

CONTRIBUȚII LA ANALIZA ȘI ÎMBUNĂTĂȚIREA PERFORMANȚELOR TURBO CODURILOR ÎN CANALE CU FADING PLAT

Teză destinată obținerii
titlului științific de doctor inginer
la
Universitatea "Politehnica" din Timișoara
în domeniul Inginerie Electronică și Telecomunicații
de către

ing. Maria Kovaci

Conducător științific:
Referenți științifici:

prof.univ.dr.ing. Miranda Naforniță
prof.univ.dr.ing. Monica Borda
conf.univ.dr.ing. Carmen Gerigan
prof.univ.dr.ing. Alexandru Isar

Ziua susținerii tezei: 06.11.2009

Seriile Teze de doctorat ale UPT sunt:

- | | |
|------------------------|---|
| 1. Automatică | 7. Inginerie Electronică și Telecomunicații |
| 2. Chimie | 8. Inginerie Industrială |
| 3. Energetică | 9. Inginerie Mecanică |
| 4. Ingineria Chimică | 10. Știința Calculatoarelor |
| 5. Inginerie Civilă | 11. Știința și Ingineria Materialelor |
| 6. Inginerie Electrică | |

Universitatea „Politehnica” din Timișoara a inițiat seriile de mai sus în scopul diseminării expertizei, cunoștințelor și rezultatelor cercetărilor întreprinse în cadrul școlii doctorale a universității. Seriile conțin, potrivit H.B.Ex.S Nr. 14 / 14.07.2006, tezele de doctorat susținute în universitate începând cu 1 octombrie 2006.

Copyright © Editura Politehnica – Timișoara, 2006

Această publicație este supusă prevederilor legii dreptului de autor. Multiplicarea acestei publicații, în mod integral sau în parte, traducerea, tipărirea, reutilizarea ilustrațiilor, expunerea, radiodifuzarea, reproducerea pe microfilme sau în orice altă formă este permisă numai cu respectarea prevederilor Legii române a dreptului de autor în vigoare și permisiunea pentru utilizare obținută în scris din partea Universității „Politehnica” din Timișoara. Toate încălcările acestor drepturi vor fi penalizate potrivit Legii române a drepturilor de autor.

România, 300159 Timișoara, Bd. Republicii 9,
tel. 0256 403823, fax. 0256 403221
e-mail: editura@edipol.upt.ro

Cuvânt înainte

Teza de doctorat a fost elaborată pe parcursul activității mele în cadrul Departamentului de Comunicații al Universității „Politehnica” din Timișoara.

Îi mulțumesc în primul rând conducătoarei mele de doctorat Prof. Dr. Ing. Miranda Naforniță pentru sprijinul și sfaturile utile pe care mi le-a dat pe parcursul realizării și elaborării acestei teze.

În egală măsură mulțumesc d-lui Prof. Dr. Ing. Ioan Naforniță pentru încrederea pe care mi-a acordat-o cât și pentru baza materială fără de care n-aș fi putut ajunge la rezultatele din cadrul acestei lucrări.

Deasemenea, țin să-mi exprim întreaga mea recunoștință membrilor comisiei fără de care susținerea acestei teze n-ar fi avut loc. Mulțumesc d-lui președinte al comisiei Prof. Dr. Ing. Marius Oteșteanu, d-nei Prof. Dr. Ing. Monica Borda de la Universitatea Tehnică din Cluj-Napoca, d-nei Conf. Dr. Ing. Carmen Gerigan de la Universitatea Transilvania din Brașov și, nu în ultimul rând, d-lui Prof. Dr. Ing. Alexandru Isar de la Universitatea „Polithenica” din Timișoara pentru faptul de a accepta să facă recenzarea acestui material, pentru acordul favorabil de a susține în public această teză, cât și pentru observațiile constructive pe care mi le-au făcut. O mențiune specială se cuvine d-nei Prof. Dr. Ing. Monica Borda deoarece, datorită domniei sale, am avut acces la cele mai importante și utile cărți din domeniul turbo codurilor.

Mulțumesc d-lui Prof. Dr. Ing. Michel Jézéquel și d-nei Prof. Dr. Ing. Catherine Douillard de la Ecole Nationale Supérieure des Telecommunications de Bretagne pentru seminariile despre turbo coduri ținute în cadrul departamentului nostru de Comunicații.

Mulțumesc colegului meu Horia Baltă pentru spiritul de echipă de care a dat dovadă cât și tuturor colegilor mei de la Facultatea de Electronică și Telecomunicații, care m-au ajutat și m-au sprijinit pe parcursul pregătirii acestei teze și în special mulțumesc d-lui Prof. Dr. Ing. Alexandru Isar pentru discuțiile avute.

Deasemenea, țin să mulțumesc tuturor colaboratorilor mei și în special d-lui cercetător Dr. Ing. Alexandre de Baynast pentru ajutorul acordat și încurajările făcute cât și pentru colaborarea concretizată în elaborarea de articole.

Mulțumesc familiei mele și tuturor prietenilor pentru susținere și încurajări.

Nu în ultimul rând, mulțumesc lui Dumnezeu pentru faptul că m-a ajutat să pot duce la bun sfârșit această teză.

Familiei mele,

Kovaci, Maria

Contribuții la analiza și îmbunătățirea performanțelor turbo codurilor în canale cu fading plat

Teze de doctorat ale UPT, Seria 7, Nr. 20, Editura Politehnica, 2009, 160 pagini, 82 figuri, 12 tabele.

ISSN:1842-7014

ISBN: 978-973-625-994-4

Cuvinte cheie: turbo coduri, coduri RSC, interleaver, algoritmi de decodare, canal cu fading plat

Rezumat,

În prezent turbo codurile se utilizează în cele mai moderne sisteme de comunicații. Un exemplu potrivit este cel al sistemelor fără fir (wireless). În cazul acestor sisteme canalele sunt variabile în timp, deoarece propagarea se face pe mai multe căi. Acestea sunt numite canale cu fading. Majoritatea lucrărilor despre turbo coduri din literatura de specialitate tratează cazul canalului AWGN (Aditive White Gaussian Noise). Scopul acestei teze este studiul sistematic al comportării turbo codurilor pe canale cu fading plat.

Capitolele 2 și 3 reprezintă o introducere în teoria turbo codurilor, pe care am considerat-o necesară pentru ca teza să aibă un caracter unitar. Capitolul 4 reprezintă partea centrală a tezei. Capitolul 5 se referă la modul în care acționează ultima generație de turbo coduri, cele multi-binare, în cazul canalelor cu fading plat.

Cuprins

Listă abrevieri	7
Listă de tabele	9
Listă de figuri	10
1. Introducere	15
2. Coduri convoluționale	18
2.1 Principiul codării convoluționale	18
2.2 Tipuri de coduri convoluționale	20
2.3 Reprezentarea grafică a codurilor convoluționale	23
2.4 Distanța liberă	25
2.5 Funcția de transfer asociată unui cod convoluțional	26
2.6 Simularea spectrului ponderilor	30
3. Turbo coduri	33
3.1 Turbo codorul	34
3.2 Turbo decodorul	35
3.3 Concatenarea paralelă a codurilor convoluționale	36
3.4 Dispozitive de întretesere	37
3.4.1 Interleaver aleator	39
3.4.2 Interleaver de tip S	39
3.4.3 Interleaver bloc	40
3.4.4 Interleaver pseudo-aleator	40
3.4.5 Interleaver Takeshita-Costello	41
3.4.6 Interleaver bloc-aleator	41
3.4.6.1 Interleaver bloc aleator în linie, BRL	42
3.4.6.2 Interleaver bloc cu linii aleatoare, BLR	43
3.4.7 Performanțele interleaver-elor utilizate în turbo coduri	44
3.5 Algoritmi de decodare	50
3.5.1 Algoritmul Viterbi	51
3.5.1.1 Algoritmul Viterbi cu decizie hard	51
3.5.1.2 Algoritmul Viterbi cu decizie soft	52
3.5.2 Algoritmul MAP	54
3.5.3 Algoritmul Max-Log-MAP	62
3.5.4 Algoritmul Log-MAP	65
3.5.5 Performanțe ale algoritmilor de decodare	67
3.6 Concluzii	68
4. Performanțele turbo codurilor în canalele cu fading plat	69
4.1 Propagarea radio în comunicațiile mobile	69
4.2 Modelul sistemului de transmisie	77
4.3 Fading plat de tip Rayleigh	79
4.4 Fading plat de tip Rice	86
4.5 Fading plat de tip Nakagami	94
4.6 Concluzii	100
5. Turbo coduri multi-binare	102
5.1 Interleaver-e pentru turbo coduri multi-binare	109
5.2 Performanțele turbo codurilor și a turbo codurilor multi-binare puncturate	113

6 Cuprins

5.3 Performanțele turbo codurilor multi-binare în canalele cu fading plat ...	120
5.3.1 Fading plat de tip Nakagami	120
5.3.2 Fading plat de tip Rayleigh	126
5.4 Concluzii	131
6. Contribuții și concluzii	132
Anexe	
Anexa A	134
Anexa B	137
Anexa C	138
Anexa D	148
Bibliografie	150

Listă abrevieri

APP	A Posteriori Probability
AWGN	Additive White Gaussian Noise
BER	Bit Error Rate
BLR	Bloc with Random Lines
BPSK	Binary Phase Shift Keying
BRL	Block Random in Line
CC	Convolutional Code
CCSDS	Consultative Committee for Space Data Systems
DVB-T	Digital Video Broadcasting-Terrestrial
FER	Frame Error Rate
ISI	Inter Symbol Interference
LDPC	Low-Density Parity-Check
LFSR	Linear Feedback Shift Register
LLR	Log Likelihoods Ratio
Log-MAP	Logarithm-MAP
LOS	Line-Of-Sight
MAP	Maximum A-Posteriori
Max-Log-MAP	Maximum-Logarithm-MAP
MBTC	Multi-Binary Turbo Code
NRNSC	Non-Recursive Non-Systematic Code
NRSC	Non-Recursive Systematic Code
psd	Power Spectral Density
QPSK	Quaternary Phase Shift Keying
RF	Radio Frequency
RNSC	Recursive Non-Systematic Code
RSC	Recursive Systematic Convolutional Code
SISO	Soft Input Soft Output
SNR	Signal to Noise Ratio
SOVA	Soft Output Viterbi Algorithm
TC	Turbo Code

8 Listă abrevieri

UMTS	Universal Mobile Telecommunications System
WCDMA	Wideband Code Division Multiple Access
WiMAX	Worldwide Interoperability for Microwave Access

Listă de tabele

Tabel 2.1 Spectrul ponderilor pentru codurile convoluționale de rată 1/2 și $K=3$.	31
Tabelul 3.1. Interleavere folosite.	44
Tabelul 4.1. $BER \cdot 10^{-9}$, în funcție de f și de tipul de interleaver folosit.	83
Tabelul 4.2. $FER \cdot 10^{-6}$, în funcție de f și de tipul de interleaver folosit.	83
Tabelul 4.3 Parametrii utilizați.	90
Tabelul 4.4 $BER \cdot 10^{-9}$ în funcție de f și K când s-a utilizat interleaver-ul S.	93
Tabelul 4.5 $BER \cdot 10^{-9}$ în funcție de f și K când s-a utilizat interleaver-ul BRL.	93
Tabelul 4.6 Impactul erorii de estimare a SNR-ului în algoritmul de turbo decodare pentru câteva valori ale lui m . Parametrul m pentru distribuția Nakagami reprezintă selectivitatea în timp a canalului.	100
Tabelul 5.1 Interleaver-ele utilizate.	113
Tabelul 5.2 Matrici de puncturare optime pentru rate 1/2, 3/5 și 2/3.	116
Tabelul 5.3 $BER \cdot 10^{-9}$ în funcție de f și în funcție de starea TCMB-ului.	128
Tabelul 5.4 $FER \cdot 10^{-6}$ în funcție de f și în funcție de starea TCMB-ului.	128

Listă de figuri

Fig. 2.1 Principiul de realizare a unui codor convoluțional	18
Fig. 2.2 Exemplu de codor convoluțional nerecursiv, nesistematic ($R=1/2$, $K=3$).	21
Fig. 2.3 CC recursiv sistematic: a) schema generală, b) exemplu ($R=1/2$, $K=3$).	22
Fig. 2.4 CC sistematic și nerecursiv, $R=1/2$, $K=3$, $G=1+D+D^2$.	23
Fig. 2.5 Trellis-ul codorului convoluțional din Fig. 2.2.	24
Fig. 2.6 Diagrama de stări a codorului din Fig. 2.2.	24
Fig. 2.7 Diagramele de stări pentru codoarele convoluționale: a) RSC; b) NRSC	25
Fig. 2.8 Cadrele folosite de un decodor convoluțional binar.	25
Fig. 2.9 Graful corespunzător diagramei de stări din Fig. 2.6.	27
Fig. 2.10 Distanța spectrală a codului RSC[1,5/7]: a) $\log(n(d)+1)/d$; b) $\log(w(d)+1)/d$.	32
Fig. 3.1 Schema unui turbo codor.	34
Fig. 3.2 Schema turbo decodorului.	36
Fig. 3.3 Coduri convoluționale concatenate paralel.	37
Fig. 3.4 Dispozitiv de întretesere aleator.	39
Fig. 3.5 Întretesere bloc.	40
Fig. 3.6 Schema interleaver-ului bloc aleator în linie, BRL.	42
Fig. 3.7 Schema interleaver-ului bloc cu linii aleatoare, BLR.	43
Fig. 3.8 a) Performanțele BER ale turbo-codurilor de rată $1/3$, [1, 15/13], pentru 5 interleaver-e cu $N \cong 400$.	45
Fig. 3.8 b) Performanțele BER ale turbo-codurilor de rată $1/3$, [1, 15/13], pentru alte 4 interleaver-e cu $N \cong 400$.	45
Fig. 3.8 c) Performanțele BER ale turbo-codurilor de rată $1/3$, [1, 15/13], pentru interleaver-e cu $N \cong 900$.	46
Fig. 3.8 d) Performanțele BER ale turbo-codurilor de rată $1/3$, [1, 15/13], pentru interleaver-e cu $N \cong 1800$.	46
Fig. 3.8 e) Performanțele BER ale turbo-codurilor de rată $1/3$, [1, 15/13], pentru interleaver-e cu $N \cong 3600$.	47
Fig. 3.9 a) Performanțele FER ale turbo-codurilor de rată $1/3$, [1, 15/13], pentru interleaver-e cu $N \cong 400$.	48
Fig. 3.9 b) Performanțele FER ale turbo-codurilor de rată $1/3$, [1, 15/13], pentru interleaver-e cu $N \cong 900$.	48
Fig. 3.9 c) Performanțele FER ale turbo-codurilor de rată $1/3$, [1, 15/13], pentru interleaver-e cu $N \cong 1800$.	49
Fig. 3.9 d) Performanțele FER ale turbo-codurilor de rată $1/3$, [1, 15/13], pentru interleaver-e cu $N \cong 3600$.	49
Fig. 3.10 Convergența supraviețuitorilor spre o cale unică în $t=n-4$ ($W=4$).	51
Fig. 3.11 Mărirea trellis-lui pentru un decodor Viterbi cu $R=1/2$.	53
Fig. 3.12 Trellis repetitiv, pentru decodor Viterbi, $R=1/2$.	53
Fig. 3.13 Decodor MAP pentru codul RSC, $K=3$.	56
Fig. 3.14 Calculul recursiv al lui $\alpha_k(0)$ și al lui $\beta_k(0)$.	59

Fig. 3.15 Sumarul operațiilor în algoritmul MAP	62
Fig. 3.16 Aproximarea rectangulară a funcției $f_c(x)$.	66
Fig. 3.17 Aproximarea liniară a funcției $f_c(x)$.	67
Fig. 3.18 Simularea TC-ului de rată 1/3, utilizând codul RSC [1, 5/7] și interleaver-ele S și pseudo-aleator, P .	68
Fig. 4.1 Funcția de autocorelație și densitatea spectrală de putere a zgomotului alb.	70
Fig. 4.2 a) O realizare particulară a zgomotului alb; b) Densitatea de probabilitate a zgomotului alb.	70
Fig. 4.3 a) Densitatea de probabilitate; b) Funcția de repartiție a probabilității unei variabile aleatoare gaussiene pentru două valori ale parametrului σ_w ($\sigma_1 > \sigma_0$).	72
Fig. 4.4 a) Densitatea de probabilitate; b) Funcția de repartiție pentru o variabilă aleatoare normal distribuită cu medie $u_0 \neq 0$.	73
Fig. 4.5 Manifestările canalului cu fading.	74
Fig. 4.6 Considerații asupra alocării legăturii pentru canale cu fading.	76
Fig. 4.7 a) Fading-ul de scară largă; b) fading-ul de scară redusă.	77
Fig. 4.8 Modele de simulare: a) sistem de transmisie necodat și b) sistem de transmisie turbo codat, cu modulație BPSK și canal cu fading plat și zgomot AWGN.	77
Fig. 4.9 Densitatea de probabilitate Rayleigh.	80
Fig. 4.10 Performanțele BER ale TC-urilor în canale cu fading plat de tip Rayleigh pentru codurile 5/7 și 15/13, utilizând: a) interleaver-ul S – I1784S29 și b) și interleaver-ul bloc aleator în linie I119BRL15.	81
Fig. 4.11 Performanțele: a) BER detaliate și b) FER, ale TC-urilor în canale cu fading plat de tip Rayleigh pentru codurile 5/7 și 15/13, utilizând interleaver-ul S – I1784S29 și interleaver-ul bloc aleator în linie I119BRL15.	82
Fig. 4.12 Performanțele: a) BER și b) FER, ale TC-urilor în canale cu fading plat de tip Rayleigh pentru codul 15/13 RSC, utilizând interleaver-ul S pentru diferite valori ale parametrului f .	84
Fig. 4.13 Performanțele: a) BER și b) FER, ale TC-urilor în canale cu fading plat de tip Rayleigh pentru codul 15/13 RSC, utilizând interleaver-ul bloc aleator în linie pentru diferite valori ale parametrului f .	85
Fig. 4.14 Canal Rice – model pentru simulare.	86
Fig. 4.15 Densitatea de probabilitate Rice de medii $A=0 \div 5$ și abatere standard $\sigma=1$.	87
Fig. 4.16 Performanța BER, funcție de parametrul K , pentru $SNR=10, 15$ și 20 dB, pentru canale Rice.	89
Fig. 4.17 Performanțele BER în canale cu fading plat de tip Rice pentru $K=0, 25, 50, 75$, și 100 [%].	90
Fig. 4.18 Performanțele BER în canale cu fading plat de tip Rice pentru $K=0, 25, 50, 75$, și 100 [%], pentru cazul în care s-a utilizat: a) interleaver-ul S și b) interleaver-ul bloc aleator în linie, BRL (Block Random in Line).	91
Fig. 4.19 Performanțele FER în canale cu fading plat de tip Rice pentru $K=0, 25, 50, 75$, și 100 [%], pentru cazul în care s-a utilizat: a) interleaver-ul S și b) interleaver-ul bloc aleator în linie, BRL.	92
Fig. 4.20 a) Densitatea de probabilitate a variabilei aleatoare x , pentru $m=2$.	96

Fig. 4.20 b) funcția de repartiție a probabilității, a variabilei aleatoare x , pentru $m=2$.	97
Fig. 4.21 Performanțele BER obținute prin simularea unui sistem de transmisie cu modulație BPSK în canale cu fading plat de tip Nakagami cu $m=2, 3, 4$ și 5 . Lui $m=1$ îi corespunde canalul Rayleigh și lui $m=\infty$ îi corespunde canalul fără fading.	97
Fig. 4.22 Performanțele BER obținute prin simularea unui sistem de transmisie turbo codat, cu modulație BPSK, în canale cu fading plat de tip Nakagami- m cu $m=2, 3, 4$ și 5 .	98
Fig. 4.23 Performanțele FER obținute prin simularea unui sistem de transmisie turbo codat, cu modulație BPSK, în canale cu fading plat de tip Nakagami- m cu $m=2, 3, 4$ și 5 .	99
Fig. 5.1 Turbo cod multi-binar. Schema generală.	102
Fig. 5.2. Performanțele: a) BER și b) FER, ale TCMB-urilor, utilizând codurile RSC de memorie 3, având matricea generatoare $\mathbf{G}=[13\ 15\ 11]_{10}$, în cazul folosirii decodărilor pe bit și pe simbol.	106
Fig. 5.3 Structura canonică a unui codor RSC multi-intrare, de rată $r/(r+1)$, construit pe baza registrului de deplasare cu reacție cu r intrări.	107
Fig. 5.4. Structura blocului turbo codat funcție de modul de închidere al trellis-urilor TCMB-ului: a) trellis-uri neînchise sau circulare, b) trellis C1 închis, trellis C2 neînchis și c) trellis-uri închise.	110
Fig. 5.5. Blocul de date transformat linie (vector): a) concatenarea liniilor, b) multiplexarea liniilor.	110
Fig. 5.6. Întrețeserea bloc cu ciclare diferențiată: a) ciclare diferențiată, b) întrețeserea bloc (rectangulară).	111
Fig. 5.7 Performanțele: a) BER și b) FER, ale TCMB-urilor, de rată 1/2 pentru codorul 13/11/15, folosind interleaver-ele menționate în Tabelul 5.1 și al TC-ului (puncturat), de rată 1/2, pentru codorul 15/13.	112
Fig. 5.8 a) Codor component RSC pentru TC; b) Codor component RSC pentru TCMB. Ambele codoare au polinomul de reacție $= (23)_8 = (10011)_2$ și polinomul de redundanță $= (35)_8 = (11101)_2$.	114
Fig. 5.9 a) Turbo codor cu codoare RSC de rată 1/2; b) Turbo codor multi-binar cu codoare RSC de rată 2/3.	115
Fig. 5.10 Performanța BER la rata 1/2 pentru TC, $\mathbf{H}=[11\ 1\ 12]$, și TCMB, $\mathbf{H}=[11\ 11\ 1\ 12]$, în cazul unei transmisii pe canal AWGN, funcție de SNR (dB).	117
Fig. 5.11 Performanța FER la rata 1/2 pentru TC, $\mathbf{H}=[11\ 1\ 12]$, și TCMB, $\mathbf{H}=[11\ 11\ 1\ 12]$, în cazul unei transmisii pe canal AWGN, funcție de SNR (dB).	117
Fig. 5.12 Performanța BER la rata 2/3 pentru TC, $\mathbf{H}=[11\ 1\ 12]$, și TCMB, $\mathbf{H}=[11\ 11\ 1\ 12]$, în cazul unei transmisii pe canal AWGN, funcție de SNR (dB).	118
Fig. 5.13 Performanța FER la rata 2/3 pentru TC, $\mathbf{H}=[11\ 1\ 12]$, și TCMB, $\mathbf{H}=[11\ 11\ 1\ 12]$, în cazul unei transmisii pe canal AWGN, funcție de SNR (dB).	118
Fig. 5.14 Performanța BER la rata 3/5 pentru TC, $\mathbf{H}=[11\ 1\ 12]$, și TCMB, $\mathbf{H}=[11\ 11\ 1\ 12]$, în cazul unei transmisii pe canal AWGN, funcție de SNR (dB).	119

- Fig. 5.15 Performanța FER la rata 3/5 pentru TC, $\mathbf{H}=[11\ 1\ 12]$, și TCMB, $\mathbf{H}=[11\ 11\ 1\ 12]$, în cazul unei transmisii pe canal AWGN, funcție de SNR (dB). 119
- Fig. 5.16 Schema unui codor duo-binar cu 16 stări, cu $\mathbf{H}=[11\ 11\ 1\ 12]_{10}$. 120
- Fig. 5.17 O comparație între două implementări ale algoritmului de decodare la TCMB pentru un canal cu fading Nakagami-5: algoritmul MAP și aproximarea Max-Log-MAP. 122
- Fig. 5.18 Performanțele TCMB-ului în cazul transmisiei pe un canal cu fading plat de tip Nakagami, utilizând algoritmul de decodare Max-Log-MAP: a) BER, b) FER, sunt trasate în funcție de SNR pentru diferite valori ale parametrului m . Deasemenea, pentru câteva dintre aceste valori ale parametrului m , sunt reprezentate pragurile corespunzătoare limitei Shannon pe canale cu fading plat de tip Nakagami (pentru intrări gaussiene, cât și pentru intrări binare ($m=1$ și $m=\infty$)). 124
- Fig. 5.19 Performanțele a) BER și b) FER pentru TCMB-urile de memorie 4 și pentru TC-urile clasice de memorie 3 (13/15) și de memorie 4 (23/31), pe canale cu fading plat de tip Nakagami (parametrul fading-ului $m=5$), utilizând algoritmul de decodare MAP. Deasemenea este reprezentat pragul corespunzător limitei Shannon cu intrări gaussiene pe canale cu fading plat de tip Nakagami. 125
- Fig. 5.20 Schema unui codor duo-binar cu 8 stări, cu $\mathbf{H}=[6\ 7\ 1\ 5]_{10}$. 126
- Fig. 5.21 Performanțele BER pentru cazul necodat și codat, cu TCMB-uri de memorie 3, respectiv de memorie 4, pe canale cu fading plat de tip Rayleigh. BER este reprezentat în funcție de media SNR (dB). Algoritmul de decodare implementat este algoritmul Max-Log-MAP. 128
- Fig. 5.22 Performanțele a) BER și b) FER ale TC-urilor duo-binare cu 8 stări și cu 16 stări pentru diferite valori ale factorului f . S-a folosit algoritmul de decodare MAP și modulația QPSK. 129
- Fig. 5.23 Performanțele a) BER și b) FER. O comparație între două implementări ale algoritmului de decodare ale TCMB-urilor (cu $r=2$), pe canal cu fading plat de tip Rayleigh: algoritmul MAP și aproximarea sa Max-Log-MAP. TCMB-urile sunt bazate pe codurile RSC de memorie 3 (linie întreruptă) și de memorie 4 (linie continuă). Lungimea cuvântului de cod este egală cu $2 \cdot 752 = 1504$ biți. 130

1. INTRODUCERE

Noi servicii de comunicații și tehnologia informației apar aproape zilnic și cererile de a avea o capacitate de comunicație și rate de transmisie din ce în ce mai mari continuă să crească. Acest progres spectaculos al comunicațiilor se datorează, în mare măsură, creșterii performanțelor codurilor și reducerii costului tehnologiei.

Calitatea unui sistem numeric de comunicații este în general evaluată prin probabilitatea de eroare a simbolurilor recepționate, sau rata erorii pe bit, BER (Bit Error Rate).

Această probabilitate depinde de raportul semnal pe zgomot al mesajului, SNR (Signal to Noise Ratio), din canalul de comunicații. Cu cât SNR este mai mare, BER este mai mic. Pentru a ameliora calitatea unui sistem de comunicații numerice este necesară creșterea acestui raport.

O soluție de a scădea BER, fără creșterea SNR este codarea mesajului de transmis. Operația de codare constă în a adăuga la mesajul de transmis simboluri, numite de redundanță, după o lege dată. Necesitatea de a se introduce redundanță în mesaj, pentru a se proteja contra erorilor de transmisie, este demonstrată în cadrul teoriei informației, [BOR99].

Pentru un mesaj lipsit de redundanță, fiecare simbol este esențial și astfel orice eroare de transmisie conduce la o pierdere ireversibilă de informație. În schimb, introducerea de simboluri de redundanță va corela simbolurile mesajului codificat. Astfel, în anumite condiții, simbolurile eronate în cursul transmisiei vor modifica legea de codare utilizată la emisie, ceea ce va permite detectarea, apoi eventual, corectarea erorilor.

În celebra sa lucrare din 1948, [SHA48], Shannon a demonstrat că o comunicație sigură este posibilă printr-un canal oricât de zgomotos dacă este îndeplinită condiția ca rata de transmisie să fie mai mică decât capacitatea canalului. Totuși, Shannon nu a propus soluții explicite de codare a canalului pentru implementări practice. Astfel, vreme de aproape 60 de ani, cercetătorii au creat coduri, limitate ca performanță de complexitatea decodării și de complexitatea canalului. În urma cercetărilor efectuate s-au impus în principal două mari familii de coduri, codurile bloc și codurile convoluționale [PRO00].

În structura oricărui receptor digital există un detector. În prezent se utilizează două tipuri de detectoare, hard și soft. Acestea din urmă utilizează valori ale unui raport de plauzibilitate. Prima soluție, care conduce la o pierdere ireversibilă de informație pentru decodor, a dat naștere la algoritmi de decodare de tip algebric, bine adaptați la structura codurilor bloc [THI93]. A doua soluție, permite decodorului să exploateze cel mai bine informația disponibilă, conduce la algoritmi de decodare de tip probabilist, utilizați în special pentru codurile convoluționale [FAN63, VIT67].

Pentru a atinge cel mai bun rezultat din punct de vedere al BER-ului și al complexității de implementare, în tehnicile de codare a informației au fost imaginat diferite asocieri de coduri elementare. Una dintre aceste asocieri constă în concatenarea a două coduri elementare astfel încât simbolurile de la ieșirea primului codor, numit exterior, să fie aplicate la intrarea celui de al doilea, numit interior. În general concatenarea codurilor este realizată prin asocierea unui cod convoluțional, interior, cu un cod bloc, exterior. În cazul concatenării a două coduri convoluționale,

[HAH89], pentru a atinge cel mai bun rezultat în urma concatenării, este necesar ca decodorul interior să poată furniza decizii ponderate decodorului exterior.

Decodarea codurilor convoluționale este în general realizată pornind de la algoritmul Viterbi care furnizează decizii hard [OMU69, VIT71]. Acest algoritm nu poate fi utilizat decât pentru decodarea codului interior. Au fost propuse mai multe soluții pentru a rezolva problema decodării codurilor convoluționale cu decizii soft la ieșire. Bahl ș.a. [BCJR74] au propus să se decodeze codurile convoluționale determinând simbolul cel mai plauzibil. Algoritmul propus se numește MAP (Maximum A-Posteriori). Acest algoritm este mai complex comparativ cu algoritmul Viterbi. Au fost propuse mai multe versiuni ale algoritmului Viterbi, ce permit furnizarea deciziilor ponderate la ieșirea decodorului, de diferiți autori printre care și Berrou ș.a. [BAAF93]. Totuși performanțele unei astfel de soluții sunt încă destul de depărtate de cele previzionate de Shannon.

O creștere semnificativă a performanțelor poate fi obținută folosind decodarea "turbo", prin optimizarea subsistemului din structura receptorului, format din decodoare și detector. Această optimizare presupune aducerea unei părți din informația de la ieșirea detectorului la intrarea primului decodor. Aceasta este așa numita informație extrinsecă. Sistemul obținut se numește turbo decodor, [BGT93]. Funcționarea acestuia este iterativă, aportul informației extrinseci la scăderea BER-ului diminuându-se odată cu creșterea numărului de iterații.

Din momentul apariției, turbo codarea a evoluat într-un ritm fără precedent, datorită eforturilor intense depuse de cercetătorii din domeniu. Ca urmare, turbo codurile au fost introduse și în standarde, ca de exemplu standardul pentru comunicații mobile de generația a treia (3G) [HOT04]. În sistemele video de difuzare, unde întârzierea asociată sistemului este mai puțin critică decât în sistemele interactive, câștigurile în performanță care pot fi atinse sunt și mai impresionante.

Conceptul de turbo codare a fost introdus în 1993, de către Berrou, Glavieux și Thitimajashima, care au raportat rezultate excelente de câștig de codare [BGT93], apropiate de limita lui Shannon. Secvența de informație este codată de două ori, având un interleaver (dispozitiv de întretesere) între cele două codoare (concatenate paralel) care servește la realizarea a două secvențe de date aproximativ independente statistic una față de cealaltă. Cel mai des sunt utilizate codoarele convoluționale recursive sistematice, RSC (Recursive Systematic Convolutional Code). Fiecare codor RSC, din structura unui turbo codor, produce o ieșire "sistematică", care conține secvența de informație originală, precum și o secvență de informație de paritate. Cele două secvențe de paritate pot fi apoi puncturate [VUY01, KBDN07], înainte de a fi transmise împreună cu secvența de informație originală. Această puncturare a informației de paritate permite o gamă largă a ratelor de codare și deseori este transmisă jumătate din informația de paritate de la fiecare codor. Puncturarea conduce la creșterea ratei de codare, obținându-se de exemplu o rată de codare a TC-ului egală cu $1/2$, comparativ cu cazul nepuncturat, unde rata de codare este $1/3$.

În prezent turbo codurile se utilizează în cele mai moderne sisteme de comunicații. Un exemplu potrivit este cel al sistemelor fără fir (wireless). În cazul acestor sisteme canalele sunt variante în timp, deoarece propagarea se face pe mai multe căi. Acestea sunt numite canale cu fading. Majoritatea lucrărilor despre turbo coduri din literatura de specialitate tratează cazul canalului AWGN (Additive White Gaussian Noise). Scopul acestei teze este studiul sistematic al comportării turbo codurilor pe canale cu fading plat. Diferența față de abordările anterioare este dată tocmai de acest caracter sistematic al studiului.

Capitolele 2 și 3 reprezintă o introducere în teoria turbo codurilor, pe care am considerat-o necesară pentru ca teza să aibă un caracter unitar. Capitolul 4 reprezintă partea centrală a tezei. Capitolul 5 se referă la modul în care acționează ultima generație de turbo coduri, cele multi-binare, în cazul canalelor cu fading plat.

Teza în continuare este structurată după cum urmează.

În capitolul 2 am făcut o prezentare a codurile convoluționale, a codării, respectiv decodării acestora. Deasemenea, am simulat și am analizat spectrul ponderilor pentru diferite coduri convoluționale, de memorie 2.

În prima parte a capitolului 3 am descris, pe scurt, turbo codul. În continuare, am prezentat câteva tipuri de interleaver-e. Apoi, am propus, simulat și analizat două noi tipuri de interleaver-e, și anume: interleaver-ul bloc aleator în linie și interleaver-ul bloc cu linii aleatoare. Tot în cadrul acestui capitol am făcut o prezentare a algoritmilor de decodare: Viterbi, MAP, Max-Log-MAP (Maximum-Logarithm-MAP) și Log-MAP (Logarithm-MAP). La sfârșitul capitolului am arătat performanțele algoritmilor de decodare MAP, Max-Log-MAP și Log-MAP, prin intermediul rezultatelor pe care le-am obținut cu ajutorul simulărilor.

La începutul capitolului 4 am prezentat câteva noțiuni legate de propagarea radio în comunicațiile mobile și am prezentat pe scurt o clasificare a manifestărilor fading-ului în canale. Apoi, am analizat și simulat comportarea turbo codurilor în diferite canale cu fading plat de tip: Rayleigh, Rice și Nakagami. Deasemenea, am făcut o estimare a canalului, necesară pentru construcția coeficientului L_c , utilizat în algoritmul de decodare MAP.

În capitolul 5 am reprezentat, structurile unui turbo cod multi-binar cât și codoarelor sale componente multi-intrare RSC. Apoi, am făcut o analiză a celor două metode de decodare, decodare pe bit și decodare pe simbol, ce pot fi realizate în cadrul turbo codurilor multi-binare. Deasemenea, am analizat performanțele BER și FER (Frame Error Rate) ale turbo codurilor puncturate și a turbo codurilor multi-binare puncturate cât și cele ale turbo codurilor multi-binare în canalele AWGN și cu fading plat (Rayleigh și Nakagami).

În capitolul 6 am prezentat principalele contribuții și concluzii pe care le-am avut, respectiv, obținut în cadrul acestei lucrări.

2. CODURI CONVOLUȚIONALE

Codurile convoluționale (CC) se utilizează frecvent în aplicații cum sunt: comunicațiile spațiale și prin satelit, telefonie celulară, televiziunea digitală, Digital Video Broadcasting Terrestrial (DVB-T), etc., [VUY01]. Răspândirea lor este datorată structurii simple și implementării facile a metodelor de decodare.

2.1. Principiul codării convoluționale

Codurile convoluționale introduse în 1954 de P. Elias [ELI54] reprezintă o clasă de coduri corectoare de erori având o mare aplicabilitate practică. Datorită simplității lor și posibilității de a utiliza algoritmi de decodare de tip Soft Input Soft Output (SISO), CC-urile sunt cele mai utilizate coduri componente în turbo coduri.

În cazul CC-urilor, fiecare bloc de n simboluri binare de la ieșirea codorului depinde atât de blocul de k simboluri binare prezent la intrarea sa, la momentul considerat, cât și de m blocuri precedente [THI93]. În consecință CC-urile introduc un efect de memorie de ordinul m .

Lungimea de constrângere a codului convoluțional, exprimată în biți ai mesajului, este definită ca numărul de deplasări după care un bit al mesajului poate influența ieșirea decodorului. Într-un codor având un registru de deplasare cu m celule, și memoria codorului egală cu m , sunt necesare $K=m+1$ deplasări pentru ca un bit al mesajului să intre în registrul de deplasare și să iasă în final. Astfel lungimea de constrângere a codorului este K [HAY00].

Principiul codorului convoluțional este prezentat în Fig. 2.1. Codorul este constituit:

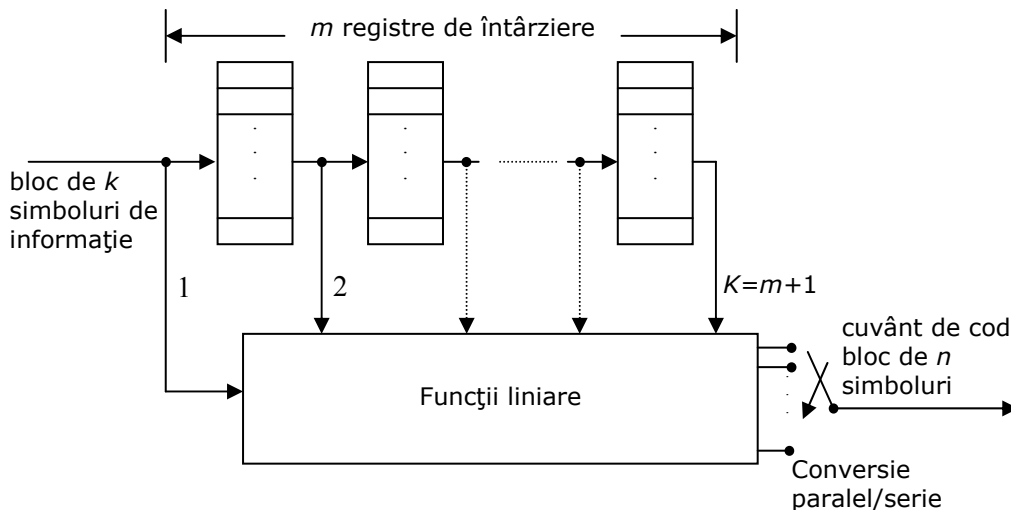


Fig. 2.1 Principiul de realizare a unui codor convoluțional.

- dintr-un sistem de m registre de întârziere, fiecare având o capacitate de k biți, care memorează cele m blocuri de k simboluri de informație,
- dintr-o mulțime de funcții liniare, ce generează blocurile de n simboluri de la ieșire,
- dintr-un convertor paralel-serie.

Raportul $R=k/n$ se numește rată de codare.

