogonalizes the input signals, the gradient adaptation algorithms using this structure are less dependent on the eigenvalue spread of the input signal and may converge faster than their transversal counterparts. The computational complexity of the Gradient Adaptive Lattice (GAL) algorithm [6] is between transversal LMS and RLS algorithms. In addition, several simulation examples and also numerical comparison of the analytical results have shown that adaptive lattice filters have better numerical properties than their transversal

[1] University "Politehnica" of Bucharest, Faculty of Electronics, Telecommunications and Information Technology, 1-3, Iuliu Maniu Bvd., Bucharest, Romania, e-mail: pale@comm.pub.ro, calin@comm.pub.ro, aenescu@comm.pub.ro

counterparts [10], [11]. Moreover, stage-to-stage modularity of the lattice structure has benefits for efficient hardware implementations.

In this paper we compare the performances of a lattice MMSE single user adaptive receiver based on GAL algorithm for the asynchronous DS-CDMA system with a transversal counterpart based on LMS algorithm.

The paper is organized as follows. In section II we briefly describe the asynchronous DS-CDMA system model, both the transmitter and adaptive receiver parts of the scheme. Section III is focused on the adaptive receiver part of the scheme, revealing in this context the GAL algorithm. The experimental results are presented in section IV. Finally, section V concludes this work.

II. DS-CDMA SYSTEM MODEL

In the transmitter part of the DS-CDMA system, each user data symbol is modulated using a unique signature waveform $a_i(t)$, with a normalized energy over a data bit interval T, $\int_0^T \|a_i(t)\|^2 dt = 1$, given by [1]:

$$a_i(t) = \sum_{j=1}^{N} a_i(j) p_c(t - jT_c), \; i = \overline{1,\; K} \qquad (1)$$

where the $a_i(j)$ represents the jth chip of the ith user's code sequence and are assumed to be elements of {-1, +1}, and $p_c(t)$ is the chip pulse waveform defined over the interval [0; T_c) with T_c as the chip duration which is related to the bit duration through the processing gain N, with T_c=T/N. K denotes the number of users in the system. In the following analysis we consider Binary Phase Shift Keying (BPSK) modulation for signal transmission.

Then, the ith user transmitted signal is given by

$$s_i(t) = \sqrt{2P_i}\, b_i(t) a_i(t) \cos(\omega_0 t + \theta_i), \; i = \overline{1,\; K} \quad (2)$$

where P_i is the ith user bit power,

$$b_i(t) = \sum_{m=1}^{N_b} b_i(m) p(t - mT), \;\; b_i(m) \in \{-1,\; +1\} \quad (3)$$

is the binary data sequence for ith user, N_b is the number of received data bits, $\omega_0 = 2\pi f_0$ and θ_i represent the common carrier pulsation and phase, respectively.

A block diagram of the lattice receiver structure is shown in Fig. 1. After converting the received signal to its baseband form using a down converter, the received signal is given by:

$$r(t) = \left[\sum_{i=1}^{K} s_i(t - \tau_i) + n(t)\right] \cos(\omega_0 t) = \\ = \sqrt{\frac{P_i}{2}} \sum_{i=1}^{K} b_i(t - \tau_i) a_i(t - \tau_i) \cos(\theta_i) + n(t) \cos(\omega_0 t) \qquad (4)$$

where $n(t)$ is the two-sided PSD (Power Spectral Density) $N_0/2$ additive white Gaussian noise (AWGN). The asynchronous DS-CDMA system consists of random initial phases of the carrier $0 \le \theta_i < 2\pi$ and random propagation delays $0 \le \tau_i < T$ for all the users $i = \overline{1,\; K}$. There is no loss of generality to assume that $\theta_k = 0$ and $\tau_k = 0$ for the desired user k, and to consider only $0 \le \tau_i < T$ and $0 \le \theta_i < 2\pi$ for any $i \ne k$ [2].

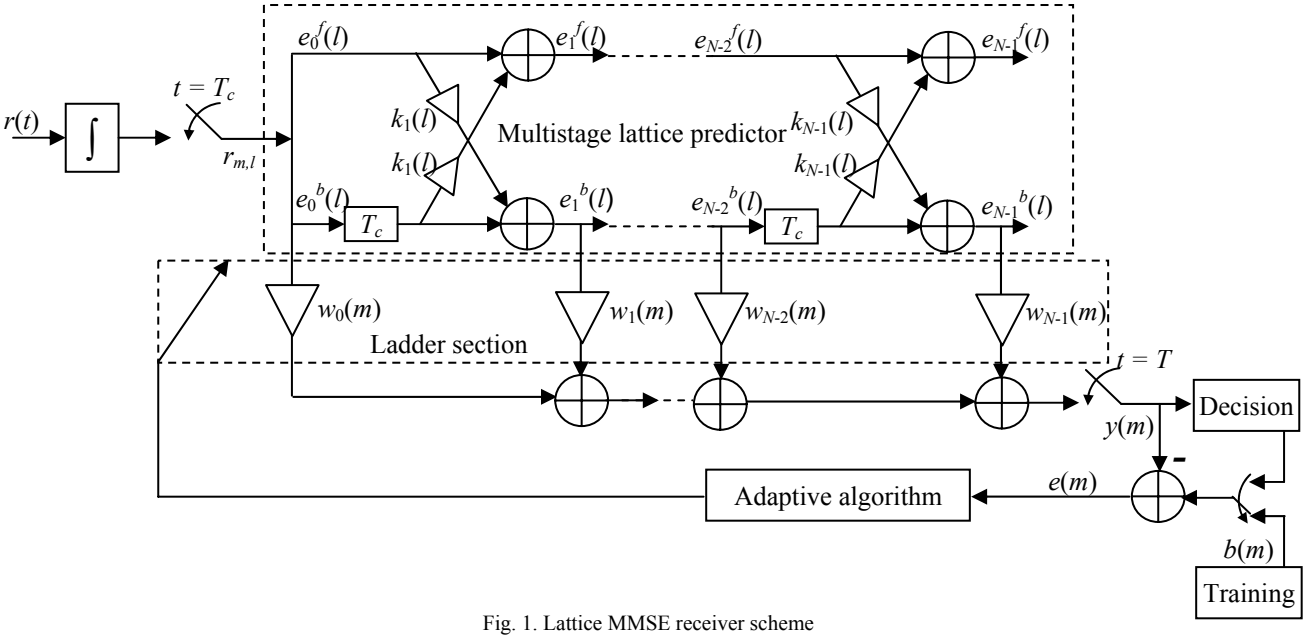


Fig. 1. Lattice MMSE receiver scheme

Assuming perfect chip timing at the receiver, the received signal in (4) is passed through a chip-matched filter followed by sampling at the end of each chip interval to give for the mth data bit interval:

$$r_{m,\,l} = \int_{mT+lT_c}^{mT+(l+1)T_c} r(t)\,p(t-lT_c)\,dt, \quad l = 0,\ 1,\ \ldots,\ N-1 \tag{5}$$

where $p(t)$ is the chip pulse shape, which is taken to be a rectangular pulse with amplitude $1/\sqrt{N}$. Using (5) and taking the kth user as the desired one, the output of the chip matched filter after sampling for the mth data bit is given by:

$$\begin{aligned} r_{m,l} &= \sqrt{\frac{P_k}{2N}}\,T_c b_k(m) a_k(l) + \\ &+ \frac{1}{\sqrt{2N}} \sum_{\substack{i=1 \\ i\neq k}}^{K} \sqrt{P_i}\cos\theta_i b_i(m) I_{i,k}(m,l) + n(m,l) \end{aligned} \tag{6}$$

where

$$I_{i,k}(m,l) = \begin{cases} b_i(m-1)[\varepsilon_i a_i(N-1-N_i+l) + \\ \quad +(T_c-\varepsilon_i)\,a_i(N-N_i+l)], & 0 \le l \le N_i-1 \\ b_i(m-1)\,\varepsilon_i a_i(N-1) + \\ \quad +b_i(m)(T_c-\varepsilon_i)\,a_i(0), & l = N_i \\ b_i(m)[\varepsilon_i a_i(l-N_i-1) + \\ \quad +(T_c-\varepsilon_i)\,a_i(l-N_i)], & N_i+1 \le l \le N-1 \end{cases} \tag{7}$$

with

$$\tau_i = N_i T_c + \varepsilon_i,\ \ 0 \le N_i \le N-1,\ \ 0 < \varepsilon_i < T_c \tag{8}$$

Let us consider the following vectors:

$$\begin{aligned} \mathbf{r}(m) &= [r_{m,\,0} \quad r_{m,\,1} \quad \ldots. \quad r_{m,\,N-1}]^T \\ \mathbf{a}_k &= [a_k(0), \quad a_k(1) \quad \ldots. \quad a_k(N-1)] \end{aligned} \tag{9}$$

with $r_{m,l}$ given by (6), the vector $\mathbf{a}_k$ represents the binary code sequence for the kth user, and the components of the noise $n(m,l)$ vector in (6) consists of independent zero-mean Gaussian random variables with variance $N_0/(2N)$.

In the training mode, the receiver attempts to cancel the MAI and adapts its coefficients using a short training sequence employing an adaptive algorithm. After training is acquired, the receiver switches to the decision-directed mode and continues to adapt and track channel variations [2]-[4].

III. GAL ALGORITHM FOR MMSE RECEIVER

During the training mode, the filter tap weights are adjusted every transmitted bit interval. The receiver forms an error signal proportional to the difference between the filter output and the known reference signal. This error signal is then used to adjust the filter tap weights using the adaptive algorithm. This process is repeated for every received bit until steady-state convergence is reached.

The $(N-1)$-th-order lattice predictor is specified by the recursive equations

$$e_p^f(l) = e_{p-1}^f(l) + k_p^*(l)\,e_{p-1}^b(l-1) \tag{10}$$

$$e_p^b(l) = e_{p-1}^b(l-1) + k_p(l)\,e_{p-1}^f(l) \tag{11}$$

where $p = 1, 2, \ldots, N-1$. We denoted by $e_p^f(l)$ the forward prediction error, by $e_p^b(l)$ the backward prediction error, and by $k_p(l)$ the reflection coefficient at the pth stage and chip-time l. The zeroth-order prediction errors are given by

$$e_0^f(l) = e_0^b(l) = r_{m,l} \tag{12}$$

The cost function used for the estimation of $k_p(l)$ is

$$J_p = \frac{1}{2} E\left\{ \left|e_p^f(l)\right|^2 + \left|e_p^b(l)\right|^2 \right\} \tag{13}$$

where E is the statistical expectation operator [6]. Substituting equations (10) and (11) into equation (13), differentiating the cost function J_p with respect to the complex-valued reflection coefficient $k_p(l)$ and then putting the gradient equal to zero, the optimum value of the reflection coefficient for which the cost function J_p is minimum results

$$k_p^{opt} = -\frac{2E\left\{e_{p-1}^b(l-1)\,e_{p-1}^{f*}(l)\right\}}{E\left\{\left|e_{p-1}^f(l)\right|^2 + \left|e_{p-1}^b(l-1)\right|^2\right\}} \tag{14}$$

Assuming that the input signal is ergodic the expectations could be substituted by time averages, resulting the Burg estimate for the reflection coefficient k_p^{opt} for stage p in the lattice predictor:

$$k_p(l) = -\frac{2\sum_{q=1}^{l} e_{p-1}^b(q-1)\,e_{p-1}^{f*}(q)}{\sum_{q=1}^{l}\left[\left|e_{p-1}^f(q)\right|^2 + \left|e_{p-1}^b(q-1)\right|^2\right]} \tag{15}$$

Let us denoted by $W_{p-1}(l)$ the total energy of both the forward and backward prediction errors at the input of the pth lattice stage, measured up to and including time l, and expressed it as:

$$W_{p-1}(l) = \sum_{q=1}^{l}\left[\left|e_{p-1}^{f}(q)\right|^2 + \left|e_{p-1}^{b}(q-1)\right|^2\right] = \\ = W_{p-1}(l-1) + \left|e_{p-1}^{f}(l)\right|^2 + \left|e_{p-1}^{b}(l-1)\right|^2 \quad (16)$$

It can be demonstrated [6] that the GAL algorithm updates the reflection coefficients using

$$k_p(l) = k_p(l-1) - \frac{\mu}{W_{p-1}(l)} \cdot \\ \cdot\left[e_p^{f*}(l)e_{p-1}^{b}(l-1) + e_p^{b*}(l)e_{p-1}^{f}(l)\right] \quad (17)$$

where μ is a constant controlling the convergence of the algorithm. The use of the time-varying step-size parameter $\mu / W_{p-1}(l)$ in the update equation (8) for the reflection coefficient $k_p(l)$ introduces a form of normalization similar to that in the Normalized LMS (NLMS) algorithm [6], [7]. For a well-behaved convergence of the GAL algorithm, it is recommended that we set $\mu < 0.1$ [6].

In practice, a minor modification is made to the energy estimator of equation (16) by writing it in the form of a single-pole average of squared data:

$$W_{p-1}(l) = \beta W_{p-1}(l-1) + (1-\beta) \cdot \\ \cdot\left[\left|e_{p-1}^{f}(l)\right|^2 + \left|e_{p-1}^{b}(l-1)\right|^2\right] \quad (18)$$

where $0 < \beta < 1$. The introduction of parameter β in equation (18) provides the GAL algorithm with a finite memory, which helps it to deal better with statistical variations when operating in a nonstationary environment. As reported in [10] and demonstrated in [12], the way to choose β is

$$\beta = 1 - \mu \quad (19)$$

As depicted in Fig. 1, the basic structure for the estimation of the user desired response $b(m)$, is based on a multistage lattice predictor that performs both forward and backward predictions, and an adaptive ladder section. We have an input column vector of the backward prediction errors

$$\mathbf{e}_N^b(m) = [e_0^b(m), e_1^b(m), \ldots, e_{N-1}^b(m)]^T \quad (20)$$

and a corresponding column vector $\mathbf{w}(m)$ representing the N coefficient vector the adaptive filter weights:

$$\mathbf{w}(m) = [w_0(m), w_1(m), ..., w_{N-1}(m)]^T \quad (21)$$

where the symbol m denotes the discrete time index of the data bit sequence. The output signal $y(m)$ will be an estimate of $b(m)$. For the estimation of $\mathbf{w}(m)$, we may use a stochastic-gradient approach. The discrete output signal $y(m)$ is given by:

$$y(m) = \sum_{l=0}^{N-1} w_l(m) e_l^b(m) \quad (22)$$

Using vector notation, (22) can be written as:

$$y(m) = \mathbf{w}^T(m)\mathbf{e}_N^b(m) \quad (23)$$

The receiver forms an error signal $e(m)$,

$$e(m) = b(m) - y(m) \quad (24)$$

and a new filter tap weight vector is estimated according to:

$$\mathbf{w}(m+1) = \mathbf{w}(m) + \tilde{\mu} e(m)\mathbf{e}_N^b(m) \quad (25)$$

The parameter $\tilde{\mu}$ in (25) is the ladder structure adaptation step size chosen to optimize both the convergence rate and the mean squared error.

Summarized, we will use equation equations (10), (11), (18) and (17) for the lattice predictor part of the scheme and equations (23)-(25) for the ladder section. Comparative with the its transversal counterpart based on LMS algorithm, the lattice MMSE receiver implies an increased computational complexity due to the multistage lattice predictor. The classical transversal receiver is based only on the equations (23)-(25), where we have to replace $\mathbf{e}_N^b(m)$ by $\mathbf{r}$(m) (see (9)). Nevertheless, due to the fact that the lattice predictor orthogonalizes the input signals, a faster convergence rate is expected.

A solution to increase the overall performances is to adjust the filter tap weights iteratively several times every transmitted bit interval [4], [5]. The error obtained during the Gth iteration of the mth data bit is used by the algorithm in the first iteration of the (m+1)th data bit. When a new data bit is received, the filter tap weights are adapted in the same manner as presented, with the initial condition given by

$$\mathbf{w}^{(0)}(m+1) = \mathbf{w}^{(G)}(m) \quad (26)$$

where $\mathbf{w}^{(0)}(m+1)$ and $\mathbf{w}^{(G)}(m)$ represent the initial tap weights at the (m+1)th received bit, and the final tap weights at time index m, respectively. It is obvious that this process will increase the computational complexity, as well as the speed requirements for the adaptive filter.

IV. SIMULATION RESULTS

The asynchronous DS-CDMA system using the lattice MMSE receiver based on GAL algorithm was tested using MATLAB programming environment. It was compared with its transversal counterpart based on LMS algorithm. A binary-phase shift keying transmission in a training mode scenario was considered. The simulation parameters were fixed as follows: the processing gain $N = 32$, the number of users $K = 64$ and the signal-to-noise ratio (SNR) is 15 dB. The mean-squared error (MSE) was estimated by averaging over 100 independent trials. The convergence results are presented in Fig. 2.

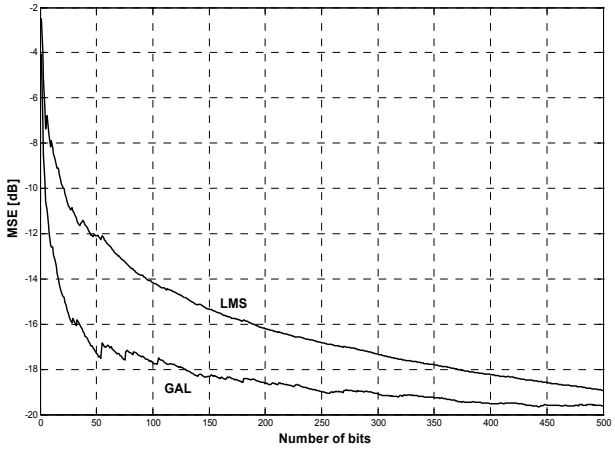


Fig. 2. Convergence of the adaptive receivers

The superior convergence rate achieved by the GAL algorithm as compared to the conventional LMS algorithm can be observed. This can be explained by the fact that the lattice predictor orthogonalizes the input signals. Hence, the gradient adaptation algorithm using this structure is less dependent on the eigenvalue spread of the input signal. In Fig. 3 the mean autocorrelation function is depicted for both the output signal from the chip-matched filter receiver $\mathbf{r}(m)$ (used as the direct input for the transversal LMS receiver) and the sequence of backward prediction errors $\mathbf{e}_N^b(m)$ (the input for the ladder section of the GAL receiver).

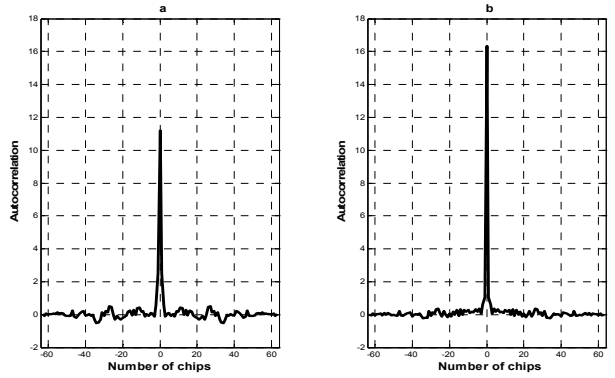


Fig. 3. Autocorrelation functions for:
(a) $\mathbf{r}(m)$ - LMS input; (b) $\mathbf{e}^b_N(m)$ - GAL ladder section input

It is obvious from Fig. 3 that the the input of the GAL ladder section has a higher variance as compared to the LMS input sequence.

As it was mentioned in the end of section III, superior performances are obtained by adapting the tap weights several times during each bit interval. A second set of simulations is dedicated to the proof of this aspect. The adaptive algorithms are iterated for 4 times each data bit. The results are presented in Fig. 4 and Fig. 5.

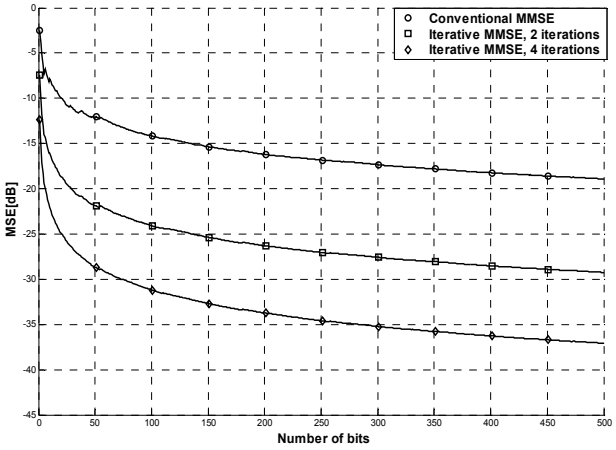


Fig. 4. Convergence of the iterative LMS receiver

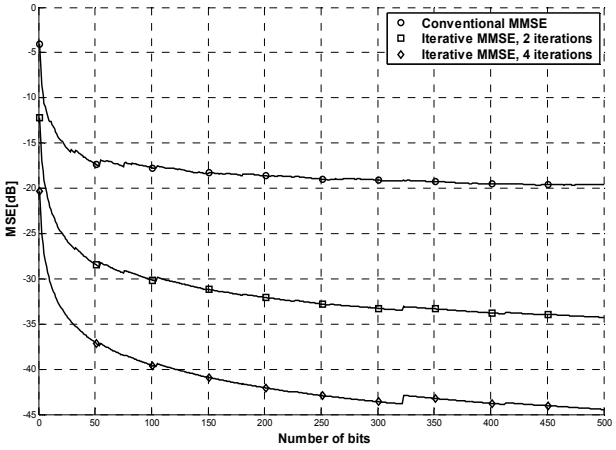


Fig. 5. Convergence of the iterative GAL receiver

In both cases the MSE is decreased every new iteration. Comparing these last two figures it can be also noticed that the GAL algorithm outperforms the LMS algorithm.

V. REMARKS

The lattice MMSE receiver considered in this paper improves the asynchronous DS-CDMA system performances over the classical transversal receiver. The lattice predictor orthogonalizes the input signals, so that the GAL algorithm using this structure is less dependent on the eigenvalue spread of the input signal and may converge faster than their transversal counterpart, the LMS algorithm. As a practical consequence, the lattice receiver will require a shorter

training sequence. Superior performances are obtained by adapting the tap weights several times during each bit interval, in order to decrease MSE every new iteration. This decrease offers a faster training mode for the receiver, thus improving the useful bit rate.

As a consequence, the receiver designing procedure may consider one of these two enhancements: to shorten the training sequence for maintaining the same MAI in the system or to strongly reduce the MAI by keeping the same length of the training sequence.

Nevertheless, the systems performances are evaluated by means of MSE. A true performance parameter for the DS-CDMA system is the mean Bit Error Rate (BER). An analytical estimation of BER for this MMSE iterative receiver will be considered in perspective.

REFERENCES

[1] S. Glisic, B. Vucetic, *CDMA Systems for Wireless Communications*, Artech House, 1997.

[2] S. Miller, "An Adaptive Direct-Sequence Code-Division Multiple-Access Receiver for Multi-user Interference Rejection", *IEEE Transactions On Communications*, vol. 43, pp. 1746-1755, Apr. 1995.

[3] P. Rapajic and B. Vucetic, "Adaptive Receiver Structures for Asynchronous CDMA Systems", *IEEE Journal on Selected Areas in Communications*, vol. 12, no. 4, pp. 685-697, May 1994.

[4] W. Hamouda, P. McLane, "A Fast Adaptive Algorithm for MMSE Receivers in DS-CDMA Systems", *IEEE Sign Proc. Letters*, vol. 11, no. 2, pp. 86-89, Feb. 2004.

[5] C. Vlădeanu, C. Paleologu, "MMSE single user iterative receiver for asynchronous DS-CDMA systems", *Proc. of IEEE Int. Conf. Communications 2006*, Bucharest, 2006, pp. 227-230.

[6] S. Haykin, *Adaptive Filter Theory*, fourth edition. Prentice Hall Upper Saddle River, NJ, 2002.

[7] S. Ciochină, C. Negrescu, *Sisteme adaptive*, Ed. Tehnică, Bucharest, 1999.

[8] J. Wang, V. Prahatheesan, "Adaptive Lattice Filters for CDMA Overlay", *IEEE Trans. on Communications*, vol. 48, no. 5, May 2000, pp. 820-828.

[9] F. Takawira, "Adaptive Lattice Filters for Narrowband Interference Rejection in DS Spread Spectrum Systems", *Proc. IEEE South African Symp. Communications and Signal Processing*, 1994.

[10] V. J. Mathews, Z. Xie, "Fixed-Point Error Analysis of Stochastic Gradient Adaptive Lattice Filters", *IEEE Trans. on Acoustics, Speech and Signal Processing*, vol. 38, no. 1, January 1990, pp. 70-79.

[11] R. C. North, J. R. Zeidler, W. H. Ku, T. R. Albert, "A Floating-Point Arithmetic Error Analysis of Direct and Indirect Coefficient Updating Techniques for Adaptive Lattice Filters", *IEEE Trans. on Signal Processing*, vol. 41, no. 5, May 1993, pp. 1809-1823.

[12] C. Paleologu, S. Ciochină, A.A. Enescu, "Modified GAL Algorithm Suitable for DSP Implementation", *Proc. of IEEE Int. Symp. ETc 2002*, vol. 1, Timişoara, 2002, pp. 2-7.

Buletinul Ştiinţific al Universităţii "Politehnica" din Timişoara

Seria ELECTRONICĂ şi TELECOMUNICAŢII
TRANSACTIONS on ELECTRONICS and COMMUNICATIONS

Tom 51(65), Fascicola 2, 2006

Mobility Mechanisms for Mobile/Wireless all-IP Networks

Emanuel Puschita[1], Tudor Palade[2], Bogdan Pop[3], Sandu Florin[4]

Abstract – For the next generation technologies, mobility is more than a necessity, it's a requirement. Actual developed architectures dedicate a special attention to this aspect. Considering that, this paper analyzes the aspect reflecting the network capability, named mobility. We test and verify the mobile nodes availability to roam in different scenarios, for computer networks and cellular networks. All simulations scenarios were implemented using ns-2 network simulator and for the tested architectures we offer the end-to-end delay during the handover process. The results demonstrate how average end-to-end delay contributes to the QoS global evaluation for a wireless scenario.
Keywords: mobility, QoS support, end-to-end delay, ns

I. INTRODUCTION

The paper is structured on five chapters. The first chapter introduces the paper. There are a number of factors and components that affect the performances of multimedia application. Grouping all these elements, we consider that quality of services problem has two major perspectives: **network perspective** and **application/user perspective**. From the network perspective, QoS refers to the service quality or service level that the network offers to applications or users in terms of network QoS parameters, including: latency or delay of packets traveling across the network, reliability of packet transmission, and throughput.
From the user/application perspective QoS generally refers to the application quality as perceived by the user. That is, the presentation quality of the video, the responsiveness of interactive voice, and the sound quality of streaming audio.
We group applications and users are in the same category because of their common way they perceive quality [1]. Considering that, we evaluate mobility as a network parameter and analyze its contribution on delay of delivering packets for different wireless technologies. The results presented on this paper complete our work on QoS support for broadband wireless network and consolidate the layered QoS approaches separate QoS aspects on each layer [2].
We consider that parameters on each layer essentially contribute to the global QoS evaluation [3]. In our simulations we use ns-2.26 network simulator. A complete information regarding installation under different operating systems and specific parameter configuration for wireless scenarios is presented by the authors [4]. We offer description and graphical representations in all simulated cases.
The second chapter gives a perspective of mobility concept for different networks, illustrating the types of roaming and the correlations with network layers.
The third chapter presents the roaming process for **computer networks**, as layer 2 and layer 3 handover.
The four chapter presents the roaming process for **cellular networks**.
Chapter five will summarize and conclude our work on intra-system mobility evaluation.

II. GENERAL CONCEPTS ON MOBILITY FOR WIRELESS SYSTEMS

An integral concept for wireless systems is roaming. It is important to understand what roaming is, how and when it occurs, what types of roaming there are, and how these types differ. Mobility is the quality of being capable of movement or moving readily from place to place.
Defining or characterizing the behavior of roaming mobile nodes involves two forms: **seamless roaming** and **nomadic roaming**.
Seamless roaming is best analogized to a cellular phone call. There is no noticeable period of network unavailability because of roaming. This type of roaming is deemed seamless because the network application requires constant network connectivity during the roaming process. Seamless roaming is characteristic for cellular communications systems and assumes that mobile node roams between cellular base stations and maintains a permanent connection with the network.
Nomadic roaming is different from seamless roaming. Nomadic roaming is best described as the use of a wireless local area network environment.
The mobile node has network connectivity while seated at his destination and maintains connectivity to a single access point. When the user decides to roam, he interrupts the connectivity with the system. At a

[1,2,3] Faculty of Electronics, Telecommunications and Information Technology/Department of Communications, Technical University of Cluj-Napoca, 26-28 Bariţiu Street, 400027, Cluj-Napoca, România, email: {Emanuel.Puschita,Tudor.Palde}@com.utcluj.ro
[4] Faculty of Electrical Engineering and Computer Science, Electronics and Computers Departement, "Transilvania" University of Braşov, 1 Politehnicii Street, Braşov, Romania, email: sandu@unitbv.ro

time, the mobile user roams from the initial access point to another access point.
This type of roaming is deemed nomadic because the mobile node is not using network services when he roams, but only when he reach his destination.
Also, depending on which layer the roaming occurs, we could define two major types of roaming: **layer 2 roaming** and **layer 3 roaming**.
A layer 2 network is defined as a single IP subnet and broadcast domain, while a layer 3 network is defined as the combination of multiple IP subnets and broadcast domains. Layer 2 roaming occurs when a mobile node moves far enough that its radio associates with a different access point. With layer 2 roaming, the original and the new access point offer coverage for the same IP subnet, so the device's IP address will still be valid after the roam.
Layer 3 roaming occurs when a mobile node moves from an access point that covers one IP subnet to an access point that covers another IP subnet. At that point, the mobile node would no longer have an IP address and default gateway that are valid within the new IP subnet.

III. MOBILITY ANALISIS ON COMPUTER NETWORKS

One of the most challenging issues in the area of wireless communication systems is the provision of fast and efficient mobility support.
Next generation applications running on wireless networks require an emerging need to both provide mobile nodes with the ability to remain connected while being truly mobile, and to quickly restore their connections during any kind of handover inside WLAN.
For IEEE 802.11 implemented wireless LANs, the handover process is known as break before make, referring to the requirement that a mobile node serves its association with one AP (Access Point) before creating an association with a new one.
This process might seem to be inefficient because it introduces the possibility for data loss during roaming, but it facilitates a simpler MAC protocol. IEEE 802.11 defines a layer 1 physical interface and a layer 2 data link layer access mechanism. Layer 2 roaming is natively supported for 802.11 mobile nodes.
As 802.11 is layer 3 unaware, some upper-layer solution is required for layer 3 roaming. Also, the type of application is directly correlated to its resilience during the roaming process.
Connection-oriented applications (TCP based) are more tolerant to packet loss incurred during roaming process because TCP requires positive ACKs, just as the 802.11 MAC does, hence any data lost during the roaming process will be retransmitted.
Connectionless applications (UDP based) are time critical and the packet loss incurred during roaming process might cause a noticeable impact to applications because UDP do not use retransmissions.

A. *Layer 2 Handover*

Depending of the decision of where to roam, the mobile node must decide which AP to roam to.
There are two different AP discovery processes: **preemptive AP discovery** (scanning the medium for APs before the decision to roam), and **roam-time AP discovery** (scanning the medium for APs after the decision to roam). Each discovery process can employ one or both of the following mechanisms: active scanning (the mobile node actively searches for an AP, waits for responses from APs and decides which AP is the best one to roam to), and passive scanning (the mobile node does not transmit any frames, listens for beacon frames on each channel and continues to change channels at a set interval).
In conclusion, there is no ideal technique for scanning. Passive scanning has the benefit of not requiring the client to transmit requests but runs the risk of potentially missing an AP because it might not receive a beacon during the scanning duration. Active scanning has the benefit of actively seeking out APs to associate to but requires the client to actively transmit requests.

A.1. *Scenarios and simulations results*

In order to demonstrate the handover concept for IEEE 802.11 wireless LAN, a simple wireless scenario was realized using the ns-2 simulator.
The so-called wired-cum-wireless scenario contains two wireless nodes, each of them communicating through its own base-station (AP). The fixed network is simulated by a simple connection between the AP's and UDP traffic is set between the two mobile nodes using a CBR application. The rate is set to 100 kbps. This situation could simulate for example a VoIP application. In order to make possible the handover process, one of the nodes moves from the coverage area of one AP to the other one. The scenario topology can be seen clearly in the next screenshots.

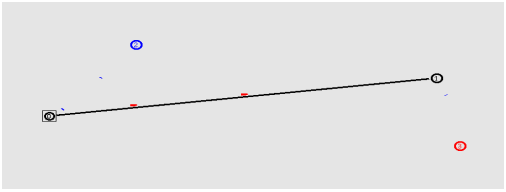


Figure 1. Mobile nodes positions before handover

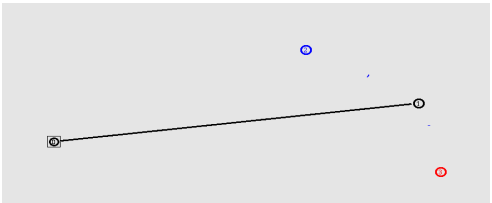


Figure 2. Mobile nodes final positions after handover

The end-to-end delay information is extracted from the corresponding trace file and the results are plotted in the graph presented below.

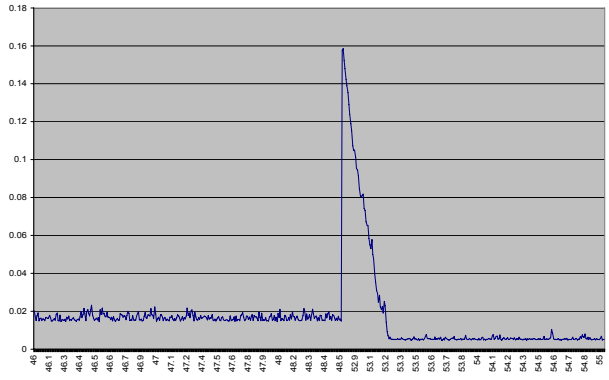


Figure 3. End-to-end delay representation in time [s] for IEEE 802.11 wireless LAN handover process

A.2. *Partial conclusions*

The maximum delay in communication indicates the initial handover moment. We can observe that the average delay is about 10 ms smaller when the two nodes use the same AP.
During the handoff procedure, we have some packets with very large delays (up to 160 ms), until the association with the new AP is achieved. Even though wireless LANs work over very high channel bandwidth, they show long network-layer handoff latency. This is a restraining factor for mobile clients using interactive multimedia applications such as voice over IP or video streaming.
For important and useful services like employing VoWLAN, the roaming latency (excessive latency and jitter, degraded voice quality) remains a challenge. New ways must, therefore, be examined for optimizing the time required to complete the inter-network APs transitions of wireless mobile nodes.

B. *Layer 3 Handover*

Layer 3 mobility is a superset of layer 2 mobility. In an 802.11 wireless LAN, the mobile node must perform a layer 2 roam, including AP discovery, before it can begin a layer 3 roam. Many applications require persistent connections with the network. To provide session persistence, it is needed a mechanism to allow a station to maintain the same layer 3 address while roaming throughout a network.
The key components in a Mobile IP enabled network are: the mobile node (MN), the home agent (HA), the foreign agent (FA), Care-of address (CoA), Co-located care-of address (CCoA). There are three main phases of Mobile IP: agent discovery, registration and tunneling.
IEEE 802.11 wireless LAN's permit network mobility, but to properly implement and deploy a mobility-enabled WLAN, we must understand the nature of the applications. There are two versions of mobile IP: IPv4 and IPv6. The main difference between Mobile IPv4 and Mobile IPv6 is that the last one includes in the core protocol some features that were just extensions for Mobile IPv4. We can include here the route optimization and the reverse tunneling.
Beside that, Mobile IPv4 uses tunnel routing to deliver data-packets to mobile node, while Mobile IPv6 uses tunnel routing and source routing with IPv6 Type 2 routing headers; the FA is used in Mobile IPv4 to de-capsulate the packets for the MN, while in Mobile IPv6 the packets are de-capsulated by the mobile node itself, eliminating the need for FA. If in Mobile IPv4, agent discovery is used for mobile detection, Mobile IPv6 uses IPv6 router discovery.
Similarly, Mobile IPv4 uses ARP to determine the link layer address of neighbors while Mobile IPv6 uses IPv6 neighbor discovery and is de-coupled from any given link layer.

B.1. *Scenarios and simulations results*

For the layer 3 handover section were design two similar simulations for Mobile IPv4 and Mobile IPv6.

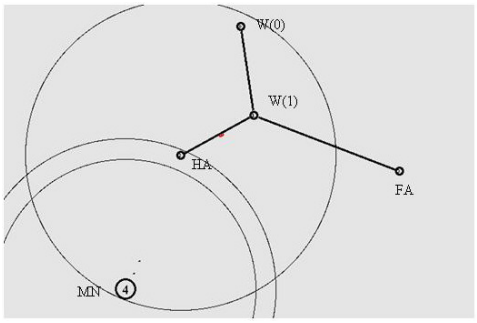


Figure 4. The mobile node in the home network

The Mobile IPv4 is natively supported by the standard version of the ns-2 simulator, but there is no support for Mobile IPv6 and an extension was patched.

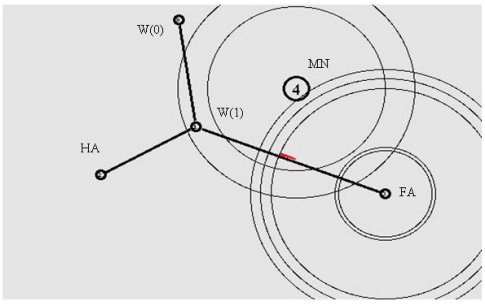


Figure 5. The mobile node in the visited network

The end-to-end delay is extracted from the two trace files (corresponding to Mobile IPv4 and Mobile IPv6) using an .awk script and plotting them we obtain the two graphs below.
The effect on the end-to-end delay of the packets can be seen. Due to the fact that a triangular routing is used, the packets received by the mobile node while being in the visited network have a significantly larger delay compared to the ones received when it is in his home network. Practically, all the packets are sent to the HA, and from here to the FA, which routes them

to the MN. This is one of the disadvantages of Mobile IPv4.

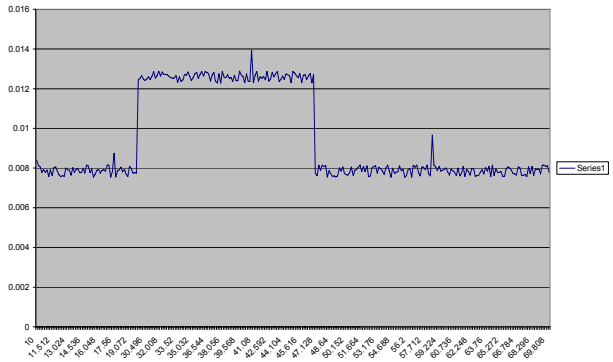


Figure 6. End-to-end delay representation in time [s] for Mobile IPv4 scenario

This disadvantage of is solved by Mobile IPv6. As discussed in a previous paragraph, Mobile IPv6 uses always the so called route optimization. So, as can be seen in the Figure 7, the difference between delays of the packets received in the foreign network and the ones received in the home network is not so large. This is due to the fact that, after the binding list and the binding cache are updated, the packets follow the shortest path to the destination without using the HA.

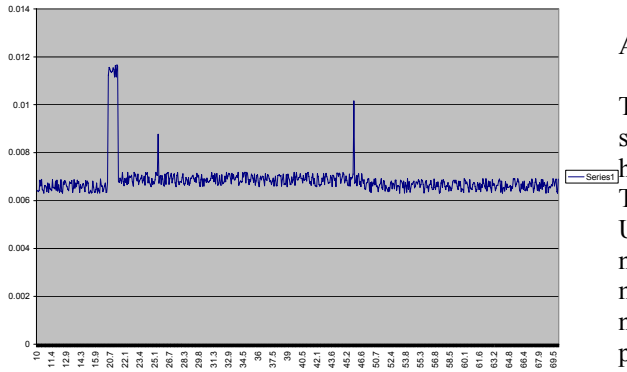


Figure 7. End-to-end delay representation in time [s] for Mobile IPv6 scenario

Only the first packets sent after the node has entered the foreign network have larger delays, because they follow the path through the home agent, until the Binding List is updated by the correspondent node. In fact, for this short period of time, the Mobile IPv6 actions exactly like Mobile IPv4 in order to route the packets.

B.2. *Partial conclusions*

The Mobile IP is generally used for nomadic mobility, although just theoretically it can be used for seamless mobility. The main reasons are the large delays, especially for the Mobile IPv4 version.
The Mobile IPv4 has several disadvantages: the triangular routing which means large delays for the packets; a HA may become a traffic and performance bottleneck and potential long handoff delay due to the registration process.

Some of these problems are solved by the Mobile IPv6 protocol (which introduces the route optimization as an alternative for the triangular routing), or by some "micro-mobility" management protocols, the Hierarchical MIP or Cellular IP.

IV. MOBILITY ANALYSIS ON CELLULAR NETWORKS

The UMTS architecture has been specified in order to offer higher flexibility to users than 2G networks could support.
Like in all the other cellular networks, handovers are the basic means of providing mobility. The idea is to reduce especially the number of handover failures compared to previous generation cellular communication systems.
Handovers can be classified in hard, inter-system and soft and softer handovers. Hard handover is the handover type where a connection is broken before a new radio connection is established between the user equipment and the radio access network. Inter-system handovers are necessary to support compatibility with other system architectures. Soft and softer handover are the CDMA specific handover types implemented in the UMTS system.

A.1. *Scenarios and simulations results*

The standard version of the ns-2 simulator doesn't support UMTS system; hence an additional package had to be installed.
The need was to extend the simulator to support UTRAN (new mobile nodes, layers and protocols), non-ad-hoc communication and routing for UE mobility. Then, as UMTS can be modeled as an all-IP network whereby all the transactions are based on IP protocols, with these entire modifications one can simulate a whole UMTS network by putting agents and applications on top of the nodes.
Two types of traffic are set between two mobile nodes. First a CBR application type is simulated with a rate of 13kbps, and then a HTTP connection with 64kbps. One of the mobile nodes stays fixed, while the other moves from one NodeB to another, performing the handover. Two screenshots of the scenario are presented below.

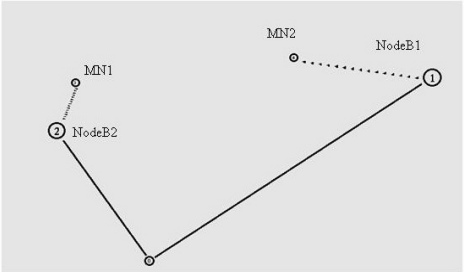


Figure 8. The mobile node before the handover process

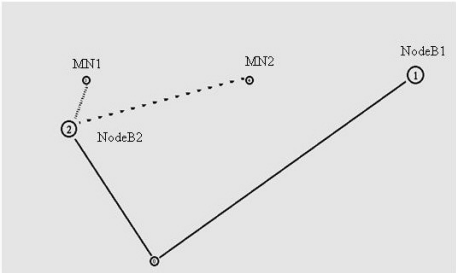


Figure 9. The mobile node after the handover process

The end-to-end delay is a very important parameter in handover analysis. This information is extracted from the trace file and then is plotted.

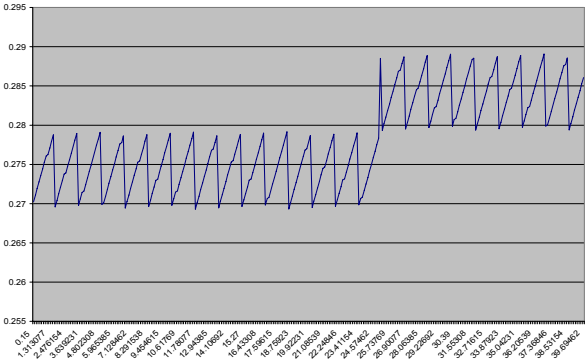


Figure 10. The end-to-end delay in a CBR UMTS application

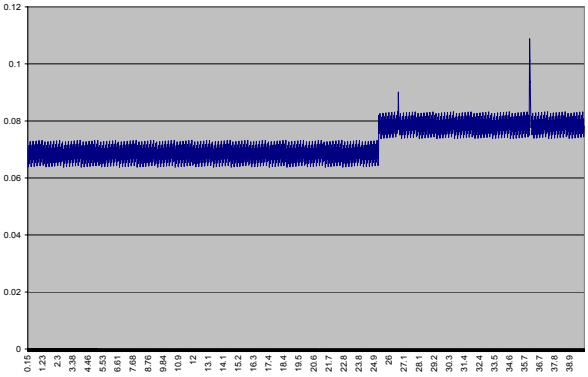


Figure 11. The end-to-end delay in a HTTP UMTS application

A.2. *Partial conclusions*

The delay is larger after the handoff, because the two mobile nodes don't communicate through the same NodeB anymore. Compared to the previous scenarios, the handover process has a minimum effect on the communication. This is normal, if we think that the cellular networks are designed to support a practically seamless handover for the mobile users.

V. CONCLUSIONS

The paper presents an intra-system mobility evaluation for different wireless technologies: computer networks (fixed and wireless networks) and cellular-based networks. The paper offers a complex simulation environment by using existing and patched modules for ns-2 network simulator.

There was simulated the handover processes for mobile IPv4, mobile IPv6, WLAN, UMTS. Also, there are tested best-effort and critical in time applications modeled by HTTP and CBR sources. Each scenario is accompanied by partial conclusions. Mobility as a network parameter and its contribution on delay of delivering packets is a good indicator of the network capabilities.

Comparing simulations results we observe that best performances are obtain in case of wireless networks handover process. That is because the handover decision is taken on layer 2 (WLAN) not on layer 3 (IPv6 and IPv4).

A layer 2 handover decision implies a less computational time on the mobile node vs. a layer 3 handover decisions. In case of fixed networks, the route optimization algorithm used by IPv6 version vs. IPv4 version makes the handover transfer more rapidly. Hence, the end-to-end delay will be smaller in case of using IPv6 version.

Evaluating the end-to-end delay for cellular networks we observe that in case of CBR application modeling critical in time applications over UMTS, the results are similar with IPv6 case.

In case of delivering HTTP non real-time services, the information is encapsulated on TCP datagram (connection-oriented transfer protocol), hence the delay increases.

We promote a layered QoS philosophy that separate QoS aspects on each layer, and each layer's parameters essentially contribute to the global QoS evaluation.

One of these parameters is end-to-end delay introduced by the handover process. Simulation results and conclusions presented in this paper complete and consolidate our work and vision related to the QoS layered approached which includes a complex analysis over different TCP/IP layers for wireless scenarios: study of medium access techniques, routing and transport protocols.

All work is done form the QoS perspective [2], [3].

Table 1. End-to-end delays for intra-system mobility on different wireless technologies

Network type		**Fixed network**		**Wireless network**	**Cellular network**	
Technology type		*IP core*		*WLAN IP core*	*UMTS IP core*	
Handover decision		*Layer 3*		*Layer 2*	*Intra-RNS handover*	
Protocol/Standard/Application		*IPv4*	*IPv6*	*IEEE 802.11*	*HTTP*	*CBR*
End-to-end delays	*min delay [s]*	0.007545s	0.00631s	0.004538s	0.269295	0.063795s
	max delay [s]	0.013936s	0.011662s	0.145834s	0.289065	0.108828s

VI. ACKNOWLEDGEMENTS

The QoS perspective concept presented in this paper was developed in BROADWAN FP6, as TCN contribution to D8 [5].
The work included in this paper was done also whit support of SIEMENS PSE SRL România.

REFERENCES

[1] A. Ganz, Z. Ganz, *Multimedia Wireless Networks: Technologies, Standards, and QoS*, Prentice Hall, 2003.

[2] Puşchiţă, E., Palade, T., *Performance Evaluation of Routing Protocols in Ad-Hoc Wireless Networks*, ETc2004, Timişoara, RO, October 2004, pp. 352-357.

[3] Puşchiţă, E., Palade, T., Chira, L., *Performance Evaluation of DCF vs. EDCF Data Link Layer Access Perspective*, TELSIKS'05, September, 2005, Niš, IEEE Catalog Number 05EX1072, pp. 356-359.

[4] Palade, T., Puşchiţă, E., Chira, L., Căruntu, A., *Network Simulator, a Simulation Tool for Wireless LAN*, METRA, Bucureşti, RO, May 2005, pp. 556-561.

[5] P. Hirtzlin (editor), E. Puşchiţă, T. Palade, D. Salazar, *Recommendations on matching hybrid wireless network technologies to deployment scenarios*, BROADWAN, D8, October 2004.

Buletinul Ştiinţific al Universităţii "Politehnica" din Timişoara

Seria ELECTRONICĂ şi TELECOMUNICAŢII
TRANSACTIONS on ELECTRONICS and COMMUNICATIONS

Tom 51(65), Fascicola 2, 2006

How to Choose a Model for Ad hoc Wireless Networks

Rodica Stoian[1], Adrian Raileanu[2]

Abstract – This paper studies ad hoc wireless networks using a network information theory point of view. Two classes of networks are analyzed in the paper, considering the location of the nodes and the traffic graphs: arbitrary and random. Three theoretical models are presented for multi-hop transport, and each of them takes into account different aspects of these types of networks: protocol restrictions, interference, bandwidth. The minimal model parameters are inventoried, and their influence on the model behavior is discussed. New metrics are introduced, to allow a more accurate representation of the information flow in wireless networks. The current status and difficulties of the traditional information theory to describe this multiple input-output system are discussed. The third model that we introduce is an extension of the interference model, that adds a new parameter, bandwidth, and an optimum criteria using results from information theory of MIMO systems. The intention to bring together information theory and network protocols is the right way to analyze the limitations of the current implementations of such systems.

Keywords: wireless networks, ad hoc networks, network information theory, network transport capacity, network model metrics.

I. INTRODUCTION

Wireless networks are communication networks that use radio as their carrier. A wireless ad hoc network is a decentralized network of nodes with radios, possibly mobile, sharing a wireless channel and asynchronously sending packets to each other, generally over multiple hops. The most notable characteristics of an ad hoc network are a lack of infrastructure, multi-hop communication by cooperative forwarding of packets, distributed coordination among nodes, dynamic topology, and the use of a shared wireless channel.

The potential for deployment of ad hoc networks exists in many scenarios, for example, in situations where infrastructure is infeasible or undesirable, like disaster relief, sensor networks, etc. Ad hoc networks also have the potential of realizing a free, omnipresent communication network for the community. This comes with a price, too. Due to the lack of a central unit, routing, medium access and power control rise many problems that did not exist in wired or cellular networks.

Medium access in ad hoc networks is a complex problem. Multiple access schemes popular in cellular networks are not easy to implement in ad hoc networks because of the need to dynamically allocate resources efficiently.

Another aspect of the wireless networks is reducing the interference caused by various transmissions, which is critical for the efficiency and scalability of any wireless system. This motivates transmission power control, which is a very complex problem for ad hoc networks.

In this paper we analyze some ad hoc networks wireless models, considering two possible scenarios: arbitrary and random network classes. Then, we discuss the metrics and parameters that can be used to characterize these communication systems in a network information theory fashion..

II. MODELS FOR ARBITRARY NETWORKS

We call *arbitrary* the class of networks that have arbitrary locations of the nodes, and arbitrary traffic patterns The wireless network model presented here consists of n nodes located arbitrarily in a limited area, a plane disc of unit area. Each node can transmit over the wireless channel with a maximum rate of W bps. However, each node can choose an arbitrary lower rate for the next transmission. The information is sent from node to node (multi-hop) to the final (arbitrary) destination.

As shown in Figure 1, several nodes can make successful transmissions simultaneously due to spatial separation and the absence of interference from others. A successful transmission over one hop is conditioned by the access to the medium and the level of interference with the neighboring nodes. For medium access, a *protocol model*, and for power related issues, a *physical model* from [1] will be presented. For both models, the nodes are denoted X_i, which also stands for the location of the node.

[1] Professor PhD, Faculty of Electronics,Telecommunication and Information Technology, University Politehnica of Bucharest, 061071 ROMANIA, rodicastoian2004@yahoo.com
[2] PhD Student, Faculty of Electronics,Telecommunication and Information Technology, University Politehnica of Bucharest, 061071 ROMANIA, adrianrai@yahoo.com

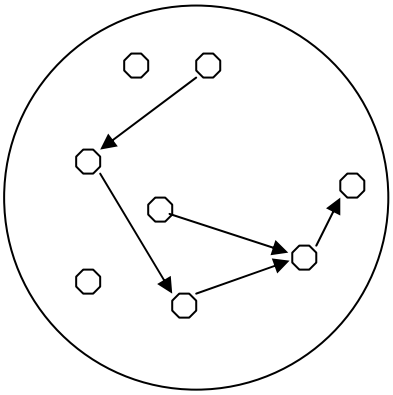


Figure 1. Multi-hop transmission in an ad hoc wireless network

The Protocol Model takes into account the distance D_{ij} between transmission nodes compared to the distance D_{kj} between interference nodes and the destination node. The transmission is considered successful if:

$$D_{kj} \geq (1+\Delta) D_{ij}, \quad D_{kj} = |X_k - X_j|, \quad D_{ij} = |X_i - X_j| \tag{1}$$

The node X_i transmits on one channel to node X_j, and X_k is another node transmitting simultaneously on the same channel. The parameter Δ stands for the guard zone specified by the protocol to prevent neighboring nodes from using the same channel at the same time.

The *Physical Model* takes into account the power transmission level chosen by each node that is simultaneously transmitting at some instant over a certain channel. Let $\{X_k; k \in T\}$ be the subset of nodes that transmit at the same time and P_k the power level chosen by node X_k. The transmission from node $X_i, i \in T$ is successfully received by a node X_j if the *signal-to-interference and noise ratio* (SINR) is larger than a threshold β:

$$SINR = \frac{\frac{P_i}{|X_i - X_j|^\alpha}}{N + \sum_{\substack{k \in T \\ k \neq i}} \frac{P_k}{|X_k - X_j|^\alpha}} \geq \beta \tag{2}$$

In equation 2, N denotes the power level of ambient noise, and α is the attenuation exponent, as the signal power decays with distance as $r^{-\alpha}$. We could suppose $\alpha > 2$, which is true for a limited neighborhood around the transmitter.

III. MODELS FOR RANDOM NETWORKS

The *random* networks class model presented here consists of n nodes located randomly, independently and uniformly distributed, in a limited area, a plane disc of unit area. Each node will transmit to a randomly chosen destination with a maximum rate of W bps. However, each node will choose a random rate for the next transmission. The information is sent by multi-hopping to the final destination. The destination node is independently chosen as the node nearest to a randomly located point (uniformly and independently distributed), thus destinations are at the average distance of 1 m.

As a opposed to the arbitrary networks, we assume that all transmissions employ the same nominal power. The two models, protocol and physical, are analyzed here.

In the *Protocol Model*, all transmissions share a common range denoted r. The distances D_{ij} (between transmission nodes) and D_{kj} (between interference nodes and the destination node) are compared now against this common range. The transmission is considered successful if these two conditions are met (see notations in eq. 1):

i) the transmission distance is less than r:

$$D_{ij} \leq r \tag{3}$$

ii) other nodes that transmit on the same channel are outside the transmission region given by r and the guard zone Δ:

$$D_{kj} \geq (1+\Delta) r \tag{4}$$

The *Physical Model* assumes that all nodes use the same power level P for all the transmissions. As in arbitrary networks case, $\{X_k; k \in T\}$ is the subset of nodes that transmit at the same time. The transmission from node $X_i, i \in T$ is successfully received by a node X_j if the SINR is larger than a threshold β:

$$SINR = \frac{\frac{P}{|X_i - X_j|^\alpha}}{N + \sum_{\substack{k \in T \\ k \neq i}} \frac{P}{|X_k - X_j|^\alpha}} \geq \beta \tag{5}$$

IV. METRICS AND ANALYSIS

The constraints defined by the Protocol Model are local. They only require certain regions of transmitters to be free of receivers. On the other hand, the Physical Model considers the cumulative interference due to all the nodes in the network. Thus, intuitively it appears that the Physical Model is a much more restrictive model, and would offer lower capacity. However, going deeper into analyzing both model's capacity may lead to invalidating this intuition. The purpose of the theoretical model of a communication system (and not exclusively communication) is to provide a means for deriving certain bounds on the performance parameters that are of interest.

One such parameter is the *network transport capacity* (NTC), i.e. how much information can possibly be transported, which can be representative for a class of wireless networks, by providing the limitations and the scaling capacity of different network architectures and protocols. The unit chosen by [1] for measuring the NTC is taking into consideration the amount of information that is transmitted successfully over time and space: the *bit-meter*. This metric, derived from information theory, is the quantity of network transported information when one bit has been transported a distance of one meter towards its destination. The same bit is taken into account only once in the case of one source transmitting to multiple destinations.

This metric combined with the individual transmission rates for each node offers an appropriate measure unit for the NTC, the *bit-meter per second* (bmps). If compared to Shannon's information capacity theorem, the bit-meter is analog to the bit-per-channel-use that measures channel capacity, and the bit-meter per second is analog to the bit-per-second measuring data rate that can be achieved using the channel.

V. THE INFORMATION THEORY MODEL

Information theory provides models and bounds that describe communications systems, and assume an information source, a transmission channel and a receiver. In the recent years, efforts are taken to use these elements to derive theoretical bounds for systems with many senders and receivers. The new flavor is called Network Information Theory and was first studied in [2]. The notions defined for single channel can be used to derive models for multiple user communications, but seem not sufficient to include all the effects that may appear: interference between users, distributed communication, bursty sources, simultaneous access, etc.

Network channels can be divided into three categories:

- MIMO (Multiple sources, multiple receivers) – this is the most general network case, where many users share the same medium to communicate, and is still one of the unknowns with respect to the model to be used.

- SIMO (Single source, multiple receivers) – also known as the broadcast channel.

- MISO (Multiple sources, single receiver) – also known as the multiple access channel.

A pure Shannon approach is not sufficient to derive bounds on channel capacity for networks. Information theory deals only with noise affecting the transmission, but cannot provide a model for packet arrival with delay. Network theory, on the other hand, is more empirical, and was based on continuous tweaking through performance observation. The matter of noise and interference is viewed as simple as "bad packet rate".

The triumphs of network information theory, as presented in [2] are mainly in the SIMO and MISO channels area. For these two cases, the channel capacity theory has taken the form of achievable rate regions that allow the users to transmit at maximum rates with a low probability of error.

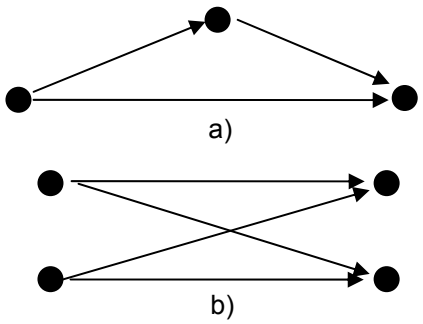


Figure 2. Examples of MIMO simple channels: a) the relay channel, b) the interference channel

The cases of the relay and interference channels are still in need of a general purpose information theory. The difficulty comes from the big number of ways of interaction between peers. In relay mode, nodes can act as repetitive units, amplifiers or coherent repetitive units. In the case of interference channel, some of the nodes can cooperate in canceling the interference introduced by other nodes.

The two network models presented in [1], do not take into account limited bandwidth and rate restriction for a successful transmission. The latter may be seen as embedded into eq. (2) and (5), i.e. the nodes transmit at different rates, and due to SINR, some transmissions may not be successful. We propose another approach to multi-node transmission, taking into account bandwidth, power, channel capacity and thus the achievable rate.

Channel capacity for a multi-access channel was studied in [2], and the results are used to describe transmission between nodes. According to this, transmission is successful (at one node), if the sum of all the incoming transmissions rates W_i does not exceed channel capacity C to that node, see eq. (6).

$$\sum_{i=1}^{n} W_i \le \frac{1}{2}\log(1+\sum_{i=1}^{n} SNR_i) = C_{node} \qquad (6)$$

Having several transmissions, at the same time, towards a node, the only combination of rates that yields safe decoding of the output is the achievable rate region ($W_1,...W_n$). The decoding is performed first on the lowest rate data, then this is subtracted from the mixed signal and next lowest rate is used for the new signal, and so on, until the highest rate data is decoded.

Considering a limited bandwidth transmission, and using the Physical model results, we consider the capacity of the channel between two nodes X_i and X_j (the transmission bandwidth is B_{ij}):

$$\begin{aligned} C_{ij} &= B_{ij}\log_2(1+SINR) \\ &= B_{ij}\log_2\left(1+\frac{\frac{P_i}{|X_i-X_j|^{\alpha}}}{N_0 B_{ij}+\sum_{\substack{k\in T\\ k\neq i}}\frac{P_k}{|X_k-X_j|^{\alpha}}}\right) \end{aligned} \qquad (7)$$

The result in (7) seems more restrictive than (6), as it forces an upper bound on the rate that can be used between the two nodes, so that transmission is performed successfully. In fact these two results have distinct roles in helping the design of an optimal rate scheme:

- the sum of all rates that are used at the same time towards one node should not exceed the capacity defined in (6), or, in other words, only a set of achievable rate transmissions will be decoded by the node;
- the rate of one particular transmission between two nodes should not exceed the capacity defined in (7); higher rates will not be decoded correctly.

Another remark is that the result can be used to calculate the overall NTC of a specific network class. Further study will provide what is the best combination of capacities as in (6) and (7) that will maximize the NTC.

This model offers at least one more parameter that will be taken into account when deriving the upper bound of NTC: bandwidth. As more parameters can be included in the model, the more restrictive it becomes, but it comes closer to the real world. Bounds are useful for developing of scheduling algorithms, assuming that are known the locations of all nodes and the overall traffic demand. These algorithms coordinate all transmissions temporally and spatially in a way to minimize collisions. If the model shows that the NTC decreases with *n* (as in the case of ALOHA system), this will help the designers to target their efforts towards developing smaller wireless networks, and inter-connecting them using wired transport.

VI. CONCLUSIONS

The paper introduces two types of models for the multi-hop ad hoc wireless network. First type consists of the Protocol Model that allows to identify successful transmissions based on the distance between nodes, and the Physical Model that is based on the signal-to-interference-and-noise ratio that affects the transmission from one node to another. To make things more generally, two classes of networks were used: arbitrary networks and random networks. The choice of having more than one class and more than one model is intended. The second type of models that we introduced, extends the existing modules by including more parameters (e.g. bandwidth). It uses an optimum criteria, the achievable rate problem, to relate NTC to rate and bandwidth. Having a simple feedback between nodes, because there is no controlling unit in ad-hoc networks, the maximum network throughput can be achieved by wisely choosing the rates, power and bandwidth. The network transport capacity, calculated for each model, will be able to drive the network designers in the right direction, when it comes to the choice of network parameters.

Together with a series of parameters that help to describe the behavior and performance of these networks, these models are a step forward towards creating theoretical models of communication that can allow us to study their limitations and seek opportunities for improvement. Distance was incorporated into these models, as new metrics were defined to allow the information theoretic approach (e.g. bit-meters per second).

There are still effects that are not introduced yet in the models. The channels employed do not take into account fading, multi-path propagation and other effects. However, the simplicity of these models allows for an initial determination of the capacity of the networks, and then, they can be developed in better models.

REFERENCES

[1] P. Gupta and P. R. Kumar, “The Capacity of Wireless Networks," IEEE Transactions on Information Theory, vol. IT-46, March 2000, pp.388

[2] T. Cover and J. A. Thomas, “Elements of Information Theory”, Wiley, 1991.

[3] V. Kawadia, “Protocols and Architecture For Wireless Ad Hoc Networks”, Dissertation, Electrical Engineering, University of Illinois at Urbana-Champaign, 2004

[4] P. Karn, “MACA - A new channel access method for packet radio", in Proc. 9th Computer Networking Conf., Sept. 1990, pp. 134–140.

[5] V. Bharghavan, A. J. Demers, S. Shenker, and L. Zhang, “MACAW: A media access
protocol for wireless LANs", in SIGCOMM, 1994, pp. 212 - 225.

Buletinul Ştiinţific al Universităţii "Politehnica" din Timişoara

Seria ELECTRONICĂ şi TELECOMUNICAŢII
TRANSACTIONS on ELECTRONICS and COMMUNICATIONS

Tom 51(65), Fascicola 2, 2006

A Very General Family of Turbo-Codes: The Multi-Non-Binary Turbo-Codes

Horia Balta[1], Maria Kovaci[1], Alexandre de Baynast[2], Calin Vladeanu[3], Radu Lucaciu[1]

Abstract – This paper presents a new family of turbo codes whose the constituent codes have $R \geq 1$ non-binary inputs and $R+1$ outputs. We refer this family as the multi input non-binary turbo codes (MNBTC), which is very general. More specifically, we show that this family includes the multi-binary turbo-codes (MBTCs) that themselves include the classical binary turbo-codes (BTCs). Moreover, it also includes the turbo-codes with Reed Solomon codes as constituent codes. In this paper, we fully describe the encoding process and the extension of the Maximum A Posteriori (MAP) decoding algorithm, especially the trellis closing issues for these codes. Additionally, we show by simulations the benefit of using this family of Turbo-codes.

Keywords: turbo-code, MAP algorithm, multi non-binary convolutional code

I. INTRODUCTION

The discovery of the turbo codes (TCs) [1] represents a major breakthrough in the coding theory since the asymptotic performance of the TCs close the gap to the Shannon limit within tenths of decibels. A sensitive component of the TCs is the interleaver. Several interleavers have been recently proposed: the S-interleaver [2], Takeshita-Costello interleaver [3] and ENST interleaver [4] just to name a few. Moreover, different designs and decoding algorithms for TCs have been investigated: the technique of the circular codes [5], the serial concatenation [6], the decoding algorithms: SOVA [7], MaxLogMAP and LogMAP [8].