Un cod convoluțional de rată R este o aplicație de la mulțimea matricilor (binare) cu un număr de k linii și număr infinit de coloane către mulțimea matricilor (binare) cu un număr de n linii și număr infinit de coloane, unde $n>k$ [THI93].

$$\mathbf{C} : \mathbf{M}_{k \times \infty} \rightarrow \mathbf{M}_{n \times \infty} \quad (2.1)$$

Astfel, prin transformarea \mathbf{C} , fiecărei matrici $\mathbf{I} \in \mathbf{M}_{k \times \infty}$, de forma :

$$\mathbf{I} = \begin{bmatrix} i_{01} & i_{11} & i_{21} & \dots & i_{j1} & \dots \\ i_{02} & i_{12} & i_{22} & \dots & i_{j2} & \dots \\ \dots & \dots & \dots & \dots & \dots & \dots \\ i_{0k} & i_{1k} & i_{2k} & \dots & i_{jk} & \dots \end{bmatrix} \quad i_{js} \in \{0,1\} \quad \forall j = \overline{0, \infty}, \quad s = \overline{1, k} \quad (2.2)$$

și se atașează o matrice $\mathbf{V} \in \mathbf{M}_{n \times \infty}$, de forma:

$$\mathbf{V} = \begin{bmatrix} a_{01} & a_{11} & a_{21} & \dots & a_{j1} & \dots \\ a_{02} & a_{12} & a_{22} & \dots & a_{j2} & \dots \\ \dots & \dots & \dots & \dots & \dots & \dots \\ a_{0k} & a_{1k} & a_{2k} & \dots & a_{jk} & \dots \\ \dots & \dots & \dots & \dots & \dots & \dots \\ a_{0n} & a_{1n} & a_{2n} & \dots & a_{jn} & \dots \end{bmatrix} \quad a_{js} \in \{0,1\} \quad \forall j = \overline{0, \infty}, \quad s = \overline{1, n} \quad (2.3)$$

Matricea \mathbf{I} conține biții de informație, în ordinea $i_{01} i_{02} \dots i_{0k} i_{11} \dots$, iar matricea \mathbf{V} secvența codată : $a_{01} a_{02} \dots a_{0n} a_{11} \dots$. Dacă $a_{js} \equiv i_{js}$ pentru orice j pozitiv și pentru orice $s = 1 + k$, atunci codul se numește sistematic. Făcând apel la o reprezentare polinomială, matricile \mathbf{I} și \mathbf{V} se pot scrie ca, [BAK04a]:

$$\mathbf{I}(D) = \begin{bmatrix} \sum_{s=0}^{\infty} i_{s1} D^s \\ \sum_{s=0}^{\infty} i_{s2} D^s \\ \vdots \\ \sum_{s=0}^{\infty} i_{sk} D^s \end{bmatrix}, \quad \mathbf{V}(D) = \begin{bmatrix} \sum_{s=0}^{\infty} a_{s1} D^s \\ \sum_{s=0}^{\infty} a_{s2} D^s \\ \vdots \\ \sum_{s=0}^{\infty} a_{sk} D^s \\ \vdots \\ \sum_{s=0}^{\infty} a_{sn} D^s \end{bmatrix}. \quad (2.4)$$

Cu aceste notații, relația de codare, poate fi scrisă astfel:

$$\mathbf{V}(D) = \mathbf{G}(D) \cdot \mathbf{I}(D), \quad (2.5)$$

unde $\mathbf{G}(D)$ se numește matricea generatoare a codului și are forma:

$$\mathbf{G}(D) = \begin{bmatrix} g_{11}(D) & g_{12}(D) & \cdots & g_{1k}(D) \\ \cdots & \cdots & \cdots & \cdots \\ g_{n1}(D) & g_{n2}(D) & \cdots & g_{nk}(D) \end{bmatrix}. \quad (2.6)$$

2.2. Tipuri de coduri convoluționale

În funcție de forma polinoamelor $g_{js}(D)$ și a criteriului adoptat, codurile convoluționale pot fi clasificate ca și, [FOR90, HLY02]:

1. Coduri sistematice, respectiv nesistematice.

1.a) Dacă primele k linii din $\mathbf{G}(D)$ formează matricea unitate de ordinul k , \mathbf{I}_k , atunci codurile sunt sistematice. La codurile sistematice, simbolurile de informație sunt plasate toate, fie la începutul fie la sfârșitul cuvântului de cod. Astfel, dacă cele k simboluri de informație, prezente la intrarea codorului din Fig. 2.1, sunt efectiv emise, adică se găsesc în mod explicit în blocul de n simboluri de la ieșirea codorului pe primele k poziții, codul se numește cod sistematic [FOR70]. Altfel spus, în cazul codurilor sistematice biții sau simbolurile de informație originale constituie parte din cuvântul de cod codat și astfel, ei pot fi recunoscuți în mod explicit la ieșirea codorului [HLY02].

1.b) Pentru codurile nesistematice, biții din \mathbf{V} sunt combinații liniare ale biților din \mathbf{I} , neexistând biți de informație și de control ca și în cazul precedent.

2. Coduri recursive, respectiv nerecursive.

2.a) Dacă toate polinoamele generatoare care compun $\mathbf{G}(D)$ sunt finite, atunci codul rezultat este nerecursiv.

2.b) Dacă polinoamele generatoare $g_{js}(D)$ pot fi scrise sub forma:

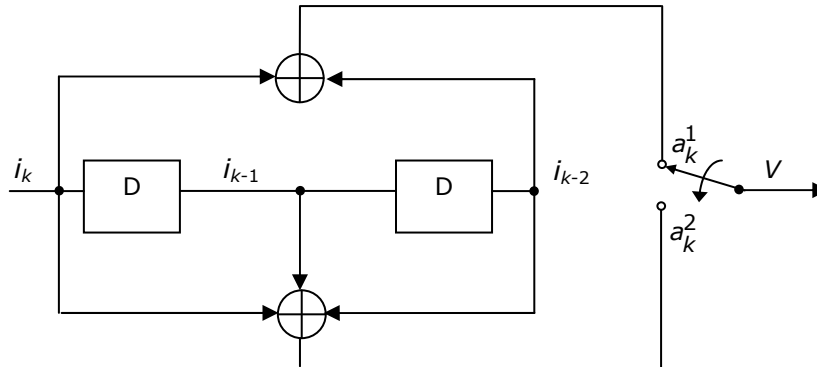
$$g_{js}(D) = \frac{a_{js}(D)}{b_{js}(D)}, \quad (2.7)$$

unde polinoamele a_{js} și b_{js} sunt finite, și dacă există cel puțin un polinom $b_{js}(D) \neq 1$, atunci codul este recursiv.

Lungimea de constrângere este unul dintre parametri importanți ai codurilor convoluționale. O altă definiție a sa este dată în relația de mai jos, [BAK04a]:

$$K = 1 + \max_{j,s} \text{grad}\{a_{j,s}(D), b_{j,s}(D)\}. \quad (2.8)$$

Pentru a ilustra codurile convoluționale nerecursive și nesistematice, în Fig. 2.2 se prezintă un exemplu de codor convoluțional de rată $R=1/2$ și de lungime de constrângere $K=m+1=3$. Intrarea sa este constituită din blocurile de $k=1$ simbol de informație și ieșirea sa de blocurile de $n=2$ simboluri codate.

Fig. 2.2 Exemplu de codor convoluțional nerecursiv, nesistematic ($R=1/2$, $K=3$).

Caracterul convoluțional al codurilor provine din faptul că fiecare ieșire a codorului este egală cu produsul de convoluție dintre șirul de simboluri prezente la intrarea codorului și răspunsul codorului, definit prin polinoamele sale generatoare. În codorul din Fig. 2.2 ieșirile a_k^i ; $i=1, 2$ sunt multiplexate de către un comutator, obținându-se o singură secvență de cod, \mathbf{V} . Relațiile de calcul ale acestor secvențe de ieșire sunt:

$$a_k^1 = \sum_{j=0}^2 i_{k-j} g_j^1, \quad (2.9)$$

$$a_k^2 = \sum_{j=0}^2 i_{k-j} g_j^2. \quad (2.10)$$

Cele două secvențe generatoare sunt:

$$\begin{aligned} g^1 &= [g_0^1, g_1^1, g_2^1] = [1, 0, 1] \\ g^2 &= [g_0^2, g_1^2, g_2^2] = [1, 1, 1] \end{aligned} \quad (2.11)$$

Ieșirile codorului fiind egale cu o combinație liniară a simbolurilor de informație, codul este liniar. Codurile convoluționale sunt de asemenea definite pornind de la polinoamele lor generatoare exprimate în funcție de variabila D (delay-întârziere) echivalentă cu variabila Z^{-1} a transformatei Z [FOR70]. Considerând tot exemplul din Fig. 2.2, polinoamele generatoare ale acestui cod au expresiile:

$$\begin{aligned} g^1 &\rightarrow G^1(D) = g_0^1 + g_1^1 D + g_2^1 D^2 \\ g^2 &\rightarrow G^2(D) = g_0^2 + g_1^2 D + g_2^2 D^2, \end{aligned} \quad (2.12)$$

și:

$$\begin{aligned} G^1(D) &= 1 + D^2 \\ G^2(D) &= 1 + D + D^2 \end{aligned} \quad (2.13)$$

rezultând matricea generatoare $G = [1 + D^2, 1 + D + D^2]$.

În general polinoamele generatoare ale codorului se exprimă în octal și astfel, pentru cazul din Fig. 2.2, avem:

$$\begin{aligned} G^1 &= [1 \ 0 \ 1] = 5(\text{în octal}) \\ G^2 &= [1 \ 1 \ 1] = 7(\text{în octal}) \end{aligned} \quad (2.14)$$

CC-urile nesistematice pot fi catastrofice. Codurile *catastrofice* sunt codurile pentru care un număr finit de erori în canalul de transmisie poate da naștere unui număr infinit de erori la ieșirea din decodor. O condiție necesară și suficientă, [BOR99], pentru apariția erorilor catastrofice în cazul codurilor cu $R=1/n$ este ca polinoamele generatoare să aibă factor comun.

Astfel, dacă un cod de rată $R=1/2$ are polinoamele generatoare de forma $G^1(D)=1+D$ și $G^2(D)=1+D^2$, el este un cod catastrofic, deoarece polinoamele generatoare îl au ca și factor comun pe $1+D$. Deoarece se face o sumare modulo doi, putem scrie: $(1+D) \cdot (1+D) = 1+D+D+D^2 = 1+D^2$. După cum se observă din matricea generatoare, \mathbf{G} , codorul din Fig. 2.2 nu generează un cod catastrofic.

Pentru codurile sistematice, există cel puțin o secvență binară de informație de pondere infinită, care dă naștere unei secvențe binare de pondere finită la ieșirea codorului. În consecință, codurile sistematice nu pot fi catastrofice.

În Fig. 2.3 a) se prezintă schema generală a unui codor RSC, iar în Fig. 2.3 b) un caz particular al acesteia.

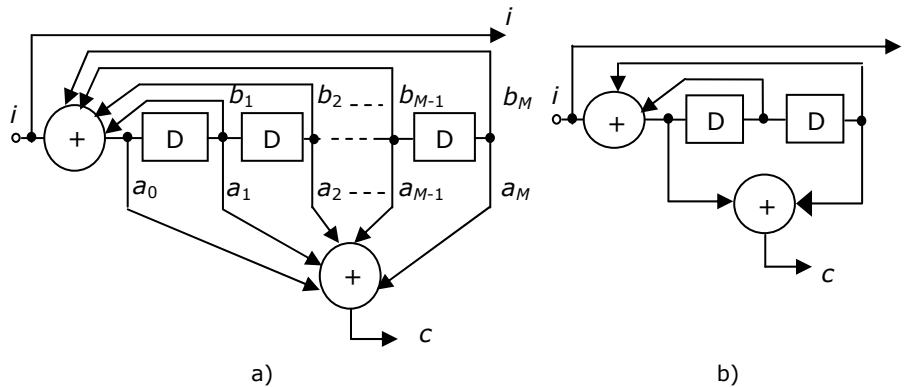


Fig. 2.3 CC recursiv sistematic: a) schema generală, b) exemplu ($R=1/2, K=3$).

În exemplul considerat matricea generatoare este de forma:

$$\mathbf{G} = \left[1, \frac{1 + D^2}{1 + D + D^2} \right]. \quad (2.15)$$

În figura următoare este prezentat un exemplu de codor sistematic și nerecursiv, NRSC (Non-Recursive Systematic Code), considerându-se rata $R=1/2$ și lungimea de constrângere $K=3$, [BOR99]:

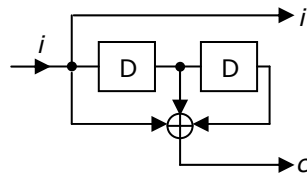


Fig. 2.4 CC sistematic și nerecursiv, $R=1/2$, $K=3$, $G=1+D+D^2$.

2.3. Reprezentarea grafică a codurilor convoluționale

Proiectarea algoritmilor de decodare poate fi simplificată dacă se utilizează reprezentări grafice ale CC-urilor. Reprezentarea cea mai uzuală și mai bine adaptată este, în mod incontestabil, reprezentarea grafică sub forma unui trellis sau sub forma unei diagrame de stări. În continuare se prezintă diagrama trellis și diagrama de stări pentru codorul convoluțional nesistematic din Fig. 2.2, [THI93].

Diagrama trellis

Fiecare bloc de $n=2$ simboluri de la ieșirea acestui codor depinde de blocul de $k=1$ simbol prezent la intrarea sa dar și de $m=2$ blocuri de k simboluri conținute în memoria sa. Aceste $mk=2$ simboluri definesc starea codorului. Se notează cu $S_0=(00)$, $S_1=(01)$, $S_2=(10)$ și $S_3=(11)$, cele patru stări posibile ale codorului din Fig. 2.2. Oricare ar fi starea inițială a codorului, după $m+1=3$ întârzieri la intrarea codorului, toate stările au fost atinse.

Funcționarea codorului poate fi explicată ținând seama doar de stările sale și de tranzițiile dintre acestea, numite ramuri (ramificații, brațe). Diagrama trellis astfel obținută este reprezentată în Fig. 2.5, pentru codorul convoluțional din Fig. 2.2, presupunând ipoteza că starea sa inițială era $S_0=(00)$.

Ramurile reprezentate prin linii punctate corespund prezenței unui simbol de informație egal cu 1, la intrarea codorului, și ramurile reprezentate prin linii pline, unui simbol de informație egal cu 0. Fiecărei ramuri i s-a asociat valoarea cuplului binar disponibil la ieșirea codorului.

După $m+1$ întârzieri, oricare ar fi starea inițială a codorului, trellis-ul se repetă. Din fiecare nod pleacă 2^k ramuri (în cazul de față sunt două ramuri) și în fiecare nod converg 2^k ramuri.

Pornind de la starea $S_0=(00)$ în momentul $t=0$, de exemplu, vedem că există patru căi care permit atingerea stării $S_0=(00)$ în momentul $t=4$.

00 00 00 00 → calea 1
 00 11 01 11 → calea 2
 11 10 10 11 → calea 3
 11 01 11 00 → calea 4

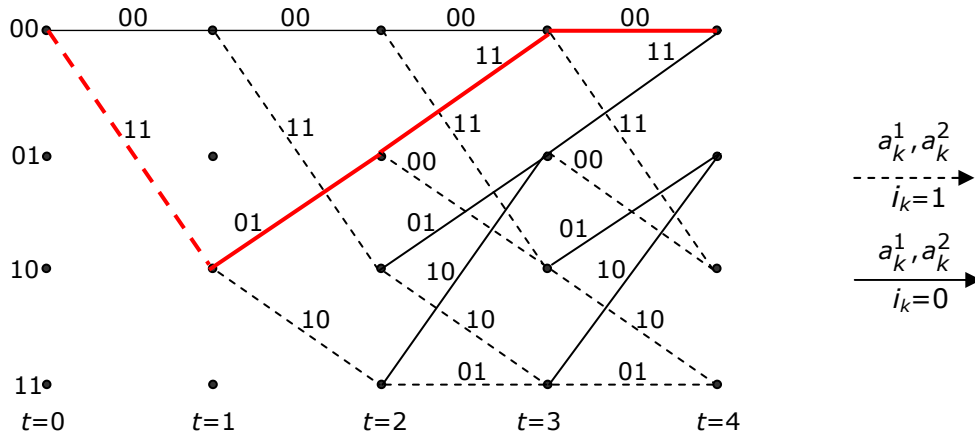


Fig. 2.5 Trellis-ul codorului convoluțional din Fig. 2.2.

Diagrama de stări

Diagrama de stări este o altă reprezentare a funcționării unui codor convoluțional, în care timpul nu apare în mod explicit. Această diagramă, care se poate deduce din trellis, nu reține decât diferitele stări ale codorului și felul în care acestea comunică.

În Fig. 2.6 s-a reprezentat diagrama de stări asociată codorului convoluțional din Fig. 2.2. Diagrama de stări permite evaluarea funcției de transfer a codorului, care va fi utilizată pentru calculul performanțelor codului.

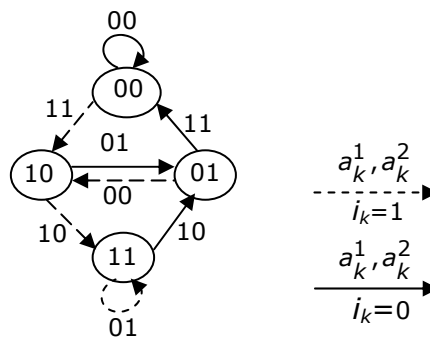


Fig. 2.6 Diagrama de stări a codorului din Fig. 2.2.

Diagramele de stări corespunzătoare codurilor convoluționale din Fig. 2.3 b) și Fig. 2.4 sunt prezentate în Fig. 2.7.

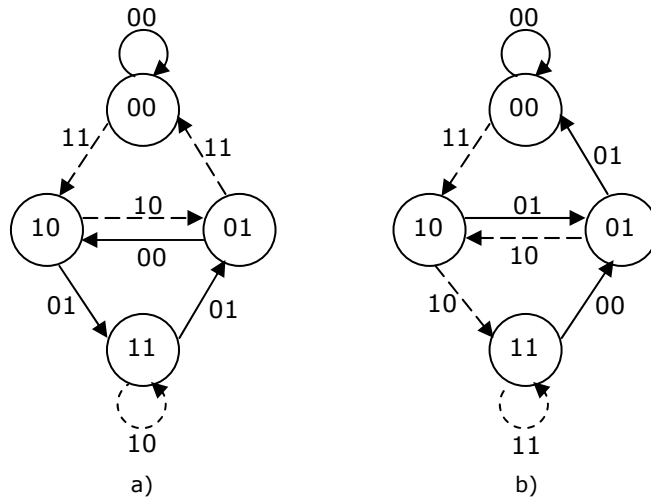


Fig. 2.7 Diagramele de stări pentru codoarele convoluționale: a) RSC; b) NRSC.

2.4. Distanța liberă

La fel ca pentru codurile bloc, distanța minimă a CC este un parametru fundamental, deoarece determină capacitatea de control a erorii când numărul mediu de erori raportat la numărul de cuvinte de cod este mic. Oricum, procesul de estimare a distanței este mai complex în cazul codului convoluțional decât în cazul unui cod bloc deoarece această operație depinde de tipul de decodor folosit. Mai precis depinde de numărul de cadre, m , folosit de decodor [WAD00] (Fig. 2.8).

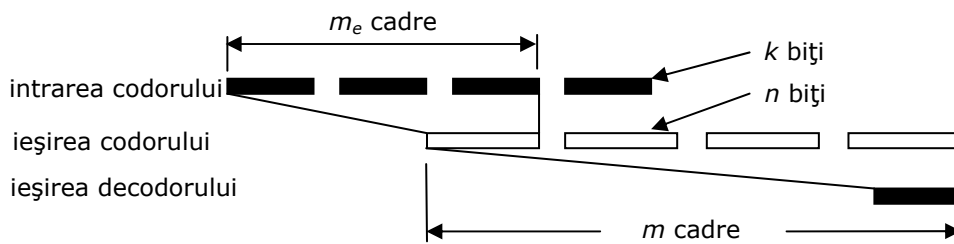


Fig. 2.8 Cadrele folosite de un decodor convoluțional binar.

Distanța minimă d_m , de ordin m , a unui CC este minimul distanțelor Hamming dintre toate perechile posibile ale secvențelor codate sau ale cuvintelor de cod de lungime de m cadre (sau ramuri) ce diferă în cadrul lor inițial:

$$d_m = \min d_H(u_m, v_m), \quad u_1 \neq v_1, \quad (2.16)$$

unde, primele m cadre ale celor două cuvinte de cod sunt notate u_m și v_m .

Fără a pierde din generalitate, relația (2.16) se poate reduce la găsirea distanței Hamming minime de la cuvântul specific la oricare din celelalte cuvinte de cod. În consecință, distanța minimă a unui CC reprezintă numărul minim de "1" din toate cuvintele de cod care nu au toți biții de informație egali cu zero.

Conform teoremei care afirmă că distanța minimă, d , a unui cod liniar este egală cu ponderea Hamming minimă a vectorilor nenuli, rezultă că pentru determinarea distanței minime trebuie identificată pe trellis-ul asociat codului studiat, calea cu ponderea cea mai mică. Prin ponderea unei căi se înțelege numărul de simboluri de „1” care apar pe acea cale. De exemplu, folosind Fig. 2.2, se identifică calea de pondere minimă (este cea marcată cu linie îngroșată).

Odată ce d_m a fost determinată putem folosi același mecanism de control al erorii ca și pentru codurile bloc. CC poate corecta orice structură de t erori sau mai puțin de t erori din oricare m cadre adiacente, dacă este satisfăcută condiția:

$$d_m \geq 2t + 1. \quad (2.17)$$

Un decodor convoluțional poate corecta orice structură (izolată) de t sau mai puține erori care apar într-o secvență continuă de mn simboluri. După cum am menționat mai sus distanța minimă, d_m , depinde de tipul de decodare utilizat.

De exemplu, tehnicile simple de *decodare cu logică de prag* au o memorie de decodare egală cu doar o lungime de constrângere. Cea mai importantă distanță măsurată pentru codurile convoluționale corespunde lui $m \rightarrow \infty$ și este denumită d_{free} , distanță liberă. Această distanță este potrivită pentru metodele de decodare probabilistice, cum ar fi decodarea Viterbi și decodarea secvențială, din moment ce aici memoria de decodare este în principiu nelimitată. Pentru a găsi distanța liberă în diagrama trellis din Fig. 2.5, se examinează toate căile care pleacă din starea zero și se întorc la această stare. Calea cu ponderea Hamming minimă ne dă d_{free} , care în cazul codului din Fig. 2.2 rezultă a fi $d_{free}=5$, corespunzător cuvântului de cod: 110111.

2.5. Funcția de transfer asociată unui cod convoluțional

Funcția de transfer a codului este utilizată pentru a determina distribuția ponderilor Hamming ale diferitelor căi care diverg de la calea nulă și apoi converg din nou spre ea. Ea se poate calcula pornind de la diagrama de stări, scindând starea $S_0=(00)$ în două stări (starea S_i și starea S_j), apoi aducând pe fiecare ramură un cuplu $D^i B^j$ unde exponenții acestor argumente indică: pentru D - ponderea secvenței emise, pentru B - ponderea secvenței de intrare corespunzătoare (0 sau 1). Graficul atașat diagramei de stări din Fig. 2.6 este prezentat în Fig. 2.9.

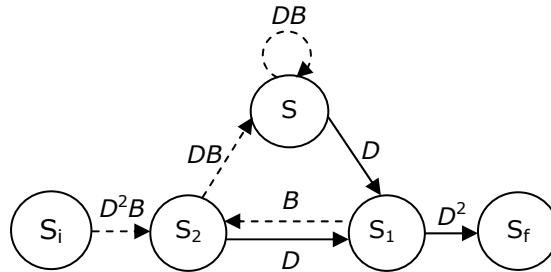


Fig. 2.9 Graful corespunzător diagramei de stări din Fig. 2.6.

Prezența lui B semnifică faptul că la intrarea codului bitul de informație, i , a fost 1. Gradul lui D ne dă ponderea secvenței emise, adică numărul de unu-ri din această secvență.

Funcția de transfer $T(D, B)$ a codului este definită de următoarea relație:

$$T(D, B) = \frac{S_f}{S_i} . \tag{2.18}$$

Utilizând graful din Fig. 2.9, putem să scriem următoarele ecuații:

$$\begin{aligned} S_f &= D^2 S_1 \\ S_1 &= D S_2 + D S_3 \\ S_2 &= D^2 B S_i + B S_1 \\ S_3 &= D B S_3 + D B S_2 \end{aligned} \tag{2.19 a}$$

rezultă:

$$S_f = (1 - 2BD) = D^5 B S_i . \tag{2.19 b}$$

După rezolvarea acestui sistem de patru ecuații, se obține funcția de transfer a codului:

$$T(D, B) = \frac{S_f}{S_i} = \frac{D^5 B}{1 - 2BD} . \tag{2.20}$$

Dezvoltând funcția de transfer $T(D, B)$ în serie de puteri, putem să scriem [GLJ96]:

$$T(D, B) = \sum_{k=0}^{\infty} 2^k D^{k+5} B^{k+1} . \tag{2.21}$$

Această dezvoltare în serie arată că există 2^k căi de pondere Hamming $(k+5)$ și că acestea corespund respectiv secvențelor de ponderi $(k+1)$ de la intrarea codorului. Exponentul variabilei D , în primul termen al dezvoltării în serie de puteri a funcției de transfer, pentru $k=0$, este egal cu distanța liberă d_{free} a codului (pentru exemplul nostru $d_{free}=5$).

Pentru un cod de rată $R=k/n$, probabilitatea de eroare P_{eb} poate fi mărginită superior de următoarea relație:

$$P_{eb} \leq \frac{1}{k} \left. \frac{\partial T(D, B)}{\partial B} \right|_{D=\Delta, B=1}, \quad (2.22)$$

unde $T(D, B)$ este funcția de transfer a codorului și Δ este o mărime ce depinde de tipul canalului utilizat.

În continuare se prezintă valorile lui Δ pentru două canale clasice.

1. Canal cu ieșire binară:

$$\Delta = \sum_{j=0}^1 \sqrt{p_{0j} p_{1j}}, \quad (2.23)$$

unde $p_{i,j}$, $i=0,1$ este probabilitatea ca un eșantion binar de la intrarea decodorului să fie egal cu j , condiționată de emisia, de la codor, a unui element binar având valoarea i .

2. Canal cu ieșire continuă:

$$\Delta = \int_{-\infty}^{+\infty} \sqrt{p_0(y) p_1(y)} dy, \quad (2.24)$$

unde $p_i(y)$, $i=0,1$ reprezintă densitatea de probabilitate a eșantioanelor analogice de la intrarea decodorului, condiționată de emisia de la codor a unui element binar având valoarea i .

Pentru a ilustra calculul probabilității de eroare voi considera două exemple:

1. Canal binar simetric

Se cunoaște că intrarea și ieșirea unui canal binar simetric sunt alcătuite din elemente binare, din alfabetul $\{0,1\}$. Cantitatea $p_{i,j}$, $i=0,1$, ce apare în relația (2.23), reprezintă așadar probabilitatea de tranziție a canalului. Notând cu p probabilitatea de eroare din canal, se obține:

$$\begin{aligned} p_{i,j} &= p, \text{ dacă } i \neq j, \\ p_{i,j} &= 1-p, \text{ dacă } i = j. \end{aligned} \quad (2.25)$$

Rezultă că Δ va fi egală cu:

$$\Delta = 2\sqrt{p(1-p)}. \quad (2.26)$$

Considerând codorul convoluțional din Fig. 2.2 ($k=1$ și $n=2$) și ținând cont de faptul că:

$$\frac{\partial T(D, B)}{\partial B} = \frac{D^5}{[1 - 2DB]^2}, \quad (2.27)$$

probabilitatea de eroare P_{eb} , conform relației (2.22), este mărginită superior de:

$$P_{eb} \leq \frac{32[\sqrt{\rho(1-\rho)}]^5}{[1 - 4\sqrt{\rho(1-\rho)}]^2}. \quad (2.28)$$

Dacă se consideră o modulație binară (modulație de fază cu două sau 4 stări), probabilitatea de eroare p pe canalul binar simetric este egală cu:

$$p = \frac{1}{2} \operatorname{erfc} \sqrt{\frac{RE_b}{N_0}}. \quad (2.29)$$

unde E_b reprezintă energia pe element binar de informație transmis, erfc este funcția eroare, $R=1/2$ reprezintă rata codului, iar N_0 este densitatea spectrală de putere unilaterală a zgomotului.

2. Canal AWGN

Ieșirea unui canal AWGN (zgomot aditiv alb gaussian) fiind formată din eșantioane analogice, mărimile $p_i(y)$, $i=0,1$ reprezintă densitățile de probabilitate condiționate ale acestor eșantioane. Mărima Δ definită de relația (2.24) este egală cu:

$$\Delta = \int_{-\infty}^{+\infty} \sqrt{p_0(y)p_1(y)} dy = \exp\left(-\frac{RE_b}{N_0}\right), \quad (2.30)$$

și probabilitatea de eroare pe element binar de informație, pentru codul convoluțional din Fig. 2.2, poate fi mărginită superior conform relației (2.22) și utilizând relația (2.27) rezultă:

$$P_{eb} \leq \frac{e^{-5E_b/2N_0}}{(1 - 2e^{-E_b/2N_0})^2}. \quad (2.31)$$

Calculul marginii superioare a probabilității de eroare P_{eb} plecând de la relația (2.22) necesită determinarea funcției de transfer a codorului. Din păcate numărul de stări ale codorului crește exponențial în funcție de lungimea sa de constrângere, iar calculul funcției de transfer devine dificil de efectuat pe măsură ce lungimea de constrângere depășește câteva unități. Această problemă poate fi rezolvată prin înlocuirea funcției de transfer cu dezvoltarea sa în serie, în expresia (2.22) a marginii superioare a probabilității de eroare.

Pentru codorul din Fig. 2.2 luând $B=1$ în relațiile (2.21) și (2.27), și dezvoltând în serii de puteri rezultă:

$$T(D, B)|_{B=1} = \frac{D^5}{1-2D} = D^5 \sum_{d=0}^{\infty} 2^d D^d = \sum_{d=d_f}^{\infty} 2^{d-d_f} D^d = \sum_{d=d_f}^{\infty} n(d) D^d, \quad (2.32)$$

$$\frac{\partial T(D, B)}{\partial B}|_{B=1} = \frac{D^5}{[1-2D]^2} = D^5 \sum_{d=0}^{\infty} 2^d (d+1) D^d = \sum_{d=d_f}^{\infty} 2^{d-d_f} (d-4) D^d = \sum_{d=d_f}^{\infty} w(d) D^d.$$

Probabilitatea de eroare va fi marginită superior de:

$$P_{eb} \leq \frac{\partial T(D, B)}{\partial B} \Big|_{D=\Delta, B=1} = \sum_{d=5}^{\infty} 2^{d-5} (d-4) \Delta^d. \quad (2.33)$$

Notând:

$$w(d) = 2^{d-5} (d-4), \quad (2.34)$$

și remarcând că $d=5$ este distanța liberă a codului, expresia (2.33) se poate scrie mai general, pentru un cod de rată k/n , sub forma:

$$P_{eb} \leq \frac{1}{k} \sum_{d=d_f}^{\infty} w(d) \Delta^d. \quad (2.35)$$

Ansamblul de coeficienți $n(d)$ și $w(d)$ este numit spectrul de distanță al codului.

2.6. Simularea spectrului ponderilor

În literatură, [THI93], am găsit spectrul ponderilor doar pentru un număr redus de coduri convoluționale de rată $R=1/2$ și lungime de constrângere $K=3$. În tabelul următor am reprezentat, cu ajutorul propriilor simulări, [BAK04a], spectrul ponderilor (coloana $n(d)$) pentru toate codurile convoluționale, exceptând codurile convoluționale recursive și nesistematice, RNSC (Recursive Non-Systematic Code), unde:

- d reprezintă ponderea căii,
- $n(d)$ este numărul de căi de pondere d ,
- $w(d)$ este suma ponderilor din secvențele de informație corespunzătoare celor $n(d)$ căi.

Tabel 2.1 Spectrul ponderilor pentru codurile convoluționale de rată 1/2 și $K=3$.

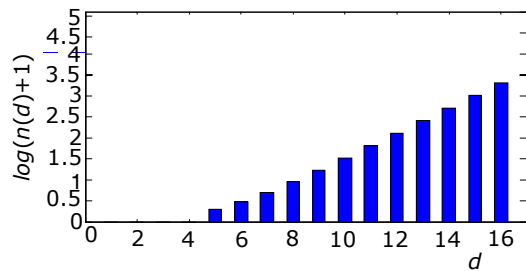
Cod d	NRSC[1,5] (1)		NRSC[1,7] (2)		RSC[1,7/3] (3)		RSC [1,1/5] (4)	
	$n(d)$	$w(d)$	$n(d)$	$w(d)$	$n(d)$	$w(d)$	$n(d)$	$w(d)$
1	0	0	0	0	0	0	0	0
2	0	0	0	0	0	0	0	0
3	1	1	0	0	0	0	1	2
4	1	2	2	3	1	2	3	6
5	1	3	0	0	2	4	5	10
6	2	6	5	15	2	6	8	18
7	4	14	0	0	5	18	12	29
8	7	30	13	58	8	32	19	49
9	11	57	0	0	13	62	31	84
10	17	102	34	201	24	128	51	145
11	27	181	0	0	40	236	81	239
12	44	324	89	655	69	452	130	401
13	72	580	0	0	120	856	210	678
14	117	1028	233	2052	205	1586	341	1151
15	189	1801	0	0	354	2956	553	1944
16	305	3130	610	6255	610	5458	885	3218
Cod d	RSC[1,7/5] (5)		RSC[1,1/7](6)		RSC[1,3/7] (7)		RSC[1,5/7] (8)	
	$n(d)$	$w(d)$	$n(d)$	$w(d)$	$n(d)$	$w(d)$	$n(d)$	$w(d)$
1	0	0	0	0	0	0	0	0
2	0	0	0	0	0	0	0	0
3	0	0	0	0	0	0	0	0
4	0	0	2	5	1	2	0	0
5	1	2	0	0	2	6	1	3
6	2	6	5	15	2	6	2	6
7	4	14	0	0	5	17	4	14
8	8	32	13	46	8	32	8	32
9	16	72	0	0	13	55	16	72
10	32	160	34	139	24	112	32	160
11	64	352	0	0	40	204	64	352
12	128	768	89	413	69	376	128	768
13	256	1664	0	0	120	704	256	1664
14	512	3584	233	1210	205	1284	512	3584
15	1024	7680	0	0	354	2354	1024	7680
16	2048	16384	610	3505	610	4302	2048	16384
Cod d	NRNSC[3,7] (9)		NRNSC[5,7] (10)					
	$n(d)$	$w(d)$	$n(d)$	$w(d)$				
1	0	0	0	0				
2	0	0	0	0				
3	0	0	0	0				
4	1	2	0	0				
5	2	4	1	1				
6	2	8	2	4				
7	5	21	4	12				
8	10	48	8	32				
9	15	87	16	80				
10	28	188	32	192				
11	54	394	64	448				
12	85	698	128	1024				
13	146	1350	256	2304				
14	269	2664	512	5120				
15	460	4906	1024	11264				
16	770	9008	2048	24576				

În continuare, pe baza rezultatelor pe care le-am obținut, am făcut o analiză a acestor coduri.

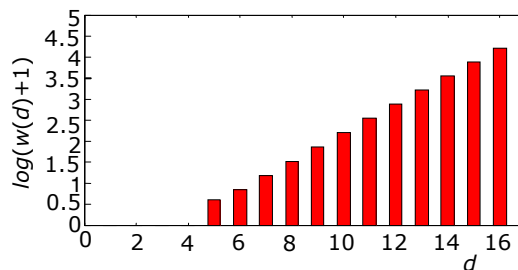
O primă comparație ce se poate face pe baza tabelului 2.1 este din punct de vedere al distanței minime a codului. Astfel, există :

- două coduri cu distanța minimă $d_{min}=3$, codul nerecursiv și sistematic, NRSC[1,5] și codul recursiv și sistematic, RSC[1,1/5],
- cinci coduri: NRSC [1,7], RSC[1,7/3], RSC[1,1/7], RSC[1,3/7] și NRNSC (Non-Recursive Non-Systematic Code) [3,7] au $d_{min}=4$,
- iar codurile: RSC[1,7/5], RSC[1,5/7] și NRNSC[5,7] au $d_{min}=5$.

Am notat în paranteze matricea generatoare în octal. De exemplu, lui $G(D)=[1,1/(1+D^2)]$ îi corespunde $G=[1,1/5]$. Distanța minimă superioară a ultimelor 3 coduri, specificate mai sus, indică o superioritate din punctul de vedere al capacității de corecție, fapt ce se va manifesta, în special la raporturi semnal/zgomot mari, unde sunt importante cuvintele (căile) de ponderi mici. În Fig. 2.10 se prezintă spectrul ponderilor codului recursiv și sistematic, RSC[1,5/7]:



a)



b)

Fig. 2.10 Distanța spectrală a codului RSC[1,5/7]: a) $\log(n(d)+1)/d$; b) $\log(w(d)+1)/d$.

Pe baza observațiilor anterioare și datorită faptului că funcția $w(d)$ ia valoarea 1 pentru codul NRNSC[5,7] și 2 pentru codul RSC[1,7/5] pentru $d_{min}=5$, se poate concluziona că cel mai performant cod la SNR-uri mari va fi NRNSC[5,7], [BAK04a]. La SNR-uri mici, vor conta căile cu ponderi mari. Cu alte cuvinte, aici se recomandă codurile NRSC[1,7] și RSC[1,1/7] sau chiar NRSC[1,5] și RSC[1,3/7].

3. TURBO CODURI

Dintre toate metodele de corecție a erorilor cunoscute până astăzi, turbo codurile (TC) alături de codurile LDPC (Low-Density Parity-Check), [CFRU01, BCVSU02] se apropie cel mai mult de limita lui Shannon [SHA48], limita teoretică a ratei maxime de transfer a informației printr-un canal cu zgomot. Datorită performanțelor lor apropiate de capacitatea lui Shannon, TC-urile, încă de la introducerea lor [BGT93], au beneficiat de o atenție deosebită. Astfel, din momentul apariției lor, TC-urile au fost studiate pe larg și adoptate în mai multe sisteme de comunicații.

Turbo codurile sunt atractive, în special, pentru sistemele de comunicații mobile, fiind incluse în standardele celulare de generația a treia (3G), cum ar fi: Sistemul de Telecomunicații Mobile Universal (UMTS-Universal Mobile Telecommunications System), cunoscut și sub denumirea de Acces Multiplu prin Divizare în Cod de Bandă Largă (WCDMA-Wideband Code Division Multiple Access), [TSG99] și CDMA2000, [TIA02]. Deasemenea, sunt incluse și în standardele video de difuzare digitală (DVB- Digital Video Broadcasting) de la canal spre satelit (DVB-RCS), [[ETSI03]] în sistemul de distribuție terestră (DVB-RCT), [ETSI02] cât și în standardele 802.16-2004 (deseori numit 802.16d), [STAND04] și 802.16e-2005 (care este o îmbunătățire a standardului 802.16-2004 și care este deseori numit 802.16e) [STAND05], primul standard făcând referire la WiMAX (Worldwide Interoperability for Microwave Access) fix deoarece acest standard nu are suport pentru mobilitate, iar al doilea la WiMAX mobil, deoarece a introdus, printre alte lucruri, și suportul pentru mobilitate.

Secvența de informație este codată de două ori, având un interleaver între cele două codoare. Rolul interleaver-ului este acela de a se asigura că cele două secvențe de biți codate sunt independente una față de cealaltă, datorită amestecării biților de informație de către acesta. Cel mai des sunt utilizate codoarele RSC cu ieșire sistematică care este echivalentă cu sevența de informație originală, precum și o secvență de informație de paritate. Cele două secvențe de paritate pot fi apoi puncturate înainte de a fi transmise împreună cu secvența de informație originală a decodorului. Această puncturare a informației de paritate permite o gamă largă a ratelor de codare și deseori este transmisă jumătate din informația de paritate de la fiecare codor. Împreună cu secvența de date originală acesta conduce la o rată de codare 1/2.

Două decodoare RSC sunt utilizate la decodare. Trebuie utilizați algoritmi speciali de decodare, care să accepte intrări soft și care generează ieșiri soft pentru secvența decodată. Aceste intrări și ieșiri soft furnizează o indicație asupra faptului că un bit oarecare a fost 0 sau 1, precum și raportul de plauzibilitate care dă probabilitatea ca bitul respectiv să fi fost corect decodat. Turbo decodorul operează în mod iterativ. În prima iterație primul decodor RSC furnizează ieșiri soft dând o estimare a secvenței de date originale bazate doar pe intrările soft furnizate de canal. De asemenea el furnizează o ieșire extrinsecă. Ieșirea extrinsecă pentru un bit dat nu se bazează numai pe intrarea canalului corespunzătoare acestui bit ci și pe informația biților vecini și pe constrângerile impuse de către codul ce a fost utilizat. Această ieșire extrinsecă de la primul decodor, este utilizată de către cel de-al doilea decodor RSC, ca informație *a-priori*, și această informație împreună cu

intrările canalului sunt utilizate de către cel de-al doilea decodor RSC pentru a genera ieșirea sa soft și informația sa extrinsecă. În ceea ce privește a doua iterație informația extrinsecă a celui de-al doilea decodor, de la prima iterație, este utilizată ca informație *a-priori* de primul decodor, și, utilizând această informație *a-priori*, decodorul poate decoda corect mai mulți biți față de prima iterație. Acest ciclu poate continua, la fiecare iterație, ambele decodoare RSC producând o ieșire soft și informație extrinsecă bazate pe intrările canalului și pe informația *a-priori*, obținută din informația extrinsecă provenită de la celălalt decodor. După fiecare iterație, BER scade, dar îmbunătățirile obținute cu fiecare iterație se micșorează pe măsură ce numărul de iterații crește, astfel că din motive de complexitate sunt utilizate de obicei între 4 și 15 iterații.

În lucrarea lor de început, Berrou ș.a., au invocat o versiune modificată a algoritmului de decodare MAP, datorită lui Bahl ș.a., [BCJR74], în structura iterativă de mai sus pentru decodarea codurilor componente. Acest algoritm, clasic, conduce la o probabilitate minimă a erorii pe bit. De la apariția turbo codurilor s-a depus un efort mare pentru a reduce complexitatea decodării, de către diverși cercetători, ca Robertson, Villebrun și Hoher [RVH95], Berrou ș.a. [BAAF93], Goff, Glavieux și Berrou [GGB94]. Robertson și Worz [ROW98] sugerează utilizarea codurilor împreună cu scheme de modulație care folosesc eficient banda de frecvențe. Lucrările lui Benedetto și Montorsi [BEM96a, BEM96b] respectiv Perez ș.a. [PSC96] au facilitat înțelegerea cauzelor performanțelor excelente ale turbo codurilor. Un număr de contribuții similare au fost aduse de către Hagenauer, Offer și Papke [HOP96] și de Pyndiah [PYN97], care extind conceptul de turbo și la codurile bloc concatenate în paralel. Bărbulescu și Peitrobon [BAP94], ca și un număr de alți autori, alocă o mare importanță realizării interleaver-ului.

Din punctul de vedere al concatenării codurilor convoluționale componente de la codor, respectiv, decodor, rezultă următoarele scheme: coduri convoluționale concatenate paralel (turbo codurile), coduri convoluționale concatenate serie și coduri convoluționale concatenate hibrid. În paragraful 3.3 se prezintă codurile convoluționale concatenate paralel.

3.1. Turbo codorul

Structura generală utilizată în turbo codor este prezentată în Fig. 3.1, [BGT93]. Sunt utilizate două coduri componente pentru a coda biții de intrare și un interleaver plasat între cele două codoare.

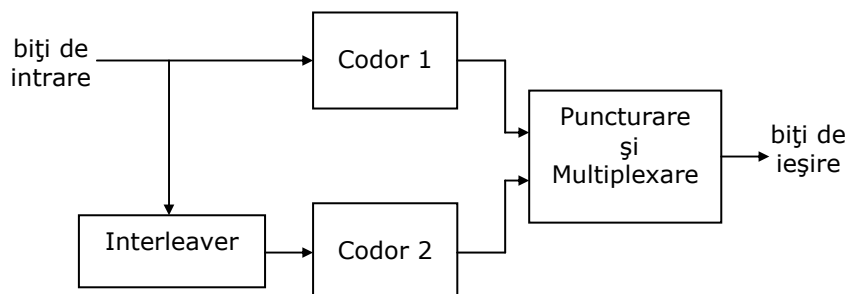


Fig. 3.1 Schema unui turbo codor.

În general sunt utilizate codurile RSC, ca și coduri componente, dar este posibil de a se obține performanțe bune, utilizând o structură ca cea din Fig. 3.1, cu ajutorul altor coduri componente, cum ar fi codurile bloc, propuse de Hagenauer, Offer și Papke [HOP96]. De asemenea, este posibil să se utilizeze mai mult de două coduri componente.

Ieșirile celor două codoare componente sunt mai apoi puncturate [THI93] și multiplexate. De obicei ambele coduri componente RSC sunt de rată $1/2$, dând un bit de paritate și un bit sistematic pentru fiecare bit de intrare.

Pentru a obține un turbo codor cu o rată de codare $R=1/2$, trebuie să puncturați jumătate din biții de ieșire ai fiecăruia dintre codoare. Astfel, se transmit toți biții sistematici ai primului codor RSC și jumătate din biții de paritate ai fiecărui codor. De precizat că biții sistematici sunt foarte rar puncturați, deoarece aceasta degradează dramatic performanța codului. În Fig. 2.3 b) s-a prezentat un cod RSC de rată $1/2$ și lungime de constrângere $K=3$, ce poate fi folosit ca și cod component în Fig. 3.1. Codurile concatenate în paralel au fost propuse și analizate de Claude Berrou ș.a. în articolul din 1993, [BGT93].

Îmbunătățiri considerabile în performanță ale turbo codurilor se datorează interleaver-ului utilizat între cele două codoare și a utilizării codurilor recursive ca și coduri componente. Articolele publicate de Benedetto și Montorsi [BEM96a, BEM96b] încearcă să explice remarcabilele performanțe ale turbo codurilor, ajungând la concluzia că turbo codurile pot avea un câștig al performanței proporțional cu lungimea interleaver-ului utilizat. Complexitatea decodării per bit nu depinde de lungimea interleaver-ului. Așadar pot fi obținute performanțe foarte bune cu o complexitate rezonabilă prin utilizarea interleaver-elor foarte lungi. Însă, în multe aplicații importante, precum transmisiile vocale, lungimile blocurilor de date extrem de mari nu sunt practice, datorită întârzierii rezultate.

3.2. Turbo decodorul

Structura generală a unui decodor iterativ este reprezentată în Fig. 3.2. Două componente decodoare sunt conectate prin interleaver-e, într-o structură similară cu cea de la codor. Așa cum se observă din figură, fiecare decodor are trei intrări: biții de ieșire din canal codați sistematic, biții de paritate transmiși de către codul component asociat și informația de la celălalt decodor numită informație *a-priori*. Decodoarele componente au de explorat ambele intrări provenite din canal și această informație *a-priori*. Ele trebuie de asemenea să furnizeze și (cele ce sunt cunoscute ca și) ieșiri soft pentru biții decodați. Aceasta înseamnă că, la fel cum furnizează secvența de biți de la ieșire decodată, decodoarele componente trebuie să genereze probabilitățile de decodare corectă, asociate pentru fiecare bit. Ieșirile soft sunt tipic reprezentate în termenii așa numiților logaritmi de rapoarte de plauzibilitate, (Log Likelihoods Ratios), LLRs. Polaritatea LLR-ului determină semnul bitului, în vreme ce amplitudinea sa cuantifică probabilitatea deciziei corecte.

Turbo decodorul din Fig. 3.2 lucrează iterativ, în prima iterație primul decodor component preia doar valorile ieșirilor canalului și produce o ieșire soft ca o estimare a biților de date. Ieșirea soft a primului decodor este apoi utilizată ca informație adițională de cel de al doilea decodor împreună cu ieșirile canalului pentru a calcula estimații săi pentru biții de date.

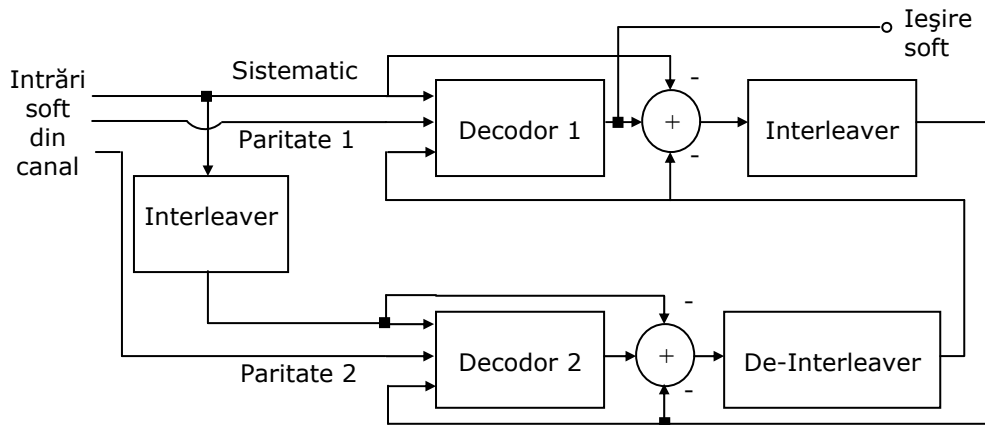


Fig. 3.2 Schema turbo decodurii.

Acum cea de a doua iterație poate să înceapă, primul decodor decodează din nou ieșirile din canal, dar de data aceasta cu o informație adițională despre valorile biților de intrare furnizată de ieșirea celui de-al doilea decodor în prima iterație. Această informație adițională permite primului decodor să obțină un set de ieșiri soft de acuratețe mai mare, care este apoi utilizat de cel de-al doilea decodor ca informație *a-priori*. Acest ciclu este repetat și cu fiecare iterație rata de eroare per bit a biților decodați, tinde să scadă. Însă, îmbunătățirea performanței cu creșterea numărului de iterații descrește. Astfel, din motive de complexitate, sunt utilizate uzual doar un număr de 8 iterații.

Datorită întrețeserii utilizată la codor, trebuie întrețesut și de-întrețesut LLR-ul care este utilizat pentru a reprezenta valorile soft ale biților, așa cum se vede în Fig. 3.2. În plus, datorită naturii iterative a decodurii, trebuie avut grijă să nu se reutilizeze aceeași informație mai mult de o dată la fiecare pas de decodare. Din acest motiv a fost utilizat conceptul așa numitei informații extrinseci și intrinseci în articolul scris de Berrou ș.a. [BGT93]. Alte decodoare, neiterative, care dau o decodare optimă a turbo codurilor au fost propuse, [BRH00, BRH97]. Însă îmbunătățirea în performanță, comparativ cu decodoarele iterative, s-a găsit a fi de doar 0,35 dB [HLY02], și sunt foarte complexe. Așadar schema prezentată în Fig. 3.2 este utilizată cel mai des.

3.3. Concatenarea paralelă a codurilor convoluționale

În cadrul acestei teze, în primele simulări realizate, am utilizat codurile convoluționale concatenate în paralel (turbo codurile), nepuncturate. Schema generală a unui turbo cod, TC, este prezentată în Fig. 3.3.

Secvența de informație, notată u , este codată de către codorul C1 rezultând secvența de paritate x_1 . Aceeași secvență de biți, u , este furnizată, însă în altă ordine obținută prin întrețesere cu ajutorul interleaver-ului „I”, codorului C2, care generează la rândul său, secvența de paritate x_2 .

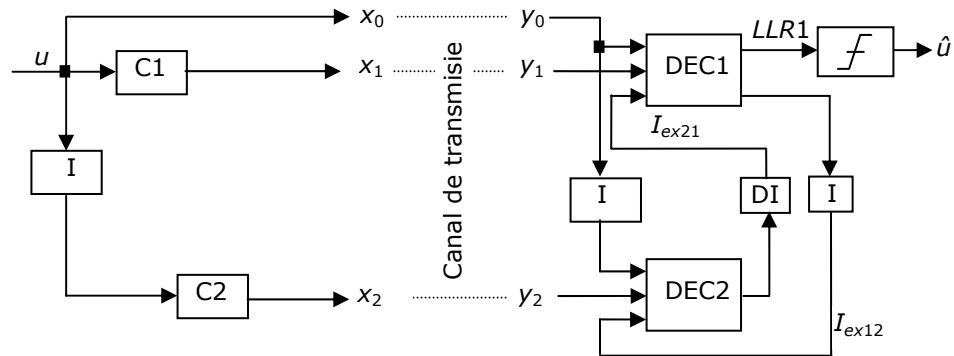


Fig. 3.3 Coduri convoluționale concatenate paralel.

Secvențele rezultate $x_0=u$, x_1 și x_2 , prin multiplexare și modulare (operații omise în Fig. 3.3) se transformă în semnalul ce va fi emis în canal. La ieșirea acestuia, prin demodulare și demultiplexare rezultă secvențele (soft) recepționate, corespunzătoare: y_0 , y_1 și y_2 .

Fiecare decodor calculează LLR-ul pentru fiecare bit din u :

$$LLR(u_j) = \ln \frac{p(u_j = 1|y)}{p(u_j = 0|y)} \quad (3.1)$$

În figură este prezentat doar logaritmul raportului de plauzibilitate al primului decodor, notat LLR1 și informația extrinsecă destinată celuiilalt decodor.

Fiecare decodor primește informație extrinsecă și pe baza ei și a secvențelor venite din canal (y_0 și y_1 respectiv y_0 întrețesută și y_2) furnizează la rândul-i informație extrinsecă. Acest proces se repetă iterativ de un anumit număr de ori (impus sau calculat, funcție de tipul turbo codului). După efectuarea tuturor iterațiilor se ia o decizie hard asupra logaritmului raportului de plauzibilitate generat după ultima iterație de unul din cele două decodoare (în Fig. 3.3 s-a ales LLR1). Secvența rezultată constituie ieșirea turbo decodorului.

3.4. Dispozitive de întrețesere

Întrețeserea este o tehnică de creștere a capacității de corecție a erorii de codare. A fost deseori utilizată împreună cu codarea controlată a erorilor pentru canalele cu erori în rafală [VUY01]. Un exemplu este cel al canalului cu fading multicăi, la care variațiile de semnal datorate propagării multicăi determină adeseori scăderea nivelului semnalului sub cel al nivelului zgomotului, ducând la un număr mare de erori.

O metodă efectivă de reducere a acestor erori este aceea de a insera un *interleaver* între codor și intrarea canalului. Datele codate sunt reordonate de interleaver și apoi transmise în canal. La receptor, de-interleaver-ul realizează operația inversă, pentru a reda datele în ordinea originală.

Ca rezultat al acestei operații de întrețesere și de-întrețesere, erorile în rafală sunt împrăștiate, astfel încât erorile dintr-un cuvânt de cod apar independent. Astfel, canalul cu erori în rafală este transformat într-un canal cu erori aleatoare și un cod realizat pentru un canal cu erori independente poate fi utilizat și în cazul canalelor cu erori în rafale.

În cazul turbo codării, întrețeserea este realizată înainte de a fi codată informația de date de cel de-al doilea codor component. În general lungimea N a interleaver-ului este semnificativ mai mare decât memoria codului și elementele interleaver-ului sunt alese în mod aleator.

Limita capacității Shannon poate fi atinsă doar de coduri bloc de lungime mare. Acestea pot fi implementate în varianta construcțională doar folosind memorii mari. Rolul de bază al interleaver-ului este de a construi un cod bloc lung pentru coduri convoluționale cu memorie mică.

Un alt rol al interleaver-ului este acela de a împrăștia erorile în rafală. Interleaver-ul furnizează informația de date amestecată spre cel de-al doilea codor component [BGT93], și decorează intrările celor două decodare componente de la recepție, astfel încât poate fi aplicat un algoritm de decodare sub-optimal iterativ bazat pe schimbul de informație necorelată dintre cele două decodare componente. De exemplu, după corecția unor erori în primul decodor component, câteva din erorile rămase pot fi împrăștiate de către interleaver astfel încât să poată fi corectate de celălalt decodor. Crescând numărul de iterații, în procesul de decodare, probabilitatea erorii pe bit se apropie de capacitatea canalului.

Un dispozitiv de întrețesere realizează o permutare a unei secvențe de numere [BAK04b), KBN05], de forma:

$$\pi : I \rightarrow I, \text{ cu } I = \{1, 2, \dots, N\}, \quad (3.2)$$

unde N reprezintă lungimea secvenței ce trebuie întrețesută. Pentru refacerea ordinii inițiale se utilizează un dispozitiv pereche, de de-întrețesere, ce implementează funcția inversă:

$$\pi^{-1} : I \rightarrow I, \text{ cu } \pi^{-1}(\pi(i)) = i, \forall i \in I. \quad (3.3)$$

Un interleaver bun trebuie să îndeplinească două condiții: să aibă o distanță minimă de întrețesere de valoare cât mai mare și să aibă un grad de împrăștiere cât mai bun.

Fiind dată funcția interleaver, π , definim distanța de întrețesere dintre pozițiile i și j ca:

$$d(i, j) = |i - j| + |\pi(i) - \pi(j)|, \quad \forall i, j \in I, i \neq j \quad (3.4)$$

Atunci distanța minimă de întrețesere este dată prin, [CRO00, MCK04]:

$$d_{min} = \min_{\substack{i, j \in I \\ i \neq j}} d(i, j) \quad (3.5)$$

Mulțimea valorilor funcției $d(\cdot)$, dată prin relația (3.5), este mărginită inferior de valoarea 2 iar superior de $2 \cdot N - 2$.

3.4.1 Interleaver aleator

Interleaver-ul aleator, [VUY01] este relativ simplu de realizat, oferă o bună împrăștiere a secvenței originale, însă are în general $d_{min}=2$, adică cea mai mică valoare posibilă. În Fig. 3.4 este ilustrată operația de permutare de tip aleator pentru un bloc de date de lungime $N=10$. Procedeu de construcție al unui dispozitiv de întrețesere aleator este următorul. Cunoscând lungimea de întrețesere, N , se construiește mulțimea $A=\{1,2 \dots N\}$. Se alege în mod aleator un număr, $n_1 \in A$. Se atribuie $\pi(1) = n_1$ și se elimină această valoare din A . Procedeu se repetă până la epuizarea mulțimii A . Așadar, se poate scrie funcția de împrăștiere aleatoare de forma:

$$\pi_r(i) = rand(i), \forall i \in I = \{1, 2 \dots N_r\}. \quad (3.6)$$

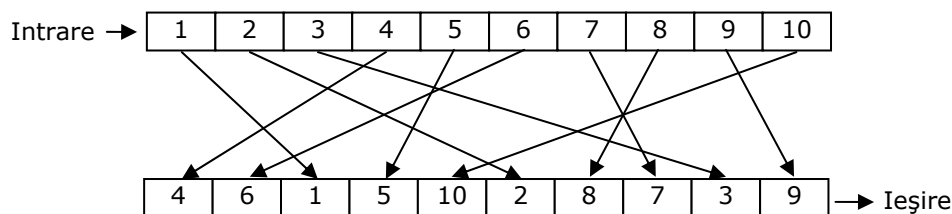


Fig. 3.4 Dispozitiv de întrețesere aleator.

Un dezavantaj major al întrețeserii aleatoare este nereproductibilitatea procedurii de generare al funcției π . Odată generată funcția de tip aleator ea trebuie memorată pentru a putea fi reprodusă.

3.4.2 Interleaver de tip S

Interleaver-ul S este de tip aleator [DIP95], însă, spre deosebire de interleaver-ul pur aleator, prin construcție se forțează o distanță minimă de întrețesere egală cu S . Algoritmul de construcție al funcției de întrețesere este următorul: se selectează o posibilă poziție viitoare pentru bitul curent. Aceasta este comparată cu pozițiile celor S biți selectați anterior în aceeași manieră aleatoare. Dacă se îndeplinește condiția ca:

$$|\pi(n) - \pi(j)| > S \text{ pentru } n \text{ și } j \text{ cu } |n - j| < S \quad (3.7)$$

adică poziția după permutarea bitului curent diferă cu cel puțin $S+1$ poziții față de cele S poziții alese anterior, atunci se trece mai departe. Dacă condiția nu este îndeplinită, se alege o altă poziție a bitului curent, care va fi la rândul său testată. Procesul se va repeta până când s-au găsit pozițiile tuturor celor N biți. Simulările pe calculator au demonstrat că dacă $S \leq \sqrt{\frac{N}{2}}$, atunci procesul va converge. Proiectarea acestor interleaver-e este dificilă pentru că după ce o mare parte a algoritmului a

fost parcursă este din ce în ce mai greu să se genereze numere aleatoare din cele rămase în secvență care să îndeplinească cerința (3.7).

3.4.3 Interleaver bloc

Interleaver-ul bloc, sau rectangular, prezintă cea mai simplă structură, [BAP98]. Lungimea acestui interleaver, N_b , este dată de relația:

$$N_b = X \times Y \quad (3.8)$$

unde X și Y sunt două numere naturale apropiate ca valoare. Funcția interleaver-ului bloc este:

$$\begin{aligned} \pi_b(i + j * X + 1) &= i * Y + j + 1, \\ \forall i \in I = \{0, 1, \dots, X - 1\} \text{ și } \forall j \in J = \{0, 1, \dots, Y - 1\}. \end{aligned} \quad (3.9)$$

Oricare doi biți, situați inițial la o distanță mai mică decât $d_{min} = \min(I, J)$, vor fi situați după întretesere la o distanță superioară lui d_{min} .

În Fig. 3.5 este prezentat un interleaver bloc având lungimea $N=10 \times 10$. Datele de intrare sunt introduse linie cu linie. Citirea se va face pe coloane, schimbându-se astfel ordinea biților. După întretesere secvența devine: $X_1, X_{11}, X_{21}, \dots, X_{91}, X_2, X_{22}, \dots, X_{90}, X_{100}$. Oricare doi biți aflați inițial la mai puțin de 10 (numărul de coloane) poziții unul de celălalt vor fi depărtați la cel puțin 10 (numărul de linii) poziții.

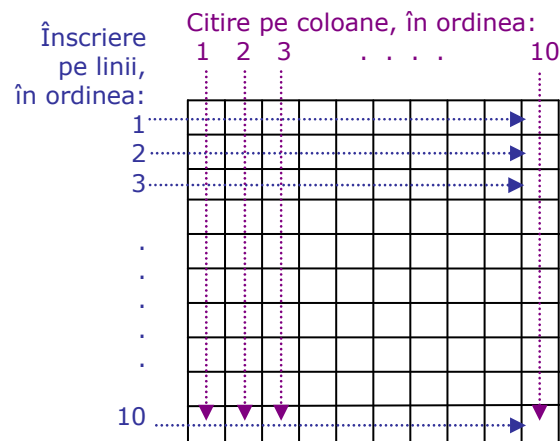


Fig. 3.5 Întretesere bloc.

3.4.4 Interleaver pseudo-aleator

Interleaver-ul pseudo-aleator [JBD04], face parte din recomandarea CCSDS (Consultative Committee for Space Data Systems) denumită "CCSDS Recommendation for Telemetry Channel Coding", [CCSDS02].

Denumirea dispozitivului provine de la faptul că permutările nu sunt aleatoare în adevăratul sens al cuvântului (avem de-a face cu o dezordine controlată). Este un dispozitiv de întrețesere performant, care îmbină avantajele furnizate de interleaver-ele aleator și bloc, adică prezintă o bună împrăștiere la o distanță minimă suficient de mare.

Permutarea pentru fiecare secvență de lungime N a blocului este dată de o reordonare particulară a biților 1, 2, ... N , generată de următorul algoritm:

- $N=k_1 \cdot k_2$ unde k_1 este un parametru fix, iar k_2 variază în funcție de lungimea dispozitivului de întrețesere;

- pentru s de la 1 la N (poziția curentă înainte de interschimbare) se efectuează următoarele operații și se obțin permutările $\pi(s)$:

$$- m = (s - 1) \bmod 2,$$

$$- i = \left\lfloor \frac{s-1}{2k_2} \right\rfloor,$$

$$- j = \left\lfloor \frac{s-1}{2} \right\rfloor - ik_2,$$

$$- t = (19i + 1) \bmod \frac{k_1}{2},$$

$$- q = t \bmod 8 + 1,$$

$$- c = (p_q \cdot j + 21m) \bmod k_2,$$

$$- \pi(s) = 2 \left(t + c \frac{N+1}{2} + 1 \right) - m;$$

în care: [] – parte întreagă,

mod – operație modulo (clase de resturi),

$q=1 \div 8$ și $p_1=31, p_2=37, p_3=43, p_4=47, p_5=53, p_6=59, p_7=61, p_8=67$.

3.4.5 Interleaver Takeshita-Costello

Interleaver-ul Takeshita-Costello, este prezentat în [TAC98] și presupune că lungimea interleaver-ului, N_{TC} , este o putere întreagă a lui 2. Construcția funcției interleaver-ului este dată în cele ce urmează. Vectorul $c_{i=1+N}$, este construit cu relația următoare:

$$c(i) = k \cdot i \cdot (i+1) / 2 \quad (3.10)$$

unde k , în mod uzual, este egal cu unu. Funcția de întrețesere va fi:

$$\pi(c(i)) = c(i+1), \forall i \in I \quad (3.11)$$

3.4.6 Interleaver bloc-aleator

În cadrul acestui paragraf propun două variante noi de interleaver-e pe care le-am realizat și analizat. Acest tip nou de interleaver-e, [KBN05], dorește să îmbine calitățile interleaver-elor bloc (d_{min} de valori mari) și aleator (o bună împrăștiere), fiind o alternativă a celui de tip S , care este greu de realizat. Astfel,

am propus două variante noi de interleaver-e: interleaver bloc aleator în linie, BRL (Block Random in Line), notat IXBRLY (X și Y sunt dați în Tabelul 3.1) și interleaver bloc cu linii aleatoare, BLR (Bloc with Random Lines), notat IXBLRY (X și Y sunt dați în Tabelul 3.1), ai căror algoritmi de generare vor fi descriși în continuare. Am presupus că lungimea acestor interleaver-e este $N_{br}=X \times Y$, la fel ca pentru un interleaver bloc.

3.4.6.1 Interleaver bloc aleator în linie, BRL

Pentru început, am construit matricea:

$$\mathbf{c}(i,j)=1+i+j \cdot X, i \in I=\{0,1 \dots X-1\}, j \in J=\{0,1 \dots Y-1\} \quad (3.12)$$

Apoi, am permutat fiecare linie a acestei matrici, $\mathbf{c}(i,J)$ folosind relația:

$$\pi_r(i) = \text{rand}(i), \forall i \in I=\{1,2 \dots N_r\}, \quad (3.13a)$$

obținând, astfel matricea \mathbf{b} de forma:

$$\mathbf{b}(i,J) = \mathbf{c}(i,\pi_r(J)) \quad \forall i \in I \quad (3.13b)$$

În final, pentru a crește distanța minimă de întretesere se reordonează liniile astfel încât pe primele $X/2$ poziții să se regăsească liniile impare. După ce realizez ordonarea liniilor fac o citire pe coloană:

$$\begin{aligned} \pi_{BRL}(k+j \cdot Y+1) &= \mathbf{b}(i, j), i=(2 \cdot k) \in I, j \in J, \\ \pi_{BRL}(k+j \cdot Y+Y_2+1) &= \mathbf{b}(i, j), i=(2 \cdot k-1) \in I, j \in J, \end{aligned} \quad (3.14)$$

unde: $Y_2 = \text{floor}[(Y-1)/2]$.

Schema interleaver-ului bloc aleator în linie este dată în Fig. 3.6:

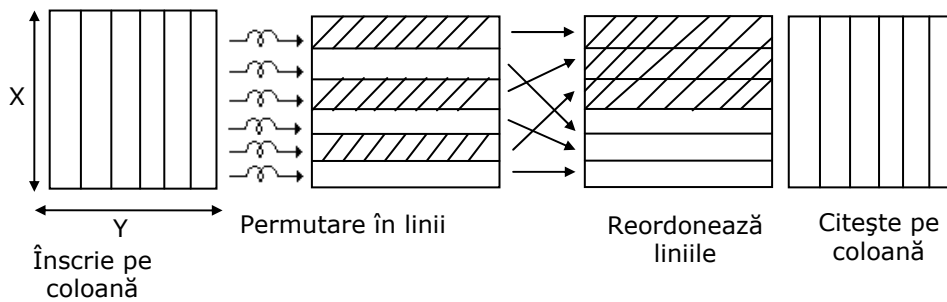


Fig. 3.6 Schema interleaver-ului bloc aleator în linie, BRL.

3.4.6.2 Interleaver bloc cu linii aleatoare, BLR

Algoritmul de generare este identic cu cel al interleaver-ului BRL, cu excepția pasului doi unde aleatorizarea nu o fac în linie ci între linii:

$$d(i, J) = c(\pi_r(i), J) \quad \forall i \in I = \{0, 1, \dots, X-1\} \quad (3.15)$$

Astfel că:

$$\pi_{BLR}(j+i \cdot Y+1) = d(i, j), \quad i \in I, \quad j \in J \quad (3.16)$$

Distanța minimă de întrețesere este $d_{min} \geq Y$ în cazul interleaver-ului BRL și $d_{min} \geq X$ în cazul interleaver-ului BLR. Pentru a obține o bună împrăștiere în cazul interleaver-ului BRL dimensiunea X trebuie aleasă cât mai mare, iar pentru BLR Y trebuie să fie cât mai mare. Un bun compromis se obține dacă se alege $X \cong Y^2$ pentru BRL respectiv $Y \cong X^2$.

În Fig. 3.7 este dată schema interleaver-ului bloc cu linii aleatoare:

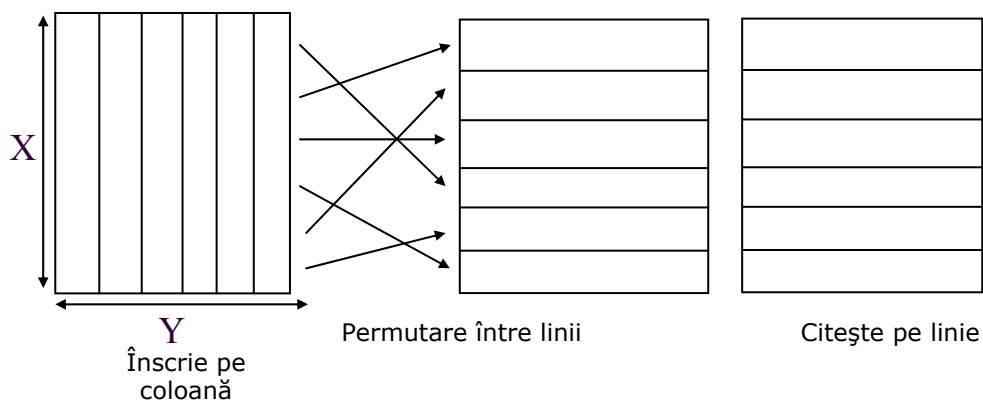


Fig. 3.7 Schema interleaver-ului bloc cu linii aleatoare, BLR.

În concluzie, se poate spune că cele două interleavere propuse se diferențiază prin următoarele caracteristici:

- În cazul interleaver-ului bloc aleator în linie se realizează o întrețesere de tip aleator pur pentru fiecare linie separat. Aceasta reprezintă un avantaj deoarece metoda pur aleatoare este ușor de aplicat și de implementat pentru blocuri mai mici (cum pot fi considerate liniile de lungime Y). Ulterior, reordonarea liniilor va conferi și o distanță minimă egală cu $X/2$. Așadar, interleaver-ul va dispune și de o bună amestecare (datorită întrețeserii aleatoare) cât și de o distanță minimă suficient de mare. Este de așteptat ca acest interleaver să ofere performanțe bune.

- În cazul celui de-al doilea interleaver, interleaver bloc cu linii aleatoare, un dezavantaj față de primul este că întrețeserea aleatoare a liniilor nu va reuși o distanță minimă impusă (întrețeserea pur aleatoare are de obicei distanța minimă egală cu 2, adică cea mai mică posibilă).

3.4.7 Performanțele interleaver-elor utilizate în turbo coduri

În scopul analizei performanțelor, BER și FER (rata erorii pe cadru), interleaver-elor (de diferite lungimi) utilizate în turbo coduri, am folosit interleaver-ele prezentate mai sus. Astfel, în Tabelul 3.1, au fost trecute notațiile corespunzătoare acestor interleaver-e.

Tabelul 3.1. Interleavere folosite.

N_{Ξ}	400	900	1800	3600
IXRY	I19R21	I29R31	I41R45	I59R61
IN_rS1	I392S1	I896S1	I1784S1	I3568S1
IN_sSY	I392S5	I896S5	I1784S10	I3568S10
	I392S10	I896S10	I1784S20	I3568S20
	I392S13	I896S20	I1784S29	I3568S40
IXP8	I392P8	I896P8	I1784P8	I3568P8
IXTC	I512TC	I1024TC	I2048TC	I4096TC
IXBRLY	I49BRL8	I81BRL11	I121BRL15	I225BRL16
IXBLRY	I7BLR56	I9BLR99	I11BLR165	I15BLR240

unde: - Interleaver-ul aleator s-a notat cu IN_rS1 ,
 - Interleaver-ul S s-a notat cu IN_sSY ,
 - Interleaver-ul bloc s-a notat cu $IXRY$,
 - Interleaver-ul pseudo-aleator s-a notat cu $IXP8$,
 - Interleaver-ul Takeshita-Costello s-a notat cu $IXTC$,
 - Interleaver-ul bloc aleator în linie s-a notat cu $IXBRLY$,
 - Interleaver-ul bloc cu linii aleatoare s-a notat cu $IXBLRY$.

N_r , N_s , X și Y , iau valorile din Tabelul 3.1.

În diagramele din Fig. 3.8, am prezentat câteva curbe cu performanțele BER pe care le-am obținut, utilizând diferite interleaver-e:

- interleaver aleator,
- interleaver S,
- interleaver bloc,
- interleaver pseudo-aleator,
- interleaver Takeshita-Costello,
- interleaver-e BLR și BRL.

Aceste interleaver-e au fost introduse într-un turbo cod nepuncturat, de rată 1/3, în care două coduri componente RSC, având matricea generatoare $G=[1, 15/13]$ și lungimea de constrângere a codului convoluțional $K=4$, sunt conectate în paralel [BGT93]. Lungimile interleaver-elor sunt în jur de: 400, 900, 1800 și 3600. Valorile exacte iau în considerație limitarea impusă de construcția fiecărui interleaver. În simulări am folosit algoritmul de decodare MAP și un număr de 12 iterații. S-a considerat canalul AWGN și modulația BPSK (Binary Phase Shift Keying).

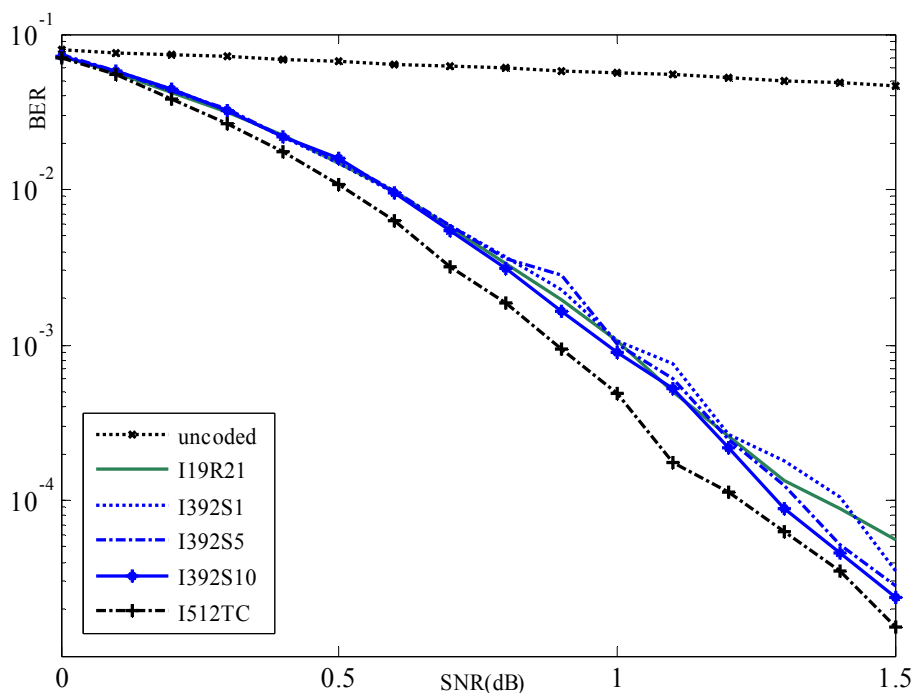


Fig. 3.8 a) Performanțele BER ale turbo-codurilor de rată 1/3, [1, 15/13], pentru 5 interleaver-e cu $N=400$.

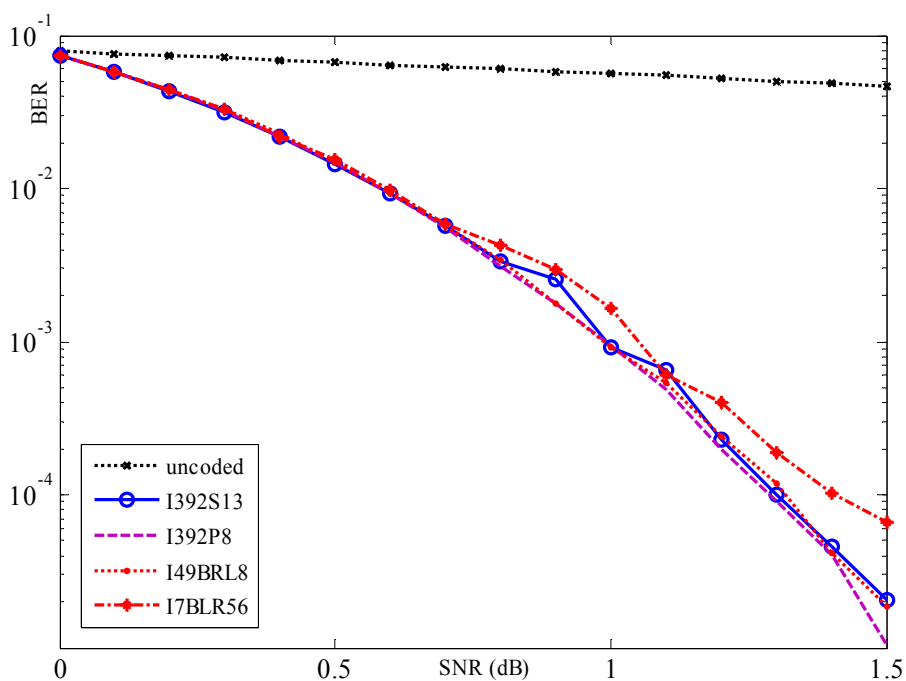


Fig. 3.8 b) Performanțele BER ale turbo-codurilor de rată 1/3, [1, 15/13], pentru alte 4 interleaver-e cu $N=400$.