The recent introduction of MBTCs in [9] is a further step to close the gap to the Shannon limit. Indeed, the MBTCs offer more advantages than BTC such as lower error floor for moderate codeword size and faster convergence [10]. These advantages are crucial in the current and future wireless systems as IEEE 802.11n and IEEE 802.16.

In this paper, we extend the MBTC concept to the non-binary case: Whereas the constituent codes of the MBTC have R binary inputs, we consider TC with constituent codes with R non-binary inputs. We refer this new family as multiple input non-binary turbo-codes (MNBTC). Formally, the code has the same structure as the multi binary code but the arithmetic operations are now performed in GF(2^Q), Galois field of order Q. In this paper, we propose to analyze the MNBTC and to compare them to the MBTC from the viewpoints of decoding algorithms and performance.

The rest of the paper has the following structure. In the next section the construction of the constitutant codes of the MNBTC is presented. Section III is dedicated to the trellis closing problems of the MNB code. In section IV we present several variations of extended versions of MAP decoding algorithms for the MNB codes. In Section V, we show that under some restriction on the polynomials, a Reed-Solomon (RS) code is a particular case of MNB, so the MNBTC family includes an interesting new class of TC, the RS-TC. Finally, some experimental results and concluding remarks are presented in Section V and VI, respectively.

II. MULTI NON BINARY CONVOLUTIONAL ENCODER AND TRANSMISSION CHANNEL

In this section, we describe the encoding scheme of the constituent codes of the MNBTC. Each constituent encoder has R non-binary inputs and is referred as multi non-binary code (MNB). In Fig.1 we present the general scheme of a MNB convolutional encoder, with rate $Rc=R/(R+1)$. Throughout the paper, we focus on recursive and systematic codes due to their superior performance. Each register in Fig.1 stores a vector of Q bits at the time. All the links are supposed to have a width equal to Q in order to carry a vector of Q bits. Each block $g_{r,m}$, with $r=1,...,R$, $m=0,...,M-1$, represents a multiplier in $GF(2^Q)$ whereas the adders perform the sum in $GF(2^Q)$. At time n, the encoder has R inputs, u_1^n, u_2^n,..., u_R^n and

[1] Facultatea de Electronică şi Telecomunicaţii, Departamentul Comunicaţii, Bd. V. Pârvan Nr. 2, 300223 Timişoara, e-mail horia.balta@etc.upt.ro, maria.kovaci@etc.upt.ro, radu.lucaciu@etc.upt.ro

[2] Department of Electrical and Computer Eingineering, Rice University, MS-366 – 6100 Main Street, Huston, Texas 77005, e-mail debaynas@rice.edu

[3]Universitatea Politehnica din Bucuresti, Facultatea de Electronică şi Telecomunicaţii si Teoria Informatie, 1-3 Iuliu Maniu, Bucuresti, e-mail cvladeanu@yahoo.com

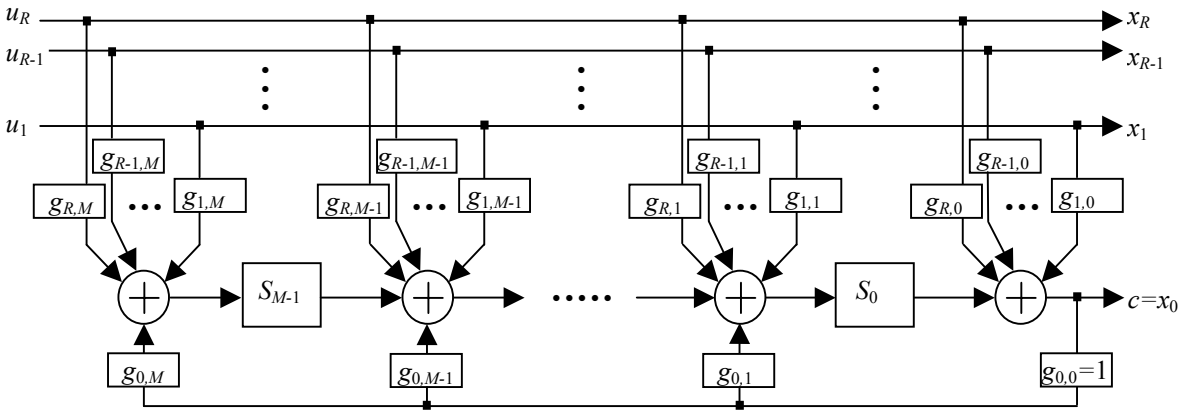


Fig.1 Multi-Non-Binary Convolutional Encoder – general scheme.

$R+1$ outputs $x_1^n, x_2^n, \ldots, x_R^n$ corresponding to the R inputs and one redundant bit x_0^n also referred as c^n. The current encoder state is given by the outputs of the M shift registers S_0^n, $S_1^n, \ldots, S_{M-1}^n$. We adopt the following compact notations: $s^n = \left[S_{M-1}^n \quad S_{M-2}^n \quad \ldots \quad S_0^n\right]^T$, $u^n = \left[u_R^n \quad u_{R-1}^n \quad \ldots \quad u_1^n\right]^T$, $0 \le n \le N$, for the encoder state vector and the ``input word'', respectively.

The input/current state and output/current state relations of the encoder at the time n can be respectively expressed in the compact form:

$$\left(s^{n+1}\right)_{M\times 1} = \left(G_T\right)_{M\times R} \cdot \left(u^n\right)_{R\times 1} + \left(T\right)_{M\times M} \cdot \left(s^n\right)_{M\times 1}. \quad (1)$$

$$c^n = G_L \cdot u^n + W \cdot s^n. \quad (2)$$

where: $G_T = G_F \cdot G_L + G_0$ and $W = [0\ 0\ \ldots\ 0\ 1]_{1\times M}$. $G_0 = [g_{r,m}]_{M\times R}$ denotes the partial generator matrix, $1 \le m \le M$, $1 \le r \le R$, which excludes the feedback coefficients and the generator coefficients for the redundant symbol. The vector $G_F = [g_{0,M}\ g_{0,M-1}\ \ldots\ g_{0,1}]^T$ contains the coefficients of the feedback loop. and $G_L = [g_{R,0}\ g_{R-1,0}\ \ldots\ g_{1,0}]$. The matrix T equals:

$$T = \begin{bmatrix} 0 & 0 & 0 & \ldots & 0 & g_{0,M} \\ 1 & 0 & 0 & \ldots & 0 & g_{0,M-1} \\ 0 & 1 & 0 & \ldots & 0 & g_{0,M-2,} \\ \ldots & \ldots & \ldots & \ldots & \ldots & \ldots \\ 0 & 0 & 0 & \ldots & 1 & g_{0,1} \end{bmatrix} = \begin{bmatrix} 0_{1\times M-1} & G_F \\ I_{M-1} & \end{bmatrix}.$$

Additionally, we define the full generator matrix G such as: $G = [g_{r,m}]_{(M+1)\times(R+1)} = [g_R\ g_{R-1}\ \ldots\ g_1\ g_0]_{10}$, $0 \le m \le M$, $0 \le r \le R$.

The necessary and sufficient condition to have a decodable code is such that the matrix G_T is full rank.

Applying the „D" transform ($X(D) = \sum_{k=-\infty}^{+\infty} x^k \cdot D^k$) to the equations (1) and (2) we obtain:

$$D^{-1} \cdot S(D) = G_T \cdot U(D) + T \cdot S(D), \quad (3)$$

$$C(D) = G_L \cdot U(D) + W \cdot S(D). \quad (4)$$

After some basic manipulations, it can be shown that:

$$C(D) = \sum_{r=1}^{R} \frac{g_r(D)}{g_0(D)} \cdot U(D) \quad (5)$$

where $g_r(D) = \sum_{m=0}^{M} g_{r,m} \cdot D^m$.

In order to understand better the encoding procedure, we give an example in $GF(4)$. We recall the addition and multiplication operations on $GF(4)$ in Fig.2., [11].

+	0	1	2	3
0	0	1	2	3
1	1	0	3	2
2	2	3	0	1
3	3	2	1	0

*	0	1	2	3
0	0	0	0	0
1	0	1	2	3
2	0	2	3	1
3	0	3	1	2

Fig. 2 The addition and the multiplication in $GF(4)$.

Consider the double-non-binary convolutional code defined by the following generator matrix:

$$G = \begin{bmatrix} 2 & 1 & 3 \\ 0 & 3 & 1 \\ 3 & 1 & 1 \end{bmatrix}. \quad (6)$$

By construction, from G, we have: G_L=[3 1], G_F=[3 1]^T, G_T=[0 2;3 2] and T= =[0 3;1 1]. Consider two input sequences u_1 and u_2 starting as follows: u_1 = [1 2 3 3 1 3 2 …] and u_2 = [0 2 1 1 3 0 3 ...]. The values of the state vector of the encoder are determined according to (1) with initial state s^0=[0 0]^T. After some basic calculations based on the operation tables given in Figure 2, it is easy to show that s^1=[0 3]^T and c^1=3 where c^1 is determined by (2), s^2=[1 1]^T and c^2=0, s^3=[1 0]^T and c^3=2, s^4=[2 1]^T and c^4=3, s^5=[2 1]^T and c^5=1 and s^6=[3 1]^T and c^6=3, etc.

After the description of the encoding process for each constituent code, we describe next the full encoding method for the MNBTC.

Fig.3 shows the scheme for the MNBTC. The input sequence $(u^n)_{0\le n<N}$ = [u^0 u^1 … u^{N-1}] represents a sequence of $N+1$ consecutive input words. Every word of size $R\times 1$ is formed by stacking R consecutive symbols, i.e., u^n = [u_{R+1}^n u_R^n … u_2^n]^T. In turn, each symbol results from a mapping of Q bits such as:

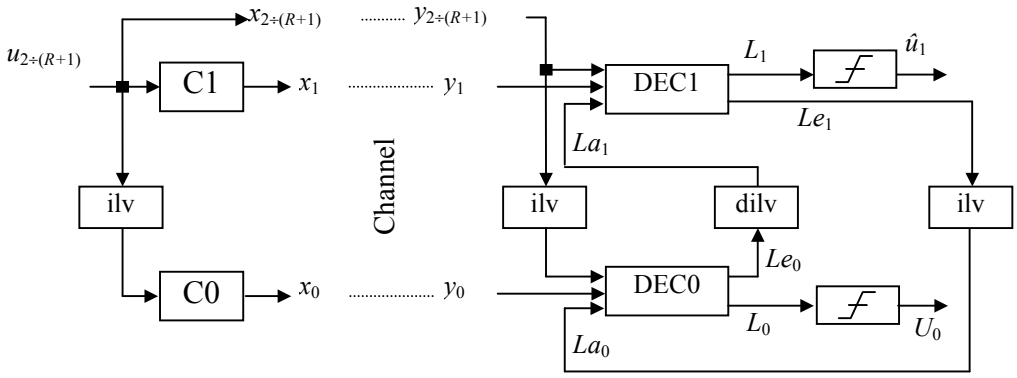


Fig. 3 Principle of the encoder and decoder for the Multiple inputs Non Binary Turbo-Code.

$u_r^n = [\, u_{r,Q-1}^n \; u_{r,Q-2}^n \; \dots \; u_{r,0}^n \,]^T$, $2 \le r \le R+1$, with: $u_{r,q}^n \in \{0, 1\} = GF(2)$, $0 \le q < Q\text{-}1$.

At the output of the encoders C1 and C0, we obtain the codeword $(x^n)_{0\le n<N} = [x^0 \; x^1 \; \dots \; x^{N-1}]$, where each word $x^n = \left[x_{R+1}^n \; x_R^n \; \dots \; x_2^n \; x_1^n \; x_0^n\right]^T$ is composed by R+2 symbols: the first R symbols correspond to the input symbols whereas the last two symbols correspond to the two redundant symbols. As for the input sequence case, each symbol of the codeword corresponds to Q bits $x_r^n = \left[x_{r,Q-1}^n \; x_{r,Q-2}^n \; \dots \; x_{r,0}^n\right]^T$.The codeword of length N that corresponds to $N\times(R+2)\times Q$ bits is then modulated (throughout the paper, for sake of simplicity we consider BPSK or QPSK signaling) and then transmitted through the channel to its destination. The destination received a noisy version $(y^n)_{0\le n\le N}$ of the transmitted signal. By using the same formulation as we did for the transmitted sequence $(x^n)_{0\le n<N}$, the received sequence can be expressed as: $y^n = \left[y_{R+1}^n \; y_R^n \; \dots \; y_2^n \; y_1^n \; y_0^n\right]^T$.

The vector y^n has R+2 components where each component can be represented itself by a vector of Q values $y_r^n = \left[y_{r,Q-1}^n \; y_{r,Q-2}^n \; \dots \; y_{r,0}^n\right]^T$. Each component $y_{r,q}^n$ can be modeled for a transmission over additive white Gaussian channel as: $y_{r,q}^n = x_{r,q}^n + w_{r,q}^n$, $0 \le q < Q$, where $w_{r,q}^n$ represents the receiver noise. We model $w_{r,q}^n$ as zero-mean mutually independent Gaussian random sequences with variance σ^2.

III. THE MULTI NON BINARY DECODING AND THE TRELLIS CLOSING

In this section, we describe the modifications of the MAP decoding algorithm that are needed in order to decode the MNBTC codes.

Considering the two decoders DEC1 and DEC0 in Fig.3. For any variation of the iterative MAP decoding algorithm, e.g., MAP, LogMAP, MaxLogMAP or SOVA, the decoding of the MNBTCs can be done in different manners: per word, per symbol or per bit. The three methods calculate differently the a priori probabilities, the extrinsic messages, and the a posteriori probabilities (APP). We successively describe the calculation for the three cases:

Word-wise decoding supposes that all probabilities, i.e. the a priori, extrinsic and a posteriori probabilities correspond to one word among the $N_W = (2^Q)^R$ possible words. Using the same notations as in Fig.3, we can express the APPs $L_j^{n,i}(d)$ and the extrinsic probabilities $Le_1^{n,i}(d)$ for each decoder j, j = 0 or 1, at iteration i as:

$$\begin{aligned} L_1^{n,i}(d) &= La_1^{n,i}(d) + Y_1^n + Le_1^{n,i}(d), \\ L_0^{n,i}(d) &= La_0^{n,i}(d) + Y_0^n + Le_0^{n,i}(d), \end{aligned} \tag{7}$$

where $d \in \Delta = \{0, 1 \dots N_W\text{-}1\}$ denotes the index of the candidate word $u^n(d)$ among the N_W possible words of the code. Y_1^n and Y_0^n correspond to the bit-wise received sequence and are determined as follows:

$$Y_j^n = \frac{1}{\sigma^2} \cdot \sum_{r=2}^{R+1} \sum_{q=0}^{Q-1} x_{r,q}^n \cdot y_{r,q}^n + \frac{1}{\sigma^2} \cdot \sum_{q=0}^{Q-1} x_{j,q}^n \cdot y_{j,q}^n \;, \tag{9}$$

with j = 0 or 1 and and noise dispersion σ^2.

Thus, the a priori probabilities are expressed as:

$$\begin{aligned} La_1^{n,i}(d) &= \pi^{-1}(\, Le_0^{n,(i-1)}(d)), \\ La_0^{n,i}(d) &= \pi(\, Le_1^{n,(i-1)}(d)). \end{aligned} \tag{8}$$

The operations $\pi(\,)$ and $\pi^{-1}(\,)$ denote the interleaving and the de-interleaving operations, respectively (ilv" and „dilv" respectively in Fig.3).

Note that the number of components for the extrinsic probabilities is equal to the number of outgoing

vertices at any node of the treillis since each vertice corresponds to one possible value of the information word.
After several iterations, the estimated word $\hat{u}$ of the transmitted sequence can be determined for each n by searching the largest value of the APP given by one of the decoders:

$$\hat{u}^n = \max_{d \in \Delta} L_j^{n,i}(d). \quad (10)$$

For the *Symbol-wise decoding* algorithm, the APPs and the extrinsic probabilities are computed per each symbol u_r^n, $1 \le r \le R$, at time n. Since a symbol corresponds to Q bits, there are 2^Q possible values for the estimate of u_r^n. Thus, at iteration i, both decoders compute $R \cdot 2^Q$ values for the APPs and extrinsic probabilities. This decoding strategy is similar to the approach proposed in [12] for the MBTC codes.

For the *Bit-wise decoding* algorithm, either the APPs, or the log likelihood ratios (LLRs) can be used. In the first variant, we compute two values which correspond to the binary values 0 or 1 for every bit $u_{r,q}^n$ of each symbol u_r^n of the every word u^n, and that from all N words of the original sequence u. Thus, both decoder, at iteration i, calculate $2 \cdot R \cdot Q$ values for APP and the extrinsic probabilities. Alternatively, the LLRs can be computed for the $R \cdot Q$ bits of u^n as in [8]. The decoding algorithms correspond to the ones that are used for the BTC.
The word-wise decoding approach has the largest computation complexity compare to the symbol-wise and bitwise approach since the computational complexity is exponential with respect to the Galois field order Q and the number of inputs R whereas the computational complexity of the word-wise and the bit-wise approaches is linear in the number of inputs R and exponential (resp. linear) with respect to the number of inputs Q for the symbol-wise (resp. bit-wise). However, the word-wise decoding provides better performance as it is demonstrated in Section V.
After describing the update of the probabilities at the decoder, we detail the trellis closure. The trellis closure of the MNBC is more complicated than from the NBC because the trellis itself is more complicated. In this paragraph, we propose a new termination scheme for the MNBTC.
For a given input sequence $[u^0\ u^1\ \dots\ u^{N-1}]$, the final state of the trellis represented by s^N is:

$$s^N = \sum_{j=0}^{N-1} T^{N-j-1} \cdot G_T \cdot u^j + T^N \cdot s^0. \quad (11)$$

The trellis closure can be performed in two different ways: i) by making the trellis circular; ii) by zero-padding as in [5,12].
The trellis is *circular* if $s^N = s^0$. Replacing this equality in (11), we obtain:

$$(I_M + T^N) \cdot s^0 = \sum_{j=0}^{N-1} T^{N-j-1} \cdot G_T \cdot u^j = s^x. \quad (12)$$

As soon as s^x has been estimated, the corresponding initial state can be determined as:

$$s^0 = (I_M + T^N)^{-1} \cdot s^x, \quad (13)$$

subject to the constraint that N is not a multiple of p, with p period of T defined as the smallest integer such that $T^p = I_M$. Indeed, if $T^p = I_M$, the right term in (13) is always equal to 0 for any s^x, so (13) cannot be not satisfied for $s^x \neq 0$.
By closing the trellis with *zero padding,* we have $s^N = 0$ and (11) becomes:

$$\sum_{j=0}^{N-1} T^{N-j-1} \cdot G_T \cdot u^j = 0_{M \times 1}. \quad (14)$$

We first have to determine the number of unknowns $u^j \in GF(2^q)$ in (14) that are required to close the trellis. We show next that it is necessary and sufficient to have M unknowns in order to close the trellis. Decompose M as:

$$M = a \cdot R + b, \quad a, b \text{ integers}, \quad 0 \le b < R, \quad (15)$$

Then, (14) becomes:

$$\left[G_T \quad \dots \quad T^{a-1} \cdot G_T \quad T^a \cdot G_T\right] \cdot \begin{bmatrix} u^{N-1} \\ \dots \\ u^{N-a} \\ u^{N-a-1} \end{bmatrix} = \quad (16)$$

$$s^{N-a-1} = \sum_{j=0}^{N-a-2} T^{N-j-1} \cdot G_T \cdot u^j.$$

Since the matrices $T^k \cdot G_T$, $0 \le k < p$, are supposed to be full column rank or full row rank, we select a set of indices $\{i_1, i_2, \dots i_b\} \subset \{0, 1, \dots R\}$ such that the corresponding truncated matrix $(T^a \cdot G_T)^b$ formed by the columns $\{i_1, i_2, \dots i_b\}$ of the matrix $T^a \cdot G_T$ is full rank. Taking into account that the matrix T is periodic of period $p > a$, the matrix $A = [G_T \dots T^{a-1} \cdot G_T \quad (T^a \cdot G_T)^b]$ is invertible. Therefore, we have to introduce $a+1$ redundant symbols u^j in order to close the trellis, i.e.:

$$\begin{bmatrix} u^{N-1} \\ \dots \\ u^{N-a} \\ \left(u^{N-a-1}\right)^b \end{bmatrix} = A^{-1} \cdot s^{N-a-1} + (T^a \cdot G_T)^{R-b} \cdot (u^{N-a-1})^{R-b}, \quad (17)$$

where the partial word $(u^{N-a-1})^b$ corresponds to the components $\{i_1, i_2, \dots i_b\}$ of the word u^{N-a-1}. $(u^{N-a-1})^{R-b}$ is the partial word of u^{N-a-1} built from the $(R-b)$ remaining components. $(u^{N-a-1})^{R-b}$ can be set to zero for simplicity or can be dedicated to some information data, for a slightly higher encoding rate.

In this section, we investigated three different manners to perform the decoding of a codeword. We also show that the trellis of the MNBTC can be terminated in two different ways (circular closure and zero padding) as for the classical TC. We determine for both cases, the conditions on the codeword that ensure the trellis closure. The circular closure (tail biting) is more efficient since it provides a higher coding rate than the closure by zero padding. Nevertheless, the closure with zero padding offers better performance, as we will show in section V.
In the next section, we show that the MNBTCs include a new family of TC with Reed-Solomon (RS) as component codes with very interesting properties.

IV. REED SOLOMON TURBO-CODE

We showed in the previous section that the MNBTC include the NBTC which include themselves the classical TC. In this section, we show that this new family also includes a very interesting family that we call RS-TC since the constituent codes are RS codes.

Proposition 1: A MNB code is equivalent to a RS code if the following constraints hold:
1) the polynomial $g_0(D)$ corresponds to the generator polynomial of the targeted RS code [11]. Implicitly, it results M=grad(g_0);
2) The encoder has a single input: R = 1 and $g_1(D)$ = 1. Thus, each word corresponds to a single symbol;
3) the trellis is closed at zero. The redundancy is determined only by the symbols that help to close the trellis in (17) [11].
4) the length of the sequence of symbols which is equal to $N = 2^q$-1 should match the RS word length.
The turbo encoder will generate $2 \cdot M$ redundant words (M from each constituent encoder) from the k=N-M input symbols. If the four conditions above hold, the RS turbo codes can be viewed as a special case of the MNBTC codes.

V. SIMULATIONS

Bit error rate and frame error rate performance of the MNBTC of memory 3 with two inputs defined by generator matrix G=[2 1 2; 3 2 2; 0 0 2; 3 1 1] are presented with respect to the signal-to-noise ratio (SNR) in Fig.4. The data block is composed of N=376 words, each word of two symbols and each symbol of two bits. The decoding algorithm that we used is the approximation MaxLogMAP of the word-wise MAP algorithm described in Section III.
The intersymbol interleaving is realized with an S interleaver, with S=21 [13]. A second interleaver is used to permute the symbols between words with an even index.
We consider an AWGN transmission channel with BPSK signaling. The trellis termination is realized with zero padding as discussed in Section III. A stopping criterion based on a threshold for the APP values is applied. The maximum number of decoding iterations is set to 15.
The performance presented in Fig.4 is slightly worse than to the 4-memory MBTC proposed in [12]. It may come from the fact that we optimized neither the component codes nor the interleavers in this study. However, the MNBTC have several intrinsic advantages: deeper slop (we conjecture that MNBTC have a larger minimum distance than the NBTC), very low error-floor even for moderate codeword length (for a FER lower than 10^{-4}) and faster convergence speed. Indeed, At equivalent SNR, the average number of iteration is smaller for the MNBTC than for the BTC proposed in [11] with also memory of 4).

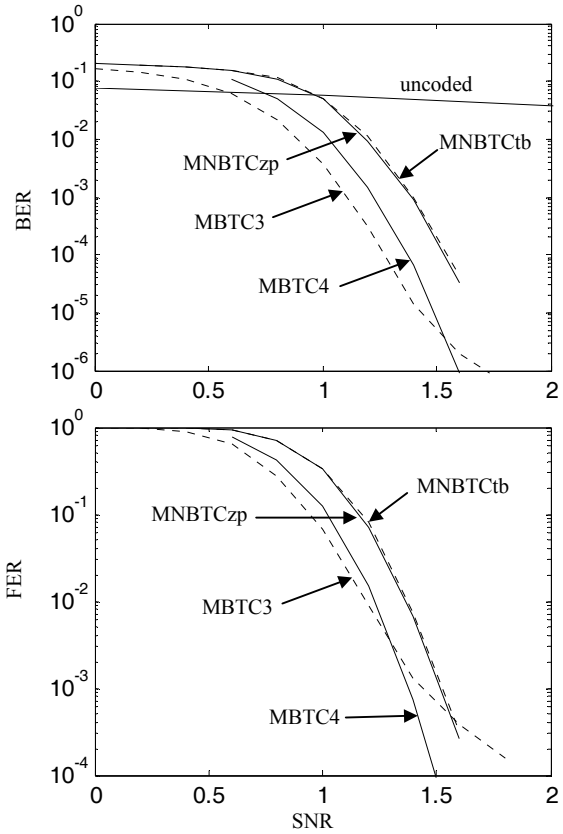


Fig 4. Comparison between the Multi Non Binary Turbo Code with encoding matrix G=[2 1 2; 3 2 2; 0 0 2; 3 1 1] (tb: tail biting, zp: zero padding) and the Multi Binary Turbo Codes (3 and 4 memory) proposed in [12] over AWGN transmission channel: Bit Error Rate (BER) and Frame Error Rate (FER) are plotted as functions of the signal-to-noise ratio (SNR).

VI. CONCLUSIONS

We introduced in this paper a new class of Turbo-codes referred as the Multiple Input Non Binary Turbo Codes where each constituent code is a non binary code with multiple inputs. We showed that this family is very general and includes all existing TCs. Particularly, it includes the classical TC, the recently proposed NBTC [12] but also a new class of turbo-codes with Reed Solomon code as constituent code. Although we did not optimize the component codes, our codes have similar performance than the codes

presented in [12]. We expect significant gain by optimizing the interleaver and the components codes.

REFERENCES

[1] C. Berrou, A. Glavieux, P. Thitimajshima, „Near Shannon Limit Error –Correcting Coding and Decoding: Turbo –Codes", Proc. of ICC, Geneve, may 1993, pp. 1064-1070.

[2] S. Dolinar, D. Divsalar, "Weight Distributions for Turbo Codes Using Random and Nonrandom Permutations", TDA Progress Report 42-122, August15, 1995.

[3] Oscar Y. Takeshita, and Daniel J. Costello. Jr., "New Classes of Algebraic Interleavers for Turbo-Codes", ISIT 1998, Cambridge, MA, USA, August 16-21.

[4] C. Berrou, Y. Saouter, C. Douillard, S. Kerouédan and M. Jézéquel, "Designing good permutations for turbo codes: towards a single model", International Conference on Communications, ICC'2004, Paris, France, June 2004, Vol. 1, pp.341-345.

[5] C. Weiss, C. Bettstetter, S. Riedel, and D.J. Costello, "Turbo decoding with tailbiting trellises", in Proc. IEEE Int. Symp.Signals, Syst., Electron., Pisa, Italy, Oct. 1998, pp. 343-348.

[6] [DIP] D.Divsalar, F.Pollara, "Serial and Hybrid Concatenated Codes with Applications", International Symposiom on Turbo Codes – Brest – France, 1997, pag. 80-87.

[7] J. Hagenauer, P. Hoeher, "A Viterbi algorithm with soft-decision outputs and its applications" ‚Proc. Of GLOBECOM'89, Dallas, Texas, 47.1.1-47.1.7, 1989.

[8] P. Robertson, P. Hoeher and E. Villebrun, "Optimal and suboptimal maximum a posteriori algorithms suitable for turbo decoding", European Trans. Telecommun., vol. 8, pp. 119-125, Mar-Apr. 1997.

[9] C. Berrou and M. Jezequel, „Nonbinary convolutional codes for turbo coding", Electron. Lett., vol. 35, no. 1, pp. 39-40, Jan. 1999.

[10] C. Berrou, M. Jezequel, C. Douillard, and S. Kerouedan, „The advantages of nonbinary turbo codes", in Proc. Inf. Theory Workshop, Caims, Australia, Sept. 2001, pp.61-63.

[11] I. Reed, and G. Solomon, "Polynomial Codes over Certain Finite Fields" Journal of the Society for Industrial and Applied Mathematics, Vol.8, No.2, Jun. 1960, pp.300-304

[12] C. Douillard, and C. Berrou, "Turbo Codes With Rate-m/(m+1) Constituent Convolutional Codes, IEEE Trans. Comm., vol.53, No. 10, October, 2005, pp.1630-1638.

[13] S. Dolinar, and D. Divsalar, „Weight Distributions for Turbo Codes Using Random and Nonrandom Permutations", TDA Progress Report 42-122, August 15, 1995

Buletinul Ştiinţific al Universităţii "Politehnica" din Timişoara

Seria ELECTRONICĂ şi TELECOMUNICAŢII
TRANSACTIONS on ELECTRONICS and COMMUNICATIONS

Tom 51(65), Fascicola 2, 2006

On Semantic Feature of Information

Valeriu Munteanu[1], Daniela Tarniceriu[1]

Abstract – Beside the objective or quantitative characteristic of a message, appraised by the probability with which it is supplied, its semantic or qualitative characteristic, appraised by a certain utility or importance is, additionally, considered. In this paper we determine the quantitative – qualitative information, the quantitative – qualitative entropy of a discrete, complete and memoryless source as well as the main properties of the quantitative – qualitative entropy.
Keywords: semantic sources, entropies.

I. INTRODUCTION

In the elaboration of the information concept in classical sense, only its quantitative feature is considered, the semantic (qualitative) one being neglected [1, 2, 3]. In this case only the probabilities of random events are considered for computing the information. In cybernetic systems the transmitted information is used for a certain goal. Considering only the probabilistic dependencies between the transmitted messages is not enough, because the transmission efficiency also depends on the choosing of those messages that serve to the pursued goal. This is the reason why, in a cybernetic system, also the quality of the transmitted messages should be measured. A semantic source is characterized both quantitatively and qualitatively. Thus, the transmitted information will depend both on the objective part of the experiment through the event probabilities and on its utility or importance, that reflects the subjective part of the experiment related to a certain goal. Identical events, with same probabilities can have different utilities for different cybernetic systems, even if their goals are the same. On the other hand, for the same cybernetic system, the same events can have different utilities, when the goal is changed. Therefore, the utilities of different events are related both to the proposed goal and to the cybernetic system used to its achievement. In [4-8] authors have studied generalized coding theorems by considering different generalized measures. A first attempt to measure information both quantitatively and qualitatively is given in [9]. In [10] an introduction of the quantitative - qualitative information is presented. In this paper a fully presentation of semantic sources is performed, by deriving the quantitative – qualitative information, its average value for a discrete, complete and memoryless source, emphasizing the main properties of semantic sources.

II. DETERMINING THE QUANTITATIVE – QUALITATIVE INFORMATION

Let S be a discrete, complete and memoryless source characterized by the distribution:

$$S:\begin{pmatrix} s_1 & s_2 & \cdots & s_n \\ p_1 & p_2 & \cdots & p_n \\ u_1 & u_2 & \cdots & u_n \end{pmatrix} \tag{1}$$

where

- s_i represents the source messages, that is, signals corresponding to ideas, images, data which has to be transmitted to a correspondent;
- p_i denotes the probabilities with which the source delivers its messages, so that

$$0 \le p_i \le 1, i = 1,...,n, \tag{2}$$

$$\sum_{i=1}^{n} p_i = 1 \tag{3}$$

- u_i denotes the utility or importance of the message s_i. This utility is appraised by a positive real number which reflects the semantic characteristics of the message as function of the given goal and of the system.

Theorem 1

The quantitative – qualitative information or the semantic information attached to the message s_k, denoted by $i_{pu}(s_k)$, is computed by

$$i_{pu}(s_k) = b\log_\alpha p_k + au_k, \tag{4}$$

where $\alpha \in \{0,1\} \cup \{1,\infty\};\ a,b,u_i \in R$.

Proof

The quantitative – qualitative information attached to the message s_i will be a function F, of the probability p_i and the utility u_i, respectively. To derive this function, two messages, independent both probabilistic and logic – causal, s_i and s_j, are

[1] Facultatea de Electronică şi Telecomunicaţii, Departamentul Comunicaţii Bd. Carol I, nr.11, 700506 Iasi, e-mail vmuntean@etc.tuiasi.ro

considered. Their probabilities are p_i and p_j, respectively and the utilities u_i and u_j, respectively.

If

$$\sigma_k = s_i s_j \ , \tag{5}$$

$$p(\sigma_k) = p_i p_j \tag{6}$$

$$u(\sigma_k) = u_i + u_j \tag{7}$$

The quantitative – qualitative information the message σ_k can deliver is, generally, a function of the probability $p(\sigma_k)$ and the utility $u(\sigma_k)$, as

$$i_{pu}(\sigma_k) = F[p(\sigma_k), u(\sigma_k)] \tag{8}$$

where F is a function to be found out.

We also assume that the quantitative – qualitative information given by two events independent both statistical and causal is equal to the sum of two pieces of information attached to each of them. Therefore,

$$i_{pu}(\sigma_k) = i_{pu}(s_i) + i_{pu}(s_j) \tag{9}$$

Considering (8), (9) becomes

$$F[p(\sigma_k), u(\sigma_k)] = F(p_i, u_i) + F(p_j, u_j) \tag{10}$$

with

$$0 \le p_i \le 1; 0 \le p_j \le 1; u_i, u_j \in R \tag{11}$$

In order to determine the solution of the functional equation (10), the following functions are defined

$$z_i = \log_\alpha p_i, z_i \in R, \tag{12}$$

$$z_j = \log_\alpha p_j, z_j \in R \tag{13}$$

where $\alpha \in \{0,1\} \cup \{1,\infty\}$.

Using (12) and (13), (10) becomes

$$F(\alpha^{z_i+z_j}, u_i + u_j) = F(\alpha^{z_i}, u_i) + F(\alpha^{z_j}, u_j) \tag{14}$$

Denoting

$$G(z,u) = F(\alpha^z, u) \tag{15}$$

equation (14) can be written as

$$G(z_i + z_j, u_i + u_j) = G(z_i, u_i) + G(z_j, u_j) \tag{16}$$

For $z_i = z_j = 0$, we have

$$G(0, u_i + u_j) = G(0, u_i) + G(0, u_j) \tag{17}$$

Denoting

$$f(u) = G(0, u) \tag{18}$$

equation (17) becomes

$$f(u_i + u_j) = f(u_i) + f(u_j). \tag{19}$$

The solution of this functional equation is given in [11], provided the function $f(u)$ is continuous at least in one point, being of the form

$$f(u) = a \cdot u, (\forall) u \in R, a \in R. \tag{20}$$

For $u_i = u_j = 0$, from (16) we obtain

$$G(z_i + z_j, 0) = G(z_i, 0) + G(z_j, 0) \tag{21}$$

Denoting by

$$g(z) = G(z, 0) \tag{22}$$

equation (17) becomes

$$g(z_i + z_j) = f(z_i) + f(z_j). \tag{23}$$

Its solution is

$$g(z) = b \cdot z, (\forall) z \in R, b \in R. \tag{24}$$

The functional equation (24) admits a solution, if $g(z)$ is continuous at least in one point.

For $u_i = z_j = 0$, from (16) we have

$$G(z_i, u_j) = G(z_i, 0) + G(0, u_j) \tag{25}$$

Making use of (18) and (22), we can write

$$G(z,u) = g(z) + f(u) \tag{26}$$

With (20) and (24), (26) becomes

$$G(z,u) = b \cdot z + a \cdot u, \quad (\forall) z \in R, (\forall) u \in R, a \in R, b \in R \tag{27}$$

or, considering (12), (13) and (15), we get the solution of the functional equation (21), as

$$F(p,u) = b \cdot \log_\alpha p + a \cdot u \tag{28}$$

According to (8) and (28), the quantitative - qualitative information attached to the message s_k, with the utility u_k and delivered with the probability p_k, can be computed by (4).

The neglect of the qualitative characteristic of information consists in removing the second term in the right hand of (4). In this way, we get the classical calculus relation for the quantitative information attached to a message [1, 2, 3].

III. DETERMINING THE AVERAGE QUANTITATIVE – QUALITATIVE INFORMATION FOR A SEMANTIC MEMORYLESS SOURCE

Let S be a discrete, complete and memoryless source characterized by the distribution given in (1).

Theorem 2

The average quantitative – qualitative information is computed by

$$H_{pu}(S) = b\sum_{k=1}^{n} p_k \log_\alpha p_k + a\sum_{k=1}^{n} p_k u_k \tag{29}$$

Proof

The quantitative – qualitative information $i_{pu}(s_k)$ defined in (4) determines a discrete random variable, which takes on values with probabilities $p_k, k = 1,2,...,n$. The average value of this information, called quantitative – qualitative entropy or semantic entropy of the source S and denoted by $H_{pu}(S)$, can be computed by

$$H_{pu}(S) = \sum_{k=1}^{n} p_k i_{pu}(s_k) \tag{30}$$

or, considering (4), we obtain (29).

If $u_1 = u_2 = ... = u_n = 0$, that is, the qualitative characteristic is neglected, (29) becomes

$$H_{pu}(S) = b\sum_{k=1}^{n} p_k \log_\alpha p_k \tag{31}$$

In order to obtain the entropy defined by Shannon [2], we set

$$b = -1, \alpha = 2. \tag{32}$$

With (32), (29) becomes

$$H_{pu}(S) = -\sum_{k=1}^{n} p_k \log_2 p_k + a\sum_{k=1}^{n} p_k u_k . \quad (33)$$

If the total utility is represented by U, that is

$$\sum_{k=1}^{n} u_k = U , \quad (34)$$

then, for $p_1 = p_2 = ... = p_n = \frac{1}{n}$, we can write

$$a\sum_{k=1}^{n} p_k u_k = \frac{aU}{n} . \quad (35)$$

But $\frac{aU}{n}$ measures the average utility per message for equally likely probabilities, which is equal to $\frac{U}{n}$. It follows that a has to be equal to unity.

In the sequel the quantitative – qualitative entropy or the semantic entropy will be computed by

$$H_{pu}(S) = -\sum_{k=1}^{n} p_k \log_2 p_k + \sum_{k=1}^{n} p_k u_k \quad (36)$$

and the quantitative - qualitative information or semantic information by

$$i_{pu}(s_k) = -\log_2 p_k + u_k . \quad (37)$$

IV. MAIN PROPERTIES OF THE QUANTITATIVE – QUALITATIVE ENTROPY

Property 1: If $u_k \in R^+$, the quantitative – qualitative entropy is nonnegative, i.e.

$$H_{pu}(S) \geq 0 \quad (38)$$

Property 2: If $p_k = 1, u_k = 0;\ 1 \leq k \leq n$, then

$$H_{pu}(S) = 0 \quad (39)$$

Property 3: The maximum value of the entropy with respect to the probabilities p_k, for given utilities, is:

$$\max_{p_k} H_{pu}(S) = \log_2\left(\sum_{k=1}^{n} 2^{u_k}\right) \quad (40)$$

This maximum entropy will be denoted by $H_m(S)$.

Proof

Let

$$\phi(p_1, p_2, ..., p_n; u_1, u_2, ..., u_n) = -\sum_{k=1}^{n} p_k \log_2 p_k + \sum_{k=1}^{n} p_k u_k + \lambda(\sum_{k=1}^{n} p_k - 1) \quad (41)$$

where λ is a real positive number (the Lagrange multiplier).

The extreme of the function ϕ with respect to p_k for u_k fixed, coincides with the extreme of the function $H_{pu}(S)$. The necessary condition of extreme is given by the system

$$\frac{\partial\phi(p_1, p_2, ..., p_n; u_1, u_2, ..., u_n)}{\partial p_k} = 0, 1 \leq k \leq n \quad (42)$$

or, equivalently, taking into account (41),

$$-\log_2 p_k - \log_2 e + u_k + \lambda = 0, 1 \leq k \leq n , \quad (43)$$

from where

$$p_k = 2^{\lambda + u_k - \log_2 e} \quad (44)$$

Since

$$\sum_{k=1}^{n} p_k = 1 \Rightarrow \lambda = \log_2 e - \log_2(\sum_{k=1}^{n} 2^{u_k}) . \quad (45)$$

Substituting (45) into (44), we have

$$p_k = \frac{2^{u_k}}{\sum_{k=1}^{n} 2^{u_k}}, k = 1, 2, ..., n . \quad (46)$$

Finally, substituting (46) into (36), relation (40) follows. It is easy to prove that this extreme is a maximum one.

Property 4

The absolute maximum value of entropy, denoted by $H_{ma}(S)$ is given by

$$H_{ma}(S) = \log_2 n + \frac{U}{n} \quad (47)$$

Proof

We want to find the utilities the messages s_k have to possess, so that

$$\left(\sum_{k=1}^{n} 2^{u_k}\right) = \max , \quad (48)$$

under the constraint of (34).

If (48) is satisfied, then the entropy computed by (40) becomes an absolute maximum one.

In order to determine the utilities u_k for which (48) is satisfied, the Lagrange multipliers method is used. The following function will be constructed:

$$\Psi(u_k) = \sum_{k=1}^{n} 2^{u_k} + \lambda\left(U - \sum_{k=1}^{n} u_k\right), \lambda > 0 , \quad (49)$$

where λ is the Lagrange multiplier. The maximum value of $\Psi(u_k)$ is attained simultaneously with the maximum value of (48).

The necessary condition for extreme is

$$\frac{\partial\Psi(u_k)}{\partial u_k} = 0, k = 1, ..., n \quad (50)$$

Substituting (49) into (50), we have

$$u_k = \log_2\left(\frac{\lambda}{\ln 2}\right), k = 1, 2, ..., n \quad (51)$$

Since

$$\sum_{k=1}^{n} u_k = U \quad (52)$$

it results

$$\lambda = 2^{U/n} \ln 2 \quad (53)$$

Substituting (53) into (51), we get

$$u_k = \frac{U}{n}, k = 1, 2, ..., n \quad (54)$$

We can easily prove that the so obtained extreme value is a maximum one, because

$$\frac{\partial^2\Psi(u_k)}{\partial u_k^2} > 0, \frac{\partial^2\Psi(u_k)}{\partial u_k \partial u_i} = 0,\ i, k = 1, ..., n, i \neq k . \quad (55)$$

According to (54), the quantitative – qualitative entropy is maximized when all source messages have

identical qualitative weights (the same utilities), equal to the arithmetic mean value of the utilities.

Substituting (54) into (46), we get

$$p_k = \frac{1}{n}, k = 1,...,n \ . \qquad (56)$$

Substituting (54) and (56) into (36), (47) results.

V. CONCLUSIONS

In this paper, besides the objective or quantitative characteristic of a message, appraised by the probability with which it is supplied, its semantic or qualitative characteristic, appraised by a certain utility is, additionally, considered. The quantitative – qualitative information attached to a message (eq. 4), as well as the quantitative – qualitative entropies of a discrete, complete and memoryless source (eq. 29) are derived. The maximum value of the entropy of a semantic source is determined with respect to the probabilities the messages are delivered. The constraints on the utilities and probabilities for obtaining the absolute maximum entropy of a semantic source are inferred. These relations represent generalizations of the classical concepts on information. The quantitative – qualitative information results as the sum between a quantitative information and a qualitative one. If the qualitative characteristic is neglected, by dropping out the second term in relations (4) and (29), the classical known relations [2] are obtained. When only the qualitative characteristic is required, the first term in the relations above is dropped out. The main properties of entropy for semantic sources are also established. The semantic entropy defined in this paper preserves the properties of the classical entropy defined by Shannon.

REFERENCES

[1] G. R.Gallager, *Information theory and reliable communications*, New York, John Wiley and Sons Inc., 1968.

[2] C. E.Shannon, "A mathematical theory of communication", BSTJ, 27 1948, pp. 379-423, 623-656.

[3] T.Cover, J. A.Thomas, *Elements of Information Theory*, Wiley, 1991.

[4] G. Longo, "A noisless coding theorem for sources having utilities", *SIAM J. Appl. Math*., 30 (4), 1976, pp. 739-748.

[5] A.Gurdial, F. Pessoa, "On useful information of order "α . *J Comb. Information and Syst. Sci*., 2, 1977, pp. 158-162.

[6] A. B. Khan, R. Autar, "On useful information of order α and β ", *Soochow J. Math.*, 5, 1979, pp. 93-99.

[7] R. Autar, A. B. Khan, "On generalized useful information for incomplete distribution", *J. of Comb. Information and Syst. Sci*., 14 (4), 1989, pp. 187-191.

[8] A. B. Khan, B. A.Bhat, S. Pirzada, "Some results on a generalized useful information measure", *J. Inequal. Pure and Appl. Math*, 6 (4), 2005, pp. 1-5.

[9] M.Belis, S. Guiasu, "A quantitative-qualitative measure of information in cybernetics", *IEEE Trans. Inf. Theory* IT – 14, 1968, pp. 593-594.

[10] V. Munteanu, P. Cotae, "The entropy with preference", *Int. J. Electronics and Comm. AEU*, 46, 1992, pp. 429 – 431.

[11] G. M. Fihtengol'c, *Differential and integral calculus*, Ed. Tehnica, Bucharest, 1964.

Buletinul Ştiinţific al Universităţii "Politehnica" din Timişoara

Seria ELECTRONICĂ şi TELECOMUNICĂŢII

TRANSACTIONS on ELECTRONICS and COMMUNICATIONS

Tom 51(65), Fascicola 2, 2006

On the Asymptotic Decorrelation of the Wavelet Packet Coefficients of a Wide-Sense Stationary Random Process

Abdourrahmane M. Atto [1], Dominique Pastor [2], Alexandru Isar [3]

***Abstract*— Consider the wavelet packet coefficients issued from the decomposition of a random process stationary in the wide-sense. We address the asymptotic behaviour of the autocorrelation of these wavelet packet coefficients. In a first step, we explain why this analysis is more intricate than that already achieved by several authors in the case of the standard discrete orthonormal wavelet decomposition. In a second step, it is shown that the autocorrelation of the wavelet packet coefficients can be rendered arbitrarily small provided that both the decomposition level and the regularity of the quadrature mirror filters are large enough.**

I. Introduction

CONSIDER a second-order random process and assume that this random process is stationary in the wide-sense. The discrete orthonormal wavelet and the wavelet packet decompositions of this process yield coefficients that are random variables. Many authors have studied the statistical correlation of these coefficients, see [1]–[6] amongst others. In the case of discrete orthonormal wavelet transform, in-scale coefficients tend to be uncorrelated when the decomposition level increases. At first sight, it would seem quite reasonable to consider that the same property remains valid for wavelet packet coefficients. Unfortunately, the analysis of the autocorrelation of these wavelet packet coefficients is significantly more intricate than expected, mainly because of the role played by the regularity of the quadrature mirror filters. This analysis is presented below. The proofs of the several theoretical results stated hereafter are postponed to a forthcoming paper because of the limited size of the present one.

This paper in organized as follows. In section II, the reader is reminded with basic results concerning the wavelet packet decomposition of a wide-sense stationary random process. In particular, for a given decomposition level, we give the expression of the autocorrelation function of the discrete sequence formed by the wavelet packet coefficients. The asymptotic behaviour of this function is then achieved in two steps. In section III, the analysis is worked out in the case of the ideal Shannon wavelet packet decomposition, which employs ideal quadrature mirror filters [7]. Since quadrature mirror filters such as the Daubechies and Battle-Lemarié filters tend to ideal filters when their regularity increases, the asymptotic behaviour of the autocorrelation function of the wavelet packet coefficients when such filters are used derives from that described in section III. This asymptotic behaviour is stated in section IV. It depends on the wavelet packet decomposition level as well as the regularity of the filters at hand.

II. The wavelet packet decomposition of a wide-sense stationary random process

A. Wavelet packet decomposition

Let m_0 and m_1 be the Fourier transform of two quadrature mirror filters such that

$$m_0(\omega) = \frac{1}{\sqrt{2}} \sum_{\ell \in \mathbb{Z}} h_0[\ell] e^{-i\ell\omega}, \tag{1}$$

and

$$m_1(\omega) = \frac{1}{\sqrt{2}} \sum_{\ell \in \mathbb{Z}} h_1[\ell] e^{-i\ell\omega}. \tag{2}$$

Let Φ be the scaling function associated to m_0. We define the sequence $(W_n)_{n\geq 0}$ of elements of $L^2(\mathbb{R})$ by recursively setting

$$W_{2n}(t) = \sqrt{2} \sum_{\ell \in \mathbb{Z}} h_0[\ell] W_n(2t - \ell) \tag{3}$$

and

$$W_{2n+1}(t) = \sqrt{2} \sum_{\ell \in \mathbb{Z}} h_1[\ell] W_n(2t - \ell) \tag{4}$$

with $W_0 = \Phi$. We then have $W_1 = \Psi$, where Ψ is the wavelet function associated to the quadrature mirror filters under consideration. If we now put

$$W_{j,n}(t) = 2^{-j/2} W_n(2^{-j}t), \tag{5}$$

and

$$W_{j,n,k}(t) = \tau_{2^j k} W_{j,n}(t) = 2^{-j/2} W_n(2^{-j}t - k), \tag{6}$$

[1] ENST Bretagne, *am.atto@enst-bretagne.fr*
[2] ENST Bretagne, *dominique.pastor@enst-bretagne.fr*
[3] Univ. 'Politehnica' din Timisoara, *alexandru.isar@etc.upt.ro*

the set $\{W_{j,n,k} : k \in \mathbb{Z}\}$ of *wavelet packets* is an orthonormal system of vectors of the Hilbert space $L^2(\mathbb{R})$. With a slight abuse of language, the vector space $\mathbf{W}_{j,n}$ generated by $\{W_{j,n,k} : k \in \mathbb{Z}\}$ will hereafter be called the *packet* $\mathbf{W}_{j,n}$. For every $j = 0, 1, 2, \cdots$, and every $n \in I_j = \{0, 1, \cdots, 2^j - 1\}$, the wavelet packet decomposition of the function space $\mathbf{W}_{0,0}$ is obtained by recursively applying the so-called *splitting lemma* [8] to every space $\mathbf{W}_{j,n}$. We thus can write that

$$\mathbf{W}_{j,n} = \mathbf{W}_{j+1,2n} \oplus \mathbf{W}_{j+1,2n+1}. \tag{7}$$

The sets $\{W_{j+1,2n,k} : k \in \mathbb{Z}\}$ and $\{W_{j+1,2n+1,k} : k \in \mathbb{Z}\}$ are orthonormal bases of the vector spaces $\mathbf{W}_{j+1,2n}$ and $\mathbf{W}_{j+1,2n+1}$, respectively. The decomposition tree of figure 1 illustrates such a decomposition

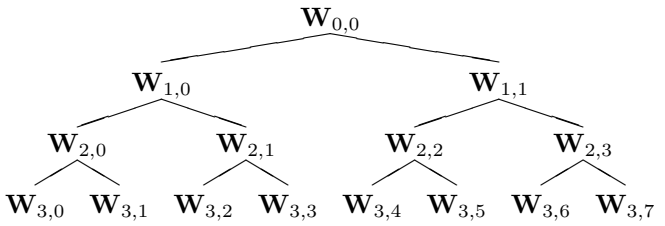


Fig. 1. Wavelet packet decomposition tree down to decomposition level $j = 3$.

Remark 1: given $j \in \mathbb{N}$, consider a binary sequence $(\epsilon_\ell)_{\ell \in \{1,2,\ldots,j\}}$ of $\{0,1\}^j$. Basically, this sequence corresponds to the sequence $m_{\epsilon_1}, m_{\epsilon_2}, \ldots, m_{\epsilon_j}$ of filters successively applied to calculate the coefficients of the packet $\mathbf{W}_{j,n}$ where

$$n = \sum_{\ell=1}^{j} \epsilon_\ell 2^{j-\ell}. \tag{8}$$

Readily, n is an element of I_j and the sequence $(\epsilon_\ell)_{\ell \in \{1,2,\ldots,j\}}$ is the unique path issued from $\mathbf{W}_{0,0}$ that leads to $\mathbf{W}_{j,n}$ in the wavelet packet decomposition tree. Conversely, let $n \in I_j$. There exists a unique sequence $(\epsilon_\ell)_{\ell \in \{1,2,\ldots,j\}}$ of $\{0,1\}^j$ such that equation (8) holds true. In the sequel, when a natural number n and a binary sequence $(\epsilon_\ell)_{\ell \in \{1,2,\ldots,j\}}$ of $\{0,1\}^j$ satisfy (8), we will say that n and $(\epsilon_\ell)_{\ell \in \{1,2,\ldots,j\}}$ are associated to each other.

Proposition 1: Consider a function W_n (defined from the recurrence (3) and (4)) where n has the form (8) for some $j > 0$. The Fourier transform $\hat{W}_n$ of W_n is given, for every real value ω, by

$$\hat{W}_n(\omega) = \left[\prod_{\ell=1}^{j} m_{\epsilon_\ell}\left(\frac{\omega}{2^{j+1-\ell}}\right)\right] \hat{W}_0\left(\frac{\omega}{2^j}\right), \tag{9}$$

where $(\epsilon_\ell)_{\ell \in \{1,2,\ldots,j\}}$ is the binary sequence associated to n.

B. The autocorrelation function of the wavelet packet coefficients of a wide-sense stationary random process

Let $X : \mathbb{R} \times \Omega \longrightarrow \mathbb{R}$ be a second-order, centred and wide-sense stationary random process where Ω is some probability space. The autocorrelation function of this random process is denoted by $R_X(t,s) = \mathbb{E}[X(t)X(s)] = R_X(t-s)$. We assume that X is continuous in quadratic mean. Then, R_X is a continuous function. We also assume that X has a power spectral density γ_X, which is the Fourier transform of R_X.

Given a decomposition level j and $n \in I_j$, the wavelet packet decomposition of X returns, at node (j,n), the random variables

$$c_{j,n}[k] = \int_{\mathbb{R}} X(t) W_{j,n,k}(t) dt, \quad k \in \mathbb{Z}, \tag{10}$$

provided that the integral

$$\iint_{\mathbb{R}^2} R_X(t,s) W_{j,n,k}(t) W_{j,n,k}(s) dt ds$$

exists [2].

Let $R_{c_{j,n}}$ stand for the autocorrelation function of the discrete processus $c_{j,n}$ defined by (10). It can be shown that, for every $m \in \mathbb{Z}$

$$R_{c_{j,n}}[m] = \frac{1}{2\pi} \int_{\mathbb{R}} \gamma_X\left(\frac{\omega}{2^j}\right) |\hat{W}_n(\omega)|^2 e^{im\omega} d\omega. \tag{11}$$

Our purpose is then to analyse the behaviour of this function for large values of j. Since $n \in I_j$, the analysis must take this dependence into account. If n is constant with j, Lebesgue's dominated convergence theorem can be used to compute the limit of $R_{c_{j,n}}[m]$ when j grows to infinity. If $n = 0$, the result thus obtained is that given in [4], [5]. The situation becomes more intricate if n is a function of j. For instance, if we choose $n = 2^{j-L}$ where $L \in \{1, \ldots, j-1\}$, the behaviour of $R_{c_{j,n}}[m]$ when j grows to infinity is no longer a straightforward consequence of Lebesgue's dominated convergence theorem.

The approach proposed below embraces these several cases by considering the binary sequence associated to a node (j,n) of the decomposition tree. By so proceeding, the crucial role played by the regularity of the quadrature mirror filters is enhanced.

III. Asymptotic behaviour of the wavelet packet coefficients for the Shannon wavelet packet decomposition

The Shannon wavelet packet decomposition corresponds to the case where the scaling function Φ is $\Phi^S = sinc$. The quadrature mirror filters of this decomposition are the ideal low and high pass filters $m_0^S(\omega) = \sqrt{2}\sum_{\ell \in \mathbb{Z}} \chi_{\Delta_0}(\omega - 2\pi\ell)$ and $m_1^S(\omega) = \sqrt{2}\sum_{\ell \in \mathbb{Z}} \chi_{\Delta_1}(\omega - 2\pi\ell)$, where χ_Δ stands for the indicator function of the set Δ, $\Delta_0 = \left[-\frac{\pi}{2}, \frac{\pi}{2}\right]$, and $\Delta_1 = \left[-\pi, -\frac{\pi}{2}\right] \cup \left[\frac{\pi}{2}, \pi\right]$. The Fourier transform $\hat{W}_0^S$ of the scaling function is then $\hat{W}_0^S = \hat{\Phi}^S = \chi_{[-\pi,\pi]}$.

According to Coifman et Wickerhauser ([9], [7, pp. 326-327]), for every $j > 0$ and every $n \in I_j$, there exists a unique $p = G[n] \in I_j$ such that $|\hat{W}_{j,n}^S(\omega)| = 2^{j/2}\chi_{\Delta_{j,p}}(\omega)$, where W_n^S stands for the map W_n recursively defined by (3, 4) when the pair of quadrature mirror filters is (m_0^S, m_1^S) and

$$\Delta_{j,p} = \left[-\frac{(p+1)\pi}{2^j}, -\frac{p\pi}{2^j}\right] \cup \left[\frac{p\pi}{2^j}, \frac{(p+1)\pi}{2^j}\right]. \tag{12}$$

The map G permutes the elements of I_j and we can prove that

$$G[2n+\epsilon] = 3G[n] + \epsilon - 2\left\lfloor \frac{G[n]+\epsilon}{2} \right\rfloor, \tag{13}$$

where $\epsilon \in \{0,1\}$ and $\lfloor z \rfloor$ is the largest integer less than or equal to z.

With the same notations as those introduced above, we define, for every natural number j,

$$\gamma_j(\omega) = \sum_{\ell=0}^{2^j-1} \gamma_X(\frac{\ell\pi}{2^j})\ \chi_{\Delta_{j,\ell}}(\omega), \tag{14}$$

where $\Delta_{j,\ell}$ is defined according to (12). We then have the following result.

Proposition 2: Let j be some natural number and n be any element of I_j. We have that

$$\frac{1}{2\pi}\int_{\mathbb{R}} \gamma_j(\frac{\omega}{2^j})|\hat{W}_n^S(\omega)|^2 e^{im\omega}d\omega = \gamma_X(\frac{p\pi}{2^j})\delta[m], \tag{15}$$

where $\delta[m] = 1$ if $m = 0$ and $\delta[m] = 0$ otherwise, $p = G[n]$ and G is given by (13).

In what follows, given an arbitrary infinite binary sequence $\kappa = (\epsilon_k)_k \in \{0,1\}^{\mathbb{N}}$ and any integer j, n_j will stand for the natural number associated to the finite subsequence $(\epsilon_k)_{k=1,2,\ldots,j}$ and we set $p_j = G[n_j]$.

It is easy to see that the sequences $(\frac{n_j\pi}{2^j})_j$ and $(\frac{p_j\pi}{2^j})_j$ are Cauchy. As such, each of them has a unique limit. In particular, the limit

$$a(\kappa) = \lim_{j\to+\infty} \frac{p_j\pi}{2^j}, \tag{16}$$

will play a crucial role in the sequel. Table I displays the value of $a(\kappa)$ for the sequences κ that were employed to carry out the experiments whose results are given in section V. These sequences are $\kappa_1 = (0,0,0,0,0,\cdots), \kappa_2 = (0,0,1,0,0,\cdots), \kappa_3 = (0,1,0,0,0,\cdots)$, and $\kappa_4 = (1,0,0,0,0,\cdots)$. The sequences $(n_j)_j$ and $(p_j)_j$ corresponding to these sequences are given in table I.

TABLE I

VALUE OF $a(\kappa) = \lim_{j\to+\infty} \frac{p_j\pi}{2^j}$ WITH RESPECT TO $(n_j)_j$

Sequence	κ_1	κ_2	κ_3	κ_4
n_j for $j \geq 3$	0	2^{j-3}	2^{j-2}	2^{j-1}
p_j for $j \geq 3$	0	$2^{j-2}-1$	$2^{j-1}-1$	2^j-1
$a(\kappa)$	0	$\frac{\pi}{4}$	$\frac{\pi}{2}$	π

Theorem 1: Let $\kappa = (\epsilon_k)_{k\in\mathbb{N}}$ be a binary sequence of $\{0,1\}^{\mathbb{N}}$. With the notations introduced just above, consider the packets $\mathbf{W}_{j,n_j}^S$, $j \in \mathbb{N}$.

Let $R_{c_{j,n_j}}^S$ be the autocorrelation function of the Shannon wavelet packet coefficients c_{j,n_j} at node (j,n_j). If $a(\kappa)$ is a continuity point of γ_X, then

$$\lim_{j\to+\infty} R_{c_{j,n_j}}^S[m] = \gamma_X(a(\kappa))\delta[m], \tag{17}$$

Remark 2: the autocorrelation function $R_{c_{j,n_j}}^S$ of the Shannon wavelet packet coefficients c_{j,n_j} at node (j,n_j) derives from (11) and is thus given by

$$R_{c_{j,n_j}}^S[m] = \frac{1}{2\pi}\int_{\mathbb{R}} \gamma_X(\frac{\omega}{2^j})|\hat{W}_{n_j}^S(\omega)|^2 e^{im\omega}d\omega. \tag{18}$$

IV. ASYMPTOTIC BEHAVIOUR OF THE AUTOCORRELATION FUNCTION OF THE WAVELET PACKETS ASSOCIATED TO A WIDE-SENSE STATIONARY PROCESS

We now consider non-ideal quadrature mirror filters m_0 et m_1. It is known that the multiplicity of the zero of m_0 in π equals the number of null moments of the analysing wavelet when m_0 is either a Daubechies or a Battle-Lemarié filter. Furthermore, the wavelet regularity increases with the number of its null moments. We then say that the regularity of a pair (m_0, m_1) of quadrature mirror filters is r if the scaling filter can be written in the form

$$m_0(\omega) = \left(\frac{1+e^{-i\omega}}{2}\right)^r Q(e^{-i\omega}). \tag{19}$$

The scaling filter has thus a zero with multiplicity r in $\omega = \pi$. This notion of regularity relates to the flatness of the filter magnitude response.

According to [10]–[12], the standard Daubechies filters converge pointwise to the ideal Shannon filters when their regularity tends to infinity. Figure 2 illustrates this convergence by displaying the magnitude response of the Daubechies scaling filters with regularity $1, 2, 4, 10, 20$, and 40.

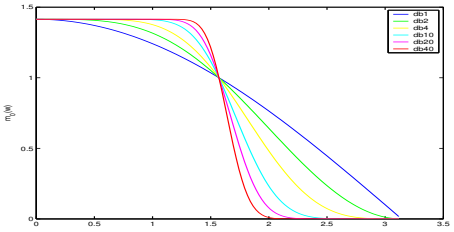


Fig. 2. Magnitude response of Daubechies scaling filters. In this figure, dbr stands for the rth-order Daubechies scaling filter.

The Battle-Lemarié filters satisfy the same property [13].

In what follows, we consider rth-order quadrature mirror filters that are denoted by $(m_\epsilon^{[r]})_{\epsilon\in\{0,1\}}$. We then can state the subsequent result where the notations introduced so far are used with the same meaning as above.

Theorem 2: Let X be a second-order random process. Assume that X is centred, stationary in the wide-sense and continuous in quadratic mean.

Assume that the power spectral density γ_X of X is bounded, with support in $[-\pi,\pi]$ and continuous at $a(\kappa)$ where κ is some binary sequence of $\{0,1\}^{\mathbb{N}}$.

For every given regularity r, the wavelet packet coefficients of $\mathbf{W}_{j,n_j}^{[r]}$ form a second-order discrete random process whose correlation function derives from (11) and is equal to

$$R_{c_{j,n_j}}^{[r]}[m] = \frac{1}{2\pi}\int_{\mathbb{R}} \gamma_X(\frac{\omega}{2^j})|\hat{W}_{n_j}^{[r]}(\omega)|^2 e^{im\omega}d\omega. \tag{20}$$

For every given positive real number $\eta > 0$, there exists an integer j_0 with the following property : for every natural number $j \geq j_0$, there exists $r_0 = r_0(j,n_j)$ such that, for every $r \geq r_0$, $|R_{c_{j,n_j}}^{[r]}[m] - \gamma_X(a(\kappa))\delta[m]| < \eta$.

V. Experimental results

A. The role played by the decomposition level

We consider a wavelet packet decomposition tree whose depth is $J = 6$. We constructed a random process as follows. The wavelet coefficients of every packet at decomposition level 6 were set to centred, independent and identically Gaussien distributed random variables. The value of the variance of these random variables was randomly chosen. By using the standard wavelet packet reconstruction algorithm, we obtained the random process whose spectral density is given by figure 3. When we decompose this random process by using the same quadrature mirror filters as those used for synthesizing it and consider the packets $\mathbf{W}_{j,n_j}$ when n_j is associated to the sequences $\kappa_1, \kappa_2, \kappa_3$, and κ_4 of table I, we note that, for $j = 3$, some coeffients c_{j,n_j} remain strongly correlated whereas some others are not (see figure 5).

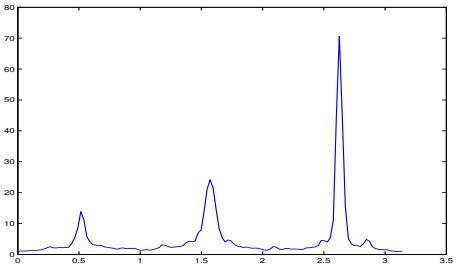


Fig. 3. Spectral density of the random process used to illustrate the role played by the decomposition level.

B. Influence of the regularity

We decompose the same random process as that used above and whose spectral density is displayed in figure 3. Now, we use the Daubechies filters with regularity 1 and 20. For the third decomposition level ($j = 3$), the results thus obtained are those of figure 6. This illustrates the role played by the regularity of the quadrature mirror filters.

C. The limit value of the correlation function

According to theorem 2, if the decomposition level and the regularity of the filters are both large enough, the correlation functions must tend to $\gamma_X(a(\kappa))\delta[m]$ where $\gamma_X(a(\kappa))$ is the value of the spectral density of the random process at $a(\kappa)$, $a(\kappa)$ is given by (16) and κ is some binary sequence.