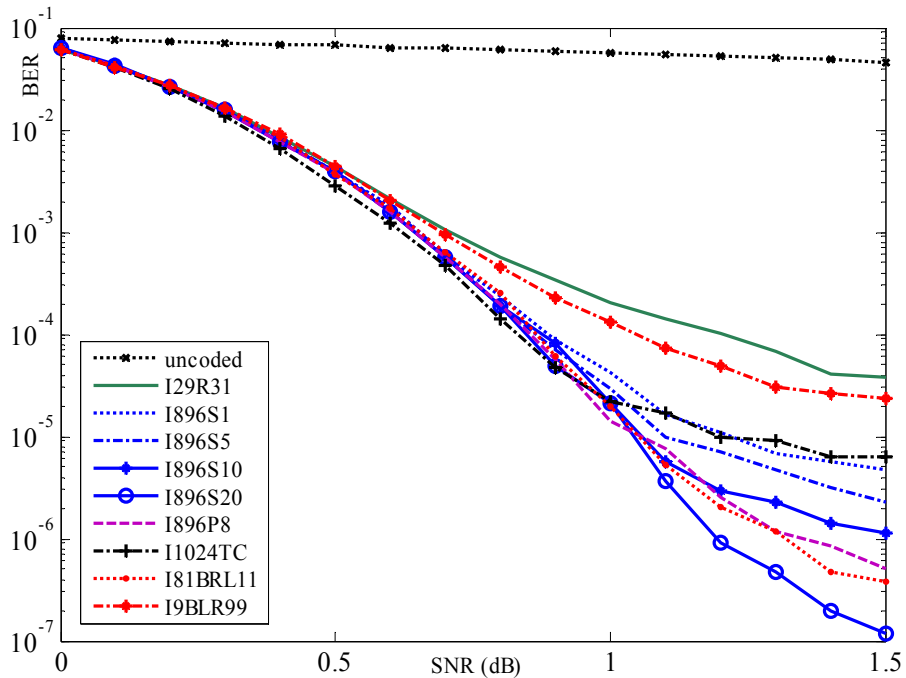


Fig. 3.8 c) Performanțele BER ale turbo-codurilor de rată 1/3, [1, 15/13], pentru interleaver-e cu $N=900$.

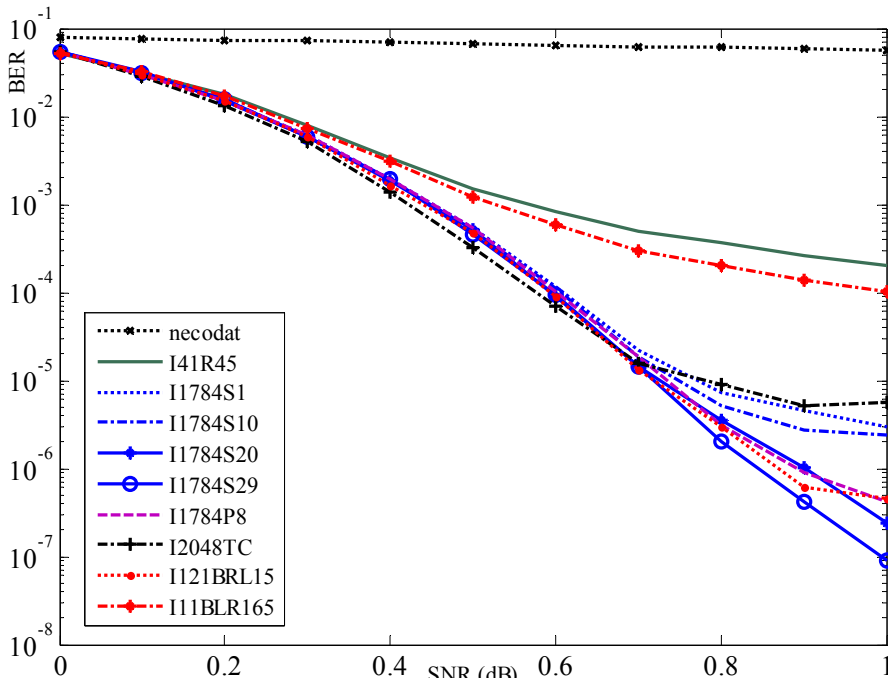


Fig. 3.8 d) Performanțele BER ale turbo-codurilor de rată 1/3, [1, 15/13], pentru interleaver-e cu $N=1800$.

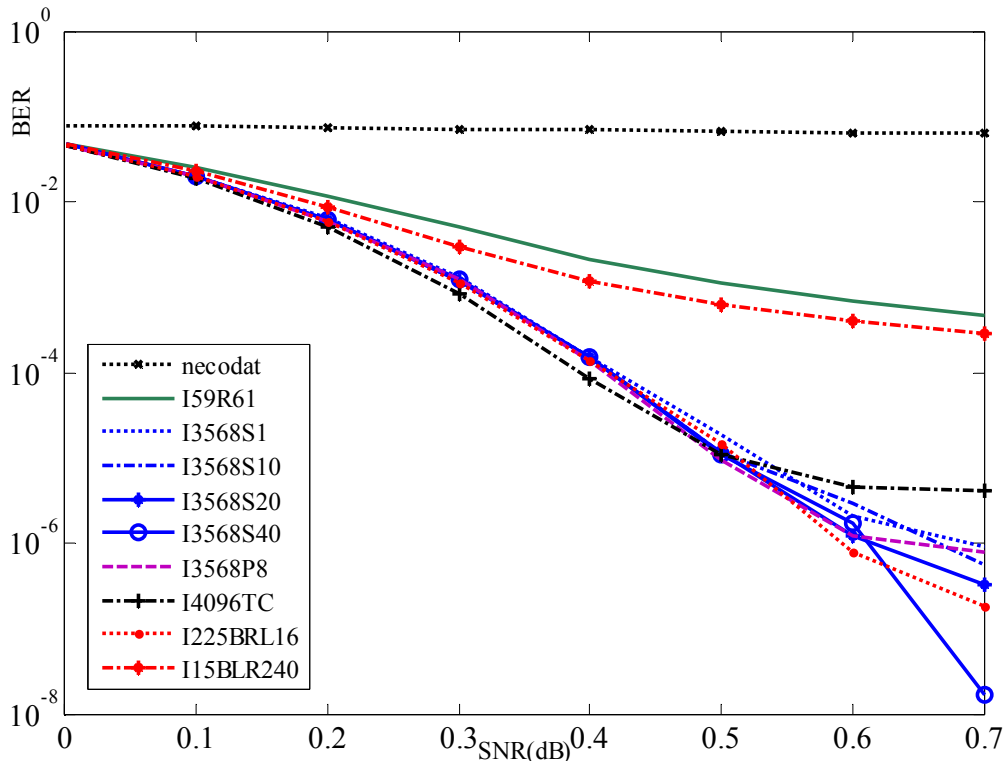


Fig. 3.8 e) Performanțele BER ale turbo-codurilor de rată 1/3, [1, 15/13], pentru interleaver-e cu $N=3600$.

În Fig. 3.8 se poate observa că performanțele interleaver-ilor S cresc odată cu S , distanța de interleav-are minimă. Deci, la un grad de împrăștiere dat (deoarece toate sunt interleaver-e aleatoare ele au același grad de împrăștiere) această distanță reprezintă o măsură a performanțelor.

În diagramele din Fig. 3.9 sunt reprezentate curbele corespunzătoare performanțelor FER, utilizând aceleași interleaver-e: interleaver aleator ($S=1$), interleaver S , interleaver rectangular (R), interleaver pseudo-aleator (P), interleaver Takeshita-Costello (TC) și interleaver-ele BLR și BRL.

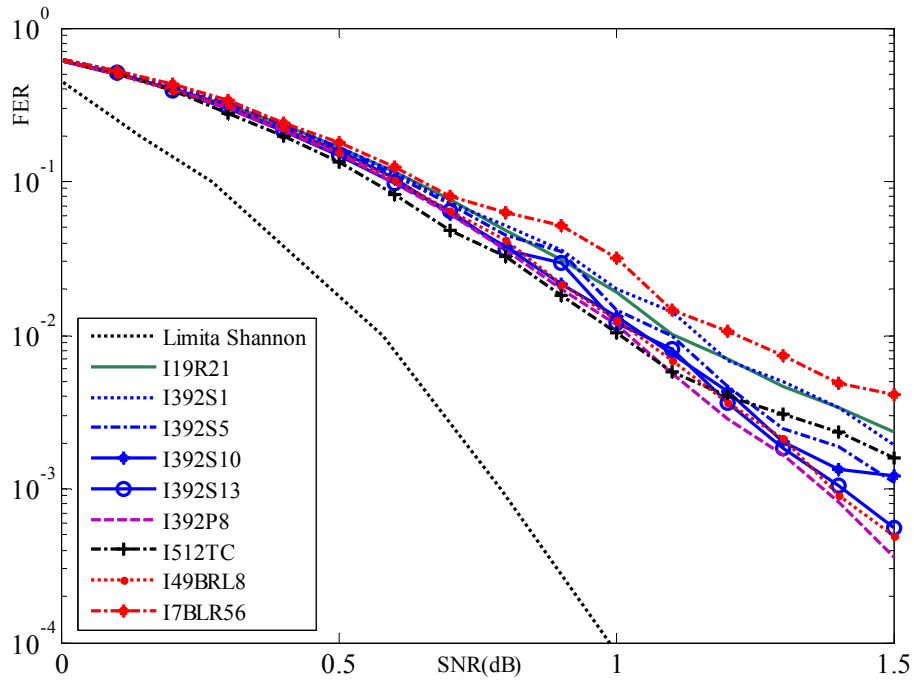


Fig. 3.9 a) Performanțele FER ale turbo-codurilor de rată 1/3, [1, 15/13], pentru interleaver-e cu $N=400$.

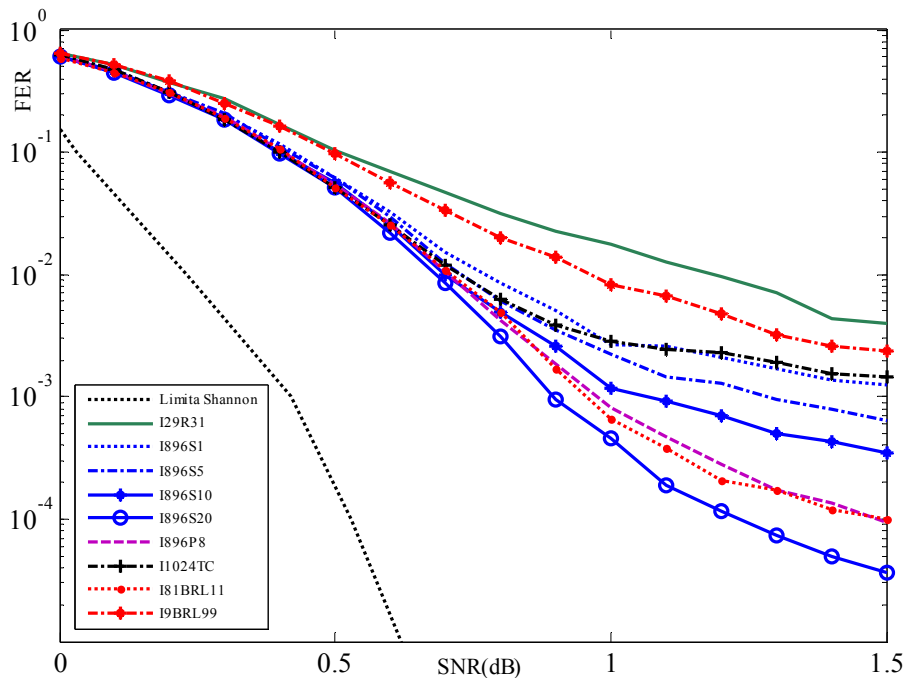


Fig. 3.9 b) Performanțele FER ale turbo-codurilor de rată 1/3, [1, 15/13], pentru interleaver-e cu $N=900$.

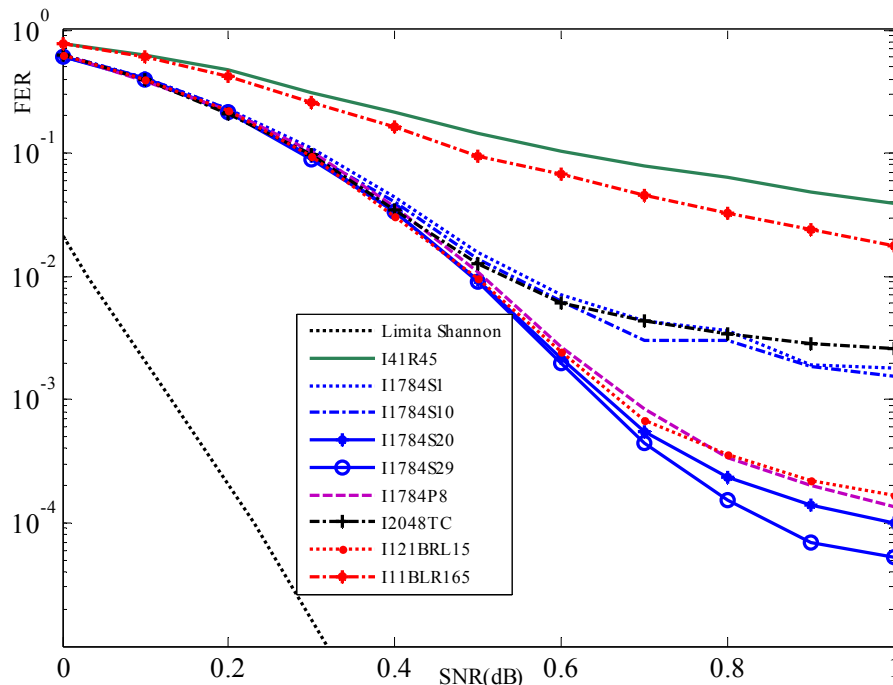


Fig. 3.9 c) Performanțele FER ale turbo-codurilor de rată 1/3, [1, 15/13], pentru interleaver-e cu $N=1800$.

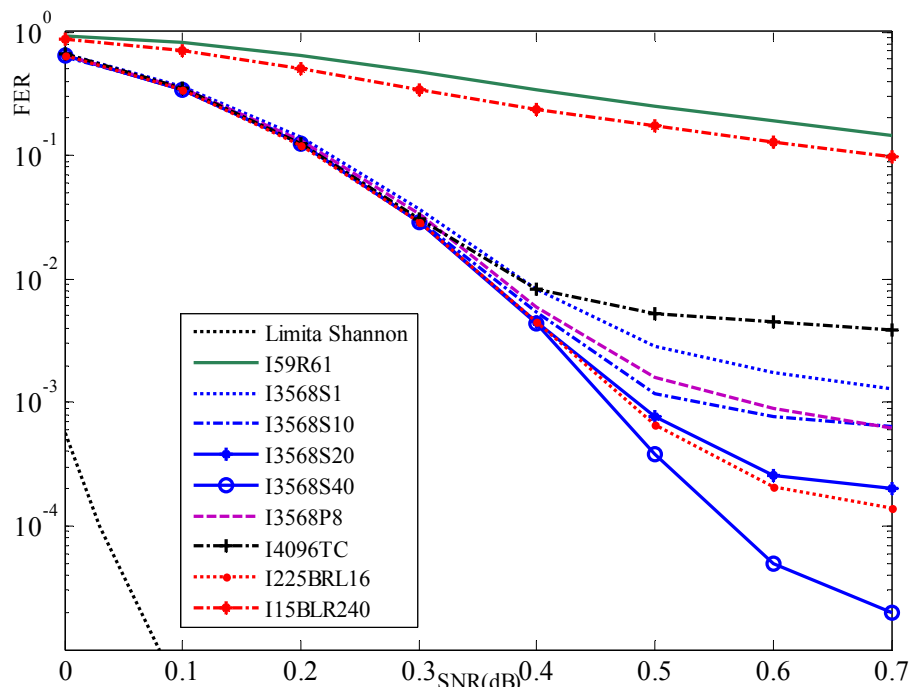


Fig. 3.9 d) Performanțele FER ale turbo-codurilor de rată 1/3, [1, 15/13], pentru interleaver-e cu $N=3600$.

Analizând curbele din Fig. 3.8 și Fig. 3.9, am observat că atunci când lungimea de întrețesere crește, gradul de împrăștiere devine factorul cel mai important în analiza performanțelor BER și FER, [KBN05, BAK04b]. Așadar, pentru lungimi mici de întrețesere, interleaver-ele au un "caracter bloc". În acest caz, interleaver-ele R, TC, BLR au performanțe similare cu celelalte. Cel mai bun interleaver este interleaver-ul TC. La o lungime de întrețesere egală cu 3600 aceste interleaver-e (R, TC, BLR) au cele mai slabe performanțe. În Fig. 3.8 se poate observa că performanțele interleaver-ilor S cresc odată cu S , distanța de interleavare minimă. Deci, la un grad de împrăștiere dat (deoarece toate sunt interleaver-e aleatoare ele au același grad de împrăștiere) această distanță reprezintă o măsură a performanțelor.

Interleaverul P (recomandat de CCSDS, [CCSDS02]), are performanțe bune la toate lungimile de întrețesere. Dar, cu creșterea lungimii de întrețesere, au performanțe mai scăzute în comparație cu interleaver-ele aleatoare.

Analizând performanțele interleaver-ilor bloc-aleatoare BLR și BRL, pot fi făcute următoarele observații. Interleaver-ul BLR are performanțe similare cu ale interleaver-ilor bloc. Performanțele interleaver-ului BRL sunt similare performanțelor interleaver-ilor S , având o distanță minimă comparabilă. Similaritățile pot fi explicate ținând cont de procedura de construcție. În construcția interleaver-ului BLR lungimea permutărilor aleatoare a fost egală cu jumătate din lungimea permutărilor aleatoare realizate în construcția interleaver-ului BRL. Mai mult, în cazul interleaver-ului BLR, toate coloanele sunt permutate în același fel. În construcția interleaver-ului BRL permutările sunt realizate independent de la linie la linie. În ciuda faptului că, la o lungime mare de întrețesere ($N > 1000$) interleaver-ul S , cu un S mare, este foarte bun, construcția sa este dificilă. În această analiză a fost utilizat doar un cod, [1, 15/13], dar această concluzie este valabilă pentru orice cod, [BAK04e]. Acesta este motivul pentru care interleaver-ul BRL propus este atractiv; are performanțe bune și o construcție simplă.

Așadar, se pot trage următoarele concluzii. Am propus două tipuri de interleaver-e, interleaver-ul bloc aleator în linie și interleaver-ul bloc cu linii aleatoare, și am comparat performanțele lor, BER și FER, cu ale altor interleaver-e cunoscute în literatură. Analiza performanțelor recomandă interleaver-ul BRL propus ca o alternativă la interleaver-ul S , care este cel mai bun la lungimi mai mari de 1000 biți. Performanțele interleaver-ului propus, BRL, sunt foarte apropiate de cele ale interleaver-ului S , cu S maxim, dar construcția noului interleaver este mai simplă. În simulările realizate, deoarece TC-urile operează la mai puțin de 1dB distanță de limita teoretică, s-a căutat obținerea unei precizii suficient de mari asupra curbelor BER/SNR (FER/SNR) în vederea unei comparații veridice, [BAK04c]

3.5 Algoritmi de decodare

Cuvintele de la ieșirea unui codor convoluțional sunt corelate și fiecare cuvânt este funcție de $m+1$ blocuri de informație. Pentru a decoda o secvență binară alcătuită din N cuvinte, este necesar să se considere secvența recepționată în ansamblul său. La ieșirea codorului sunt posibile doar anumite secvențe binare, ele corespunzând diferitelor căi ce există în trellis. Ca și pentru codurile bloc, decodarea unui cod convoluțional în prezența unui canal binar simetric va consta în a căuta în trellis, secvența binară (corespunzătoare unei căi particulare) cea mai apropiată de secvența recepționată. Această secvență este numită secvența cea mai probabilă.

Adoptând același criteriu ca și pentru codurile bloc, secvența emisă cea mai probabilă este cea care se găsește la distanță minimă față de secvența recepționată. Algoritmul Viterbi permite decodarea codurilor convoluționale, căutând secvența cea mai probabilă din trellis, și este bine adaptat la decodarea codurilor ce au lungimea de constrângere puțin mai mare (tipic, $K=7$), [WAD00]. Algoritmul Viterbi poate fi implementat atât cu decizie hard cât și cu decizie soft.

O variantă a algoritmului Viterbi, este SOVA (Soft Output Viterbi Algorithm – algoritmul Viterbi cu ieșire soft), [HAH89]. Acest algoritmul este folosit în special la schemele cu coduri concatenate și anume în turbo coduri.

În afara utilizării algoritmului SOVA, în turbo coduri se folosesc, ca și algoritmi de decodare, algoritmi: MAP, Max-Log-MAP și Log-MAP.

3.5.1 Algoritmul Viterbi

3.5.1.1 Algoritmul Viterbi cu decizie hard

Pentru prezentarea acestui algoritmul se consideră un canal binar simetric (fără memorie), intrarea decodorului fiind alcătuită dintr-o secvență de simboluri binare.

În fiecare moment două ramuri, aparținând la două căi diferite, converg spre fiecare nod al trellis-ului (Fig. 2.5). Din aceste două căi una este mai probabilă, altfel spus, se găsește la cea mai mică distanță Hamming față de secvența recepționată, decât cealaltă cale. Distanța fiind o funcțională aditivă, în fiecare nod se păstrează calea cea mai probabilă numită *cale supraviețuitoare*. Dacă se obțin două căi cu aceeași distanță Hamming, doar o singură cale este păstrată, alegându-se în mod arbitrar una din cele două căi posibile, [GLJ96].

În general este dificil să se aștepte ca toată secvența binară emisă să fie recepționată pentru a se începe operația de decodare. De fapt, aceasta va introduce o întârziere importantă la decodare, și va fi necesară o memorie de talie foarte mare pentru a memora toate căile supraviețuitoare.

În practică, un decodor Viterbi are o memorie finită ce poate fi văzută ca o fereastră, W , numită *fereastră de decodare*, ce păstrează o porțiune finită a trellis-ului. Observând derularea algoritmului Viterbi până la momentul curent $t=n$, se poate remarca că, urcând suficient în timp, căile supraviețuitoare converg aproape întodeauna spre același drum în $t=n-W$, Fig. 3.10.

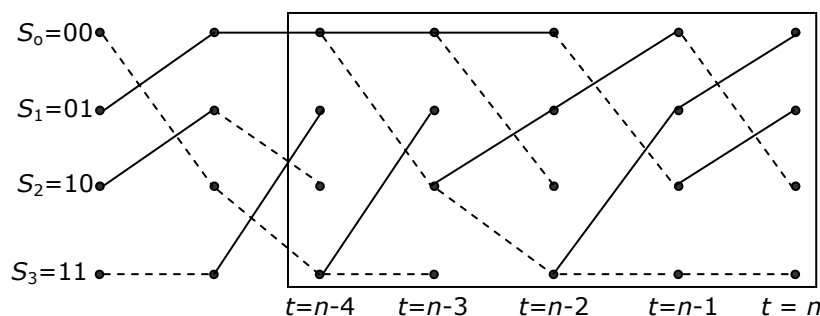


Fig. 3.10 Convergența supraviețuitorilor spre o cale unică în $t=n-4$ ($W=4$).

Astfel, pentru a decoda elementul binar de informație emis la $t=n-W$, rareori este necesar să se observe secvența recepționată începând de la $t=n$. Practic, memorarea supraviețuitorilor poate fi așadar limitată la un interval temporal de durată W , și astfel, întârzierea la decodare rămâne finită pentru a decoda o secvență infinită. În fiecare moment $t=n$, decodorul Viterbi furnizează o decizie cu privire la elementul binar de informație prezent la intrare codorului la momentul $t=n-W$. W trebuie să fie suficient de mare pentru a asigura o decizie corectă asupra celui mai vechi cadru transmis. Durata intervalului de timp necesar pentru obținerea unui drum unic este o variabilă aleatoare. Simulări pe calculator au demonstrat că durate de:

$$W \cong (4 + 5)K, \quad (3.17)$$

produc aproximări cu erori neglijabile comparativ cu o memorie infinită ($W \rightarrow \infty$) a decodorului, fapt pentru care decodările Viterbi practice au ferestre de decodare dimensionate după relația (3.17).

3.5.1.2 Algoritmul Viterbi cu decizie soft

În paragraful anterior, s-a prezentat algoritmul Viterbi considerând că intrarea decodorului a fost alcătuită dintr-un șir de simboluri binare (decizii ferme). Codurile convoluționale se comportă bine în *decodarea ponderată* utilizând algoritmul Viterbi. În cazul decodării ponderate intrarea decodorului este în mod sigur constituită de o secvență de simboluri analogice. Dacă se consideră cazul unei transmisii pe un canal cu zgomot aditiv alb și gaussian, aceste simboluri sunt gaussiene și necorelate, condiționând simbolurile binare emise de codor. Făcând această ipoteză distanța utilizată pentru măsurarea verosimilității fiecărei căi din trellis, este distanța euclidiană.

Utilizarea deciziilor ponderate nu modifică derularea algoritmului Viterbi, este suficientă înlocuirea distanțelor Hamming cu distanțele euclidiene în calculul metricii de ramură. Astfel, se calculează succesiv metricile de ramură și se acumulează valorile lor în metricile de cale. De exemplu, dacă pentru o ramură particulară se cunosc etichetele ramurilor (ieșirile codorului), a_1 și a_2 , și respectiv intrările decodorului corespunzătoare sunt w_1 și w_2 , rezultă metrica ramurii ca fiind:

$$b_m = \sum_{k=1}^2 (w_k - a_k)^2. \quad (3.18)$$

Se presupune că ieșirile codorului sunt +1 sau -1. Pentru simplitate, intrările decodorului w_{2j} și w_{2j+1} , din cadrul j , sunt luate ca și valori continue. Pentru $0 \leq j \leq K-2$, trellis-ul crește, și se utilizează tranziția de stare generală din Fig. 3.11.

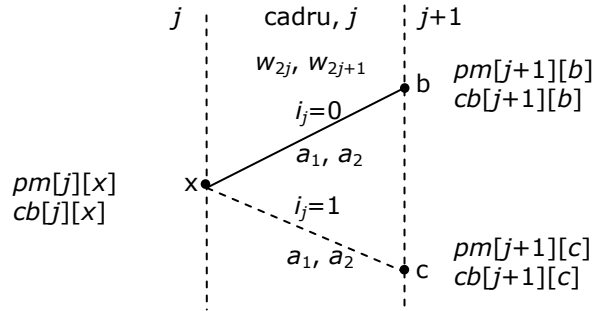


Fig. 3.11 Mărirea trellis-ului pentru un decodor Viterbi cu $R=1/2$.

Metrica de cale este atribuită fiecărui nod și pentru nodul b este:

$$pm[j + 1][b] = pm[j][x] + (w_{2j} - a_1)^2 + (w_{2j+1} - a_2)^2. \quad (3.19)$$

Aici, a_1 și a_2 sunt etichetele ramurii pentru cazul în care $i_j=0$ dă starea curentă n . Pe lângă cele două metrice de cale, se determină informația de înlănțuire înapoi (chainback information), pentru fiecare nod, în scopul de a stabili o cale probabilă înapoi prin trellis.

Avem:

$$cb[j + 1][b] = x \text{ și } cb[j + 1][c] = x ,$$

și astfel înlănțuirea înapoi indică starea anterioară.

Partea repetitivă a trellis-ului începe în cadrul $K-1$ și este reprezentată în Fig. 3.12:

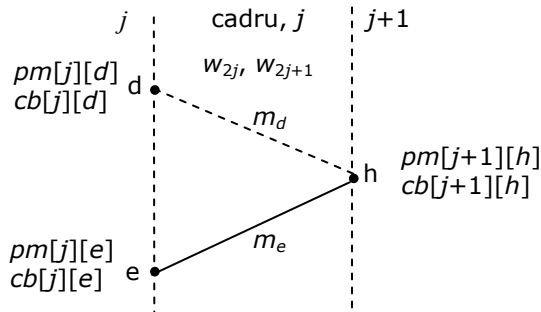


Fig. 3.12 Trellis repetitiv, pentru decodor Viterbi, $R=1/2$.

Pentru ramura dh se calculează o metrică de cale temporară:

$$m_d = pm[j][d] + (w_{2j} - a_1)^2 + (w_{2j+1} - a_2)^2 , \quad (3.20)$$

iar pentru ramura eh se calculează o altă metrică de cale temporală:

$$m_e = pm[j][e] + (w_{2j} - a_1)^2 + (w_{2j+1} - a_2)^2. \quad (3.21)$$

Dacă, $m_d < m_e$, rezultă :

$$pm[j+1][h] = m_d; \quad cb[j+1][h] = d. \quad (3.22)$$

În echipamentele de transmisie, decodarea ponderată nu utilizează chiar o secvență de simboluri analogice, ci o secvență de simboluri cuantificate și codate pe 3 sau 4 elemente binare. Această cuantificare pe 8 sau 16 nivele este în general suficientă pentru a se păstra la decodor toată capacitatea sa de corecție. În Anexa A se prezintă un exemplu de decodare Viterbi cu decizie soft.

Algoritmul Viterbi poate avea deasemenea și ieșire soft, [HAH89]. Această variantă se numește algoritmul Viterbi cu ieșire soft, SOVA.

Ieșirile soft ale decodarelor sunt tipic reprezentate în termenii așa numitelor rapoarte de plauzibilitate logaritmice. LLR-urile, așa cum reiese din denumirea lor, se definesc ca fiind logaritmii rapoartelor a două probabilități. De exemplul LLR-ul, $L(u_k)$ pentru valoarea unui bit decodat u_k , este dat de relația:

$$L(u_k) = \ln \left(\frac{P(u_k = +1)}{P(u_k = -1)} \right),$$

unde $P(u_k = +1)$ este probabilitatea bitului $u_k = +1$ și $P(u_k = -1)$ este probabilitatea bitului $u_k = -1$. De observat că cele două valori posibile ale bitului u_k sunt alese să fie $+1$ și -1 și nu 1 și 0 deoarece aceasta simplifică explicațiile ce urmează.

3.5.2 Algoritmul MAP

A. Introducere și elemente de matematică

Bahl, Cocke, Jelinek și Raviv au propus, în 1974 [BCJR74], un algoritm cunoscut sub numele de algoritmul MAP sau BCJR, pentru a determina probabilitățile *a-posteriori* ale stărilor și tranzițiilor unei surse Markov [ASH65], afectată de un zgomot necorelat. Ei au arătat cum poate fi folosit algoritmul pentru decodarea atât a codurilor convoluționale cât și a codurilor bloc. Când este folosit pentru decodarea codurilor convoluționale, algoritmul este optimal, în sensul minimizării BER la decodare, spre deosebire de algoritmul Viterbi care minimizează probabilitatea de cale incorectă prin trellis, selectată de decodor. Astfel, algoritmul Viterbi poate fi privit ca minimizând numărul de *grupuri* de biți asociați căii incorecte și nu numărul de biți decodați incorect. De altfel, în majoritatea aplicațiilor, performanțele celor doi algoritmi sunt aproape identice [BCJR74]. Algoritmul MAP examinează fiecare din căile posibile prin trellis-ul decodului convoluțional și de aceea este extrem de complex. Această complexitate a împiedicat folosirea sa în majoritatea sistemelor. Situația s-a schimbat însă odată cu descoperirea turbo codurilor.

Algoritmul MAP generează nu numai secvența estimată de biți ci și probabilitățile ca fiecare bit să fi fost corect decodat. Acest lucru a fost esențial pentru decodarea iterativă propusă de Berrou ș.a., astfel că în prima lor lucrare a fost folosit algoritmul MAP. Ulterior au fost depuse mari eforturi pentru reducerea la un nivel rezonabil a complexității algoritmului MAP. În cele ce urmează, vor fi expuse bazele teoretice ale algoritmului MAP folosit pentru decodarea ieșirilor soft ale codurilor convoluționale ce compun un turbo cod. Se presupune că se folosesc coduri binare.

Algoritmul MAP generează pentru fiecare bit u_k decodat, probabilitatea ca acest bit să fi fost +1 sau -1, având dată secvența de simboluri \underline{y} . Acest lucru este echivalent cu găsirea lui $L(u_k/\underline{y})$ sau LLR *a-posteriori* [HLY02], adică:

$$L(u_k|\underline{y}) = \ln \left(\frac{P(u_k = +1|\underline{y})}{P(u_k = -1|\underline{y})} \right). \quad (3.23.a)$$

Ținând cont de relația lui Bayes (Anexa B), relația (3.23.a) poate fi rescrisă:

$$L(u_k|\underline{y}) = \ln \left(\frac{P(u_k = +1 \wedge \underline{y})}{P(u_k = -1 \wedge \underline{y})} \right). \quad (3.23.b)$$

Dacă în trellis sunt cunoscute starea precedentă $S_{k-1}=s'$ și starea prezentă $S_k=s$ atunci va fi cunoscut și bitul de intrare u_k care determină tranziția între aceste stări. Acest lucru, împreună cu regula lui Bayes și faptul că tranzițiile dintre starea precedentă S_{k-1} și starea curentă S_k sunt mutual exclusive (în codor poate apărea doar una dintre ele), permite rescrierea ecuației (3.23.b) sub forma :

$$L(u_k|\underline{y}) = \ln \left(\frac{\sum_{(s',s) \Rightarrow u_k=+1} P(S_{k-1}=s' \wedge S_k=s \wedge \underline{y})}{\sum_{(s',s) \Rightarrow u_k=-1} P(S_{k-1}=s' \wedge S_k=s \wedge \underline{y})} \right), \quad (3.24)$$

unde $(s',s) \Rightarrow u_k=+1$ este setul tranzițiilor din starea precedentă $S_{k-1}=s'$ în starea prezentă $S_k=s$ care pot apărea dacă bitul de intrare $u_k=+1$ și la fel $(s',s) \Rightarrow u_k=-1$. Pentru concizie se rescrie $P(S_{k-1}=s' \wedge S_k=s \wedge \underline{y})$ ca $P(s' \wedge s \wedge \underline{y})$.

Se consideră în continuare probabilitățile individuale $P(s' \wedge s \wedge \underline{y})$ de la numărătorul și numitorul ecuației (3.24). Secvența recepționată \underline{y} poate fi separată în trei părți: cuvântul de cod recepționat asociat tranziției \underline{y}_k prezente, secvența recepționată anterior tranziției prezente $\underline{y}_{j<k}$ și secvența care va fi recepționată după tranziția prezentă $\underline{y}_{j>k}$. Se pot scrie astfel probabilitățile individuale :

$$P(s' \wedge s \wedge \underline{y}) = P(s' \wedge s \wedge \underline{y}_{j<k} \wedge \underline{y}_k \wedge \underline{y}_{j>k}) \quad (3.25)$$

În Fig. 3.13 este prezentată o secțiune din trellis-ul cu patru stări pentru un cod RSC cu constrângerea $K=3$, fiind evidențiate secvențele recepționate din canal. Cu linie continuă sunt reprezentate tranzițiile rezultate ca urmare a apariției lui -1 ca bit de intrare, iar cu linie punctată tranzițiile pentru +1 ca bit de intrare. Parametrii $\alpha_{k-1}(s')$, $\gamma_k(s',s)$ și $\beta_k(s)$ reprezintă valori calculate de algoritmul MAP, care vor fi definite în cele ce urmează.

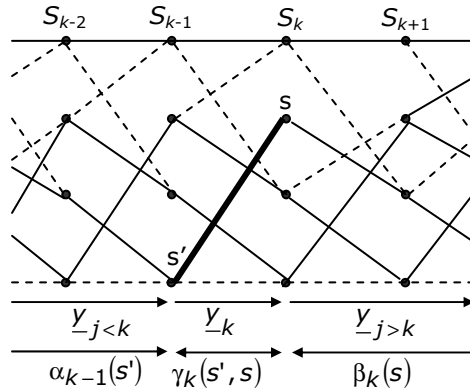


Fig. 3.13 Decodor MAP pentru codul RSC, $K=3$.

Se folosește relația $P(a \wedge b) = P(a/b)P(b)$ consecință a regulii lui Bayes și se folosește un canal fără memorie. Rezultă ca viitoarea secvență recepționată $y_{j>k}$ va depinde doar de starea prezentă s și nu de starea anterioară s' sau de secvența recepționată anterior $y_{j<k}$ respectiv secvența prezentă y_k , deci se poate scrie:

$$\begin{aligned}
 P(s' \wedge s \wedge \underline{y}) &= P(s' \wedge s \wedge y_{j < k} \wedge y_k \wedge y_{j > k}) = \\
 &= P(y_{j > k} | s) P(s' \wedge s \wedge y_{j < k} \wedge y_k) = \\
 &= P(y_{j > k} | s) P(y_k \wedge s | s') P(s' \wedge y_{j < k}) = \\
 &= \beta_k(s) \gamma_k(s', s) \alpha_{k-1}(s'),
 \end{aligned} \tag{3.26}$$

unde :

$$\alpha_{k-1}(s') = P(S_{k-1} = s' \wedge y_{j < k}) \tag{3.27}$$

este probabilitatea ca trellis-ul să fie în starea s' la momentul $k-1$ și secvența recepționată din canal până în acest punct este $y_{j < k}$ (vezi Fig. 3.13),

$$\beta_k(s) = P(y_{j > k} | S_k = s) \tag{3.28}$$

este probabilitatea ca, dacă la momentul k trellis-ul se află în starea s , viitoarea secvență recepționată din canal să fie $\underline{y}_{j>k}$, și în final,

$$\gamma_k(s', s) = P(\{\underline{y}_{\underline{k}} \wedge S_k = s\} | S_{k-1} = s') \quad (3.29)$$

este probabilitatea ca, dacă trellis-ul se află în starea s' la momentul $k-1$, el să treacă în starea s și secvența recepționată din canal pentru această tranziție să fie $\underline{y}_{\underline{k}}$.

Ecuția (3.26) arată că probabilitatea $P(s' \wedge s \wedge \underline{y})$ de trecere a codorului din starea $S_{k-1} = s'$ în starea $S_k = s$ și de recepționare a secvenței \underline{y} , poate fi descompusă într-un produs de factori, $\alpha_{k-1}(s')$, $\gamma_k(s', s)$ și $\beta_k(s)$. Semnificația celor trei factori este dată în Fig. 3.13, unde este marcată cu linie îngroșată tranziția din $S_{k-1} = s'$ în starea $S_k = s$. Din ecuațiile (3.24) și (3.26) se poate scrie pentru LLR condiționată a lui u_k , fiind dată secvența \underline{y}_k :

$$\begin{aligned} L(u_k | \underline{y}) &= \ln \left(\frac{\sum_{(s', s) \Rightarrow u_k = +1} P(S_{k-1} = s' \wedge S_k = s \wedge \underline{y})}{\sum_{(s', s) \Rightarrow u_k = -1} P(S_{k-1} = s' \wedge S_k = s \wedge \underline{y})} \right) = \\ &= \ln \left(\frac{\sum_{(s', s) \Rightarrow u_k = +1} \alpha_{k-1}(s') \gamma_k(s', s) \beta_k(s)}{\sum_{(s', s) \Rightarrow u_k = -1} \alpha_{k-1}(s') \gamma_k(s', s) \beta_k(s)} \right). \end{aligned} \quad (3.30)$$

Algoritmul MAP găsește $\alpha_k(s)$, și $\beta_k(s)$ pentru toate stările s prin trellis, de exemplu pentru $k=0, 1, \dots, N-1$ și $\gamma_k(s', s)$ pentru toate tranzițiile posibile din starea $S_{k-1} = s'$ în starea $S_k = s$, din nou pentru $k=0, 1, \dots, N-1$. Aceste valori sunt apoi folosite cu relația (3.30), pentru a obține LLR-urile condiționate $L(u_k / \underline{y})$ pe care decodorul MAP le furnizează. În cele ce urmează este descris modul de calcul al valorilor $\alpha_k(s)$, $\beta_k(s)$ și $\gamma_k(s', s)$.

B. Calculul valorilor $\alpha_k(s)$ prin recursivitate înainte (la ieșire)

Din definiția lui $\alpha_{k-1}(s')$, (relația (3.27)), se poate scrie:

$$\begin{aligned} \alpha_k(s) &= P(S_k = s \wedge \underline{y}_{j < k+1}) = \\ &= P(s \wedge \underline{y}_{j < k} \wedge \underline{y}_{\underline{k}}) = \\ &= \sum_{\text{toti } s'} P(s \wedge s' \wedge \underline{y}_{j < k} \wedge \underline{y}_{\underline{k}}), \end{aligned} \quad (3.31)$$

unde, în ultima linie s-a descompus probabilitatea $P(s \wedge y_{j < k})$ într-o sumă de probabilități mutuale $P(s \wedge s' \wedge y_{j < k})$ pe toate stările posibile precedente s' . Folosind regula lui Bayes și ipoteza că, canalul este fără memorie, se poate proceda astfel:

$$\begin{aligned}
 \alpha_k(s) &= \sum_{\text{toți } s'} P(s \wedge s' \wedge y_{j < k} \wedge y_k) = \\
 &= \sum_{\text{toți } s'} P(\{s \wedge y_k\} | \{s' \wedge y_{j < k}\}) P(s' \wedge y_{j < k}) = \\
 &= \sum_{\text{toți } s'} P(\{s \wedge y_k\} | s') P(s' \wedge y_{j < k}) = \\
 &= \sum_{\text{toți } s'} \alpha_{k-1}(s') \gamma_k(s', s).
 \end{aligned} \tag{3.32}$$

Astfel, odată ce valorile $\gamma_k(s', s)$ sunt cunoscute, valorile $\alpha_k(s)$ pot fi calculate recursiv. Presupunând că trellis-ul are starea inițială $S_0=0$, condițiile inițiale pentru calculul recursiv sunt:

$$\begin{aligned}
 \alpha_0(S_0=0) &= 1 \\
 \alpha_0(S_0=s) &= 0 \text{ pentru } s \neq 0.
 \end{aligned} \tag{3.33}$$

În Fig. 3.14 este prezentat un exemplu de calcul al unei valori $\alpha_k(s)$ pentru $s=0$, în mod recursiv, folosind valorile lui $\alpha_{k-1}(s')$ și $\gamma_k(s', s)$, pentru codul RSC cu $K=3$. De menționat este faptul că, deoarece trellis-ul este binar, doar două stări precedente și anume $S_{k-1}=0$ și $S_{k-1}=1$ au cale spre starea $S_k=0$, astfel că suma din relația (3.32) are doar doi termeni.

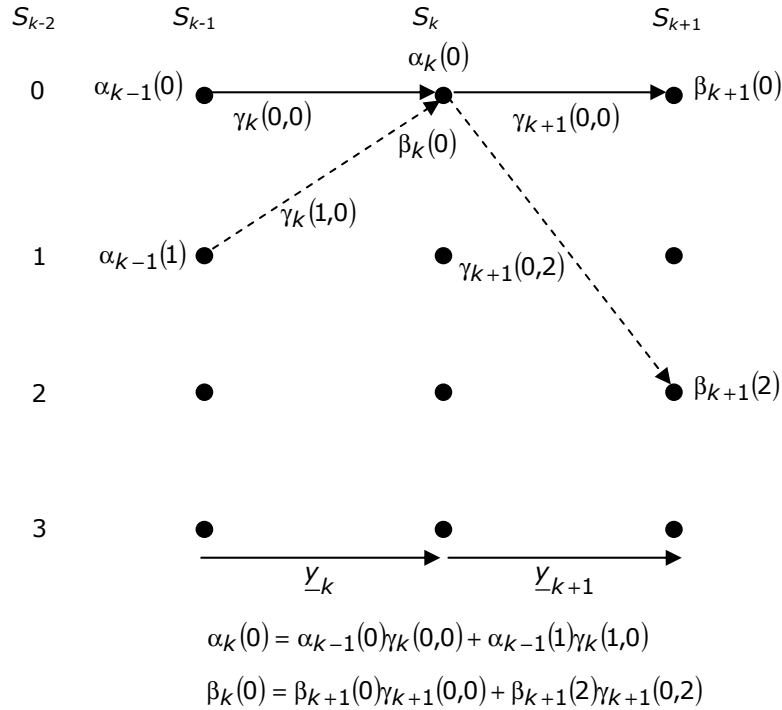


Fig. 3.14 Calculul recursiv al lui $\alpha_k(0)$ și al lui $\beta_k(0)$.

C. Calculul recursiv înapoi al valorilor $\beta_k(s)$

Valorile $\beta_k(s)$ pot fi calculate recursiv, în mod asemănător. Folosind o deducere similară cu cea a relației (3.32), se poate arăta că:

$$\beta_{k-1}(s') = P(\underline{y}_{-j > k-1} | S_k = s') = \sum_{\text{toti } s} \beta_k(s) \gamma_k(s', s). \tag{3.34}$$

Astfel că, odată ce sunt cunoscute valorile $\gamma_k(s', s)$ poate fi folosită o recursivitate înapoi pentru a calcula valorile lui $\beta_{k-1}(s')$ din valorile lui $\beta_k(s)$ folosind relația (3.34). În Fig. 3.14 se vede calculul recursiv al lui $\beta_k(0)$ folosind valorile lui $\beta_{k+1}(s)$ și $\gamma_{k+1}(0, s)$ pentru codul RSC cu $K=3$ considerat.

D. Calculul valorilor lui $\gamma_k(s', s)$

În continuare se poate vedea cum pot fi calculate valorile $\gamma_k(s', s)$ ale probabilității tranziției, din relația (3.26), din secvența recepționată din canal și orice

informație disponibilă *a-priori*. Folosind definiția lui $\gamma_k(s',s)$ (relația (3.29)) și regula lui Bayes, rezultă:

$$\begin{aligned}\gamma_k(s',s) &= P(\{\underline{y}_k \wedge s | s'\}) = \\ &= P(\underline{y}_k | \{s' \wedge s\})P(s|s') = \\ &= P(\underline{y}_k | \{s' \wedge s\})P(u_k) = \\ &= P(\underline{y}_k | \underline{x}_k)P(u_k),\end{aligned}\tag{3.35 a}$$

unde:

- u_k este bitul de intrare necesar pentru a determina tranziția din starea $S_{k-1}=s'$ în starea $S_k=s$,
- $P(u_k)$ este probabilitatea *a-priori* a acestui bit,
- \underline{x}_k este cuvântul de cod transmis asociat acestei tranziții.

Astfel, probabilitatea tranziției $\gamma_k(s',s)$ este dată de produsul dintre probabilitatea *a-priori* a bitului de intrare u_k necesar pentru această tranziție și probabilitatea ca având dat cuvântul de cod \underline{x}_k care s-a transmis, asociat acestei tranziții, să recepționăm secvența de canal \underline{y}_k . Probabilitatea *a-priori* $P(u_k)$:

$$P(u_k) = \left(\frac{e^{-L(u_k)/2}}{1 + e^{-L(u_k)}} \right) \cdot e^{(u_k L(u_k)/2)}\tag{3.35 b}$$

este dedusă într-un decodor iterativ din ieșirea decodului component precedent, și probabilitatea condiționată a secvenței recepționate $P(\underline{y}_k | \underline{x}_k)$ care este dată, în ipoteza canalului gaussian fără memorie cu modulație BPSK, de:

$$\begin{aligned}P(\underline{y}_k | \underline{x}_k) &= \prod_{l=1}^n P(y_{kl} | x_{kl}) = \\ &= \prod_{l=1}^n \frac{1}{\sqrt{2\pi\sigma}} \cdot e^{\left(-\frac{E_b}{2\sigma^2} (y_{kl} - ax_{kl})^2 \right)} = \\ &= \frac{1}{(\sqrt{2\pi\sigma})^n} \cdot e^{\left(-\frac{E_b}{2\sigma^2} \sum_{l=1}^n (y_{kl}^2 + a^2 x_{kl}^2 - 2ax_{kl}y_{kl}) \right)} = \\ &= \frac{1}{(\sqrt{2\pi\sigma})^n} \cdot e^{\left(-\frac{E_b}{2\sigma^2} \sum_{l=1}^n y_{kl}^2 \right)} \cdot e^{\left(-\frac{E_b}{2\sigma^2} a^2 \sum_{l=1}^n x_{kl}^2 \right)} \cdot e^{\left(\frac{E_b}{2\sigma^2} 2a \sum_{l=1}^n y_{kl}x_{kl} \right)},\end{aligned}\tag{3.35 c}$$

unde:

$-x_{kl}$ și y_{kl} sunt biții individuali, din cuvintele de cod \underline{x}_k transmis și \underline{y}_k recepționat;

- n este numărul biților din fiecare cuvânt de cod;
- E_b este energia per bit transmisă;
- σ^2 este dispersia zgomotului (varianța);
- a este amplitudinea fading-ului; pentru canale fără fading, de tip AWGN:

$a=1$.

Relația (3.35 a), ținând cont de relațiile (3.35 b) și (3.35 c) poate fi rescrisă astfel:

$$\begin{aligned}
 \gamma_k(s', s) &= P(u_k) \cdot P(\underline{y}_k | \underline{x}_k) = \\
 &= \left(\frac{e^{-L(u_k)/2}}{1 + e^{-L(u_k)}} \right) \cdot e^{(u_k L(u_k)/2)} = \\
 &= \frac{1}{(\sqrt{2\pi\sigma})^n} \cdot e^{\left(-\frac{E_b}{2\sigma^2} \sum_{l=1}^n y_{kl}^2 \right)} \cdot e^{\left(-\frac{E_b}{2\sigma^2} a^2 \sum_{l=1}^n x_{kl}^2 \right)} \cdot e^{\left(\frac{E_b}{2\sigma^2} 2a \sum_{l=1}^n y_{kl} x_{kl} \right)} = \quad (3.35 d) \\
 &= C \cdot e^{(u_k L(u_k)/2)} \cdot e^{\left(\frac{E_b}{2\sigma^2} 2a \sum_{l=1}^n y_{kl} x_{kl} \right)} = \\
 &= C \cdot e^{(u_k L(u_k)/2)} \cdot e^{\left(\frac{L_c}{2} \sum_{l=1}^n y_{kl} x_{kl} \right)},
 \end{aligned}$$

unde mărimea C nu depinde de semnul bitului u_k sau de cuvântul de cod transmis \underline{x}_k , iar:

$$L_c = 4a \frac{E_b}{2\sigma^2} \quad (3.36)$$

este valoarea de încredere a canalului.

E. Sumar al algoritmului MAP

Din cele prezentate se vede că decodarea MAP a secvenței recepționate \underline{y} pentru a obține LLR *a-posteriori*, $L(u_k | \underline{y})$ poate fi rezumată după cum urmează. Pe măsura recepționării valorilor y_{kl} , ele sunt folosite împreună cu LLR-urile *a-priori* $L(u_k)$ (care într-un decodor turbo iterativ sunt furnizate de celălalt decodor component) pentru a calcula $\gamma_k(s', s)$ conform relațiilor (3.35). Pe măsura recepționării din canal a valorilor y_{kl} și calculării valorilor $\gamma_k(s', s)$, se poate folosi recursivitatea înainte (relația (3.32)), pentru a calcula $\alpha_k(s', s)$. Odată ce toate valorile din canal au fost recepționate și au fost calculate $\gamma_k(s', s)$ pentru toți $k=1, 2, \dots, N$, poate fi folosită recursivitatea înapoi (relația (3.34)) pentru a calcula valorile $\beta_k(s', s)$. În final, toate valorile calculate pentru $\alpha_k(s', s)$, $\beta_k(s', s)$ și $\gamma_k(s', s)$ sunt folosite în relația (3.30) pentru a calcula $L(u_k | \underline{y})$. Aceste operații sunt prezentate schematic în Fig. 3.15.

Algoritmul MAP este extrem de complex, din punct de vedere al calculului, în forma descrisă, din cauza înmulțirilor necesare în relațiile (3.32) și (3.34), pentru calculul recursiv al valorilor $\alpha_k(s',s)$ și $\beta_k(s',s)$, a înmulțirii și exponențialelor necesare la calculul lui $\gamma_k(s',s)$ folosind relația (3.35 d), precum și a înmulțirilor și logaritmului natural necesare pentru calculul lui $L(u_k|\underline{y})$, folosind relația (3.30). De aceea, s-a depus un efort continuu pentru reducerea complexității de calcul a algoritmului, și s-a ajuns la algoritmul Log-MAP propus de Robertson, ș.a. având aceleași performanțe ca și algoritmul MAP, dar cu o complexitate semnificativ mai mică și fără problemele de scalare menționate.

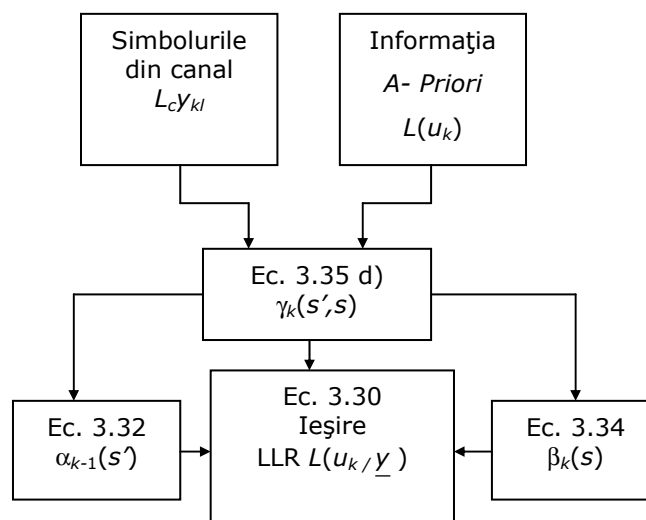


Fig. 3.15 Sumarul operațiilor în algoritmul MAP

3.5.3 Algoritmul Max-Log-MAP

În scopul reducerii complexității algoritmul MAP, au fost creați algoritmi Max-Log-MAP și Log-MAP. Inițial, algoritmul Max-Log-MAP a fost propus de Koch și Baier [KOB90] și Erfanian ș.a. [EPG94]. Această tehnică simplifică algoritmul MAP prin transferarea recursivității în domeniul logaritmic și prin utilizarea aproximării în scopul reducerii drastice a complexității. Din cauza acestei aproximări performanțele sale sunt sub-optimale în comparație cu cele ale algoritmul MAP. Algoritmul MAP calculează LLR-urile *a-posteriori* $L(u_k|\underline{y})$, utilizând relația (3.30). În acest scop se cer următoarele valori:

1. $\alpha_{k-1}(s')$, care sunt calculate prin recursivitate înainte, utilizând relația (3.32);
2. $\beta_k(s)$, care sunt calculate prin recursivitate înapoi, utilizând relația (3.34);
3. probabilitățile tranzițiilor ramurii $\gamma_k(s',s)$, care sunt calculate utilizând relația (3.35a).

Algoritmul Max-Log-MAP simplifică aceasta prin transferarea acestor ecuații în domeniul logaritmic și utilizarea apoi a aproximării:

$$\ln \left(\sum_i e^{x_i} \right) \approx \max_i(x_i), \quad (3.37)$$

unde $\max_i(x_i)$ reprezintă valoarea maximă a lui x_i . Apoi, cu $A_k(s)$, $B_k(s)$, $\Gamma_k(s',s)$, definite după cum urmează:

$$A_k(s) \stackrel{\Delta}{=} \ln(\alpha_k(s)), \quad (3.38)$$

$$B_k(s) \stackrel{\Delta}{=} \ln(\beta_k(s)), \quad (3.39)$$

$$\Gamma_k(s',s) \stackrel{\Delta}{=} \ln(\gamma_k(s',s)), \quad (3.40)$$

se poate rescrie relația (3.32) astfel:

$$\begin{aligned} A_k(s) &\stackrel{\Delta}{=} \ln(\alpha_k(s)) = \\ &= \ln \left(\sum_{\text{toti } s'} \alpha_{k-1}(s') \gamma_k(s',s) \right) = \\ &= \ln \left(\sum_{\text{toti } s'} \exp[A_{k-1}(s') + \Gamma_k(s',s)] \right) \approx \\ &\approx \max_{s'} (A_{k-1}(s') + \Gamma_k(s',s)). \end{aligned} \quad (3.41)$$

Ecuția (3.41) implică faptul că pentru fiecare cale din Fig. 3.15, de la starea anterioară a trellis-ului la starea actuală $S_k=s$, algoritmul adună termenul metrică a ramurii la valoarea anterioară $A_{k-1}(s)$, pentru a găsi noua valoare $\tilde{A}_k(s)$ pentru acea cale. Noua valoare a lui $A_k(s)$, corespunzător relației (3.41) este atunci maximul valorilor $\tilde{A}_k(s)$ a diferitelor căi ce intră în starea $S_k=s$. Aceasta poate fi văzută ca alegerea unei căi ca și supraviețuitoare din totalul căilor ce intră în starea respectivă. Valoarea lui $A_k(s)$ va da logaritmul natural al probabilității ca trellis-ul să fie în starea $S_k=s$, la momentul k , știind că secvența canalului recepționat până în acest punct a fost $\underline{y}_{j \leq k}$. Datorită aproximării din relația (3.37), utilizată în

obținerea relației (3.41), este considerată doar calea de probabilitate maximă (Maximum Likelihood) din starea $S_k=s$, când se calculează această probabilitate. Această valoare pentru $A_k(s)$, în algoritmul Max-Log-MAP, dă acum probabilitatea celei mai plauzibile căi din trellis pentru starea $S_k=s$, ignorând probabilitatea pentru oricare altă cale prin trellis, către starea $S_k=s$. Această aproximație este unul dintre motivele performanțelor sub-optimale ale algoritmului Max-Log-MAP comparativ cu algoritmul MAP.

Se observă din ecuația (3.41) că recurența înainte utilizată în algoritmul Max-Log-MAP, pentru a calcula $A_k(s)$, este exact aceeași ca recurența înainte utilizată în algoritmul Viterbi, pentru fiecare pereche de căi convergente,

supraviețuitorul se găsește utilizând două însumări și o comparație. Pentru trellisurile binare sumarea și adunarea s-au realizat peste toate stările anterioare $S_{k-1}=s'$, în relația (3.41) vor fi de fapt doar două stări, deoarece vor exista doar două stări anterioare $S_{k-1}=s'$, cu căi către starea prezentă $S_k=s$. Pentru toate celelalte valori ale lui s' vom avea $\Gamma_k(s',s)=0$.

Similar ecuației (3.41) pentru recurența înainte utilizată în calculul lui $A_k(s)$, se poate scrie ecuația (3.34) în forma:

$$\begin{aligned}
 B_{k-1}(s') & \stackrel{\Delta}{=} \ln(\beta_{k-1}(s')) = \\
 & = \ln \left(\sum_{\text{toti } s} \beta_k(s) \gamma_k(s', s) \right) = \\
 & = \ln \left(\sum_{\text{toti } s} \exp[B_k(s) + \Gamma_k(s', s)] \right) \approx \\
 & \approx \max_s (B_k(s) + \Gamma_k(s', s)),
 \end{aligned} \tag{3.42}$$

obținându-se recurența înapoi utilizată pentru calculul valorilor $B_{k-1}(s')$. De asemenea, aceasta este echivalentă cu recurența utilizată în algoritmul Viterbi, doar că aici se realizează înapoi și nu înainte pe trellis. Astfel, valoarea lui $B_{k-1}(s')$ se află prin adunarea, pentru orice stare $S_k=s$ ce are o cale dinspre $S_{k-1}=s'$ (două în trellisul binar), a metricii ramurii $\Gamma_k(s',s)$ la valoarea $B_k(s)$ și selectând cea care dă cea mai mare valoare $B_{k-1}(s')$.

Utilizând relațiile (3.35) și (3.36), se pot scrie metricile ramurilor $\Gamma_k(s',s)$ în ecuațiile recursive de mai sus, rezultate pentru $A_k(s)$ și $B_{k-1}(s')$, ca:

$$\begin{aligned}
 \Gamma_k(s', s) & \stackrel{\Delta}{=} \ln(\gamma_k(s', s)) = \\
 & = \ln \left(C \cdot e^{(u_k L(u_k)/2)} \exp \left[\frac{E_b}{2\sigma^2} 2a \sum_{l=1}^n u_{kl} x_{kl} \right] \right) = \\
 & = \ln \left(C \cdot e^{(u_k L(u_k)/2)} \exp \left[\frac{L_c}{2} \sum_{l=1}^n u_{kl} x_{kl} \right] \right) = \\
 & = C' + \frac{1}{2} u_k L(u_k) + \frac{L_c}{2} \sum_{l=1}^n y_{kl} x_{kl},
 \end{aligned} \tag{3.43}$$

unde $C' = \ln C$ nu depinde de u_k sau de cuvântul de cod transmis \underline{x}_k și astfel poate fi constant și omis. Așadar, metrica ramurii este echivalentă cu cea utilizată în algoritmul Viterbi, cu adunarea termenului LLR *a-priori* $u_k L(u_k)$. În plus, termenul corelație $\sum_{l=1}^n y_{kl} x_{kl}$ este ponderat de valoarea de încredere a canalului [PRO00, HAG95], L_c , care depinde doar de raportul semnal/zgomot din canal și de a , amplitudinea fading-ului din canal (pentru AWGN, $a=1$).

În final, din ecuația (3.30), se pot scrie LLR-urile *a-posteriori*, $L(u_k | \underline{y})$, care se calculează cu algoritmul Max-Log-MAP:

$$\begin{aligned}
 L(u_k | \underline{y}) &= \ln \left(\frac{\sum_{(s',s) \Rightarrow u_k = +1} \alpha_{k-1}(s') \gamma_k(s',s) \beta_k(s)}{\sum_{(s',s) \Rightarrow u_k = -1} \alpha_{k-1}(s') \gamma_k(s',s) \beta_k(s)} \right) = \\
 &= \ln \left(\frac{\sum_{(s',s) \Rightarrow u_k = +1} \exp(A_{k-1}(s') + \Gamma_k(s',s) + B_k(s))}{\sum_{(s',s) \Rightarrow u_k = -1} \exp(A_{k-1}(s') + \Gamma_k(s',s) + B_k(s))} \right) \approx \quad (3.44) \\
 &\approx \max_{(s',s) \Rightarrow u_k = +1} (A_{k-1}(s') + \Gamma_k(s',s) + B_k(s)) - \\
 &\quad - \max_{(s',s) \Rightarrow u_k = -1} (A_{k-1}(s') + \Gamma_k(s',s) + B_k(s)).
 \end{aligned}$$

Aceasta înseamnă că în algoritmul Max-Log-MAP, pentru fiecare bit u_k , LLR-ul *a-posteriori* $L(u_k | \underline{y})$ se calculează considerând toate tranzițiile de la starea trellis-ului S_{k-1} la starea S_k . Aceste tranziții sunt grupate între acelea ce s-ar fi putut produce doar dacă $u_k = +1$, și acelea ce s-ar fi putut produce dacă $u_k = -1$. Pentru ambele grupuri se află tranziția ce dă valoarea maximă a lui: $A_{k-1}(s') + \Gamma_k(s',s) + B_k(s)$, iar LLR-ul *a-posteriori* este calculat doar pentru „cele mai bune tranziții”.

Algoritmul Max-Log-MAP se poate rezuma după cum urmează. Recurențele înainte și înapoi, ambele similare cu recurența înainte utilizată în algoritmul Viterbi, calculează $A_k(s)$ și $B_k(s)$, utilizând relațiile (3.41), respectiv (3.42). Metrica ramurii, $\Gamma_k(s',s)$ este dată de relația (3.43), unde termenul constant C poate fi omis. Din moment ce ambele recurențe înainte și înapoi sunt făcute, se poate calcula LLR-ul *a-posteriori*, utilizând relația (3.44). Astfel, complexitatea algoritmului Max-Log-MAP nu este cu mult mai mare decât cea a algoritmului Viterbi, în loc de o recurență există două, metrica ramurii din ecuația (3.43) are adunat termenul *a-priori* $u_k L(u_k)$, iar pentru fiecare bit trebuie utilizată ecuația (3.44) pentru a da LLR-urile *a-posteriori*.

3.5.4 Algoritmul Log-MAP

Algoritmul Max-Log-MAP are o ușoară degradare în performanță comparativ cu algoritmul MAP, datorită aproximării făcute în ecuația (3.37). Aproximarea ecuației (3.37) poate fi făcută prin utilizarea logaritmului Jacobian:

$$\begin{aligned}
 \ln(e^{x_1} + e^{x_2}) &\cong \max(x_1, x_2) + \ln\left(1 + e^{-|x_1 - x_2|}\right) \\
 &= \max(x_1, x_2) + f_c(|x_1 - x_2|),
 \end{aligned} \quad (3.45)$$

unde $f_c(x)$ poate fi gândit ca un termen de corecție. Acesta este baza algoritmului Log-MAP propus de Robertson ș.a. [RVH95]. Similar cu algoritmul Max-Log-MAP, valorile pentru $A_k(s) \stackrel{\Delta}{=} \ln(\alpha_k(s))$ și $B_k(s) \stackrel{\Delta}{=} \ln(\beta_k(s))$ sunt calculate utilizând recurența

înainte și înapoi. Însă, maximizările din ecuațiile (3.41) și (3.42) sunt completate cu termenul de corecție din ecuația (3.45). Aceasta înseamnă că sunt calculate mai exact decât în cazul algoritmului Max-Log-MAP. Deci se pot corecta aproximările făcute în ecuațiile (3.41) și (3.42) doar prin adunarea termenului $f_c(x)$. Similar, aproximarea din ecuația (3.44), ce dă LLR-ul *a-posteriori* $L(u_k|\underline{y})$, poate fi eliminată utilizând logaritmul Jacobian.

Termenul de corecție $f_c(x)$ nu trebuie calculat de fiecare dată, ci poate fi memorat într-un tabel. Diferitele variante ale algoritmului Log-MAP diferă prin modul de construcție al termenului de corecție, adică prin modul de aproximare al funcției:

$$f_c(x) = \ln(1+e^{-x}), \quad x \geq 0. \quad (3.46)$$

Funcțiile ce aproximează $f_c(x)$ trebuie să fie simplu de implementat și să reproducă cât mai fidel forma funcției $f_c(x)$. *Aproximarea rectangulară*, $g_c(x)$, a fost propusă de Robertson ș. a. în [RVH95] și este o extrapolare de ordin zero a funcției $f_c(x)$. Valorile funcției $g_c(x)$ utilizate în implementarea algoritmului Log-MAP rectangular sunt $\{0,6 \ 0,3 \ 0,14 \ 0,065 \ 0,03 \ 0,014 \ 0,005 \ 0,002 \ 0\}$. *Aproximarea liniară*, pe care am propus-o și analizat-o în [BAK04d)], $h_c(x)$, este de forma:

$$h_c(x) = \begin{cases} 0,7 - \frac{0,7}{x_0} \cdot x, & x \leq x_0 \\ 0, & x > x_0 \end{cases}. \quad (3.47)$$

Valoarea lui x_0 pentru care $h_c(x)$ aproximează cel mai bine pe $f_c(x)$ (în sensul minimizării ariei cuprinse între graficele funcțiilor $h_c(x)$ și $f_c(x)$) este $x_0=2,347$. Aproximările funcției $f_c(x)$ le-am reprezentat în Fig. 3.16 și Fig. 3.17.

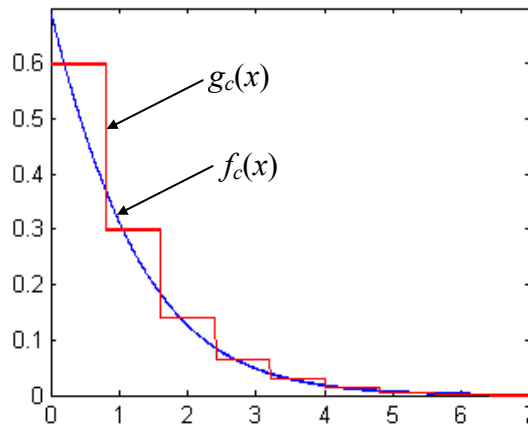
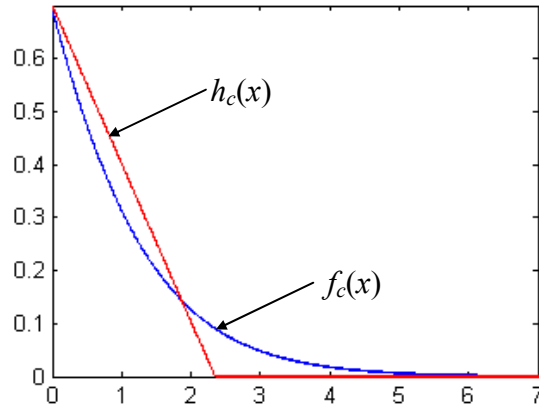


Fig. 3.16 Aproximarea rectangulară a funcției $f_c(x)$.

Fig. 3.17 Aproximarea liniară a funcției $f_c(x)$.

3.5.5 Performanțe ale algoritmilor de decodare

În scopul analizei performanțelor algoritmilor de decodare, în cadrul simulării am utilizat un TC de rată 1/3, constituit din două coduri RSC [1, 5/7], acest cod fiind cel mai performant dintre toate codurile de memorie 2, pe care le-am analizat în [BAK04e], având lungimea de constrângere $K=3$. Interleaver-ele pe care le-am folosit au fost următoarele:

- un interleaver pseudo-aleator și,
- un interleaver S (definite în paragraful 3.4), cu $S=29$, ambele interleaver-e având lungimea $N=1784$, lungime care dă de fapt și dimensiunea blocurilor de date.

Simularea am realizat-o considerând un canal AWGN și o modulație BPSK. Numărul maxim de iterații utilizat la decodare a fost egal cu 8. Pentru fiecare decodor am considerat un criteriu de oprire a iterațiilor [BAK04e]: când, pentru un bloc de date, numărul de erori rezultate este egal cu zero, restul de iterații nu mai sunt efectuate, trecându-se la blocul următor.

În Fig. 3.18 sunt prezentate rezultatele pe care le-am obținut cu diferiți algoritmi de decodare: MAP, Max-Log-MAP și Log-MAP, utilizând cele două tipuri de interleaver-e: pseudo-aleator și S .

Dacă situăm pe o scară ierarhică performanțele algoritmilor de decodare, din grafic, surprinzător, se observă că primul loc este ocupat de algoritmul de decodare Log-MAP, în special pentru valori ale SNR sub 1dB. Pentru valori ale SNR-urilor mai mari decât 1dB performanțele algoritmilor MAP și Log-MAP sunt aproximativ la fel. Algoritmul Max-Log-MAP este cu 0,2 dB inferior celorlalte două enunțate mai sus. Deasemenea, se observă superioritatea interleaver-ului S față de interleaver-ul pseudo-aleator. De exemplu, în cazul algoritmului MAP, interleaver-ul S aduce o îmbunătățire de 0,1 dB, la un $BER=10^{-5}$, comparativ cu interleaver-ul pseudo-aleator.

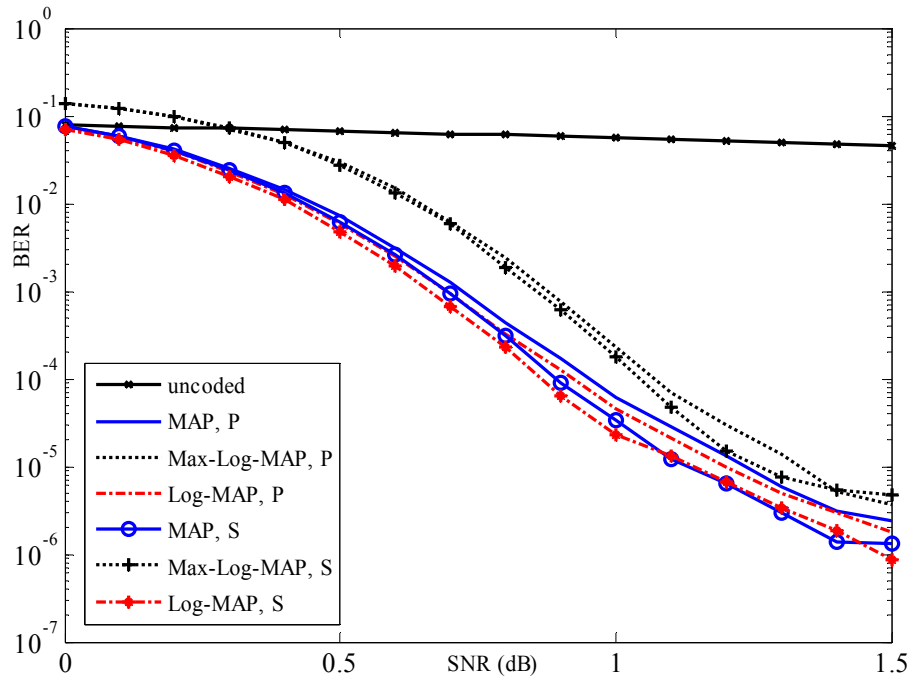


Fig. 3.18 Simularea TC-ului de rată 1/3, utilizând codul RSC [1, 5/7] și interleaver-ele S și pseudo-aleator, P.

3.6. Concluzii

În paragraful 3.4 am propus și analizat două noi interleaver-e, interleaver-ul bloc aleator în linie și interleaver-ul bloc cu linii aleatoare. Am comparat performanțele lor BER și FER cu cele ale altor interleaver-e clasice. Rezultatele pe care le-am obținut, în urma simulărilor, au condus la faptul că performanțele interleaver-ului bloc aleator în linie sunt foarte apropiate cu cele ale interleaver-ului S, cu S maxim, în special la lungimi ale interleaver-ului mai mari de 1000 de biți.

Un avantaj important al interleaver-ului bloc aleator în linie față de interleaver-ul S este acela că are o construcție mai simplă.

Dacă, din simulări, a rezultat faptul că performanțele interleaver-ului bloc aleator în linie sunt similare performanțelor interleaver-elor S, în cazul interleaver-ului bloc cu linii aleatoare, a rezultat faptul că are performanțe similare cu cele ale interleaver-elor bloc clasice, [KBN05].

În paragraful 3.5 am făcut o prezentare a algoritmilor de decodare, prezentare care s-a încheiat printr-o analiză a performanțelor lor BER. În analiza pe care am făcut-o am folosit interleaver-ul S și interleaver-ul pseudo-aleator.

În urma rezultatelor pe care le-am obținut am ajuns la concluzia că, pentru SNR-uri mai mari decât 1dB, performanțele algoritmilor MAP și Log-MAP sunt similare. S-a mai putut observa și faptul că algoritmul Max-Log-MAP este cu 0,2 dB inferior algoritmilor MAP și Log-MAP, [BAK04e].

4. PERFORMANȚELE TURBO CODURILOR ÎN CANALELE CU FADING PLAT

În cazul comunicațiilor wireless în canalele cu fading, codarea canalului este o metodă importantă pentru îmbunătățirea performanțelor. Așa cum s-a specificat și în capitolul anterior, turbo codurile au furnizat rezultate foarte apropiate de limita lui Shannon pe canalele AWGN [BGT93]. Fiind o tehnică puternică de codare, TC-urile oferă promisiuni mari în îmbunătățirea siguranței comunicațiilor pe canalele wireless unde fading-ul este o problemă. În acest capitol am simulat și analizat performanțele TC-urilor în canalele cu fading plat: Rayleigh, Rice și Nakagami. În continuare fac o prezentare generală a tipurilor de fading care pot apărea în comunicațiile mobile, după care, în paragrafele următoare, prezint câteva rezultate experimentale.

4.1. Propagarea radio în comunicațiile mobile

Pentru înțelegerea relațiilor referitoare la performanțele de bază ale sistemelor de comunicații un prim pas îl reprezintă studiul *canalului AWGN*, necorelat, ce afectează eșantioanele de date, în absența *interferenței intersimbol ISI* (Inter Symbol Interference). Zgomotul termic din receptor reprezintă cauza primară a degradării performanțelor. De multe ori însă, interferența este mai importantă decât zgomotul termic. Această interferență externă se cuantifică printr-un parametru numit temperatura antenei. Zgomotul termic are de obicei o densitate spectrală de putere plată în banda semnalului și o repartiție gaussiană de medie nulă.

Se cunoaște faptul că zgomotul AWGN este un semnal aleator idealizat ce modelează o mare parte din zgomotele întâlnite în canalele reale de comunicație. Faptul că este *aditiv* se materializează printr-o relație intrare-ieșire a canalului (numit și el AWGN după numele zgomotului) de forma:

$$r(t) = s(t) + w(t) \quad (4.1)$$

unde cu $s(t)$ am notat semnalul transmis, $r(t)$ cel recepționat, iar $w(t)$ este zgomotul aditiv.

Atributul de „alb” (white) se referă la densitatea spectrală de putere *psd* (Power Spectral Density). Și anume, *psd* este o constantă de valoare $N_0/2$, pentru toate frecvențele, așa cum reiese și din figura următoare.

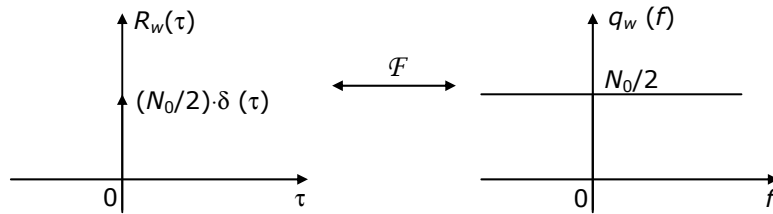


Fig. 4.1 Funcția de autocorelație și densitatea spectrală de putere a zgomotului alb.

Bineînțeles această ipoteză este neconformă cu realitatea deoarece conduce la o putere infinită a zgomotului:

$$P_W = \int_{-\infty}^{\infty} q_W(f) \cdot df = \infty. \quad (4.2)$$

O posibilă realizare a zgomotului alb alături de densitatea sa de probabilitate sunt reprezentate în Fig. 4.2.

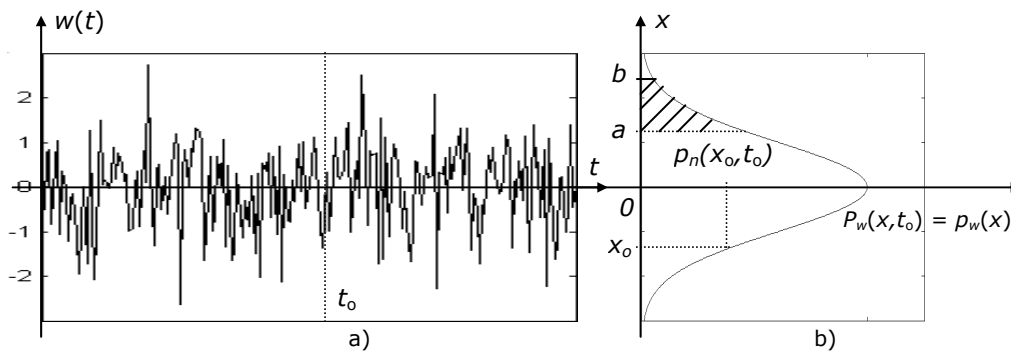


Fig. 4.2 a) O realizare particulară a zgomotului alb; b) Densitatea de probabilitate a zgomotului alb.

S-a considerat momentul de timp $t=t_0$, unde $p_w(x, t_0)$ dă o măsură a șanselor ca semnalul aleator (zgomotul AWGN $w(t)$) să ia diferite valori x . Probabilitatea ca $w(t_0)=x_0$ este $p_w(x_0) \cdot dx$, și este egală cu zero. Aceasta se datorează împrăștierei probabilității pe un domeniu continuu de valori. Altfel spus $p_w(x, t_0)$ din Fig. 4.2, având expresia, [PRO00]:

$$p_w(x) = \frac{1}{\sqrt{2\pi}\sigma_w} \exp\left(-\frac{x^2}{2\sigma_w^2}\right) \quad (4.3)$$

este o densitate de probabilitate (*d.p.*) distribuită, unde σ_w reprezintă *abaterea standard*, iar σ_w^2 *dispersia*. Nu are sens să se vorbească despre probabilitatea unui punct x_0 , în cazul unei *d.p.* distribuite, deoarece ea este nulă. Probabilitate nenulă

are doar un interval de valori. Astfel probabilitatea ca zgomotul AWGN $w(t)$ să ia valori, la momentul t_0 , cuprinse între a și b este:

$$P([a, b]) = \int_a^b p_W(x, t_0) dx . \quad (4.4)$$

Rezultă faptul că:

$$P((-\infty, +\infty)) = 1 = \int_{-\infty}^{+\infty} p_W(x, t_0) dx \quad (4.5)$$

și că $p_W(x, t_0)$ este o funcție nenegativă. Mai mult $p_W(x, t_0) \rightarrow 0$ pentru $x \rightarrow -\infty$ și $x \rightarrow +\infty$.