Let us consider the random process whose spectral density is that of figure 4.

Quite rapidly, the value of the autocorrelation function at the origin becomes close to $\gamma_X(a(\kappa))$. This is pointed out by figure 7. This figure displays the autocorrelation functions obtained at the sixth level of the wavelet packet decomposition tree for the wavelet packets respectively associated to the sequences $\kappa_1, \kappa_2, \kappa_3$, and κ_4 of table I when the quadrature mirror filters are the Daubechies filters with regularity 1, 4 and 10. The same figure also pinpoints that all the wavelet packet coefficients become reasonably uncorrelated when the regularity of the filters is large enough. This illustrates also and once again the crucial role played by the regularity of the quadrature mirror filters.

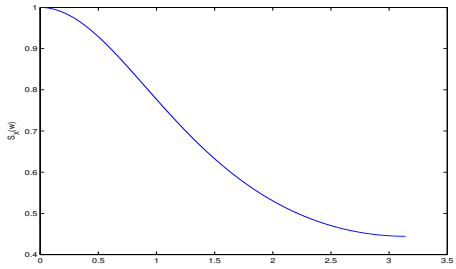


Fig. 4. Spectral density function of the random process employed to compute the limit value of the correlation function. This random process was synthesized by filtering some white noise with an autoregressive filter.

VI. Conclusion

This paper provides further details concerning the asymptotic behaviour of the autocorrelation function of the wavelet packet coefficients issued from the decomposition of a wide-sense stationary random process. By choosing a sufficiently large decomposition level and, then, by increasing the regularity of the filters with respect to the chosen decomposition level, the wavelet packet coefficients tend to become uncorrelated.

The result presented in this paper complements thos established in [1]–[5] and justifies the assumption on which many signal processing techniques based on wavelet packet decompositions are based, namely, that signals are corrupted by white noise.

References

[1] A. Cohen, *Ondelettes et traitement numérique du signal*. Masson, Paris, 1992.

[2] G. G. Walter, *Wavelets and other orthogonal systems with applications*. CRC Press, 1994.

[3] D. Pastor and R. Gay, "Décomposition d'un processus stationnaire du second ordre : Propriétés statistiques d'ordre 2 des coefficients d'ondelettes et localisation frequentielle des paquets d'ondelettes." *Traitement du Signal.*, vol. 12, no. 5, 1995.

[4] A. Isar and I. Naforniţă, *Représentations temps-frequence*. Ed Politehnica, Timişoara, Roumanie, 1998.

[5] D. Leporini and J. C. Pesquet, "High-order wavelet packets and cumulant field analysis," *IEEE Transactions on Information Theory.*, vol. 45, no. 3, pp. 863+, April 1999.

[6] P. F. Craigmile and D. B. Percival, "Assymptotic decorrelation of between-scale wavelet coefficients," *IEEE Transactions on Information Theory.*, vol. 51, no. 3, pp. 1039+, Mar. 2005.

[7] S. Mallat, *A wavelet tour of signal processing*. Academic Press, 1998.

[8] I. Daubechies, *Ten lectures on wavelets*. SIAM, Philadelphie, PA, 1992.

[9] M. V. Wickerhauser, *Adapted Wavelet Analysis from Theory to Software*. AK Peters, 1994.

[10] N. Saito and G. Beylkin, "Multiresolution representation using the autocorrelation functions of compactly supported wavelets," *IEEE Transactions on Signal Processing*, vol. 41, 1993.

[11] J. Shen and G. Strang, "Asymptotic analysis of daubechies polynomials," *Proceedings of the American Mathematical Society.*, vol. 124, no. 12, pp. 3819+, December 1996.

[12] ——, "Asymptotics of daubechies filters, scaling functions, and wavelets," *Applied and Computational Harmonic Analysis.*, vol. 5, no. HA970234, pp. 312+, 1998.

[13] A. Aldroubi, M. Unser, and M. Eden, "Cardinal spline filters: Stability and convergence to the ideal sinc interpolator," *Signal Process.*, vol. 28, no. 8, pp. 127–138, Aug. 1992.

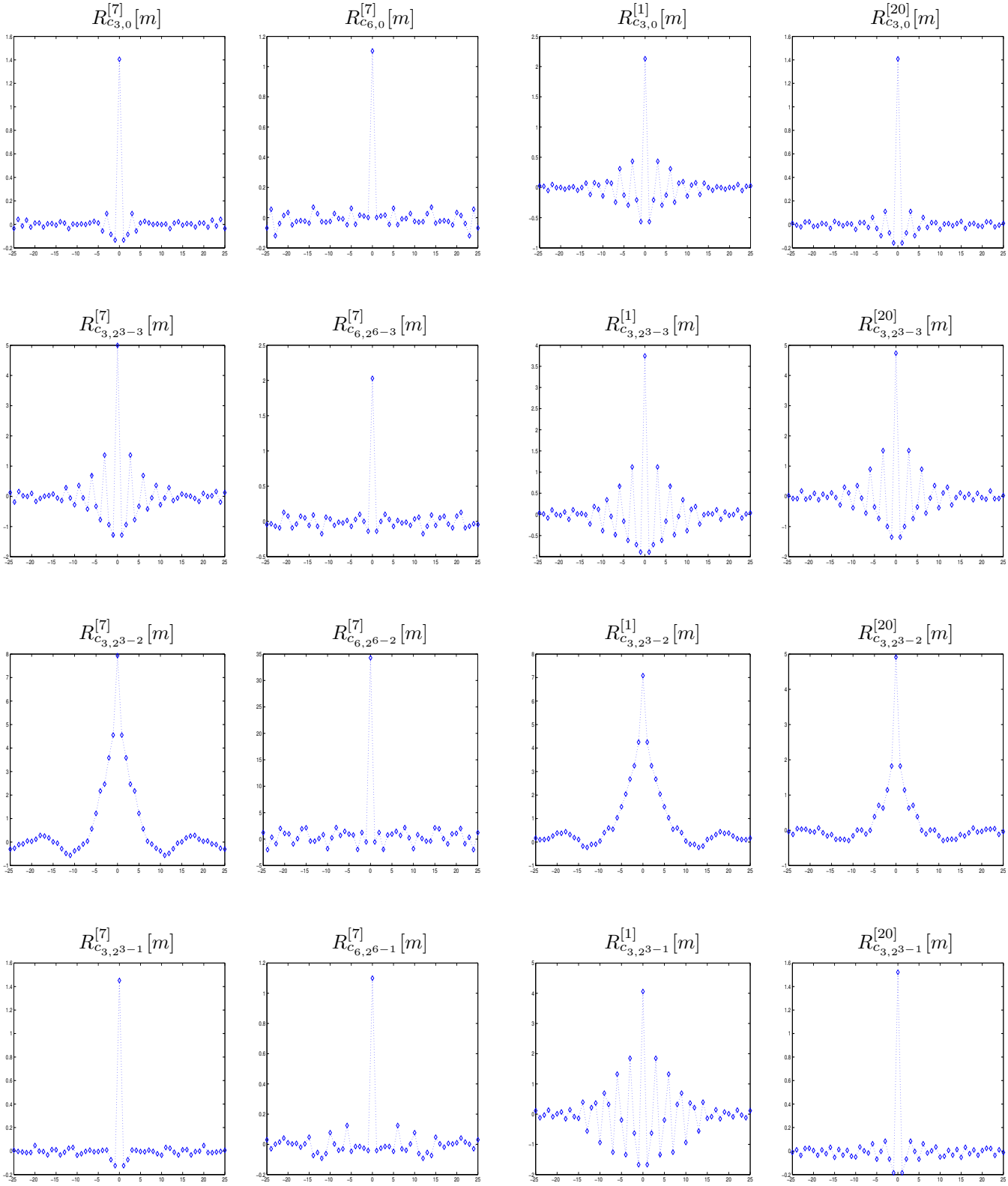


Fig. 5. Autocorrelation function of the wavelet packet coefficients returned by the decomposition of the random process whose spectral density is that of figure 3. The decomposition was achieved by using the Daubechies filters with regularity 7. The first column displays results obtained for $j = 3$ whereas the second column concerns $j = 6$ where j stands for the decomposition level.

Fig. 6. Autocorrelation functions of the wavelet packet coefficients at decomposition level $j = 3$ for the random process with power spectral density given by figure 3. The first column displays the results obtained with the Daubechies filters of regularity 1; the second column presents the results obtained by using the Daubechies filters with regularity 20.

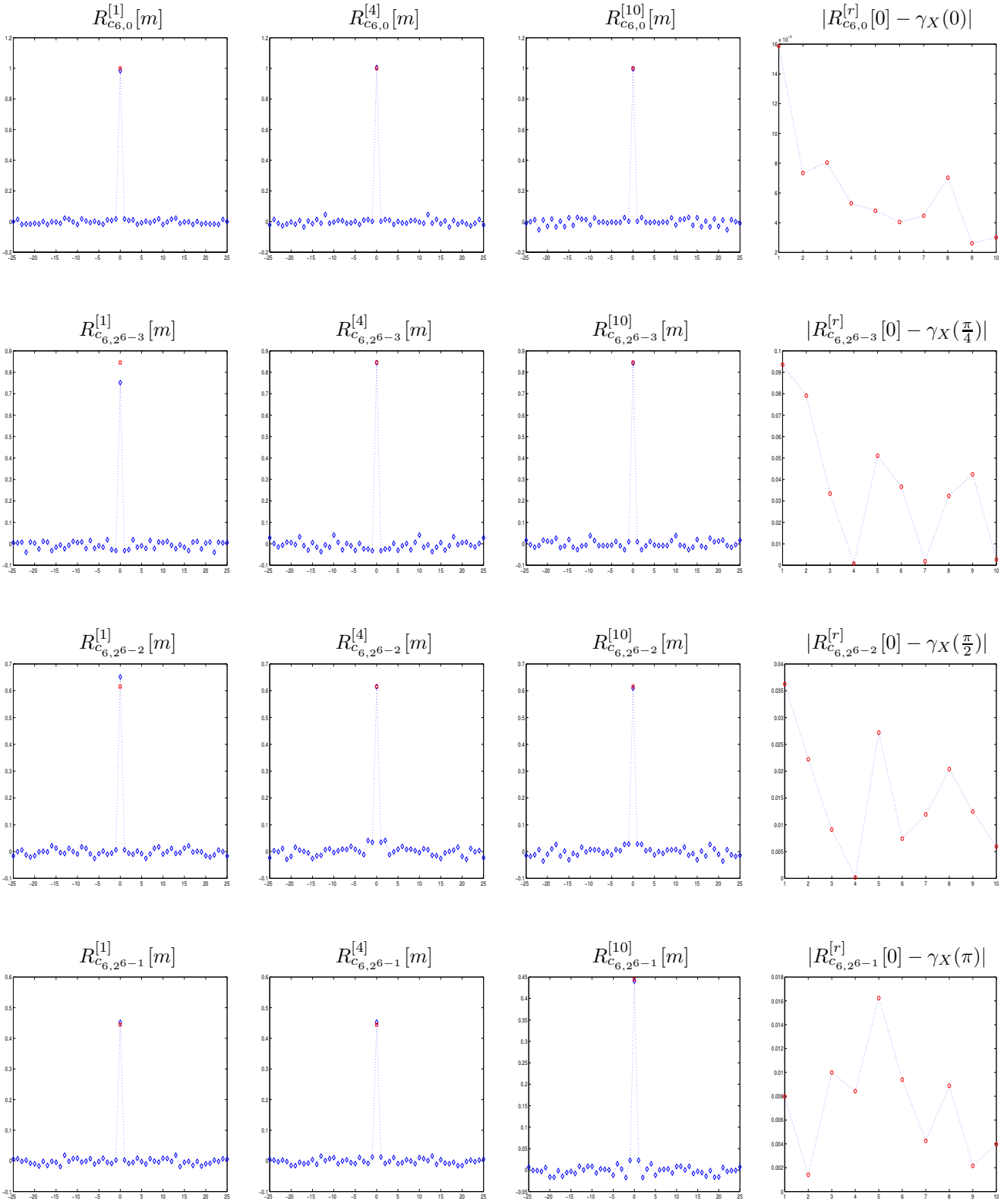


Fig. 7. The first third columns display the autocorrelation functions of the wavelet packet coefficients obtained by decomposing the random process whose spectral density is given by figure 4. The values of these functions are representated by $-\diamond$'s. On each figure, the value of the spectral density function at $a(\kappa)$, where κ is a binary sequence, is represented by a $\square$. For each packet, the maximum of the autocorrelation function must be close to this square. The fourth column displays the differences $|R^{[r]}_{c_{j,n_j}}[0] - S_x(a(\kappa))|$ when the regularities of the Daubechies filters range from 1 to 10, $j = 6$ et $n_j = 0$, 2^{6-3}, 2^{6-2}, et 2^{6-1}.

Buletinul Stiintific al Universitatii "Politehnica" din Timisoara

Seria ELECTRONICA si TELECOMUNICATII
TRANSACTIONS on ELECTRONICS and COMMUNICATIONS

Tom 51(65), Fascicola 1-2, 2006

ON THE MMSE ITERATIVE EQUALIZATION FOR TDMA PACKET SYSTEMS

Adrian-Florin Paun [1] Serban-Georgica Obreja [2]

Abstract – Original turbo equalization using a trellis-based channel equalizer and channel decoder improves significantly the bit error rate performance. However, a large alphabet modulation employed in the systems with multipath channels requires an excessive high number of states in such equalizer, so the optimal maximum *a posteriori* probability (MAP) becomes prohibitively complex. Therefore, sub-optimum equalizers with *a priori* information from the channel decoder have to be considered in order to enhance its performance. In this paper we investigate the performances of minimum mean square error (MMSE) filter based iterative equalization for the Enhanced General Packet Radio Service (EGPRS) radio link. The simulation results demonstrate that MMSE turbo equalization constitutes an attractive candidate for single-carrier wireless transmissions with multilevel modulation, in long delay-spread environments.

Keywords: turbo equalization, MMSE filter, TDMA systems

1. INTRODUCTION

Turbo-equalization [1], [2] is a powerful mean to perform joint equalization and decoding, when considering coded data transmission over time dispersive channels. The association of the code and the discrete-time equivalent channel (separated by an interleaver) is seen as the serial concatenation of two codes. The turbo principle, the iterative exchange of extrinsic information between a soft-in/soft-out (SISO) equalizer and a SISO decoder, may then be used at the receiver for improve its performances.
Classically, these SISO modules are implemented using conventional a posteriori probability (APP) algorithms [3]. This leads to a complexity which evolves in $O(M^L)$ for the equalization; M, L being respectively the modulation alphabet size and the discrete-time equivalent channel length. Obviously, the optimal equalization process needed at the receiver becomes rapidly untractable for long impulse response channels and for high constellations order.

In this paper we study the performances of low complexity SISO equalizer, based on minimum mean square error (MMSE) equalization, proposed in [4] and [5], in context of packet transmission TDMA systems.

A joint design of the equalization and demodulation parts, based on a gaussian assumption, allows producing good estimates of the symbol extrinsic probabilities. Special care is then taken to the computation of the bit extrinsic log-likelihood ratios (LLRs), in order to fully exploit the mutual information between the bits associated with a given complex symbol, capitalizing on methods presented in [6], [7], [8].

2. SYSTEM DESCRIPTION

A. Signal model

The transmission scheme is represented in Fig. 1. A frame of information bits b_k is encoded by a rate-r convolutional encoder. The resulting encoded bits c_m are interleaved using a random permutation function to give the interleaved coded bits x_m. The q bits $x_n^p = x_{(n-1)q+p}$ $(p = 1,...,q)$ are grouped and mapped to a complex symbol d_n, among the $M = 2^q$ possible symbols of the considered constellation. The resulting complex symbols are transmitted over the channel, which is assumed static over a frame and perfectly known. At the receiver, we assume matched filtering to the whole transmission chain, symbol-rate sampling and discrete-time noise whitening. Thus, the channel may be represented by its equivalent discrete-time white noise filter model, i.e. a causal discrete-time filter with coefficients h_j $(j = 0,...,L)$ corrupted by white gaussian noise samples w_n of variance s_w^2.

The symbols r_n at the output of the channel may thus be expressed as

[1]University "Politehnica" of Bucharest; Electronics, Telecom & Inf Technology Faculty
Telecommunications Dep. , Spl. Independentei 313, 060032 Bucharest, Romania, e-mail adi@radio.pub.ro
[2] University "Politehnica" of Bucharest; Electronics, Telecom m& Inf Technology Faculty
Telecommunications Dep., Spl. Independentei 313, 060032 Bucharest, Romania, serban@radio.pub. ro

$$r_n = \sum_{l=0}^{L} h_l d_{n-l} + w_n \qquad (1)$$

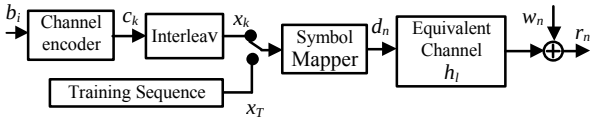


Fig. 1. Transmitter scheme

B. Iterative receiver

A global view of the proposed receiver scheme is given in Fig. 2. Seen at this level of generality, it is similar to a classical turbo-equalizer [1]. It consists of two stages: a SISO equalizer/demapper and a SISO decoder separated by bit-deinterleavers and bit-interleavers. Those two stages exchange extrinsic information, on an iterative fashion, in order to improve the performances. The decoder is implemented using an optimal APP algorithm. We focus in this paper on the presentation of the proposed equalizer/demapper. Note that the a priori inputs and extrinsic outputs of the equalizer/demapper are bit LLRs. It has thus to deal with all the bit/symbol conversion aspects associated with the considered modulation.

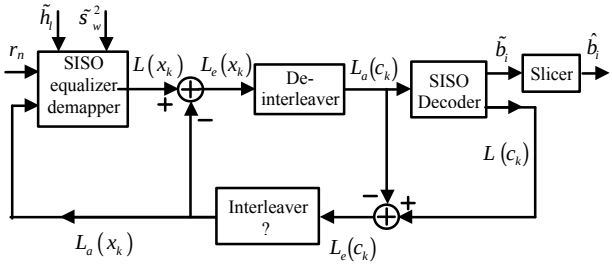


Fig. 2. Turbo equalization receiver

3. SISO EQUALIZER

A. General description

A more detailed scheme of the proposed equalizer is given in Fig. 3. Using the bit a priori LLRs $L_a(x_m)$ produced by the decoder, it begins by computing the first and second order statistics of the symbols d_n. Using those statistics and the received samples r_n, a symbol equalizer produces estimates $\hat{d}_n$ in order to minimize $E\left\{\left|d_n - \hat{d}_n\right|^2\right\}$. The M corresponding symbol extrinsic probabilities are then approximated as $\Pr\left(\hat{d}_n \middle| d_n\right)$ using an equivalent gaussian channel assumption at the output of the symbol equalizer. The parameters of this equivalent channel are calculated for each estimate $\hat{d}_n$ on the basis of the equalizer structure and of the symbols statistics. Based on the obtained symbol extrinsic probabilities, we finally evaluate, for $p = 1,...,q$, the bit extrinsic probabilities $\Pr_e\left(x_i^p\right)$ using the a priori information about the other bits x_n^r $(r \neq p)$ associated with the considered symbol d_n, and output an extrinsic LLR $L_e\left(x_n^p\right)$. To keep the analogy with an optimal APP equalizer, we only use the a priori information available about symbols d_i with $i \neq n$ when computing the estimate $\hat{d}_n$ and the symbol extrinsic probabilities $\Pr\left(\hat{d}_n \middle| d_n\right)$.

B. Symbols statistics using the a priori information

We first calculate an estimation of the mean and variance of each symbol on the basis of the a priori information available. Noting S the set of all possible symbols, with $\text{card}(\mathcal{S}) = M$, for each transmitted symbol d_n, we compute first the symbols a priori probabilities $\Pr_a\left(d_n = s_j\right)\ \forall s_j \in \mathcal{S}$. Assuming independence between the interleaved coded bits and basing on the corresponding bits a priori probabilities this probability can be written as:

$$\Pr_{a,n}\left(s_j\right) = \Pr_a\left(d_n = s_j\right) = \prod_{p=1,...,q} \Pr_a\left(x_n^p\right) \qquad (2)$$

where the q bits x_n^p $(p = 1,...,q)$ takes values in $\{0,1\}$ as a function of the considered symbols s_j. The probabilities $\Pr_a\left(x_n^p\right)$ are classically calculated from $L_a\left(x_n^p\right) = \ln\dfrac{\Pr_a\left(x_n^p = 1\right)}{\Pr_a\left(x_n^p = 0\right)}$, the a priori LLR of the bit x_n^p. The mean value $\bar{d}_n$ of symbol d_n on the basis of the a priori information at time *n* is then:

$$\bar{d}_n = E\left\{d_n \middle| L_{a,n}\right\} = \sum_{s_j \in \mathcal{S}} s_j \times \Pr_{a,n}\left(s_j\right) \qquad (3)$$

Similarly, the variance u_n^2 of symbol at time *n*, on the basis of the a priori information, is:

$$u_n^2 = E\left\{\left|d_n - \bar{d}_n\right|^2 \middle| L_{a,n}\right\} = \sum_{s_j \in \mathcal{S}} |s|^2 \times \Pr_{a,n}\left(s_j\right) - \left|\bar{d}_n\right|^2 \qquad (4)$$

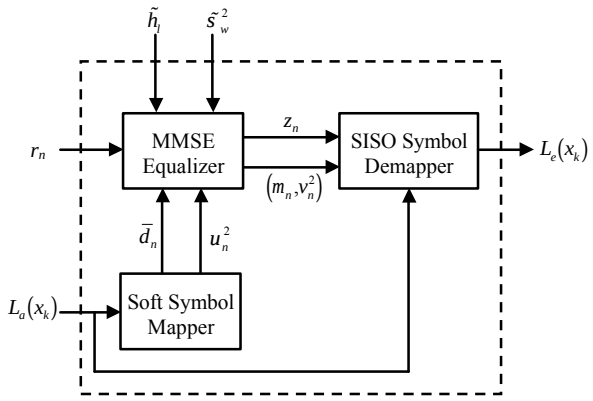


Fig. 3. Block diagram of generic SISO MMSE equalizer with demapper

The expression of u_n^2 depends on the considered modulation. If $|s| = \text{const}\ \ \forall s \in \mathcal{S}$ (i.e. M-PSK), the expectation $E\left\{|d_n|^2\right\}$ is constant and equals s_d^2, that is the variance of a symbol without a priori information. If not (i.e. M-QAM), it has to be calculated explicitly as in (4).

C. Symbol equalizer

Defining the equalizer length as $N = N_1 + N_2 + 1$, we first introduce a sliding-window model using the vectors

$$\mathbf{r}_n = \left[r_{n-N_1} \ldots r_n \ \ldots \ r_{n+N_2} \right]^T \tag{5.a}$$

$$\mathbf{d}_n = \left[d_{n-N_1-L} \ldots d_n \ \ldots \ d_{n+N_2} \right]^T \tag{5.b}$$

$$\mathbf{w}_n = \left[w_{n-N_1} \ldots w_n \ \ldots \ w_{n+N_2} \right]^T \tag{5.c}$$

and the $\left(N \times (N+L)\right)$ -channel matrix

$$\mathbf{H} = \begin{bmatrix} h_L & \cdots & h_0 & 0 & \cdots & \cdots & 0 \\ 0 & h_L & \cdots & h_0 & 0 & \cdots & 0 \\ \vdots & \ddots & \ddots & \ddots & \ddots & \ddots & \vdots \\ 0 & \cdots & \cdots & 0 & h_L & \cdots & h_0 \end{bmatrix} \tag{6}$$

At each time step *n*, we may then write:

$$\mathbf{r}_n = \mathbf{H}\mathbf{d}_n + \mathbf{w}_n \tag{7}$$

where $\mathbf{w}_n$ is a complex gaussian noise vector, i.e. $\mathbf{w}_n \sim \mathcal{N}\left(\mathbf{0}, s_w^2 \mathbf{I}\right)$; $\mathbf{I}$ being the $N \times N$ identity matrix. Considering this sliding-window channel model, we present a low-complexity equalizer [5] in order to produce estimates $\hat{d}_n$, assuming the knowledge of the noise and symbols first and second order statistics.
An MMSE estimator may be expressed in this context as:

$$\hat{d}_n = E\{d_n\} + \mathbf{p}_n^H \left[\mathbf{r}_n - E\{\mathbf{r}_n\} \right] \tag{8}$$

with the length-*N* complex vector $\mathbf{p}_n$ given by:

$$\mathbf{p}_n = \text{cov}\{\mathbf{r}_n, \mathbf{r}_n\}^{-1} \cdot \text{cov}\{\mathbf{r}_n, \mathbf{d}_n\} \tag{9}$$

where ^{H} denotes the conjugate transpose operator and

$$\text{cov}\{\mathbf{x}, \mathbf{y}\} = E\left\{ \left[\mathbf{x} - E\{\mathbf{x}\} \right] \left[\mathbf{y} - E\{\mathbf{y}\} \right]^H \right\}$$

In conformity with turbodecoding and turboequalization principle [5],[6], the a priori information about symbol d_n should not be used in the evaluation of its estimate $\hat{d}_n$. In other words, at time *n*, for symbols d_i with $i \neq n$, we can use the mean $\bar{d}_i$ and the variance u_i^2 computed on the basis of the a priori information in (2) and (3). On the contrary, the mean and variance of symbol d_n is computed without using the corresponding a priori information, which leads to 0 and s_d^2 respectively. The expectation in (8) can be computed as follow:

$$E\{d_n\} = 0 \ \text{ and } \ E\{d_n\} = E\{\mathbf{H}\mathbf{d}_n + \mathbf{w}_n\} = \mathbf{H}\bar{\mathbf{d}}_n \tag{10}$$

where we have defined $\bar{\mathbf{d}}_n = E\{\mathbf{d}_n\}$ as:

$$\bar{\mathbf{d}}_n = \left[\bar{d}_{n-N_1-L} \ \ldots \ \bar{d}_{n-1} \ 0 \ \bar{d}_{n+1} \ \ldots \ \bar{d}_{n+N_2} \right]^T \tag{11}$$

The first factor in (9) may be calculated as follows:

$$\begin{aligned} \text{cov}\{\mathbf{r}_n, \mathbf{r}_n\} &= \mathbf{H} \cdot \text{cov}\{\mathbf{d}_n, \mathbf{d}_n\} \cdot \mathbf{H}^H + s_w^2 \mathbf{I} \\ &= \mathbf{H}\mathbf{R}_{dd,n}\mathbf{H}^H + s_w^2 \mathbf{I} \end{aligned} \tag{12}$$

where

$$\mathbf{R}_{dd,n} = \text{cov}\{\mathbf{d}_n, \mathbf{d}_n\} = E\left\{ \left[\mathbf{d}_n - \bar{\mathbf{d}}_n \right] \left[\mathbf{d}_n - \bar{\mathbf{d}}_n \right]^H \right\} \tag{13}$$

Using the definition given in (3) and once again avoiding using the a priori information available about symbol d_n at time step *n*, this matrix can be expressed as:

$$\begin{aligned} \mathbf{R}_{dd,n} = \text{diag}[&\text{var}\left(d_{n-N_1-L}\right) \ \ldots \ \text{var}\left(d_{n-1}\right) \ s_d^2 \\ &\text{var}\left(d_{n+1}\right) \ \ldots \ \text{var}\left(d_{n+N_2}\right)] \end{aligned} \tag{14}$$

where we used the independence assumption between the coded bits, so that $\text{cov}\left(d_n, d_i\right) = 0$ for $n \neq i$
The second factor in (9) may be calculated as follows:

$$\begin{aligned} \text{cov}\{\mathbf{r}_n, \mathbf{d}_n\} &= \mathbf{H}\text{cov}\left(\mathbf{d}_n, \mathbf{d}_n\right) = \mathbf{H}E\left\{ \left[\mathbf{d}_n - \bar{\mathbf{d}}_n \right] d_n^* \right\} \\ &= \mathbf{H}\,\text{cov}\left(\mathbf{d}_n, d_n^*\right) = \mathbf{H}\mathbf{e}s_d^2 \end{aligned} \tag{15}$$

where **e** denotes a length-(*N*+*L*) vector of all zeros, except for the (N_1+*L*+1)th element, which is 1, and $\mathbf{h} = \mathbf{H}\mathbf{e}$.
Using (9), (12) and (15), the complex vector $\mathbf{p}_n$, which is seen as a time-varying equalization filter, becomes:

$$\mathbf{p}_n = s_d^2 \left[\mathbf{H}\mathbf{R}_{dd,n}\mathbf{H}^H + s_w^2 \mathbf{I} \right]^{-1} \mathbf{h} \tag{16}$$

Finally, using (8), (10) and (16), we obtain the following expression of the estimate symbol $\hat{d}_n$:

$$\begin{aligned} \hat{d}_n &= \mathbf{p}_n^H \left[\mathbf{r}_n - \mathbf{H}\bar{\mathbf{d}}_i \right] = \\ &= s_d^2 \mathbf{h}^H \left[\mathbf{H}\mathbf{R}_{dd,n}\mathbf{H}^H + s_w^2 \mathbf{I} \right]^{-1} \left[\mathbf{r}_n - \mathbf{H}\bar{\mathbf{d}}_n \right] \end{aligned} \tag{17}$$

The generalized MMSE equalizer reduces to classical MMSE equalization for the first iteration of the iterative process when a priori information is not available. This scheme may also be seen as an improved interference canceler taking the statistical nature of the soft values into account. It reduces to a classical soft-interference canceler [9], when perfect a priori information is available.

D. Equivalent AWGN channel assumption

At the output of the equalizer, in order to be able to demodulate the symbols, we assume that the estimate $\hat{d}_n$ is the output of an equivalent AWGN channel having d_n as its input:

$$\hat{d}_n = m_n d_n + h_n \tag{18}$$

m_n is the equivalent amplitude of the signal at the output and h_n is a complex white gaussian noise with zero mean and variance n_n^2. This is equivalent to say that the estimates are complex gaussian distributed,

i.e. $\hat{d}_n \sim \mathcal{N}_c\left(m_n d_n, n_n^2\right)$. The parameters m_n and n_n^2 are calculated at each time step n as a function of the equalizer structure, and thus also of the symbols statistics.

The mean m_n is calculated by first evaluating:

$$E\left\{\hat{d}_n, d_n^*\right\} = m_n E\left\{|d_n|^2\right\} = m_n s_d^2 \quad (19)$$

which can be expressed as:

$$E\left\{\hat{d}_n, d_n^*\right\} = \mathbf{p}_n^H \mathbf{h} s_d^2 \quad (20)$$

and, from (16) and (20), we finally obtain:

$$m_n = \mathbf{p}_n^H \mathbf{h} = \mathbf{h}^H \left[\mathbf{H}\mathbf{R}_{dd,n}\mathbf{H}^H + s_w^2 \mathbf{I}\right]^{-1} \mathbf{h} s_d^2 \quad (21)$$

The variance n_n^2 may be expressed as:

$$n_n^2 = E\left\{|h_n|^2\right\} = E\left\{\left|\hat{d}_n - m_n d_n\right|^2\right\} = E\left\{\left|\hat{d}_n\right|^2\right\} - m_n^2 s_d^2 \quad (22)$$

We have thus:

$$n_n^2 = \mathbf{p}_n^H \left[\mathbf{H}\mathbf{R}_{dd,n}\mathbf{H}^H + s_w^2 \mathbf{I}\right]\mathbf{p}_n - m_n^2 s_d^2 \quad (23)$$

Using (16), (21) and (23), we find the following expression:

$$n_n^2 = m_n s_d^2 - m_n^2 s_d^2 \quad (24)$$

E. Symbol extrinsic probabilities computation

In order to compute the bit extrinsic probabilities, we have first to approximate the symbol extrinsic probabilities. We use therefore the gaussian equivalent channel assumption given in (18) and estimate the symbol posterior probabilities $\Pr(d_i)$ as:

$$\Pr(d_n) = \Pr(d_n|\mathbf{r}) \approx \Pr\left(d_n|\hat{\mathbf{d}}\right) \approx \Pr\left(d_n|\hat{d}_n\right) = \frac{\Pr\left(d_n|\hat{d}_n\right)\cdot \Pr_a(d_n)}{\Pr\left(\hat{d}_n\right)} \quad (25)$$

where $\mathbf{r}$ and $\hat{\mathbf{d}}$ are the sequences of the received symbols and of the estimates respectively and we used the Bayes rule and the equivalent AWGN channel for the frequency selective channel. From (25), the symbol extrinsic probabilities $\Pr_e(d_n)$ may then be written as follows:

$$\Pr_e(d_n) = \mathcal{K}_d \frac{\Pr(d_n)}{\Pr_a(d_n)} \sim \frac{\Pr\left(d_n|\hat{d}_n\right)}{\Pr\left(\hat{d}_n\right)} \sim \Pr\left(\hat{d}_n|d_n\right) \quad (26)$$

where $\mathcal{K}_d$ is a normalization constant and where the last equivalence is obtained when omitting the terms common to all hypotheses. Using the parameters m_n and n_n^2 of the equivalent AWGN channel computed for the estimate $\hat{d}_n$, the M symbol extrinsic probabilities at time n may finally be approximated as follows:

$$\Pr\left(\hat{d}_n|d_n\right) = \frac{1}{n_n^2 p}\exp\left(-\frac{\left|\hat{d}_n - m_n d_n\right|^2}{n_n^2}\right) \quad (27)$$

Note that these probabilities were obtained without using the a priori information available about symbol d_n, which is consequent with the optimal algorithm. Note also, that the equivalent channel assumption allowed taking the symbols statistics into account

F. Bit extrinsic LLR computation

The extrinsic probabilities of a given coded bit x_n^p may be expressed as a function of the symbol extrinsic probabilities $\Pr_e(d_n)$ [7], as follows:

$$\Pr_e\left(x_n^p\right) = \mathcal{K}_x \frac{\Pr\left(x_n^p\right)}{\Pr_a\left(x_n^p\right)} \approx \sum_{s_i : x_n^p} \Pr_e(d_n)\left[\prod_{\substack{r=1,\ldots,q \\ r\neq p}} \Pr_a\left(x_n^r\right)\right] \quad (28)$$

where $\mathcal{K}_x$ is a normalization constant, $\Pr\left(x_n^p\right) = \Pr\left(x_n^p|\mathbf{r}\right)$ are the posterior probabilities of the bit x_n^p and the notation $s_i : x_n^p$ represents the subset of the symbols $s_i \in \mathcal{S}$ with a given value of x^p. $\Pr_e(d_n) = \Pr_e(d_n = s_i)$ denotes the extrinsec probability that the emitted symbol at time n is the symbol s_i from $\mathcal{S}$ set of possible symbols. Equation (28) allows taking the a priori probabilities of the other bits x_n^r $(r = 1,\ldots,q\ ;\ r \neq p)$ associated with the considered symbol d_n in order to evaluate the extrinsic probabilities of the bit x_n^p. So, the mutual influence between encoded bits is used for a better demapping. Using (27) and (28), we can then approximate $\Pr_e\left(x_n^p\right)$ as:

$$\Pr_e\left(x_n^p\right) \approx \sum_{s_i : x_n^p} \Pr_e\left(\hat{d}_n|d_n\right)\left[\prod_{\substack{r=1,\ldots,q \\ r\neq p}} \Pr_a\left(x_n^r\right)\right] \quad (29)$$

The finally form of bit extrisec LLR $L_e\left(x_n^p\right) = \ln\dfrac{\Pr_e\left(x_n^p = 1\right)}{\Pr_e\left(x_n^p = 0\right)}$ at the output of the equalizer is:

$$L_e\left(x_n^p\right) \approx \ln\frac{\sum_{s_i : x_n^p = 1} \Pr\left(\hat{d}_n|d_n\right)\left[\prod_{\substack{r=1,\ldots,q \\ r\neq p}} \Pr_a\left(x_n^r\right)\right]}{\sum_{s_i : x_n^p = 0} \Pr\left(\hat{d}_n|d_n\right)\left[\prod_{\substack{r=1,\ldots,q \\ r\neq p}} \Pr_a\left(x_n^r\right)\right]} \quad (30)$$

As shown in [6], we can obtain a more robust implementation in the logarithmic domain, using the well-known generalized maximum function and taking the considered constellation into account to get further simplifications

G. Asymptotic performances

In the iterative process at the receiver, asymptotic performances are reached when perfect a priori information is available at the equalizer/demapper. In this case, it can be shown that the proposed scheme manages to totally suppress ISI and reaches the

matched filter bound (MFB). The only difference remaining while using the equivalent AWGN channel model at the output of the equalizer is the noise correlation. However, this noise correlation is not taken into account in the demodulation process and, from the point of view of the decoder, it is broken due to the presence of the deinterleaver. The asymptotic performances of the scheme are thus logically identical to those of iterative demodulation and decoding on an AWGN channel [7]. They can be obtained by simulation considering iterative demodulation and decoding on an AWGN channel with perfect a priori information at the demapper, considering the same code and the same mapping. There is no optimal solution; it depends on the considered E_b / N_0, code, mapping, channel and on the number of iterations allowed at the receiver

4. SIMULATION RESULTS

Simulations of the MMSE turbo equalization were performed in the Enhanced Data for GSM Evolution (EDGE) radio access scheme. The burst structure is described in [10]. A burst carries 2x58 playload data symbols, includes a middle training sequence of 26 symbols and 8.25 guard symbols at the end. We performed the simulation for MCS-5 coding scheme [10], which employs 8-PSK modulation. So 3 interleaved encoded bits are mapped, using Gray labelling, into one burst symbol. For the sake of simplicity, the performance results will be examined only over user data, which are encoded by a rate 1/3 non-recursive non-systematic convolutional encoder with constraint length 7 and octal generator polynomials 133,171,145. In MCS-5 coding scheme, to obtain a user data code rate R=0,37 [10], the CPS=20 puncturing pattern are used. The coded punctured user data are interleaved with deterministic interleaver from EDGE technical recommendation, combined with the header part and the flags, forming a block of 1392 bits. This bits are partitioned over 4 data blocks of 348 bits, i.e. 2x58 8-PSK symbols, which are finally mapped onto 4 different bursts. We evaluated the bit error rate (BER) for the transmission over channel A, a channel with 11 taps [0.04, -0.05, 0.07, -0.21, -0.5, 0.72, 0.36, 0.21, 0.03, 0.07], and over channel B, a channel of length five [2-0.4j, 1.5+1.8j, 1, 1.2-1.3j, 0.8+1.6j]. The real or complex path gains were normalized such that $\sum_{l=0}^{L-1}|h_l|^2 = 1$.

In all simulations, we totalized 50 frame errors for each E_s / N_0 and performed six iterations. The reference curve, represented by a dashed line in both figures, corresponds to the performance of the coded transmission scheme over an ISI-free AWGN channel. In figure 4 are plotted the receiver performance (BER after decoding) for transmission over channel A. The filter length is 10, N_1=0 and N_2=10. We can see that, for medium and high SNR, the receiver improves its performance from an iteration to another and after five-six iterations it achieves the ISI free transmission performance, for E_s / N_0 greater than three.

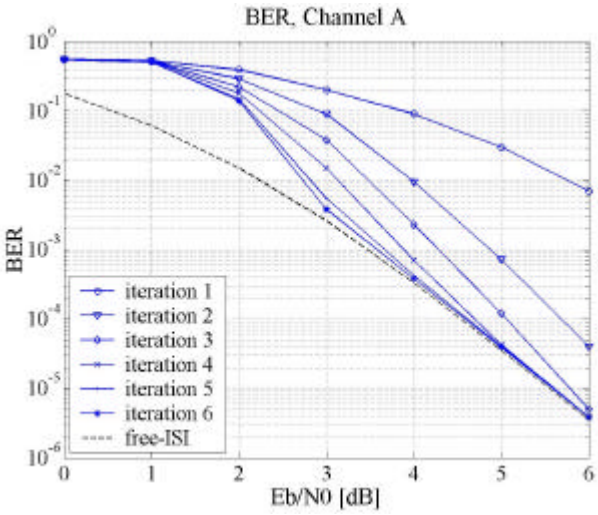


Fig. 4. BER-curves for MCS-5 over channel A.

In figure 5 the results of the transmision over channel B and N_1=3 , N_2=7 are reproduced. Because this channel is harder to equalize, the receiver improves its performances by iteration technique for higher threshold of E_s / N_0. We can draw the same conclusion: the ISI is totaly supressed after an iteration number, depending on amount of E_s / N_0 for the medium and high SNR.

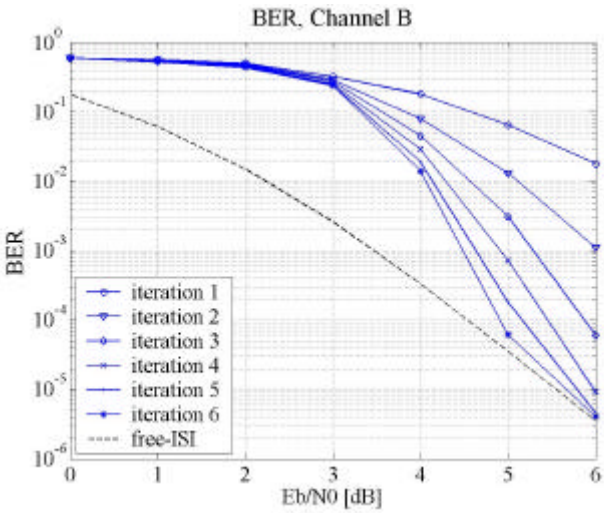


Fig. 5. BER-curves for MCS-5 over channel B.

This turboequalization scheme preserve the performance previous reported in [5], even in a dereministic interleaver context of EDGE system.

5. CONCLUSIONS

The 8-PSK modulation scheme used in EDGE requires a sub-optimum equalizer. The MMSE filter based equalizers has the lowest complexity, which is independent of the size of the symbol alphabet. Their

moderate performance can be increased using iterative principle in so called turbo-equalization scheme.

REFERENCES

[1] C. Douillard *et al.*,"Iterative correction of intersymbol interference: Turbo-equalization", *ETT*, Vol.6, No. 5, pp. 507–511, Sep.-Oct. 1995.

[2] G. Bauch, H. Khorram and J. Hagenauer, "Iterative equalization and decodingin mobile communications systems", in *Proc. 2nd EPMCC'97*, pp. 307–312, Bonn, Germany, Sep.-Oct. 1997.

[3] L. R. Bahl, J. Cocke, F. Jelinek and J. Raviv, "Optimal decoding of linear codes for minimizing symbol error rate", *IEEE Trans. on Inform. Theory*, Vol. 20, pp. 284–287, Mar. 1974.

[4] A. Dejonghe and L. Vandendorpe, "Turbo-equalization for multilevel modulation: an efficient low-complexity scheme", *in Proc IEEE ICC 2002* , pp1863-1867, May 2002

[5] C. Laot, R. Le Bidan, D. Leroux, "Low-Complexity MMSE Turbo Equalization: A Possible Solution for EDGE", in *IEEE Trans. on Wireless Comm*., May 2005

[6] M. Tuchler, A. Singer and R. Koetter, " Turbo equalization: principles and new results", *IEEE Trans. on Communications*, pp754-767, May 2002

[7] S. ten Brink, J. Speidel and R.-H. Yan, "Iterative demapping and decoding for multilevel modulations", in *Proc IEEE Globecom'98*, pp. 579–584, Sydney, Australia, Nov. 1998.

[8] P. Magniez *et al.*, "Improved turbo-equalization with application to bit interleaved modulations", in *Proc. Asilomar Conf. on Signals, Systems and Computers*, Monterey (CA), USA, Oct. 2000.

[9] A. Glavieux, C. Laot and J. Labat, "Turbo equalization over a frequency selective channel", *Proc. Int. Symp. on Turbo Codes and Related Topics*, pp. 96–102, Brest, France, Sep. 1997.

[10] GSM 05.03: "Digital cellular telecommunications system (Phase 2+); Channel coding".

[11] M. Tüchler, A.C. Singer and R. Koetter, "Minimum Mean Squared Error Equalization Using A Priori Information," *IEEE Trans. Signal Proc*, Vol. 50, pp. 673-683, Mar 2002.

[12] N. Al-Dhahir and J. M. Cioffi, "Fast Computation of Channel-Estimate Based Equalizers in Packet Data Transmission", *IEEE Trans on Signal Proc*, Vol. 43, No. 11, pp. 2462-2473 Nov 1995,.

[13] P. Vila *et al.*, "Reduced-complexity soft demapping for turbo-equalization", in *Proc. of the second Int. Symp. on Turbo Codes and Related Topics* , pp. 515–518, Brest, France, Sep. 2000.

[14] M. T¨uchler, J. Hagenauer, "Linear time and frequency domain turboequalization", in *Proc. 53rd Vehicular Technology Conference, Spring* , pp. 1449–1453, May 2001.

Buletinul Ştiinţific al Universităţii "Politehnica" din Timişoara

Seria ELECTRONICĂ şi TELECOMUNICAŢII
TRANSACTIONS on ELECTRONICS and COMMUNICATIONS

Tom 51(65), Fascicola 2, 2006

Benefits of building information system with wireless connected mobile device - PDPT Framework

Ondrej Krejcar[1]

Abstract – The proliferation of mobile computing devices and local-area wireless networks has fostered a growing interest in location-aware systems and services. Additionally, the ability to let a mobile device determine its location in an indoor environment at a fine-grained level supports the creation of a new range of mobile control system applications. Main area of interest is in model of radio-frequency (RF) based system enhancement for locating and tracking users of our control system inside buildings. The framework described here joins the concepts of location and user tracking in an extended existing control system. The experimental framework prototype uses a WiFi network infrastructure to let a mobile device determine its indoor position as well as to deliver IP connectivity. User location is used to data pre-buffering and pushing information from server to user's PDA. Experiments show that location determination can be realized with a room level granularity.

I. INTRODUCTION

The usage of various wireless technologies that enable convenient continuous IP-level (packet switched) connectivity for mobile devices has increased dramatically and will continue to do so for the coming years. This will lead to the rise of new application domains each with their own specific features and needs. Also, these new domains will undoubtedly apply and reuse existing (software) paradigms, components and applications. Today, this is easily recognized in the miniaturized applications on network-connected PDAs that provide more or less the same functionality as their desktop application equivalents. The web browser application is such an example of reuse. Next to this, it is very likely that these new mobile application domains adapt new paradigms that specifically target the mobile environment. We believe that an important paradigm is context-awareness. Context is relevant to the mobile user, because in a mobile environment the context is often very dynamic and the user interacts differently with the applications on his mobile device when the context is different. While a desktop machine usually is in a fixed context, a mobile device goes from work, to on the road, to work in-a-meeting, to home, etc. Context is not limited to the physical world around the user, but also incorporates the user's behavior, and terminal and network characteristics.

Context-awareness concepts can be found as basic principles in long-term strategic research for mobile and wireless systems such as formulated in [5]. The majority of context-aware computing to date has been restricted to location-aware computing for mobile applications (location-based services). However, position or location information is a relatively simple form of contextual information. To name a few other indicators of context awareness that make up the parametric context space: identity, spatial information (location, speed), environmental information (temperature), resources that are nearby (accessible devices, hosts), availability of resources (battery, display, network, bandwidth), physiological measurements (blood pressure, hart rate), activity (walking, running), schedules and agenda settings. Context-awareness means that one is able to use context information.

We consider location as prime form of context information. Our focus here is on position determination in an indoor environment. Location information is used to determine an actual user position and his future position. We have performed a number of experiments with the control system, focusing on position determination, and are encouraged by the results. The remainder of this paper describes the conceptual and technical details of this.

II. BASIC CONCEPTS AND TECHNOLOGIES OF USER LOCALIZATION

The proliferation of mobile computing devices and local-area wireless networks has fostered a growing interest in location-aware systems and services. A key distinguishing feature of such systems is that the application information and/or interface presented to the user is, in general, a function of his physical location. The granularity of location information needed could vary from one application to another. For example, locating a nearby printer requires fairly coarse-grained location information whereas locating

[1] VSB Technical University of Ostrava, Centre for Applied Cybernetics, Department of Measurement and Control, 17. listopadu 15, 708 33 Ostrava, Czech Republic, ondrej.krejcar@vsb.cz, http://cak.vsb.cz

a book in a library would require fine-grained information.
While much research has been focused on development of services architectures for location-aware systems, less attention has been paid to the fundamental and challenging problem of locating and tracking mobile users, especially in in-building environments. We focus mainly on RF wireless networks in our research. Our goal is to complement the data networking capabilities of RF wireless LANs with accurate user location and tracking capabilities for user needed data pre-buffering. This property we use as information ground for extension of control system.

2.1 Location-Based Services

Location-based services (LBS) are touted as 'killer apps' for mobile systems. An important difference between fixed and mobile systems is that the latter operate in a particular context, and may behave differently or offer different information and interaction possibilities depending on this context. Location is often the principal aspect determining the context. Many different technologies are used to provide location information. Very common is the GPS system, which uses a network of satellites and provides position information accurate within 10–20 m. However, due to its satellite based nature, it is not suited for indoor positioning. In cellular telecommunication networks such as GSM, the cell ID gives coarse-grained position information with an accuracy of about 200 m to 10 km. For fine-grained indoor location information, various technologies are available, based on infrared, RF, or ultrasonic technologies often using some type of beacon or active badge. Given the ubiquity of mobile devices like PDAs, however, active badges will probably be superseded by location technologies incorporated in these devices.
In the context of our experimental setup, we need indoor position information accurate enough to determine the room in which the user is located. We must deploy a separate location technology, where we use the information available from a WiFi network infrastructure to determine the location with room-level accuracy. By this information possible user track is estimate.

2.2 WiFi - IEEE 802.11

The Institute of Electrical and Electronics Engineers (IEEE) develops and approves standards for a wide variety of computer technologies. IEEE designates networking standards with the number 802. Wireless networking standards are designated by the number 11. Hence, IEEE wireless standards fall under the 802.11 umbrella. Ethernet, by the way, is called 802.3 [1].
The 802.11b is an updated and improved version of the original IEEE 802.11 standard. Most wireless networking products today are based on 802.11b. 802.11b networks operate at a maximum speed of 11 Mbps, slightly faster than 10-BASE-T Ethernet, providing a more than fivefold increase over the original 802.11 spec. The 802.11 standard provided for the use of DSSS and FHSS spread-spectrum methods. In 802.11b, DSSS is used.
We use only 802.11b infrastructure (PDA has only this standard) so other standards (802.11a or g) is not needed to describe. However, it can be possible to develop a PDPT framework with it.

2.3 Data Collection

A key step in the proposed research methodology is the data collection phase. We record information about the radio signal as a function of a user's location. The signal information is used to construct and validate models for signal propagation. Among other information, the WaveLAN NIC makes available the signal strength (SS) and the signal-to-noise ratio (SNR). SS is reported in units of dBm and SNR is expressed in dB. A signal strength of s Watts is equivalent to 10*log10(s/0.001) dBm. A signal strength of s Watts and a noise power of n Watts yields an SNR of 10*log10(s/n) dB. For example, signal strength of 1 Watt is equivalent to 30 dBm. Furthermore, if the noise power is 0.1 Watt, the SNR would be 10 dB. The WaveLAN driver extracts the SS and the SNR information from the WaveLAN firmware each time a broadcast packet is received. It then makes the information available to user-level applications via system calls. It uses the wlconfig utility, which provides a wrapper around the calls, to extract the signal information.

2.4 Localization Methodology

The general principle is that if a WiFi-enabled mobile device is close to such a stationary device – Access Point (AP), it can "ask" the location provider's position by setting up a WiFi connection. If the mobile device knows the position of the stationary device, it also knows that its own position is within a 100-meter range of this location provider. Granularity of location can improve by triangulation of two or several visible WiFi APs as described on figure [Fig. 1].

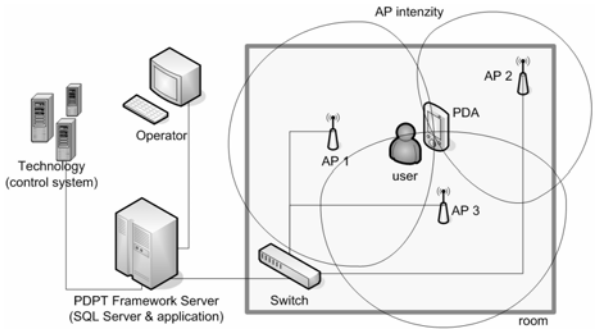


Fig. 1. Localization principle - triangulation.

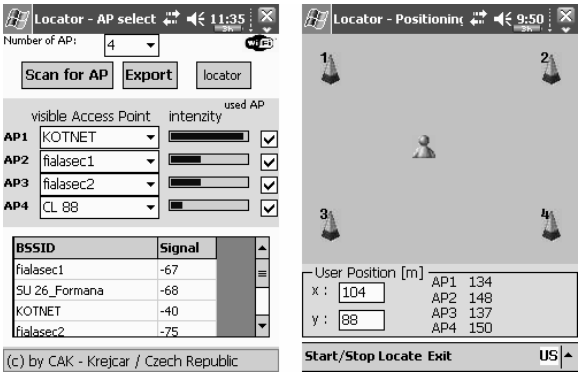


Fig. 2. PDA Locator – AP intensity & Positioning.

The PDA client will support the application in automatically retrieving location information from nearby location providers, and in interacting with the server. Naturally, this principle can be applied to other wireless technologies.
The application (locator) based on .NET language is now created for testing. It is implemented in C# using the MS Visual Studio .NET 2003 with compact framework and a special OpenNETCF library enhancement [3] and [5]. Current application [Fig. 2] records just one set of signal strength measurements. By this set of value the actual user position is determined.

2.5 WiFi Middleware

The WiFi middleware implements the client's side of location determination mechanism on the Windows Mobile 2005 PocketPC operating system and is part of the PDA client application. The libraries used to manage WiFi middleware are: AccessPoint, AccessPointCollection, Adapter, AdapterCollection, AdapterType, ConnectionStatus, Networking, NetworkType, and SignalStrength. Methods from the Net library are used for example to display Visible WiFi AP. See figure [Fig. 3].

```
dtVisibleAP = new DataTable("Visible AP");
DataRow drDataRow;
adptrColection = Networking.GetAdapters();
foreach (Adapter adptr in adptrColection)
{
  Application.DoEvents();
  if (adptr.Type==AdapterType.Ethernet)
  {
    foreach (AccessPoint ap in
            adptr.NearbyAccessPoints)
    {
       drDataRow = dtVisibleAP.NewRow();
       drDataRow["BSSID"] =
            (ap.Name.ToString());
       drDataRow["Signal [%]"] =
((ap.SignalStrength.Decibels).ToString());
       dtVisibleAP.Rows.Add(drDataRow);
    }
  }
}
```

Fig. 3 Sample code – signal strength from AP.

2.6 Predictive data push technology

This part of project is based on model of location-aware enhancement, which we used in created control system. These information about are useful in framework to increase real dataflow from wireless access point (server side) to PDA (client side). Primary dataflow is enlarged by data pre-buffering. These techniques form the basis for predictive data push technology (PDPT). PDPT copies data from information server to clients PDA to be on hand when user comes at desired location.
The benefit of PDPT consists in reduction of time needed to display desired information requested by a user command on PDA. Time delay may vary from a few seconds to number of minutes. It depends on two aspects. First one is the quality of wireless Wi-Fi connection used by client PDA. A theoretic speed of Wi-Fi connection is max 825 kB/s. However, the test of transfer rate from server to client's PDA, which we have carried out within our Wi-Fi infrastructure provided the result speed only 160 KB/s. The second aspect is the size of copied data.
The application (locator) based on .NET language is now created for testing. Current application (see figure [Fig. 2]) records just one set of signal strength measurements. By this set of value the actual user position is determined.

2.7 Framework design

PDPT framework design is based on most commonly used server-client architecture. To process data the server has online connection to the control system. Data from technology are continually saved to SQL Server database [3] and [1]. The part of this database (desired by user location or his demand) is replicated online to client's PDA where it is visualized on the screen. User PDA has location sensor component which continuously sends to the framework kernel the information about nearby AP's intensity. The kernel processes this information and makes a decision if and how a part of SQL Server database will be replicated to client's SQL Server CE database.
The kernel decisions constitute the most important part of whole framework because the kernel must continually compute the position of the user and track and make a prediction of his future movement. After doing this prediction the appropriate data (part of SQL Server database) are pre-buffered to client's database for future possible requirements. The PDPT framework server is created as Microsoft web services to handle as bridge between SQL Server and PDPT PDA Clients.

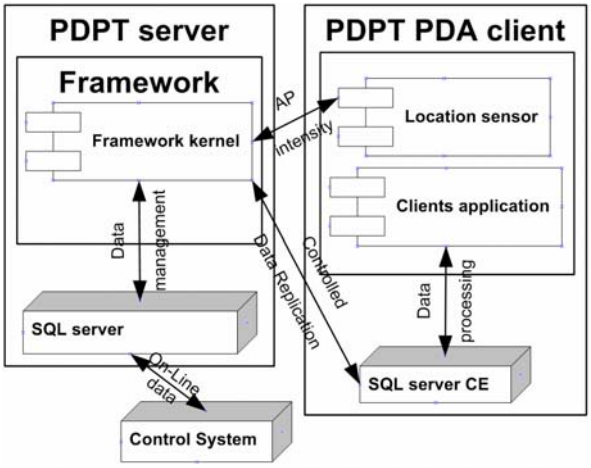


Fig. 4 System architecture – UML design.

III. EXPERIMENTS

We have executed a number of indoor experiments with the PDPT framework, using the PDPT PDA application. WiFi access points are placed at different locations in building, where the access point cells partly overlap. We have used triangulation principle of AP intensity to get better granularity.

It has been found that the location determination mechanism selects the access point that is closest to the mobile user as the best location provider. Also, after the loss of IP connectivity, switching from one access point to another (a new best location provider) takes place within a second in the majority of cases, resulting in only temporary loss of IP connectivity. This technique partially uses a special Radius server [4] to realize "roaming" known in cell networks. User who loss the existing signal of AP must ask the new AP to get IP. This is known as "renew" in Ethernet networks. At the end of this process, user has his same old IP and connection to new AP. Other best technique to realize roaming is using of WDS (Wireless Decision System).

Currently, the usability of the PDPT PDA application is somewhat limited due to the fact that the device has to be continuously powered. If not, the WiFi interface and the application cannot execute the location determination algorithm, and the PDPT server does not receive location updates from the PDA client.

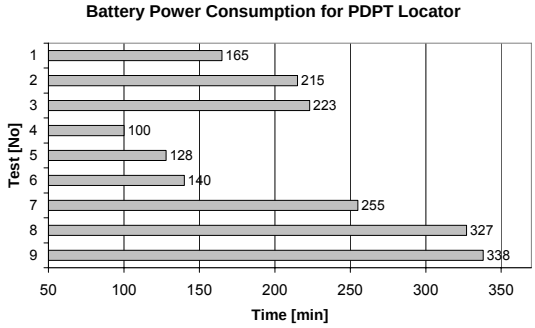


Fig. 5 Battery power consumption graph.

3.1 Battery power consumption tests

We have executed a number of tests of battery power consumption with three PDA devices running PDPT Locator. The tests were executed from 100 % battery level to 20 % battery level with balanced load. The first was HTC Blue Angel PH20B which is known also as MDA III from T-Mobile (Intel XScale PXA263 CPU, MS WM2005 OS). The second one was iPAQ h4150 from Hewlett & Packard Company (H&P) (Intel XScale PXA255 CPU, MS WM2003 OS). The third one was iPAQ hx4700 from H&P (Intel XScale PXA270 CPU, MS WM2003 SE OS). These devices have Li-Ion battery with different capacity (1490 mAh, 1000 mAh and 1800 mAh). MDA III device has integrated GSM module in addition.

Test	Type	CPU	Scan	WiFi	Backlight
1	MDA III	400	2 s	ext.	50%
2		400	10 s	norm.	off
3		100	2 s	norm.	off
4	h4150	400	2 s	ext.	50%
5		400	10 s	norm.	off
6		100	2 s	norm.	off
7	hx4700	624	2 s	ext.	50%
8		624	10 s	norm.	off
9		104	2 s	norm.	off

Fig. 6 Battery tests description.

The test chart [Fig. 5] shows number of results. The first, most evident and expected result is caused by different battery packs. The power consumption is worse for h4150 model and the best for hx4700 model. The second aspect is evident as well. The score is better when the backlight is turned off. However, the very large score is at hx4700 test case comparing to two other PDA. The last interested result is however in 100 MHz CPU speed setting. The speed decreasing was controlled by special utility managing the core of operating system. When the maximum speed of CPU was decreased, the working time of PDA increased about several percent. This result is last useful thing for enlarge battery power consumption.

The practical type of PDA usage with PDPT application is somewhere between minimum and maximum score of these tests for such model of PDA. For example the mean usage of the worse PDA iPAQ h4150 is about two hours of working time so it is not very comfortable, but it is usable for many types of operations. Other two devices have battery consumption time higher so practical use is without remarkable limitation.

3.2 Data transfer increase tests using PDPT Framework

The result of utilization of PDPT framework is mainly at data transfer speed reducing. The second test is

focused on real usage of developed PDPT Framework and his main issue at increased data transfer. At table [Fig. 7] are summary of eighteen tests with three type of PDA and three type of data transfer mode. Each of these eighteen tests is fivefold reiterated for better accuracy. At table are only average values from each iteration.

Test	Type	Mode	Data	Time	Speed
1	MDA III	SQl CE	257	0.4	643
2		SQl CE	891	0.4	2228
3		SQL	257	5	51
4		SQL	891	13	69
5		PDPT	257	1.1	234
6		PDPT	891	3.2	278
7	h4150	SQl CE	257	0.5	514
8		SQl CE	891	0.5	1782
9		SQL	257	5	51
10		SQL	891	14	64
11		PDPT	257	1.2	214
12		PDPT	891	3.7	241
13	hx4700	SQl CE	257	0.3	857
14		SQl CE	891	0.4	2228
15		SQL	257	5	51
16		SQL	891	13	69
17		PDPT	257	0.9	286
18		PDPT	891	2.5	356

Fig. 7 Data transfer tests description.

The data mode column has three data transfer mode. The SQL CE mode represents the data saved at mobile device memory (SQL Server CE) and the data transfer time is very high. The second mode SQL means data which are stored at server (SQL Server 2005). Primary the data are loaded over Ethernet / Internet to SQL Server CE of mobile device and secondary the data are shown to user. The data transfers time consumption of this method are generally very high and the waiting time for user is very large. The third data mode PDPT is combination of previous two methods. The PDPT mode has very good results in form of data transfer acceleration. Realization of this test consists at user movement from location A to B at different way direction. Location B was a destination with requested data which are not contained at SQL CE buffer in mobile device before test.

CONCLUSION

The main objective of this paper is in the enhancement of control system for locating and tracking of users inside a building. It is possible to locate and track the users with high degree of accuracy.

In this paper, we have presented the control system framework that uses and handles location information and control system functionality. The indoor location of a mobile user is obtained through an infrastructure of WiFi access points. This mechanism measures the quality of the link of nearby location provider access points to determine actual user position. User location is used in the core of server application of PDPT framework to data pre-buffering and pushing information from server to user PDA. Data pre-buffering is most important technique to reduce time from user request to system response.

The experiments show that the location determination mechanism provides a good indication of the actual location of the user in most cases. The median resolution of the system is approximately five meters. Some inaccuracy does not influence the way of how the localization is derived from the WiFi infrastructure. For the PDPT framework application this was not found to be a big limitation as it can be found at chapter Experiments. The experiments also show that the current state of the basic technology used for the framework (mobile device hardware, PDA operating system, wireless network technology) is now at the level of a high usability of the PDPT application.

REFERENCES

1. Reynolds, J.: Going Wi-Fi : A Practical Guide to Planning and Building an 802.11 Network, CMP Books, 2003. ISBN 1578203015
2. Wigley, A., Roxburgh, P.: ASP.NET applications for Mobile Devices, Microsoft Press, Redmond, 2003. ISBN 073561914X
3. Tiffany, R.: SQL Server CE Database Development with the .NET Compact Framework, Apress, 2003. ISBN 1590591194
4. The Internet Engineering Task Force RADIUS Working Group: http://www.ietf.org/
5. The Wireless World Research Forum (WWRF): http://www.wireless-world-research.org/
6. OpenNETCF - Smart Device Framework: http://www.opennetcf.org/
7. Krejcar, O.: User Localization for Intelligent Crisis Management. In 3rd IFIP Conference on Artifical Intelligence Applications and Innovations – AIAI 2006, Athenes, Springer, 2006, p. 221-227, ISBN 0-387-34223-0
8. Krejcar, O.: Predictive Data Push Technology Framework - Wireless User Localization Usability in Control Systems, IFAC PDeS, Brno, Czech Republic, 2006, pp. 378-383, ISBN 80-214-3130-X

Buletinul Ştiinţific al Universităţii "Politehnica" din Timişoara

Seria ELECTRONICĂ şi TELECOMUNICAŢII
TRANSACTIONS on ELECTRONICS and COMMUNICATIONS

Tom 51(65), Fascicola 2, 2006

Performance of Multi Binary Turbo-Codes on Nakagami Flat Fading Channels

Maria Kovaci[1], Alexandre de Baynast[2], Horia G. Balta[1], Miranda M. Nafornita[1]

Abstract – In this paper, performance in terms of Bit Error Rate (BER) and Frame Error Rate (FER) of multi binary turbo codes (MBTC) over Nakagami frequency-nonselective fading channels are presented. The proposed MBTCs consist of the parallel concatenation of two identical 2/3-rate recursive systematic, convolutional (RSC) double binary codes. We choose to model the channel fading with Nakagami-*m* distribution since it fits well to the empirical fading data of the current wireless transmission systems. The simulation results show that the MBTCs outperform the classical turbo codes for low-targeted FER (around 10e-4) since their error-floor is negligible.

Keywords: multi binary turbo code, flat fading channel, Nakagami distribution

I. INTRODUCTION

The multi-binary turbo-codes (MBTC) recently proposed by C. Douillard and C. Berrou [1], outperform the classical turbo-codes (TC) invented in 1993 by C. Berrou, A. Glavieux and P. Thitimajshima [2]. Indeed they show in [1] that parallel concatenation of *r*-input binary RSC codes offers several advantages versus to single-input-binary TCs over additive white Gaussian noise channel (AWGN) especially for their very low error-floor.

In this paper we investigate performance of the MBTC codes over fading channels. The fading phenomenon occurs in radio transmission channels and it is due to the presence of multiple paths that vary during the transmission, [3]. The transmission scheme that we considered in this paper is shown in Fig.1. The input-output relation of the transmission can be expressed as:

$$y_k = \alpha_k \cdot x_k + w_k \ , \qquad (1)$$

where x_k and y_k are the transmitted and received data at time k, respectively; the parameter α_k is a random value which characterizes the fluctuations from symbol to symbol (fast fading) or from block to block (block fading) [4]. Its distribution determines the channel type: Rayleigh, Rice or Nakagami.

At the output of the encoder, the binary encoded sequence $\{u_k\}$ is mapped into a binary phase shift keying (BPSK) modulation. The resulting sequence $\{x_k\}$ is normalized such that it has unitary variance. The noise samples $\{w_k\}$ are assumed to be zero-mean independent identically distributed Gaussians with a variance equal to $\overline{w_k^2}$. This variance can be expressed as [5]:

$$\overline{w_k^2} = \frac{1}{2 \cdot 10^{SNR/10}} , \qquad (2)$$

where *SNR* represents the signal-to-noise ratio in decibels.

Then, the coeffcient α_k is equal to:

$$\alpha_k = \sqrt{\frac{\gamma_k}{SNR}} , \qquad (3)$$

where γ_k is a Nakagami distributed random variable. The Nakagami-*m* distribution models well the fading in physical radio channels [6]. Through the parameter *m*, the Nakagami distribution can model various fading conditions that range from severe to moderate.

A new accurate random number generator for Nakagami *m* distribution has been recently proposed in [5]. We use it to evaluate the behavior of the MBTCs over Nakagami flat fading channel. The simulation results are compared to the classical TC. The paper is organized as follows.

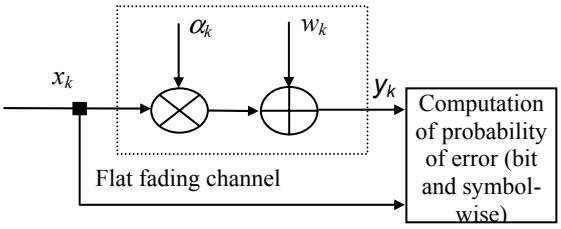


Fig.1 Considered transmssion scheme over flat time-varying channels.

[1] Department of Communications, Faculty of Electronics and Telecommunications, Bd. V. Pârvan Nr. 2, 300223 Timişoara, e-mail: kmaria@etc.upt.ro, balta@etc.upt.ro, monica.nafornita@etc.upt.ro
[2] Department of Electrical and Computer Engineering, Rice University, MS-366-6100 Main Street, Houston, Texas 77005, e-mail: debaynas@rice.edu

In Section II the general scheme of the multi input convolutional encoder is presented. Section III describes the multi binary turbo-encoding scheme, and the advantages of this construction compare to the classical TC. Implementation issues are discussed in Section IV. Simulation results are presented in Section V and Section VI offers some concluding remarks.

II. MULTI BINARY CONVOLUTIONAL ENCODER

The main novelty of the MBTC compare to TC is the use of encoders with mutiple inputs as component codes [1]. The general scheme of a multi input encoder (with r/(r+1) rate) is presented in Fig.2.

Let $S_t = [\, s_m^t \ldots s_2^t \; s_1^t \,]^T$ and $U_t=[\, u_r^t \; u_{r-1}^t \; \ldots \; u_1^t \,]^T$ denote the encoder state and the r-component column input vector at time t, respectively. The operator $(.)^T$ denotes the transpose of a vector. The full generator matrix of the multi input encoder has the following form:

$$H = \begin{bmatrix} h_{r+1,m} & h_{r,m} & \ldots & h_{1,m} & h_{0,m} \\ \ldots & \ldots & \ldots & \ldots & \ldots \\ h_{r+1,2} & h_{r,2} & \ldots & h_{1,2} & h_{0,2} \\ h_{r+1,1} & h_{r,1} & \ldots & h_{1,1} & h_{0,1} \end{bmatrix} \qquad (4)$$

By eliminating first and last columns in matrix H, i.e. the weights for the redundant bit and the weights for the recursive part, we obtain the matrix H_0 as follows:

$$H_0 = \begin{bmatrix} h_{r,m} & \ldots & h_{1,m} \\ \ldots & \ldots & \ldots \\ h_{r,2} & \ldots & h_{1,2} \\ h_{r,1} & \ldots & h_{1,1} \end{bmatrix} \qquad (5)$$

The feedback vector and the output vector have the form:

$$\begin{aligned} H_R &= [h_{0,m} \; \ldots \; h_{0,2} \; h_{0,1}]^T \\ H_{out} &= [h_{r+1,m} \; \ldots \; h_{r+1,2} \; h_{r+1,1}]^T. \end{aligned} \qquad (6)$$

By using the previous notations, the main equation that describes the encoder shown in Fig.2 becomes:

$$(S_{t+1})_{m\times 1} = (H_0)_{m\times r} \cdot (U_t)_{r\times 1} + (T)_{m\times m} \cdot (S_t)_{m\times 1} \qquad (7)$$

where the matrix T is defined as:

$$T = \begin{bmatrix} 0 & 1 & 0 & \ldots & 0 & 0 \\ 0 & 0 & 1 & \ldots & 0 & 0 \\ \ldots & \ldots & \ldots & \ldots & \ldots & \ldots \\ 0 & 0 & 0 & \ldots & 0 & 1 \\ h_{0,m} & h_{0,m-1} & h_{0.m-2} & \ldots & h_{0.2} & h_{0,1} \end{bmatrix} = \begin{bmatrix} 0_{m-1\times 1} & I_{m-1} \\ & H_R \end{bmatrix} \qquad (8)$$

The redundant output is equal to:

$$c^t = H_{out} \cdot S_t + W \cdot S_{t+1} \qquad (9)$$

where the vector W is equal to $[0 \quad 0 \quad \ldots \quad 0 \quad 1]_{1\times m}$.

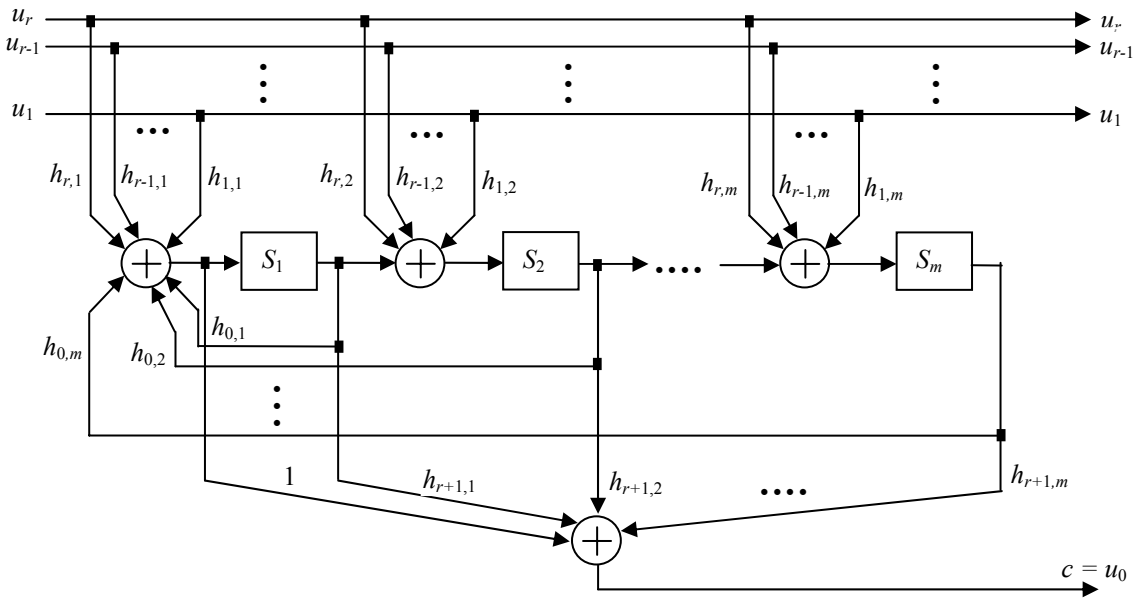


Fig. 2 Multi input convolution encoder – general scheme.

Throughout the paper, we consider a 16-state duo binary encoder, which has a larger minimum distance than an 8-state duo binary encoder. Therefore, we expect very good performance especially for low FER. Among the 16-state duo binary codes, we select the code with generator polynomials [11 11 1 12], which has been proposed in [1]. The memory of the encoder m is equal to 4 as shown in Fig. 3. For this particular code, the vectors S_t and U_t defined in Section II are equal to:

$$S_t = \begin{bmatrix} s_4^t & s_3^t & s_2^t & s_1^t \end{bmatrix}^T \text{ and } U_t = \begin{bmatrix} u_2^t & u_1^t \end{bmatrix}^T$$

and Equation (5) becomes:

$$H = \begin{bmatrix} 1 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 \\ 1 & 1 & 0 & 0 \\ 1 & 1 & 1 & 0 \end{bmatrix} = \begin{bmatrix} 11 & 11 & 1 & 12 \end{bmatrix}_{10}$$

We expressed here the matrix H in a compact form where each column of H is represented by a decimal value corresponding to its binary column-vector.
Therefore, we have:

$$H_0 = \begin{bmatrix} 1 & 0 \\ 0 & 0 \\ 1 & 0 \\ 1 & 1 \end{bmatrix},\ H_R = \begin{bmatrix} 1 & 1 & 0 & 0 \end{bmatrix},\ H_{out} = \begin{bmatrix} 1 & 0 & 1 & 1 \end{bmatrix}$$

The relations (7) and (8) can be rewritten as:

$$(S_{t+1})_{4\times 1} = (H_0)_{4\times 2} \cdot (U_t)_{2\times 1} + (T)_{4\times 4} \cdot (S_t)_{4\times 1}$$

with: $T = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 1 & 0 & 0 \end{bmatrix}$

This particular duo binary convolutional encoder is used for both constituent codes of the multi binary turbo coder that we proposed in the next section.

III. MULTI BINARY TURBO CODES

In this section, we propose a multi-binary turbo-code based on the parallel concatenation of two duo-binary convolutional encoders that have been described in the previous section. The main advantage of the multi-binary turbo-codes is their minimum distance, which generally is larger than for the classical TC.
First we investigate the interleaver design for proposed the MBTC.
A parallel concatenation of two identical r-ary RSC encoders with an r-bit-word interleaver (Π) is presented in Fig. 4
For example, suppose that we indexed two input sequences of 5 binary words each (r=2) as they are entering into the corresponding encoder. At the output of the r-bit-word interleaver, we observe:
$\{\{1,3,5,7,9\},\{2,4,6,8,10\}\} \xrightarrow{\Pi} \{\{5,1,9,3,7\},\{6,2,10,4,8\}\}$
Blocks of k bits (k being a multiple of r) are encoded twice by the bi-dimensional code, whose rate is $r/(r+2)$, to obtain a multi binary turbo encoder scheme in Fig.4.

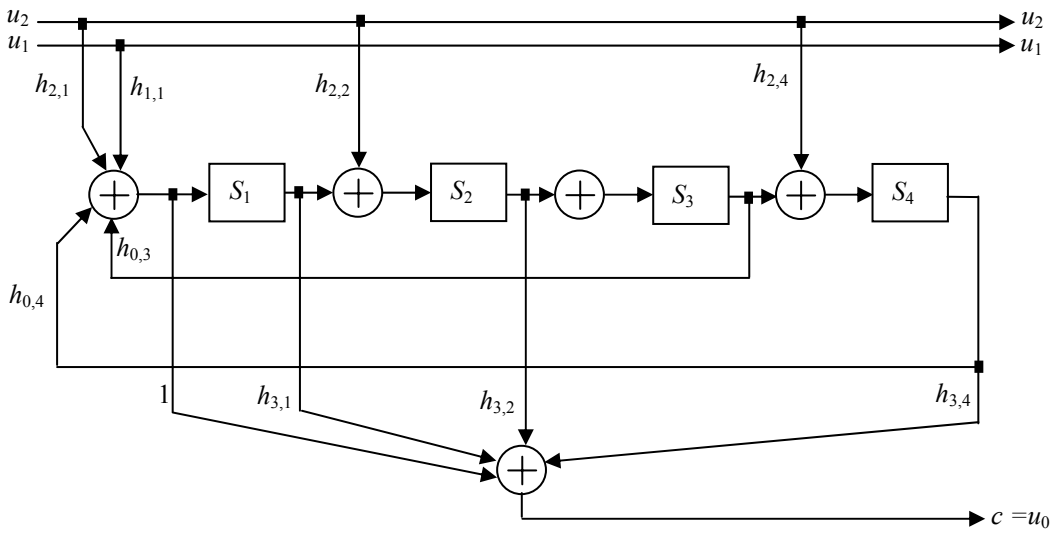


Fig. 3 The scheme of the 16-state duo binary encoder with H=[11 11 1 12].

In order to increase the minimum distance of the proposed code, we propose to use an additional symbol permutation for the 16-state duo-binary turbo code. The permutation parameters have been careful chosen in order to maximize the minimum distance [1].

The permutation function $i = \Pi(j)$, is done in two steps.

For $j = 0, \ldots, N-1$:

Step 1: the intra-symbol interleaver swaps $r_{j,1}$ and $r_{j,2}$ if j mod 2 =0. Otherwise, no action is taken.

Step 2: the mapping of intersymbol interleaver is given by:

$$i = (P \times j + Q(j) + 3) \text{ mod N, with}$$

$$\begin{aligned} Q(j) &= 0 && \text{if } j \text{ mod } 4 = 0 \\ Q(j) &= Q_1 && \text{if } j \text{ mod } 4 = 1 \\ Q(j) &= 4Q_0 + Q_2 && \text{if } j \text{ mod } 4 = 2 \\ Q(j) &= 4Q_0 + Q_3 && \text{if } j \text{ mod } 4 = 3 \end{aligned} \qquad (10)$$

with: P=35, Q_0=1, Q_1=4, Q_2=4, Q_3=12.

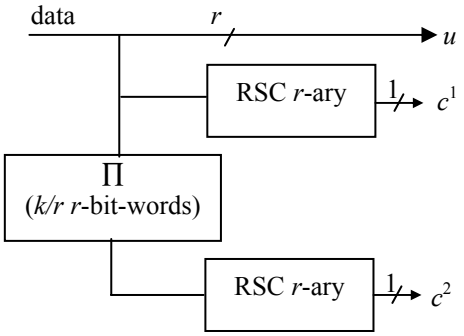


Fig. 4 The r-ary turbo-encoder proposed in [2].

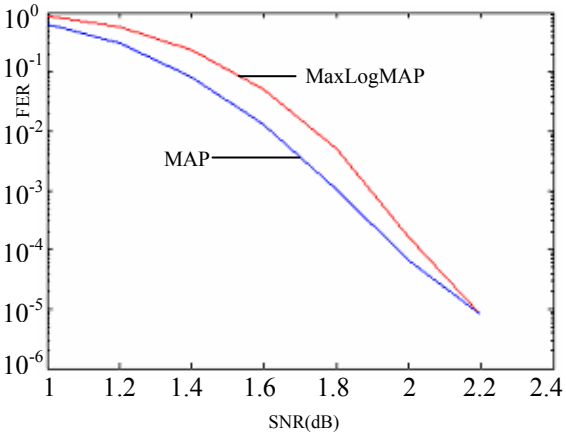


Fig. 5 Comparison between two implementations for the decoding algorithm of MBTCs over Nakagami-5 fading channel: MAP algorithm and MaxLogMAP approximation. In both case, we evaluate the performance in term of Frame Error Rate. The MBTC that we used is described in Fig. 4 (based on two 4-memory RSC codes). Codeword size is equal to 1504 bits with 1/2-rate.

IV. IMPLEMENTATION ISSUES

We decode the received sequence with the iterative decoding algorithm proposed in [7]. In this algorithm, computation of logarithms is required. To tackle this important issue, two approximations have been proposed in the literature [8]: 1) the logarithmic function is approximated by few values that are stored in a look-up table (LUT) of moderate size, 2) Max-Log-MAP [8] for which the log-function is approximated by the max function with some properly chosen offset coefficient. We choose the second solution for its robustness to fixed-point implementation. Interestingly, this algorithm converges slightly faster than the original MAP decoding algorithm [9].

In this section, we evaluate the performance loss due to the Max Log MAP approximation for the MBTCs over Nakagami-5 fading channel.

Fig.5 shows the FER performance of the rate-1/2: double binary 16-state TC as a function of SNR with both implementations: full MAP decoding algorithm [9] and the Max-Log-MAP decoding algorithm [8]. We consider a block size of 188 information bytes. For low SNR, the loss does not exceed 0.17 *dB*. Interestingly, for low FER (smaller than 10e-5), the loss becomes negligible.

It is worth noting that the performance loss due to the Max-Log-MAP approximation is smaller for MBTC than for the classical TCs [1].

V. EXPERIMENTAL RESULTS

We consider the following setup for our simulations in Fig.6. The considered MBTC consists of the parallel concatenation of two identical rate 2/3 recursive, systematic, convolutional code (RSC) with a memory of 4 and with the encoding matrix H equal to [11 11 1 12]). The trellis of the first encoder is closed at zero and the trellis of the second encoder is unclosed. The rate of this duo-binary turbo code is equal to 1/2. It is worth noting that no puncturing is needed. We used the interleaver described in the previous paragraph. The data block length, k=2·N, is equal to 188 bytes=2×752 bits. In our simulation we assume QPSK signaling with perfect channel phase recovery at the receiver. As mentioned in the previous section, we used the Max-Log-MAP version of the decoding algorithm [8]. The extrinsic information is less reliable, especially at the beginning of the iterative process. A more robust approach consists in scaling the extrinsic information with a scaling factor smaller than 1.0. The best observed performance is obtained for a scaling factor of 0.75 instead of 0.7 [8] since the performance results are similar and 0.75 can be represented only with 2 bits. A maximal number of 15 iterations with a stopping criterion are used.

The transmission channel that we considered in our simulations is the m-Nakagami time selective channel. The performance of MBTC for Nakagami flat fading channels with m>1 are upper bounded by the

performances over Rayleigh channel (which corresponds to the Nakagami flat fading channel with m=1, i.e. the most time-selective channel) and lower bounded by static channels (which corresponds to the Nakagami flat fading channel with m=∞, i.e. time non-selective channel). We compare the performance of the proposed MBTC to the Shannon limit. For Gaussian inputs, the channel capacity for Nakagami flat fading channel with parameter m is given by Equation (6) in [10]. For binary inputs, achievable rate can be found in [11] for the particular cases: m equal to 1 (Rayleigh fading) and m equal to +∞ (no fading, AWGN channel).

In Fig.6, we show that MBTC performs well over time-selective channels even for moderate codeword length (in our case, the codeword size is 1504 bits). Indeed, the performance loss from m=∞ to m=1 does not exceed 3 *dB* for any FER and BER whereas the theoretical loss given by the Shannon limits is about 1.6 *dB*. This moderate additional loss is due to the fact we are using small codeword size.

In [1], the authors show by simulation that MBTC outperform classical TC over AWGN channel. We propose a similar comparison over Nakagami channels in Fig.7.

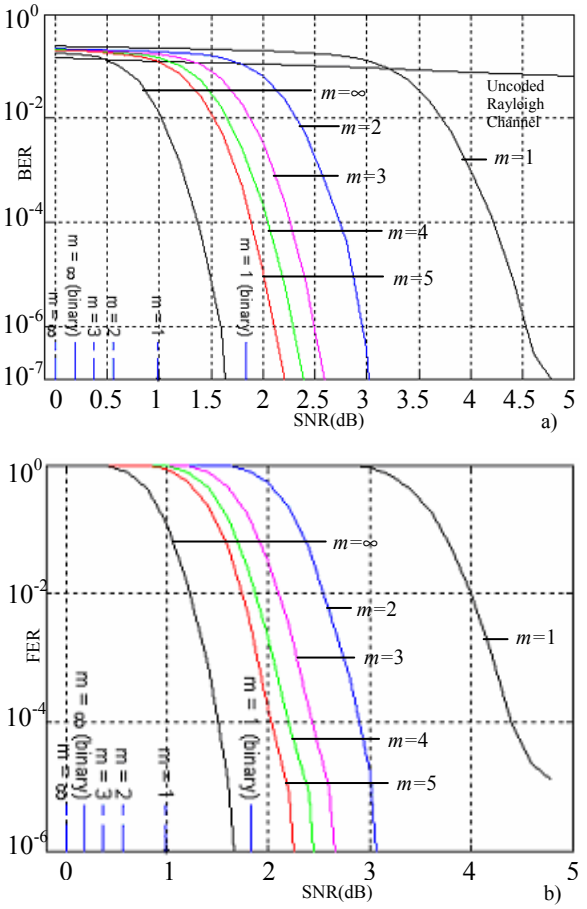


Fig. 6 Performance for Multi Binary Turbo-coded transmission over Nakagami flat fading channel: a) Bit Error Rate (BER), b) Frame Error Rate (FER), are plotted as functions of SNR for various fading parameters m. The thresholds corresponding to the Shannon limit for-Nakagami flat fading channels are also represented for several values of m (dashed line: Gaussian inputs; solid line: Binary inputs).

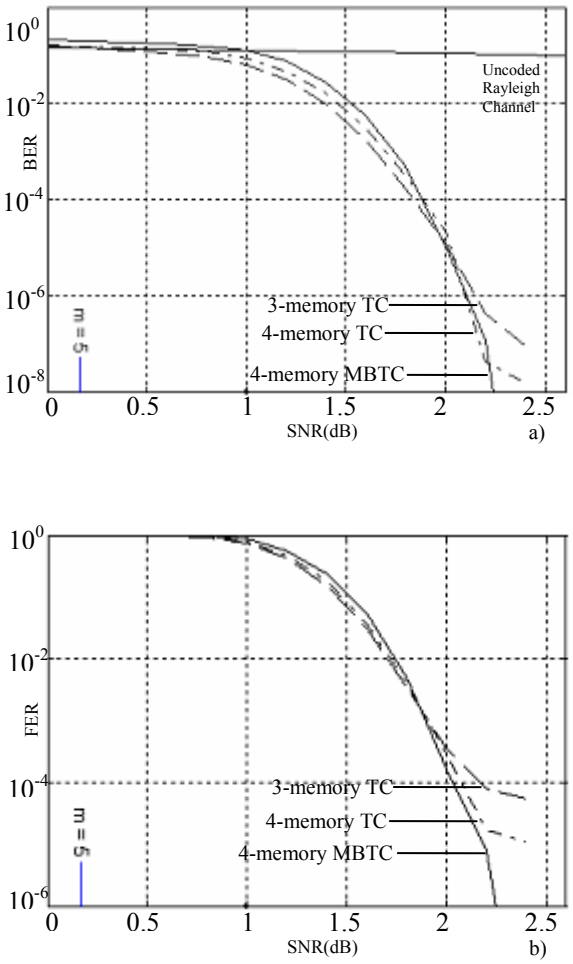


Fig.7 BER (a)) and FER(b)) performance for 4-memory MBTC, 3-memory and 4-memory classical TCs over Nakagami flat fading channel (fading parameter m=5), with the MAP decoding algorithm. Threshold corresponding to the Shannon limit with Gaussian inputs is also represented.

In Fig.7, we considered for all TCs a data block length k equal to 1504 bits, a rate R=1/2 and a fading parameter m=5 (severe time-selective channel). Thanks to the stopping criterion, the average number of iterations is about 3 iterations for a maximum number of iterations of 15. We assume QPSK modulation. For 5·10e-3<FER<10e-5 performance of TCs and MBTCs are similar. Interestingly, for very low FER (let say roughly smaller than 10e-5), MBTCs do not exhibit any error floor to the contrary of the TCs thanks to their large minimum distance. This feature is particularly interesting in future wireless high data rate systems where high quality of service is required.

VI. CONCLUSIONS

In this paper we presented the BER and FER performance for 1/2-rate MBTC based on the Max-Log-MAP approximation over the m- Nakagami time selective channel. The channel models with Nakagami flat fading cover a large scale of practice situations from no time selectivity (AWGN channel) to very

severe time selectivity (Rayleigh channel). By simulations we show that MBTC performs well over time-selective channels even for moderate codeword length (in all simulations we assumed a codeword size of 188 bytes). In a future work, it would be interesting to evaluate the impact of the channel estimation error on the performance.

REFERENCES

[1] C. Douillard, C. Berrou, "Turbo Codes With Rate-m/(m+1) Constituent Convolutional Codes", *IEEE Transactions on Communications*, Vol. 53, No. 10, Oct. 2005, pp.1630-1638.

[2] C. Berrou, A. Glavieux, P. Thitimajshima , "Near Shannon limit error-correcting coding and decoding: Turbo-codes", *Proc.ICC'93*, Geneva, Switzerland, May 1993, pp. 1064 – 1070.

[3] John G. Proakis, "Digital communications", *McGraw-Hill Series in Electrical and Computer Engineering Stephen W*., 2001.

[4] A.J. Coulson, A.G. Williamson, R.G. Vaughan, "Improved fading distribution for mobile radio Communications", *IEE Proceedings*, Vol. 145, Issue 3, June 1998, pp.197 – 202.]

[5] H. Balta, M. Kovaci, A. De Baynast " Performance of Turbo-Codes on Nakagami Flat Fading (Radio) Transmission Channels", *Signals, Systems and Computers, 2005. Conference Record of the Thirty-Ninth Asilomar Conference* on October 28 - November 1, 2005, pp. 606 – 610.

[6] N. Beaulieu, C. Cheng, "Efficient Nakagami-*m* Fading Channel Simulation", IEEE journal on Vehicular Technology, vol. 54, no. 2, March 2005.

[7] J. Hagenauer, E. Offer, L. Papke, "Iterative decoding of binary block and convolutional codes", *IEEE Transactions on Information Theory*, vol. 42, pp. 429-445, March 1996.

[8] J. Vogt and A. Finger, "Improving the max-log-MAP turbo decoder," *Electron. Lett.*, vol. 36, no. 23, pp. 1937–1939, Nov. 2000.

[9] L.R. Bahl, J. Cocke, F. Jelinek, and J. Raviv, "Optimal Decoding of Linear Codes for Minimising Symbol Error Rate", *IEEE Transactions on Information Theory*, Vol. 20, pp. 284-287, March 1974.

[10] Jilei Hou; P.H. Siegel, L.B. Milstein, "Performance analysis and code optimization of low density parity-check codes on Rayleigh fading channels", *IEEE Journal on Selected Areas in Communications*, vol 19, Issue 5, pp. 924-934, May 2001.

[11] Y.-D. Yao, A.U.H. Sheikh, "Investigations into cochannel interference in microcellular mobile radio systems", *IEEE Transactions on Vehicular Technology,* vol 41, pp. 114 - 123, May 1992.

Buletinul Ştiinţific al Universităţii "Politehnica" din Timişoara

Seria ELECTRONICĂ şi TELECOMUNICAŢII
TRANSACTIONS on ELECTRONICS and COMMUNICATIONS

Tom 51(65), Fascicola 2, 2006

Pixel-wise masking for watermarking using local standard deviation and wavelet compression

Corina Nafornita[1], Alexandru Isar[1], Monica Borda[2]

Abstract – Perceptual watermarking in the wavelet domain has been proposed for a blind spread spectrum technique, taking into account the noise sensitivity, texture and the luminance content of all the image subbands. In this paper, we propose a modified perceptual mask, where the texture content is appreciated with the aid of the local standard deviation of the original image, which is further compressed in the wavelet domain. The effectiveness of the new perceptual mask is appreciated by comparison with the old watermarking system.

Keywords: image watermarking, discrete wavelet transform, wavelet statistical analysis, perceptual watermark

I. INTRODUCTION

Because of the unrestricted transmission of multimedia data over the Internet, content providers are seeking technologies for protection of copyrighted multimedia content. Watermarking has been proposed as a means of identifying the owner, by secretly embedding an imperceptible signal into the host signal [1]. Important properties of an image watermarking system include perceptual transparency, robustness, security, and data hiding capacity [2].

In this paper, we choose to study a blind watermarking system, which operates in the wavelet domain. The watermark is masked according to the characteristics of the human visual system (HVS), taking into account the texture and the luminance content of all the image subbands. The detection is blind (it does not use the original image). The system that inspired this study is described in [3].

We propose a different perceptual mask based on the local standard deviation of the original image. The local standard deviation is compressed in the wavelet domain to have the same size as the subband where the watermark is to be inserted.

The paper is organized as follows. Section 2 discusses perceptual watermarking; section 3 describes the system proposed in [3]; section 4 presents the new masking technique; some simulation results are discussed in section 5; finally some conclusions are drawn in section 6.

II. PERCEPTUAL WATERMARKING

One of the qualities required to a watermark is its imperceptibility. There are some ways to assure this quality. One way is to exploit the statistics of the coefficients obtained computing the discrete wavelet transform, DWT, of the host image. We can estimate the coefficients variance at any decomposition level and detect (with the aid of a threshold detector), based on this estimation, the coefficients with large absolute value. Embedding the message in these coefficients, corresponding to the first three wavelet decomposition levels, a robust watermark is obtained. The robustness is proportional with the threshold's value. This solution was proposed by Nafornita, Isar and Borda in [4], where the robustness was also increased by multiple embedding. All the message symbols are embedded using the same strength. The coefficients with large absolute values correspond to pixels localized on the contours of the host image. The coefficients with medium absolute value correspond to pixels localized in the textures and the coefficients with low absolute values correspond to pixels situated in zones with high homogeneity of the host image. The difficulty introduced by the embedding technique already described [4] is to insert the entire message into the contours of the host image, especially when the message is long enough, because only a small number of pixels lie on the contours of the host image. For long messages or for multiple embedding of a short message the threshold value must be decreased and the message is also inserted in the textures of the host image. Hence, the embedding technique already described is perceptual. Unfortunately, the method's robustness analysis is not simple, especially when the number of repetitions is high. The robustness increases due to the increased number of repetitions but it also decreases due to the decreased threshold required (some symbols of the

This work was financed in the framework of CNCSIS grants TD/47/2005/34702 and A/29/27688/2005.
[1] Politehnica University of Timisoara, Communications Department,
Bd. V. Parvan Nr. 2, 300223 Timisoara, e-mail {corina.nafornita, alexandru.isar}@etc.upt.ro
[2] Technical University of Cluj-Napoca, Communications Department
Cluj-Napoca, e-mail monica.borda@com.utcluj.ro

message are embedded in regions of the host image with high homogeneity). In fact, there are some coefficients not used for embedding. This is the reason why, some authors like Barni, Bartolini and Piva [3] proposed a different approach for embedding a perceptual watermark in all the coefficients. They prefer to insert the message in *all* detail wavelet coefficients but using *different strengths* (only at the first level of decomposition). For the coefficients corresponding to the contours of the host image they use a higher strength, for the coefficients corresponding to the textures of the host image they use a medium strength and for the coefficients corresponding to the regions with high regularity in the host image they use a lower strength. This is in accordance with the analogy between water-filling and watermarking proposed by Kundur in [5].

III. THE SYSTEM PROPOSED IN [3]

A. Embedding

The image is decomposed into 4 levels using Daubechies-6 wavelet mother, where I_l^θ is the subband from level $l \in \{0,1,2,3\}$, and orientation $\theta \in \{0,1,2,3\}$. A binary watermark $x^\theta(i,j)$ is embedded in all coefficients from the subbands from level 0 by addition:

$$\tilde{I}_0^\theta(i,j) = I_0^\theta(i,j) + \alpha w^\theta(i,j) x^\theta(i,j) \quad (1)$$

where α is the embedding strength and $w^\theta(i,j)$ is a weighing function, which is a half of the quantization step $q_l^\theta(i,j)$.

The quantization step of each coefficient is computed by the authors in [3] as the weighted product of three factors:

$$q_l^\theta(i,j) = \Theta(l,\theta)\Lambda(l,i,j)\Xi(l,i,j)^{0.2} \quad (2)$$

and the embedding takes place only in the first level of decomposition, for l = 0.

The first factor is the sensitivity to noise depending on the orientation and on the level of detail:

$$\Theta(l,\theta) = \begin{Bmatrix} \sqrt{2}, & \theta = 1 \\ 1 & otherwise \end{Bmatrix} \cdot \begin{Bmatrix} 1.00 & l=0 \\ 0.32 & l=1 \\ 0.16 & l=2 \\ 0.10 & l=3 \end{Bmatrix} \quad (3)$$

The second factor takes into account the local brightness based on the gray level values of the low pass version of the image (the 4th level approximation image):

$$\Lambda(l,i,j) = 1 + L'(l,i,j) \quad (4)$$

where

$$L'(l,i,j) = \begin{Bmatrix} 1 - L(l,i,j), & L(l,i,j) < 0.5 \\ L(l,i,j), & \text{otherwise} \end{Bmatrix} \quad (5)$$

and

$$L(l,i,j) = \frac{1}{256} I_3^3\left(1 + \left\lfloor \frac{i}{2^{3-l}} \right\rfloor, 1 + \left\lfloor \frac{j}{2^{3-l}} \right\rfloor\right)$$

The third factor is computed as follows:

$$\Xi(l,i,j) = \sum_{k=0}^{3-l} \frac{1}{16^k} \sum_{\theta=0}^{2} \sum_{x=0}^{1} \sum_{y=0}^{1} \left[I_{k+l}^\theta\left(y + \frac{i}{2^k}, x + \frac{j}{2^k}\right)\right]^2 \cdot \mathrm{Var}\left\{ I_3^3\left(1 + y + \frac{i}{2^{3-l}}, 1 + x + \frac{j}{2^{3-l}}\right)\right\}_{\substack{x=0,1 \\ y=0,1}} \quad (6)$$

and it gives a measure of texture activity in the neighborhood of the pixel. In particular, this term is composed by the product of two contributions; the first is the local mean square value of the DWT coefficients in all detail subbands, while the second is the local variance of the low-pass subband (the 4th level approximation image). Both these contributions are computed in a small 2 × 2 neighborhood corresponding to the location (*i*, *j*) of the pixel. The first contribution can represent the distance from the edges, whereas the second one the texture. This local variance estimation is not so precise, because it is computed with a low resolution. We propose another way of estimating the local standard deviation. In fact, this is our figure of merit.

B. Detection

Detection is made using the correlation between the marked DWT coefficients and the watermarking sequence to be tested for presence:

$$\rho = \frac{1}{3MN} \sum_{\theta=0}^{2} \sum_{i=0}^{M-1} \sum_{j=0}^{N-1} \tilde{I}_0^\theta(i,j) x^\theta(i,j) \quad (7)$$

The correlation is compared to a threshold *T*, computed to grant a given probability of false positive detection, using the Neyman-Pearson criterion. For example, if $P_f \le 10^{-8}$, the threshold is $T = 3.97\sqrt{2\sigma_\rho^2}$, with σ_ρ^2 the variance of the wavelet coefficients, if the image was watermarked with a code Y other than X:

$$\sigma_\rho^2 \approx \frac{1}{(3MN)^2} \sum_{\theta=0}^{2} \sum_{i=0}^{M-1} \sum_{j=0}^{N-1} \left(\tilde{I}_0^\theta(i,j)\right)^2 \quad (8)$$

IV. IMPROVED PERCEPTUAL MASK

Another way to generate the third factor of the quantization step is by segmenting the original image, finding its contours, textures and regions with high homogeneity. The criterion used for this segmentation can be the value of the local standard deviation of each pixel of the host image. In a rectangular moving window N(k, l) containing M·M pixels, centered on each pixel y(k, l) of the host image, the local mean is computed with:

$$\hat{\mu}(k,l) = \frac{1}{M \cdot M} \sum_{y(i,j) \in N(k,l)} y(i,j) \quad (9)$$

and the local variance is given by:

$$\hat{\sigma}^2(k,l) = \frac{1}{M \cdot M} \sum_{y(i,j) \in N(k,l)} \left(y(i,j) - \hat{\mu}(k,l)\right)^2 \quad (10)$$

Its square root represents the local standard deviation. For example, the image Barbara is segmented in classes whose elements have a value of the

normalized local standard deviation, belonging to one of six possible intervals $I_p = (\alpha_p, \alpha_{p+1})$, $p = 1,\ldots,6$, where $\alpha_1=0$, $\alpha_2=0.025$, $\alpha_3=0.05$, $\alpha_4=0.075$, $\alpha_5=0.1$, $\alpha_6=0.25$, $\alpha_7=1$ (Fig.2-7).

This image (Fig.1) was selected for its rich content. It contains a lot of contours, textures and zones with high homogeneity. In each of the Fig. 2-7 is represented the class corresponding to the interval I_p, $p = 1,\ldots,6$, the elements of the other classes being ignored (represented in black). These figures prove the good quality of the segmentation based on the local standard deviation values. Such images can be used like masks for the embedding in the wavelet detail coefficients. The quantization step for a considered coefficient is given by a value proportional with the local standard deviation of the corresponding pixel from the host image.

To assure this perceptual embedding, the dimensions of different detail sub-images must be equal with the dimensions of the corresponding masks. So, the local standard deviation image must be compressed. The compression factor required for the mask corresponding to the l^{th} wavelet decomposition level is 4^{l+1}, with $l=0,\ldots,3$. This compression can be realized with the aid of the DWT. To generate the mask required for the embedding into the detail sub-images corresponding to the l^{th} decomposition level, the DWT of the local standard deviation image is computed (making $l+1$ iterations). The approximation sub-image obtained represents the compression result (the mask required). This type of compression is illustrated in the Fig. 8-11.

The unique difference between the watermarking method proposed in this paper and the one presented in section 3, is given by the computation of the local variance – the second term – in (6). To obtain the new values of the texture, the local variance of the image to be watermarked is computed, using the relations (9) and (10). The local standard deviation image is decomposed using one iteration wavelet transform, and only the approximation image is kept. A scheme is provided in Fig.13. Some practical results of the new watermarking system are reported in the next paragraph.

V. EVALUATION OF THE METHOD

To asses the validity of our algorithm, we give in Fig. 14-17 the results for JPEG compression. The image Barbara is watermarked with various embedding strengths α. The watermarked Barbara for $\alpha=1.5$ is shown in Fig.12. The binary watermark is embedded in all the detail wavelet coefficients of the first resolution level using eq. (1) to (5). Each watermarked image is compressed using the JPEG standard, for six different quality factors: 5, 10, 15, 20, 25, 50.

We choose to show in Fig. 14 & 15 only the ratio ρ/T, as a function of the peak signal-to-noise ratio (PSNR) between the marked (un-attacked) image and the original one, and respectively as a function of α.

For each PSNR and each compression quality factor Q, the correlation ρ and the threshold T are computed. The probability of false positive detection is set to 10^{-8}. The effectiveness of the proposed watermarking system can be measured using the ratio ρ/T. If this ratio is greater than 1 then the watermark can be extracted.

Analyzing Fig. 14, it can be observed that the watermark can be extracted for a large PSNR interval and for a large interval of compression quality factors. For PSNR values higher than 30 dB, the watermarking is invisible. For compression quality factors higher or equal than 25 the distortion introduced by JPEG compression is tolerable. For all values of the PSNR from 30 dB to 35 dB, of practical interest, the watermark can be extracted for all the significant compression quality factors (higher or equal than 25). So, the proposed watermarking method is of high practical interest.

Fig. 15 shows the dependency of the ratio ρ/T on the embedding strength α in case of JPEG compression. Increasing the embedding strength, the PSNR of the watermarked image decreases, and the ratio ρ/T increases.

The ratio ρ/T decreases for higher embedding strengths and for higher compression ratios (Fig.14) or lower embedding strengths (Fig.15). The watermark is still detectable even for very small values of α. For the quality factor Q=5 (or a compression ratio CR=32), the watermark is still detectable even for $\alpha=0.5$.

Fig.16 shows the detection of a true watermark for various quality factors, in the case of $\alpha=1.5$; the threshold is well beyond the detector response.

Finally the selectivity of the watermark detector used is illustrated in Fig. 17, when a number of 1000 different marks were tested. The second highest detector response is shown together with the threshold value, for each quality factor. We can see that false positives are rejected.

In Table 1 we give a comparison between our method and Barni et al method [3]. This time, the algorithm was tested on the Lena image, for $\alpha=1.5$ and a JPEG compression with a quality factor of 5, which yields into a compression ratio of 46. P_f was set to 10^{-8}. We give the detector response for the original embedded watermark ρ, the detection thresold T, and the second highest detector response ρ_2. P_f was set to 10^{-8} and 1000 marks were tested. The detector response is higher than in the case of the method in [3].

VI. CONCLUSION

We have proposed a new type of pixel-wise masking, based on the local standard deviation of the original image. Wavelet compression was used in order to obtain a texture subimage of the same size with the subimages where the watermark is inserted. We tested the method against compression, and found out that it works better than the method proposed in [3]. Future work will involve testing the new mask on a large

image database and possibly look into using lower resolution levels for embedding, in order to increase robustness.

REFERENCES

[1] G. Voyatzis, I. Pitas, "Problems and Challenges in Multimedia Networking and Content Protection," *TICSP Series* No. 3, Editor Iaakko Astola, March 1999.

[2] I. Cox, M. Miller, J. Bloom, *Digital Watermarking*, Morgan Kaufmann Publishers, 2002.

[3] M. Barni, F. Bartolini, A.Piva; "Improved wavelet-based watermarking through pixel-wise masking," *IEEE Transactions on Image Processing,* Volume 10, Issue 5, May 2001, pp.783 – 791.

[4] C. Nafornita, , A. Isar, M. Borda, "Image Watermarking Based on the Discrete Wavelet Transform Statistical Characteristics," Proc. of *IEEE EUROCON 2005*, The International Conference on "Computer as a tool", November 21-24, 2005, Belgrade, Serbia & Montenegro, pp. 943-946.

[5] D. Kundur, "Water-filling for Watermarking?," Proc. IEEE Int. Conf. On Multimedia and Expo, New York City, New York, pp. 1287-1290, August 2000.

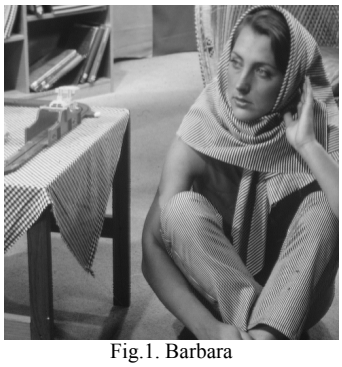


Fig.1. Barbara

Fig. 2. The class corresponding to the interval I_6, contains the contours and the larger textures.

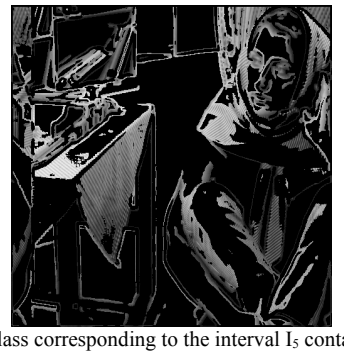


Fig. 3. The class corresponding to the interval I_5 contains contours and textures.

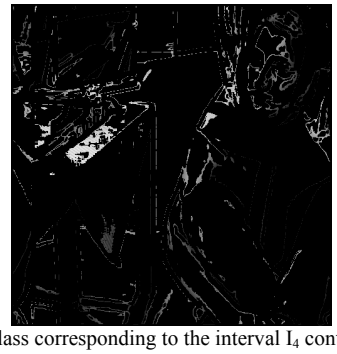


Fig. 4. The class corresponding to the interval I_4 contains textures.

Fig. 5. The class corresponding to the interval I_3 contains textures.

Fig. 6. The class corresponding to the interval I_2 contains textures.

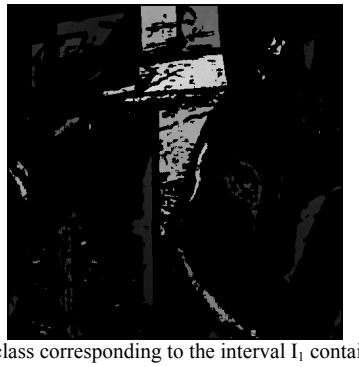


Fig. 7. The class corresponding to the interval I_1 contains the high homogeneity zones.

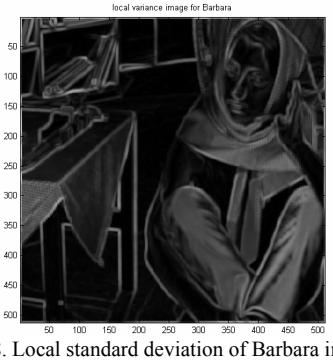


Fig. 8. Local standard deviation of Barbara image.

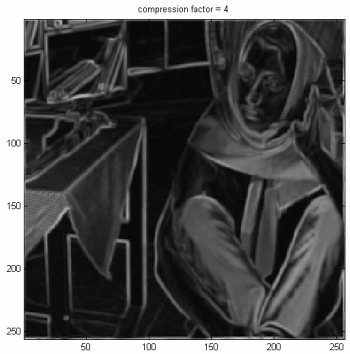


Fig. 9. The last image compressed with CR = 4.

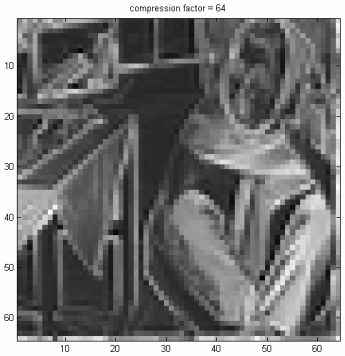


Fig. 11. The image in Fig. 8 compressed with a CR = 64.

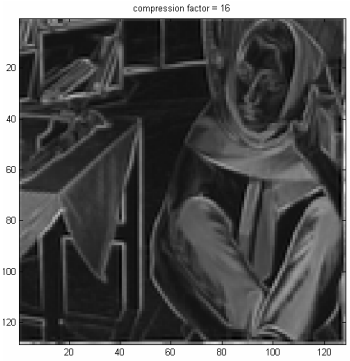


Fig. 10. The image in Fig. 8 compressed with CR = 16.

Fig.12 Watermarked Barbara image with $\alpha = 1.5$.

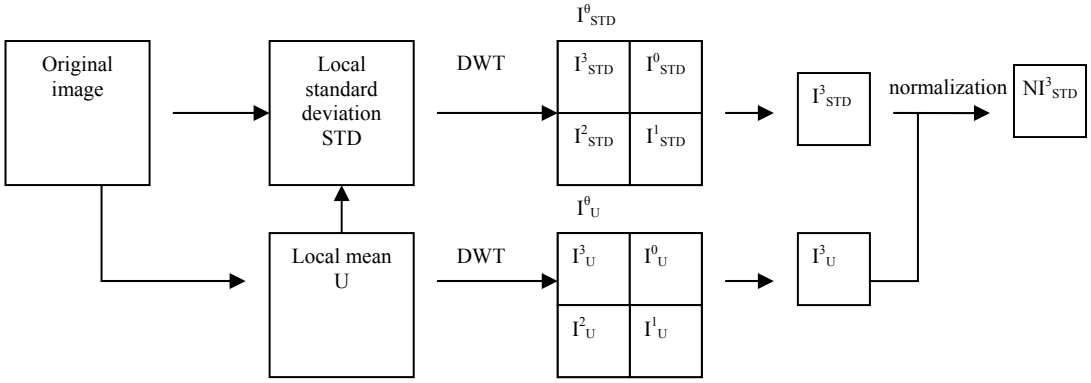


Fig.13: A general scheme for obtaining the texture mask.

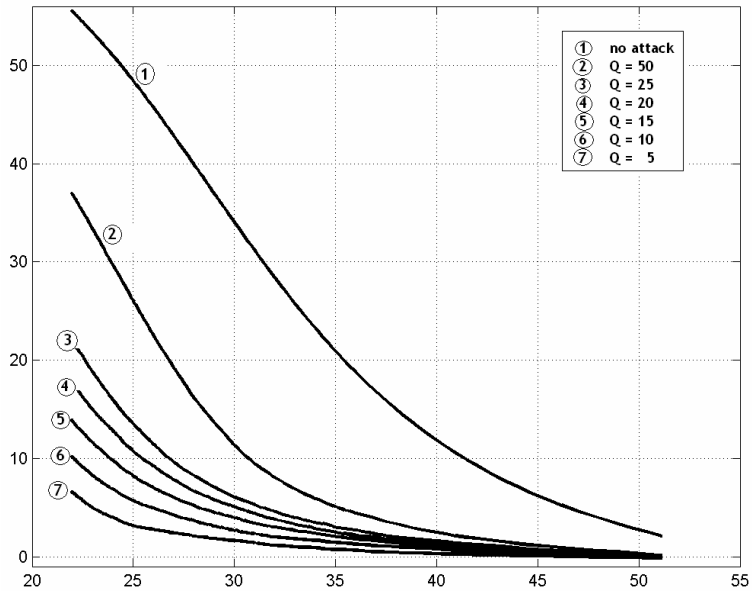


Fig. 14. The ratio ρ/T as a function of the PSNR between the marked and the original images, for different quality factors (JPEG compression). P_f is set to 10^{-8}.

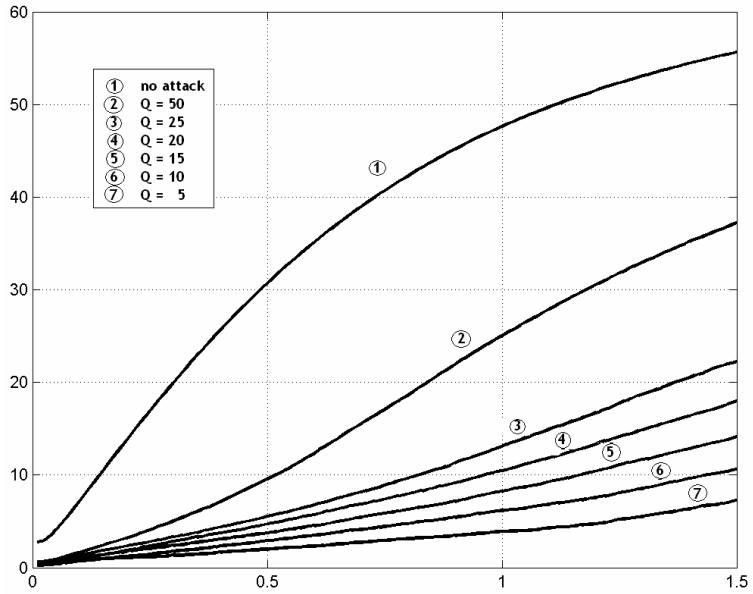


Fig. 15. The ratio ρ/T as a function of the embedding strength, for different quality factors (JPEG compression). P_f is set to 10^{-8}.

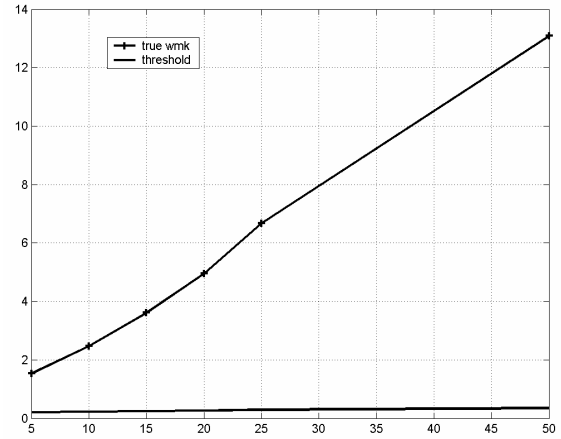


Fig. 16: Detector response ρ, and threshold T, as a function of different quality factors (JPEG compression). The watermark is successfully detected. P_f is set to 10^{-8}.

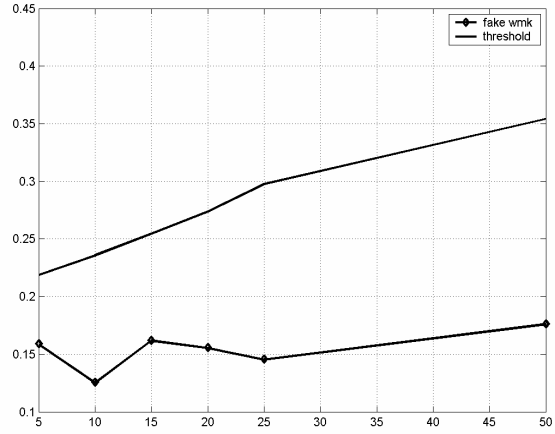


Fig. 17: Highest detector response, ρ_2, corresponding to a fake watermark and threshold T. The threshold is above the detector response.

Table 1. A comparison between Barni et al method and the proposed one.

CR = 46, JPEG compression	Our method	Barni et al [3] method
ρ	0.3199	0.038
T	0.0844	0.036
ρ_2	0.0516	0.01

Buletinul Ştiinţific al Universităţii "Politehnica" din Timişoara

Seria ELECTRONICĂ şi TELECOMUNICAŢII
TRANSACTIONS on ELECTRONICS and COMMUNICATIONS

Tom 51(65), Fascicola 2, 2006

Real-Time Process Monitoring in Operating System Linux

Zdenek Slanina, Vilem Srovnal[1]

Abstract – The article deals with a design of system module for the selected processes monitoring in the operating system RT-Linux. The designed module will be able to observe states of selected processes in real-time (start, stop, interruption ...) and visualize changes of states on the remote Linux system. For the better explanation of problems are given basic characteristics of operating systems Linux and RT-Linux. There are described the initiate problems solution, process states monitoring and time sequence of task processing in real time.

Keywords: Linux, RT-Linux, process, scheduling, monitoring, embedded systems, real-time systems

I. INTRODUCTION

The present technological processes control uses number of technical resources as intelligent sensors, microcontrollers, PLC's, personal computers and workstations. The communication between resources is realized by different types of industrial buses, computer networks and operating systems.
If control systems are realized with personal computers, these computers demand the real-time processing mode. There are required the preemptive multi-processing of concurrent tasks or the pseudo-parallel technique of processing.
The real-time operation systems are ready to process external events any time. The processing of demanded solutions is obtained in prior given intervals. The system can accept data as casual events or data are periodically scanned in intervals in advance with respect of appropriate application.
Real-time systems have to react at signals from external environment, events, according to given time pre-limits. The proper behavior of such system depends not only on evaluation's results executed by processes, but also on the elapsed time for their evaluation. The delayed reaction need not to be up-to-date for the appropriate control action, the delay can cause crash of the corresponding application [4], [5].
The design of control system needs the knowledge of its behavior in many situations as standard or emergency and so on. The creation of monitoring kernel module is very useful for the system debugging and error detection especially in the real-time processing.

II. OPERATING SYSTEM LINUX

Basic description

The operating system Linux is obtainable in the form of free distributed implementation of UNIX kernel [1]. This is the base of lowest operating system level. The operating system core is compiled and installed on the computer with many specific free distributed programs, which make possible to design the complex operating system. Such installations are called Linux systems while kernels are not unique. The complicate installation originates a Linux distribution [2].
Distributions are realized by various mediums (floppy, CD). There are combined kernels and many next support programs, programming languages and utilities. The X-Windows server is involved as graphical user interface of UNIX systems too.
The kernel is the crucial part of each operating system. Linux kernel is compiled by several important subsystems (modules), which are briefly described below. The created interface between the user, operating system and hardware is shown on the figure 1.
Files and devices are controlled by the small number of functions in Linux. These functions are called as system calls. They are Linux components and make interface between the operating system and applications [3].
The problem is the efficiency direct using of these functions for inputs and outputs. The performance of system goes down as a result of switching between user and kernel mode all the time. Function's libraries are used scores of time. It is possible to use the function, which is dedicated directly to work with the specific device. Linux provides a range of standard libraries as the sophisticated interface for devices and disc files.

Kernel modules

Virtual File System (VFS) creates the universal interface for the using of various file systems. The each type of file systems provides the implementation of specific set of operations, which are common for all file systems.

[1] Department of Measurement and Control, VSB Technical University of Ostrava, 17. listopadu 15, 708 33 Ostrava-Poruba, Czech Republic, e-mail zdenek.slanina@vsb.cz

If any system component sends the request to use the one of file systems, its request goes through VFS. VFS forwards it to the relevant file system driver. VFS provides the user interface both for file systems (FAT, ext2...) and devices. The kernel provides the unified interface for user applications.
Devices include partly hardware devices (hard disc, tape memory...) partly software devices (/dev/random - device for generating of random data...). Special services require networks. While these services are non-standard (different then for file systems), they belong to the VFS too. Users communicate as with network devices as with standard devices.

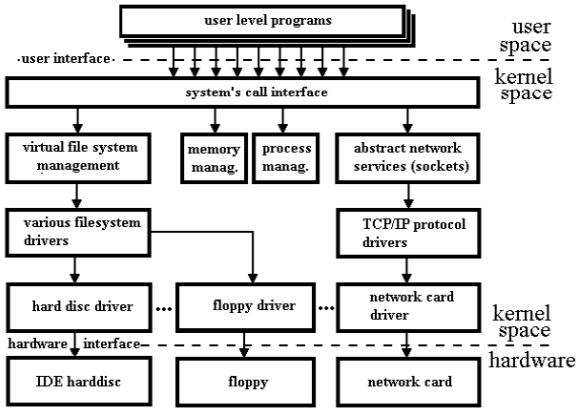


Fig. 1. Amplitudes in the standing wave

The memory manager provides following functions:

- Virtual address space – the operating system provides the virtual memory. The size of virtual memory is much greater than the size of physical memory in the system.
- Memory protection - each process in the system has its own virtual address space. Virtual address spaces are completely mutually separated. The running application process doesn't affect other processes. The hardware mechanism of virtual memory protects relevant memory areas against writing. The code and data are protected in the memory against destructive operations of other applications.
- Memory mapping – the memory mapping serves for mapping program's images and data files to the address space of the process. When the memory mapping is used the content of file is directly linked with the virtual address space of the process.
- Physical memory allocation – the memory manager subsystem allows each running process to allocate appropriate part of the system physical memory.
- Virtual memory sharing – while the virtual memory allocate to the process the separated address space, within the running of processes are situations when processes need to share virtual memory among themselves. Dynamic libraries are the one example of sharing code by several processes. The shared memory is also a buffer, which is used in the interprocesses communication when information is exchanged among processes. The Linux supports the interprocesses communication by using UNIX system V IPC mechanism.

Linux provides the virtual memory system as the extension of RAM memory. The efficient size of memory is much greater. The kernel swaps contents of just unused memory blocks on the disc and releases memory for other functions. If it is requested the content of blocks is loaded back to the memory. These operations are the fully transparent for users. Running Linux programs allocate only the appropriate size of accessible physical memory and don't take care of the virtual disc space. Of course, disc operations are not as quick as on the physical memory - RAM.
Linux use the plain file or the special disc area for swapping. The advantage of independent disc segment is speed. The advantage of swap file is possibility to change size of swap space simply. If the size of swap space is known, then is better using a disc segment. In case of no direct demands is better using a swap file. Linux provides multiple usages of swap areas or swap files.
The process management module control multitasking. It concerns the creating of processes and switching processor among active processes. Linux threads implementation is called one-to-one executed at kernel level. Each thread means independent process for the kernel. The scheduler of processes doesn't make differences between processes and threads. Disadvantage of this model is too big overhead through threads switching. P-thread library is provided for threads, which are implementing in agreement with POSIX standard.
The data structure *task_struct* enable the process management in Linux. The terms task and process are equivalent in Linux. The *task_struct* describes properties and states of processes in the system. These data structures create the task vector, which is the array of pointers to all structures *task_struct* in the system. It means that maximal number of processes is limited by the size of task vector (512 items implicitly). The new structure *task_struct* is allocated in the memory as the part of vector task during a process creation. There is possible a reference by the current pointer to the actual process for the searching facilitation. Individual items of *task_struct* are separated to several areas.
The first is a state - the state of the process changes according to processing conditions. Processes in Linux are found in following states:

- RUNNING - process is running now (actually process) or it is ready to run (process waiting for the processor).

- INTERRUPTIBLE - process is waiting for processing and it is wake up by signal or timer expiration.
- UNINTERRUPTIBLE - process waiting for processing and it can't be wake up.
- ZOMBIE - finished process with structure *task_struct* in vector task by any reason. This process is inactive.
- STOPPED - process was stopped by any signal usually. In that state is a debugged process by example.
- EXCLUSIVE – this state is created as a logical combination of states with state INTERRUPTIBLE and UNINTERRUPTIBLE.

The Linux and Unix use in the file descriptors attributes for the unauthorized access protection. Each file and directory has its owners. Attributes define the access right for user (owner), group and anyone. The basic file protection defines other three protection bits as rights for read, write and execution. Each group of users can have another access rights. For example, the owner can read and write in the file, group can read only and all other users (processes) have the access to the file disabled.
The group definition enables to assign privileges to the groups of users, not only to one user or all users in the system. The right for process execution is possible assigned to number of groups (maximal number is 32 implicitly). These groups are saved to the group's vector in the structure *task_struct* of each process. If a group has access rights to a file and the process belongs to this group, then the process has group rights to the file.
There are user and group pairs of process attributes in the structure above:

- *uid* and *gid* - identifiers of user and group in the name of user running process
- *effective uid* and *gid* - some processes change their *uid* and *gid* within running process, their own are saved as attributes in inods of executing image. These processes are called setuid processes and they are very useful because they present way to restrict access to services executed by name of any other as network daemons. Effective *uid* and *gid* are set according to attributes of setuid process, values of *uid* and *gid* are unchanged. Effective *uid* and *gid* use the kernel for checking of access rights.
- *filesystem uid* and *gid* - similar to effective *uid* and *gid*. They are used for access rights checking to file system. It is necessary for connected file systems, when NFS server in user mode need access to files as some process. In this case, filesystem *uid* and *gid* are changed instead of effective *uid* and *gid*. This way eliminates the situation when some sends to the server the kill signal. Kill signals are submitted to processes with effective *uid* and *gid*.
- *saved uid* and *gid* - values required by POSIX standard and they are used in processes changing *uid* and *gid* of the process using system calls. When values of *uid* and *gid* are changed, real values of *uid* and *gid* are saved in them.

Each process has its process identifier. Identifier is not an index in the task vector, it is only a number. In the Linux there is no system process to depend on any other processes. All processes have their generic processes excluding the initial process. Each *task_struct* structure of each process contains a pointer to its generic process and siblings (rest processes with the same generic process) and pointers to its descent processes. Moreover all processes in the system are related in the both directions list, its root is *task_struct* structure of init process. The kernel uses this list to the view above all processes in the system.
The kernel keeps information about time of process starting and the total processor time of process. The kernel keeps also values, which processing time is the process in the system and user mode. The Linux supports interval timers of processes. Process can call set timers using system calls to call signal after time period is expired. These timers can be one-off or periodic.
All processes run partly in the user mode and partly in the system mode. These modes are supported by the low-level hardware. There is a specific security mechanism for the switching between user and system mode. In the user mode, a process has obviously minor privileges than in the system mode.
Always when system calls are used, the processing is switched from the user mode to the system mode. The kernel works in the name of process in the time of system mode. Linux uses preemptive tasks planning. The one of planning strategies is round-robin. The each process is running a set time (for example 200 ms). When this time expired, other process use processor and previous process has to wait for the next opportunity to run. This time period is called time-slice.
The scheduler decides which process will run. Linux scheduler selects the actual processes on the base of priority algorithm. The scheduler saves the actual process status, values of processor registries and other context information to data structure *task_struct* when the new process is choosing. Then the scheduler restores the state of new planned process. The scheduler keeps following information in structure *task_struct* of each process for a realization of planning strategies:

- *policy* - scheduling strategy is associated to the relevant process. There are two types of processes in Linux - standard and real-time. Real-time processes have higher priorities than all other processes. If real-time process

is ready to go, it will be run. Two strategies are applied for realtime processes either round-robin or FIFO (First In First Out). In round-robin scheduling is used the cyclic switching of processes. They are executed cyclic in queue. The strategy FIFO means execution of processes in the order of ready to execution.

- *priority* – the scheduler assigns the priority to the process. It is a quantity of the time (in jiffy units), that process can use, when it is running. The priority of processes is possible to change using system calls and with renice command.
- *rt_priority* - Linux supports real-time processes with high priority than other processes in the system. This item allows to scheduler assign to each process its relative priority. The priority of real-time processes is possible to change using system calls.
- *counter* - number of time jiffy when the process is running. At the process planning is this value set as the priority value. The counter is decremented with every time pulse.

The scheduler is activated in several points in the kernel: actual process is transferred to queue of waiting processes; system call is finished; before the switching of process from system to the kernel mode. The next reason is the decrement of counter value to zero.

Process selection to execution – the scheduler looks in the priority queue of processes. If the realtime process is in the queue, its rate is higher than standard processes. The weight of standard process is equal to the counter value. The real-time process weight is 1000 higher. It means that real-time processes will execute before standard processes. The actual process, which is running (value counter is decremented) has handicap before other processes with the same priority. When priorities of processes are equal, the scheduler chooses the first process in the queue. The actual process is scheduled for the end of queue at the switching. Processes are executed one by one in the balanced system with same priorities of processes. It is round-robin planning – the cyclic planning of processes. The sequence of waiting processes is possible to change.

Process switch - if switching conditions occurs, the actual process is stopped and the new process is ready to run. The running process uses registers and processor and system memories. The every call of routines sets parameters in registers and use values in the stack, for example, to save a return address of calling routine. If the process is suspending, it is necessary to save its state including the program counter and all registers of processor to its *task_struct* structure. Then the state of new planned process is necessary to restore. This operation is a machine dependent, each processor use an own way with the hardware support.