Funcția de repartiție a probabilității zgomotului AWGN este:

$$F_W(x, t) = \int_{-\infty}^x p_W(y, t) dy . \quad (4.6)$$

Integrala dată de (4.6) nu este calculabilă analitic ci doar prin metode numerice. Astfel:

$$\begin{aligned} F_W(x, t) &= \int_{-\infty}^x \frac{1}{\sqrt{2\pi}\sigma_W} \cdot \exp\left(-\frac{y^2}{2\sigma_W^2}\right) \cdot dy = \\ &= \frac{1}{\sqrt{\pi}} \int_{-\infty}^{\frac{x}{\sqrt{2}\sigma_W}} \exp(-u^2) \cdot du = \\ &= 1 - \frac{1}{2} \operatorname{erfc}\left(\frac{x}{\sqrt{2}\sigma_W}\right), \end{aligned} \quad (4.7)$$

unde $\operatorname{erfc}()$ este *funcția eroare complementară*.

În Fig. 4.3 sunt prezentate *d.p.* normală (gaussiană) și funcția de repartiție corespunzătoare. Zgomotul AWGN este adesea asimilat cu zgomotul termic. Dacă atributul de „alb” este o idealizare (ce conduce la o ipoteză nereală – putere infinită) cel de „gaussian” este în conformitate cu realitatea. Astfel, mișcarea termică a electronilor într-o rezistență produce la capetele sale o tensiune, numită de zgomot termic, ce ia valori normal distribuite. Această distribuție, însă, depinde de temperatură.

Dacă rezistența este încălzită la $T_1 > T_0$ atunci agitația crește și șansele ca tensiunea de zgomot termic să ia valori mai mari cresc. Bineînțeles forma curbei rămâne gaussiană (cele mai probabile valori rămân cele mici însă nu atât de probabile ca și la temperatura T_0). În figura de mai sus cele două situații sunt sugerate a fi la momentele de timp t_0 și t_1 .

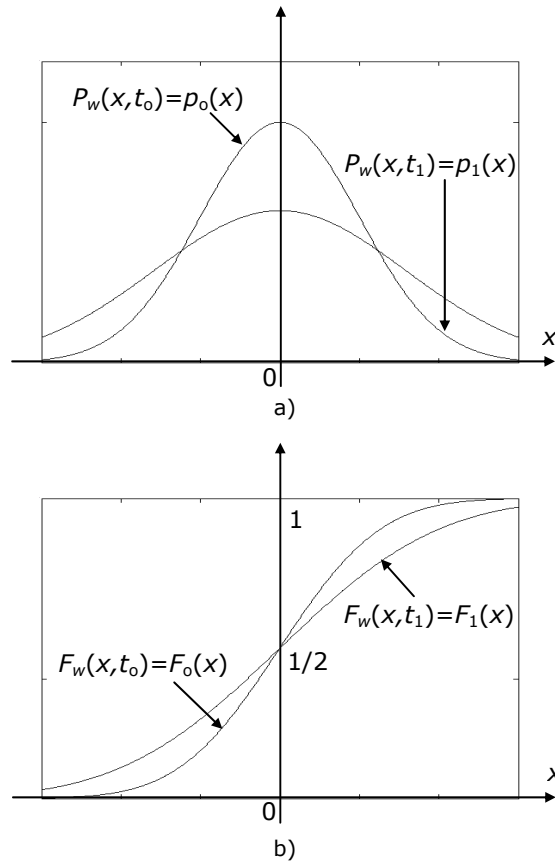


Fig. 4.3 a) Densitatea de probabilitate; b) Funcția de repartiție a probabilității unei variabile aleatoare gaussiene pentru două valori ale parametrului σ_w ($\sigma_1 > \sigma_0$).

Dacă la capetele rezistenței se aplică o tensiune exterioară riguros constantă de valoare u_0 atunci *d.p.* și funcția de repartiție se modifică ca și în Fig. 4.4.

Analitic, relațiile (4.3) și (4.7) devin:

$$p_w(x, t) = \frac{1}{\sqrt{2\pi}\sigma_w} \exp\left(-\frac{(x - u_0)^2}{2\sigma_w^2}\right), \quad (4.8)$$

$$F_w(x, t) = 1 - \frac{1}{2} \operatorname{erfc}\left(\frac{x - u_0}{\sqrt{2}\sigma_w}\right), \quad (4.9)$$

unde u_0 - media și σ_w^2 - dispersia sunt responsabile de o eventuală dependență de timp. Dacă semnalul aleator este staționar atunci $p_w(x, t) = p_w(x)$ și $F_w(x, t) = F_w(x)$.

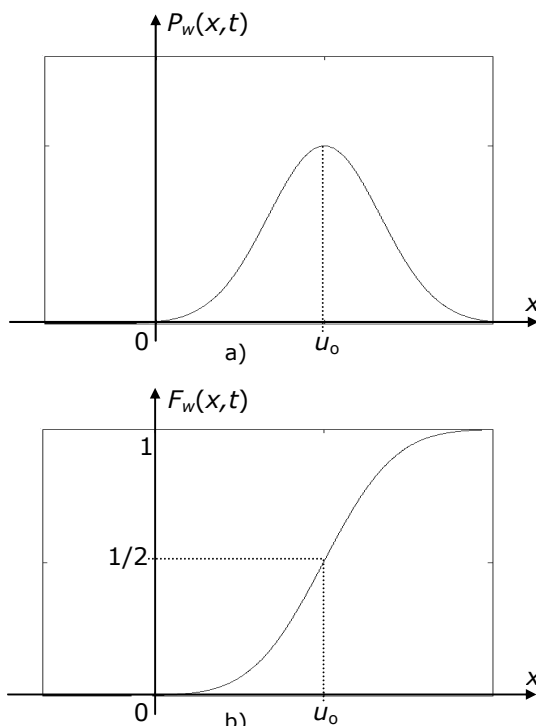


Fig. 4.4 a) Densitatea de probabilitate; b) Funcția de repartiție pentru o variabilă aleatoare normal distribuită cu medie $u_0 \neq 0$.

Canalul cu fading este un alt tip de canal. În cazul canalului cu fading, parametrii semnalului util fluctuează în timp, acest fenomen fiind ca și o modulație parazită (zgomot multiplicativ).

Dacă nu sunt specificate caracteristicile de propagare ale canalului radio, se presupune de obicei că atenuarea semnalului în funcție de distanță are loc ca în cazul propagării printr-un spațiu liber ideal, [SKL88]. La modelul spațiului liber se consideră că regiunea dintre antena transmițătoare și cea receptoare este liberă, fără obiecte care ar putea să reflecte sau să absoarbă energia frecvențelor radio, RF (Radio Frequency). De asemenea se presupune că în interiorul regiunii, atmosfera se comportă ca un mediu uniform neabsorbant iar pământul se consideră infinit de departe față de semnalul care se propagă (deci are un coeficient de reflexie neglijabil). În acest spațiu liber idealizat, atenuarea energiei RF dintre transmițător și receptor are o lege pătratică. Puterea recepționată în funcție de puterea transmisă, este atenuată cu un factor $L_S(d)$, numit *pierderele căii* sau *pierderele în spațiu liber*. Pentru o antenă receptoare izotropă acest factor este, [SKL88]:

$$L_S(d) = \left(\frac{4\pi d}{\lambda} \right)^2, \quad (4.10)$$

unde: d – distanța dintre transmițător și receptor, iar λ – lungimea de undă a semnalului care se propagă.

În cazul propagării idealizate, poate fi prezisă puterea semnalului recepționat.

În realitate, propagarea semnalelor are loc în atmosferă și aproape de pământ, astfel că modelul propagării libere nu este adecvat pentru descrierea canalului și estimarea performanței sistemului, [SKL88]. În sistemele de comunicații mobile fără fir, un semnal ajunge de la transmitător la receptor, prin căi de reflexie multiple, acest fenomen se numește *propagare multicăi*. Acest efect poate cauza fluctuații ale semnalului recepționat ca amplitudine, fază și unghi de sosire, dând naștere terminologiei de *fluctuație multicăi (multipath fading)*. Fluctuația se referă la fluctuații aleatoare ale semnalului sau fading datorat propagării multicăi. Modelarea și proiectarea sistemelor care să evite sau care să atenueze efectul fading-ului sunt mult mai dificile decât cele ale unei surse ale cărei performanțe sunt degradate de AWGN.

O posibilă clasificare a manifestărilor fading-ului în canale este schițată în Fig. 4.5, [SKL88].

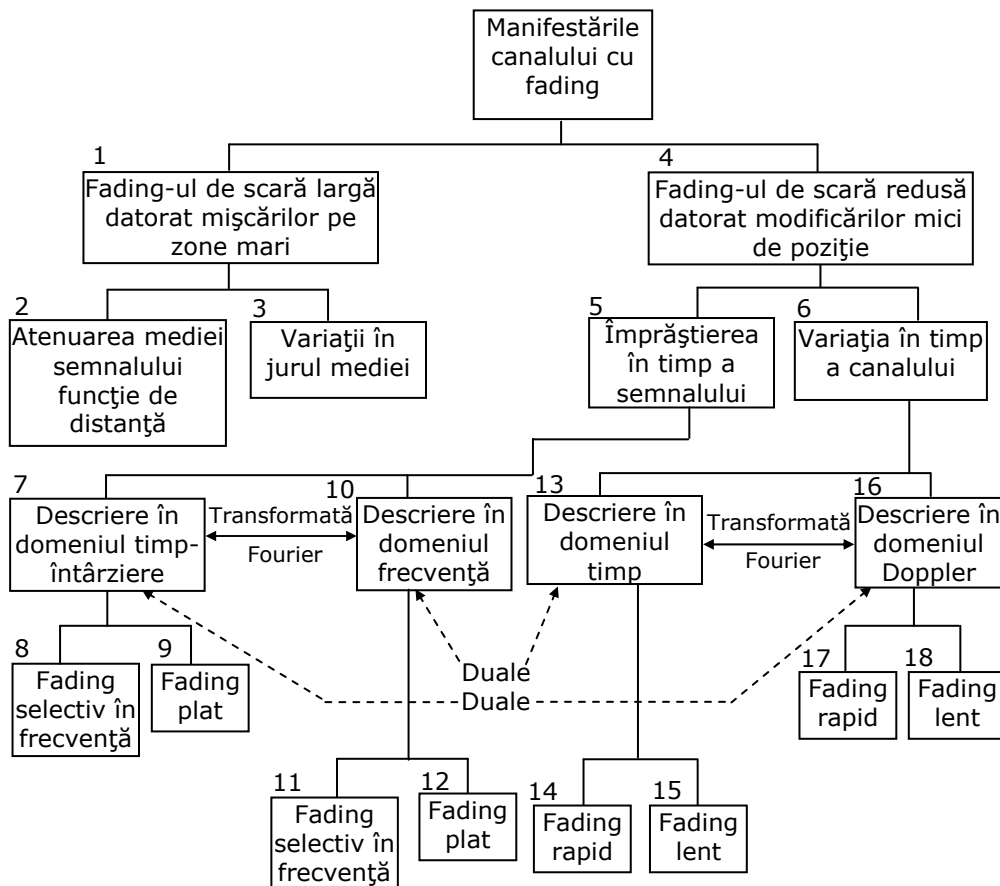


Fig. 4.5 Manifestările canalului cu fading.

Sunt două tipuri principale de fading care caracterizează comunicațiile mobile, și anume: fading-ul de scară largă și fading-ul de scară redusă.

Fading-ul de scară largă reprezintă atenuarea puterii medii a semnalului sau pierderea căii cauzată de mișcarea pe zone mari (blocurile 1, 2 și 3 din Fig. 4.5). Acest fenomen este asociat cu topografia de teren: dealuri, păduri, grupuri de clădiri, etc., aflate între transmițător și receptor. De multe ori receptorul este reprezentat ca fiind *umbrit* de aceste proeminențe. Statistica fading-ului pe scară largă furnizează o cale de calcul a estimatului pierderii căii în funcție de distanță. Acesta apare ca o pierdere medie a căii (o lege de puterea n) și o variație, în jurul mediei, de tip log-normal.

Fading-ul de scară redusă se referă la schimbări dramatice ale amplitudinii și fazei semnalului cauzate de modificări mici, de ordinul unei jumătăți de lungimi de undă, în separarea spațială dintre emițător și receptor. După cum se vede din Fig. 4.5 (blocurile 4, 5 și 6) fading-ul de scară redusă se manifestă prin două mecanisme și anume împrăștierea în timp a semnalului (sau dispersia semnalului) și comportarea variabilă în timp a canalului. Analiza acestora se poate face în domeniul timp și în domeniul frecvență (Fig. 4.5, blocurile: 7, 10, 13, 16). Pentru dispersia semnalului tipurile de degradare cauzate de fading sunt clasificate ca fading selectiv în frecvență, respectiv ca fading neselectiv în frecvență (plat) (blocurile 8, 9, 11, 12). Pentru manifestările variabile în timp, degradările produse de fading se clasifică în: fading lent, respectiv fading rapid (blocurile 14, 15, 17, 18). Pentru aplicații radio mobile, canalul este variabil în timp din cauză că mișcarea dintre transmițător și receptor duce la modificări ale căii de propagare. Fading-ul de scară redusă este numit și fading *Rayleigh* deoarece atunci când căile rezultate prin reflexie sunt numeroase și nu există o componentă de cale directă LOS (Line-Of-Sight) anvelopa semnalului recepționat este descrisă statistic printr-o *pdf* de tip Rayleigh. Atunci când este prezentă componenta dominantă de semnal non-fading, ca și calea de propagare directă LOS, anvelopa fading-ului de scară redusă este descrisă de o *pdf* Rice-ană. Mobilul radio, care se deplasează pe zone mari, trebuie să proceseze semnale afectate de ambele tipuri de fading: fading-ul de scară redusă suprapus peste fading-ul de scară largă.

În Fig. 4.6 sunt prezentate diferitele contribuții care trebuiesc luate în considerare când se estimează pierderea pe cale pentru analiza *alocării* legăturii în aplicațiile celulare.

Aceste contribuții sunt:

- pierderea medie a căii ca funcție de distanță, cauzată de fading-ul de scară largă,
- variații apropiate de cazul cel mai defavorabil a pierderii medii a căii (tipic 6-10 dB), sau marginea fading-ului de scară largă,
- cazul aproape cel mai defavorabil Rayleigh, sau marginea fading-ului de scară redusă (tipic 20-30 dB).

În Fig. 4.6, notația " $\sim 1-2\%$ " indică o zonă (probabilitate) sub zona de sfârșit a fiecărei *pdf*, ca obiectiv de proiectare. Astfel, totalul până la marginea indicată se dorește să furnizeze o putere a semnalului recepționat de 98-99% din fiecare tip de variație a fading-ului (de scară largă sau redusă).

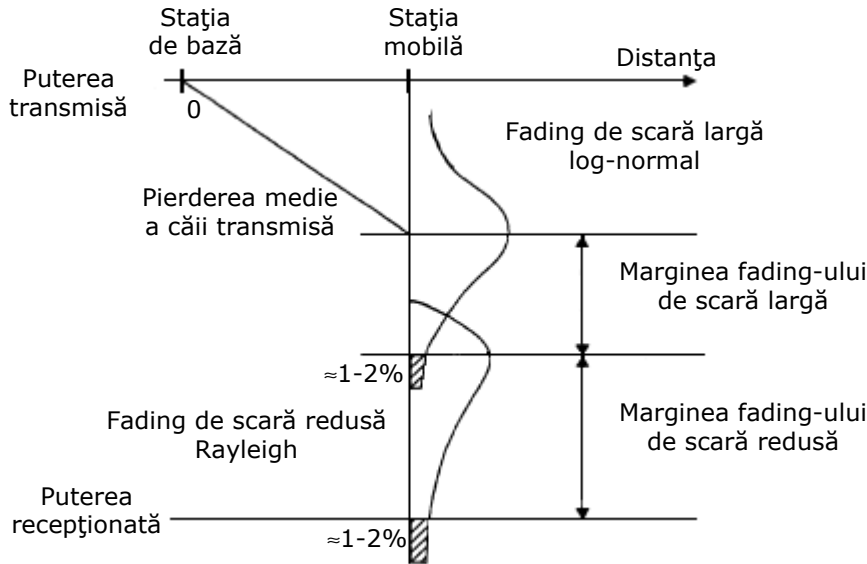


Fig. 4.6 Considerații asupra alocării legăturii pentru canale cu fading.

Semnalul recepționat, $r(t)$ este descris în funcție de semnalul transmis, $s(t)$, în convoluție cu $h_c(t)$, răspunsul la impuls al canalului. Neglijând efectul zgomotului, putem scrie:

$$r(t) = s(t) * h_c(t), \quad (4.11)$$

unde „*” reprezintă convoluția. În cazul comunicațiilor radio mobile $r(t)$ poate fi exprimat prin două variabile aleatoare:

$$r(t) = m(t) \cdot r_0(t), \quad (4.12)$$

unde $m(t)$ este componenta de fading de scară largă iar $r_0(t)$ este componenta de fading de scară redusă. Uneori $m(t)$ este denumit *medie locală* sau fading *log-normal* deoarece mărimea lui $m(t)$ este descrisă printr-o *pdf* log-normală (mărimea măsurată în decibeli are o *pdf* gaussiană). La fel $r_0(t)$ este uneori numit fading *multicăi* sau fading *Rayleigh*. În Fig. 4.7 sunt prezentate relațiile dintre fading-ul de scară largă și redusă. În Fig. 4.7 a) puterea semnalului recepționat $r(t)$ este reprezentată față de deplasarea antenei (tipic, în unități de lungime de undă), pentru cazul comunicațiilor mobile. Fading-ul de scară redusă este suprapus peste fading-ul de scară largă și poate fi identificat ușor. Deplasarea tipică a antenei, dintre zerourile semnalului de scară redusă, este aproximativ o jumătate de lungime de undă. În Fig. 4.7 b) fading-ul de scară largă, sau media locală $m(t)$, a fost extras pentru a se vedea fading-ul de scară redusă $r_0(t)$ în jurul unei anumite puteri constante.

În paragraful următor este descris modelul canalului utilizat în simulările pe care le-am realizat pentru diferite tipuri de fading.

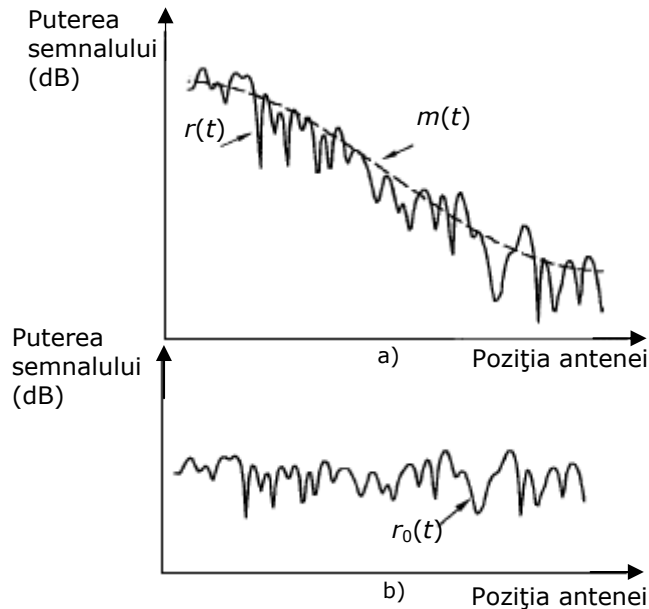


Fig. 4.7 a) Fading-ul de scară largă; b) fading-ul de scară redusă.

4.2. Modelul sistemului de transmisie

În Fig. 4.8 a) este prezentat un model pentru simularea funcționării unui sistem de transmisie cu modulație BPSK printr-un canal cu fading plat și zgomot AWGN fără codare iar, în Fig. 4.8 b), modelul pentru simularea funcționării sistemului de transmisie cu modulație BPSK printr-un canal cu fading plat și zgomot AWGN turbo codat, [BAK05b].

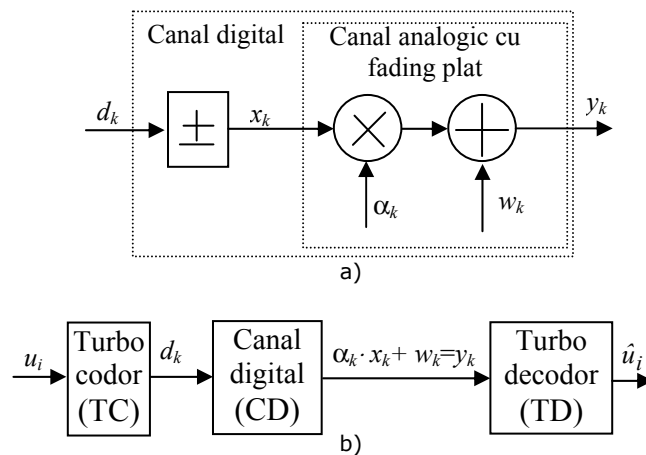


Fig. 4.8 Modele de simulare: a) sistem de transmisie necodat și b) sistem de transmisie turbo codat, cu modulație BPSK și canal cu fading plat și zgomot AWGN.

unde:

- cu u_i s-a notat secvența de date de la intrarea turbo codorului,
- cu d_k s-a notat secvența de date de la ieșirea turbo codorului, o secvență unipolară,
- cu x_k și cu y_k am notat secvența de date bipolară transmisă în canal, respectiv recepționată din canal, într-un moment de timp k ,
- parametrul α_k este o valoare aleatoare ce caracterizează fluctuația în timp a canalelor de la simbol la simbol (fast fading), sau de la bloc la bloc (block fading) [CWW98], a cărei distribuție determină tipul canalului: Rayleigh, Rice sau Nakagami,
- cu \hat{u}_i s-a notat secvența de date de la ieșirea turbo decodorului,
- w_k este un eșantion din zgomotul presupus AWGN, de valoare medie pătratică $\overline{w_k^2} = \frac{1}{2 \cdot 10^{SNR/10}}$, relație pe care am prezentat-o în [BAK05b, BKDB05], și care rezultă imediat din definiția raportului semnal/zgomot efectiv dată în [JSC98].

La început se va face o conversie a secvenței binare unipolară într-una bipolară. Această operație este realizată în blocul notat „±” în Fig. 4.8 a), iar relația corespunzătoare este:

$$x_k = 2 \cdot d_k - 1, \quad k = 1 \div N/R. \quad (4.13)$$

Aici d_k și x_k se referă la secvența turbo-codată, având lungimea de $1/R$ ori mai mare (R –rata de turbo-codare) decât lungimea de întretesere, N . Semnalul de la ieșirea din canal va avea forma din ecuația (4.14), unde cu E_{bc} s-a notat energia per bit transmis din secvența codată:

$$y_k = \alpha_k \cdot x_k \cdot \sqrt{E_{bc}} + w_k, \quad (4.14)$$

unde $\alpha_k = 1$ în cazul canalului AWGN, respectiv o variabilă aleatoare cu $\overline{\alpha_k^2} = 1$, în cazul canalului cu fading plat (Rayleigh, Rice sau Nakagami). Cu convenția că, [VUY01]:

$$E_{bc} = 1 = R \cdot E_{bi} \quad \text{și} \quad T_b = 1, \quad (4.15)$$

unde E_{bi} reprezintă energia per bit din secvența de informație iar R rata de codare, rezultă că „puterea” conținută în secvența de numere $\alpha_k \cdot x_k \cdot \sqrt{E_{bc}}$ este unitară:

$$S = \frac{1}{M} \cdot \sum_{k=1}^M \alpha_k^2 \cdot x_k^2 \cdot E_{bc} = \frac{1}{M} \cdot \sum_{k=1}^M \alpha_k^2 = \overline{\alpha_k^2} = 1, \quad (4.16)$$

unde $M = nb \cdot N/R$ reprezintă numărul de biți transmiși, adică egal cu produsul dintre numărul de blocuri transmise, nb , și numărul de biți dintr-un bloc, N/R .

Atunci când turbo codorul este utilizat într-un canal cu fading, algoritmul MAP trebuie modificat pentru a ține cont de acest fapt. Modificarea constă în

utilizarea unui coeficient L_c calculat după relația dată în [HLY02] și folosită în toate simulările mele din [KOB09, KBN09, BAK05b]:

$$L_c = 2 \cdot \bar{\alpha} \cdot E_b / \sigma^2 = 4 \cdot \bar{\alpha} \cdot R \cdot B, \quad (4.17)$$

unde R este rata de codare iar B valoarea absolută a SNR-ului.

Diferența față de canalul AWGN o dă termenul $\bar{\alpha}$ care reprezintă valoarea medie a secvenței de numere ce simulează amplitudinea fading-ului canalului, relația (4.14). În cazul canalului AWGN acest termen este unitar. În cazul canalului cu fading valoarea sa va fi calculată funcție de tipul fading-ului (Rayleigh, Rice sau Nakagami).

4.3. Fading plat de tip Rayleigh

Fluctuațiile aleatoare ale semnalului recepționat datorate fading-ului pot fi modelate considerând un proces aleator în t , $c(\tau, t)$, cu întârzierea τ . Deoarece semnalul recepționat se compune dintr-o mulțime de componente provenite prin diferite căi din reflexiile și împrăștierea semnalului pe diferite suprafețe neregulate, în virtutea teoremei limită centrală, $c(\tau, t)$ poate fi modelat ca și un proces aleator complex, gaussian. Astfel, la orice moment de timp t , părțile reală și imaginară ale lui $c(\tau, t)$ sunt variabile aleatoare distribuite normal.

Dacă $c(\tau, t)$ este de medie nulă, atunci anvelopa sau modulul $R(\tau, t) = |c(\tau, t)|$ are o densitate de probabilitate Rayleigh [PRO00]:

$$p_R(r) = \begin{cases} \frac{r}{\sigma^2} \cdot \exp\left(-\frac{r^2}{2 \cdot \sigma^2}\right), & r \geq 0 \\ 0, & r < 0 \end{cases} \quad (4.18)$$

În Fig. 4.9 am reprezentat densitatea de probabilitate Rayleigh, pentru $\sigma=1$, ținând cont de relația (4.18).

Dacă $c(\tau, t)$ este de medie $A > 0$, atunci anvelopa sau modulul $R(\tau, t) = |c(\tau, t)|$ are o densitate de probabilitate Rice:

$$p_R(r) = \frac{r}{\sigma^2} \cdot \exp\left(-\frac{r^2 + A^2}{2 \cdot \sigma^2}\right) \cdot J_0\left(\frac{A \cdot r}{\sigma^2}\right), \quad r \geq 0, \quad (4.19)$$

unde $J_0(\cdot)$ este funcția Bessel modificată de ordin zero.

În cazul canalului cu fading plat de tip Rayleigh, secvența recepționată (la ieșirea din canalul neregulat, canal ce include modulatorul și demodulatorul), Fig. 4.8 a), este:

$$y_k = \alpha_k * x_k + w_k, \quad (4.20)$$

unde x_k este secvența emisă, w_k este un eșantion din zgomotul presupus AWGN iar α_k este o variabilă aleatoare de tip Rayleigh, responsabilă de caracterul fluctuant în timp.

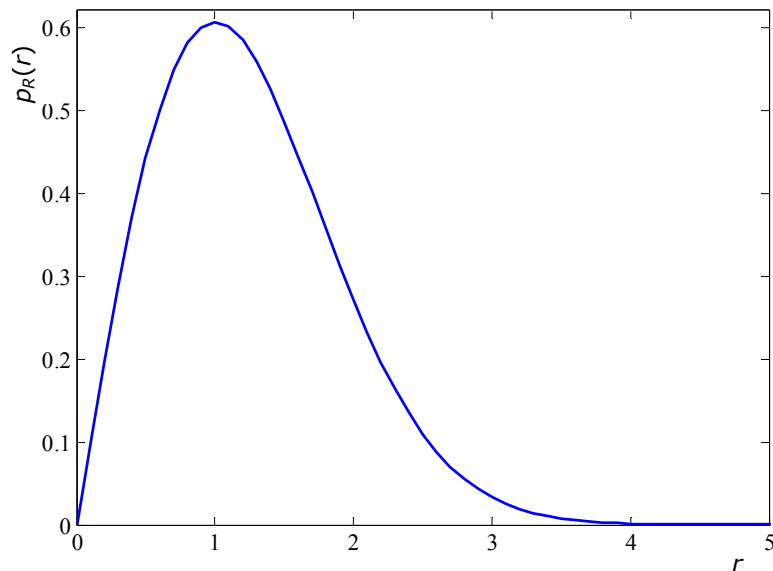


Fig. 4.9 Densitatea de probabilitate Rayleigh.

În diagramele din Fig. 4.10 și Fig. 4.11 am prezentat rezultatele simulărilor pe care le-am obținut în [KOV05, KOB09], pentru TC-urile, de rată 1/3, având ca și coduri componente codurile convoluționale RSC, concatenate în paralel, de memorie 2, respectiv 3, în variantele [1, 5/7] și [1, 15/13], considerând canalul cu fading plat de tip Rayleigh.

De asemenea, în toate cazurile, am considerat varianta în care cele două secvențe de paritate obținute de la cele două codoare RSC ale turbo codorului nu sunt puncturate, trellis-ul primului codor este închis iar al celui de-al doilea este deschis. Simulările au fost realizate considerând două tipuri de interleaver-e: interleaver-ul S, a cărui lungime de întretesere este egală cu 1784 biți, lungime aleasă conform cu recomandarea CCSDS [CCSDS02] și distanța minimă de împrăștiere este egală cu 29 (I1784S29), și noul interleaver propus în capitolul anterior, și anume interleaver-ul bloc aleator în linie, BRL, a cărui lungime este egală cu 1785 (IXBRLY, unde numărul de linii este $X=119$ și numărul de coloane este $Y=15$: I119BRL15). Modulația folosită este modulația BPSK, iar algoritmul de decodare utilizat de către turbo decodor este algoritmul de decodare MAP, la care coeficienții $\beta(\cdot)$ de start ai celui de al doilea decodor au fost setați echiprobabili, [BAK04d]. S-a folosit un număr maxim de 15 iterații și un criteriu de oprire al iterațiilor, iar valoarea pragului LLR este egală cu 10.

Observație: Toate graficele sunt reprezentate în funcție de "media SNR". Dacă pentru canalul AWGN semnalul recepționat este $y=x+w$ (x este semnalul emis și w este zgomotul) și $SNR=P_x/P_w$, pentru canalul cu fading $y=\alpha \cdot x+w$ unde α este o variabilă aleatoare, rezultă că și $SNR=\alpha^2 \cdot P_x/P_w$ este o variabilă aleatoare, și deci putem scrie media $SNR = \overline{\alpha^2} \cdot P_x / P_w$.

În Fig. 4.10 prezint performanțele BER pe care le-am obținut din simularea turbo codurilor în canale cu fading plat de tip Rayleigh, pentru codurile și interleaver-ele menționate mai sus. Așa cum rezultă din figură, în lipsa codării, până la $\text{SNR} \approx 40$ dB (corespunzător unui $\text{BER} \approx 10^{-5}$), BER scade liniar cu SNR-ul. Pentru valori mai mari decât 43 dB ale SNR-ului, rata erorii tinde să se mențină la aceeași valoare.

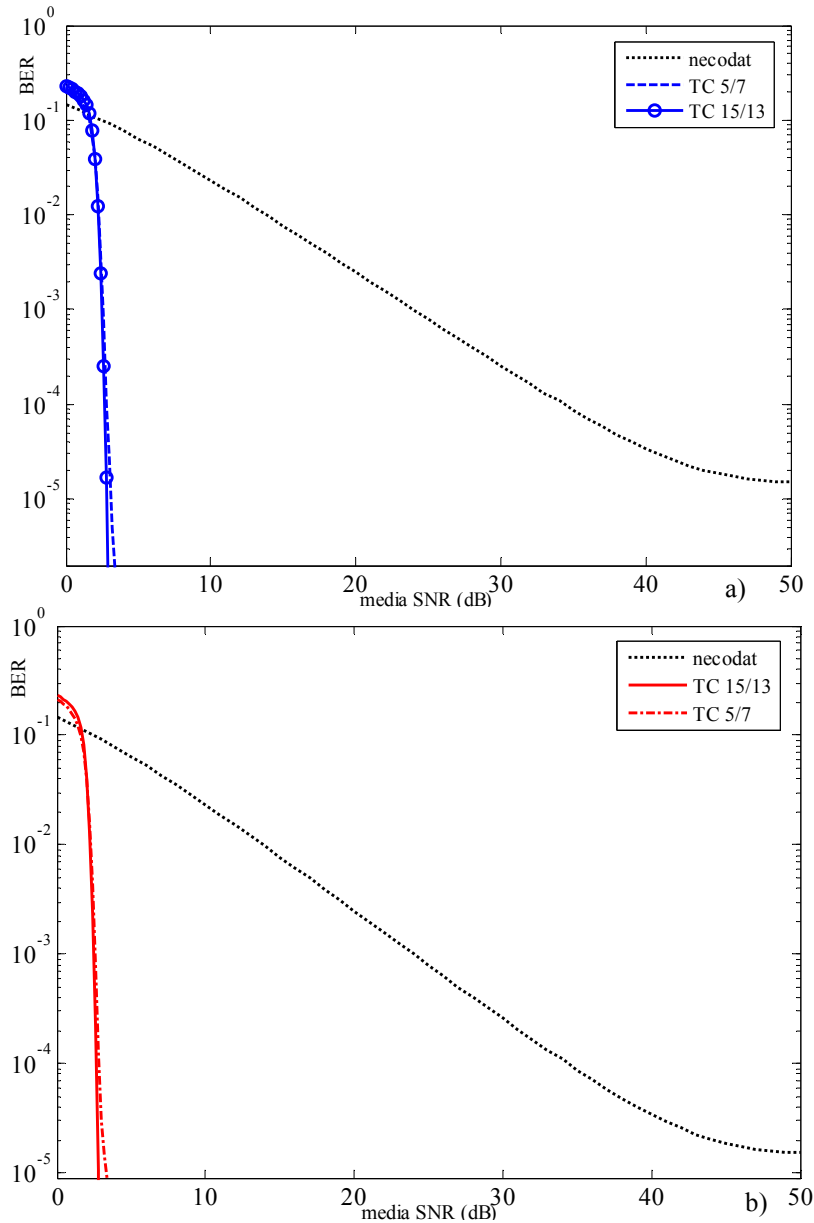


Fig. 4.10 Performanțele BER ale TC-urilor în canale cu fading plat de tip Rayleigh pentru codurile 5/7 și 15/13, utilizând: a) interleaver-ul S – I1784S29 și b) și interleaver-ul bloc aleator în linie I119BRL15.

Altfel spus, fără o codare, în canalul cu fading plat de tip Rayleigh nu se pot obține rate de eroare sub pragul de 10^{-5} , chiar dacă SNR-ul crește. Aceste valori, însă, pot fi obținute prin utilizarea turbo codurilor la un raport semnal pe zgomot sub 5 dB, atât în cazul folosirii interleaver-ului S , cât și în cazul noului interleaver (BRL) pe care l-am propus și analizat în [KBN05, KOB09].

În Fig. 4.11 a) am prezentat în detaliu, pentru valori mici ale SNR, performanțele BER ale TC-urilor, iar în Fig. 4.11. b) am prezentat performanțele lor FER.

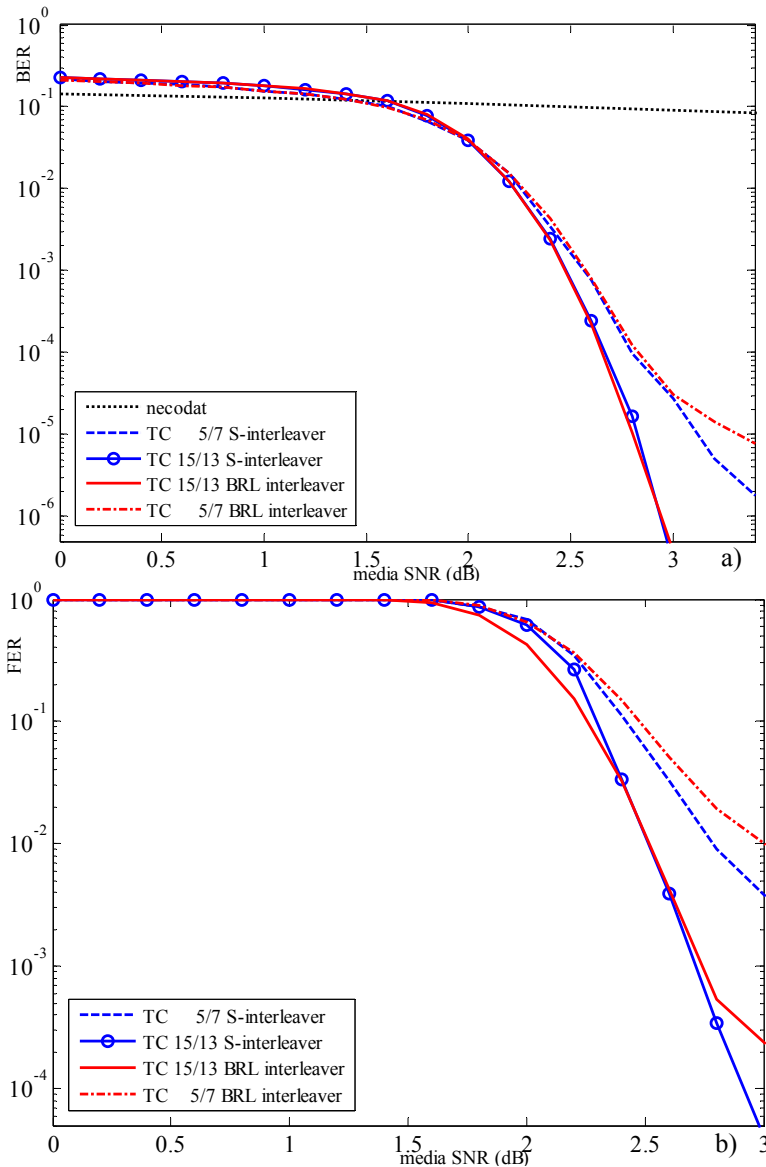


Fig. 4.11 Performanțele: a) BER detaliate și b) FER, ale TC-urilor în canale cu fading plat de tip Rayleigh pentru codurile 5/7 și 15/13, utilizând interleaver-ul S - I1784S29 și interleaver-ul bloc aleator în linie I119BRL15.

O primă concluzie ce poate fi trasă pe baza rezultatelor obținute în Fig. 4.11 este aceea că interleaver-ul bloc aleator în linie se comportă la fel ca și interleaver-ul S , în cazul utilizării codului 15/13 pentru valori ale BER de până la $5 \cdot 10^{-7}$, cât și în cazul utilizării codului 5/7 pentru valori ale BER de până la $3 \cdot 10^{-5}$.

O altă remarcă ce poate rezulta este aceea că există o discrepanță sesizabilă la 3 dB între codul 15/13, având lungimea de constrângere $K=4$ și codul 5/7 cu lungimea de constrângere $K=3$. Se mai poate observa că pentru valori ale FER de până la $5 \cdot 10^{-4}$, interleaver-ul pe care l-am propus se comportă similar cu interleaver-ul S , în cazul utilizării codului 15/13. Dar, după această valoare a FER-ului performanțele interleaver-ului S sunt mai bune, ajungând astfel la un FER= $2 \cdot 10^{-4}$ să existe o diferență de aproximativ 0.15 dB.