The process switch is the last scheduler's operation. The saved context of previous process is image of hardware context in time of end of process scheduling. So when is loaded a new process, there are know information about the situation before, including the content of counter of instructions and registers.

III. OPERATING SYSTEM RT-LINUX

There are two different approaches to obtain RT tasks executing in Linux:

1. Improving the Linux kernel preemption.
2. Adding a new software layer beneath Linux kernel with full control of interrupts and processor key features.

These two approaches are known as "*preemption improvement*" and "*interrupt abstraction*" respectively. This second approach is the one used by RTLinux.

RT-Linux scheduler uses Linux kernel as its inactive task. Linux is running in the case that no realtime process in the real-time mode is active. The process in Linux unblocks interruption or prevent switch in any time. This mechanism is possible thanks to the software emulation of hardware interruption.

There are important features, which are achieved in real-time processing in the kernel mode:

- Thread processing is in the operating memory of kernel.
- Threads processing is in the kernel mode and threads have complete access to basic layer.
- Application is compiled and installed in the same memory space like real-time operating system. System calls are implemented using simple system call that doesn't use software interruption by the reason of decrement time of operating system overhead.

RT-Linux is following the POSIX 1003.13 minimal realtime operating system standard. The design of RT-Linux is subordinated to POSIX requirements. The system can run on i386, PPC and ARM architectures.

Following tools are provided for applications debugging:

- Debugging at source code level with SMP support at the target machine, cross-debugging is not possible.
- Tracing - kernel tracing and application events.
- POSIX tracing.

Memory management:

- static

- dynamic - dynamic memory allocation is not available (functions malloc and free); RT-Linux doesn't allows it nor use internally
- protected address space - application threads and RT-Linux threads run at same address space; by some point of view is Linux host system for RTLinux; Linux has complete control above system memory

Interprocess communication:

- FIFO - communication mechanism is determined to communication between real-time processes and Linux user processes (not compatible with POSIX norm)

Synchronization:

- mutexes - POSIX mutexes; system allows PRIORITY_PROTECT protocol for handling with priority inversion problem
- condition variables - POSIX condition variables
- semaphores - POSIX semaphores

Nowadays are developed various new components for RT-Linux for more effective work with, for measurement and control is Linux interface Comedi, etc.

IV. MONITORING MODULE

The basic goal of monitoring module is the maximal usage of data structure *task_struct* contains all information about processes in system Linux. Then the process monitoring is possible separate to two basic parts.

The first part is a process status, which is read from the task vector. This procedure has to be very fast. In the case when this procedure is integrated with scheduler, it will have following consequences. Each reading of status evokes a delay of process switching; it can be a problem in real-time systems. The effective processor time is decreased as a result of following actions: switch for status reading; own reading; compare with table of desired monitored processes; writing of data in case of positive result; switching back to the actual process. On the other side, when the monitoring is finished, the processor will have more system time for other process services.

If the actual process is the one of monitored processes, the status information of actual process is at disposal to the second part of monitoring system. The second part is a visualization system, which is running slowly. Changes in kernel are very quick and in the case of exact visualization, it is not scan able by the operator. The second part writes data in the given format on the appropriate device. It is possible use as the device a monitor, hard disk or Ethernet. In the case of remote visualization and unavailability of devices above is possible to create the special device for direct connection with PC buses (PCI, PC/104 ...).

The format of written data is depended on the used device, for example, in the memory medium (hard disk) is saved names of processes and times. For the visualization on the monitor is used the one of graphic libraries (GtK).

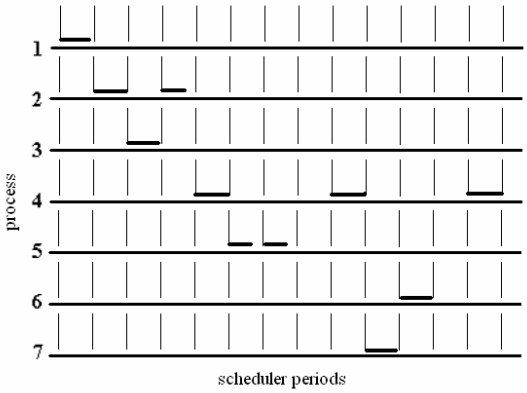


Fig. 2. The monitoring example

The monitoring example is shown on the figure 2. There is an example of processes which are scanning by the monitoring module. There are processes with numbers (1 to 7). Processes 2 to 6 are application processes which are debugging. Processes 1 and 7 are system processes, e.g. drivers for measurement cards, etc. The monitoring module allows saving time data to the file. It is possible analyze a system behavior after the system halt: events in system; exact times of input or output events; times of processing; feedback reactions. It is possible to visualize a behavior on the remote computer. The block diagram of monitoring system is shown on the figure 3.

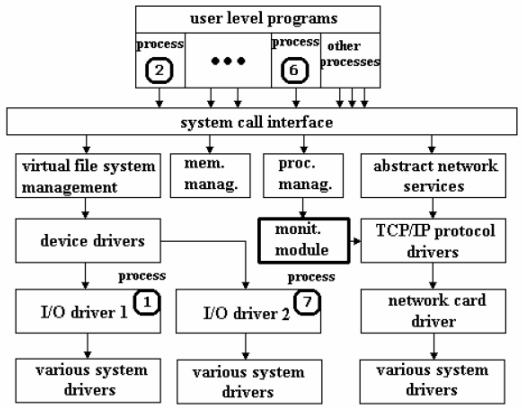


Fig. 3. The block diagram of process monitoring

V. CONCLUSION

The main goal of this project is a support of embedded systems design. The designer obtains information of process behavior in the phase of design, testing and real operation. Because in phase of

testing is difficult catch all situations, usage of such system could be expedient.
It is possible a testing if the chosen hardware is suitable for the real-time application or it is necessary use a more powerful hardware.

Acknowledgement: This work was supported by the Ministry of Education of the Czech Republic under Project 1M0567.

REFERENCES

[1] Sobell M.G.: *A practical guide to Linux*, Addison-Wesley 1997
[2] Matthew N., Stones R.: *Beginning Linux programming*, Wrox Press 2000
[3] Rubini A., Corbet J.: *Linux device drivers*, Computer Press 2001
[4] Srovnal, V.: *Operating Systems for Real-time Control*, VŠB Technical University of Ostrava 2003 (In Czech)
[5] Kocis T., Srovnal V.: *Operating Systems for Embedded Computers*. In : Programmable Devices and Systems 2003-IFAC Workshop, Pergamon Press-Elsevier 2003, pp. 359-364

Buletinul Ştiinţific al Universităţii "Politehnica" din Timişoara

Seria ELECTRONICĂ şi TELECOMUNICAŢII
TRANSACTIONS on ELECTRONICS and COMMUNICATIONS

Tom 51(65), Fascicola 2, 2006

Single Microphone Noise Canceller Based on a Robust Adaptive Kalman Filter

Marcel Gabrea[1]

Abstract – This paper deals with the problem of speech enhancement when a corrupted speech signal with an additive noise is the only information available for processing. Kalman filtering is known as an effective speech enhancement technique in which speech signal is usually modeled as autoregressive (AR) process and represented in the state-space domain. In the above context, all the Kalman filter-based approaches proposed in the past operate in two steps: they first estimate the noise and the driving variances and parameters of the signal model, then estimate the speech signal. This paper presents an alternative solution that does not require the explicit estimation of noise and driving process variances. This deals with a new formulation of the steady-state Kalman filter gain estimation based on the use of external description of systems. Unlike the conventional approaches, no suboptimal Kalman filter is needed here.
Keywords: speech enhancement, Kalman filtering, noise reduction.

I. INTRODUCTION

Speech enhancement using a single microphone system has become an active research area for audio signal enhancement. The aim is to minimize the effect of noise and to improve the performance in voice communication systems when input signals are corrupted by background noise.

Kalman filtering is known as an effective speech enhancement technique, in which speech signal is usually modeled as autoregressive (AR) process and represented in the state-space domain.

Many approaches using Kalman filtering have been referenced in the literature. They usually operate in two steps: first, noise and driving process variances and speech model parameters are estimated and second, the speech signal is estimated by using Kalman filtering. In fact these approaches differ only by the choice of the algorithm used to estimate model parameters and the choice of the models adopted for the speech signal and the additive noise.

Paliwal and Basu [1] have used estimates of the speech signal parameters from clean speech, before being contaminated by white noise. They then used a delayed version of Kalman filter in order to estimate the speech signal.

In [2], Oppenheim et al. have used a time-adaptive algorithm to adaptively estimate the speech model parameters and the noise variance.

Gannot et al. [3] have proposed the use of the EM algorithm to iteratively estimate the spectral parameters of speech and noise parameters. The enhanced speech signal was obtained as a byproduct of the parameter estimation algorithm.

Lee and Jung [4] have developed a time-domain approach, with no a priory information, to enhance speech signals. The autoregressive-hidden filter model (AR-HFM) with gain contour was proposed for modeling the statistical characteristics of the speech signal. The EM algorithm was used for signal estimation and system identification. In the E-step, the signal was estimated using multiple Kalman filters with Markovian switching coefficient and the probability was computed using the Viterbi Algorithm (VA). In M-step, the gain contour and noise parameter were recursively updated by an adaptive algorithm.

Grivel et al. [5] have suggested that the speech enhancement problem can be stated as a realization issue in the framework of identification. The state-space model was identified using a subspace non-iterative algorithm based on orthogonal projection.

Gabrea and O'Shaughnessy [6] have proposed estimating the noise and driving process variances using the property of the innovation sequence, obtained after a preliminary Kalman filtering with an initial gain.

The methods proposed in [7] and [8] avoid the explicit estimation of noise and driving process variances by estimating the optimal Kalman gain. After a preliminary Kalman filtering with an initial sub-optimal gain, an iterative procedure is derived to estimate the optimal Kalman gain using the property of the innovation sequence.

In this paper a quite different and simple approach to the estimation of the steady-state optimal Kalman filter gain based on the use of external description of systems is presented. This method avoids the explicit estimation of noise and driving process variances by estimating the optimal Kalman gain. Unlike the conventional approaches, no suboptimal Kalman filter is needed here. Thus, the divergence problem of the

[1] École de technologie supérieure, Département de génie électrique,
1100 Notre-Dame Ouest, H3C 1K3 Montréal, e-mail mgabrea@ele.etsmtl.ca

Kalman filter does not occur. The performance of this algorithm is compared to the one of alternative speech enhancement algorithms based on the Kalman filtering. A distinct advantage of the proposed algorithm is that no voice activity detector (VAD) is required to estimate noise variance. Another advantage of this algorithm compared to [7] and [8] is the superiority in terms of computational load. An iterative procedure is not required in the steady-state optimal Kalman gain estimation.

This paper is organized as follows. In Section II we present the speech enhancement approach based on the Kalman filter algorithm. Section III is concerned with the estimation of AR parameters and optimal Kalman gain. Simulation results are the subject of Section IV.

II. NOISY SPEECH MODEL AND KALMAN FILTERING

The speech signal $s(n)$ is modeled as a p^{th} order AR process:

$$s(n) = \sum_{i=1}^{p} a_i(n)s(n-i) + u(n) \tag{1}$$

$$y(n) = s(n) + v(n) \tag{2}$$

where $s(n)$ is the n^{th} sample of the speech signal, $y(n)$ is the n^{th} sample of the observation, $a_i(n)$ is the i^{th} AR parameter, $u(n)$ and $v(n)$ are uncorrelated Gaussian white noise sequences with zero means and the variances $\sigma_u^2(n)$ and $\sigma_v^2(n)$.

This system can be represented by the following state-space model:

$$\boldsymbol{x}(n) = \boldsymbol{F}(n)\boldsymbol{x}(n-1) + \boldsymbol{G}u(n) \tag{3}$$

$$y(n) = \boldsymbol{H}\boldsymbol{x}(n) + v(n) \tag{4}$$

where:

1. $\boldsymbol{x}(n) = \begin{bmatrix} s(n-p+1) & \cdots & s(n) \end{bmatrix}^T$ is the $p \times 1$ state vector
2. $\boldsymbol{F}(n)$ is the $p \times p$ transition matrix

$$\boldsymbol{F}(n) = \begin{bmatrix} 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & 1 \\ a_p(n) & a_{p-1}(n) & a_{p-2}(n) & \cdots & a_1(n) \end{bmatrix}$$

3. $\boldsymbol{H} = \boldsymbol{G}^T = \begin{bmatrix} 0 & 0 & \cdots & 0 & 1 \end{bmatrix}$ is the $1 \times p$ observation row vector and the input vector.

The standard Kalman filter [9][10] provides the updating state-vector estimator equations:

$$e(n) = y(n) - \boldsymbol{H}\hat{\boldsymbol{x}}(n/n-1) \tag{5}$$

$$\boldsymbol{K}(n) = \boldsymbol{P}(n/n-1)\boldsymbol{H}^T\left[\boldsymbol{H}\boldsymbol{P}(n/n-1)\boldsymbol{H}^T + \sigma_v^2(n)\right]^{-1} \tag{6}$$

$$\hat{\boldsymbol{x}}(n/n) = \hat{\boldsymbol{x}}(n/n-1) + \boldsymbol{K}(n)e(n) \tag{7}$$

$$\boldsymbol{P}(n/n) = \left[\boldsymbol{I} - \boldsymbol{K}(n)\boldsymbol{H}\right]\boldsymbol{P}(n/n-1) \tag{8}$$

$$\hat{\boldsymbol{x}}(n+1/n) = \boldsymbol{F}(n)\hat{\boldsymbol{x}}(n/n) \tag{9}$$

$$\boldsymbol{P}(n+1/n) = \boldsymbol{F}(n)\boldsymbol{P}(n/n)\boldsymbol{F}^T(n) + \boldsymbol{G}\boldsymbol{G}^T\sigma_u^2(n) \tag{10}$$

where:

1. $\hat{\boldsymbol{x}}(n/n-1)$ is the minimum mean-squares estimate of the state vector $\boldsymbol{x}(n)$ given the past observations $y(1), \cdots, y(n-1)$
2. $\tilde{\boldsymbol{x}}(n/n-1) = \boldsymbol{x}(n) - \hat{\boldsymbol{x}}(n/n-1)$ is the predicted state-error vector
3. $\boldsymbol{P}(n/n-1) = E\left[\tilde{\boldsymbol{x}}(n/n-1)\tilde{\boldsymbol{x}}^T(n/n-1)\right]$ is the predicted state-error correlation matrix
4. $\hat{\boldsymbol{x}}(n/n)$ is the filtered estimate of the state vector $\boldsymbol{x}(n)$
5. $\tilde{\boldsymbol{x}}(n/n) = \boldsymbol{x}(n) - \hat{\boldsymbol{x}}(n/n)$ is the filtered state-error vector
6. $\boldsymbol{P}(n/n) = E\left[\tilde{\boldsymbol{x}}(n/n)\tilde{\boldsymbol{x}}^T(n/n)\right]$ is the filtered state-error correlation matrix
7. $e(n)$ is the innovation sequence
8. $\boldsymbol{K}(n)$ is the Kalman gain

The estimated speech signal can be retrieved as the p^{th} component of the state-vector estimator $\hat{\boldsymbol{x}}(n/n)$. However, the transition matrix and the driving process statistics are unknowns and hence must be estimated. Here a quite different and simple approach to the estimation of the steady-state optimal Kalman filter gain based on the use of external description of systems is used. This method avoids the explicit estimation of noise and driving process variances by estimating the optimal Kalman gain. In this case the Kalman filter equations are:

$$e(n) = y(n) - \boldsymbol{H}\hat{\boldsymbol{x}}(n/n-1) \tag{11}$$

$$\hat{\boldsymbol{x}}(n+1/n) = \boldsymbol{F}(n)\hat{\boldsymbol{x}}(n/n-1) + \boldsymbol{F}(n)\boldsymbol{K}^{opt}e(n) \quad (12)$$

The estimated speech signal can be retrieved from the state-vector estimator:

$$\hat{s}(n) = \boldsymbol{H}\hat{\boldsymbol{x}}(n/n-1) + \boldsymbol{H}\boldsymbol{K}^{opt}e(n) \quad (13)$$

The parameter estimation (the transition matrix and the optimal Kalman gain) is presented in the next section.

III. PARAMETER ESTIMATION

The estimation of the transition matrix, which contains the AR speech model parameters, was made using a adaptation of the robust recursive least square algorithm with variable forgetting factor proposed by Milosavljevic et al. [11]. The estimation of the steady-state optimal Kalman filter gain is based on the external description of the systems.

A. *Estimation of the Transition Matrix*

In our approach, getting $\boldsymbol{F}$ requires the AR parameter estimation. The equation (1) can be rewritten in the form:

$$s(n) = \boldsymbol{x}^T(n-1)\theta(n) + u(n) \quad (14)$$

where:

$$\theta(n) = \begin{bmatrix} a_p(n) & a_{p-1}(n) & \cdots & a_1(n) \end{bmatrix}^T \quad (15)$$

The robust recursive least square approach estimates the vector $\hat{\theta}(n)$ by minimizing the M-estimation criterion [11]:

$$J(n) = \frac{1}{n}\sum_{i=1}^{n} \lambda^{n-i}\rho[\varepsilon^2(i)] \quad (16)$$

where:

$$\psi(x) = \rho'(x) = \min\left[\frac{|x|}{\sigma_u^2(n)}, \frac{\Delta}{\sigma_u(n)}\right] \quad (17)$$

is the Huber influence function and Δ is a chosen constant. The true state vector $\boldsymbol{x}(n)$ used in (14) is unknown but can be approximated by the state-vector estimator $\hat{\boldsymbol{x}}(n/n)$. In this case the robust recursive least square approach gives the estimation equations:

$$\varepsilon(i) = \boldsymbol{H}\hat{\boldsymbol{x}}(i/i) - \hat{\boldsymbol{x}}^T(i-1/i-1)\theta(i) \quad (18)$$

$$\boldsymbol{g}(i) = \frac{\boldsymbol{Q}(i-1)\hat{\boldsymbol{x}}(i-1/i-1)}{\lambda(i) + \psi'[\varepsilon(i)]\hat{\boldsymbol{x}}^T(i-1/i-1)\boldsymbol{Q}(i-1)\hat{\boldsymbol{x}}(i-1/i-1)} \quad (19)$$

$$\boldsymbol{Q}(i) = \frac{1}{\lambda(i)}\left[\boldsymbol{Q}(i-1) - \boldsymbol{g}(i)\hat{\boldsymbol{x}}^T(i-1/i-1)\boldsymbol{Q}(i-1)\psi'[\varepsilon(i)]\right] \quad (20)$$

$$\hat{\theta}(i) = \hat{\theta}(i-1) + \boldsymbol{Q}(i)\hat{\boldsymbol{x}}(i-1/i-1)\psi[\varepsilon(i) \quad (21)$$

The forgetting factor $\lambda(i)$ is a data weighting factor that is used to weight recent data more heavily and thus to permit tracking slowly varying signal parameters. If a nonstationary signal is composed of stationary subsignals the estimation of the AR parameters can be given by using a forgetting factor varying between $\lambda_{\min}$ and $\lambda_{\max}$. The modified generalized likelihood ratio algorithm [12] is used for the automatic detection of abrupt changes in stationarity of signal. This algorithm uses three models of the same structure and order, whose parameters are estimated on fixed length windows of signal. These windows are $[i-N+1, i]$, $[i+1, i+N]$ and $[i-N+1, i+N]$, and move one sample forward with each new sample. In the first step of this algorithm is calculated the discrimination function:

$$\begin{aligned} D(i,N) &= L(i-N+1, i+N) - L(i-N+1, i) \\ &\quad -L(i+1, i+N) \end{aligned} \quad (22)$$

where:

$$L(a,b) = (b-a+1)\ln\left[\frac{1}{b-a+1}\sum_{i=a}^{b}\varepsilon^2(i)\right] \quad (23)$$

denotes the maximum of the logarithmic likelihood function. In the second step a strategy for choosing the variable forgetting factor is defined by letting $\lambda(i) = \lambda_{\max}$ when $D = D_{\min}$ and $\lambda(i) = \lambda_{\min}$ when $D = D_{\max}$, as well as by taking the linear interpolation between these values.

B. *Steady-State Optimal Kalman Gain Estimation*

The Kalman filter always requires the knowledge of noise variances. When they are unknown, we must estimate them with some methods or we must estimate the steady-state optimal Kalman filter gain $\boldsymbol{K}^{opt} = \lim_{n\to\infty}\boldsymbol{K}(n)$ directly from the output data. Let $f(z) = z^m + \alpha_1 z^{m-1} + \cdots + \alpha_m$ be the minimal polynomial of the matrix $\boldsymbol{F}$ with $f(\boldsymbol{F}) = 0$.
From (11) and (12) in the steady-state:

$$y(n-m+i) = \boldsymbol{H}\boldsymbol{F}^{i}\hat{\boldsymbol{x}}(n-m/n-m-1) + \sum_{j=0}^{i-1}\boldsymbol{H}\boldsymbol{F}^{i-j}\boldsymbol{K}^{opt}e(n-m+i) + e(n-m+i) \quad (24)$$

and multiplying (24) by α_{m-i} ($\alpha_0 = 1$) and summing for $i = 0, 1, \cdots, m$ we obtain:

$$\sum_{i=0}^{m}\alpha_{m-i}y(n-m+i) = \sum_{i=0}^{m}\alpha_{m-i}\sum_{j=0}^{i-1}\boldsymbol{H}\boldsymbol{F}^{i-j}\boldsymbol{K}^{opt}e(n-m+j) + \sum_{i=0}^{m}\alpha_{m-i}e(n-m+i) \quad (25)$$

or:

$$y(n) + \sum_{i=1}^{m}\alpha_i y(n-i) = e(n) + \sum_{i=0}^{m}\beta_i e(n-i) \quad (26)$$

where:

$$\beta_i = \alpha_i + \sum_{j=0}^{i-1}\alpha_j \boldsymbol{H}\boldsymbol{F}^{i-j}\boldsymbol{K}^{opt}, \quad i = 1, \cdots, m \quad (27)$$

We can obtain the optimal gain $\boldsymbol{K}^{opt}$ by solving (26) with the knowledge of β_i for $i = 1, \cdots, m$. Define:

$$\xi(n) = e(n) + \sum_{i=1}^{m}\beta_i e(n-i) \quad (28)$$

It is known that in the optimal case the innovation process $e(n)$ is orthogonal to all past observations $y(1), \cdots, y(n-1)$ and it consists of a sequence of random variables that are orthogonal to each other. In this case the autocorrelation of the innovation process $r_e(k) = E[e(n)e(n-k)]$ is zero for $k > 0$ [13]. From (28) for $k = 0, 1, \cdots, m$ we obtain $r_\xi(k)$ the autocorrelation of $\xi(n)$, $r_\xi(k) = E[\xi(n)\xi(n-k)]$ as:

$$r_\xi(k) = r_e(0)\sum_{i=k}^{m}\beta_i\beta_{i-k}, \quad \beta_0 = 1 \quad (29)$$

The equations (28) can be solved for β_i, $i = 1, \cdots, m$ and $r_e(0)$ by using the estimate of the autocorrelation $\hat{r}_\xi(k)$:

$$r_\xi(k) = \frac{1}{N}\sum_{i=1}^{N}\xi(i)\xi(i-k) \quad (30)$$

where N is the sample size and is given by :

$$\xi(n) = y(n) + \sum_{i=1}^{m}\alpha_i y(n-i) \quad (31)$$

Now from (27) the estimate of the optimal gain $\hat{\boldsymbol{K}}^{opt}$ is given by:

$$\hat{\boldsymbol{K}}^{opt} = \begin{bmatrix} \boldsymbol{H}\boldsymbol{F} \\ \vdots \\ \sum_{j=0}^{m-1}\alpha_j \boldsymbol{H}\boldsymbol{F}^{m-j} \end{bmatrix}^{\dagger} \begin{bmatrix} \beta_1 - \alpha_1 \\ \vdots \\ \beta_m - \alpha_m \end{bmatrix} \quad (32)$$

IV. SIMULATION RESULTS

The proposed method was first tested using an AR signal that offers a good approximation of the spectral envelope of a speech signal and an additive Gaussian white noise. In the experiment, 256 samples of the AR signal were generated. In Table 1 we present the mean value, the standard deviation and the maximum value based on 1000 simulations.

Table 1

Input SNR (dB)	Output SNR (dB)		
	Mean	Std	Max
-5.00	2.93	0.48	4.46
0.00	5.72	0.29	7.33
5.00	9.82	0.21	11.27
10.00	12.71	0.15	13.72
15.00	17.08	0.07	17.31

The approach was also tested using a speech signal and additive noise. The speech signals are sentences from the TIMIT database. Table 2 offers a comparison with others approaches, by showing averaged SNR gain based on 10 speech signals and 10 noise simulations for each speech signal. Figures 2, 3 and 4 represent, respectively, the time signal followed by the spectrogram of the free-noise speech, the noisy speech and the enhanced speech. For this example, the SNR of the noisy speech signal is 0 dB.

Table 2

Input SNR (dB)	Output SNR (dB)			
	[14]	[7]	[8]	Prop.
-5.00	2.46	-2.52	2.48	2.56
0.00	4.57	2.61	4.72	4.88
5.00	7.96	6.83	8.29	8.37
10.00	11.92	10.95	12.31	12.48
15.00	16.00	15.08	16.47	16.76

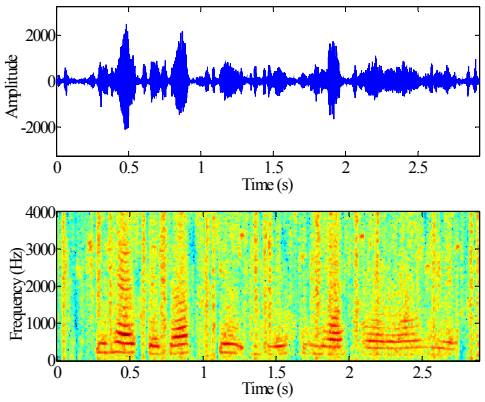


Fig. 1: Noise-free speech signal

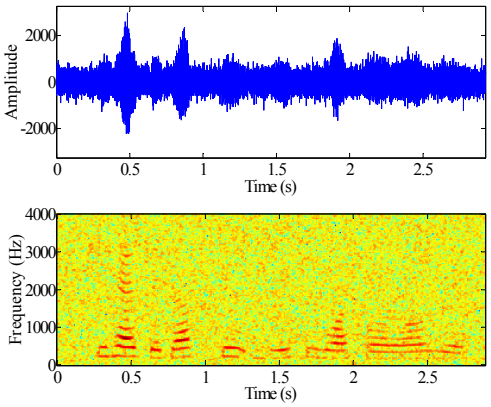


Fig. 2: Noisy speech signal

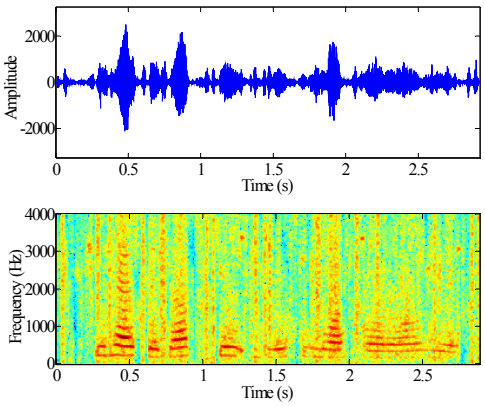


Fig. 3: Enhanced speech signal

Compared to the methods similar in structure previously proposed by the author in [7] and in [8] and to the Gibson's algorithm [14], the proposed method provides increases in SNR, as well as improved speech quality and intelligibility for input SNR between -5 and 15 dB. Gibson's algorithm needs two or three iterations to get the highest SNR gain. It uses a voice activity detector to determine silence periods. The above factors lead to computational requirements higher than those corresponding to the proposed approach.

REFERENCES

[1] K. K. Paliwal and A. Basu, "A Speech Enhancement Method Based on Kalman Filtering," in *Proc. ICASSP'87*, pp. 177-180, 1988.

[2] A. V. Oppenheim, E. Weinstein, K. C. Zangi, M. Feder, and D. Gauger, "Single-Sensor Active Noise Cancellation", *IEEE Trans. Speech and Audio Processing*, vol. 2, pp. 285-290, Apr. 1994.

[3] S. Gannot, D. Burshtein and E. Weinstein, "Iterative and Sequentiel Kalman Filter-Based Speech Enhancement Algorithms," *IEEE Trans. Speech and Audio Processing*, vol. 6, pp. 373-385, July 1998.

[4] K. Y. Lee and S. Jung, "Time-Domain Approach Using Multiple Kalman Filters and EM Algorithm to Speech Enhancement with Nonstationary Noise", *IEEE Trans. Speech and Audio Processing*, vol.8, pp. 282-291, May 2000.

[5] E. Grivel, M. Gabrea and M. Najim, "Speech Enhancement as a Realisation Issue", *Signal Processing*, vol. 82, pp. 1963-1978, Dec. 2002.

[6] M. Gabrea and D. O'Shaugnessy, "Speech Signal Recovery in White Noise Using an Adaptive Kalman Filter," in *Proc. EUSIPCO'00*, 2000.

[7] M. Gabrea, E. Grivel and M. Najim, "Single Microphone Kalman Filter-Based Noise Canceller", *IEEE Signal Processing Lett.*, vol. 6, pp. 55-57, Mar. 1999.

[8] M. Gabrea, "Adaptive Kalman Filtering-Based Speech Enhancement Algorithm", in *Proc. IWAENC'01*, pp. 207-210, 2001.

[9] B. A. Anderson and J. B. Moore, *Optimal Filtering*, NJ:Prentice-Hall, Englewood Cliffs, 1979.

[10] M. Najim, Modelization and Identification in Signal Processing, France:Masson, Paris, 1988.

[11] B. D. Kovacevic, M. M. Milosavljevic and M. Dj. Veinovic, Robust Recursive AR Speech Analysis", *Signal Processing*, vol. 44, pp. 125-138, 1995.

[12] M. Milosavljevic and I. Konvalinka, "The modified generalized likelihood ratio algorithm for automatic detection of abrupt changes in stationarity of signals", in *Proc. ISS'88*, 1988.

[13] T. Kailath, "An Innovations Approach to Least-squares Estimation, part I: Linear filtering in additive white noise", *IEEE Tran. Automatic Control*, vol. AC-13, pp. 646-655, Dec. 1968.

[14] J. D. Gibson, B. Koo and S. D. Gray, "Filtering of Colored Noise for Speech Enhancement and Coding," *IEEE Trans. Signal Processing*, pp. 1732-1742, Aug. 1991.

Buletinul Ştiinţific al Universităţii "Politehnica" din Timişoara

Seria ELECTRONICĂ şi TELECOMUNICAŢII
TRANSACTIONS on ELECTRONICS and COMMUNICATIONS

Tom 51(65), Fascicola 2, 2006

Software Tool for Passive Real-Time Measurement of QoS Parameters

Mihai Vlad [1], Ionut Sandu [1], Virgil Dobrota [2], Ionut Trestian [2], Jordi Domingo-Pascual [3]

Abstract – The paper presents the designing of a software tool for real-time measurement of the following quality of service parameters: one-way delay, average one-way delay, IP packet delay variation and average IP packet delay variation. The solution is an improved version of OreNETa (One-way delay REaltime NETwork Analyzer), by optimizing the traffic between the meter and the analyzer. When a new flow is detected, the meter assemblies a flow descriptor and sends it to the analyzer. Following the flow recording, it will announce the meter to send a shorter message, called header, for all the packets belonging to the newly registered flow.

Keywords: measurement tool, OreNETa, QoS parameters

I. INTRODUCTION

A major step toward the next generation networks is to implement the quality of service mechanisms for IP parameters. The work carried out in this paper is related to the FP6 European project *EUQoS,* focused on end-to-end quality of service support over heterogeneous networks. Its main objectives address: a) the standardization of end-to-end QoS issues in European and International bodies (especially the IETF); b) promoting the creation of new business models to enable the deployment of QoS applications by the Internet community and; c) foster the interoperability of end-to-end QoS solutions for the end user, across heterogeneous research, scientific and industrial network domains [1]. A flexible and secure QoS assurance system could be validated within EUQoS by using the herein proposed software tool for passive real-time measurement. Moreover, this application can be used in monitoring the SLA (Service Level Agreement) between partners and spot some errors during the testing phase. The initial functionalities of the Abel Navaro's *OreNETa (One-way delay REaltime NETwork Analyzer),* described in [2], were extended. The new proposed version passively captures the traffic already existing on a network and it measures a series of QoS parameters (one-way delay, IP packet delay variation) in real-time. It also logs all the captured data for offline processing. The passive measurements were chosen because they provide information about the existing current traffic within the network section investigated. Since no test traffic is generated, they can be applied for most applications where statements about the actual situation in the network are required (like SLA validation, traffic engineering). Active measurements can always be applied supplementary, in order to predict the future network situation during times where no regular traffic is transmitted. The reliability and quality of the link can be expressed in terms of number of packets lost too. Every time a packet belonging to a flow does not reach its destination, a counter is incremented to express the packet loss.

II. QUALITY OF SERVICE PARAMETERS

The QoS parameters that are intended to be measured herein using the proposed software tool are the following: one-way delay, average one-way delay, IP packet delay variation, average IP packet delay variation and packet loss. OWD (One-Way Delay) represents the time that takes a packet to travel through the network from source to destination, which means the time passing between the moment when the *first* bit of the packet leaves the source host and the moment when the *last* bit of the same packet reaches the destination host. This definition can be expressed mathematically by:

$$OWD_i = t_{1i} - t_{0i}, \text{ for } 1 \le i \le N. \quad (1)$$

where N is the total number of packets belonging to a flow. Fig. 1 illustrates the one-way delay for an n-byte packet traversing a network segment. The same

[1] Alcatel Romania, Strada Gheorghe Lazar 9, 300081 Timisoara, Romania, e-mail {Mihai.Vlad, Ionut.Sandu}@alcatel.ro
[2] Technical University of Cluj-Napoca, Communications Department, Strada George Baritiu 26-28, 400027 Cluj-Napoca, Romania, e-mail {Virgil.Dobrota, Ionut.Trestian}@com.utcluj.ro
[3] Universitat Politecnica de Catalunya, Jordi Girona 1-3, Barcelona, Spain, e-mail Jordi.Domingo@ac.upc.es

packet i sent at t_{0i} by the source is received at t_{1i} by the destination.

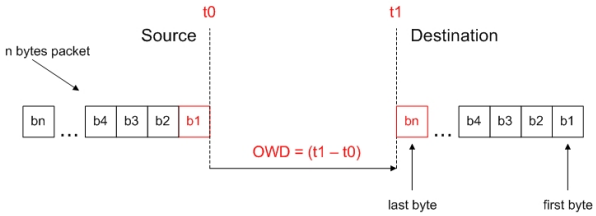


Fig. 1. One-Way Delay

AOWD (Average OWD) could be computed as follows:

$$AOWD = \frac{\sum_{i=1}^{N} OWD_i}{N} . \quad (2)$$

The term "jitter" refers to the variation of a parameter with respect to some reference parameter. A definition of IPDV (IP Packet Delay Variation), also referred to as *delay jitter,* can be given for packets inside a stream of packets. The IPDV is defined for a given pair of consecutive packets within the stream going from measurement point *MP1* to measurement point *MP2*. It is actually the difference between the one-way-delays of two consecutive packets.

$$IPDVi = OWD(i-1) - OWDi \text{, for } 1 < i \leq N . \quad (3)$$

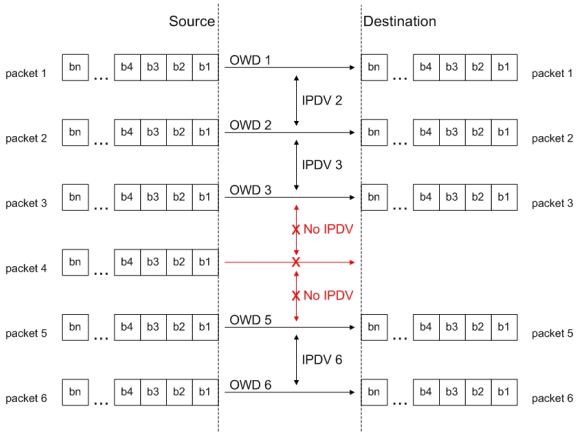


Fig. 2. IP Packet Delay Variation

In Fig. 2 the source is *MP1*, whilst *MP2* is the destination. If a packet is lost (e.g. packet 4), the IPDV (with respect to its adjacent packets) cannot be computed. Similar to one-way delay, AIPDV (Average IPDV) can be calculated as:

$$AIPDV = \frac{\sum_{i=2}^{N} IPDVi}{N} . \quad (4)$$

III. DESIGNING OF THE MEASUREMENT TOOL

The main building blocks for implementing a measurement tool are shown in Fig. 3. The processes involved are packet capturing, time-stamping, generation of flow ID, classification, generation of a packet ID and transfer of measurement data. Each of these processes adds a piece of information to the final message sent to the control application.

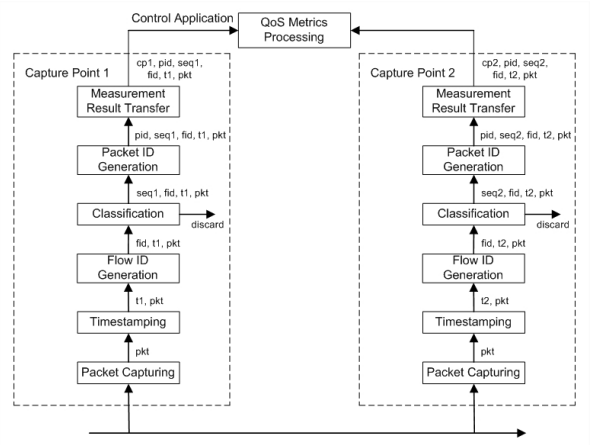


Fig. 3. Building Blocks

A. *Packet Capturing*

A certain amount of bytes needs to be captured per packet as basis for the generation of a packet ID. The packet ID collision probability (see subsection E) depends on the generation function and the number of bytes that are used as input. The first 40 Bytes starting at the IP header are considered to be sufficient for this purpose. However, the number of bytes that can be captured also depends on the processing power remaining for the measurement task. The packet capturing performance of a machine is limited by the following parameters: number of interrupts generated by the NIC; number of context switches; amount of bytes transferred to user space; and current load of the machine caused by other processes (e.g. packet ID generation) [3].

B. *Time-stamping*

A number of issues have to be considered for the basic function of assigning timestamps to packets for subsequent delay calculation. Internal buffering in the hardware and on the way through the kernel causes additional packet delay. Even if all involved measurement devices are equipped with the same hardware and operating system, packets can experience different delays (e.g. due to CPU load and the level of buffer filling). In order to reduce effects from additional variable delays, the timestamp should be assigned to the packet as early as possible. A further problem that has to be solved when using two measurement points is clock synchronization between both points. Best results are based on GPS (Global Positioning System).

C. *Flow ID Generation*

A flow is a sequence of packets sent from the same source to the same destination and receiving the same

level of service from the nodes. In order to obtain the same ID for every packet belonging to a given flow we must refer to a combination of fields found within Layer 3 and Layer 4 headers [4]. A similar process can be found in routers also, and is often referred to as flow identification. Since there is no specific information in the packet header to indicate if a packet belongs to a given flow or not, flow identification must be performed on every packet. However, modern routers must support wire-speed forwarding. To support that, a router must cope with the worst-case scenario rather than the average one. In the worst case, multiple packets that belong to reserved flows may arrive at the speed of the incoming link; that is, packets arrive back-to-back. A router must be able to deal with such a scenario.

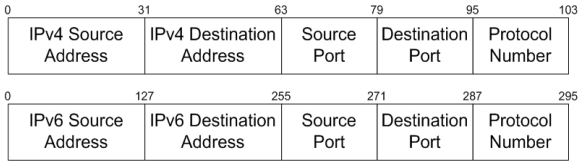


Fig. 4. IPv4 and IPv6 Five-Tuple

At high speeds the per-packet processing time is extremely small. For example, to support 64-byte packets at OC12 speed (622 Mbps), the per-packet processing time is less than 1 microsecond. Service providers are expected to upgrade their backbone to OC48 (2.5 Gbps) and OC192 (10 Gbps) soon. As the Internet expands, the number of concurrent flows can also be very large. An OC12 backbone trunk currently may have tens of thousands of concurrent flows. Thus the design of a flow identification module must be able to perform lookup at high speeds with a large number of flows [4].

D. *Classification*

After the flow ID has been generated for the captured packet we are able to take some decisions based on the flow identified by the new flow ID. There can be a discussion upon where to place the classification in the processing chain; before time-stamping or after it. Placing the classification before the time-stamping will introduce a variable delay equal to the classification processing time. Depending on the size of the classification table, this delay can be significant, and will lead to erroneous measurements. The benefit is that the timestamps will be generated only for relevant packets (i.e. with proper flow ID). The best option is to go for an early time-stamping to obtain accurate results.

E. *Packet ID Generation*

In order to get the same packet ID for one packet at two or more measurement points the packet ID generation should be based on the following fields that: a) already exist within the packet; b) are invariant or predictable during the transport; c) are highly variable between the different packets. The goal is to achieve an acceptable low probability of collisions with a packet ID that does not exceed the available capacity for the measurement result data transfer [3]. As for the timestamp, the packet ID only needs to be unique in the given time interval. This limits the possible combinations to the number of packets that can be observed within this interval. For example, on an 155 Mbps link with an average packet size of 512 bytes and a maximum time to traverse the network of 10 seconds, the maximum number of packets would be 378,421 (19,375,000/ 512*10 = 378,421). This amount of combinations can be represented by 19 bits (2^{19} = 524,288). Knowledge about the expected traffic mix therefore can reduce the number of required bits.

F. *Measurement Result Transfer*

In order to calculate the QoS parameters herein at least two timestamps have to be compared. If more than one measurement point is involved the results (timestamps and packet ID) from the different MPs have to be collected at a common location. This point can be located on a separate host or it can be co-located with one of the MPs. The transfer of the measurement results involved the following methods:

a) in-packet: timestamps and packet ID are carried within the packet, which should be modified.
b) in-band: the results are sent directly on the same path as the data.
c) out-of-band: a separate path is needed.

In all the cases additional capacity (either on the existing network or on a separate network) is required. For economic reasons even a separate reporting network would probably have a lower capacity then the “production network”. Therefore to save resources (storage capacity and bandwidth) the measurement results traffic should be kept as low as possible.

IV. PROPOSED ARCHITECTURE FOR PASSIVE REAL-TIME MEASUREMENTS

In order to fulfill the requirements especially that ones related to resource restrictions, real-time representation and data collection, a distributed architecture is proposed, with four components as in *OreNETa*, but with extended functionalities.

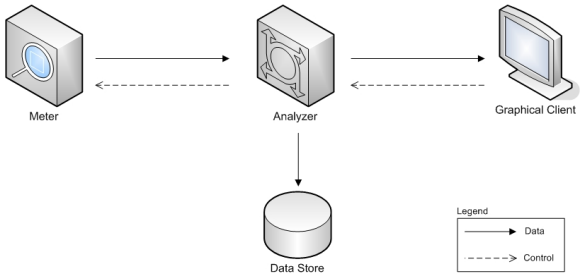


Fig. 5. Proposed Architecture

The *meter* is the component responsible for the first five processes (packet capturing, time-stamping, flow ID generation, classification and packet ID generation). This component can be placed either on the routers of the network being tested, either on dedicated machine for capturing traffic (behind a monitoring port of a switch or behind an optical splitter). For packet capturing and time-stamping, the *meter* uses a shared library *libpcap* (involved in applications like *tcpdump* and *ethereal*).

The *analyzer* is the control application responsible for collecting the measurements results from all the *meters* deployed in the tested network. It is the core component of the architecture and performs the synchronization of the received messages and the computation of QoS parameters. Other functionalities include logging the received measurement results in a data repository, converting the logged data to standardized formats (like MGEN) and supplying real-time statistics about the current traffic. The *analyzer* determines the functionality of the *meters* by sending control messages.

A *data store* is represented by a set of binary files containing all the measurement results received from each *meter*. The reason for not using a dedicated database engine was related to the poor performance of inserting records at very high rates. As an improvement one can develop a conversion tool for moving the data from the binary files to a database.

The *graphical client* is no more than a proof-of-concept application for highlighting the blueprints in extracting real-time statistics from the analyzer.

In Fig. 5 one can observe a flow of control messages along with the data flow between the four components. In order to address the requirements regarding the size of the messages and the limited resources of the control network, a special communication protocol needs to be designed. The bottleneck of the communication is represented by the segments joining the *meters* and the *analyzer*. All other segments presented in Fig. 5 are not relevant, since the *analyzer* and the *data store* are deployed on the same machine (and the operating system will be responsible for exchanging of information), and the messages between the *analyzer* and the *graphical client* are very rare (one every second). Before going any further with the explanation we need to define the term - *flow*, as the collection of packets having the same unique combination of Layer 3 Address, Layer 4 Protocol Type and Layer 4 Ports. The first version of the communication protocol between the *meter* and the *analyzer* was defined in [2]. The major drawback of this version is represented by the big amount of redundant data sent between the two components, especially for IPv6 measurements. For each captured packet on the *meter*, a packet report message was aggregated and sent to the *analyzer*. The fields "L3 Protocol", "IPv4 Src Address", "IPv4 Dst Address", "L4 Protocol", "L4 Src Port" and "L4 Dst Port" are the same for each packet belonging to a flow. Therefore, this data characterizes a flow, not a unique packet and the set of bytes was sent along with each packet report, increasing the redundancy: 14 bytes for IPv4, 38 bytes for IPv6. In order to optimize the traffic between the *meter* and the *analyzer*, a mechanism was developed to send two kinds of messages: *flow descriptors* and *headers*, as in Fig. 6.

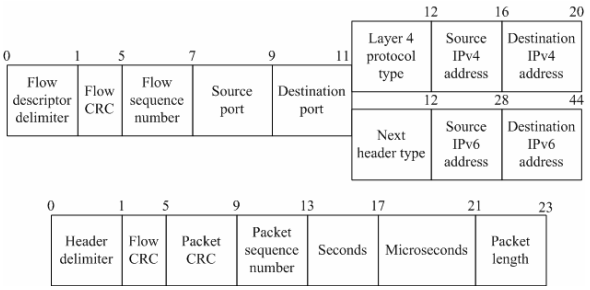


Fig. 6. Flow descriptor and header

When a new flow is detected (a packet is captured and it does not belong to an already detected flow), the *meter* assemblies a *flow descriptor* and sends it to the analyzer. Using this message, the *analyzer*, creates a structure which uniquely identifies the new flow. All packets captured later will be "routed" internally by the *analyzer* into one of these structures. This action is equivalent to the "Flow Identification" performed by real routers. After the flow has been recorded by the *analyzer*, it will announce the *meter* to send a shorter message, called *header*, for all the packets belonging to the newly registered flow. The information contained in the *header* is represented by data that is different from one packet to another and uniquely identifies it. One must not mistake this *header* for the IP or TCP headers. The name was chosen like this because the information sent in each *header* is extracted mainly from the real headers of the packets. By means of this mechanism, an important amount of traffic is reduced, mainly because the values identifying a flow (*five-tuple*) are only sent with the *flow descriptor*. This reduces the size of old packet reports from 28 bytes to 23 in the case of IPv4 and from 52 to 23 bytes for IPv6 (more than twice).

Another problem encountered within the first version of *OreNETa* was the fact that packet reports were sent for all captured packets, including those which were not part of any flow. For instance ICMP and ARP packets are not relevant to measuring OWD and IPDV, since a flow is defined using TCP or UDP. In the current version of the program, all the irrelevant packets are discarded at the meter, improving though the bandwidth used for the control network.

V. EXPERIMENTAL RESULTS

In order to test the new proprietary protocol, the testbed included two Debian Linux-based computers (*pampol* and *verdi*) running the *meter*, and one

Slackware Linux-based computer hosting the *analyzer*.

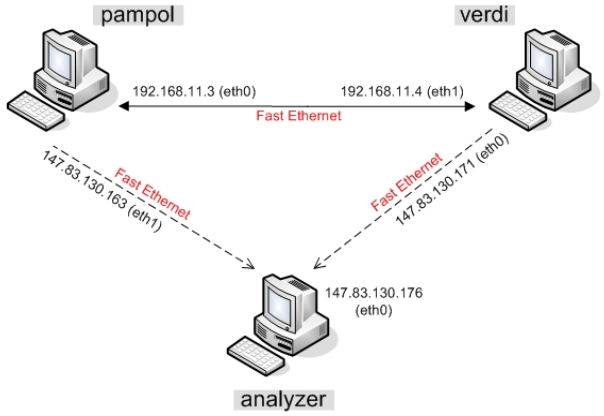


Fig. 7. Communication Protocol Stress Test

The connection between *pampol* and *verdi* represents the tested network, while the link connecting the *meters* with the *analyzer* would be the control network. These two networks are separated using VLANs configured on a Catalyst 2950 switch.

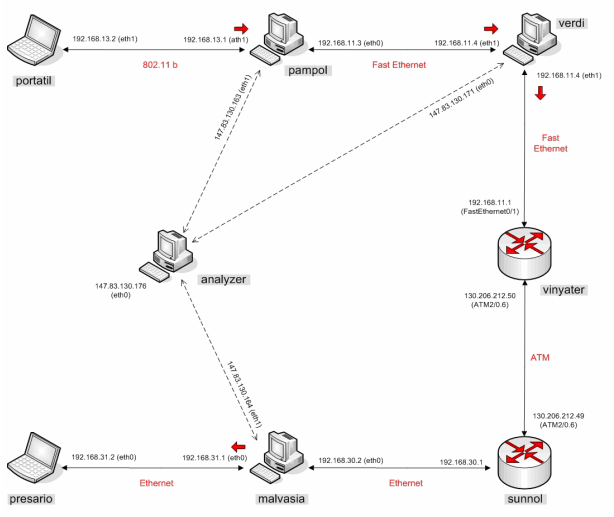


Fig. 8. Testbed

Experiment 1. The test consists in generating a large number of packets per second and monitoring any buffer overflows that might occur at the *meters* or at the *analyzer*. For this scenario, MGEN tool is installed on *pampol*, and different UDP flows are generated through the tested network using the following command:

```
root@pampol:~# mgen -i eth0 -b 192.168.11.4
2000000 -s <packet_size> -r <packet_rate>
```

The packet size did not exceed 1500 bytes, i.e. the Layer 2's MTU (Maximum Transmission Unit), to avoid the packet fragmentation. MGEN generated a set of up to 17,000 pps to *verdi*. The maximum packet rate is limited because of too many resynchronizations needed at the *analyzer* side, the packets being lost due to the limitations of *libpcap*, which is too slow. A new capturing solution may increase the performances of this software tool.

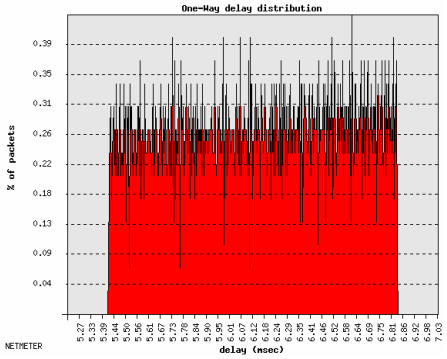


Fig. 9. OWD at 100 packets/s with 370 bytes/packet

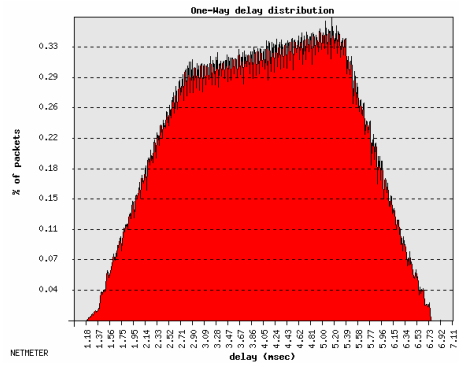


Fig. 10. OWD at 8,000 packets/s with 1,300 bytes/packet

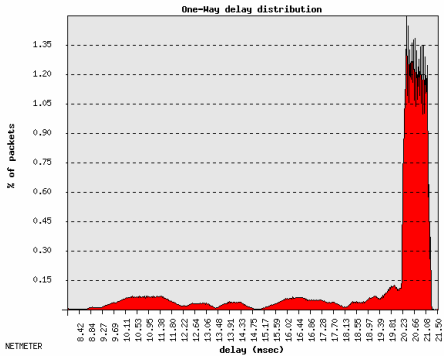


Fig. 11. OWD at 14,000 packets/s with 872 bytes/packet

Experiment 2. To prove that the software modules developed are able to measure the QoS parameters over a heterogeneous network, a more complex testbed, presented in Fig. 8, has been used. It included several Linux-based computers, as well as two Cisco 7600 routers. The links tested were Ethernet, Fast Ethernet, 802.11 WLAN and ATM. Table 1 presents the configuration used for all four capture points. The test traffic will be generated from *portatil* to *presario* using MGEN. The capture points and the direction of the traffic are marked with arrows.

Meter *verdi* is used twice in this configuration, but the two instances running on this computer will extract the traffic at different capture points and with different directions (eth1-IN vs. eth1-OUT). The same

interface is used on *verdi* both for incoming and outgoing traffic, so the *meters* should be able to distinguish the direction of the traffic.

Table 1 Experiment 2 Network Configuration

Meter	Control network	Tested network	Intf.	Link	I/O
Pampol	147.83.130.163	192.168.13.1	eth1	802.11	I
Verdi	147.83.130.171	192.168.11.4	eth1	FE	I
verdi	147.83.130.171	192.168.11.4	eth1	FE	O
malvasia	147.83.160.164	192.168.31.1	eth0	ETH	O

```
root@portatil:~# mgen -i eth1 -b 192.168.31.2
2000000 -s 512 -r 200
```

```
[   MTR    ] M[1]                M[2]                M[3]                M[4]
[   PPS    ] |      200 packets/s |      200 packets/s |      200 packets/s |
[THROUGHPUT] |   108000 bytes/s   |   108000 bytes/s   |   108000 bytes/s   |
[   OWD    ] | 0.002356 seconds   | 0.000180 seconds   | 0.009256 seconds   |
[   IPDV   ] |-0.000000 seconds   | 0.000000 seconds   |-0.000001 seconds   |
[  PKLOSS  ] |        0 packets/s |        0 packets/s |        0 packets/s |
```

Fig. 12. Experiment 2 results

Note that the number of packets captured on each segment is exactly as expected (200 packets/s). The packet loss is zero since the network is not under stress (108,000 bytes/s). The throughput recorded is bigger than the expected one (200 packets x 512 bytes/packet = 102,400 bytes). The difference in throughput was due to the fact that MGEN appends another 28 bytes to each sent packet. Therefore the size would grow to 512 + 28 = 540 bytes (in this case the computation is correct: 200 packets x 540 bytes/packet = 108,000 bytes). IPDV has negative values, which is correct since the variation of OWD might lead to negative results. Note that the resolution of 6 digits after the decimal point could be extended to improve the granularity of the results. The OWD obtained may vary depending on the Layer 2 link. The very small value of OWD (0.000180 seconds) between *meter 2* and *meter 3* is correct since the two meters run on the same computer. The biggest OWD is obtained for the segment which contains the extra to Cisco hops and the ATM line. Since the routers might be heavily used, by other traffic from the production network, the OWD will increase.

Experiment 3. Suppose the *meters* are perfectly synchronized in time. The *analyzer* is able to determine the route for the requested flows irrespective of the order of the meters supplied at command line. Let us change now the order of the meters, by placing M[4] not at the edge of the testbed, but on one of *verdi's* interfaces. The new configuration is presented in Table 2. We generate the same traffic as in the previous test using MGEN.

Table 2 Experiment 3 Network Configuration

Meter	Control network	Tested network	Intf.	Link	I/O
pampol	147.83.130.163	192.168.13.1	eth1	802.11	I
verdi	147.83.130.171	192.168.11.4	eth1	FE	O
malvasia	147.83.130.164	192.168.31.1	eth0	ETH	O
verdi	147.83.160.171	192.168.11.4	eth1	FE	I

```
[   MTR    ] M[1]                M[4]                M[2]                M[3]
[   PPS    ] |      200 packets/s |      200 packets/s |      200 packets/s |
[THROUGHPUT] |   108000 bytes/s   |   108000 bytes/s   |   108000 bytes/s   |
[   OWD    ] | 0.002580 seconds   | 0.000132 seconds   | 0.007500 seconds   |
[   IPDV   ] | 0.000000 seconds   | 0.000000 seconds   | 0.000000 seconds   |
[  PKLOSS  ] |        0 packets/s |        0 packets/s |        0 packets/s |
```

Fig. 13. Experiment 3 results

As expected, the order of the meters changed according to the path of the captured flow. M[4] follows M[1] since they are close to each other (*pampol-IN* and *verdi-IN*). The other 2 meters, M[2] placed on *verdi-OUT* and M[3] placed on *malvasia-OUT* are ordered accordingly to the route presented in Fig. 10. If the time synchronization between the *meters* is not accurate, this functionality will be useless, as well as the whole project. If the time difference between two *meters* is almost equal to the OWD value, we might expect to obtain negative values for the OWDs.

VI. CONCLUSIONS

Compared to the previous version of *OreNETa* the present work came up with a set of improvements, optimizations and additional functionalities. In order to optimize the traffic between the *meter* and the *analyzer*, a mechanism to send two kinds of messages (*flow descriptors* and *headers*) was developed. An important amount of traffic is reduced, mainly because the values identifying a flow (*five-tuple*) are only sent with the *flow descriptor*. This reduces the size of old packet reports from 28 to 23 bytes in the case of IPv4, and from 52 to 23 bytes for IPv6. All the irrelevant packets (e.g. ICMP and ARP) were discarded at the meter, improving though the bandwidth used for the control network. Another improvement is the possibility to connect more than one *analyzer* to a *meter* and vice-versa. The flow processing was implemented based on the fact that the packet IDs will always be captured in the same order by every *meter*, so the *headers* will be arranged chronologically when they arrive at the *analyzer*. A mechanism was implemented to order the *meters* on the tested network by comparing the timestamps of the first packets received. This simple mechanism is useful for tracing a route a flow is using, in case all the *meters* are perfectly synchronized in time. The tool uses pure binary files to store only the needed data, and the processing is performed later, when the capture is finished.

REFERENCES

[1] ***, EuQoS End-to-end Quality of Service Support over Heterogeneous Networks, http://www.euqos.org/

[2] A. Navarro, "OreNETa". Master Thesis, *Universitat Politecnica de Catalunya*, Barcelona, Spain, 2004

[3] T. Zseby, S. Zander, G. Carle, „Evaluation of Building Blocks for Passive One-Way Delay Measurements", *GMD FOKUS*, 2001

[4] Z. Wang, *Internet QoS: Architectures and Mechanisms for Quality of Service*. Morgan Kaufmann Publishers, 2001

[5] ***, MGEN-UDP Traffic Generator Tool, http://manimac.itd.nrl.navy.mil/MGEN/

[6] ***, IST-MOME Cluster of European Projects Aimed at Monitoring and Measurements, http://www.ist-mome.org/cluster/associated.html

Buletinul Ştiinţific al Universităţii "Politehnica" din Timişoara

Seria ELECTRONICĂ şi TELECOMUNICAŢII
TRANSACTIONS on ELECTRONICS and COMMUNICATIONS

Tom 51(65), Fascicola 2, 2006

Some Properties of Semantic Sources

Valeriu Munteanu[1], Daniela Tarniceriu[1]

Abstract – In this paper we derive the quantitative – qualitative entropy an extension of order *m* of a semantic source, as well as the semantic entropy of discrete ergodic sources with memory. The Kraft inequality and Shannon's first theorem are generalized for these sources. Some applications of semantic sources are also presented.

Keywords**: semantic sources, entropies, Kraft's inequality, Shannon's first theorem.**

I. INTRODUCTION

Unlike the quantitative characterization of information [1,2,3], in [4] we have introduced the concept of quantitative – qualitative information, and derived the entropy for semantic sources. The main properties of entropy for these sources are also established. Longo [5], Guardial and Pessoa [6], Khan and Autar [7], Autar and Khan [8], Khan and Bhat [9] have studied generalized coding theorems by considering different generalized measures of information. In this paper we determine the entropy for an extension of a semantic source, and then we consider semantic sources with memory to determine their entropy. The Kraft inequality [3] and the first Shannon theorem [2] are extended for semantic sources. In the last part of our work we present some applications of semantic entropy.

II. DETERMINING THE ENTROPY FOR AN EXTENSION OF A SEMANTIC SOURCE

For many source models it is useful to consider that the source delivers groups of messages, instead individual ones. Generally, from a discrete, complete and memoryless source S which delivers *n* messages, $s_1, s_2, ..., s_n$, with probabilities $p_1, p_2, ..., p_n$ and utilities $u_1, u_2, ..., u_n$, characterized by the distribution

$$S: \begin{pmatrix} s_1 & s_2 & \cdots & s_n \\ p_1 & p_2 & \cdots & p_n \\ u_1 & u_2 & \cdots & u_n \end{pmatrix} \quad (1)$$

We can derive another source, called the extension of the order *m* of the former, denoted by S^m, consisting in groups of *m* messages of the source *S,* in all possible combinations .

The extension of order *m* contains n^m composite symbols:

$$\sigma_k = s_{k1}s_{k2}s_{k3}...s_{km},\ k = 1,2,...,n^m \quad (2)$$

where $s_{k1}, s_{k2}, s_{k3}, ..., s_{km}$ are messages of the source *S*.

Assuming *S* memoryless, and its messages independent both probabilistic and logic – causal, we have

$$p(\sigma_k) = p(s_{k1}) \cdot p(s_{k2}) \cdot ... \cdot p(s_{km}) \quad (3)$$

and

$$u(\sigma_k) = u(s_{k1}) + u(s_{k2}) + ... + p(s_{km}) \quad (4)$$

Theorem 1

The quantitative – qualitative entropy of the extension of order *m* is *m* times the entropy of the semantic source, that is

$$H_{pu}(S^m) = mH_{pu}(S) \quad (5)$$

Proof

We prove this theorem by induction. First, we verify easily that (5) is true for $m = 1,2$. Next, we assume that (5) holds true for *m* and prove that

$$H_{pu}(S^{m+1}) = (m+1)H_{pu}(S) \quad (6)$$

Let ξ_i be a composite symbol made of $m+1$ messages of the source *S*, as

$$\xi_i = \sigma_k s_j,\ k = 1,2,...,n^m;\ j = 1,2,...,n \quad (7)$$

where s_j is a messages the source S delivers.

Since

$$p(\xi_i) = p(\sigma_k)p(s_j),\ u(\xi_i) = u(\sigma_k) + u(s_j) \quad (8)$$

$$\sum_{k=1}^{n^m} p(\sigma_k) = 1,\ \sum_{j=1}^{n} p(s_j) = 1, \quad (9)$$

the entropy of the extension of order (*m*+1) becomes

$$\begin{aligned} &H_{pu}(S^{m+1}) = \\ &-\sum_{i=1}^{n^{m+1}} p(\xi_i)\log_2 p(\xi_i) + \sum_{i=1}^{n^{m+1}} p(\xi_i)u(\xi_i) = \\ &-\sum_{k=1}^{n^m} p(\sigma_k)\log_2 p(\sigma_k) + \sum_{k=1}^{n^m} p(\sigma_k)u(\sigma_k) - \\ &-\sum_{j=1}^{n} p(s_j)\log_2 p(s_j) + \sum_{j=1}^{n} p(s_j)u(s_j) \end{aligned} \quad (10)$$

which is equivalent to (6).

[1] Facultatea de Electronică şi Telecomunicaţii, Departamentul Comunicaţii Bd. Carol I, Nr. 11, 700506 Iasi, e-mail vmuntean@etc.tuiasi.ro

III. DETERMINING THE ENTROPY FOR A SEMANTIC SOURCE WITH MEMORY

Let us consider a discrete, complete and ergodic source with memory of order *m*, with the distribution given in (1).

Theorem 2

The entropy of sources with memory of order *m* is

$$H_{pu}^{m}(S) = -\sum_{i=1}^{n^m}\sum_{j=1}^{n} p_i \cdot p(s_j \mid S_i)\cdot \log_2 p(s_j \mid S_i) + \\ +\sum_{i=1}^{n^m}\sum_{j=1}^{n} p_i \cdot p(s_j \mid S_i)\cdot u(s_j \mid S_i) \quad (11)$$

Proof

Let S_i be the state characterized by the sequence

$$S_i \rightarrow s_{i1}, s_{i2}, ..., s_{im}. \quad (12)$$

We denote

$$p(s_j \mid s_{i1}, s_{i2}, ..., s_{im}) \overset{not.}{=} p(s_j \mid S_i) \quad (13)$$

the probability that the source delivers the message s_j, given the state S_i, and

$$u(s_j \mid s_{i1}, s_{i2}, ..., s_{im}) \overset{not.}{=} u(s_j \mid S_i) \quad (14)$$

the utility the message s_j possesses, given the state S_i.

In [4], we proved that for a memoryless semantic source, the information attached to the message s_k having the probability p_k and the utility u_k, is given by

$$i_{pu}(s_k) = -\log_2 p_k + u_k. \quad (15)$$

Then, the quantitative – qualitative information obtained when the source is in the state S_i and it delivers the message s_j is then given by

$$i_{pu}(s_j \mid s_{i1}, s_{i2}, ..., s_{im}) = -\log_2 p(s_j \mid S_i) + u(s_j \mid S_i) \quad (16)$$

From the state S_i any message s_j can be delivered with a certain conditional probability (even equal to zero, if from that state a certain message cannot be delivered). The average quantitative – qualitative information the state S_i can deliver is

$$i_{pu}(S_i) = \sum_{j=1}^{n} p(s_j \mid S_i) i_{pu}(s_j \mid S_i) \quad (17)$$

or, considering (16)

$$i_{pu}(S_i) = -\sum_{j=1}^{n} p(s_j \mid S_i)\log_2 p(s_j \mid S_i) + \\ +\sum_{j=1}^{n} p(s_j \mid S_i) u(s_j \mid S_i) \quad (18)$$

Denoting by p_i, $i = 1, 2, ..., n^m$, the state probabilities of the ergodic, discrete, complete source with memory, the average quantitative – qualitative information, or the quantitative – qualitative entropy, denoted by $H_{pu}^{m}(S)$ can be computed by

$$H_{pu}^{m}(S) = \sum_{i=1}^{n^m} p_i i_{pu}(S_i) \quad (19)$$

Considering (18) and (19), we get (11).

IV. KRAFT'S THEOREM FOR SEMANTIC SOURCES

Theorem 3

The Kraft's theorem for semantic sources is

$$\sum_{k=1}^{n} M^{-l_k} \cdot 2^{u_k} \le 1, \quad (20)$$

where n is the number of messages the information source supplies, M – the number of symbols in the code alphabet and l_k - the length of the codeword c_k, $(\forall)k = 1, 2, ..., n$.

Proof

Let S be the semantic source characterized by the distribution given in (1).

Let

$$X = \{x_1, x_2, ..., x_M\} \quad (21)$$

be the alphabet of the code,

$$C = \{c_1, c_2, ..., c_n\} \quad (22)$$

the code words attached to the messages, and

$$L = \{l_1, l_2, ..., l_n\} \quad (23)$$

the length of the code words.

Due to the one – to one correspondence between the messages $s_k \in S$ and the code words $c_k \in C$, the information attached to the message s_k is equal to that attached to the code word c_k, i. e.

$$i_{pu}(s_k) = i_{pu}(c_k) \quad (24)$$

On the other hand, the maximum information per symbol of the code alphabet is $H_{\max}(X) = \log_2 M$, which can be reached when the symbols of the code alphabet are used independently and equally likely.

The length l_k corresponding to the code word c_k has to satisfy

$$l_k \ge \frac{i_{pu}(c_k)}{\log_2 M} = \frac{-\log_2 p_k + u_k}{\log_2 M} \quad (25)$$

From (24) and (25), we have

$$p_k \ge 2^{u_k} \cdot M^{-l_k}. \quad (26)$$

As

$$\sum_{k=1}^{n} p_k = 1 \quad (27)$$

eq. (20) results.

V. SHANNON'S FIRST THEOREM FOR SEMANTIC SOURCES

Let there be the source characterized by (1) and the code characterized by (21), (22) and (23).

Obviously, the probabilities and utilities of the source messages are equal to the probabilities $p(c_k)$ and utilities $u(c_k)$ of the code words, respectively, i. e.

$$p_k = p(c_k) \quad (28)$$

$$u_k = u(c_k) \tag{29}$$

The length of each code word must belong to the set of positive integers, therefore l_k must be chosen as an integer satisfying the condition

$$\frac{-\log_2 p_k + u_k}{\log_2 M} \le l_k < \frac{-\log_2 p_k + u_k}{\log_2 M} + 1 \tag{30}$$

where $\log_2 M$ is the maximum value $H(X)$ can take on.

By multiplying (30) by p_k and summing up from 1 to n, we have

$$\frac{-\sum_{k=1}^{n} p_k \log_2 p_k + \sum_{k=1}^{n} p_k u_k}{\log_2 M} \le \sum_{k=1}^{n} p_k l_k < \frac{-\sum_{k=1}^{n} p_k \log_2 p_k + \sum_{k=1}^{n} p_k u_k}{\log_2 M} + 1 \tag{31}$$

or,

$$\frac{H_{pu}(S)}{\log_2 M} \le \bar{l} < \frac{H_{pu}(S)}{\log_2 M} + 1 \tag{32}$$

where

$$H_{pu}(S) = -\sum_{k=1}^{n} p_k \log_2 p_k + \sum_{k=1}^{n} p_k u_k \tag{33}$$

is the entropy of the semantic source [4] and

$$\bar{l} = \sum_{k=1}^{n} p_k l_k \tag{34}$$

is the average length of the code words. Relation (32) holds true also for the extension of order m, for which we can write

$$\frac{H_{pu}(S^m)}{\log_2 M} \le \bar{l}_m < \frac{H_{pu}(S^m)}{\log_2 M} + 1 \tag{35}$$

where $\bar{l}_m$ is the average length of the code words corresponding to a sequence of m messages of the source S and $H_{pu}(S^m)$ is its entopy.

Since

$$\bar{l} = \frac{\bar{l}_m}{m}, \tag{36}$$

we have

$$\frac{H_{pu}(S)}{\log_2 M} \le \bar{l} < \frac{H_{pu}(S)}{\log_2 M} + \frac{1}{m} \tag{37}$$

From (37) it follows that, when $m \to \infty$, the average length of the code words becomes equal to the minimum average length. Thus, (37) becomes a generalization of C. E. Shannon's Theorem [2], for encoding discrete sources for noiseless channels, in the case of cybernetics systems. If the utilities of the messages of source S are zero, the classical results are obtained [1].

VI. APPLICATIONS OF QUALITATIVE – QUANTITATIVE ENTROPY

A first application of quantitative – qualitative entropy regards the calculus of the entropy for a binary block code.

Let us consider a binary block code, for which each codeword contains N binary symbols. If k denotes the number of information symbols in each codeword, the number of code words, n, is determined by

$$n = 2^k \tag{38}$$

Due to the one - to – one correspondence between the code words and the messages of the information source, each codeword will have the same probability and utility as the corresponding message.

Let us also suppose that in order to correct the errors in each codeword, m parity – check symbols are used.

Considering the utility of each transmitted codeword equal to the number of parity check symbols m, the following distribution results:

$$C: \begin{pmatrix} c_1 & c_2 & \cdots & c_n \\ m & m & \cdots & m \end{pmatrix} \tag{39}$$

The absolute maximum entropy $H_{ma}(C)$ of this source is attained when the messages and the code words are equally likely delivered.

From [4] we have

$$H_{ma}(C) = \log_2 n + \frac{U}{n} \tag{40}$$

Considering (38) and the fact that the whole utility of the code words is

$$U = n \cdot m, \tag{41}$$

we get

$$H_{ma}(C) = k + m = N \tag{42}$$

According to (42) the absolute maximum entropy of a binary block error correcting code is equal to the codeword length.

A second application consists in the establishing of the delivering probabilities of unequal protected code words, so that the average information per codeword is maximum one. With this purpose in view, let us consider a binary block code of length N, in which the first codeword has m_1 parity check symbols, the second one, m_2, and so on, the last one having m_n parity check symbols.

Further on, we consider the general case, in which code words with the same number of parity check symbols could exist. Obviously, the more parity check symbols the code words contain, the more errors can be corrected.

We also consider that the parity check symbols in each codeword represent the utilities.

In order to obtain the maximum average information per codeword, the source characterized by the distribution

$$S: \begin{pmatrix} s_1 & s_2 & \cdots & s_n \\ m_1 & m_2 & \cdots & m_n \end{pmatrix} \tag{43}$$

provides its messages with the probabilities computed by [4]

$$p_k = \frac{2^{m_k}}{\sum_{j=1}^{n} 2^{m_j}}, k = 1,2,...,N \quad (44)$$

The average information per codeword may be computed by means of [4]

$$H_m(S) = \log_2\left(\sum_{k=1}^{n} 2^{m_k}\right) \quad (45)$$

The average information per symbol in a codeword results as follows:

$$i = \frac{H_m(S)}{N} = \frac{\log_2\left(\sum_{j=1}^{n} 2^{m_j}\right)}{N} \quad (46)$$

where n is given by (38).

In the particular case, when all code words are identically protected

$$m_1 = m_2 = ... = m_N = m\,, \quad (47)$$

an information $i = 1$ bit/symbol is obtained.

The third application consists in the implementation of an encoding method for noiseless channels of sources characterized by the distribution

$$S:\begin{pmatrix} s_1 & s_2 & \cdots & s_n \\ u_1 & u_2 & \cdots & u_n \end{pmatrix}, u_k \in R, k = 1,2,...,n\,, \quad (48)$$

so that the average information per codeword is maximum and the average codeword length is minimum.

From (48) we observe that each message utility is known and we want to find the message delivering probabilities, so that the average information per message is maximum one. To this aim the probabilities p_k with which the messages s_k have to be delivered are computed by means of (44). Therefore, the source entropy becomes maximum one. In order to obtain the code words of the smallest length, for a noiseless channel, one can use the Huffman encoding procedure [12], using the above obtained probabilities.

VII. CONCLUSIONS

In this paper, we derive the quantitative – qualitative entropies of an extension of order *m* of the source (eq. 5) and of a discrete ergodic source with memory (eq. 11). These relations represent generalizations of the classical concepts on information. The quantitative – qualitative information results as the sum between a quantitative information and a qualitative one. If the qualitative characteristic is neglected, the classical known relations [1] are obtained. When only the qualitative characteristic is required, the first term in the relations above is dropped out.

By extending the notion of entropy to cybernetic systems, the Kraft inequality and Shannon's first theorem have been generalized. Three applications of sources with preferences are also presented. The first one regards the error correcting block codes, the second one, the unequally error protection block codes and the third one, the possibility to encode sources characterized only qualitatively, so that the average information per codeword is maximum and the average length of code words is minimum.

REFERENCES

[1] G. R.Gallager, *Information theory and reliable communications*, New York, John Wiley and Sons Inc., 1968.

[2] C. E.Shannon, "A mathematical theory of communication", BSTJ, 27 1948, pp. 379-423, 623-656.

[3] T.Cover, J. A.Thomas, *Elements of Information Theory*, Wiley, 1991.

[4] V.Munteanu, D. Tarniceriu, "On Semantic Feature of Information", Symposium Etc. '06", *Buletinul Universității "Politehnica", Seria Electrotehnica, Electronica si Telecomunicatii*, Tom 51 (65), 2006, Fascicola 1-2, 2006.

[5] G. Longo, "A noisless coding theorem for sources having utilities", *SIAM J. Appl. Math*., 30 (4), 1976, pp. 739-748.

[6] A.Gurdial, F. Pessoa, On useful information of order α. J Comb. Information and Syst. Sci., 2, 1977, pp. 158-162.

[7] A. B. Khan, R. Autar, "On useful information of order α and β", *Soochow J. Math.*, 5, 1979, pp. 93-99.

[8] R. Autar, A. B. Khan, "On generalized useful information for incomplete distribution", *J. of Comb. Information and Syst. Sci*., 14 (4) (1989), 187-191.

[9] A. B. Khan, B. A. Bhat, S. Pirzada, "Some results on a generalized useful information measure". *J. Inequal. Pure and Appl. Math*, 6 (4), 2005, pp. 1-5.

[10] M. Belis, S. Guiasu, "A quantitative-qualitative measure of information in cybernetics", *IEEE Trans. Inf. Theory* IT – 14, 1968, pp. 593-594.

[11] V. Munteanu, P. Cotae, "The entropy with preference", *Int. J. Electronics and Comm. AEU*, 46, 1992, pp. 429 – 431.

[12] D. A. Huffman, "A Method for the construction of minimum redundancy codes", *Ed. Jackson, Communication Theory. Butterwoths Scientific Publications London*. 1953 pp. 102-110.

Buletinul Ştiinţific al Universităţii "Politehnica" din Timişoara

Seria ELECTRONICĂ şi TELECOMUNICAŢII
TRANSACTIONS on ELECTRONICS and COMMUNICATIONS

Tom 51(65), Fascicola 2, 2006

Spectral analysis for detecting protein coding regions based on a new numerical representation of DNA

Şerban Mereuţă [1]

Abstract – The major signal in coding regions of genomic sequences has a three-base periodicity. By proposing a new numerical representation for the DNA chain, our aim is to use spectral analysis for recognizing the coding regions of a gene. Since the peak at *f*=1/3 in the Fourier spectrum is a good discriminator of the coding potential of an intronless DNA strand, we utilized this feature within a sliding window in order to detect probable exons in a DNA sequence. Our technique is independent of training sets or existing database information, and thus can find general application.

Keywords: genomic signal processing, spectral analysis, exon detection.

I. INTRODUCTION

A single strand of DNA is a biomolecule consisting of many linked, smaller components called nucleotides. Each nucleotide (base) is one of four possible types designated by the letters *A*, *G*, *C* and *T* and has two distinct ends, the 5' end and the 3' end, so that the 5' end of a nucleotide is linked to the 3' end of another nucleotide by a strong chemical bond, thus forming a long, one-dimensional chain (backbone) of a specific directionality. Therefore, each DNA single strand is mathematically represented by a character string which, by convention, specifies the 5' to 3' direction when read from left to right. The double helix DNA is formed together with a complementary strand by linking *A* with *T* and vice versa, and *C* with *G* and vice versa.

The worldwide genome sequencing triggered the necessity of developing new approaches to rapidly assess the potential of a given DNA sequence. In this context, the gene identification problem through computational means is of great interest [1]-[5], [10]. But accurate gene prediction becomes complicated because of the fact that, in advanced organisms, protein coding regions in DNA are typically separated into several isolated subregions called *exons*. The regions between two successive exons are called *introns*, and they are eliminated before protein coding through a process called *splicing*.

In this paper, we investigate a spectral analysis technique based on a distinctive feature of protein coding regions of DNA sequences, i.e., the existence of short-range correlations in the nucleotide arrangement. The most prominent of these is a 3-base periodicity, which has been shown to be present in coding sequences [1], [3], [6], [10]. The signature of this periodicity (and any other) can be seen most directly, through the Fourier analysis, as a spectral peak [7].

II. DNA AS BINARY CODE INDICATOR SEQUENCES

In order to apply the techniques specific to digital signal processing, the DNA symbolic form given in the public genomic databases [8] must be represented by numerical sequences. This symbolic – numeric mapping must be done in such a manner that it doesn't distort the properties of the original DNA sequence, nor it introduces noise-like artifacts [3], [6], [7], [9].

In this paper, we propose a new numerical representation of the nucleotidic chain to be analyzed. This representation preserves the properties of the original genomic sequence and opens the possibility of an information theoretic approach, based on the source coding nature of the resulting binary string.

First, we start by collecting the statistics of the DNA sequence and compute the occurrence probabilities of the nucleotides *A*, *G*, *C* and *T*. Arranging the symbols in ascending order of probability, we assign binary code (00, 01, 10, 11) to the four bases.

For example, given the DNA sequence:

$$5' - C\text{-}C\text{-}G\text{-}A\text{-}C\text{-}A\text{-}T\text{-}T\text{-}C\text{-}A - 3',$$

the occurrence probabilities are $p(A) = 0.3$, $p(G) = 0.1$, $p(C) = 0.4$ and $p(T) = 0.2$. Hence, the corresponding binary code is $G \to 00$, $T \to 01$, $A \to 10$ and $C \to 11$.

In general, considering a sequence of N nucleotides, the numerical sequence attached can be written as:

$$x[n] = b_A[n] + b_G[n] + b_C[n] + b_T[n], \quad n = 0,1,2,...,N-1 \tag{1}$$

[1] Facultatea de Electronică şi Telecomunicaţii, Catedra de Telecomunicaţii, Bd. Carol I nr. 11, 700506 Iaşi, Romania, e-mail: smereuta@zeta.etc.tuiasi.ro

where $b_A[n]$, $b_G[n]$, $b_C[n]$ and $b_T[n]$ are the *binary code indicator sequences*, which either have or haven't the code assigned to a specific nucleotide, depending on whether the corresponding character exists or not, respectively, at location *n*.

For example, in Table 1 we show the four binary code indicator sequences of a part of the previous DNA stretch. For computational reasons, in the implementation of our algorithm, we considered a level representation of the binary code.

Table 1

	C	*G*	*A*	*C*	*A*	*T*
$b_A[n]$	0 0	0 0	1 -1	0 0	1 -1	0 0
$b_G[n]$	0 0	-1-1	0 0	0 0	0 0	0 0
$b_C[n]$	1 1	0 0	0 0	1 1	0 0	0 0
$b_T[n]$	0 0	0 0	0 0	0 0	0 0	-1 1

In this manner, any DNA character string becomes a numerical sequence, having at each location *n* the code assigned to that particular nucleotide with respect to the corresponding occurrence probability.

III. ALGORITHM

There have been numerous proposed "protein coding measures" used for gene identification [1], [2], [3], [10]. In this paper, we predict whether or not a given DNA segment is a coding one using a similar methodology [3], from the magnitude of a properly defined spectral measure. We start by presenting the main tools of our algorithm.

A. *Discrete Fourier Transform*

The Discrete Fourier Transform (DFT) of a sequence $x[n]$, of length *N*, is itself another sequence $X[k]$, of the same length *N*:

$$X[k] = \sum_{n=0}^{N-1} x[n]\, e^{-j2\pi\frac{k}{N}n}, \quad k = 0,1,2,...,N-1 \tag{2}$$

The sequence $X[k]$ provides a measure of the frequency content at "frequency" *k*, which corresponds to an underlying "period" of $\frac{N}{k}$ samples, where the maximum frequency (period 2) corresponds to $k = \frac{N}{2}$, assuming that *N* is even.

Using the definition in (2), the resulting sequences $B_A[k]$, $B_G[k]$, $B_C[k]$ and $B_T[k]$ are the DFTs of the binary code indicator sequences $b_A[n]$, $b_G[n]$, $b_C[n]$ and $b_T[n]$, respectively.

From (1) and (2) it follows that:

$$X[k] = B_A[k] + B_G[k] + B_C[k] + B_T[k], \quad k = 0,1,2,...,N-1. \tag{3}$$

Supposedly that instead of the binary code indicator sequences introduced in section II, we consider only the indicator sequences [7] – $u_A[n]$, $u_G[n]$, $u_C[n]$ and $u_T[n]$. These sequences take on the value of either 1 or 0 at location *n*, depending on whether the corresponding nucleotide exists or not at that location. Then, in the case of pure DNA character strings (i.e., without assigning numerical values), the resulting DFTs, $U_A[k]$, $U_G[k]$, $U_C[k]$ and $U_T[k]$, provide a four-dimensional representation of the "frequency spectrum" of the character string. The quantity

$$S[k] = |U_A[k]|^2 + |U_G[k]|^2 + |U_C[k]|^2 + |U_T[k]|^2 \tag{4}$$

has been used as a measure of the total power spectral content of the DNA character string, at "frequency" *k* [6], [10]. The DFT frequency $k = \frac{N}{3}$ corresponds to a period of three samples. It is known [1], [7], [10] that the spectrum of protein coding DNA typically has a peak at that frequency. For example, in Fig. 1 we have plotted the sequence $S[k]$, as defined in (4), for a coding region of length $N = 1320$ inside the genome of the baker's yeast (formally known as *Saccharomyces Cerevisiae*), demonstrating a peak at frequency $k = 440$ $(= N/3)$.

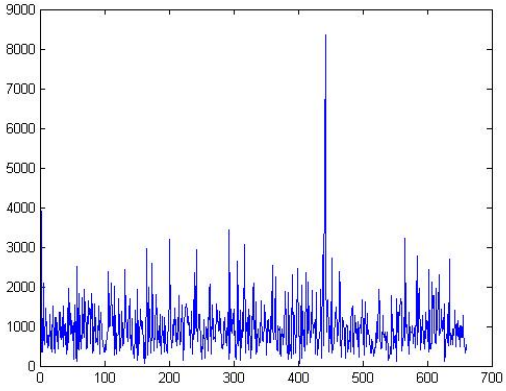


Fig. 1. Plot of the spectrum of a coding DNA region of length *N*, demonstrating peak at $k = N/3$.

B. *Short Time Fourier Transform*

Instead of evaluating the DFT of a full-length sequence, we have the option of evaluating the DFTs of several of its subsequences. This strategy makes sense particularly in the case of long sequences consisting of several segments with different characteristics.

For example, we may apply a "sliding window" of length *L* to a sequence of length *N*, where $N > L$,

resulting in a sequence of DFTs. Each of these DFTs provides a localized measure of the frequency content, and is an example of a location-dependent Fourier transform, known as the *short-time Fourier transform* (STFT).

Implementation

Taking into consideration [3] and the previously presented tools for spectral analysis, if we define the following normalized DFT coefficients at frequency $k = \frac{N}{3}$:

$$W_{\frac{N}{3}} = \frac{1}{N} X\left[\frac{N}{3}\right]$$
$$A_{\frac{N}{3}} = \frac{1}{N} B_A\left[\frac{N}{3}\right], \quad G_{\frac{N}{3}} = \frac{1}{N} B_G\left[\frac{N}{3}\right], \qquad (5)$$
$$C_{\frac{N}{3}} = \frac{1}{N} B_C\left[\frac{N}{3}\right], \quad T_{\frac{N}{3}} = \frac{1}{N} B_T\left[\frac{N}{3}\right],$$

then it follows from (3), with $k = \frac{N}{3}$, that:

$$W_{\frac{N}{3}} = A_{\frac{N}{3}} + G_{\frac{N}{3}} + C_{\frac{N}{3}} + T_{\frac{N}{3}}. \qquad (6)$$

In other words, for each DNA segment of length N (where N is a multiple of three) it corresponds a complex number $W_{\frac{N}{3}}$.

We evaluated the magnitude of the 351-point STFT ($L = 351$) for a DNA stretch of *Caenorhabditis Elegans* (obtained by searching database [8] under 'Nucleotide', with the accession number AF099922), which contains 8040 nucleotides starting from location 7021. Inside this segment, the gene F56F11.4 is present, having five exons, with the positions relative to 7021 as in Table 2.

Table 2

Exon #	Relative position	Exon length
I	929 – 1135	207
II	2528 – 2857	330
III	4114 – 4377	264
IV	5465 – 5644	180
V	7255 – 7605	351

By collecting the statistics of the DNA sequence, it results the following occurrence probabilities of the nucleotides and their corresponding binary codewords: $p(C) = 0.157 \to 00$, $p(G) = 0.162 \to 01$, $p(T) = 0.337 \to 10$, $p(A) = 0.344 \to 11$.

In Fig. 2 it is shown the square magnitude $\left|W_{\frac{N}{3}}\right|^2 = \left|A_{\frac{N}{3}} + G_{\frac{N}{3}} + C_{\frac{N}{3}} + T_{\frac{N}{3}}\right|^2$, and all the five exons of the gene F56F11.4 are identified by the peaks of the plot at the positions shown in Table 2.

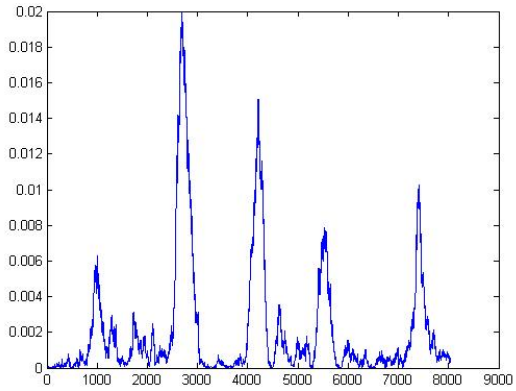


Fig. 2. Plot of $\left|W_{\frac{N}{3}}\right|^2$ for the five exons shown in Table 2

For comparison purposes, in Fig. 3 we represent the plot of $S[k]$ as defined in (4). This plot also proves that the performance of the spectral content measure $\left|W_{\frac{N}{3}}\right|^2$ is significantly superior to that of the one proposed by Tiwari et al. [10]. A demonstration of this fact was presented in [3], but in that case, for the numerical representation of DNA, Anastassiou used optimized values based on a training set of nucleotidic strings.

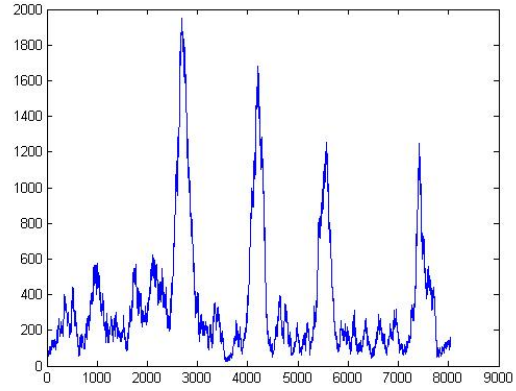


Fig. 3. Plot of $S[k]$ as defined in (4) and proposed in [10]

As it can be seen, the representation of $S[k]$ fails in detecting the first exon.

IV. CONCLUSIONS

In this paper, we presented a spectral analysis method for detecting DNA coding regions. We introduced a new numerical representation for the DNA stretch, by assigning binary codewords according to the occurrence probabilities of the nucleotides.

The advantage of our method is that it is independent of training sets or existing database

information, because the nucleotidic – numeric conversion is adapted to the statistics of the investigated DNA sequence.

Our experimental results prove that the algorithm we implemented offers a very good discrimination between coding and noncoding regions in DNA stretches. The capability to distinguish exons from introns is better than what has already been proposed. We also intend to extend our analysis on a bigger set of eukaryotic DNAs, as the examples used in this paper, and also to apply our algorithm in the case of prokaryotic organisms.

Because the numerical characterization we introduced for DNA strings results from the symbolic chain itself, the total computing time is smaller and can find general application.

REFERENCES

[1] J. W. Fickett, "The gene identification problem: an overview for developers", *Computers & Chemistry*, vol. 20(1), p. 103-118, 1996.

[2] J.-M. Claverie, "Computational methods for the identification of genes in vertebrate genomic sequences", *Hum. Mol. Genet.*, vol. 6, p. 1735-1744, 1997.

[3] D. Anastassiou, "Frequency-domain analysis of biomolecular sequences", *Bioinformatics*, vol. 16 (12), p. 1073-1082, 2000.

[4] S. Seneff, C. Wang, C. B. Burge, "Gene structure prediction using an orthologous gene of known exon-intron structure", *Appl. Bioinformatics*, vol. 3, p. 81-90, 2004.

[5] B. Brejova, D. G. Brown, M. Li, T. Vinar, "ExonHunter: a comprehensive approach to gene finding", *Bioinformatics*, vol.21(1), p. 57-65, 2005.

[6] B. D. Silverman, R. Linsker, "A measure of DNA periodicity", *Journal of Theoretical Biology*, vol. 118, p. 295-300, 1986.

[7] R. Voss, "Evolution of long-range fractal correlations and 1/f noise in DNA base sequences", *Physical Review Letters*, vol. 68(25), p. 3805-3808, 1992.

[8] GenBank [Online]: http://www.ncbi.nih.gov/GenBank

[9] P. D. Cristea, "Conversion of nucleotides sequences into genomic signals", *J. Cell. Mol. Med.*, vol. 6 (2), p. 279-303, 2002.

[10] S. Tiwari, S. Ramachandran, A. Bhattacharya, S. Bhattacharya, R. Ramaswamy, "Prediction of probable genes by Fourier analysis of genomic sequences", *CABIOS*, vol. 13 (3), p. 263-270, 1997.

Buletinul Ştiinţific al Universităţii "Politehnica" din Timişoara

Seria ELECTRONICĂ şi TELECOMUNICAŢII
TRANSACTIONS on ELECTRONICS and COMMUNICATIONS

Tom 51(65), Fascicola 2, 2006

Speech and Speaker Recognition Application on the TMS320C541 board

Eugen Lupu[1], Petre G. Pop[1], Radu Arsinte[1]

Abstract – The paper presents a speech and speaker recognition application developed on the EVM C541 board using the CCS®. The application represents the implementation of the TESPAR coding method on a DSP support. The TESPAR alphabet for the coding process was obtained formerly. The speech/speaker information contained in the utterances is extracted by TESPAR coder and provides the TESPAR A matrices. For the recognition decision, the distances among the TESPAR A test matrix and the TESPAR A reference matrices are computed. The results of the experiments prove the high capabilities of the TESPAR method in the classification tasks.

I. INTRODUCTION

TESPAR *(Time Encoded Signal Processing and Recognition)* coding is a method based on the approximations to the locations of the 2TW (where W is the signal bandwidth and T the signal length) real and complex zeros, derived from an analysis of a band-limited signal under examination. Numerical descriptors of the signal waveform may be obtained via the classical 2TW samples ("Shannon numbers") derived from the analysis. The key features of the TESPAR coding in the speech-processing field are the following:

-the capability to separate and classify many signals that cannot be separated in the frequency domain

-an ability to code the time varying speech waveforms into optimum configurations for processing with Neural Networks

-the ability to deploy economically, parallel architectures for productive data fusion [2].

The key in the interpretation of the TESPAR coding possibilities consists in the complex zeros concept. The band-limited signals generated by natural information sources include complex zeros that are not physically detectable. The real zeros of a function (representing the zero crossing of the function) and some complex zeros can be detected by visual inspection, but the detection of all zeros (real and complex) is not a trivial problem. To locate all complex zeros involves the numerical factorization of a $2TW^{th}$-order polynomial. A signal waveform of bandwidth W and duration T, contains 2TW zeros; usually 2TW exceeds several thousand. The numerical factorization of a $2TW^{th}$-order polynomial is computationally infeasible for real time. This fact had represented a serious impediment in the exploitation of this model. The key to exceed this deterrent and use the formal zeros-based mathematical analysis is to introduce an approximation in the complex zeros location [5].

Instead of detecting all zeros of the function the following procedure may be used:

- The waveform is segmented between successive real zeros and
- This duration information is combined with simple approximations of the wave shape between these two locations.

These approximations detect only the complex zeros that can be identified directly from the waveform.

In this transformation of signals, from time-domain in the zero-domain:

- The real zeros, in the time-domain, are identical to the locations of the real zeros in the zero-domain, and
- The complex zeros occur in conjugate pairs and these are associated with features (minima, maxima, points of inflexion etc.) that appear in the wave shape between the real zeros [3][4].

In this way examining the features of the wave, shape between its successive real zeros may identify an important subset of complex zeros.

In the simplest implementation of the TESPAR method [1], two descriptors are associated with every segment or epoch of the waveform.

These two descriptors are:

- The *duration (D), in number of samples,* between successive real zeros, which defines an e*poch*
- The shape *(S),* the *number of minima* between two successive real zeros.

The TESPAR coding process is made by using an alphabet (symbol table) to map the duration/shape (D/S) attributes of each epoch to a single descriptor or symbol [5]. The mode to get the TESPAR alphabet is well presented in [6][9].

The TESPAR symbols string may be converted into a variety of fixed-dimension matrices. For example, the

[1] Facultatea de Electronică Telecomunicaţii şi Tehnologia Informaţiei, Catedra Comunicaţii, str. Bariţiu 26-28, 400027 Cluj-Napoca, Eugen.Lupu@com.utcluj.ro

S-matrix is a single dimension 1xN (N- number of symbols of the alphabet) vector, fig.1. which contains the histogram of symbols that appear in the data stream (Nr. App). Another option is the A-matrix, which is a two dimensional NxN matrix that contains the number of times each pair of symbols appears at a "*lag*" distance of n symbols (fig. 2) [1][2]. The "*lag*" parameter provide the information on the short-term evolution of the analyzed waveform if its value is less than 10 or on the long-term evolution if its value is higher than 10. This bidimensional matrix assures a greater discriminatory power.

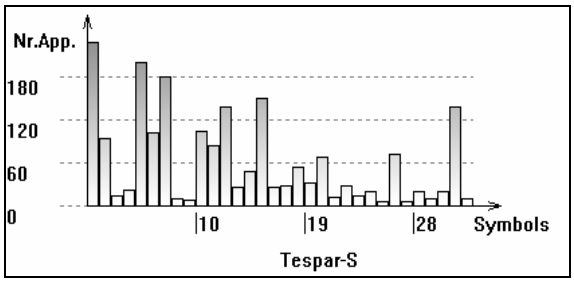


Fig. 1. TESPAR S-matrix

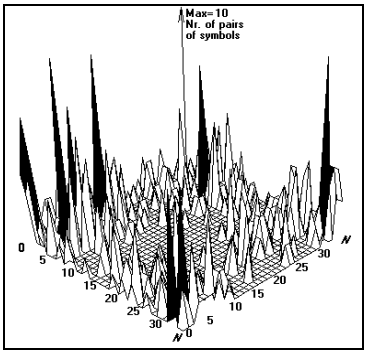


Fig. 2. TESPAR A-matrix

These matrices are ideal to be used as fixed-sized training and interrogation vectors for the MLP neural-networks. There are two main methods of classifying using TESPAR:

- Classifying using archetypes
- Classifying with neuronal networks [1][5].

This paper deals with the first method that was implemented on the system. An archetype is obtained by averaging several TESPAR A-matrices obtained from different versions of the same utterance. Such archetypes tend to outline the basic mutual characteristics and dim the particular cases that might appear in different utterances of the same word, for example.

The created archetype may be loaded in the database and then used. In the classification process, a new matrix might be created and then compared to the archetype. Many different forms of correlation can be used to achieve the classification. A threshold is required to establish whether the archetype and the new matrix are sufficiently alike; the archetype with the highest ratings is chosen after it has been compared to a threshold.

II. RECOGNITION SYSTEM OVERVIEW

Fig.3 shows the block diagram of the application, this being shared between PC and the DSP board. In order to run the application we have to load the ***VoiceR/ SpeakeR*** program on the EVM DSP board and to run the program A-Matrix Tools (on PC) if some reference TESPAR A-matrices are to be loaded by the DSP program.

The applications on the DSP board are built up using the CCS® (*Code Compose Studio*) environment that allows the fast application development using its own resources: C compiler, linker, debugger, simulator, RTDX (Real time Data Exchange) and DSP/BIOS components [7][8].

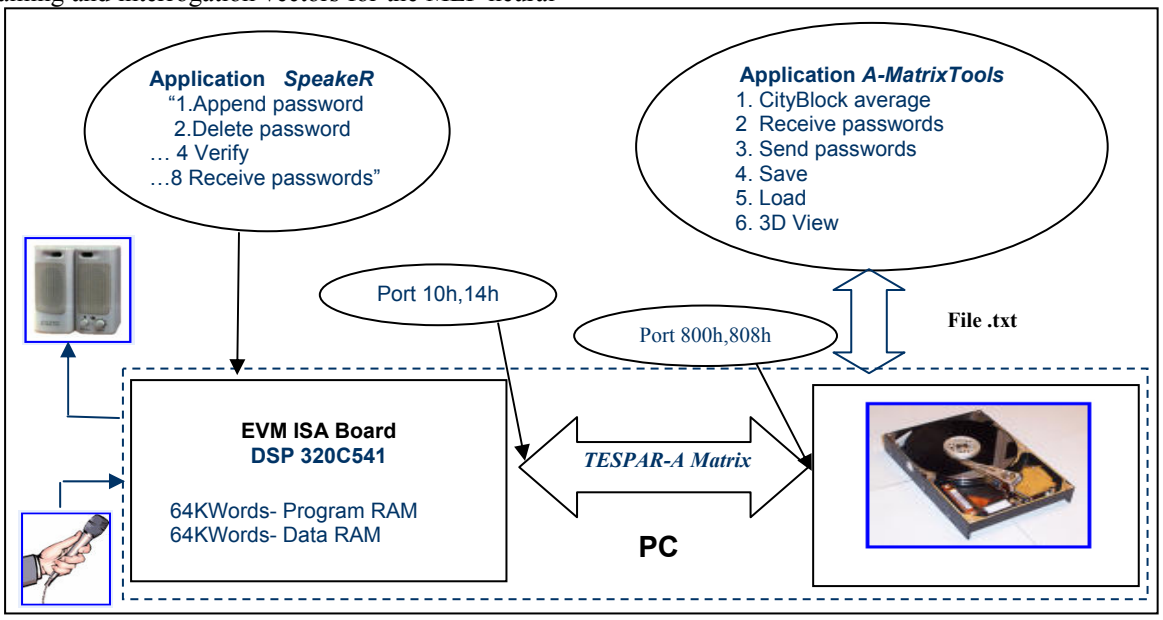


Fig. 3. The block diagram of the SpeakeR application

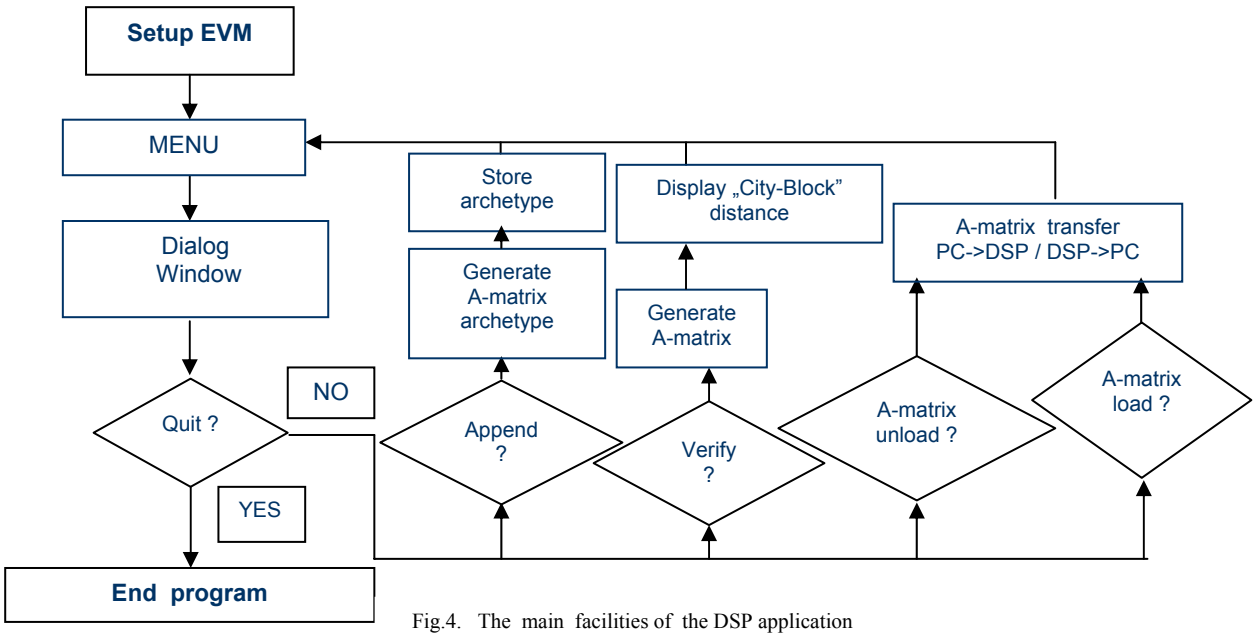


Fig.4. The main facilities of the DSP application

The main facilities of the ***SpeakeR*** application can be remark in the flow diagram, fig. 4.

The A-Matrix Tools program allows the TESPAR A-matrices transfer between the host PC and the EVM C541 board and offer facilities to extend the SpeakeR program operation. The tasks of this program are the following:

- TESPAR A-matrices collections transfer between host PC and EVM board
- TESPAR A-matrix transfer between EVM board and host PC
- save matrix/matrices collections to host PC hard disk in text files
- load matrix/matrices collections from host PC hard disk to EVM board
- 3D TESPAR A diagrams visualization City-block distance computation between two TESPAR-A matrices
- Archetype generation for TESPAR A-matrices collections.

For the polling communication between the host PC and the DSP board the following ports are employed; for data the port 800h (PC) and 10H (DSP) and for control 808h (PC) and 14h (DSP)[8].

III. EXPERIMENTS, RESULTS AND CONCLUSIONS

The applications facilitate to perform "on-line" speech/speaker recognition experiments. In the classification process, the distance calculation between the TESPAR A-matrices archetypes and the test matrices or parallel MLP neural networks may be employed. In this paper, the experiments focus on the use of "city-block"distance calculation between the A-matrices archetypes and test matrices in the classification task. The EVM board resources limit the number of enrolled speakers to 10 and the dictionary dimension for the speech recognition experiments.

A. *Speech recognition experiments*

Two types of experiments were been made, one using the ten digits as utterances and the other using different commands (left, right, up…). In the first experiment, seven speakers were enrolled for the system training and 10 speakers for the test. Each of the enrolled speakers uttered three times every digit for the training and ten times for the recognition. The results of this experiment are presented in fig.5. In this case an average recognition rate of 92% was obtained that we find to be good in the condition of using for test also utterances of not enrolled speakers.

For the other type of experiments, we used 10 commands words. In the "Test2", experiment the training was made by using the three utterances from every speaker to build its own archetype for every command. For the "Test3" experiment the archetype were been built by using three utterances from two enrolled speakers.

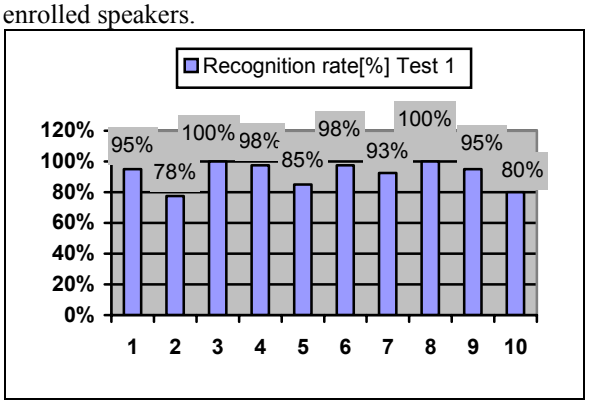


Fig.5. Digits recognition rate for the "Test 1" experiment

In every experiment to test the system all the speakers has uttered 10 times every command [10]. The results are presented in table 1. For the "test2" experiment, a

98% average recognition rate was provided by the system and for the "test 3" slightly lower than 97%.
The results of the experiments prove the high capabilities of the TESPAR method in the classification tasks, noticed also in [1][3]. The results generally are better than 90% and the DSP resources are not very highly used. In order to improve the recognition rate the employment of MLP neural networks for classification will be employed.

Table 1

Word	Recognition Rate [%] Test 2	Recognition Rate [%] Test 3
Up	100	100
Down	95	100
Left	100	100
Right	100	100
Enter	95	95
Cancel	100	95
Abort	100	85
Ok	100	100
Back	90	95
Forward	100	100

B. *Speaker recognition experiments*

Two types of speaker identification experiments were been made, one using different passwords for each enrolled speaker (his name) and the other using the same password. For the first experiment, eight speakers were enrolled. Each of them uttered three times the same password to provide the TESPAR-A matrix archetype. For the identification experiment, 10 sessions of attempts were been made during a week, each speaker uttering its own password 10 times in every session. The results of this experiment are presented in fig.6. In this case the system provide an average recognition rate of 96.1*%.*

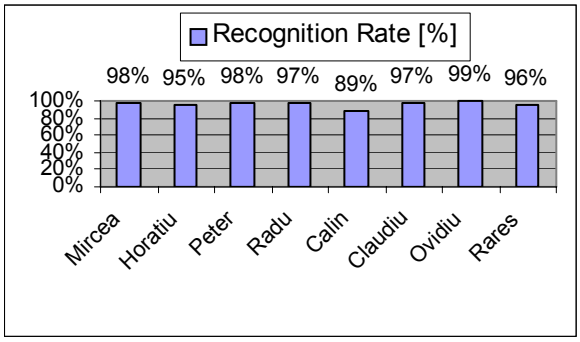


Fig. 6. The recognition rate for the experiment using different passwords

For the other experiment, the eight enrolled speakers use the same password. In this case, the training was made by using three utterances from every speaker to build its own archetype for every password.
To test the system all the speakers had uttered 10 times the password. The experiment was repeated for ten different short passwords. The recognition rates for every speaker and all passwords are presented in fig.7. An overall average recognition rate of *89.25%*

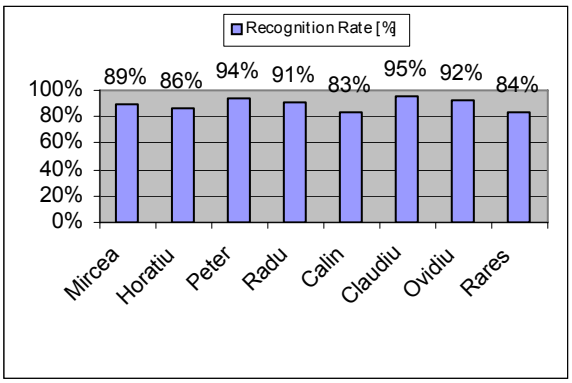


Fig. 7. The recognition rate for the experiment using the same password

was provided by the system for this experiment. The results of the experiments prove the high capabilities of the TESPAR method in the classification tasks noticed also in [1][5][9]. These results generally are better in the experiment that uses different passwords; the DSP resources are not very highly used. In order to improve the recognition rate the employment of MLP neural networks for classification is recommended. To validate the system more experiment are to be made using much amounts of utterances and different speakers are advisable to be tested. In addition, the effects of other signal processing algorithms applied before the coding process are to be studied.

REFERENCES

[1] King, R. A., Phipps, T. C. "Shannon, TESPAR and Approximation Strategies", *ICSPAT 98,* Vol. 2, pp. 1204-1212, Toronto, Canada, September 1998.
[2] Phipps, T.C., King, R.A. "A Low-Power, Low-Complexity, Low-Cost TESPAR-based Architecture for the Real-time Classification of Speech and other Band-limited Signals" International Conference on Signal Processing Applications and Technology (ICSPAT) at DSP World, Dallas, Texas, October 2000, *www.dspworld.com/icspat/spchrec.htm.*
[3] Voelcker, H. B. "Toward A Unified Theory of Modulation Part 1: Phase-Envelope Relationships*", Proc. IEEE*, vol. 54, no. 3, pp 340-353, (March 1966).
[4] Requicha, A. A. G. "The zeros of entire functions, theory and engineering applications" Proceedings of the IEEE, vol. 68 no. 3, pp. 308-328, March 1980.
[5] Lupu, E., Feher, Z., Pop, P.G. **"**On the speaker verification using the TESPAR coding method", IEEE Proceedings of Int. Symposium on "Signals, Circuits and Systems", Iassy, Romania, 10-11 July 2003, pp.173-176, ISBN 0-7803-7979-9
[6] Lupu, E., Pop, P.G., Digital speech processing. Analysis and recognition, Cluj-Napoca, Risoprint 2004, ch.11, pp.164-175
[7]*** Texas Instruments - TMS320C54x DSP Reference Set
[8]*** Texas Instruments – TMS320C54x Evaluation Module Tehnical Reference
[9] Lupu, E., Moca,V., Pop, P.G "Environment for speaker recognition using speech coding" Proc. of Communications 2004, Bucharest, 3-5 June, Vol. 1, pp.199-204, ISBN 973-640-036-0
[10] Lupu, E., Pop, G. P., Pătraș, M." Low Complexity Speaker Recognition System Developed on the DSP TMS320C541 Board" Proceedings of the 9th International conference "Speech and Computer" SPECOM' 2004 , 20-22 sept. 2004, St. Petresburg pp. 398-402 ISBN 5-7452-0110-x

Buletinul Ştiinţific al Universităţii "Politehnica" din Timişoara

Seria ELECTRONICĂ şi TELECOMUNICAŢII
TRANSACTIONS on ELECTRONICS and COMMUNICATIONS

Tom 51(65), Fascicola 2, 2006

Streaming Multimedia Information Using the Features of the DVB-S Card

Radu Arsinte, Eugen Lupu[1]

Abstract – This paper presents a study of audio-video streaming using the additional possibilities of a DVB-S card. The board used for experiments (Technisat SkyStar 2) is one of the most frequently used cards for this purpose.
Using the main blocks of the board's software support it is possible the implement a really useful and full functional system for audio-video streaming. The streaming is possible to be implemented either for decoded MPEG stream or for transport stream. In this last case it is possible to view not only a program, but any program from the same multiplex. This allows us to implement a full functional system useful for educational purposes.
Keywords: Multimedia, DVB-S, Networking

I. INTRODUCTION

Many consumers are currently using analog TV cards to watch TV on a PC screen. This represents a feature that has already combined the TV and PC experiences. The market for PC analog TV cards has grown over the last few years. With its viewing experience and enhanced interactive data services, a DVB-PC card is a more compelling solution than current analog TV tuner cards for PCs. The DVB experience is significantly enhanced compared with analog TV, as the entertainment is combined with the real-time interactivity of the PC. A potential analog TV card user is inclined to spend a little more for a combination card that supports both analog and DTV broadcasts in order to avoid the risk of quick obsolescence.
Broadcasters will be encouraged to create more DTV content if the installed base of DTVPC cards grows, which will help the DTV industry as a whole.
The main facility for DVB-S ([1]) card users is the possibility to combine local audio-video viewing with streaming. In this way it is possible to give access to audio-video stream for other clients from the same network.
Unlike the Internet, broadcast networks have been optimized for the transmission of rich content to large numbers of users in a predictable, reliable, and scalable manner. The advantages they bring to the infrastructure are as follows:
• Broadcast networks are designed to carry rich, multimedia content. Traditional broadcast networks—including television, cable, and satellite networks-have been explicitly designed to deliver high-quality, synchronized audio and video content to a large population of listeners and viewers. In addition, over the last years many of these networks have migrated from analog to digital transmission systems, thus greatly enhancing their ability to carry new types of digital content, including Internet content.
• Broadcast networks are inherently scalable. By virtue of their point-to-multipoint transmission capability, broadcast networks are inherently scalable. It takes no more resources, bandwidth or other provisions, to send content to a million locations as it does to one, as long all of the receiving locations are within the transmission footprint of the broadcast network. In contrast, with the traditional Internet, each location that is targeted to receive the content will add to the overall resources required to complete the transmission.
• Broadcast networks offer predictable performance. Again, by virtue of the point-to-multipoint nature of transmission on broadcast networks, there are no variances in the propagation delay of data throughout the network, regardless of where a receiver is located. This inherent capability assures a uniform experience to all users within the broadcast network.
Our experience in TS generation ([2],[3]) and use was extremely useful in this work.

II. DVB-S CARD ARCHITECTURE

Hardware Architecture
The DVB card consists of several components, both on the hardware and software aspects. The basic building block as shown in Figure 3 serves as a platform for standard-definition (SDTV, resolution of 720*576, which is defined as MP@ML (Main Profile

[1] Facultatea de Electronica, Telecomunicaţii şi Tehnologia Informaţiei, Catedra Comunicaţii, Str. Gh. Barițiu, 26-28, Cluj-Napoca, email: Radu.Arsinte@com.utcluj.ro

at Mail Level)) television program decoding. The main schematic is close to the stand-alone version presented in [4].

A DVB PC card is composed form a channel-decoding module and a source-decoding module. The channel-decoding module deals with the transmission over the physical media and its main task is to deliver an error free signal to the source-decoding module. It is usually grouped under the term "forward error correction" (FEC) as it provides error detection and correction to the received signal.

On the other hand, the source decoding module descramble, demultiplex and decode the audio and video signal for reproduction. The main task of each functional module is presented in figure 3. The main tasks of source decoding, in a DVB PC-Card, are performed by the host processor (usually at least from Pentium III class).

We are briefly describing the functions of each module:

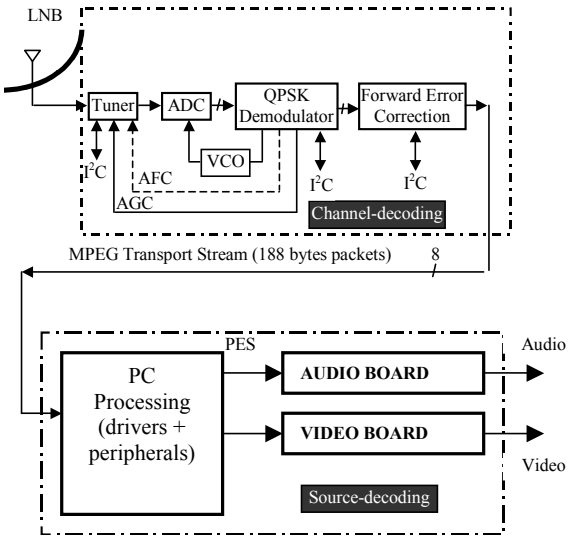


Figure 1. The Hardware Architecture of the DVB Card

1. Tuner

The tuner (sometimes known as the 'front-end'), generally select one of the RF (Radio Frequency) channel and converts it into IF (Intermediate Frequency).

2. ADC (Analogue to Digital Converter)

The ADC receives the analogue signals and converts it into a digital signal for QPSK processing.

3. QPSK Demodulation

This is the key element in the channel decoding process: It performs digital demodulation and half-Nyquist filtering, and reformatting/demapping into an appropriate form for the FEC circuit. It also plays a part in the clock and carrier recovery loops, as well as generating the AGC (automatic gain control) for control of the IF and RF amplifiers at the front end.

4. FFT (Fast Fourier Transform) processor

The FFT processor provides timing and frequency synchronization, channel estimation and equalization, generation of optimal soft decisions using the channel state information, symbol and bit de-interleaving.

5. Forward Error Correction (FEC)

The FEC block performs de-interleaving, Reed-Solomon decoding and energy dispersal de-randomizing. The output data are the 188 bytes transport packets in parallel form (8 bit data, clock and control signals). The channel-decoding module is highly integrated and it is usually offered as a single chip solution (module 2 to 5).

Software Architecture

The software required to power the DVB set-top boxes or PC cards is apparently more complex than the hardware requirement since most of the hardware are already highly integrated. An example of the software model used for the development is presented in figure 2.

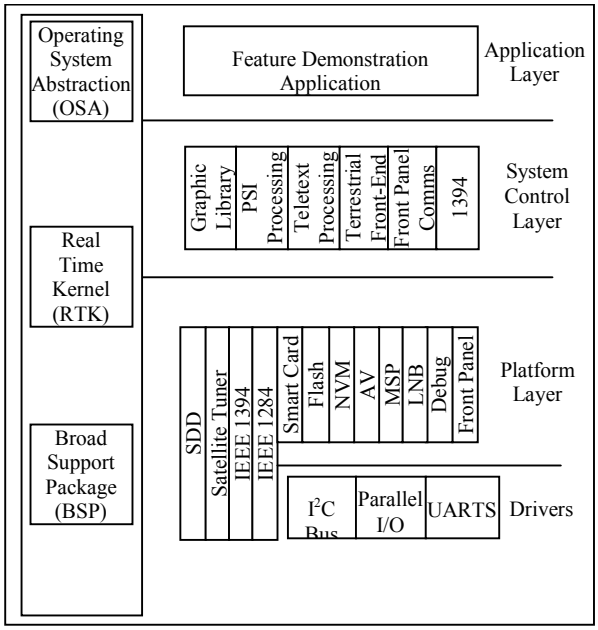


Figure 2: Software Architecture for the DVB-S Card

The task of decoding is mainly done in hardware, the software deals with configuring these devices upon power up and to handle user requests. Most of the software modules are required to program the EPG (Electronic Program Guide) and Interactive TV (if included). Not all the components are required for a specific function (TV reception, for example), but the software driver could be updated permanently to match new requirements in functionality. Our experience in TS generation and use was extremely useful in this work.

III. STREAMING OPTIONS

Streaming information in a DVB-S based environment, is not different basically from a normal network-based information streaming. The main differences are related with a minimal bandwidth necessary to transfer al large amount of information,

characteristic to multimedia information. A rough estimation of the compression/storage ratio obtained by a common MPEG2 encoding is: 1.8Gbytes per

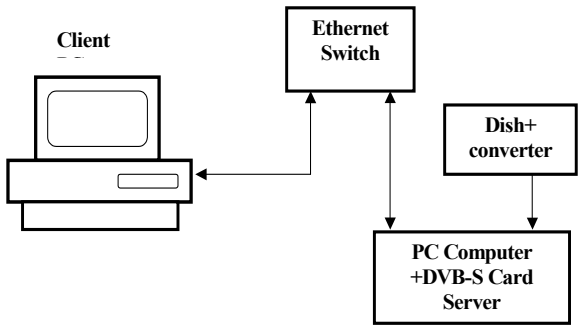


Figure 3.a Test connection using a switch

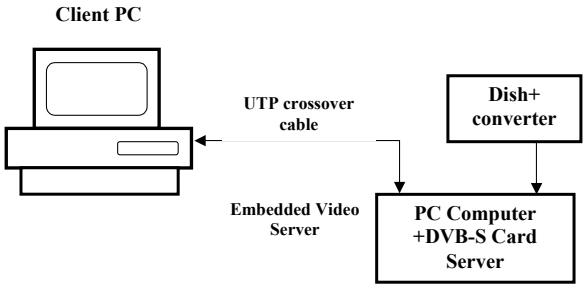


Figure 3.b Test connection using a direct cable

hour, for instance to store a full screen DVD in average quality on your harddisk. In terms of real-time streaming, this means about 4 Megabits per second (4Mb/s), which can also be quantified as 582 KiloBytes per second (582KB/s).
Here is a short explanation of the above calculations:
1. 2GigaBytes of stored video /60 mins /60 secs = 596523 bytes per sec
a. 596523 bytes per sec /1024 = 582 KB/s (Kilobytes per sec)
*2. 596523 bytes per sec *8 = 4772184 bits per sec*
a. 4772184 bits per sec /1024 /1024 = 4 Mb/s (Megabits per sec)

Hardware configuration
The test server uses basic functions of the Video server.
The minimal configuration of the test system is composed from a server (PC system with DVB-S board) and a client (a normal PC). The proposed configurations are presented in Fig.3 (a and b), similar with the configurations presented in ([5]) or ([6]).
If this test system (involving a local network and a DHCP server – Fig.3.a) is not possible to be implemented, it is possible to use a simplified version based on a direct connection using a UTP crossover cable (Fig.3.b).

Software configuration
The main element for streaming is the Server4PC utility (described also in [7]) delivered with SkyStar board. This utility must be configured properly to perform the requested tasks.

It is necessary to provide the IP-address of the network interface., the multicast stream is sent for for distribution. A common address used in most test applications is 192.168.0.1.
The second information, necessary for IP multicast is the multicast address and the port, where the stream is located. The multicast IP range is specified in RFC1112. Multicast IP address are defined to the range between 224.0.0.0 to 239.255.255.255. To send the stream to all clients from the subnet, it is necessary to use the multicast IP 224.0.0.1. The multicast port number, can be chosen in the range 0 and 65500. The first 1024 ports are reserved for IP services.
The main settings are presented in a screen capture from figure 4.

The server part uses a well-known program called VLC media Player, developed in VideoLAN project. The main screen of the program is presented in figure 5.

The VideoLAN project targets multimedia streaming of MPEG-1, MPEG-2, MPEG-4 and DivX files, DVDs, digital satellite channels, digital terrestial

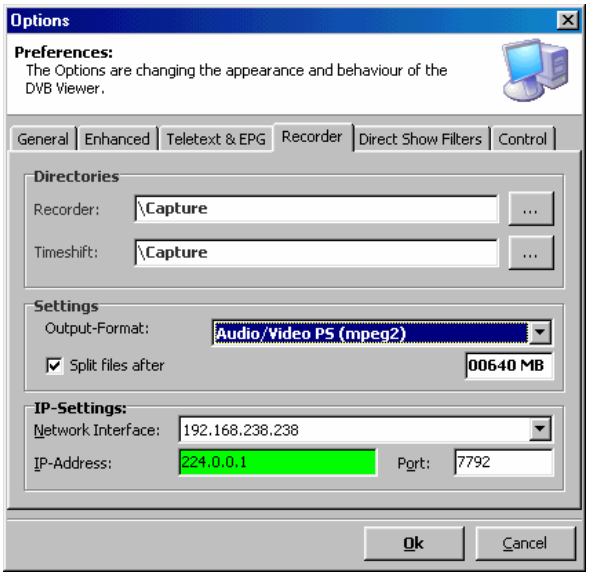


Figure 4. DVB Viewer configuration

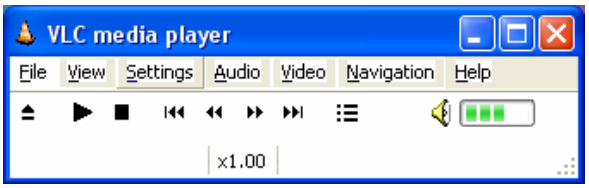


Figure 5. Main screen of VLC Media Player

television channels and live videos on a high-bandwidth IPv4 or IPv6 network in unicast or multicast under many OSes. VideoLAN also features a cross-plaform multimedia player, VLC, which can be used to read the stream from the network or display video read locally on the computer under all

GNU/Linux flavours, all BSD versions, Windows, Mac OS X, BeOS, Solaris, QNX, Familiar Linux.
The latter feature is one of the most important points for VideoLan, being in fact the only free software being able to fill the gaps between video and audio formats on different platforms, while it's even supporting ARM and MIPS based hand-held devices, being ready to be integrated in consumer-grade embedded solutions. VideoLan supports the functionality offered by the IVTV driver, taking advantage of hardware encoding on a growing number of cards. It is basically made out of two software components: VideoLanClient (VLC) and VideoLanServer (VLS). VLC is a multipurpose streaming a/v client and source: it can play streams as well it can capture and send a stream to another VLC or VLS. VLS is just a server which has no visualization output for the streams it handles, its fully capable of capturing and streaming from the local machine, as well reflect streams coming from other VLC/VLS nodes. The way VideoLan distributes functionality among its nodes is therefore very flexible and permits to easily build streaming topologies to distribute real-time audio/video streams.

IV. RESULTS

The proposed configuration was implemented in a local network with 5 computers – one as server and four clients. The performance of the computers is the following:
Server: Pentium II – 400MHz -10GB- 128MB RAM
Clients: Pentium II-III – 233...800MHz - 128-256MB RAM
Experiments proved that the streaming is possible for all the computers, with some limitations for low-grade computers (Pentium II – 233).
The goal of the work being an investigation of this technology for an educational use, the results could be interpreted as satisfactory.
We tried to extend the streaming over a wider network, this fact being extremely difficult without modifying the network security settings.

V. CONCLUSION AND FUTURE WORK

Our activity brought us the following achievements:
1. Installing and configuring the driver of DVB-S card, eliminating any incompatibilities;
2. Streaming of information in a local network;
3. Configuring the server for the PC containing the DVB card;
4. Configuring the client applications using both the board's viewer and VLC;
5. Measurements of the streaming performance in the network.
The experiments revealed that an efficient streaming is possible to be implemented, but the quality of the network is essential. The uninterrupted streaming (and of course viewing) could be realized only in moderate loaded networks.
All the test work was done in a Windows environment (both for client or server). The next experiments will try to verify and measure the same performance in a Linux based environment, or a hybrid environment (for example Linux for server, and Windows for clients).
We will try also to expand the test network, to be able the send Transport Streams in a large area (for example in entire faculty network).

REFERENCES

[1] * * * - Technisat, *Sky Star 2, User Guide* – 2004.
[2] Radu Arsinte, Ciprian Ilioaei - *Some Aspects of Testing Process for Transport Streams in Digital Video Broadcasting* – Acta Technica Napocensis, Electronics and Telecommunications, vol.44, Number 1, 2004
[3] Radu Arsinte - *A Low Cost Transport Stream (TS) Generator Used in Digital Video Broadcasting Equipment Measurements* – Proceedings of AQTR 2004 (THETA 14) - 2004 IEEE-TTTC-International Conference on Automation, Quality and Testing,Robotics May 13-15, 2004, Cluj-Napoca, Romania
[4] * * * , *OM5730, STB5860 (Set-Top Box) STB concept*, Application note, Philips Semiconductors, 1999
[5] * * * - Technisat, *HOWTO IP-Streaming with DVBViewer TE* – 2004
[6] Radu Arsinte - *Implementing a Test Strategy for an Advanced Video Acquisition and Processing Architecture*, Acta Technica Napocensis, nr.2/2005
[7] Radu Arsinte - *Effective Methods to Analysis Satellite Link Quality Using the Built-in Features of the DVB-S Card*, Acta Technica Napocensis, nr.1/2006

Buletinul Ştiinţific al Universităţii "Politehnica" din Timişoara

Seria ELECTRONICĂ şi TELECOMUNICAŢII
TRANSACTIONS on ELECTRONICS and COMMUNICATIONS

Tom 51(65), Fascicola 2, 2006

Telemedicine Application for Distant Management of Oro-maxilo-facial Tumors

Bogdan Orza, Aurel Vlaicu, Adrian Chioreanu, Vlad Mihalcea[1], Laura Grindei[2]

Abstract – TeleOralTum software is intended to collect on one server specific data from departments that work on the facial cancer diagnosis. Classical medical services imply the existence of a direct link between medical staff in different departments and the patient. Telemedicine is an old concept, which arise debates for more then 30 years. The application that we developed proves the need for a diagnose and management system that will allow a rapid gathering of data, a rapid elaboration of the diagnose, but also elaboration of reports and estimates that are so much needed in this field of medical research. The application works on a three layer distributed architecture, thus taking advantage of a high security for the patient data, easiness in splitting the work among software development teams, and also the easiness in which other medical departments can be added to the application. The user administration section is used to divide the accessibility domain for different types of client users. The architecture of TeleOralTum is an innovative one and has at its origin open-software tools.
Keywords: telemedicine, medical management, Hibernet, telediagnosis.

I. INTRODUCTION

Telemedicine is intended to raise the standards of the actual medical act, by using electronic content management and state of the art telecommunication techniques. Classical medical services are limited by a certain geographical area, area defined by where the medical specialist work, and also by the pour capabilities of structuring data. The medicine applications are a modern and viable alternative in the development of the medical science. Among the benefits of telemedicine we can see the access of rural area patients to state of the art medical solutions that can be found only in urban areas. The telemedicine applications are divided in clinical related applications, and non clinical related applications. Among the last ones we can find the medical educational applications, and the information management applications that use digital telecommunications. The clinical applications deal with diagnosing and treating patients using teleconference systems, telediagnosis systems, or telemonitoring systems of patients.

By looking at the specifications for implementation of the TeleOralTum application we can classify it as a non clinical tele-oncologycal application, which serves at the study of oro-maxilo-facial cancer.

The application wants to implement an electronic management system for the oro-maxilo-facial tumors cases, which with the help of telemedicine will allow the collaboration between different medical centers involved in the treatment of this kind of disease. Telemedicine is at the boundary between medical science and state of the art technology, and appeared with the purpose of enhancing the existing medical services (Figure 1).