În tabelele următoare, prezint cum sunt influențate performanțele BER și FER, ale TC-urilor ce au ca și coduri componente codurile RSC 15/13, de către estimarea SNR-ului, estimare dată de factorul L_c , [HLY02]. Astfel, folosesc valori ale lui L_c date de următoarea relație:

$$L_c = 4 \cdot R \cdot B \cdot f, \quad (4.21)$$

unde R este rata turbo codului, B este valoarea absolută a SNR-ului și f este raportul dintre valoarea factorului L_c utilizat în simulări și cel teoretic, [KOB09]:

$$f = L_c / (4 \cdot R \cdot B). \quad (4.22)$$

Tabelul 4.1. BER $\cdot 10^9$, în funcție de f și de tipul de interleaver folosit.

Ilv	SNR (dB)	$f=0.4$	$f=0.5$	$f=0.6$	$f=0.7$	$f=0.8$	$f=0.9$
		BER	BER	BER	BER	BER	BER
<i>S-ilv</i>	2.8	38403775	18411	2130	3175	6047	11504
<i>BRL</i>	2.8	36961840	18434	3814	4412	6142	16023
Ilv	SNR (dB)	$f=1$	$f=1.1$	$f=1.2$			
		BER	BER	BER			
<i>S-ilv</i>	2.8	33293	67827	135499			
<i>BRL</i>	2.8	35052	70013	114227			

Tabelul 4.2. FER $\cdot 10^6$, în funcție de f și de tipul de interleaver folosit.

Ilv	SNR (dB)	$f=0.4$	$f=0.5$	$f=0.6$	$f=0.7$	$f=0.8$	$f=0.9$	$f=1$
		FER	FER	FER	FER	FER	FER	FER
<i>S-ilv</i>	2.8	485125	445	115	145	190	315	645
<i>BRL</i>	2.8	473333	1016	736	790	836	1066	1403
Ilv	SNR (dB)	$f=1.1$	$f=1.2$					
		FER	FER					
<i>S-ilv</i>	2.8	1265	2070					
<i>BRL</i>	2.8	1996	2723					

Din rezultatele pe care le-am obținut în tabelele 4.1 și 4.2, la un SNR=2.8 dB, considerând ambele interleaver-e, se observă că, cele mai bune performanțe BER și FER s-au obținut pentru $f=0.6 \div 0.8$. Cele mai slabe performanțe se obțin pentru $f < 0.5$.

Această concluzie este verificată în Fig. 4.12 și Fig. 4.13. Este evident că performanțele BER și FER cele mai bune le-am obținut pentru valorile menționate mai sus, $f=0.6 \div 0.8$. De exemplu, pentru un $\text{BER}=10^{-5}$, distanța dintre cele două curbe generate la un $f=0.6$, respectiv $f=0.5$ este aproximativ egală cu 0.1 dB.

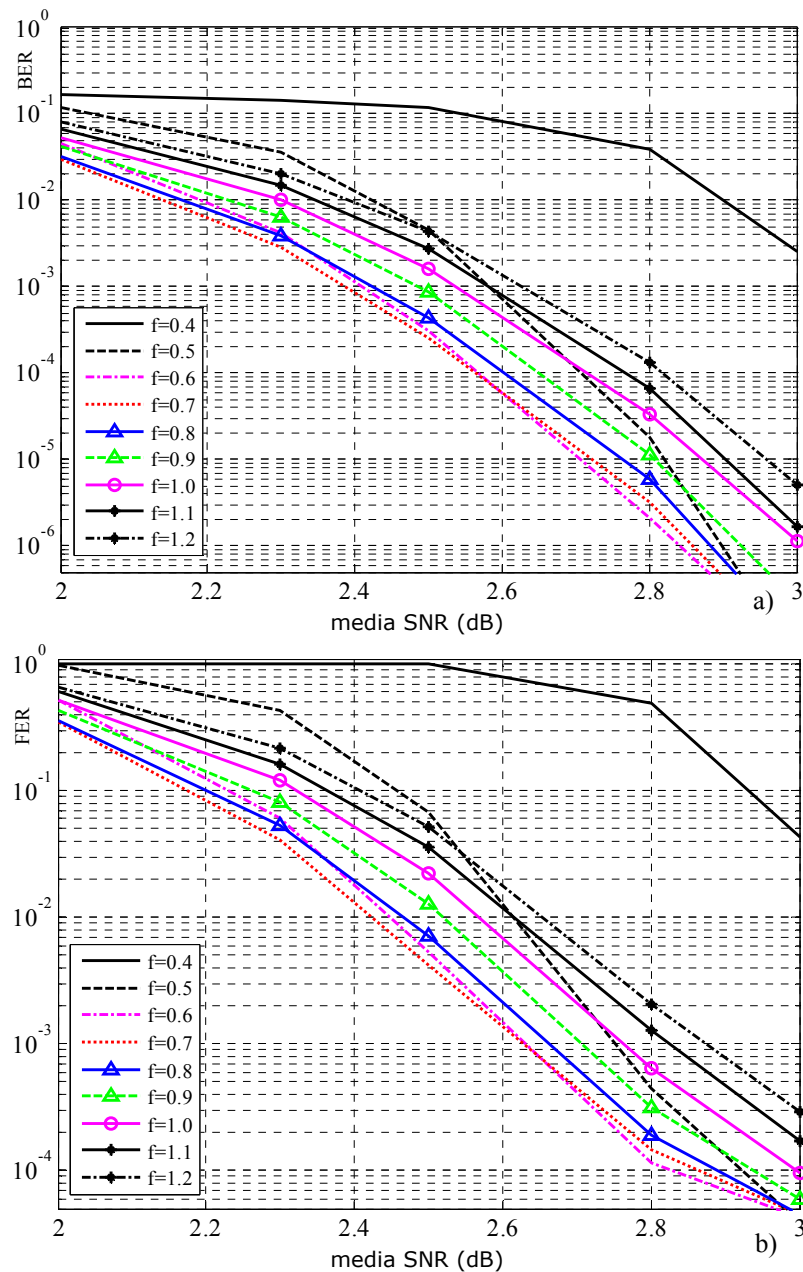


Fig. 4.12 Performanțele: a) BER și b) FER, ale TC-urilor în canale cu fading plat de tip Rayleigh pentru codul 15/13 RSC, utilizând interleaver-ul S pentru diferite valori ale parametrului f .

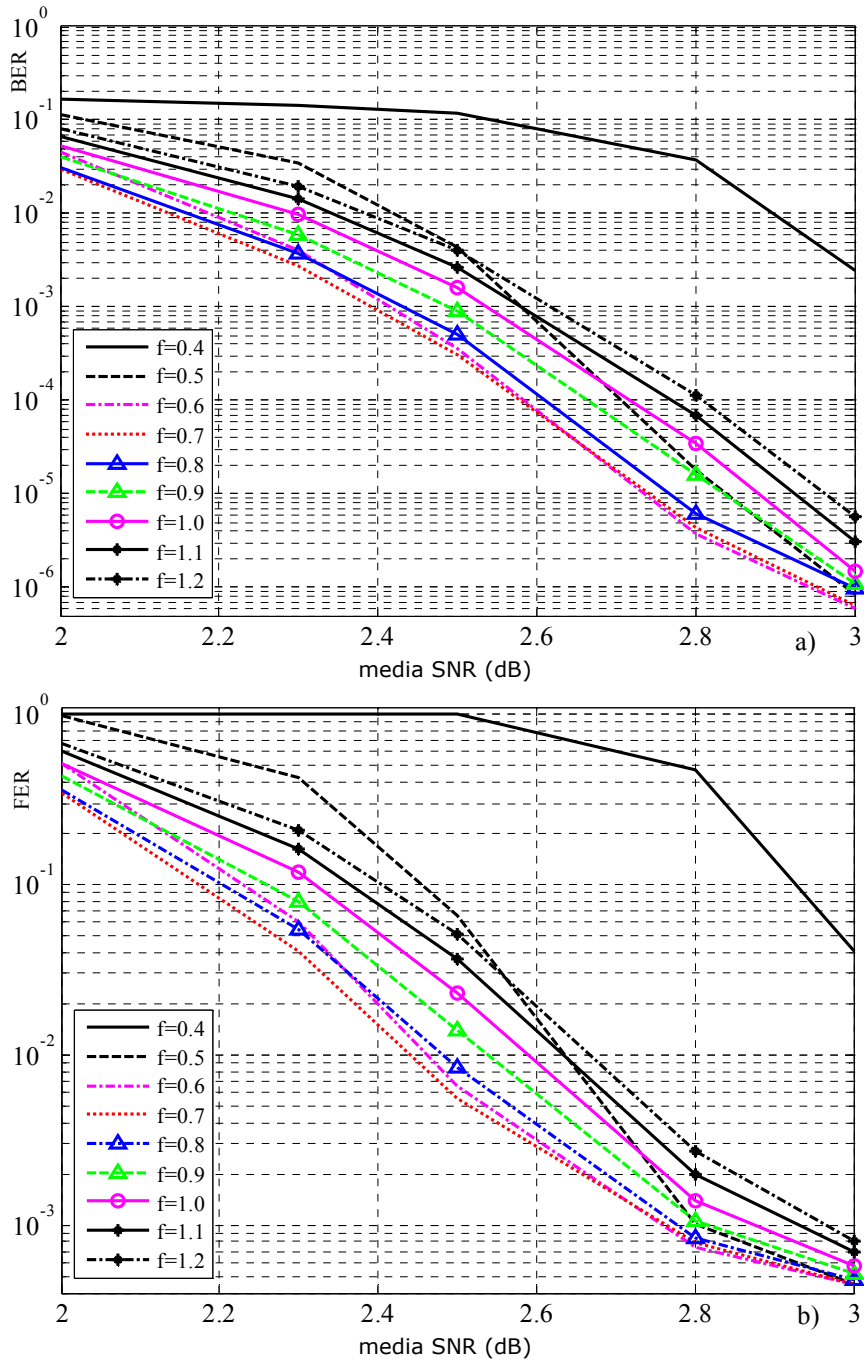


Fig. 4.13 Performanțele: a) BER și b) FER, ale TC-urilor în canale cu fading plat de tip Rayleigh pentru codul 15/13 RSC, utilizând interleaver-ul bloc aleator în linie pentru diferite valori ale parametrului f .

O altă concluzie pe care o putem trage este aceea că, în urma simulărilor pe care le-am efectuat rezultă că utilizarea interleaver-ului bloc aleator în linie în canalul cu fading Rayleigh (ca și în canalul AWGN) are o comportare bună din punct de vedere a performanțelor atingând performanțele BER ale interleaver-ului S , dar care are un avantaj important în practică, față de interleaver-ul S , prin faptul că, construirea sa este mai ușor de realizat.

Indiferent, însă, de aceste mici *detalii*, rămâne evident că utilizarea turbo codurilor contribuie, în cazul transmisiilor în canalele cu fading plat de tip Rayleigh, cu un câștig de zeci de decibeli așa cum am obținut și în [KOV05, BKN05b].

4.4. Fading plat de tip Rice

Fading-ul plat de tip Rice modelează canalele de comunicații radio, [PRO00], pentru care semnalul recepționat (y_k) are o componentă directă (y_{uk} , asimilată canalului fără fading, adică canalului AWGN) și o componentă cu fading (y_{fk} , asimilată canalului Rayleigh), așa cum reiese din modelul unui canal Ricean, pe care l-am propus și prezentat în Fig. 4.14, [BKN06, KBN09].

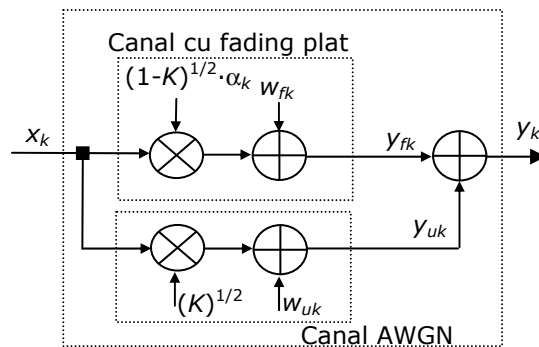


Fig. 4.14 Canal Rice – model pentru simulare.

Parametrul K este definit ca fiind raportul dintre puterea componentei directe a semnalului și puterea totală a semnalului, [SKL88]:

$$K = \text{puterea unei directe} / \text{puterea totală} [\%]. \quad (4.23)$$

Pentru $K=0$ nu există decât componenta fluctuantă. Pentru $K=1$ există doar unda directă, canalul este pur AWGN. Densitatea de probabilitate Rice, dată în relația (4.19), poate fi scrisă ca fiind, [PRO00]:

$$p(r) = \begin{cases} \frac{r}{\sigma^2} \cdot \exp\left(-\frac{r^2 + A^2}{2 \cdot \sigma^2}\right) \cdot I_0\left(\frac{A \cdot r}{\sigma^2}\right), & \text{pentru } r \geq 0 \\ 0, & \text{pentru } r < 0 \end{cases}, \quad (4.24)$$

în care:

$$I_0(x) = \frac{1}{2\pi} \cdot \int_0^{2\pi} e^{x \cdot \cos \psi} d\psi, \quad (4.25)$$

este funcția Bessel modificată, de speța I-a și ordin zero.

În Fig. 4.15 am reprezentat densitățile de probabilitate Rice pentru $\sigma=1$ și câteva valori particulare ale parametrului A , [BAK05b]. În cazul în care $A=0$, distribuția Rice se reduce la distribuția Rayleigh.

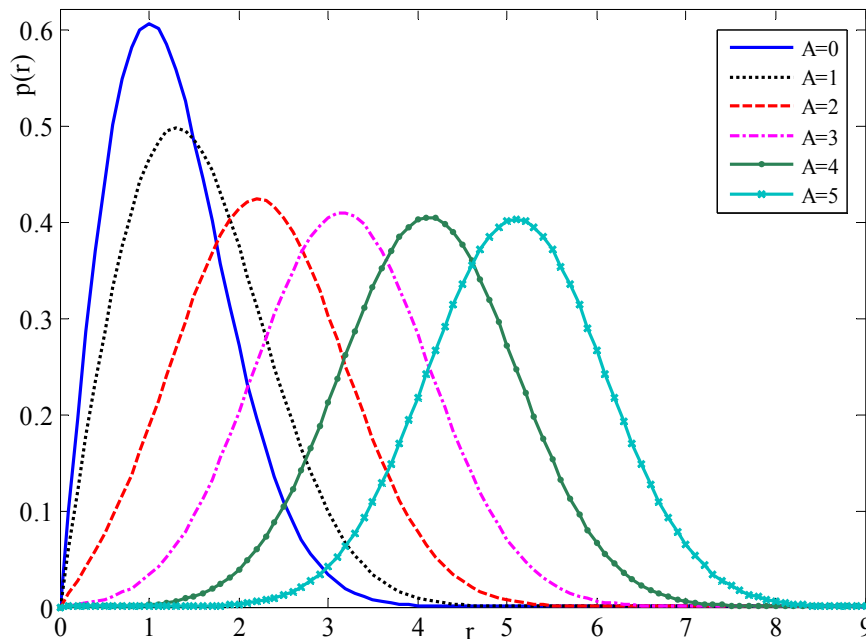


Fig. 4.15 Densitatea de probabilitate Rice de medii $A=0 \div 5$ și abatere standard $\sigma=1$.

O variabilă aleatoare, α , cu distribuție Rice, se obține componând două variabile aleatoare normale x_1 și x_2 , având ambele aceeași dispersie σ^2 , însă doar una cu medie nulă:

$$\alpha = \sqrt{x_1^2 + x_2^2}. \quad (4.26)$$

Fie variabila x_1 de forma:

$$x_1 = x_3 + A, \quad (4.27)$$

unde x_3 este o variabilă normală cu medie nulă și dispersie σ^2 , iar A , valoarea medie nenulă, fiind responsabilă de componenta nefluctuantă a lui x_1 (putând fi asimilată cu o componentă continuă).

Rezultă că:

$$\alpha = \sqrt{(x_3 + A)^2 + x_2^2} = \sqrt{r^2 + 2 \cdot A \cdot r \cdot \cos \Phi + A^2}, \quad (4.28)$$

unde:

$$r = \sqrt{x_2^2 + x_3^2}, \quad (4.29)$$

este o variabilă aleatoare cu distribuție Rayleigh, iar Φ reprezintă faza distribuției complexe ale cărei părți reală și imaginară sunt date de variabilele aleatoare x_2 și x_3 .

Dacă u este o variabilă aleatoare cu distribuție constantă pe $[0, 1]$, atunci prin transformarea:

$$r = F_{RY}^{-1}(u) = \sqrt{-2 \cdot \sigma^2 \cdot \ln(u)}, \quad (4.30)$$

unde F_{RY} reprezintă funcția de repartiție corespunzătoare densității de probabilitate (relația (4.18)), se poate obține o variabilă aleatoare distribuită Rayleigh. Utilizând în continuare relația (4.28), se obține:

$$\alpha = \sqrt{-2 \cdot \sigma^2 \cdot \ln(u_1) + 2 \cdot A \cdot \sqrt{-2 \cdot \sigma^2 \cdot \ln(u_1)} \cdot \cos(\pi \cdot u_2) + A^2}, \quad (4.31)$$

unde u_1 și u_2 reprezintă variabilele aleatoare cu densități de probabilitate uniforme pe $[0, 1]$.

Spre deosebire de generarea variabilei aleatoare cu distribuție Rayleigh, relația (4.28), relația (4.31) prin care se poate genera o variabilă aleatoare cu distribuție Rice, implică două variabile cu distribuție uniformă. Relația (4.31) poate fi scrisă și sub forma:

$$\alpha = \sqrt{-(1-K) \cdot \ln(u_1) + 2 \cdot \sqrt{K \cdot (1-K)} \cdot \sqrt{-\ln(u_1)} \cdot \cos(\pi \cdot u_2) + K}, \quad (4.32)$$

unde K , așa cum am specificat și mai sus, reprezintă raportul dintre puterea componentei fără fading (nefluctuante) și puterea totală a variabilei α , care s-a considerat egală cu 1.

În continuare, în Fig. 4.16 am prezentat rezultatele obținute din simulările pe care le-am efectuat în [KBN09], performanța BER, pentru canale de tip Rice, funcție de parametrul K , pentru trei valori ale $SNR=10, 15$ și 20 dB.

Se observă că pentru o plajă mare de valori ale lui K , până în jur de 80%, canalul se comportă ca și unul pur fluctuant. Performanța BER se îmbunătățește semnificativ abia pentru o proporție de sub 15% a unei fluctuante.

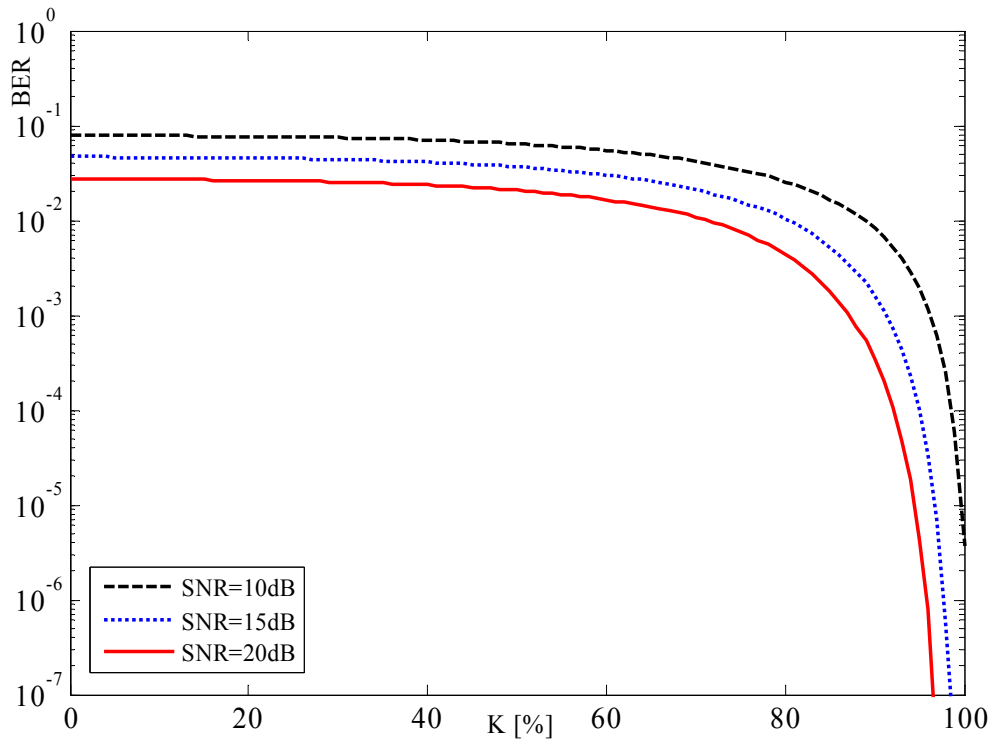


Fig. 4.16 Performanța BER, funcție de parametrul K , pentru $SNR=10, 15$ și 20 dB, pentru canale Rice.

În Fig. 4.17 am reprezentat performanțele BER, pe care le-am obținut în [BKN06] prin simularea sistemului de transmisie cu canal fading plat de tip Rice, pentru diferite valori ale factorului K : $K=0$, $K=25$, $K=50$, $K=75$ și $K=100$ (în procente). Pentru $K=0\%$ canalul este Rayleigh, iar pentru $K=100\%$ canalul este AWGN.

Deoarece valoarea medie a unei variabile de tip Rice este dificil de evaluat, valoarea coeficientului L_c , în vederea simulărilor, s-a ales identică cu cea stabilită pentru fading-ul plat de tip Rayleigh (acesta fiind un caz particular al fading-ului de tip Rice), adică pentru $K=0$, caz în care $\bar{\alpha} = \sqrt{\pi/4} = 0.8862$, [HAW98, VUY01], urmând ca la sfârșitul acestui paragraf să prezint simulările pentru valori ale L_c -ului (relația (4.21)) ce se abat cu câteva procente, și pe care le-am obținut în [BAK05b] și [KBN09].

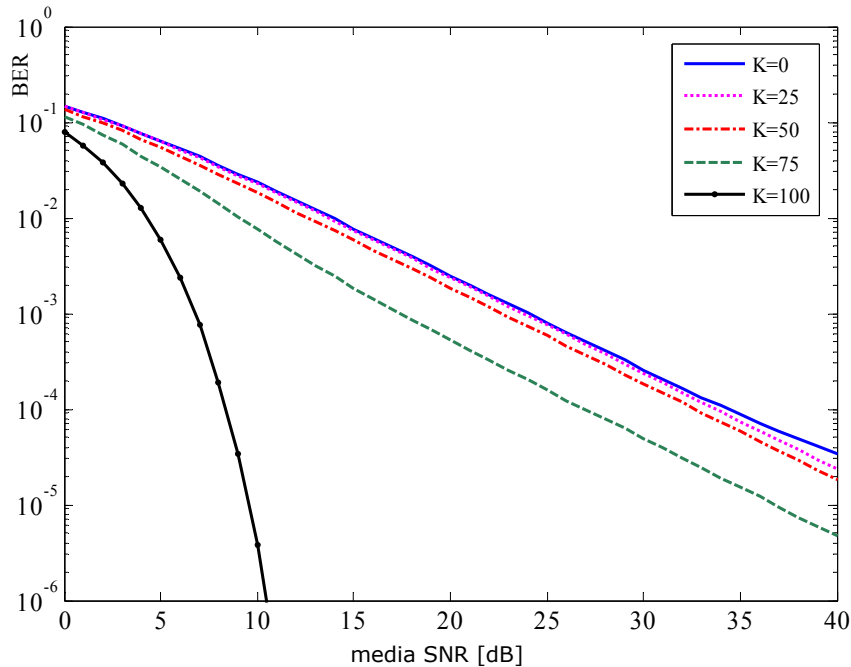


Fig. 4.17 Performanțele BER în canale cu fading plat de tip Rice pentru $K=0, 25, 50, 75$, și $100[\%]$.

În Fig. 4.18 și Fig. 4.19 am prezentat rezultatele simulărilor efectuate cu un TC definit prin parametrii prezentați în tabelul 4.3, pentru cinci valori ale parametrului K .

Tabelul 4.3 Parametrii utilizați.

	Parametrul	Varianta (-ele)
1	Codul component	15/13
2	Configurația TC-ului	paralelă
3	Întreșeserea Lungimea blocurilor	-interleaver-ul S ($N=1784$ $S=29$) -interleaver-ul bloc aleator în linie, BRL
4	Puncturarea	fără
5	Canal	Fading plat Rice
6	Modulație	BPSK
7	Algoritm de decodare	MAP
8	Închiderea trellisului	C1-închis, C2-neînchis $\beta(\)$ de start = echiprob.
9	Nivel de cuantizare	infinat
10	Număr de iterații Criteriu de oprire	15 $prag$ LLR = 10

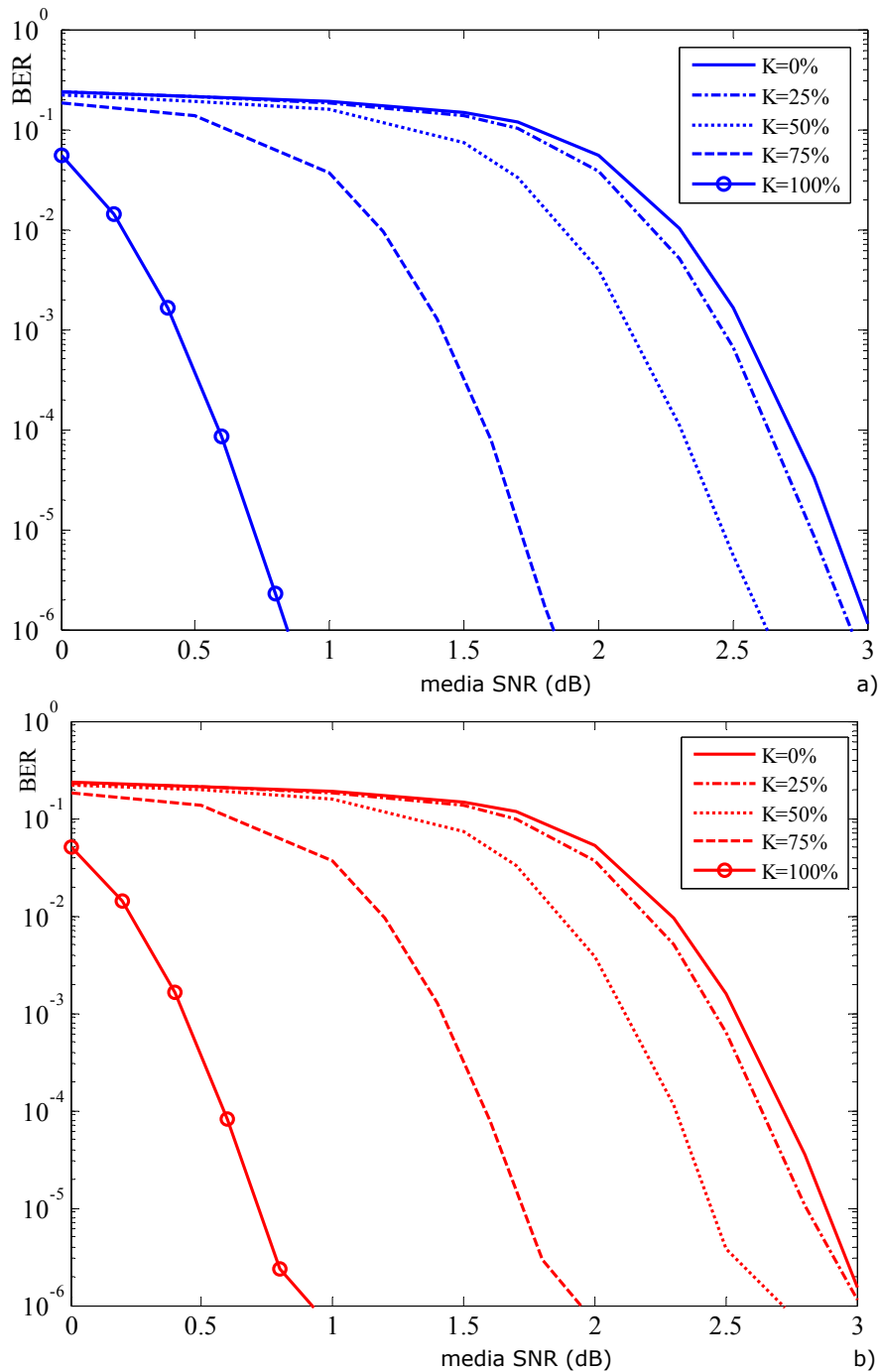


Fig. 4.18 Performanțele BER în canale cu fading plat de tip Rice pentru $K=0, 25, 50, 75$, și 100 [%], pentru cazul în care s-a utilizat: a) interleaver-ul S și b) interleaver-ul bloc aleator în linie, BRL (Block Random in Line).

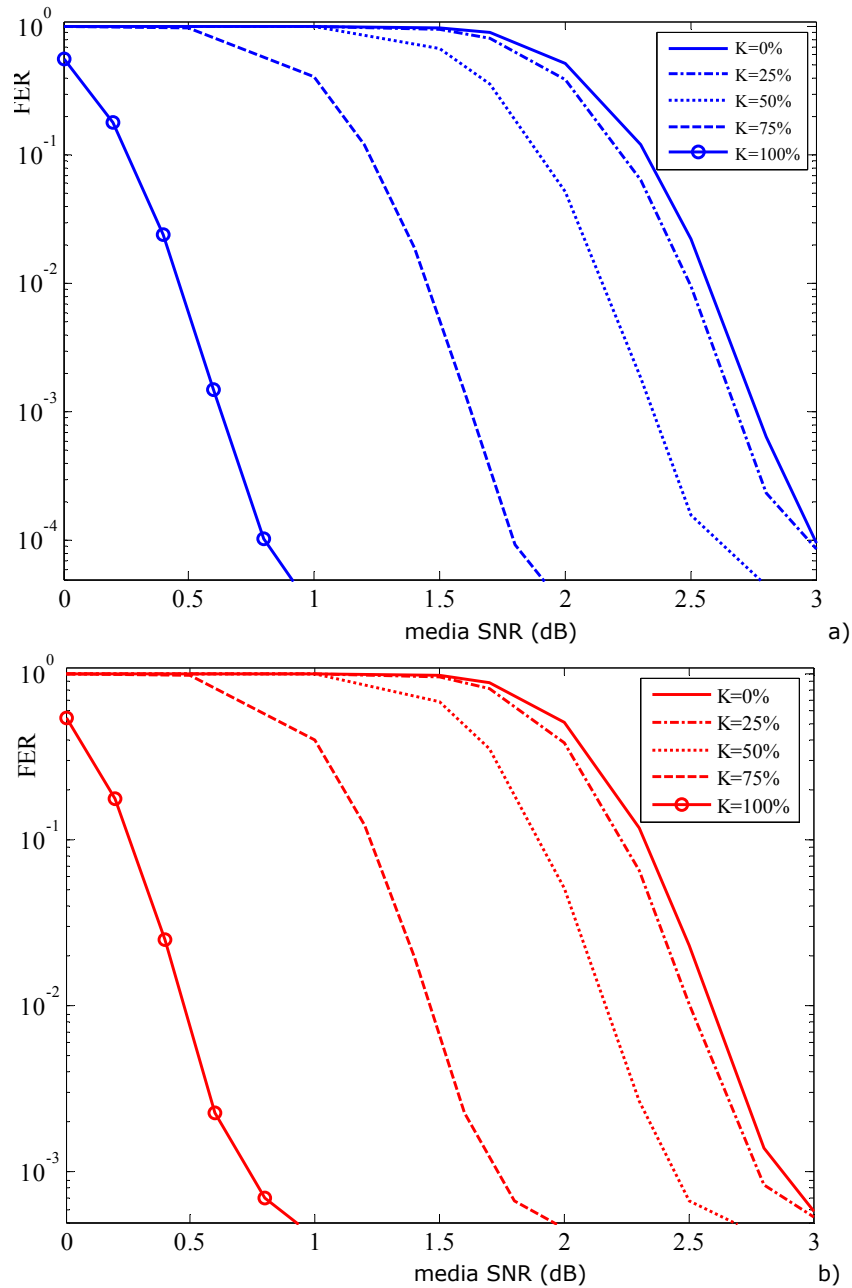


Fig. 4.19 Performanțele FER în canale cu fading plat de tip Rice pentru $K=0, 25, 50, 75$, și 100% , pentru cazul în care s-a utilizat: a) interleaver-ul S și b) interleaver-ul bloc aleator în linie, BRL.

Așa cum se observă din Fig. 4.18 și Fig. 4.19, performanțele TC-urilor în canalele cu fading plat de tip Rice sunt mărginite în partea superioară a figurii de

performanța canalului cu fading plat de tip Rayleigh ($K=0$) și în partea inferioară a figurii de performanța canalului static, AWGN ($K=1$). Performanța TC-urilor nu crește liniar cu parametrul K . Astfel, dacă puterea componentei continue este sub 25% din puterea totală ($K < 0.25$), canalul Rice-an se comportă asemenea canalului fluctuant (Rayleigh). Pentru valori peste 50% ale lui K , performanța BER se îmbunătățește semnificativ. Aceste concluzii sunt valabile atât în cazul folosirii interleaver-ului S cât și a folosirii interleaver-ului bloc aleator în linie.

O concluzie importantă ce rezultă imediat din simulări este aceea că utilizarea turbo codurilor contribuie, și în cazul transmisiilor în canalele cu fading plat de tip Rice, cu un câștig de zeci de decibeli [YUV99, VMB02, BKN06, KBN09]. În continuare prezint o estimare a canalului Rice, cât și Rayleigh ($K=0\%$) și AWGN ($K=100\%$), necesară pentru construcția coeficientului L_c utilizat în algoritmul MAP. Așadar, așa cum arată Tabelul 4.4 și Tabelul 4.5, erori în estimare de până la 20% ale factorului L_c , din relațiile (4.21) și (4.22) influențează performanța TC-ului cu aproximativ 0.2dB.

Tabelul 4.4 $BER \cdot 10^{-9}$ în funcție de f și K când s-a utilizat interleaver-ul S .

K [%]	SNR [dB]	$f=0.4$	$f=0.5$	$f=0.6$	$f=0.7$	$f=0.8$
		BER	BER	BER	BER	BER
0	2.8	38403775	18411	2130	3175	6047
25	2.8	24747613	4454	1102	857	1779
50	2.5	65787759	17243	598	333	638
75	1.8	127592588	3645775	3168	484	230
100	0.8	156845480	113437832	2876700	9045	1536
K [%]	SNR [dB]	$f=0.9$	$f=1.0$	$f=1.1$	$f=1.2$	
		BER	BER	BER	BER	
0	2.8	11504	33293	67827	135499	
25	2.8	5408	8442	18985	31768	
50	2.5	2371	5433	9893	18167	
75	1.8	707	1714	5259	10172	
100	0.8	1197	2244	3249	5218	

Tabelul 4.5 $BER \cdot 10^{-9}$ în funcție de f și K când s-a utilizat interleaver-ul BRL.

K [%]	SNR [dB]	$f=0.4$	$f=0.5$	$f=0.6$	$f=0.7$	$f=0.8$
		BER	BER	BER	BER	BER
0	2.8	36961840	18434	3814	4412	6142
25	2.8	1004489	4270	1512	1684	2253
50	2.5	64317620	15194	1257	1561	1893
75	1.8	127455555	3451242	2580	1578	1333
100	0.8	156914211	113170650	2725095	7796	1632
K [%]	SNR [dB]	$f=0.9$	$f=1.0$	$f=1.1$	$f=1.2$	
		BER	BER	BER	BER	
0	2.8	16023	35052	70013	114227	
25	2.8	5373	10746	18282	34411	
50	2.5	2509	3778	9874	18022	
75	1.8	1817	2967	4637	8709	
100	0.8	1488	2355	3737	5753	

Se poate observa totuși, atât din tabelele 4.4 și 4.5, cât și din graficele prezentate în Anexa C, că pentru diferite valori ale lui K se obțin rezultate mai bune pentru anumite valori ale mărimii f . Prin urmare, în cazul în care $K=0\%$ și 25% , se obțin rezultate BER și FER foarte bune pentru valori ale lui $f=0.6, 0.7$ și 0.8 , trecute în ordinea crescătoare a performanței. Pentru $K=50\%$ și 75% cele mai bune rezultate BER și FER se obțin la valori ale lui $f=0.7, 0.8$ și 0.9 (cel mai bun rezultat se obține pentru $f=0.7$, în cazul în care $K=50\%$, iar pentru $f=0.8$, la $K=75\%$), iar pentru $K=100\%$, performanțele BER și FER sunt foarte bune pentru valori ale lui $f=0.8, 0.9$ și 1 , ceea ce era de așteptat în cazul canalului AWGN, unde teoretic $L_c = 4 \cdot R \cdot B$.

Așadar, din rezultatele experimentale pe care le-am obținut, se poate afirma că pe măsură ce valoarea parametrului K crește trebuie să crească și valoarea coeficientului f , în scopul obținerii unor valori BER și FER cât mai mici. Tot pe baza rezultatelor prezentate rezultă că pentru valori ale lui $K=0\%$ și 25% , coeficientul f trebuie să fie mai mare sau egal cu 0.5 , deoarece se observă că sub această valoare performanțele BER și FER sunt foarte slabe. În cazul în care $K=50\%$ și 75% rezultă că f trebuie să fie mai mare sau cel mult egal cu 0.6 , iar pentru $K=100\%$, f trebuie să fie mai mare sau cel mult egal cu 0.7 .

Pentru a confirma, încă o dată în plus, concluzia de mai sus am prezentat în anexa D tabele ale BER funcție de variația L_c -ului, și pentru cazul în care se consideră încă două tipuri de interleaver-e: interleaver-ul pseudo-aleator, [CCSDS02] și interleaver-ul Takeshita-Costello, [TAC98], cât și un codor 25/23 RSC de memorie 4.

4.5. Fading plat de tip Nakagami

Ambele distribuții prezentate anterior, Rayleigh și Rice, sunt utilizate frecvent în descrierea fluctuațiilor statistice ale semnalelor recepționate dintr-un canal cu fading multical. O altă distribuție, ce este frecvent utilizată în caracterizarea statistică a semnalelor ce se transmit prin astfel de canale, este distribuția Nakagami- m . Această distribuție este relativ nouă, fiind pentru prima dată propusă în 1960, [PRO00], și se utilizează la transmisia *wireless*, [PAR92].

Funcția densității sale de probabilitate este dată de relația [PRO00]:

$$p_N(\alpha) = \frac{2 \cdot m^m}{\Gamma(m)} \cdot \frac{\alpha^{2m-1}}{(2\sigma^2)^m} \cdot \exp\left(-m \frac{\alpha^2}{2\sigma^2}\right) \quad (4.33)$$

unde: - N , este o variabilă aleatoare;

- m se numește *raportul momentelor (fading figure)* și este parametrul care caracterizează fading-ul canalului de transmisie, definit prin următoarea relație:

$$m = \frac{\Omega^2}{E\left[\left(N^2 - \Omega\right)^2\right]}, \quad m \geq \frac{1}{2} \quad (4.34)$$

unde Ω se definește ca:

$$\Omega = E(N^2) = 2\sigma^2. \quad (4.35)$$

Valoarea medie $\bar{\alpha}$ și valoarea medie pătratică $\overline{\alpha^2}$, pentru $m \geq 1$, număr întreg, sunt date de:

$$\bar{\alpha} = \sqrt{2\sigma^2} \cdot \sqrt{\frac{\pi}{4m}} \cdot P, \quad \overline{\alpha^2} = 2\sigma^2, \quad (4.36)$$

unde $P=1$ pentru $m=1$ caz în care ecuația (4.33) se reduce la o densitate de probabilitate Rayleigh și în general $P = \prod_{i=1}^{m-1} \frac{2i+1}{2i}$ pentru $m \geq 2$.