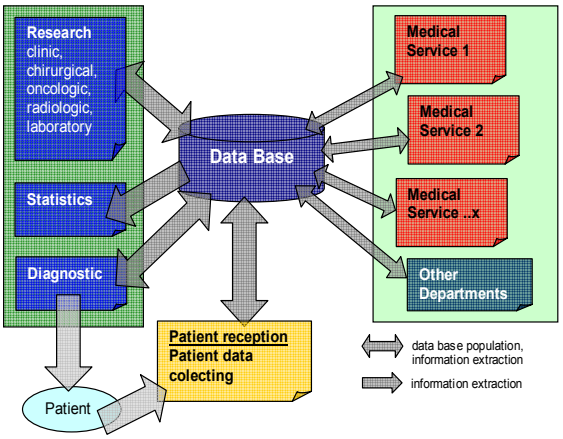


Figure 1 The architecutre of the telemedicine/telediagnosis application

The TeleOralTum is a telemedicine project developed by the Center of Multimedia Technologies and Distance Education (CTMED) from the Technical University of Cluj-Napoca, in collaboration with the Medicine and Pharmacology University from Cluj-Napoca. The project aims to develop a web application using the Microsoft .Net technology and the database server Microsoft SQL Server. The application brings to the user an attractive, intuitive

[1] Technical University of Cluj Napoca, Faculty of Electronics, Telecommunications and Information Technologies, Communication department, 26-28 G. Bariţiu, 400027 Cluj Napoca, e-mail Bogdan.Orza@com.utcluj.ro
[2] Electrical department 26-28 G. Bariţiu, TUCN,

and easy to use graphical interface. We also decided to implement this application in such a way that it will be easy to develop it in the future. When designing the software architecture we took into consideration that in the future other medical departments, which have interest in the results from this field of research, will be integrated. TeleOralTum can be very easily adapted, the software intervention in this direction are very small and they are not resource consuming. The application is portable on multiple database servers, even without the recompiling the source files.

This project main objective is to develop a database containing the cases of patients with oral cancer. The implementation of the application that will generate and manage this database has the following objectives:

- an improved management of information gathered from different medical departments involved in the diagnosis and monitoring of oral cancer,
- the elaboration of an exact epidemiologic report,
- the evaluation of risk factors that may cause tumors,
- standardization of the diagnosis, cure criteria, and patient monitoring,
- monitoring in time of medical cases.

II. APLICATION ACHITECTURE

The TeleOralTum architecture is distributed on three layers (Figure 2), in order to take advantage of the distributions capabilities. The three layer architecture is composed of:

- the web layer,
- the application layer,
- the data layer.

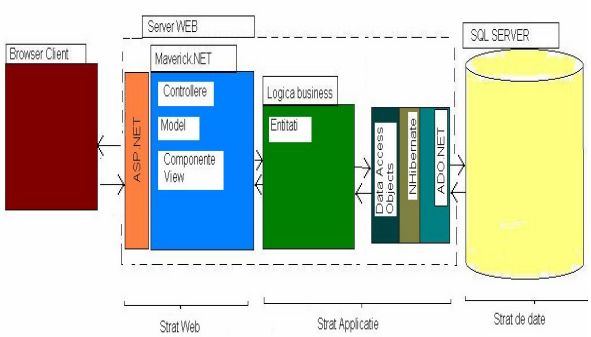


Figure 2 The architecture of the TeleOralTum application

The web layer is responsible for taking the client requests and sending them to the application layer. When the application layer finishes serving the request, the web layer composes the answer that will be sent back to the client, and displayed. The application layer is responsible for the logics of the application. The data layer contains the database server and the database. The three layer architecture increases the reusability of the software components for similar applications, as well as in the case when we need to move the application on a different database, or in the case when the web layer is replaced by a different web layer or by a windows interface. The security of a three layer application is increase because one is able to implement three points that check the authenticity of the client: at the web gate, at the application gate, and at the database entry.

The Web Layer

Microsoft ASP.NET offers for the web layer an architecture model based on treatment of events triggered by the user. The server keeps in a HTTP variable the state of all the HTTP parameters from every page, and in the moment when an entity from the web browser is modified, the server knows what event was triggered, and will treat that event in a special routine. This approach is based on the command of the desktop graphical interfaces. ASP .NET doesn't offer the implementation of the MVC frame (Model-View-Controller). The Model-View-Controller (MVC) frame consists in the division of the application in function blocks: the Model, the View Components, and the Controller. What we have chosen is an open source project, called Maverick.NET, which is a replica of the MVC framework called Maverick, developed in Java technology. Maverick .NET is minimal and extensive MVC framework which offers the following functionalities:

- the routing of requests and answers taking into account a configuration map written in XML,
- the construction of View controls based on the ASPX or XSLT page transforms, or based on the type of the model defined for a certain request,
- the possibility of globalizing (of internationalization) of the application,
- the easy extension of the framework.

The Application Layer

The application layer consists of Business Logic objects, Data Access Objects and Entity objects. This layer is designed of the ADO.NET library, a set of utilitarian classes for the management of data base connections using .NET. On top of the ADO.NET we placed the Nhibernate [NHibernate] framework, a persistence layer that makes the connection between the object world of the business entities used by the business logic, and the rational world of the tables used in the database. We didn't use ADO.NET, because our application works directly with the Nhibernate framework, and thus is independent of the database server. So we didn't use the Microsoft layers that regard the connection and management of databases, because as themselves stated in the article [Microsoft1] don't correspond to the needs of business application. From the ADO.NET technology, Nhibernate uses the pooling connection technology [Gamma], which manages the data base

connections. The Nhibernate technology is a predecessor of the successful Hibernate technology, that was developed in Java and that wined the race of the relational-object mapping frameworks. We can read about the Persistence Layer in the [Mark]. The idea of the Persistence Layer framework consists in the existence of a special sub-layer of the application which offers an interface for the business entities.
On the tables from the data base we mapped the business entities using XML configuring files. The link between the relational and object world is not made in a programmatic fashion because the XML files can be modified without the need of recompiling the application.

The advantages of this technology are: the independence from the database server that we use for the application; the integration of the ADO.NET technology, which lets us take advantage of the pooling technology; the use of the business object entities to manage data from the data base. The disadvantages of using this technology consist in the increased code and memory redundancy, due to the fact that the server has to maintain the map of the business entity objects, and because the translation of the HQL (Hibernate Query Language) queries in data base specific queries increases the time of task completion.

Nether the less we must remember that C# and ASP.NET are technologies that offer a much smaller execution time then Java technology.

Data Base Layer

This is the most sensitive layer of the application because it contains the data itself. This layer consists of a Microsoft SQL Server 2000 data base server, which will host the data for our application.

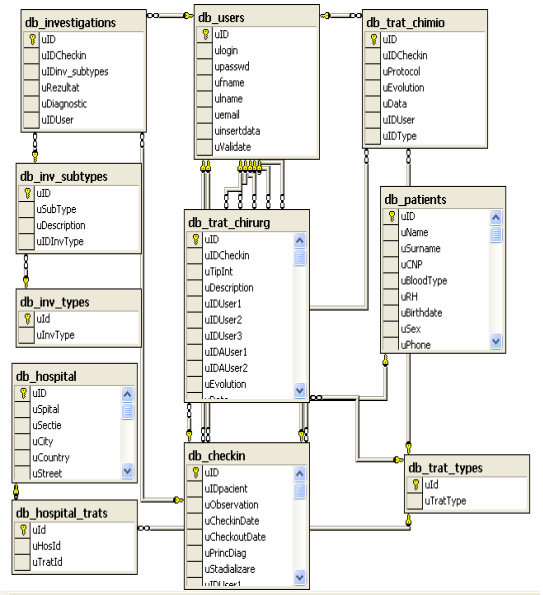


Figure 3 The data base table diagrams

Because we use a persistence layer for the management of the date in the data base, we don't need to use stored procedures, all the management queries used to manipulate the data from the data base being written in HQL (Hibernate Query Language), language that is independent of the data base platform. We choose the Microsoft data base server because is a very stable, reliable tool, which proved its efficiency in other applications that we developed.

III. APPLICATION DESCRIPTION

The application defined five different roles: *the general administrator, the department administrator, the medic, the nurse* and *the guest.*

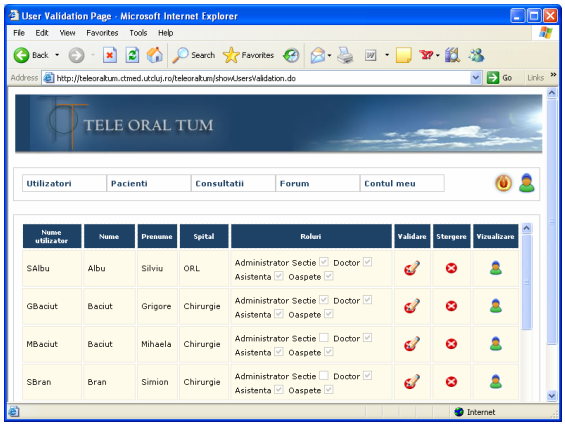


Figure 4 The user management (administration) page

Every user that joins the application has to choose a role. If the user subscribes as a guest, he will be automatically validated, and so he will be able to enter in the forum to read the existent debates. For every other role the user request for subscription must be validated according to the algorithm presented in each role description. At the same time, in order to increase the flexibility of the environment, each user is able to create its own user profile. By simply validating the roles of a user, that person can be simultaneously a department administrator, a medic, a nurse or a combination of the above.
The general administrator is able make modifications in the user accounts and in their profiles. He's also able to delete patient's files after following a two step confirmation procedure.

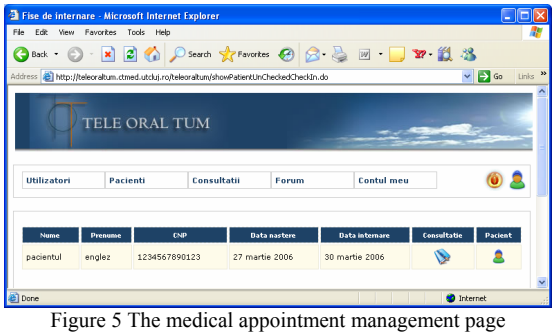


Figure 5 The medical appointment management page

If we need to visualize the personal user data we have to press the visualize button. The administrator is not able to see the user passwords, even if he looks for them in the data base, because they are encrypted with

the MD5 algorithm. Besides the capability to validate and modify the roles of users in his validation domain, the administrator can't change their personal data. The visualization button let the administrator user know the exact identity of the user that he wants to validate or delete. In a similar manner works the validation of medics and nurses by the department administrator, or the validation of nurses by the medics.

The department administrator manages the patients and the lower rank users (medics and nurses) from his department, but is not able to affect the users form parallel departments.

The nurse is supposed to insert the patient's information in the database. Practically she creates a hospitalization file for each of the patients. Here the file will be completed with the patient personal data and hospitalization observations. The application offers to the nurse an array of reports that shows the list of patients with an open hospitalization file or with investigation files from her department.

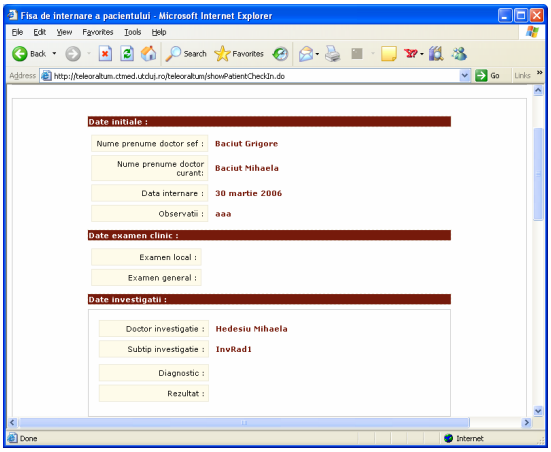


Figure 6 The page that shows hospitalization file

The medic can visualize the hospitalization files (Figure 6) of his patients. At the same time, the medic can open an investigation or treatment file at other departments then his own. The files contain textual information about the diagnostic, treatment, investigation, as well as visual information as images files, video files, DICOM files etc., which will allow the currant medic to give an accurate diagnostic. The investigation and treatment files can be seen by the department administrators, nurses of that department (Figure 7), but can be modified only by the medic that received the patient (Figure 8).

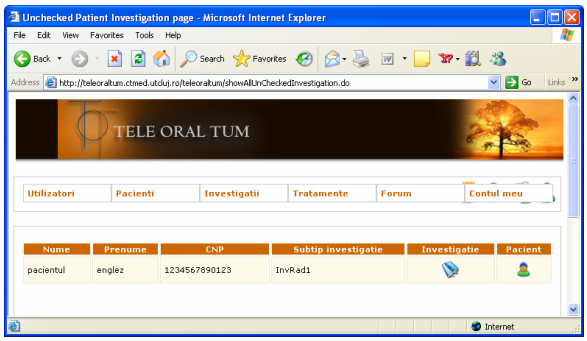


Figure 7 The page that show the investigation and treatment files

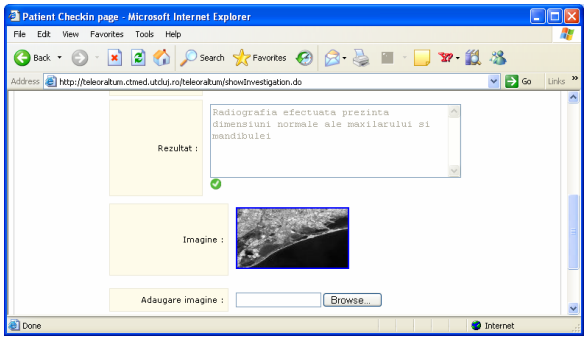


Figure 8 The page that allows to edit the investigation and treatment file

Taking into account the need of communication between the users of this system, that is intended to be also a tool used in oro-maxilo-facial cancer research, the application contains also a forum module for each medical department, which gives the users the ability to open new debate themes.

IV. CONCLUSIONS

From the technology point of view the application has a degree of novelty in the field of telemedicine applications. The novelty consists in the combined use of .NET C# with the Nhibernate framework and the dynamical connection to the database using the HQL language.

TeleOralTum is in the finalization status, all that needs to be added is a series of reports that are necessary for medical departments. The reports will extract from the data base information of the current patients as well as information stored in the archive, offering the medic the ability to evaluate rapidly the degree of wellness of the patient by comparing the diagnostic and visual information from the data base.

REFERENCES

[1] E. Gamma, E. Helm, R. Johnson, J. Vlissides, „Design Patterms", Addison-Wesley, ISBN 0-201-63361-2, 1995

[2] M. Grand, „Java Enterpise Design Patterns", John Willey and Sons, Inc ISBN 0-471-33315-8, 2002, 2002

[3] C. Bauer, G. King, "Hibernate in Action", Manning Publication Co, ISBN 1932395-15-X, 2005

[4] http://nhibernate.sourceforge.net/

[5] C. Cavaness, "Programming Jakarta Struts", O'Reilly, 2003

[6] http://icsl.ee.washington.edu/projects/emedicine/emed-spie2001.pdf

[7] http://www.trestlecorp.com/corp_dllit/Broch_MR_4pg.pdf

[8] http://www.onk.ns.ac.yu/Letopis/LSA2001/PDFs/oa-malbasa.pdf

[9] Liqiong Deng Poole, "Learning through telemedicine networks" Proceedings of the 36th Annual Hawaii International Conference 2003

[10] R. Komiya, "A proposal for telemedicine reference model for future standardization", Enterprise networking and Computing in Healthcare Industry, 2005. HEALTHCOM 2005. Proceedings of 7th International Workshop

[11] H. Dhillon, P. Forducey, "Implementation and Evaluation of Information Technology in Telemedicine", Proceedings of the 39th Annual Hawaii International Conference.

Buletinul Ştiinţific al Universităţii "Politehnica" din Timişoara

Seria ELECTRONICĂ şi TELECOMUNICAŢII
TRANSACTIONS on ELECTRONICS and COMMUNICATIONS

Tom 51(65), Fascicola 2, 2006

The concept of time-frequency-phase analysis

Cornel Ioana[1], Cédric Cornu[1], François Léonard[2], Arnaud Jarrot[1], André Quinquis[1]

Abstract – The time-frequency representations constitute the main tool for analysis of non-stationary signals arising in real-life systems. There is a huge number of time-frequency approaches adapted for a wide class of signals. The common drawback is that these methods use only a part of instantaneous phase information. In this paper, we propose a new concept that takes advantage on time, frequency and phase characteristics of the signal. The extraction of time-frequency-phase characteristic are done in two steps : conventional time-frequency analysis and phase continuity analysis.
The results will prove the efficiency of this concept in the case of two digital modulation types : FSK and PSK.

I. INTRODUCTION

The field of signal analysis is a very important element in a system of representation and/or information extraction. Considering the generally non-stationary character of the observations encountered in real applications, their analysis in the field of time-frequency constitutes the best suited technique to identify the relevant structures for information processing. There are a large number of approaches developed in time-frequency analysis field. In spite of their diversity it is commonly used to classify the time-frequency methods as *non-parametric* or *parametric* representations. The non-parametric representations are mainly represented by the Cohen's class distribution [1]. There are numerous other works attempting to provide "nice" time-frequency visual information (see [2] for some widely followed directions). In order to better describe the time-frequency content of a given class of signals, the parametric time-frequency methods have been also introduced [2], [3]. While they are relatively efficient for a given signal class their adaptation of other signals classes is very difficult and, sometimes, even impossible. One example is the polynomial phase modeling [4] which could be difficultly applied to analyze fast-varying time-frequency structures.

A common deficiency of both types of TFRs is the marginally use of the phase information. Namely, since the large majority of the methods are mainly looking only for the instantaneous frequency law (IFL; ie, the first derivative of the instantaneous phase law), the phase law remains almost unexplored. The usual TFR ignores the phase in spite of its richness. For example, the spectrogram is a magnitude time-frequency representation.

Recently, the phase in the time-frequency domain becomes to be studied thanks to the complementary information about the analyzed signal which is brought out [5]. The idea is straightforward : while the phase is one of the fundamental parameters of a signal, its exploitation might lead to a more efficient characterization.

In this paper we propose the joint use of magnitude time-frequency information and the phase of the signal. The work context is the extraction of time-frequency information from uncertain systems. Digital modulation identification and analysis of natural signals form both an uncertain systems since the overall properties are unknown, their estimation being the purpose of the analysis.

Contrary to [4] the phase will be evaluated in the original time-domain with help of information provided by the magnitude TFR. Proceeding in this way, we show that the phase analysis will be less affected by the noise or other component than in the case of direct estimation of the phase.

The paper is structured as follows. In the section 2 we present a short overview of the conventional time-frequency tools. In the section 3 we investigate the properties of the phase in analyzing time-frequency structures. In the section 4 we describe the concept of time-frequency-phase analyzer. The results provided in section 5 illustrate the benefits of this concept. We conclude in section 6.

II. OVERVIEW OF CONVENTIONAL TIME-FREQUENCY ANALYSIS

The time-frequency analysis is a very challenging research field thanks to its importance for the understanding of real-life signals. A good interpretation of a signal guarantees the efficiency of its processing. For this reason a lot of researches in signal processing concentrated their attention on time-frequency methods. It would be a very difficult task to synthesis all of works done in the past twenty years and it is not the purpose of this section. We are just

[1] ENSIETA, E3I2 Laboratory (EA3876) 2 rue François Verny, 29806, Brest FRANCE, e-mails: [ioanaco, cornuce, jarrotar, quinquis]@ensieta.fr
[2] Hydro-Québéc (IREQ) 1800 Bd. Lionel-Boulet, Varennes, Québec, CANADA, e-mail: leonard.françois@ireq.ca

proposing a general framework, starting from [6], that allows us to point out on common problems arising in this field. In the most general case any TFR of a multi-component signal

$$s(t)=\sum_{k=1}^{N} A_k e^{j\phi_k(t)} \quad (1)$$

can be expressed as

$$TFR_s(t,\omega)=\sum_{k=1}^{N} 2\pi\delta\left(\omega-\phi_k'(t)\right)*_{\omega} FT\left\{e^{j2\pi Q_k(t,\tau)}\right\}+CrossTerms \quad (2)$$

where FT stands for the Fourier transform, ϕ_k' is the first-order derivative of the phase law of the k^{th} component of the signal (ie the IFL of this component), $*_\omega$ is the spectral convolution operator, τ is the lag used for the computation of the TFR and $Q_k(t,\tau)$ is a function measuring the spreading of the time-frequency energy of the k^{th} component around its IFL. This function is helpful in the mono-component signal case in order to appreciate the performances of a considered TFR [6]. It measures the inner-interference terms (ie the artifact generated in the case of non-linear IFL) and, ideally, this function should be 0. The next table gives few examples of function Q for some well-known TFRs. Note its definition based on the derivative of phase law ϕ.

Table 1 Spreading functions for several TFrs (courtesy of authors of [6])

Distribution	Definition Spread factor
STFT	$STFT(t,\omega)=\int_{-\infty}^{\infty} w(\tau)x(t+\tau)e^{-j\omega\tau}d\tau$ $Q(t,\tau)=\phi^{(2)}(t)\frac{\tau^2}{2!}+\phi^{(3)}(t)\frac{\tau^3}{3!}+\phi^{(4)}(t)\frac{\tau^4}{4!}+\ldots$
Wigner distribution	$WD(t,\omega)=\int_{-\infty}^{\infty} w(\tau)x(t+\frac{\tau}{2})x^*(t-\frac{\tau}{2})e^{-j\omega\tau}d\tau$ $Q(t,\tau)=\phi^{(3)}(t)\frac{\tau^3}{2^2 3!}+\phi^{(5)}(t)\frac{\tau^5}{2^4 5!}+\ldots$
L-Wigner distribution	$LWD(t,\omega)=\int_{-\infty}^{\infty} w(\tau)x^L(t+\frac{\tau}{2L})x^{*L}(t-\frac{\tau}{2L})e^{-j\omega\tau}d\tau$ $Q(t,\tau)=\phi^{(3)}(t)\frac{\tau^3}{2^2 3!L^2}+\phi^{(5)}(t)\frac{\tau^5}{2^4 5!L^4}+\ldots$
Fourth order polynomial Wigner-Ville distribution	$PWD(t,\omega)=\int_{-\infty}^{\infty} w(\tau)x(t+0.675\tau)x^*(t-0.675\tau)$ $\times x^*(t+0.85\tau)x(t-0.85\tau)e^{-j\omega\tau}d\tau$ $Q(t,\tau)=-0.327\phi^{(5)}(t)\frac{\tau^5}{5!}-0.386\phi^{(7)}(t)\frac{\tau^7}{7!}+\ldots$

The *CrossTerms* in (2) stands for the cross-terms issued from the combination of the TFRs of each possible combination of components. The majority of works in time-frequency analysis attempts to reduce the level of these terms. In any case, the consequence of a such intention is the increasing of the importance of the function Q. Every method is subject of relative strong or weak assumptions. Very often, these assumptions transform the considered TFR in a parametric one restricting also its area of application (ie the types of signal that allow us to efficiently use the method). For this reason there is no a general method for efficient extraction of time-frequency information. This establishment is a critical issue if we cannot make any assumption. In our case, characterization of a process in a blind configuration restricts drastically the a priori hypothesis that can be made. The existing approach in this case is to test many TFRs trying to find the "best" one, but the choice of the criterion still remains a difficult task.

The next example illustrates the problems related to the choice of a best TFR in an uncertain configuration. In this sense, we consider two digital modulations TFRs that can be meet in a real scenario [7]. The ideal laws of modulation are given in the figure 1.a.

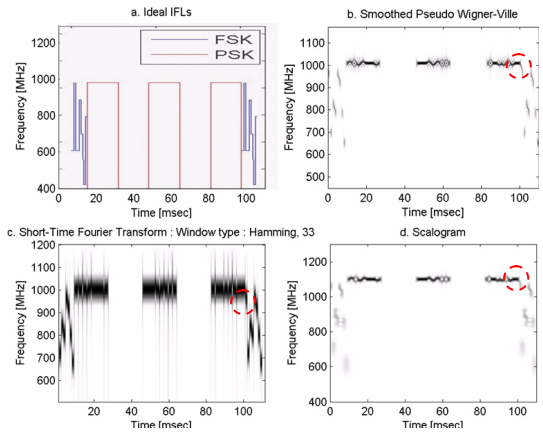


Fig. 1. Several TFRs of a mixed signal

For this mixture of two digital modulation types we show the results obtained by several well-known TFRs. In the case of Smoothed Pseudo Wigner-Ville Distribution (Fig.1.b) we remark the good resolution but the phase transitions corresponding to the PSK are erased by the smoothing. These transitions are better represented in the case of the Short-Time Fourier Transform – STFT (Fig. 1.c) but the paid price is the poor resolution. For this signal, a good result seems to be provided by the scalogram (Fig. 1.d). However, all the considered TFRs provide only qualitative information about the signal. Estimation of the parameters or separation of the modulations are the kind of tasks inappropriate for these TFRs. The main reason is the representation ambiguities generated by the trade-off between resolution and artifacts level. For example, it is almost impossible to separate the third PSK packet and the first frequency step of the second FSK packet (see Fig1. b, c, d circular marker).

As shown by the expression (2) the existing TFRs use only the information provided by the first-order derivative of the phase. For this reason, the interpretation of time-frequency representation is often subject to ambiguities. To illustrate this, let us consider two different signals

$$s_1[n]=e^{j2\pi\left[.1n-2\cdot10^{-4}n^2+10^{-6}n^3\right]}+e^{j2\pi\left[.2n-2\cdot10^{-4}n^2+10^{-6}n^3\right]};$$
$$s_2[n]=e^{j2\pi\left[\varphi+.1n-2\cdot10^{-4}n^2+10^{-6}n^3\right]}+e^{j2\pi\left[.2n-2\cdot10^{-4}n^2+10^{-6}n^3\right]} \quad (3)$$
$$\varphi=\pi/3 \text{ (for } n=0,..,256) \text{ and } \pi/2 \text{ (for } n=257,..,512)$$

whose effect in TFR domain is quite similar (Fig. 2).

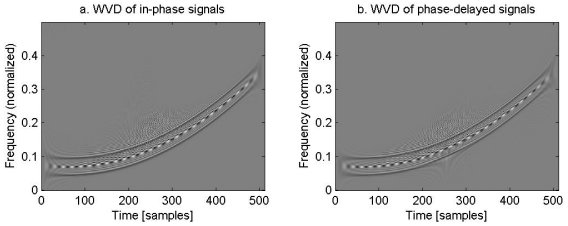


Fig. 2. Two different signals producing a similar T-F effect

Obviously, the both TFRs show that the WVD is able to follow the time-frequency content variation over time. But they fail to establish the nature of each signal. One can say that the signals are of the same type but it is false as the expression (3) states.

In conclusion, the standard TFRs cannot provide the complete information about the analyzed process. A solution to solve this problem is to take into account the phase. Its properties are discussed in the next section.

III. COMPLEMENTARITY OF TIME-FREQUENCY AND TIME-PHASE REPRESENTATIONS

The instantaneous phase is a fundamental parameter of any signal, because it describes its features. It is defined as the arctangent of the ratio between imaginary and real parts of the signal (or in-quadrate Q and in-phase I components).

$$\hat{\phi}(t) = IPL_s = \arctan\left(\frac{\mathrm{Im}\{s(t)\}}{\mathrm{Re}\{s(t)\}}\right) \quad (4)$$

The complete description of a signal provided by the phase is illustrated by the following example in the case of a cubic frequency modulation defined by

$$s[n] = e^{j\left(-3.3+.31543n-6.37\cdot10^{-4}n^2+3.14\cdot10^{-6}n^3\right)}; n = 0,..,511 \quad (5)$$

The WVD of this signal (Fig. 3.a) offers information about the IFL but it is affected by the "intra-ference" terms (inner-terms) as stated by the table 1. The exact IFL is depicted in Fig. 3.c and is obtained by the derivation of the IPL. Due to the derivation, the initial phase information is lost.

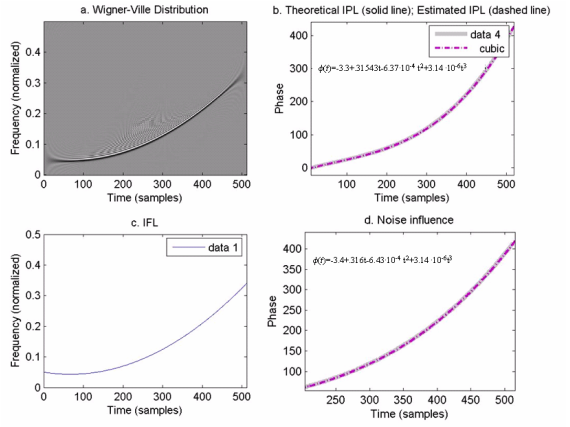


Fig. 3. Description of signal features by IPL

The IPL has been extracted by unwrapping the arctangent of Q/I ratio. As we can remark in the Fig. 3.b the polynomial phase estimation of this law provides all the signal parameters. Nevertheless, this procedure is not robust in noisy signals (Fig. 3.d) even if the noise is relatively low (20 dB). For this reason, the polynomial phase modeling approaches have been subject a great attention [4]. In spite of the variety of proposed methods, problems still remain especially in a multi-component signal case and when the polynomial model is not appropriate. This is the reason why we address the problem of T-F analysis in a non-parametric way (ie without any assumption about the signal type).

Example 3 shows that a signal is completely characterized by its IPL. Therefore, if the phase is *appropriately* used it can help us to eliminate the ambiguities in the T-F plane. Namely, the phase has to be carefully interpreted in the multi-component context. If we consider the signals (3), we remark, in Fig. 4.a, that the phases estimated via (4) contain all the information about the T-F behavior of the mixed signal but the useful information (ie the phase delay parameters) is "hidden". Fast transitions appearing in the first derivative of the phase (Fig. 4.a) are associated to the cross-terms and they mask the phase delay apparition.

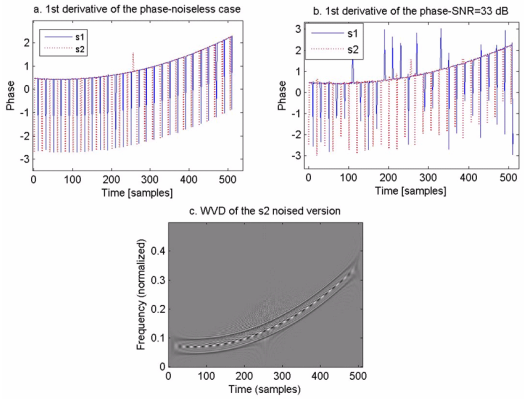


Fig. 4. Complementarity of time-frequency and time-phase information

However, since the cross-terms generate phase transitions having a regular variation, we can find the phase delay transitions by locking for irregular peaks. One might think that it is similar to the analysis of the cross-terms directly in the WVD plane (Fig. 2). Of course, the results would be similar but this procedure is very poor when the signal is embedded in noise. This is shown in Fig. 4.b : due to the noise it is almost impossible to distinguish between the cross-terms and the phase transitions (for a SNR about 33 dB !). Nevertheless, the TFR of this signal (Fig. 4.c) is almost untouched by the noise : the signal components are robustly represented.

Consequently, in order to find the phase information of a component, the proposed idea is to use the complementarity between the time-frequency and the time-phase representations. The concept based on this idea is addressed in the next section.

IV. TIME-FREQUENCY-PHASE ANALYZER

As figure 3 shows the instantaneous phase includes all the information about the corresponding signal. On the other hand, figure 4 shown that analysis of the phase using the operator (4) is limited in the case of multi-component signals and even for a relative reduced noise. Complementarily, the time-frequency information is well suited for analysis of the multi-component signals and it is less sensitive to the noise. Consequently, for the analysis of the instantaneous phase of each component being part of a signal, we propose the use of the time-frequency representation

and of a phase estimation operator. The choice of these items is explained as follows.

A. *Initial Time-Frequency Representation*

Given an unknown multi-component noised signals, the first natural operation is to represent the signal in an initial representation space (IRS). Since the unknown signal is generally non-stationary, non-parametrics TFRs are the naturally the best candidates. Among the existing TFRs, everybody could choice its representation. However, since the aim is to represent the signal component, artifacts should be avoided. For this reason, the WVD is not really well advised for IRS generation purpose. Alternatively, the spectrogram is a typical example belonging to the TFR class based on a time windowing procedure. However, since this kind of procedure introduces artificial phase transitions, we prefer to avoid such process. Therefore, we build the IRS only around a filterbank-based partition of the signal's spectrum. In order to highlight the signal details we propose a Gabor-type filterbank whose transfer functions are overlapped in frequency [8]. The effect of this filterbank-processing is mathematically expressed as :

$$\mathbf{W}s = \left\{ s * h_k \,\middle|\, k = 1,..,N_{filters} \right\} \qquad (6)$$
$$h_k(t) = IFFT\left\{ e^{-2\pi^2\sigma^2(f-f_k)^2} \right\}$$

where IFFT stands for the Inverse Fourier transform.

The choice of the spectral centers of the filters, f_k, ensures that the filterbank covers the spectral range of interest. The bandwidth of the transfer function, σ, and the number of filters, $N_{filters}$, are selected according to the required spectral range and the desired overlap.

The interest of the filterbank analysis-based concept is obvious. Since the analyzed signal has generally a complex T-F structure, its representation in several sub-bands leads to a reduction of the representation complexity. Analyzing a signal in a given sub-band and around its neighborhood allows us to identify signal structures much easier than searching in all T-F plan. As identification criterion, we can use the local energy criterion as proposed in [9], for example. The idea is to depict time-frequency structures whose energy is higher than a local threshold and that are composed by contiguous energy atoms. Hence, given two atoms s_1 and s_2, $s_1, s_2 \subset \mathbf{W}s$ having partially overlapped time-frequency supports, we generally associate them to the same component $\boldsymbol{c}$ if one, part of or all the following conditions are satisfied :

- $$\mathbf{E}s_1 > \varepsilon \wedge \mathbf{E}s_2 > \varepsilon \qquad (7)$$

where $\mathbf{E}$ stands for the energy and ε is a threshold computed locally ;

- $$\sqrt{(t_{s1}-t_{s2})^2 + (f_{s1}-f_{s2})^2} \le \gamma \qquad (8)$$

where (t_{si}, f_{si}) are the time and frequency centers of the signals s_i (i=1,2) and γ is a time-frequency distance threshold;

- $$\left| \frac{d\mathbf{E}s_1 \Theta s_2}{d\theta} \right| \le \eta \qquad (9)$$

where $\mathbf{E}s_1 \Theta s_2$ is the time-frequency energy comprised between s_1 and s_2, η is threshold measuring the degree of energy continuity. If the gradient of the energy between the both atoms is below a certain threshold, one can conclude that both atoms belong to the same component $\boldsymbol{c}$.

The use of these criteria allows associating the time-frequency atoms as suggested in the next figure.

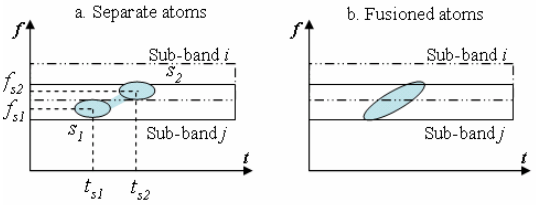


Fig. 5. Regrouping time-frequency atoms

The both atoms illustrated in Fig. 5.a could be merged with help of criteria (7)-(9) – Fig. 5.b. Another remark concerns the importance of the overlapping. If the filters have disjoint frequency supports the structures located at the border (s_2 in Fig. 5.a) would be incorrectly represented.

The choice of criteria (7)-(9) is a matter of time-frequency regrouping strategy which is selected according to the application (see [1], [2], [8], [9] for some existing methods).

Another reason for using the filterbank analysis is to segment the spectral domain such that noise, generally colored, can be split of in several white-like perturbations. This is important since the majority of the adaptive threshold setting procedures assume the noise is white in the considered spectral band. In our work we used the method presented in [9].

There are a large number of works combining criteria (7)-(9) and filterbank analysis. However, there are situations when the criteria (7)-(8) are limited. Such a case arises when the atoms are too far in time-frequency plane. Energy is between the atoms is spread out many sub-bands and they cannot be regrouped. This is illustrated for the signal used in the figure 1 (Fig. 6.a).

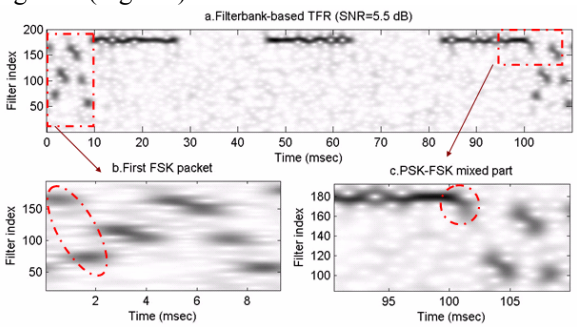


Fig. 6. Limitations of classical T-F regrouping criteria

As we observe in Fig. 6.b the two encircled atoms of the FSK are too far to be associated by the criteria (8) and (9). In the same time (Fig. 6.c) the last part of the PSK and the first atom of the FSK will be associated since they are very close.

This kind of situation will be solved by the phase evaluation in an appropriate way.

B. *Phase analysis operator*

We saw previously that the time-frequency IRS provides robust T-F information about the analysis process but it remains deficient when the components are too far (the case of the fast modulations) or too close (in this case, the finite resolution of the filterbank acts). A possible solution to eliminate the ambiguities is to apply the phase operator analysis (defined, for example, by (4) but other methods could be imagined [10]). As we explained in the section 3, the direct application of the operator (4) is not judicious because of either interferences either the noise. The idea we propose is to apply this operator **only** for the signal corresponding to the time-frequency region we are about to analyze it. In the same time, in order to avoid the noise and the artifacts as much as possible, it is necessary to define this region in an appropriate manner. Intuitively (see Fig. 7), the design of the region around the connection line of the atoms is more appropriate than the simple rectangular definition. In other words, with help of the IRS, we can define the T-F region according to the content of data inside it.

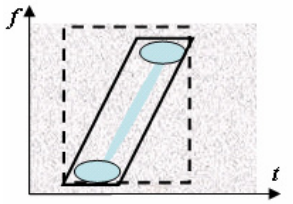


Fig. 7. Content-based definition of T-F region

This content-based design of T-F region **R** allows us to define the parameters of a time-frequency filter. The method we used is based on the non-unitary time-warping concept [11]. Namely, the filter is designed by analogy of the FIR methodology but for a time axis that follows the nature of the region. The result will be an impulse response h_w that matches very efficiently the features of the signal bounded by **R**. Applying this filter on the original signal s is equivalent with the extraction of the signal located inside this region :

$$s_{\mathbf{R}} = s * h_w \tag{10}$$

Since the ultimate goal of this operation is the estimation of the phase via (4), we impose that the filter h_w be able to conserve the phase feature of the original signal. An interesting idea is device in [11] where a forward-backward strategy is used in order to design filters with zero phase characteristic. That is, thanks to [11], we can extract correctly the signal corresponding to two or more detected atoms in the IRS. Let denote with $\phi_{\mathbf{R}}$ the phase of the signal $x_{\mathbf{R}}$. For this function we can define the *continuity* concept in the considered time-frequency region. Mathematically, this could be done with help of singularity concept. In our context, the continuity concept could be used in the following way. If the phase $\phi_{\mathbf{R}}$ contains one or many singularities it means that the phase of the signal $x_{\mathbf{R}}$ has fast variation. This is typical for atoms belonging to distinct components. Alternatively, if the phase is slowly varying we can conclude that the atoms belong to the same component. Mathematically, the concept of phase-based continuity test can be formulated as follows.

FOR all t

$FOR\ k = 1,..,N$

$IF \left| \frac{d^k \phi_{\mathbf{R}}}{dt^k}(t) \right| \geq \mu$

$IF\ \phi_{\mathbf{R}}(t-\Delta t) \neq \phi_{\mathbf{R}}(t+\Delta t) \Rightarrow$

$\Rightarrow\ t$ - separating point of two atoms

$ELSE \Rightarrow\ t$ - phase transition (eg PSK)

$ELSE \Rightarrow$ the atoms belong to a continue T-F structure

where μ is a threshold applied in the k^{th}-order derivative time-phase domain. Note that if the atoms are simple Gabor atoms ones (eg. PSK or FSK components) the first order derivative is sufficient to apply this continuity criterion.

Both time-frequency and time-phase analysis concepts form the time-frequency-phase (T-F-Ph) analyzer whose general structure is depicted in the next figure.

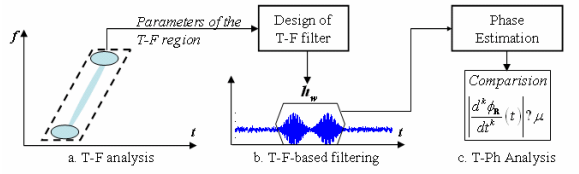


Fig. 8. The concept of time-frequency-phase analyzer

V. RESULTS

In this section, we illustrate how the time-frequency-phase analyzer performs and avoid the ambiguities existing in the case of a simple time-frequency analysis. For these purposes we consider the previous examples.

Let consider the two-component signal defined by (3). In order to take advantage on the complementarity of the time-frequency and time-phase representations, we apply, according to the fig. 8, a time-frequency filtering procedure as illustrated in the figure 9.b.

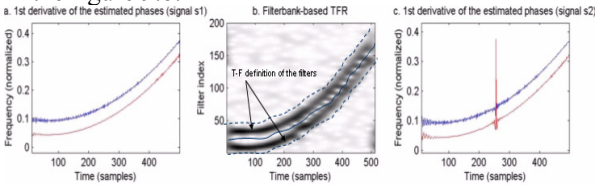
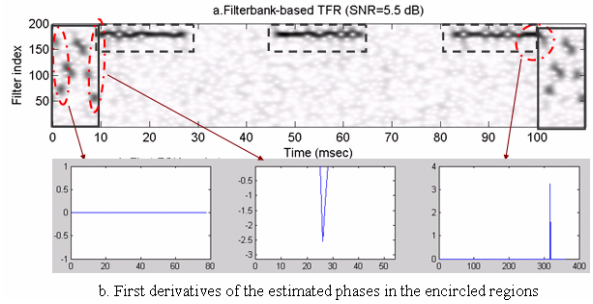


Fig. 9. T-F-Ph analysis of signals defined by (3)

The signals issued by filtering as shown in Fig. 9.b. are analyzed with the phase analysis operator (4). Since the both components are separated the phase is correctly estimated. Hence, we can clearly distinguish the differences between the both signals. In the same

time we obtain all the details about the phase of each component (Fig. 9.a, c) : the time-frequency shapes are correctly estimated as well as the phase transition (Fig. 9.c).

Another example consists in segmentation of the FSK-PSK signal defined in Fig. 1. As it is shown in the Fig. 10, using the T-F-Ph analyzer we can correctly separate the components of each modulation.



b. First derivatives of the estimated phases in the encircled regions

Fig. 10. Segmentation of a FSK-PSK composite signal

In order to prove the richness of the phase information we illustrate the phase of three "tricky" regions. The first consists of two FSK atoms; as we can remark, the 1st phase derivative is constant which means that both atoms belong to the same signal. In the second case, we consider the last atom of the FSK and the first PSK packet. In this case, the phase discontinuity indicates that both parts of signals do not belong to the same one. The same establishment for the third case : in spite of the proximity between components, the phase discontinuity shows unambiguously that there are two different signals. In the same time the classical TFRs (see Fig. 1) are not able to associate these atoms to the right component.

Applying the T-F-Ph analyzer we successfully segment the signal according to the modulation types. Furthermore, this benefit will allow us to estimate the features of each modulation very important task in the field of signal intelligence (SIGINT) [7]. Otherwise, the case illustrated in Fig. 10 can arise in active communication or radar field. Due to the parasite communication signals or artifacts, the pre-processing device of receiver could be subject of wrong detection. This error can be felt further, during the adaptive filtering process. In conclusion, in the domain of communications in non-cooperative and/or signal-sense high density environment, the T-F-Ph analyzer constitutes a potential solution.

VI. CONLUSION

The purpose of this paper was to show how the time-frequency and the time-phase informations could be used jointly in order to solve difficult situations, untrateable by the convetional time-frequency concepts. Conceptually, phase information will work towards avoiding the ambiguities by defining the notion of *continuity* of time-frequency structure. It can help us to solve situations when the conventional T-F analysis falls. For example, it allows us to connect components belonging to the same structure even if the TFR is affected by any type of artifacts (cross-terms, noise, attenuation, etc.). The joint use of conventional time-frequency representations and the phase of the signal constitutes the background of the time-frequency-phase analyzer.

Of course, the methods we used in this paper to implement the T-F-Ph analyzer are just sugestions. Everybody could use his own methods. Concerning our future works, two axes will be addressed. The first one deals with the generalized definition of the initial representation space. The second one consists in analyzing and proposing more efficient phase analysis operators, especially from noise robustness and multi-component point of view.

REFERENCES

[1] L. Cohen , *Time-Frequency Analysis*, Pretince Hall, New Jersey, 1993.

[2] A.Papandreou-Suppappola, ed., *Applications in time-frequency signal processing,* CRC Press, Boca Raton, 2003.

[3] Y. Grenier, *Parametric Time-Frequency Representations,* in Signal Processing, eds. J.L. Lacoume, Les Houches, Session XLV, Amsterdam, vol. 1, pp. 339-397, 1987.

[4] C. Ioana, A. Quinquis, *Time-Frequency Analysis using Warped-Based High-Order Phase Modeling*, EURASIP Journal of Applied Signal Processing, 2005(17):2856-2873, Sept. 2005.

[5] F. Léonard, *Phase spetrogram and frequency spectrogram,* Traitement du Signal, vol. 17, n°4, pp. 269-286, 2000.

[6] S. Stankovic, L. Stankovic, *Introducing time-frequency distributions with a complex time arguments*, Electronic Letters, vol. 32, No. 14, pp. 1265-1267, July 1996.

[7] P.E. Pace, *Detection and Classifying low Probability of Intercept Radar,* Artech House, Norwood, 2004.

[8] A. Teolis, *Computationnal Signal Processing with Wavelets*, Birkhäuser, Boston, 1998.

[9]] C. Cornu et *all*, *Time-frequency detection using Gabor filter banks and Viterbi based grouping algorithm*, Proceedings of IEEE International Conference on Acoustic, Speech and Signal Processing ICASSP 2005,Vol 4, pp. 499-500, Philadelphia, USA.

[10] C. Ioana *et all*, *Analysis Of Time-Frequency Transient Components Using Phase Chirping Operator*, IEEE International Conference on Acoustic, Speech and Signal Processing ICASSP 2006, Toulouse.

[11] A. Jarrot, C. Ioana, A. Quinquis, *A class of time-frequency filters based on non-unitary time-warping operators,* Paper submitted to IEEE Transaction on Signal Processing, March 2006.

Buletinul Ştiinţific al Universităţii "Politehnica" din Timişoara

Seria ELECTRONICĂ şi TELECOMUNICAŢII
TRANSACTIONS on ELECTRONICS and COMMUNICATIONS

Tom 51(65), Fascicola 2, 2006

The estimation of the instantaneous frequency using time-frequency methods

Marius Salagean[1], Ioan Nafornita[2]

Abstract – Instantaneous frequency (IF) is a very important parameter in a large number of applications. Generally, the IF is a non-linear function of time. For such cases the analysis of time-frequency content provides an efficient solution. In this paper is analyzed the performance in IF estimation of the two time-frequency based methods. The first estimation method uses the complex argument distribution (CTD) and the second one uses the ridges extraction method from the time-frequency distribution based on mathematical morphology operators (TF-MO). Monocomponent signals with non-linear and highly non-linear IF corrupted by Gaussian white noise are considered as numerical examples.

Keywords: Instantaneous frequency, time-frequency distribution, complex argument, mathematical morphology, signal analysis, image analysis.

I. INTRODUCTION

In signal processing the decision (detection, denoising, estimation, recognition or classification) is a basic problem. Knowing that the real environments are generally highly non-stationary, it is necessary to use a method able to provide suggestive information about the signal structure. A potential solution is based on time-frequency representations that provide a good concentration around the law of the IF and realize a diffusion of the perturbation noise in the time-frequency plane.

The CTD has been introduced in [1] as an efficient way to produce almost completely concentrated representations along the IF, it considerably reduces the artifacts due to the complexity of the analyzed signal.

The TF-MO estimation method [2] is based on the conjoint use of two very modern theories, that of time-frequency distributions and that of mathematical morphology. This strategy permits the enhancement of the set of signal processing methods with the aid of some methods developed in the context of image processing.

The paper is organized as follow. In section II is presented the CTD. The TF-MO method is illustrated in section III. In section IV some simulation results are depicted. Section V will close this communication.

II. COMPLEX TIME DISTRIBUTION

The complex time distribution as an IF estimator have been proposed and analyzed in [1]. It provides a highly concentrated distribution along the IF law with reduced interferences (noise and cross-terms).

Mathematically, the complex time distribution (CTD) of signal is defined as:

$$CTD(t,\omega)=\frac{1}{2\pi}\int_{-\infty}^{\infty} s\left(t+\frac{\tau}{4}\right)s^{*}\left(t-\frac{\tau}{4}\right) \times s^{-j}\left(t+j\frac{\tau}{4}\right)s^{j}\left(t-j\frac{\tau}{4}\right)\,e^{-j\omega\tau}\,d\tau \tag{1}$$

where the continuous form of the "complex-time signal" is:

$$s(t+j\tau)=\frac{1}{2\pi}\int_{-\infty}^{\infty} S(j\omega)\,e^{-\omega\tau}\,e^{j\omega t}\,d\omega \tag{2}$$

where $S(j\omega)$ is the Fourier transform of signal s.

The main property of the CTD consists in the capability to attenuate the high-order terms of the polynomial decomposition of the IF, for signals of the form $s(t)=r\,e^{j\phi(t)}$. The spread factor $Q(t,\tau)$ around the IF for this distribution is:

$$Q(t,\tau)=\phi^{(5)}(t)\frac{\tau^{5}}{4^{4}5!}+\phi^{(9)}(t)\frac{\tau^{9}}{4^{8}9!}+\ldots \tag{3}$$

[1] Facultatea de Electronică şi Telecomunicaţii, Departamentul Comunicaţii Bd. V. Pârvan Nr. 2, 300223 Timişoara, e-mail marius.salagean@etc.upt.ro
[2] Facultatea de Electronică şi Telecomunicaţii, Departamentul Comunicaţii Bd. V. Pârvan Nr. 2, 300223 Timişoara, e-mail ioan.nafornita@etc.upt.ro

As proved in [4], in the case of the CTD, this function has a fifth order dominant term (for comparison, the spectrogram and the Wigner-Ville distribution (WVD) have a second and a third order dominant term, respectively), which corresponds to a drastic reduction of the higher terms of $Q(t,\tau)$.

The CTD satisfies some important properties:

1) the CTD is real for the frequency modulated signals $s(t)=re^{j\phi(t)}$;

2) the time-marginal property $\frac{1}{2\pi}\int_{-\infty}^{\infty} CTD(t,\omega)d\omega=|s(t)|^2$;

3) the unbiased energy condition $\frac{1}{2\pi}\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} CTD(t,\omega)\,dt\,d\omega=\int_{-\infty}^{\infty}|s(t)|^2\,dt=E_s$;

4) the frequency-marginal property;

5) time-frequency shift-invariance properties;

6) the CTD of the scaled signal $\sqrt{|a|}\,s(at)$ is $CTD(at,\omega/a)$.

III. TIME-FREQUENCY MORPHOLOGICAL OPERATORS METHOD

The estimation method based on time-frequency and image processing techniques has been introduced in [2]. The quality of estimating the IF depends on the time-frequency distribution and on its ridges projection mechanism. The TF-MO method proposes a time-frequency representation based on cooperation of linear and bilinear distributions: the Gabor and the Wigner-Ville distributions. It is known [3] that the Gabor representation has a good localization and free interference terms properties. Unfortunately, the linear distributions, except the Discrete Wavelet transform, correlate the zero mean white input noise, as shown in [4]. The WVD is a spectral-temporal density of energy that does not correlates the input noise, thus having a spreading effect of the noise power in the time-frequency plane [2]. The WVD has also a good time-frequency concentration.

To combine these useful advantages, the time-frequency distribution is calculated according to the following algorithm [2]:

1) Calculate the Gabor transform for the signal *s*, $G(t,\omega)$.

2) Filter the image obtained with a hard-thresholding filter:

$$Y(t,\omega)=\begin{cases}1, & if\ |G(t,\omega)|\ge tr\\ 0, & if\ |G(t,\omega)|<tr\end{cases} \tag{4}$$

where *tr* is the threshold used.

3) Calculate the WVD for the signal *s*, $WV(t,\omega)$.

4) Multiply the modulus of the $Y(t,\omega)$ distribution with the $WV(t,\omega)$ distribution.

In step 2) the proposed threshold value is:

$$tr=\frac{\max\limits_{(t,\omega)}\{|G(t,\omega)|\}}{5} \tag{5}$$

This operation decreases the amount of noise that perturbs the ridges of $G(t,\omega)$ and brings to zero the values in the rest of the time-frequency plane. The effect of the multiplication in step 4) is the reduction of the interference terms of the WVD and the very good localization of the ridges of the resulting ditribution.

To estimate the ridges of the obtained distribution, some mathematical morphology operators are used, the above resulting distribution being regarded as an image. This mechanism is applied through the following steps [2]:

1) Convert the image obtained in step 4) in the procedure described earlier in binary form.

2) Apply the dilation operator on the image in 1).

3) Skeletonization of the last image, an estimation of the IF of the signal being obtained. This image represents the result of the TF-MO method. The conversion in binary form realizes a denoising of the time-frequency distribution. The role of the dilation operator is to compensate the connectivity loss, produced by the preceding conversion. The skeleton produces the ridges estimation.

IV. RESULTS

Example 1: Consider a noisy monocomponent signal with non-linear IF:

$$s(t)=\exp\{j(5\pi t^3-9.5\pi t)\}+n(t) \tag{6}$$

within the interval [-1, 1] where *n(t)* is a Gaussian white noise.

Based on the CTD and TF-MO method, the IF is estimated for various values of the SNR. In Fig. 1.a and Fig. 1.b, are presented the estimated IF with the two methods, for SNR=30dB (left image) and SNR=3dB (right image) along with the real IF law.

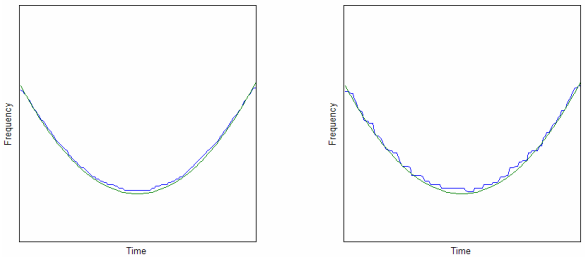


Fig. 1.a. IF estimation based on TF-MO method for SNR=30dB (left image) and SNR=3dB (right image)

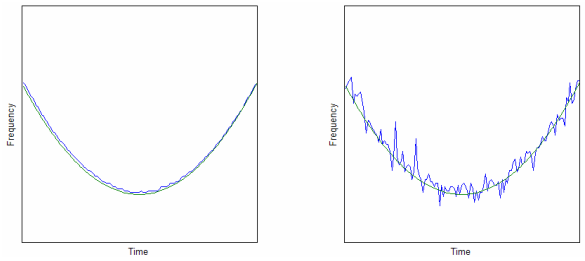


Fig. 1.b. IF estimation based on CTD method for SNR=30dB (left image) and SNR=3dB (right image)

From Fig. 1.b can be also observed that for low SNR (right image) the variance in the CTD is higher thus degrading the performance of the estimation.
Mean squared errors of the IF estimation calculated in 128 realizations for SNR values within the interval [3dB, 30dB], based on CTD, TF-MO method and WVD are showed in Fig. 2.a and 2.b (zoomed image).

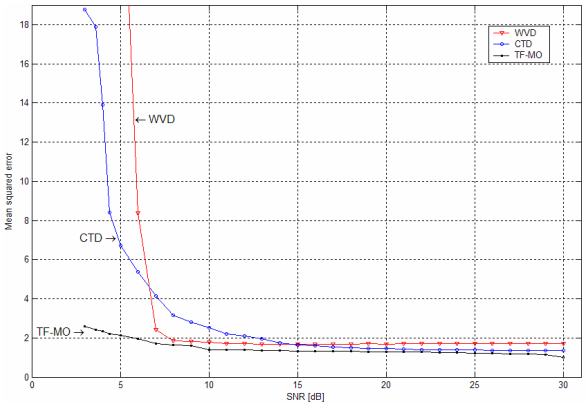


Fig. 2.a. Mean squared error of the IF estimation for SNR between [3dB, 30dB] based on CTD , TF-MO method and WVD

It can be noticed that the performances for the three distributions in the SNR range [10dB, 30dB], are slightly similars, nevertheless the TF-MO method providing a better results. As the SNR decreases, the estimation error in the CTD and WVD increases more rapidly than in the TF-MO method.

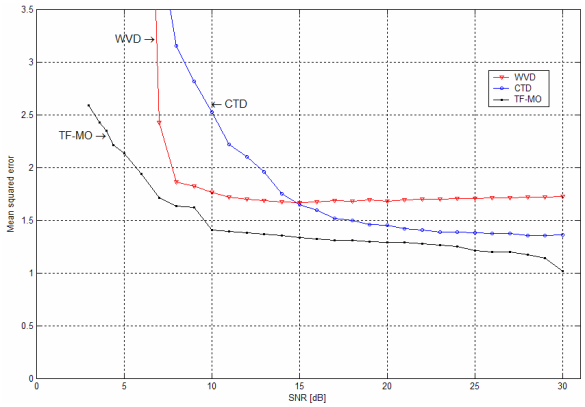


Fig. 2. b. Mean squared error of the IF estimation based on CTD , TF-MO method and WVD (zoomed image)

Example 2: Consider now a noisy monocomponent signal with highly non-linear IF:

$$s(t) = \exp\{j(3\cos(\pi t) - \cos(3\pi t)/2 + \\ + \cos(5\pi t)/1.5))\} + n(t) \qquad (7)$$

within the interval [-1, 1] where *n(t)* is a Gaussian white noise.
The IF is estimated for various values of the SNR, based on the CTD and TF-MO method. Fig. 3.a and Fig. 3.b, represent the estimated IF with the two methods, for SNR=30dB (left image) and SNR=3dB (right image) along with the real IF law.

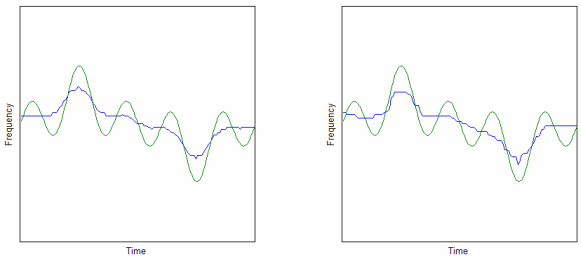


Fig. 3.a. IF estimation based on TF-MO method for SNR=30dB (left image) and SNR=3dB (right image)

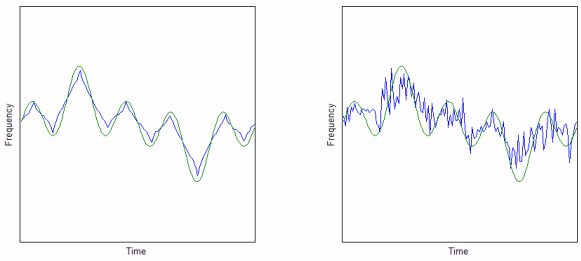


Fig. 3.b. IF estimation based on CTD method for SNR=30dB (left image) and SNR=3dB (right image)

Mean squared errors of the IF estimation calculated in 128 realizations for SNR values within the interval [3dB, 30dB], based on CTD, TF-MO method and WVD are showed in Fig. 4.

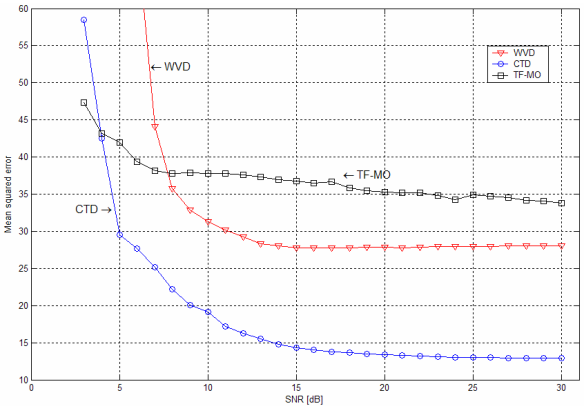


Fig. 4. Mean squared error of the IF estimation for SNR between [3dB, 30dB] based on CTD , TF-MO method and WVD

Fig. 3.a illustrates the fact that the bias in the TF-MO method is significant and dominates in the estimation error. The IF estimator in this case cannot accurately follow the rapid transitions in the IF. When the noise is increased, the variance in the CTD is higher (Fig. 3.b) thus reducing the quality of the estimation, whereas the variance and the bias in the TF-MO method remains slightly unchanged, which assures almost the same estimation error. Fig. 4 proves that behavior.

From these numerical examples, it can be noticed that the TF-MO method is very dependent on the choice of the threshold *tr*. A high value can preserve the noise peaks in the time-frequency plane outside the region where the signal component is located. This side effect is very significant for relatively high noise, thus degrading the estimation process. The performance can be improved by applying the morphological operators only in the region around the signal component. This can be done using a detection technique. Moreover, the parameters of the morphological operators have an important role for the ridges extraction precision and it has been observed that for high SNR skipping the application of the dilation operator in the TF-MO method provides an improvement of the IF estimation.

A low value for the *tr* can cause connectivity breaks of the time-frequency distribution used, this inconvenient being compensated by the reconstruction capacity of the morphological operators. This can be seen from Fig. 3 and Fig. 4, where for relatively high noise the performances of the TF-MO method are remaining acceptable. However, a greater connectivity breaks can induce false IF estimation.

From the analysis of the two examples considered it could be concluded that for signal with swift transitions over a short duration of time and SNR high enough, the accuracy of the IF estimation in the TF-MO method is poorer than in the CTD case. Still, for low SNR the performances become relatively equals, the TF-MO method being robust at the noise influence.

For signals not so complicated, the performances of the methods analyzed are similar, but the TF-MO method provides the better results.

V. CONCLUSIONS

In this paper, it has been analyzed the performances of the IF estimation for the CTD distribution and TF-MO method, in two illustrative numerical examples.

For monocomponent signals with highly non-linear IF and high SNR, the CTD distribution is more adapted than the TF-MO method. It is possible to improve the IF estimation results combining the qualities of the two methods. An immediate solution is to apply the mechanism of the ridges projection of the TF-MO method on the image obtained in the CTD distribution. Another possibility is to analyze the benefit of the reallocation principle in time-frequency domain, [5].

For signals with a time-frequency structure not so complicated, the performances of the distributions studied in this paper are relatively the same.

Further research could be directed toward the estimation of the polinomyal order of the estimated IF.

Other possible further research directions are the following:

- the comparison of the two IF estimation methods for multicomponent signals;
- the conception of a new IF estimation method combining the qualities of the two methods analyzed in this paper;
- the simulation of a system for time-frequency signal analysis, starting from the model proposed in [6].

REFERENCES

[1] LJ. Stankovic, "*Time-frequency distributions with complex argument*", IEEE Tran. On Signal Processing, vol. 50, no.3, pp. 475-486, March 2002.

[2] Monica Borda, Ioan Nafornita, Dorina Isar, Alexandru Isar, "*New instantaneous frequency estimation method based on image processing techniques*", Journal of Electronic Imaging, Vol. 14, Issue2, 023013_1-023013_11, April-June 2005.

[3] S. Stankovic, L. Stankovic, "*Introducing time-frequency distributions with a complex time arguments*", Electronic Letters, vol. 32, No. 14, pp. 1265-1267, July 1996.

[4] A. Isar, D. Isar, M. Bianu, "*Statistical Analysis of Two Classes of Time-Frequency Representations*", Facta Universitatis, series Electronics and Energetic, vol. 16, no.1, April 2003, Nis, Serbia, 115-134.

[5] F. Auger, P. Flandrin, "*Improving the readability of time frequency and time scale representations by reassignment method*", IEEE Trans. Signal Process. 43, May 1995, 1068-1089.

[6] S. Stankovic, L. Stankovic, "*An architecture for the realization of a system for time-frequency signal analysis*", IEEE Transactions on Circuits and Systems, Part II, No. 7, July 1997, 600-604.

[7] M. Salagean, M. Bianu, C. Gordan "*Instantaneous frequency and its determination*", UPT Scientific Bulletin. Vol. 48 (62), Electronics and Communications, Nr. 1-2, 2003.

Buletinul Ştiinţific al Universităţii "Politehnica" din Timişoara

Seria ELECTRONICĂ şi TELECOMUNICAŢII
TRANSACTIONS on ELECTRONICS and COMMUNICATIONS

Tom 51(65), Fascicola 2, 2006

The Minimum Likelihood APP Based Early Stopping Criterion for Multi-Binary Turbo Codes

Horia Balta[1], Catherine Douillard[2], Maria Kovaci[1]

Abstract – This paper presents a simple and efficient criterion for stopping the iteration process in multi-binary symbol turbo-decoding with a negligible degradation of the error performance. The criterion is devised starting to minimum log-likelihood ratio (LLR) based stopping criterion used for binary turbo codes (BTC). Two variants consist in particularizations of the same idea in the MAP and MaxLogMAP decoding algorithm cases. The proposed two variants criterion has efficiency close to the optimum (genie) criterion and is simple to perform.

Keywords: Iterative decoding, stopping criterion, multi-binary turbo codes.

I. INTRODUCTION

A classical turbo-code [1] calls for a parallel concatenation of two single-binary recursive, systematic convolutional codes (RSC) (with 1/2 coding rate). In the decoding process, the corresponding constituent decoders exchange extrinsic information through an iterative process.
With each iteration in the turbo decoding, the signal to noise ratio (SNR) required to obtain a specified bit-error rate (BER) decreases [1]. But the improvement in SNR becomes smaller with each iteration.
For the SNR of practical interest, after a limited number of iterations (3 or 6), the turbo decoder corrects the received block and is able, through hard decision, to retrieve the transmitted original data sequence. Only for a small proportion of the received blocks, the turbo decoder must perform a greater iteration number (8 or 15) to manage the total correction or in big proportion of these blocks. But this computational effort will be reflected by a consistent diminution of the BER.
Thus, it becomes rightful to perform a different iteration number for each received block in part, number which will be established by an early stopping iterations criterion. By the diminution of the computational effort, the using of such stopping iterations criterion will bring also others advantages: the decrease of the decoding average time (in the case of buffers use), the increasing possibility of the maximum number of the imposed iterations, the decrease in the used power in the decoding.
But, on the other side, the stop criterion must not alter the BER performance obtained through the realization of all iterations. The utility of such iteration stopping criterion, which constitutes an optimal compromise between the two constraints (the elimination of the unnecessary iterations and the conservation of the BER performance) is proved by the considerable number of publications on this theme, for instance [2] [3] [4] [5].
These publications apply to BTC. Owing to the decoding of the specific Multi-Binary Turbo Codes (MBTCs), [6], the stop criteria, built for BTC, have not the same efficiency for MBTCs, or they can even not be applied to MBTCs.
After an extensive review of existing stopping criteria, this paper presents a new stopping criterion, in two variants, applicable in the case of the symbol decoding [7] used in the MBTCs case. The two variants of the stopping criterion represent its adaptation to the MAP and, respective, Max Log MAP decoding algorithms.

II. TURBO DECODING AND EARLY STOPPING CRITERIA

Through this paragraph, after a short presentation of the used notations, we review the existing early stopping criteria.

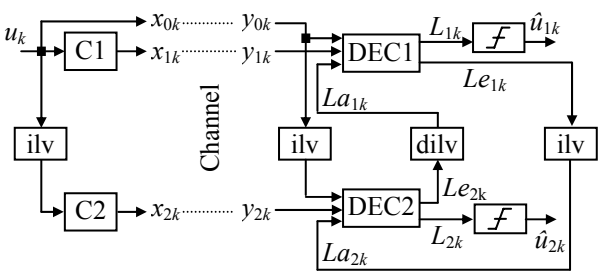


Fig. 1 A Turbo-Code scheme.

[1] Department of Communications, Faculty of Electronics and Telecommunications, Bd. V. Pârvan Nr. 2, 300223 Timişoara, e-mail: balta@etc.upt.ro, kmaria@etc.upt.ro
[2] Electronic Department,ENST Bretagne - Technopôle de Brest Iroise CS 83 818 - 29238 BREST Cedex 3, France, e-mail: Catherine.Douillard@enst-bretagne.fr

For the beginning, we consider the BTC that consists of two rate-1/2 recursive and systematic convolutional codes (RSC), shown in Fig.1. Let $\boldsymbol{u} = (u_1, u_2, \ldots u_N)$ be an information block of length N and $\boldsymbol{x} = (\boldsymbol{x_1}, \boldsymbol{x_2}, \ldots \boldsymbol{x_N})$ be the corresponding coded sequence, where $\boldsymbol{x_k} = (x_{0k}, x_{1k}, x_{2k})$, for $k \in I = \{1, 2, \ldots N\}$ is the output code block at time k. Assuming BPSK transmission over an AWGN channel, u_2 and x_{jk} all take values in $\{-1, +1\}$ for $k \in I$ and $0 \le j \le 2$. Let $\boldsymbol{y} = (\boldsymbol{y_1}, \boldsymbol{y_2}, \ldots \boldsymbol{y_N})$ be the received sequence, where $\boldsymbol{y_k} = (y_{0k}, y_{1k}, y_{2k})$ is the received block at time k. Then $y_{jk} = x_{jk} + w_{jk}$, where w_{jk} is a Gaussian random variable with zero mean and variance σ^2. Like it is shown in Fig.1, $\hat{\boldsymbol{u}}_{\boldsymbol{n}} = (\hat{u}_{n1}, \hat{u}_{n2}, \ldots \hat{u}_{nN})$, with n =1 or 2, we may denote the estimate of $\boldsymbol{u}$ given by DEC1 and DEC2 respectively. We also note by La_{nk}^i, Le_{nk}^i and L_{nk}^i the a priori information, the extrinsic one and the log-likelihood ratio (LLR), respectively, from decoder n for the u_k bit at ith iteration.

Mainly, an early stopping rule consists in a comparison of a measure, calculated after each iteration, with a threshold μ. In the following, we briefly present some of the existing main stopping criterions.

(1) Cross Entropy (CE) [2]. Based on the assumptions presented in [3] the CE between the distribution of the estimates at the outputs of the decoders at iteration i can be approximated by:

$$T(i) \approx \sum_{k=1}^{N} \frac{\left(Le_{2k}^i - Le_{2k}^{i-1}\right)^2}{e^{\left|L_{1k}^i\right|}} \qquad (1)$$

The decoding process is stopped after iteration i for $i \ge 2$, if:

$$\frac{T(i)}{T(1)} < \mu\,, \qquad (2)$$

where $T(1)$ is the approximated CE after the first iteration and the threshold μ is around 10^{-3}.

(2) Sign-Change Ratio (SCR) [3]. In the SCR criterion the decoding will be stopped at the ith iteration if the ratio $C(i)/N$ is lower than μ where $C(i)$ is the number of the sign differences between Le_{2k}^{i-1} and Le_{2k}^i. The threshold μ can takes values between 0.005 and 0.01 as a function of SNR and N.

(3) Sign-Difference Ratio (SDR) [4]. The SDR criterion is a replica of SCR in which $C(i)$ is calculated as the number of sign differences between La_{2k}^i and Le_{2k}^i.

(4) Hard Decision-Aided (HDA) [3]. Proceeding from the HDA criterion, the decoding process is stopped after iteration i for $i \ge 2$, if:

$$sign\left(L_{2k}^i\right) = sign\left(L_{2k}^{i-1}\right) \quad \forall k \in I. \qquad (3)$$

(5) Improved Hard Decision-Aided (IHDA) [8]. According to IHDA, at iteration i, we compare the hard decisions of the information bit based on $L_{2k}^i - Le_{2k}^i$ with the hard decision based on L_{2k}^i. If they agree with each other for the entire block, the decoding process is terminated at iteration i.

(6) Mean Estimate (ME) [5]. After each iteration i the mean $M(i)$ of the absolute values of the LLRs is calculated, and the decoding process is stopped if:

$$M(i) = \frac{1}{N} \cdot \sum_{k=1}^{N} \left|L_{2k}^i\right| > \mu\,. \qquad (4)$$

(7) Minimum LLR (mLLR) [9]. The mLLR rule stops the decoding process after iteration i for $i \ge 1$, if:

$$\min_{1 \le k \le N} \left|L_{2k}^i\right| < \mu \qquad (5)$$

(8) Sum-Reliability (SR) [10]. After each iteration i the sum of the absolute values of the LLRs is calculated:

$$S(i) = \sum_{k=1}^{N} \left|L_{2k}^i\right| \qquad (6)$$

and the decoding process is stopped after iteration i for $i \ge 2$, if $S(i) \le S(i\text{-}1)$.

(9) CRC Rule (CRC). A separate error-detection code, especially a CRC, can be concatenated as an outer code with an inner Turbo Code in order to flag erroneous decoded sequences. The decoding process is stopped after iteration i whenever the syndrome of the CRC is zero.

(10) Genie (GENIE). The GENIE (optimum) criterion can be used in the case where the original information bits are known. Then, the iteration is stopped immediately after the frame is correctly decoded. This unrealizable criterion gives a limit for all possible criteria.

Other criteria based on the presented ones have also been developed. In [11] a two decision threshold and a measure are introduced to enhance performance and simplify the stopping (CE) criterion. Combinations of these techniques are used too. In [5] two such combined rules (ME combined with HDA give the MSC criterion which is further combined with CRC rule) are presented. Another combination, between SR and mLLR stopping rules is done in [10]. An interesting idea is presented in [12]. The proposed (bit level) criterion stops the decoding process just for same bits (which are satisfying the rule). The iterations are performed only for the remaining bits, thus decreasing the computation volume. The simulations show that the bit level stopping is particularly effective for fading channels. Another useful idea, presented in [13], consists of taking into account and stopping also the non-convergent frames.

All of the presented stopping criterions were developed for binary turbo-codes. Considering the MBTCs with symbol decoding [6], some of these rules are not applicable (or not in that form). This fact is due to the disappearance of the LLR and their substitution by a posteriori probability (APP).

Note: When symbol decoding is performed, the decoder can compute Log-APPs or, in practice, normalized Log-APPs that can be considered as LLRs (this is the case in hardware decoders).

Moreover, we can not calculate the CE in the form (1). Also, because the La_{nk} and Le_{nk} constitute the pure probabilities, we can not speak about their sign. So, only the CRC stopping rule can be performed for MBTCs without any changes.

Remark: We say that the above stopping criteria can't be directly applied to MBTC, but they can probably be modified in order to cope with these codes.

With a small change the HDA and the derived IHDA could be adapted for MBTC. It can be made in two ways. The differences between symbols could be encountered without any other modification in the decoding structure. This means that the maximum likelihood (ML) sequences after two consecutives iterations will be compared. Some of these criteria are presented in [14] and [15]. But, as an alternative, the symbols could be decomposed in bits, and the HDA or its derived IHDA could be applied in original form over these bits.

In this paper we present an early stopping criterion, in two variants adapted for MAP and MaxLogMAP MBTC's algorithms, which is an adaptation of the mLLR rule for the MBTC case.

III. THE PROPOSED EARLY STOPPING CRITERION

Considering that the turbo code depicted in Fig.1 is multi-binary, as we mentioned above, we must replace the LLR by APP or Log-APP. With this only change in meaning we can keep all the notations, but we must add one more index for the input number. So, let R be the inputs number and r the index of these inputs. Consequently we have $\boldsymbol{u} = (\boldsymbol{u}_1, \boldsymbol{u}_2, \ldots \boldsymbol{u}_N)$ the symbols information block with $\boldsymbol{u_k} = (u_{1k}, u_{2k}, \ldots u_{Rk})$; $\boldsymbol{x_j} = (\boldsymbol{x_{j1}}, \boldsymbol{x_{j\,2}}, \ldots \boldsymbol{x_{j\,N}})$ the coded sequences, with $\boldsymbol{x_{jk}} = (x_{1jk}, x_{2jk}, \ldots x_{Rjk})$ and $x_{rjk} = \pm 1$; $\boldsymbol{y_j} = (\boldsymbol{y_{j1}}, \boldsymbol{y_{j2}}, \ldots \boldsymbol{y_{jN}})$ the received sequences, with $\boldsymbol{y_{jk}} = (y_{1jk}, y_{2jk}, \ldots y_{Rjk})$ and $y_{rjk} = x_{rjk} + w_{rjk}$, where w_{rjk} is a Gaussian random variable with zero mean and variance σ^2; $\hat{\boldsymbol{u}}_{\boldsymbol{n}} = (\hat{\boldsymbol{u}}_{\boldsymbol{n1}}, \hat{\boldsymbol{u}}_{\boldsymbol{n2}}, \ldots \hat{\boldsymbol{u}}_{\boldsymbol{nN}})$, with $\hat{\boldsymbol{u}}_{\boldsymbol{nk}} = (\hat{u}_{1nk}, \hat{u}_{2nk}, \ldots \hat{u}_{Rnk})$ and $n = 1$ or 2, denotes the estimate of $\boldsymbol{u}$ given by DEC1 and DEC2 respectively. For simplicity, we shall consider that $\boldsymbol{u_k}$ and $\hat{\boldsymbol{u}}_{\boldsymbol{nk}}$ are integers from the $J = \{0, 1, \ldots (2^R\text{-}1)\}$ set. Thus, in the MAP decoding algorithm, $La^i_{nk}(d)$, $Le^i_{nk}(d)$ and $L^i_{nk}(d)$ represent the a priori, the extrinsic and the a posteriori (APP) probabilities that the n decoder estimates (after ith iteration) the original u_k symbol at d integer, i.e. the above probabilities that $\hat{\boldsymbol{u}}_{\boldsymbol{nk}} = d \in J$. We also note that an information block contains N symbols and $R \times N$ bits.

Considering the decoding of a convergent block, as the iterative process advances, the APP probabilities corresponding to the original symbols sequence take values close to 1. Because of that, any other APP probabilities, i.e. $L^i_{nk}(d)$ with $d \neq u_k$, will take values close to 0. We use this fact to construct an early stopping criterion:

The iterative decoding process is stopped at iteration i if, at any time k, an APP probability value is greater than an imposed threshold μ:

$$\begin{gathered} \textit{stop iterations if } \forall k \in I,\ \exists d \in J \textit{ so that} \\ \mu < L^i_{2k}(d) < 1. \end{gathered} \tag{7}$$

We call this criterion the minimum likelihood APP (mlAPP). However, the mlAPP rule could not be used in this form for the case of the MaxLogMAP decoding algorithm. This is because the transfer of the APP values in the log domain.

For the MaxLogMAP case we consider the following judgement. If $d = \boldsymbol{u_k}$ than $L^i_{2k}(d) \approx -log(\mathrm{P}\{\hat{\boldsymbol{u}}_{\boldsymbol{nk}} = \boldsymbol{u_k}\} \approx 1)$ so $L^i_{2k}(d) \to 0$ when i is increasing. If $d \neq \boldsymbol{u_k}$, then $L^i_{2k}(d) \approx -log(a\ probability\ that\ is\ very\ small)$ so $L^i_{2k}(d) >> 0$. (In the previous judgement we suppose that in MaxLogMAP decoding algorithm normalization has been performed between the APP values corresponding to the same k time, i.e. from all APP values has been subtracted the minimum one.)

Thus, the mlAPP stopping criterion for the MaxLogMAP algorithm case is:

The iterative decoding process is stopped at iteration i if, at any time k, all the APP probabilities, except the zeros one, are bigger than an imposed threshold μ:

$$\begin{gathered} \textit{stop iterations if } \forall k \in I \textit{ and } \forall\ L^i_{2k}(d) > 0 \\ \textit{result that } L^i_{2k}(d) > \mu\,. \end{gathered} \tag{8}$$

Or in another formulation:

$$\textit{stop iterations if } \min_{d \in J^*} \left\{ L^i_{2k}(d) \right\} > \mu\,,\ \forall k \in I \tag{8'}$$

where $J^* = J \setminus \{d^*\}$ and d^* is the value of the k symbol for which $L^i_{2k}(d^*) = 0$.

IV. SIMULATIONS RESULT

We have performed simulations using the 8 state double-binary turbo code and the interleaving defined in [6], with data block length $N = 752$ double-binary symbols, that is = 1504 bit data blocks. We assumed a transmission over an AWGN channel, using QPSK modulation and no quantization is performed at the decoder input.

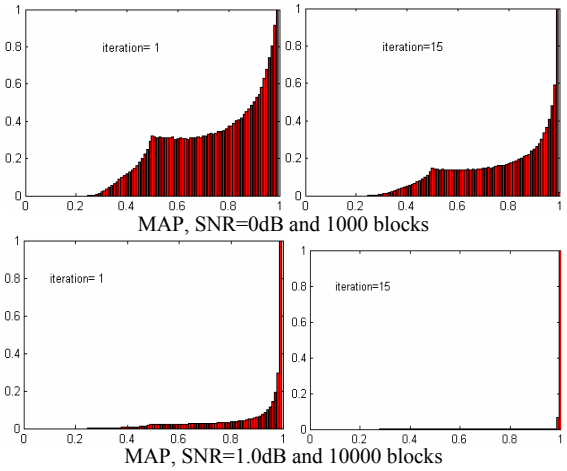


Fig. 2 APP normalized histograms for MAP decoding algorithm

The diagrams from Fig.2 and Fig.3 present the APP normalized histograms for both MAP and MaxLogMAP cases, performing 15 iterations without stop. The histogram's normalization means that all the obtained values have been divided by the biggest value after each iteration. In the MAP case only the APP for the maximum likelihood symbols were considered. The simulations show that these APP values get near to 1 when the SNR and iteration *i* increasing.

In the MaxLogMAP case all APP values were taken into account. Except for the maximum likelihood symbols APPs (which all are zero) the other APPs take values bigger when the SNR and *i* is increasing.

These results confirm the previous suppositions about the APP values. The remaining question is what the optimum values for the threshold μ are?

The BER and FER curves from Fig.4 compare the turbo-code performances with or without stop criterion. We used thresholds with the values: 0.99 and 0.9999 for MAP and 4 and 10 for MaxLogMAP.

The results show that the second threshold values for each algorithm give practically the same performance as is the case without iterations stop.

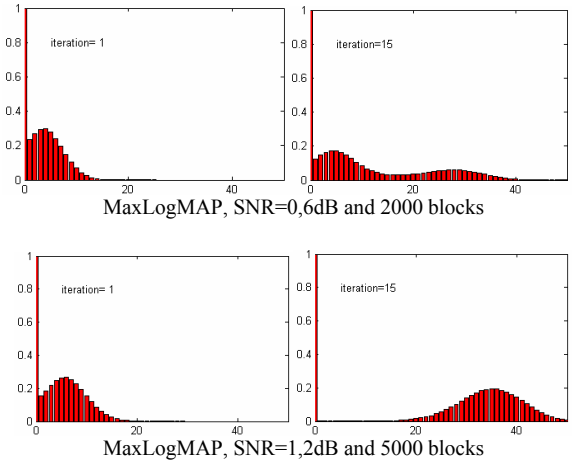


Fig. 3 APP normalized histograms for MaxLogMAP decoding algorithm

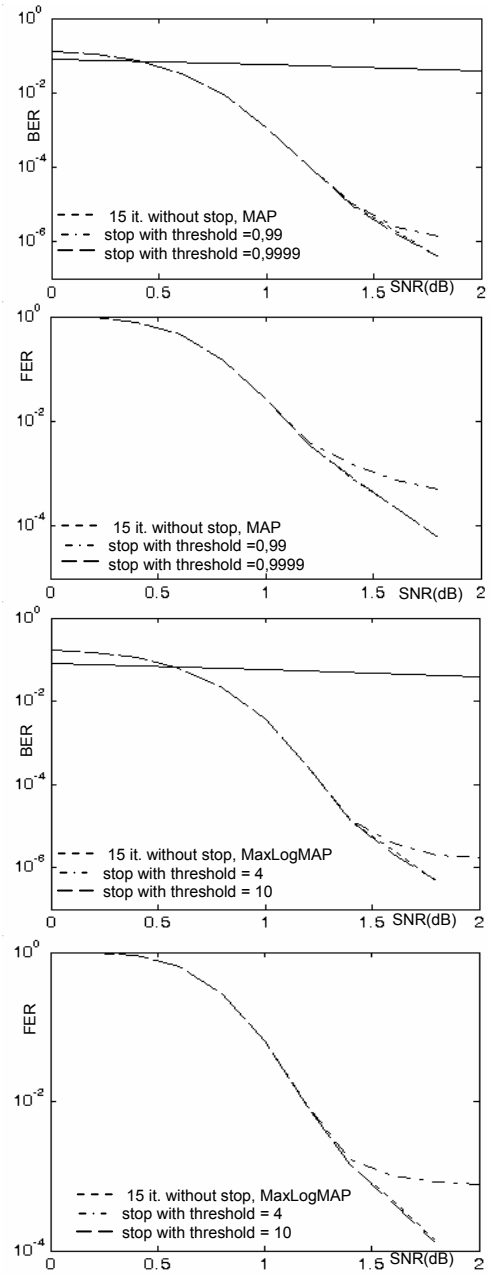


Fig. 4 BER and FER performance for two values of the threshold μ

In order to evaluate the efficiency of the proposed mlAPP criteria in Fig.5 we have plotted the average number of performed iterations as a function of SNR, and we have compared with the cases "without stop" and "genie". The genie criterion stops the iteration when (and only when) there are no more errors in the decoded block.

V. CONCLUSIONS

In this paper we present a new early stopping criterion, mlAPP, usable in the multi-binary turbo-codes cases when the symbol decoding is performed. The proposed criterion has two variants corresponding to the MAP and the MaxLogMAP decoding algorithm.

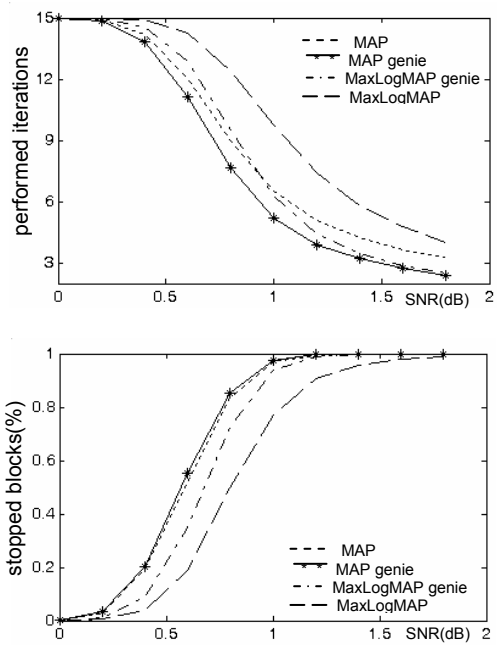


Fig. 5 The mlAPP stoping criterion efficiency.

The simulations show that both variants perform without any differences compared to the case of fixed number iterations and have the same efficiency close to the optimum (genie) stopping criterion.

REFERENCES

[1] C. Berrou, A. Glavieax, and P. Thitimajshima, "Near Shannon limit error-correcting coding and decoding: turbocodes", Proc. ICC'93, Switzerland, May 1993, pp.1064-1070

[2] J. Hagenauer, E. Offer, and L. Papke, "Iterative decoding of binary block and convolutional codes", IEEE Trans. Inf. Theory, 1996, 42, pp.429-445

[3] Rose Y. Shao, Shu Lin, and Marc P. C. Fossorier, "Two Simple Stopping Criteria for Turbo Decoding", IEEE Trans. Comm., vol.47, No.8, August 1999, pp.1117-1120

[4] Yufei Wu, Brian D. Woerner, and William J. Ebel, "A Simple Stopping Criterion for Turbo Decoding", IEEE Comm. Lett., vol.4, No. 8, August 2000, pp.258-260

[5] Fengqin Zhai, and Ivan J. Fair, "Techniques for Early Stopping and Error Detection in Turbo Decoding", IEEE Trans. Comm., vol.51, No. 10, October 2003, pp. 1617-1623

[6] C. Douillard, and C. Berrou, "Turbo Codes With Rate-m/(m+1) Constituent Convolutional Codes, IEEE Trans. Comm., vol.53, No. 10, Ocober, 2005, pp.1630-1638

[7] J. Tan, G. L. Stüber, "A MAP equivalent SOVA for non-binary turbo codes", in Proc. *IEEE International Conference on Communications, ICC*, pp. 602-606, New Orleans, L. A., June 2000.

[8] T. M. N. Ngatched, and F. Takawira, "Simple stopping criterion for turbo decoding", IEEE Electronics Letters, vol. 37, No. 22, October 2001, pp.1350-1351

[9] A. Matache, S. Dolinar, and F. Pollara, "Stopping Rules for Turbo Decoders", TMO Progress Report 42-142, August 2000, http://tda.jpl.nasa.gov/progress_report/, Jet Propulsion Laboratory, Pasadena, California

[10] Gilben Frank, Frank Kienle, and Norbert Wehn, „Low Complexity Stopping Criteria for UMTS Turbo-Decoders", in Proc. *Vehicular Techn. Conf. (VTC Spring '03)*, Jeju, Korea, April 2003.

[11] Nam Yul Yu, Min Goo Kim, Yong Serk Kim, and Sang Uoon Chung, "Efficient stopping criterion for iterative decoding of turbo codes", IEEE Electronics Letters, vol.39, No.1, January 2003, pp.73-74

[12] Dong-Ho Kim, and Sang Wu Kim, "Bit-Level Stopping in Turbo Decoding", *IEEE Communication Letters*, vol.10, March 2006, pp. 183-185.

[13] A. Taffin**,** "Generalised stopping criterion for iterative decoders", IEEE Electronics Letters, vol.39, No.13, June 2003, pp.993-994

[14] A. Hunt, S. Crozier, K. Gracie and P. Guinand, "A completely safe early stopping criterion for max-log Turbo code decoding", in Proc. *4th International Symposium on Turbo Codes and Rlated Topies*, Munich, Germany, April 3-7, 2006

[15] K. Gracie, A. Hunt, and S. Crozier, "Performance of Turbo Codes using MLSE-Based Early Stopping and Path Ambiguity Cheching for Inputs Quantized to 4 Bits", in Proc. *4th International Symposium on Turbo Codes*, Munich, April 3-7, 2006.

Buletinul Ştiinţific al Universităţii "Politehnica" din Timişoara

Seria ELECTRONICĂ şi TELECOMUNICAŢII
TRANSACTIONS on ELECTRONICS and COMMUNICATIONS

Tom 51(65), Fascicola 2, 2006

VoiceStudio: A HMM-based Tool for Research and Teaching in the Speech Recognition Field

Alexandru Căruntu[1], Gavril Toderean[1], Alina Nica[1]

Abstract – This paper introduces the Romanian speech recognition system *VoiceStudio*. As most state-of-the-art Automatic Speech Recognition (ASR) systems today, it is based on Hidden Markov Models. Although there are numerous toolkits designed for this task, they usually have no visual interface, which means that the student or the researcher needs to spend some considerably amount of time in order to learn their functionality. The system's modular design, together with some implementation issues are pointed out, as well as the future plans of development.

Keywords: speech recognition, Hidden Markov Models, visual interface

I. INTRODUCTION

Being a researcher in the speech recognition field is not easy. Before having some revolutionary ideas that will amaze the whole scientific community you have to understand all those feature extraction algorithms, not to talk about the Hidden Markov Models. The student who learns about speech recognition is in the same position. Although the beauty of the things, that you discover in this time, will make you say, in the end, that the whole journey worth, an easier way to learn will definitely help a lot of people.

A good example in this sense is the HMM toolbox included in the BNT package developed in MATLAB [1]. Since it is not designed for speech related experiments there is no feature extraction algorithms included, nor language or acoustic modeling. Not to forget the considerable time that MATLAB needs to process a large set of data. But the HMM implementation, based on [2], is very useful in understanding both discrete and continuous Hidden Markov Models.

A truly state-of-the-art tool is the HTK [3], developed in C++. It has very good capabilities of feature extraction, as well as HMM training. Its major drawback is the fact that it has no visual interface, all the commands being entered from the command line, many of them having lots of options and parameters to set. Although is an open source tool, the learning curve in understanding the code is significant.

In our approach we tried to combine the strongest points of both toolkits, in an attempt to obtain a strong, flexible, easy to use tool for speech recognition. We used Visual C++ and an object oriented solution for the problem. The goal was to create a good speech recognition engine which runs behind an easy to use interface.

We will present in this paper the results that we have obtained so far, namely the feature extraction and HMM modules. The work is still in progress and soon a language modeling module for Romanian language will be added also. Rest of the paper is organized as follows: a brief outline of the implementation issues, followed by a few experimental results, and finally, conclusions and future plans.

II. ISSUES OF IMPLEMENTATION

This paragraph will give a brief review of the modules which compose our system (Figure 1). When needed, more details will be revealed about the difficulties that we have encountered and the way that we have solved them.

A. *Data preparation*

The first step in the development of any recognizer is data preparation, since speech data is needed both for training and testing. *VoiceStudio* currently supports two file formats, WAV and TIMIT. No recording capabilities are included at this stage of development, so the data must be pre-recorded with another tool.

Of course, before working with any data we need a good matrix library. In our case, we used one developed in our laboratories as a part of a speech analysis tool [4].

A dictionary containing all the words to be recognized is also needed. It must be created by hand by the user, but further implementations will generate it from the sample sentences present in the training data.

B. *Feature extraction*

A detailed explanation of our work in this field can be found in [4]. The feature extraction module described in that paper is included in *VoiceStudio*. The features that we have incorporated in our recognition tool are

[1] Technical University of Cluj-Napoca,
e-mail: Alexandru.Caruntu@com.utcluj.ro

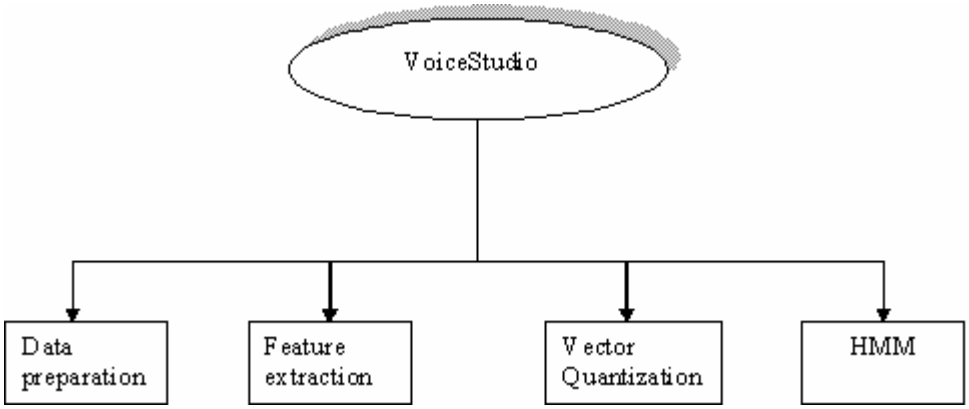


Fig. 1. The architecture of the *VoiceStudio* system

Linear Prediction Coefficients (LPC), Linear Prediction Cepstral Coefficients (LPCC), and Mel-Frequency Cepstrum Coefficients (MFCC).
A discussion is needed regarding the implementation of the last ones. As pointed out in [5], there are several ways of computing the MFCC coefficients:

- The solution proposed in 1980 by Davis and Mermelstein [6]: a filter bank of 20 filters, a sampling frequency of 10 kHz, and a speech bandwidth between 0 and 4600 Hz.
- The implementation from HTK, described in [7]: 24 filters, sampling rate greater than 16 kHz, [0, 8000] Hz speech bandwidth.
- The implementation from Malcom Slaney's Auditory Toolbox for MATLAB [8], which assumes 40 filters in the filter bank, a sampling rate of 16 kHz, and a speech bandwidth between 133 and 6854 Hz.

Several studies [9 – 11] have been done in order to compare the results of these implementations for speech recognition task. Based on them and our point of view regarding the best ratio between their results and the simplicity of implementation, we have chosen the last one for our system.
This implementation uses a filter bank of 40 equal area filters, which cover the frequency range from 133 to 6854 Hz in the following manner [5], [8]:

- The centre frequencies of the first 13 filters are linearly spaced in the range [200, 1000] Hz at 66.67 Hz one from the other.
- The centre frequencies of the last 27 filters are logarithmically spaced between 1071 and 6400 Hz, with a step of 1.0711703 Hz.

The filters in the filter bank are defined as:

$$H_i(k)=\begin{cases} 0, & k<f_{b_{i-1}} \\ \dfrac{2(k-f_{b_{i-1}})}{(f_{b_i}-f_{b_{i-1}})(f_{b_{i+1}}-f_{b_{i-1}})}, & f_{b_{i-1}}\le k\le f_{b_i} \\ \dfrac{2(f_{b_{i+1}}-k)}{(f_{b_{i+1}}-f_{b_i})(f_{b_{i+1}}-f_{b_{i-1}})}, & f_{b_i}\le k\le f_{b_{i+1}} \\ 0, & k>f_{b_{i+1}} \end{cases}, \quad (1)$$

where $i = 1,2,...,M$ stands for the i^{th} filter, f_{b_i} are $M + 2$ boundary points that specify the M filters, and $k = 1,2,...,N$ corresponds to the k^{th} coefficient of the N point FFT [5].
A detailed explanation of the whole procedure can be found in [12]. We will only point out here that the equalization of the area below the filters from equation (1) is due to the term

$$\frac{2}{f_{b_{i+1}} - f_{b_{i-1}}}. \quad (2)$$

Because of (2), the filter bank given by equation (1) is normalized in such a way that the sum of coefficients for every filter equals one.

C. *Vector Quantization*

Two algorithms for Vector Quantization (VQ) were implemented: *Linde – Buzo – Gray (LBG)* and *k-means*. VQ must be used in conjunction with discrete HMMs in order to represent by a single code a vector of features resulted through analysis. For this task any of the two algorithms can be used. Also, we used the latter to initialize some of the parameters of the continuous HMMs.
A detailed explanation of both of the algorithms is beyond the scope of this paper. We only point out that our implementation is based on [13] and [14].

However we encountered a major problem for both of the algorithms: empty clusters.
The principle of VQ technique is to find a set of vectors which describes best a set of data. These vectors are called *centroids*, and they form the *codebook* or the *dictionary*.
Vectors from the dataset are grouped into *clusters* based on their proximity to centroids. As a consequence of this process, empty clusters can result. We implemented a simple procedure in order to solve this problem:

Step 1. *First, we check to see if there are empty clusters.*

Step 2. *If yes, for the corresponding centroid, we try to find out which is the closest vector from dataset.*

Step 3. *We identify the cluster to which belongs the vector that we have found.*

Step 4. *If there are at least 2 vectors in this cluster we move the vector that we found in the empty cluster.*

Step 5. *If not, we go back to Step 2 (otherwise, if we move the vector, we will obtain another empty cluster).*

Future developments will take into account improvements of this algorithm in terms of computer efficiency.

D. *Hidden Markov Models*

As stated in [15], modern architectures for ASR systems are mostly software structures which generate a sequence of word hyphoteses from an acoustic signal. Most of the speech recognizers developed nowadays is based on Hidden Markov Models (HMM), because of their capability of best modeling the statistical nature of the speech signals. Since they outperformed all the other speech recognition techniques, we have also chosen them for our speech recognition engine.
A HMM is described by three parameters: the initial state distribution, the state transition probability distribution, and the observation symbol probability distribution (or output probabilities) in a certain state. Each of them is represented by a matrix, which must be estimated. A complete reference of the algorithms used for this task can be found in [2]. Here we will only point out the methods to solve the problems that occur during this process.
Currently *VoiceStudio* supports discrete *(DHMM)* and *continuous (CHMM)* HMMs. The difference between them is given by the nature of the output probabilities, which are distribution functions. If those functions are defined on a finite space, the models are discrete. In this case, the observations are vectors of symbols in a finite alphabet [15]. If distribution functions are defined as probability densities on a continuous observation space, the models are continuous. The most popular approach is to use mixture of Gaussians to characterize the model transitions, so we will need to estimate another three parameters in this case: the weights of the gaussians in the mixture, the means and the covariances.
While means are easily to initialize, using a k-means procedure, the problems arrise when we perform the same operation for covariances. In their case, we have to check if the matrix is singular, and if so, we have to adjust its values. This can be done by adding to the diagonal values a small quantity, until the determinant becomes different than zero.
VoiceStudio supports two kinds of covariance matrices: *full* and *diagonal*. Latter are preferred since they are defined with far less parameters, so we need less acoustic data to train them. Also, they are easier to estimate than full matrices. Ussually, setting a minimum value for their elements solves the singularity problem.
Finally, a problem which appears in any stage of designing a HMM is the small values of the probabilities. Performing multiplying operations over this data can lead to numbers which can not be represented by the computer. The solution is to scale the data with a scalling coefficient so that the values will not be close to zero.

III. EXPERIMENTAL RESULTS

An example of using *VoiceStudio* will be given in this section. As it can be seen in Figure 2, the user must take a few steps in order to perform a speech recognition experiment.
The *Settings* option from the menu is designated for setting up the parameters of the experiment. We divided these parameters into three categories. First of them (*Preprocessing parameters*) is formed by the type of window (Hamming or rectangular), frame size and rate expressed in milliseconds, an option to remove the DC mean from the signal, and an option to pre-emphasize the signal with a certain pre-emphasis coefficient. The second category is called *Features* and deals with the parameters used to represent the speech signal. Here, the user can choose between LPC, LPCC, and MFCC coefficients. Finally, the parameters of Hidden Markov Model are set, under the option *Model*: type of the model (DHMM or CHMM), number of observations, states, and mixtures, and the type of the covariance matrix (full or diagonal).
The *Data* option allows the selection of the location where the data is, and the parameterization of the data files. Also, when selecting the Vector Quantization option, a VQ algorithm (k-means or LBG) can be chosen by the user in order to attach the data to different clusters. This is a required step if we use a DHMM.
The last option, *Actions*, consists of two elements: *Training* and *Testing*. As suggested by their names, they deal with those two phases mandatory in the

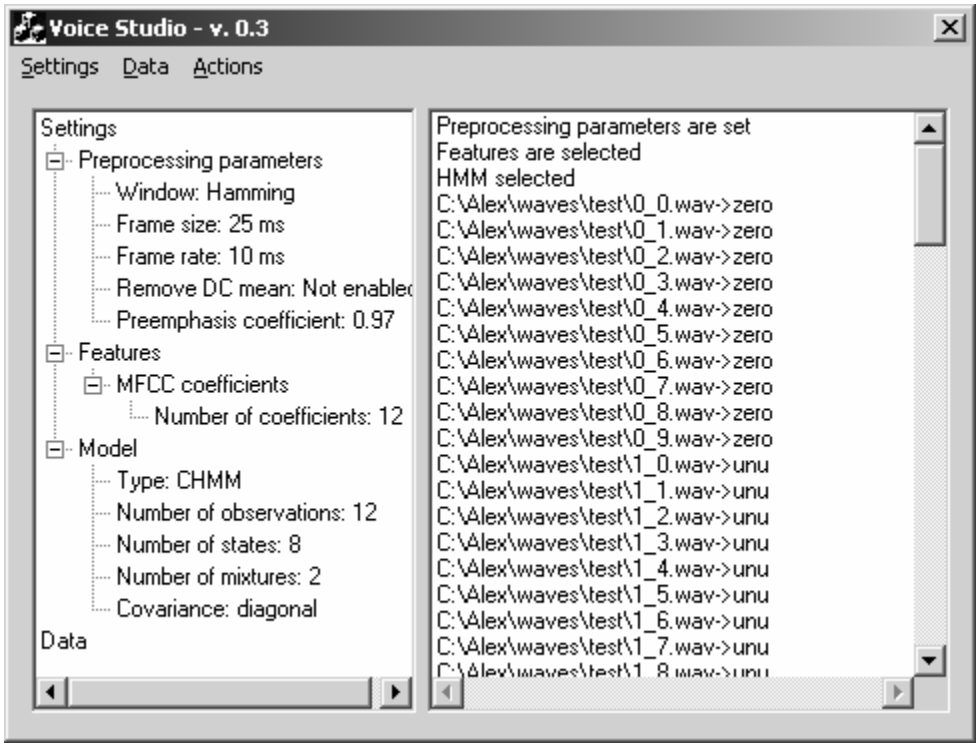


Fig. 2. The interface of *VoiceStudio*

design of any speech recognition system.
A number of test were performed in order to evaluate the system. Since our main interest was to build a robust HMM library, the task that we have chosen was isolated words recognition. We used for that a database formed by the digits from 0 to 9 in Romanian language. All three kinds of features mentioned before were used both with DHMMs and CHMMs. The results obtained range between 86% and 94% recognition rates, and can be considered satisfactory.

IV. CONCLUSIONS AND FUTURE PLANS

A speech recognition tool based on Hidden Markov Models was presented in this paper. Its major advantage compared with another existing products in this field is the visual interface, which helps a lot the process of research and learning. The problems that we have encountered during implementation, as well as their solutions, were emphasized. The experiments that we have performed showed that our HMM library is accurate and reliable.
Future directions of development will focus on language modeling for Romanian language and on increasing the visual capabilities of our software. Also, improving a number of algorithms that we have implemented, in terms of computer times, will also be taken into consideration.

REFERENCES

[1] "http://www.cs.ubc.ca/~murphyk/Software/HMM/hmm.html"

[2] L. R. Rabiner, "A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition", Proc. IEEE, 77(2):257--286, 1989.

[3] "htk.eng.cam.ac.uk/"

[4] A. Căruntu, G. Toderean, and A. Nica, "Software Environment for Speech Processing", WSEAS Transactions on Communications, 4(8): 664 – 671, 2005.

[5] T. Gancev, N. Fakotakis, G. Kokkinakis, "Comparative Evaluation of Various MFCC Implementations on the Speaker Verification Task", Proc. of the 10th International Conference on Speech and Computer, SPECOM 2005, Vol. 1, pp. 191-194, 2005.

[6] S. B. Davis, P. Mermelstein, "Comparison of Parametric Representations for Monosyllabic Word Recognition in Continuously Spoken Sentences", IEEE Trans. on Acoustic, Speech and SignalProcessing, 28(4):357–366, 1980.

[7] S. J. Young, J. Odell, D. Ollason, V. Valtchev, P. Woodland, *The HTK Book. Version 2.1*, Department of Engineering, Cambridge University, UK, 1995.

[8] M. Slaney, "Auditory Toolbox. Version 2", Technical Report #1998-010, Interval Research Corporation, 1998.

[9] F. Zheng, G. Zhang, Z. Song, "Comparison of different implementations of MFCC", J. Computer Science & Technology, 16(6):582-589, Sept. 2001.

[10] B. J. Shannon., K. K. Paliwal, "A comparative study of filter bank spacing for speech recognition", Proc. of Microelectronic engineering research conference, Brisbane, Australia, Nov. 2003.

[11] M. D. Skowronski, J. G. Harris, "Exploiting independent filter bandwidth of human factor cepstral coefficients in automatic speech recognition", Journal of the Acoustical Society of America, 116(3):1774–1780, Sept. 2004.

[12] X. Huang, A. Acero, H. Hon, *Spoken Language Processing: A Guide to Theory, Algorithm, and System Development*, Prentice-Hall, 2001.

[13] S. Furui, *Digital Speech Processing, Synthesis and Recognition, Second Edition, Revised and Expanded*, Marcel Dekker, Inc., 2001.

[14] G. Toderean, A. Caruntu, *Metode de recunoastere a vorbirii*, Risoprint, 2005.

[15] R. A. Cole, J. Mariani, H. Uszkoreit, A. Zaenen, V. Zue, *Survey of the State of the Art in Human Language Technology*, Technical report, Center for Spoken Language Understanding CSLU, Carnegie Mellon University, Pittsburgh, PA, USA, 1996.

Buletinul Ştiinţific al Universităţii "Politehnica" din Timişoara

Seria ELECTRONICĂ şi TELECOMUNICAŢII
TRANSACTIONS on ELECTRONICS and COMMUNICATIONS

Tom 51(65), Fascicola 2, 2006

Widespread Deployment of Voice over IP and Security Considerations

Mihai Constantinescu[1], Doina Cernăianu[2], Dragoş Mischievici[3], Victor Croitoru[4]

Abstract – During the last years, Internet facilities like email, the world-wide-web (WWW), and e-commerce have generated a boost of Internet growth, making offering services possible in fundamentally new ways. One of these services is Voice over IP (VoIP), also named Internet Telephony (IP telephony). With most major telecommunications carriers preparing for VoIP mass deployment, the security of service cannot remain a second priority anymore. This paper analyzes the main aspects of VoIP wide deployment and highlights the benefits of using a new security concept, Session Border Controller (SBC), in solving VoIP security issues.
Keywords: VoIP, SIP, DoS, Firewall, SBC.

I. INTRODUCTION

The ubiquitous presence of Internet caused a powerful change in human life. People begin to use Internet facilities for almost all communications purposes (fig.1).

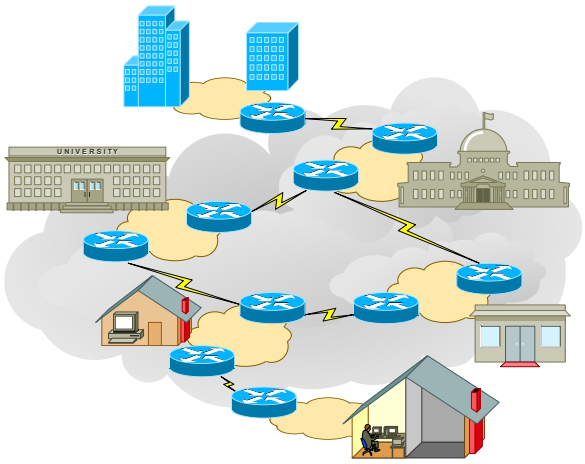


Fig. 1. Internet communications

One of these facilities is VoIP, or Internet telephony. Based on transport of voice traffic over IP networks, Internet telephony by its nature relies on technology that does not distinguish geographic borders. VoIP is not another facet of the traditional telephony service, but a new frontier in communications for individuals and businesses alike.

VoIP service has some major benefits. First, it allows costs saving by eliminating the toll charges for long-distance and even local calls. For enterprises with multiple branch offices, the use of VoIP eliminates all costs associated with calls between offices. Each phone extension is reached in the same way, despite the distance between extension and the main office. VoIP enables call transferring over a building, inside a town, or over continents. More than that, Internet telephony enables new applications such as conferencing, voice mail, unified communications (fig.2), and click to dial, all of these resulting in enhanced productivity.

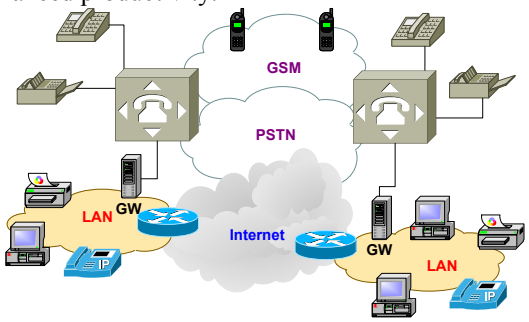


Fig. 2. Unified communications

VoIP reduces the costs and complexity of service implementation and management. Most of VoIP equipment allow remote configuration via a web-browser interface, eliminating the need for third party interventions.

However, the wide adoption of VoIP generates an increased risk of widespread security violations, and raises new security issues related to the privacy of communications, unified services, and transparency of service access over different networks and carriers. This paper presents the principles of VoIP communications, VoIP special characteristics related

[1] „Politehnica" University Bucharest, Electronics, Telecommunications and IT Faculty, Telecommunication Dept, 1-3, Iuliu Maniu Blvd, sector 6, Bucharesti, e-mail mac_2470@yahoo.com
[2] Teletrans SA Bucharest
[3] Teletrans SA Bucharest
[4] Politehnica" University Bucharest, Electronics, Telecommunications and IT Faculty, Telecommunication Dept, 1-3, Iuliu Maniu Blvd, sector 6, Bucharesti, e-mail croitoru@adcomm.pub.ro

to data networks, analyzes security problems occurred in VoIP use, and shows the role of Session Border Controller in solving them.

II. THE VOICE OVER IP PRINCIPLES AND SPECIAL CHARACTERISTICS

The transmission of voice signals through a network, either an IP network or an old public telephone network (PSTN-Public Switched Telephone Network/POTS-Plain Old Telephone System) involves the following:

- A coder/decoder (CODEC) equipment/operation that transforms analog voice signals into a digital stream;
- A signaling mechanism/protocol that coordinates the actions of network elements in order to complete a call between the endpoints (usual phones /IP phones);
- A call control (bearer-control) mechanism to transport the voice digital stream over the network;
- A database for addressing and billing purposes.

In the past, telephony was designed to cover a wide area and to provide basic voice communications services. In order to achieve this, telephony networks use a centralized architecture. There is a continuous wired connection (telephone circuit) from the telephone itself to the first telephone office (central office-CO). The old telephony network concentrates all intelligence in the core network switch in the CO. The telephone itself is a dump terminal. The call signaling and audio path use the same telephone circuit.

IP telephony delivers the same facilities, but in a totally different way. There are three approaches for voice services over Internet:

- Using signaling concepts from the telephone industry (ITU-T recommendation H.323); [1]
- Using control concepts from the telephone industry (Softswitches); [2]
- Using the Internet protocols (Session Initiation Protocol –SIP). [3]

The nature of interactive communications and the type of service is defined and determined by the signaling used for establishing the communication. Due to high flexibility and adaptability to a great diversity of IP networks, SIP is the most used VoIP protocol, and all discussion about VoIP will be directly related to it.

"Session Initiation Protocol (SIP) is an application-layer control (signaling) protocol for creating, modifying, and terminating sessions with one or more participants. These sessions include Internet telephone calls, multimedia distribution, and multimedia conferences [3]."

"SIP is not a vertically integrated communications system. SIP is rather a component that can be used with other IETF protocols to build complete multimedia architectures. Typically, these architectures will include protocols such as the Real-time Transport Protocol (RTP) [4] for transporting real-time data and providing QoS feedback, the Real-Time Streaming Protocol (RTSP) [5] for controlling delivery of streaming media, the Media Gateway Control Protocol (MEGACO) [6] for controlling gateways to the Public Switched Telephone Network (PSTN), and the Session Description Protocol (SDP) [7] for describing multimedia sessions."[3]

SIP has been designed as a multimedia protocol using a distributed architecture with universal resource location (URL) for text-based messages, trying to take advantage of the Internet model for creating VoIP netwoks and applications (fig.3).

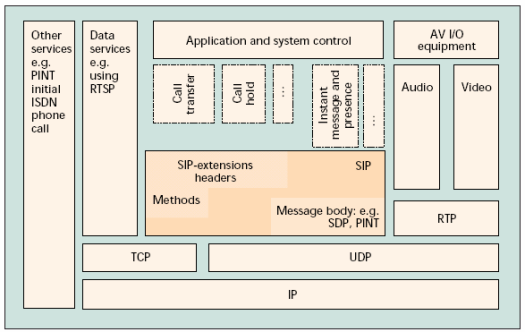


Fig. 3. SIP protocol stack

Because SIP is an IP-based protocol working in peer-to-peer architecture, unlike other VoIP protocols such as H.323, MGCP, or MEGACO, its intelligence resides at the network edge.

The SIP standard defines four entity types, as shown in fig.4:

- user agent (UA)
- proxy server (Proxy)
- registration server (Registrar)
- Redirect server (Redirect).

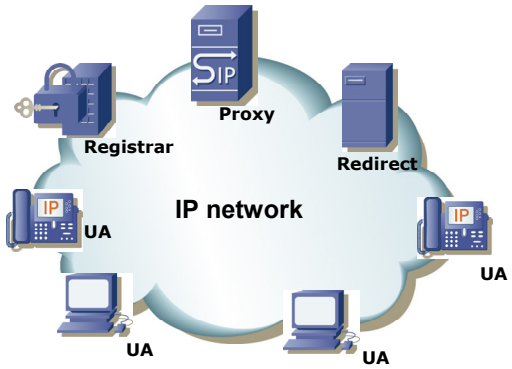


Fig. 4. SIP network elements

SIP signaling consists of an exchange of short messages that contain session descriptions, which allow participants to agree on a common set of media

parameters. The path between a pair of SIP clients is handled by SIP proxy/registrar servers. They keep information related to the current location of clients, authenticate and authorize users for services, and route requests to those clients.
Each SIP client has to register before to communicate, sending REGISTER requests to a SIP registrar server (fig.5).

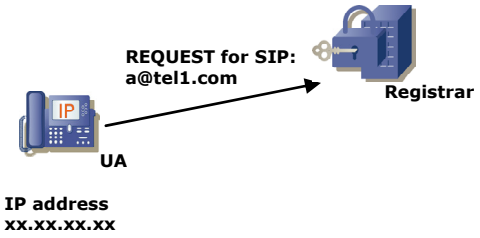


Fig. 5. SIP Registrar server

After a successfully registration, the SIP client can communicate with other SIP client using SIP proxy servers or SIP redirect servers.
The SIP proxy is a device in the signaling path that routes requests to their destinations (fig.6).

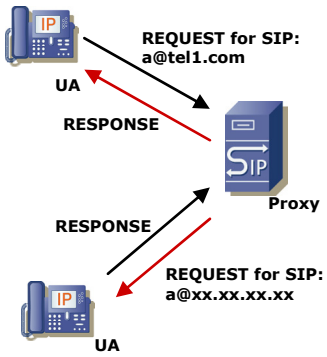


Fig. 6. SIP proxy server

Also named rerouting server, SIP redirect server responds to requests by redirecting them to another device (fig.7).

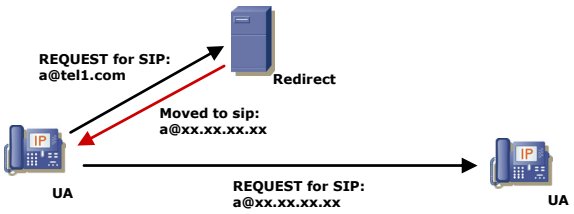


Fig.7. SIP Redirect server

SIP communication is made up of messages that are sent between devices using UDP, or TCP.
A SIP dialog means a persistent link between two devices that is used to associate transactions. A call contains multiple dialogs.
A SIP message contains a call identifier field (Call-ID) that is used to link the dialogs and transaction into an application-level concept of a call.

The default operation for SIP using UDP/TCP consists in responses generated by SIP proxy/registrar servers to requests generated by SIP phones (User Agent Client). The response message contains the source address from the request message in the SIP "via" header and in the "received" parameter, and the source port from the request message in the SIP "via" header.
The RTP is the most common media transport protocol used in SIP communications. Negotiation of RTP parameters is done using the SDP protocol. In the SDP part of the SIP message there are specified the address and port of each client to receive media.
SIP initiates a call through an INVITE message and an answer from the called party. Both the invitation and the answer contain a session description which indicates the terminal capacity. Proxy and rerouting servers are responsible for the parties' user names and IP addresses translation. (fig. 8).

```
INVITE sip:7170@iptel.org
SIP/2.0
Via: SIP/2.0/UDP
195.37.77.100:5040;rport
Max-Forwards: 10
From: "jiri"
<sip:jiri@iptel.org>;tag=76ff
7a07-c091-4192-84a0-
d56e91fe104f
To: <sip:jiri@bat.iptel.org>
Call-ID: d10815e0-bf17-4afa-
8412-
d9130a793d96@213.20.128.35
CSeq: 2 INVITE
Contact:
<sip:213.20.128.35:9315>
User-Agent: Windows RTC/1.0
Proxy-Authorization: Digest
username="jiri",
realm="iptel.org",
  algorithm="MD5",
uri="sip:jiri@bat.iptel.org",

.....
```

Fig.8. SIP message example

As a protocol used in a distributed architecture, SIP allows companies to build scalable, resilient and redundant large scale newtorks. The protocol provides mechanisms to interconnect with other VoIP networks in order to add intelligence and new options to each of the terminals, SIP proxy servers and rerouting servers.

III. SECURITY PROBLEMS IN VoIP USE

Security issues in VoIP are different and in ways more complex than security for data applications. IP telephony involves multiple layers of the protocol stack, requiring interoperability among different new

and legacy protocols, and interactions among multiple network elements. Denial-of-Service, eavesdropping, connection hijacking, and call fraud will take new forms in a voice-data unified network. New security risks arise from network interconnections, and also due to VoIP network vulnerability to virus and worm spreading through the data network elements. There is an individual solution for each threat, but a global one is much more desired.

Denial-of-Service (DoS)/Distributed DoS (DDoS)

"A DoS attack is a malicious attempt by a single person or a group of people to cause the victim, site, or node to deny service to its customers. When this attempt derives from a single host in the network, it constitutes a DoS attack. On the other hand, it is also possible that a lot of malicious hosts coordinate to flood the victim with an abundance of attack packets, so that the attack takes place simultaneously from multiple points. This type of attack is a Distributed Denial-of-Service."[8]
A DoS/DDoS attack can cause an enterprise a dramatic loss of revenue, due the loss of communication.

Eavesdropping (Call Interception)

Call interception is the possibility of unauthorized monitoring of RTP traffic. It can occur especially from within the network, and it exploits the vulnerability of SIP servers to registration hijacking, impersonation, and DoS/DDoS. The server can be tricked in acting as a codec converter between the two SIP clients, allowing voice traffic to be recorded or routed to other destination.

Signal Protocol Tampering

It occurs when a malicious user captures and changes the packets involved in the initiation of the call. Thus he can change different fields in VoIP packets, acting for the VoIP network as an authorized network user. In that way, the thief can make expensive VoIP calls.

Presence Theft

Presence theft consists in the impersonation of an authorized user sending or receiving data. It is linked with the Signal Protocol Tampering.

Toll Fraud

The ability of a hacker to use the resources of the VoIP network in order to make unauthorized VoIP calls.

Call Handling OS

The call management is done by machines running different operating systems (OS). If the OS is compromised, it opens a security gap to be used further.

Spam over Internet Telephony (SPIT)

Even though SPIT seems to be just an inconvenience, it is truly DoS and greatly reduces the bandwidth of the network.

Other security problems occur at VoIP service use over different networks, or carriers, involving call/user authentication protocol translation.

There are several solutions to these security issues, but most of them mean solving one issue in the detriment of the others.

Virtual LANs

The service providers look to protect the integrity of their networks, using firewalls and Network Address Translators (NAT), in order to implement Virtual LANs. By keeping VoIP and data in different VLANs, the network performance and security increase. Meanwhile, the peer-to-peer model of SIP encounters serious problems at NAT traversal. First, NAT does not allow any incoming calls from public to private hosts. Second, SIP messages encapsulate the source address and port at application level. The NAT changes the address and port of packets, but only in IP and TCP/UDP headers, so the messages will be discarded by the SIP client. Moreover, SIP uses different ports to communicate, therefore several SIP messages will be blocked by NAT due to port filtering. [9-19]

Encryption

Encryption is a good method of protection, but can be done only within the network. The users will be isolated from the outside. Even so, the existence of multiple encryption points can affect the performance of the VoIP network itself.

Direct Firewall Support

It implies the modification of firewall functionality, adding an Application Layer Gateway facility. That solution makes the security policy more complex to manage. It is also more restrictive to a scalable VoIP network. [9-19]

Reverse Proxies

It is based on segmenting the VoIP traffic using multiple servers that are acting as B2BUAs (Back-to-

Back User Agents). A B2BUA terminates the call signaling from one endpoint and initiates the same call to the other endpoint, but with other identity. In that way, the identity of the call is hidden to the rest of network elements. The use of reverse proxies solely for this purpose is irrelevant.

IV. SESSION BORDER CONTROLLER (SBC)

At first glance, SBC is a kind of firewall for VoIP traffic. In reality, due its complex activity, an SBS is much more than a simple multimedia firewall.

"In its simplest form, a Session Controller enables interactive communication across the borders or boundaries of disparate Internet Protocol (IP) networks. In doing this, Session Controllers connect islands of IP voice and/or video traffic without requiring all IP traffic to first be converted into TDM at a handoff point between networks. Session Controllers operate at Layer 5 of the network and work with - but don't replace – devices such as Softswitches, NAT devices and firewalls."[20]

An SBC cooperates with firewalls in order to enable authorized connectivity from the outside to inside, avoiding the "incoming signaling from public network" issue. An SBC performs some NAT functions, but does not interfere with it. The address and port changes affect only the current SIP connection, the rest of data traffic being under NAT control.
A SBC contains two logical entities [21]: SBC signaling server (SBC-SIG) and SBC media server (SBC_MEDIA).

SBC-SIG

SBC media server is dealing with SIP signaling between SIP clients behind the NAT and the SIP proxy server. It is configured as a transit point for SIP signaling messages and provides complete visibility and control of call establishment. The SBC signaling server also controls the interval for SIP register update, in order to avoid the "binding timer" issue.
The SBC-SIG processes the SIP user registration, modifies SIP header (contact and via header), in order to allows the correct processing of SIP clients and SIP server messages. It performs address and port modification to permit NAT traversal. This behavior qualifies it as a B2BUA.[21]
Communication with the SBC media server, for traffic management and synchronization, and resolution of SIP servers through DNS, is also done by SBC-SIG.

SBC-MEDIA

SBC media server operates under the control of the SBC signaling server. It acts as a transit point for RTP and RTCP traffic between SIP clients. It modifies SDP parameters to allow NAT traversal for media using NAT's pin-holes, but without interfering with NAT security policy. This is a typically RTP proxy behavior. [21]
Being under control of SBC signaling server, the SBC media server provides full visibility and control of the media traffic for each SIP connection. Additionally, it can act as a dynamic NAPT that hides details of the network elements and topology. [9-19]

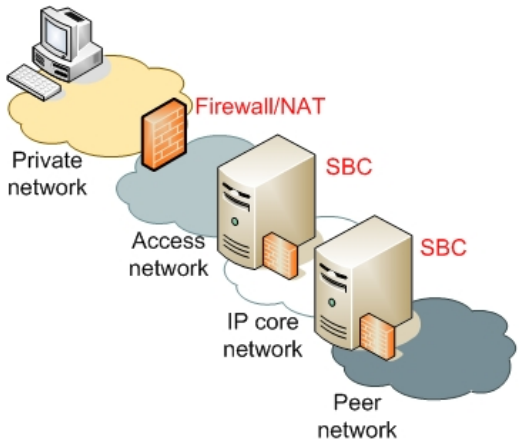


Fig. 9 SBC in a scalable VoIP architecture

V. SBC ROLE IN SOLVING VoIP SECURITY ISSUES

DoS /DDoS

SBC-SIG may identify badly-formed messages. It can knows the users identity. It will rejects the messages with wrong format and stop the signaling from the sourse IP identified as attacker. Additional, the firewall may be configured to filter the RTP traffic that don't have the SBC-SIG appouval (don't belong to a call).

Eavesdropping (Call Interception)

Call Admission Control (CAC) is the main function what differentiates SBC from ALG. Each SBC-SIG controls the call signalling though the network. Acting as a B2BUA, it hides the network internal topology from the exterior, protecting the identity of network users.

Signal Protocol Tampering

Each call signalling is monitored and all traffic from malicious sources is dropped.

Presence Theft

Presence theft is avoided by eliminating the Signal Protocol Tampering threat.

Toll Fraud

Each call is identified and SBC controls also VoIP network bandwidth use.

Call Handling OS

The data firewall protects the network from usual attacks, and the SBC gives more protection controlling the signalling and media traffic. Doing so, network elements are keeping safe.

Spam over Internet Telephony (SPIT)

CAC function of SBC has the ability to limit the traffic for each call, avoiding SPIT.

Virtual LANs

The NAT/firewall presence is no more a problem for VoIP traffic. The SBC-SIG acts as B2BUA, and opens pine-holes in firewall for the media traffic controlled by SBC-MEDIA. All fields in VoIP messages are changed by SBC-SIG.

Encryption

Encryption is no longer necessary, due the CAC function.

Direct Firewall Support

SBC work in cooperation with NAT/firewall, but does not affect the general security policy implemented on firewall. Doing so, the security remains simple and efficient. It allows also the scalability of VoIP networks.

Reverse Proxies

There is no need for additional media gateways acting as reverse proxies, due the B2BUA behavior of SBC-SIG.

VI. CONCLUSION

SBC is now the key element for VoIP security. By having full control and visibility of all media sessions, a SBC can easy implement a scalable VoIP network architecture over multiple boundaries, all existing equipment remaining unchanged. SBC can be used as an interface between an enterprise and the service provider network, on the border between two providers with a reciprocal agreement related to VoIP traffic, or within a provider offering VPN services to its customers, to bridge calls across the customers' VPN sites.

The future large deployment of SBC depends on a standardization that is still missing, and on the acceptance of SBC's manufacturers to refer to these standards.

REFERENCES

[1] ITU-T Recommendation H.323: *"Packet-Based Multimedia Communications Systems"*

[2] *Softswitch definition* http://en.wikipedia.org/wiki/Softswitch

[3] J. Rosenberg, et al., *"SIP: Session Initiation Protocol"*, RFC 3261, June 2002

[4] H. Schulzrinne, S. Casner, R. Frederick, V. Jacobson, *"RTP: A Transport Protocol for Real-Time Applications "*, RFC 1889

[5] H. Schulzrinne, A. Rao, R. Lanphier, *"Real Time Streaming Protocol (RTSP)"*, RFC 2326, April 1998

[6] IETF RFC 3015*: "Megaco Protocol Version 1.0"*

[7] Handley, M., Schultzrinne, H., IETF RFC 2327: *"SDP: Session Description Protocol"*, April 1998

[8] C. Patrikakis, et al, *"Distributed Denial of Service"*,in The Internet Protocol Journal, Volume 7, Number 4

[9] G. Huston, *"Anatomy: A Look Inside Network Address Translators"*, in The Internet Protocol Journal, Volume 7, Number 3

[10] M. Constantinescu, *"NAT/Firewall Traversal for SIP: issues and solutions"*, ISSCS 2005

[11] P. Srisuresh, M. Holdrege, *"IP Network Address Translator (NAT) Terminology and Considerations"*, RFC 2663, August 1999

[12] J. Rosenberg, D. Drew,H. Schulzrinne, *"Getting SIP through Firewalls and NATs"*, Internet Draft, draft-rosenberg-sip-firewalls-00.txt, February 2000

[13] J. Rosenberg, H. Schulzrinne, *"An Extension to the Session Initiation Protocol (SIP) for Symmetric Response Routing"*, RFC 3581, August 2003

[14] J. Rosenberg, A. Hawrylyshen, *"SIP Conventions for Connection Usage"*, Internet Draft , draft-ietf-jennings-sipping-outbound-00 (work in progress), July 2004

[15] D. Wing, *"Symmetric RTP and RTCP Considered Helpful"*, Internet Draft, draft-wing-mmusic-symmetric-rtprtcp-01 (work in progress).

[16] J. Rosenberg, J. Weinberger, C. Huitema, R. Mahy, *"STUN - Simple Traversal of User Datagram Protocol (UDP) Through Network Address Translators (NATs)"*, RFC 3489, March 2003.

[17] J. Rosenberg, R. Mahy, C. Huitema, *"Traversal Using Relay NAT (TURN)"*, Internet Draft, draft-rosenberg-midcom-turn-06, October 2004.

[18] J. Rosenberg, *"Interactive Connectivity Establishment (ICE): A Methodology for Network Address Translator (NAT) Traversal for Multimedia Sessions Establishment Protocols"*, Internet Draft, draft-ietf-mmusic-ice-03, October 2004.

[19] B. Carpenter, *"Middleboxes: Taxonomy and Issues"*, RFC 3234, February 2002.

[20] White Paper - *SignallingProxy™ - Accelerating the Deployment of SIP Services*, http://www.newport-networks.com/whitepapers/spwpes.html

[21] J. Hardwick, "Session Border Controllers, Enabling the VoIP Revolution" www.dataconnection.com

Index of Authors