Fără a se pierde din generalitate, se presupune că $\overline{\alpha^2} = 1$, presupunere pe care am făcut-o și în referința bibliografică [BKDB05]. Va rezulta: $2\sigma^2 = 1$ și $\bar{\alpha} = \sqrt{\pi/4m} \cdot P$. În relația (4.33), funcția $\Gamma(m)$ este bine cunoscuta funcție gamma dată de relația:

$$\begin{aligned} \Gamma(m) &= \int_0^{\infty} t^{m-1} \cdot e^{-t} \cdot dt, \quad m > 0; \\ \Gamma(m) &= (m-1)!, \quad m \text{ este un întreg și } m > 0; \\ \Gamma\left(\frac{1}{2}\right) &= \sqrt{\pi}, \quad \Gamma\left(\frac{3}{2}\right) = \frac{1}{2}\sqrt{\pi}. \end{aligned} \quad (4.37)$$

Generarea unui număr aleator cu distribuție Nakagami este realizată prin următoarea schimbare de variabilă:

$$x = m \frac{\alpha^2}{2\sigma^2} = m\alpha^2. \quad (4.38)$$

Prin înlocuirea relației (4.38) în relația (4.33), densitatea de probabilitate Nakagami devine:

$$p_x(x) = \frac{p_N(\alpha)}{\frac{dx}{d\alpha}} = \frac{x^{m-1} \cdot e^{-x}}{\Gamma(m)}. \quad (4.39)$$

Prin integrarea relației (4.39) se obține funcția de repartiție a probabilității variabilei aleatoare x :

$$F_x(x) = \int_0^x p_x(t) \cdot dt = \frac{1}{\Gamma(m)} \cdot \int_0^x t^{m-1} \cdot e^{-t} \cdot dt = \frac{\Gamma_x(m)}{\Gamma(m)}. \quad (4.40 \text{ a})$$

unde s-a folosit notația:

$$\Gamma_x(m) = \int_0^x t^{m-1} \cdot e^{-t} \cdot dt. \quad (4.40 \text{ b})$$

Dacă m este un număr natural, atunci funcția de repartiție poate fi scrisă sub forma :

$$F_x(x) = 1 - \sum_{k=0}^{m-1} \frac{x^k}{k!} \cdot e^{-x}. \quad (4.41)$$

Pentru a asigura convergența generatorului iterativ aleator, este importantă determinarea punctului de inflexiune al funcției de repartiție. $F_x(x)$ are un punct de inflexiune x_F care se obține prin anularea derivatei de ordinul doi, [PRO00, BEC05], așa cum am prezentat și în [BKN05a]:

$$\frac{d^2 F_x(x)}{dx^2} = \frac{dp_x(x)}{dx} = \frac{1}{\Gamma(m)} \cdot [(m-1) \cdot x^{m-2} - x^{m-1}] \cdot e^{-x}, \quad x \geq 0, \quad m \neq 1 \quad (4.42)$$

de unde rezultă că $x_F = m-1$.

Am simulat și reprezentat, în Fig. 4.20, densitatea de probabilitate și funcția de repartiție Nakagami a variabilei aleatoare x , pentru $m=2$.

Distribuția Nakagami- m are aplicații numeroase în modelarea canalelor radio cu fading [BEC05]. Prin intermediul parametrului m , se pot modela semnale cu fading, a căror stare poate varia de la valori mari la valori moderate. De asemenea, în literatură, a fost raportată analiza performanțelor teroretice și numerice, ale diverselor sisteme de comunicații ce operează cu fading Nakagami, [WON04, BEAD91]. Totuși, au fost raportate puține rezultate referitoare la generarea variabilei aleatoare Nakagami- m [BEC05]. De aceea, în [BKDB05] am prezentat un nou algoritm de generare a numerelor aleatoare cu distribuție Nakagami, algoritm ce va fi utilizat în continuare pentru a evalua comportamentul TC-urilor în canalele cu fading plat Nakagami.

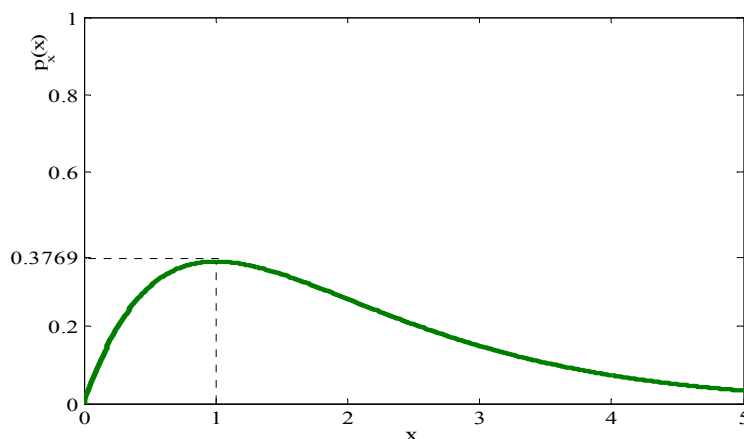


Fig. 4.20 a) Densitatea de probabilitate a variabilei aleatoare x , pentru $m=2$.

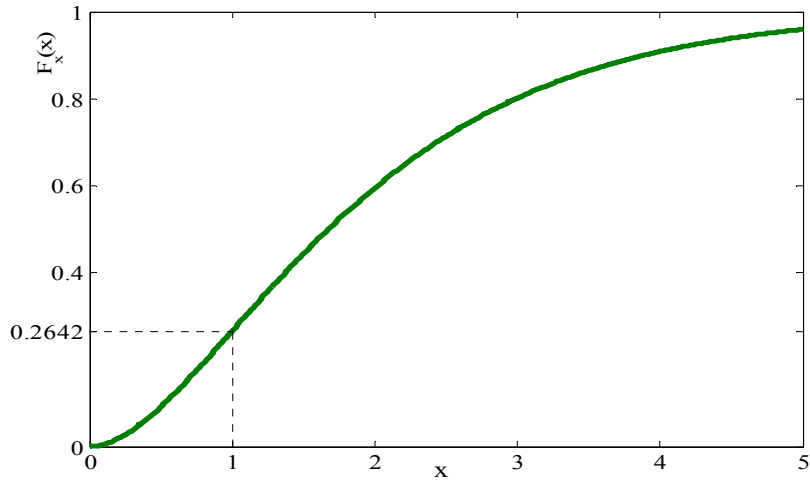


Fig. 4.20 b) funcția de repartiție a probabilității, a variabilei aleatoare x , pentru $m=2$.

În Fig. 4.21, am arătat rezultatele simulărilor BER ale unui sistem de transmisie necodat printr-un canal cu fading plat Nakagami, pe care le-am obținut pentru diferite valori ale parametrului m .

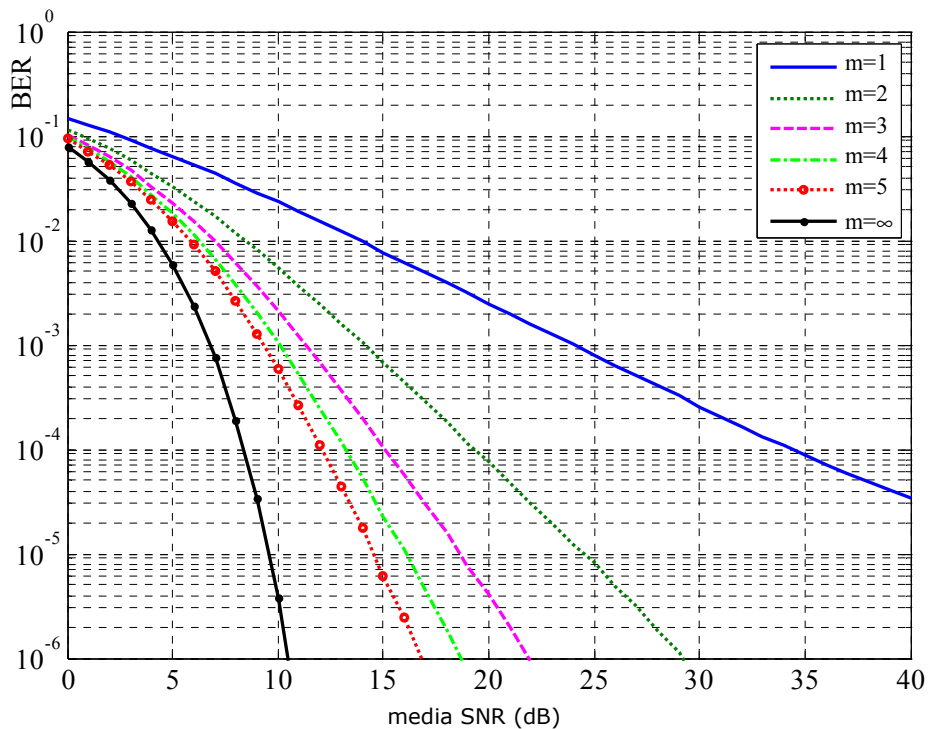


Fig. 4.21 Performanțele BER obținute prin simularea unui sistem de transmisie cu modulație BPSK în canale cu fading plat de tip Nakagami cu $m=2, 3, 4$ și 5 . Lui $m=1$ îi corespunde canalul Rayleigh și lui $m=\infty$ îi corespunde canalul fără fading.

Performanțele unui sistem de transmisie necodat în canalele cu fading plat de tip Nakagami, cu $m > 1$ sunt mărginite în partea superioară a figurii de performanțele fading-ului plat de tip Rayleigh (ce corespunde canalului Nakagami cu $m=1$) și mărginite în partea inferioară a figurii de canalele fără fading (ce corespund canalului Nakagami cu $m=\infty$). Așa cum era de așteptat, pierderea în performanță crește pe măsură ce crește m (selectivitatea în timp a canalului). De exemplu la un $\text{BER}=10^{-4}$ pierderea între $m=2$ și $m=1$ este de 15 dB.

În cele ce urmează, în Fig. 4.22 și Fig. 4.23, am prezentat performanțele unui sistem de transmisie codat în canale Nakagami- m . Turbo codul utilizat, de rată $1/3$, are în componența sa două coduri RSC (13/15) de rată $1/2$ și un interleaver S ($S=29$). Lungimea cuvintelor de cod este de 1784 biți. Aplic un criteriu de oprire la sfârșitul iterației curente pentru a verifica dacă secvența decodată se potrivește sau nu cu un cuvânt de cod, iar numărul maxim de iterații este setat pe 15.

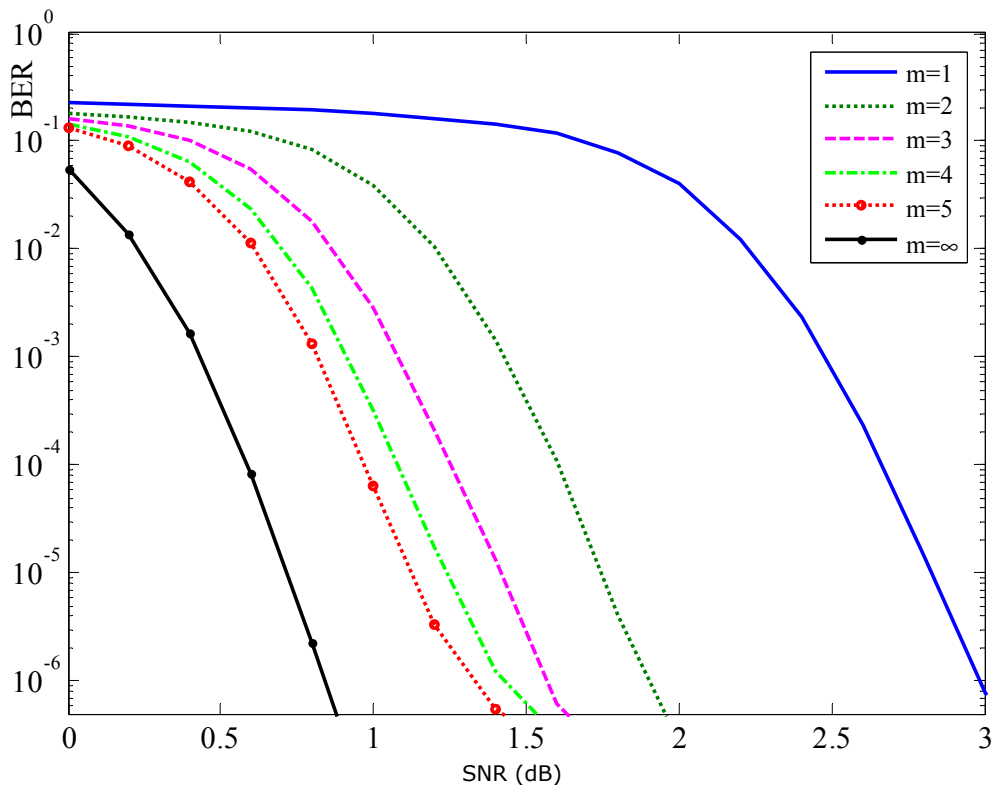


Fig. 4.22 Performanțele BER obținute prin simularea unui sistem de transmisie turbo codat, cu modulație BPSK, în canale cu fading plat de tip Nakagami- m cu $m=2, 3, 4$ și 5 .

O primă concluzie ce rezultă este aceea că performanțele unui sistem de transmisie codat, cu un TC, în canalele cu fading plat de tip Nakagami, cu $m > 1$, sunt mărginite în partea superioară a figurii de performanțele fading-ului plat de tip Rayleigh ($m=1$) și mărginite în partea inferioară a figurii de canalele fără fading ($m=\infty$), ca și în cazul unui sistem de transmisie necodat.

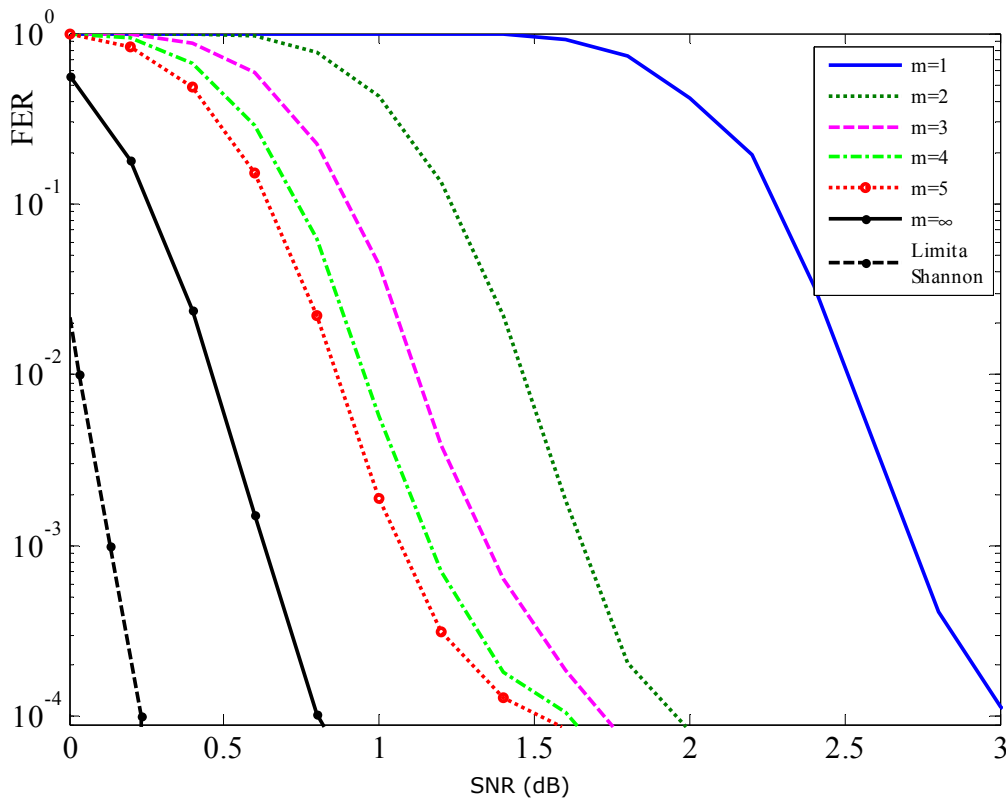


Fig. 4.23 Performanțele FER obținute prin simularea unui sistem de transmisie turbo codat, cu modulație BPSK, în canale cu fading plat de tip Nakagami- m cu $m=2, 3, 4$ și 5 .

Din rezultatele pe care le-am obținut am constatat că TC-ul se comportă bine în canalul selectiv în timp, chiar și pentru lungimi moderate ale cuvintelor de cod, (ca în cazul nostru $N=1784$).

O altă concluzie ce rezultă imediat din simulări este aceea că utilizarea TC-urilor și în cazul canalului cu fading plat de tip Nakagami (caz ce modelează o gamă largă de situații practice) oferă câștiguri de zeci de decibeli.

Pentru aproape toate valorile BER și FER, pierderea în performanțe dintre $m=5$ și $m=1$, este mai mică de 2 dB, așa cum rezultă din simulările pe care le-am efectuat în [BKDB05]. Acest fapt arată că am luat în considerare în mod eficient parametrul m pentru calculul coeficientului L_c , relația (4.17), în care $\bar{\alpha}$ depinde de m , cu scopul de a construi raportul de plauzibilitate al canalului. Dacă $m=1$, $\bar{\alpha} = \sqrt{\frac{\pi}{4}} = 0.8862$, [HAW98].

Pentru calculul L_c -ului este necesară cunoașterea SNR-ului (relația (4.21)). Astfel, în practică, este necesară o estimare a SNR-ului. De aceea în cele ce urmează am studiat impactul erorii de estimare în calculul raportului de plauzibilitate a canalului L_c . Pentru a lua în considerare estimarea erorii se scalează valoarea teoretică L_c cu un factor definit ca fiind raportul: $f_{L_c} = L_c \text{ utilizat} / L_c \text{ teoretic}$. De exemplu, dacă f_{L_c} este mai mare decât unu, SNR a fost supraestimat și invers.

În Tabelul 4.6 sunt trecute rezultate BER pentru diferite valori ale lui m , $m=2\div 5$, în funcție de factorul de scalare f_{L_c} .

Tabelul 4.6 Impactul erorii de estimare a SNR-ului în algoritmul de turbo decodare pentru câteva valori ale lui m . Parametrul m pentru distribuția Nakagami reprezintă selectivitatea în timp a canalului.

m	2	3	4	5
SNR [dB]	1.8	1.4	1.2	1.0
$f_{L_c} = 0.7$	BER=3.27e-6	BER=1.35e-5	BER=3.66e-5	BER=2.18e-4
$f_{L_c} = 0.8$	BER=2.00e-6	BER=1.01e-5	BER=1.37e-5	BER=5.44e-5
$f_{L_c} = 0.9$	BER=2.76e-6	BER=1.06e-5	BER=1.54e-5	BER=5.07e-5
$f_{L_c} = 1.0$	BER=3.70e-6	BER=1.34e-5	BER=1.76e-5	BER=6.44e-5
$f_{L_c} = 1.1$	BER=8.70e-6	BER=2.06e-5	BER=2.45e-5	BER=1.03e-4
$f_{L_c} = 1.2$	BER=1.71e-5	BER=2.95e-5	BER=5.36e-5	BER=1.82e-4

O altă concluzie ce rezultă este că în mod surprinzător, pentru valori scalate ale lui L_c cu aproximativ 10% mai mici decât cele teoretice, date de relația (4.17) (pentru factorul de scalare egal cu 0.9) algoritmul acesta de decodare MAP, pe care l-am propus și în [BKDB05], are rezultate mai bune decât în cazul optimal. Mai mult, aproximarea pe care am făcut-o este deosebit de robustă la estimarea erorii SNR-ului deoarece performanțele BER rămân acceptabile și la erori de estimare a SNR-ului mai mici de 10%. Interesant este faptul că am obținut că estimarea canalului trebuie realizată cu o acuratețe mai mică decât 10% pentru a se lucra în condiții optime cu algoritmul de decodare MAP, [BKDB05].

4.6. Concluzii

În acest capitol am analizat performanțele BER și FER ale turbo codurilor în canalele cu fading plat de tip Rayleigh, Rice și Nakagami.

O concluzie importantă, ce a rezultat în urma simulărilor pe care le-am realizat în cadrul acestui capitol, este aceea că utilizarea turbo codurilor în transmisiile în canalele cu fading plat aduce un câștig de zeci de decibeli, [KOV05, BKN05b, BKN06, KBN09, BKDB05].

În paragrafele 4.3 și 4.4, în analiza performanțelor am folosit două tipuri de interleaver-e. Unul, propus de mine, interleaver-ul bloc aleator în linie (BRL) pe care l-am descris în paragraful 3.4.6.1, iar celălalt este un interleaver S. Rezultatele experimentale obținute au dus la concluzia că performanțele BER ale celor două interleaver-e sunt similare. Avantajul adus însă de interleaver-ul BRL este că are o construcție mai simplă, [KOB09, KBN09].

Deoarece, am utilizat turbo codul într-un canal cu fading, am modificat algoritmul MAP, modificarea constând în utilizarea coeficientului L_c , relația (4.17). În paragrafele 4.3, 4.4 și 4.5 s-a realizat o estimare a canalului, necesară pentru construcția acestui coeficient L_c , relația (4.21).

S-a putut observa faptul că erori în estimare de până la 20% ale factorului L_c , influențează performanța TC-ului, în canalele cu fading plat de tip Rayleigh și Rice, cu aproximativ 0.2 dB, [KBN09]. În canalele cu fading plat de tip Nakagami algoritmul de decodare MAP, pe care l-am propus, are rezultate mai bune decât în cazul optimal, pentru valori scalate ale lui L_c cu aproximativ 10% mai mici decât cele teroretice (pentru factorul de scalare egal cu 0.9), [BKDB05] .

5. TURBO CODURI MULTI-BINARE

Turbo codurile multi-binare, TCMB (Multi-Binary Turbo Code - MBTC) sau turbo codurile ne-binare, [BEJ99, BJK01, GHA04, BAK05a], ca și TC-urile clasice, formează o clasă de coduri corectoare de erori ce operează în apropierea limitei Shannon. Printre avantajele TCMB-urilor față de TC-uri se poate menționa faptul că au o convergență mai rapidă și un efect de error floor mai scăzut, [DOB05]. Un avantaj principal al TCMB-urilor este acela că distanța lor minimă, în general este mai mare decât cea a TC-urilor binare.

În Fig. 5.1 se prezintă schema unui TCMB în configurație paralelă, [BKBP09, BDK06]. Trebuie precizat faptul că, în cazul TCMB-urilor, codoarele componente (C1 și C0) prezintă R intrări, așadar toate prelucrările ce privesc secvența de informație se vor face pe R biți. O deosebire esențială față de TC-uri este aceea că în cadrul algoritmului MAP nu se mai calculează LLR-urile ci probabilitățile a-posteriori (APP - A Posteriori Probability), [BKDVL06].

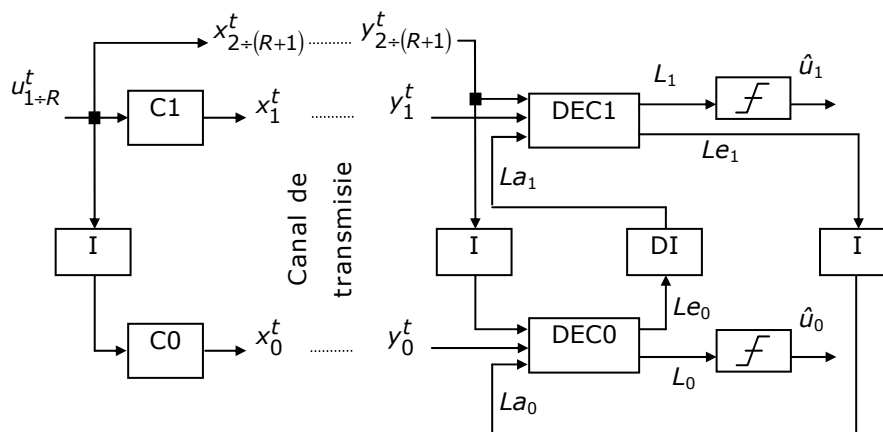


Fig. 5.1 Turbo cod multi-binar. Schema generală.

Notațiile făcute în cadrul figurii au următoarele semnificații:

- C1 și C0 sunt codoarele componente, iar DEC1 și DEC0 sunt decodoare;
- I/DI, sunt inițialele pentru interleaver/de-interleaver;
- $u = (u^1, u^2, \dots, u^T)$, reprezintă blocul simbolurilor de informație de lungime

T (astfel un bloc de informație conține T simboluri sau $R \cdot T = N$ biți, cu $r \geq 2$ intrări),

cu $u^t = (u_1^t, u_2^t, \dots, u_R^t)^T$;

- $x = (x^1, x^2, \dots, x^T)$, reprezintă secvența codată emisă, cu $x^t = (x_0^t, x_1^t, \dots, x_{R+1}^t)^T$ și $x_r^t = \pm 1$;
- $y = (y^1, y^2, \dots, y^T)$, reprezintă secvența recepționată, cu $y^t = (y_0^t, y_1^t, \dots, y_{R+1}^t)^T$ și $y_r^t = x_r^t + w_r^t$, unde w_r^t este o variabilă aleatoare gaussiană de medie zero și dispersie σ^2 ;
- $\hat{u}_j = (\hat{u}_j^1, \hat{u}_j^2, \dots, \hat{u}_j^T)$, reprezintă estimatul lui u , dat de DEC1, respectiv de DEC0, cu $\hat{u}_j^t = (\hat{u}_{j,1}^t, \hat{u}_{j,2}^t, \dots, \hat{u}_{j,R}^t)$ și cu $j=1$ sau 0 ;
- $La_j^{t,i}(d)$, $Le_j^{t,i}(d)$ și $L_j^{t,i}(d)$ probabilitățile *a-priori*, extrinseci și *a-posteriori* de la decodorul j , $j = 0$ sau 1 (după iterația i , și pasul t), pentru ca simbolul original u_j^t să fie estimat la întregul d , adică probabilitățile indicate ca $u_j^t = d \in J = \{0, 1, \dots, (2^R - 1)\}$.

Decodarea poate fi făcută pe bit, [XIA03] sau pe simbol, [TAS00]. Decodarea pe bit este aceeași cu cea utilizată în TC-urile clasice, [BGT93].

Așa cum am precizat în [BKBP09], decodarea pe simbol constituie o *analiză mai fină* a decodorului asupra secvenței recepționate. Astfel, considerând o secvență de date de lungime N biți, decodorul uni-binar, la fel ca și cel multi-binar cu decodare pe bit, calculează N valori LLR. Pentru codorul multi-binar cu R intrări, blocul de date de $R \times T$ biți fiind organizat într-o matrice cu dimensiunile $R \times T$, decodarea multi-binară pe simbol implică calculul a 2^R valori APP pentru fiecare simbol din cele L . Rezultă un total de $T \cdot 2^R$ valori APP pentru întreaga secvență de N biți, care reprezintă o *analiză cu atât mai fină* a decodorului asupra secvenței recepționate, cu cât R este mai mare.

În continuare analizez cele două metode de decodare, pe bit și pe simbol, în cazul turbo-codurilor multi-binare ce utilizează în decodare algoritmi de tip MAP, prin intermediul simulărilor, cu scopul de a stabili ierarhia în performanța BER și FER.

A. Decodarea pe bit

Considerând că cele două decodoare DEC1 și DEC0 implementează algoritmul de decodare iterativ MAP, decodarea TCMB-urilor pe bit decurge asemănător cazului unibinar. Voi descrie în continuare atât algoritmul MAP cât și turbo-decodarea bazată pe acesta.

Algoritmul MAP propus în [BCJR74] calculează, conform relației (3.30), LLR-ul pentru fiecare bit, u_r^t , cu $1 < t < T$, și $1 < r < R$, din fiecare simbol al secvenței originale, u , sub forma:

$$L_r^t = \ln \left(\frac{\sum_{\substack{(\hat{s},s) \Rightarrow \\ u_r^t = +1}} \alpha_{t-1}(\hat{s}) \cdot \gamma_t(\hat{s},s) \cdot \beta_t(s)}{\sum_{\substack{(\hat{s},s) \Rightarrow \\ u_r^t = -1}} \alpha_{t-1}(\hat{s}) \cdot \gamma_t(\hat{s},s) \cdot \beta_t(s)} \right) \quad (5.1)$$

unde $\alpha_{t-1}(\hat{s})$ este probabilitatea ca trellis-ul codorului să fie în starea \hat{s} la momentul $t-1$ și secvența recepționată din canal, înainte de acest moment să fie $[y^1, y^2, \dots, y^{t-1}]$.

Conform relației (3.32), coeficienții alfa pot fi calculați prin recursivitate înainte cu relația:

$$\alpha_t(s) = \sum_{\text{toti } \hat{s}} \gamma_t(\hat{s},s) \cdot \alpha_{t-1}(\hat{s}), \quad (5.2)$$

$\beta_t(s)$ este probabilitatea ca, dacă la momentul t trellis-ul se află la momentul s , viitoarea secvență recepționată din canal să fie $[y^t, y^{t+1}, \dots, y^T]$:

$$\beta_{t-1}(\hat{s}) = \sum_{\text{toti } s} \gamma_t(\hat{s},s) \cdot \beta_t(s), \quad (5.3)$$

$\gamma_t(\hat{s},s)$ este probabilitatea ca trellis-ul să treacă din starea \hat{s} în starea s și secvența recepționată din canal să fie y^t , așa cum am prezentat-o în [BKBP09]:

$$\gamma_t(\hat{s},s) = \exp \left(\sum_{r=1}^R u_r^t \cdot La_r^t / 2 + \sum_{r=0}^R L_c \cdot y_r^t \cdot x_r^t \right). \quad (5.4)$$

În relația (5.4), La_r^t reprezintă informația extrinsecă corespunzătoare bitului r al simbolului t , iar L_c reprezintă factorul de încredere al canalului.

Procesul iterativ de turbo-decodare presupune schimbul de informații extrinseci, calculate la ieșirile celor două decodare prin relațiile:

$$\begin{aligned} Le_{1,r}^{t,i} &= L_{1,r}^{t,i} - La_{1,r}^{t,i} - L_c \cdot y_{1,r}^t / 2, \\ Le_{0,r}^{t,i} &= L_{0,r}^{t,i} - La_{0,r}^{t,i} - L_c \cdot y_{0,r}^t / 2. \end{aligned} \quad (5.5)$$

Informațiile *a-priori*, la fiecare iterație i , sunt obținute prin întrețeserea secvențelor de informații extrinseci:

$$\begin{aligned} La_{1,r}^{t,i} &= \pi^{-1} \left(Le_{0,r}^{t,(i-1)} \right), \\ La_{0,r}^{t,i} &= \pi \left(Le_{1,r}^{t,(i-1)} \right). \end{aligned} \quad (5.6)$$

Operațiile $\pi(\cdot)$ și $\pi^{-1}(\cdot)$ semnifică întrețeserea, respectiv de-întrețeserea (notate cu „I” și „DI” în Fig. 5.1).

B. Decodarea pe simbol

Decodarea pe simbol (cuvânt) presupune ca pentru fiecare caracter din secvența de date să se calculeze setul probabilităților *a-posteriori* (APP). În acest caz, probabilitățile APP, extrinseci și *a-priori* se referă, în fiecare moment t de timp, la un caracter din cele 2^R posibile. Cu notațiile din Fig. 5.1, se pot scrie ecuațiile:

$$\begin{aligned} L_1^{t,i}(d) &= La_1^{t,i}(d) + Y_1^t + Le_1^{t,i}(d), \\ L_0^{t,i}(d) &= La_0^{t,i}(d) + Y_0^t + Le_0^{t,i}(d), \end{aligned} \quad (5.7)$$

iar probabilitățile *a-priori* se calculează cu relațiile:

$$\begin{aligned} La_1^{t,i}(d) &= \pi^{-1}\left(Le_0^{t,(i-1)}(d)\right), \\ La_0^{t,i}(d) &= \pi\left(Le_1^{t,(i-1)}(d)\right). \end{aligned} \quad (5.8)$$

Y_1^t și Y_0^t sunt termeni calculați pe baza secvenței recepționate și a dispersiei zgomotului σ^2 :

$$Y_j^t = \frac{1}{\sigma^2} \cdot \sum_{r=2}^{R+1} x_r^t \cdot y_r^t + \frac{1}{\sigma^2} \cdot x_j^t \cdot y_j^t \quad (5.9)$$

cu $j = 0$ sau 1 . De remarcat că setul probabilităților extrinseci este numeric egal cu cel al ramurilor ce pleacă dintr-un nod al trellis-ului (fiecare ramură corespunde unei valori posibile a cuvântului de informație), [BKBP09].

În Fig. 5.2.a) și b) am prezentat performanțele BER și FER ale TCMB-urilor obținute din simulările pe care le-am efectuat în [BKBP09], pentru ambele tipuri de decodare (pe bit și pe simbol). Pentru a obține valori mici ale BER și FER în diagrame, am generat 500000 de blocuri a câte 1504 biți (adică, 752 simboluri duobinare). Cu alte cuvinte, pentru întrețesere am folosit interleaver-e aleatoare, cu lungimea de 1504 biți în cazul MBTC-ului cu decodare pe bit, iar în cazul MBTC-ului cu decodare pe simbol un interleaver aleator cu lungimea de 752 biți, [CRO00].

Datorită decodării separate în cazul decodării pe bit a biților din simbol (se calculează câte un LLR pentru fiecare bit) este posibil ca blocul de date să fie rearanjat într-un vector linie cu dimensiunea $N=2 \cdot T$ și să fie întrețesut cu un interleaver de această lungime.

În cazul decodării pe simbol această operație (de rearanjare în vector și întrețesere binară) nu mai este posibilă deoarece decodoarele componente calculează APP-uri pentru întreg simbolul, ca atare biții componenți nu pot fi separați.

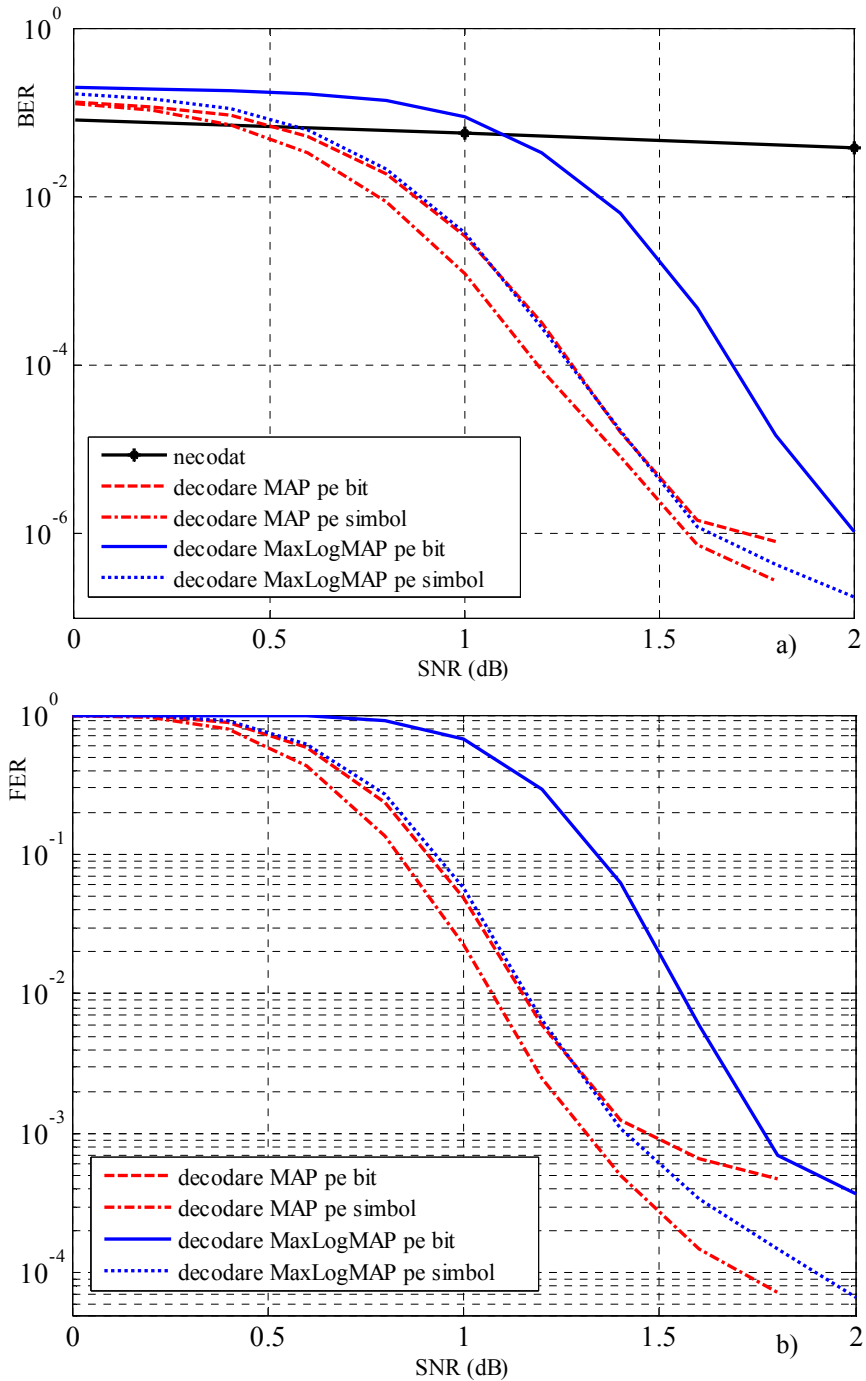


Fig. 5.2. Performanțele: a) BER și b) FER, ale TCMB-urilor, utilizând codurile RSC de memorie 3, având matricea generatoare $\mathbf{G}=[13 \ 15 \ 11]_{10}$, în cazul folosirii decodărilor pe bit și pe simbol.

Codorul convoluțional RSC folosit în realizarea TCMB-ului, are matricea generatoare $\mathbf{G}=[13\ 15\ 11]_{10}$, iar schema sa am prezentat-o în [BKBP09]. Am considerat un canal AWGN, nu am utilizat puncturare astfel încât rata de codare este $1/2$, numărul maxim de iterații pe care l-am considerat este egal cu 15 și am folosit criteriul de oprire al iterațiilor, care este prezentat pe larg în [BDK06].

Analizând rezultatele pe care le-am obținut, în Fig. 5.2 a) și b), rezultă faptul că metoda de decodare pe simbol este superioară metodei de decodare pe bit. Un alt lucru remarcabil ce poate fi observat este acela că, atunci când se folosește decodarea pe bit, diferența dintre algoritmul de decodare MAP și MaxLogMAP este aproximativ egală cu 0.5 dB, iar când se folosește decodarea pe simbol diferența dintre cei doi algoritmi de decodare este mai mică de 0.1 dB.

Diminuarea efectului de *error floor* este un alt avantaj important pe care îl aduce decodarea pe simbol față de decodarea pe bit. Acest efect este substanțial în curbele FER, unde *error floor* apare imediat sub o valoare a FER-ului de 10^{-3} , în cazul folosirii decodării pe bit. În schimb, în cazul folosirii decodării pe simbol, acest efect este practic inexistent.

În continuare, prezint în Fig. 5.3 structura generală a unui codor RSC cu intrări multiple, pe care am folosit-o și în [KDBN06, BDBK08], conform cu [DOB05], corespunzătoare celor două codoare (C1 și C0). Forma canonică prezentată poate fi denumită ca fiind forma canonică H. Codorul este bazat pe registrul de deplasare cu reacție, LFSR (Linear Feedback Shift Register), cu r intrări. Acest codor, în general, nu este decompozabil în r codoare uni-intrare, cu alte cuvinte nu există câte un registru cu reacție pentru fiecare intrare. Mai există o altă formă canonică, G, în structura căreia există un registru de reacție, pentru o unică ieșire c , [JOZ99]. O echivalare între cele două forme canonice a fost realizată în [BDBK08].

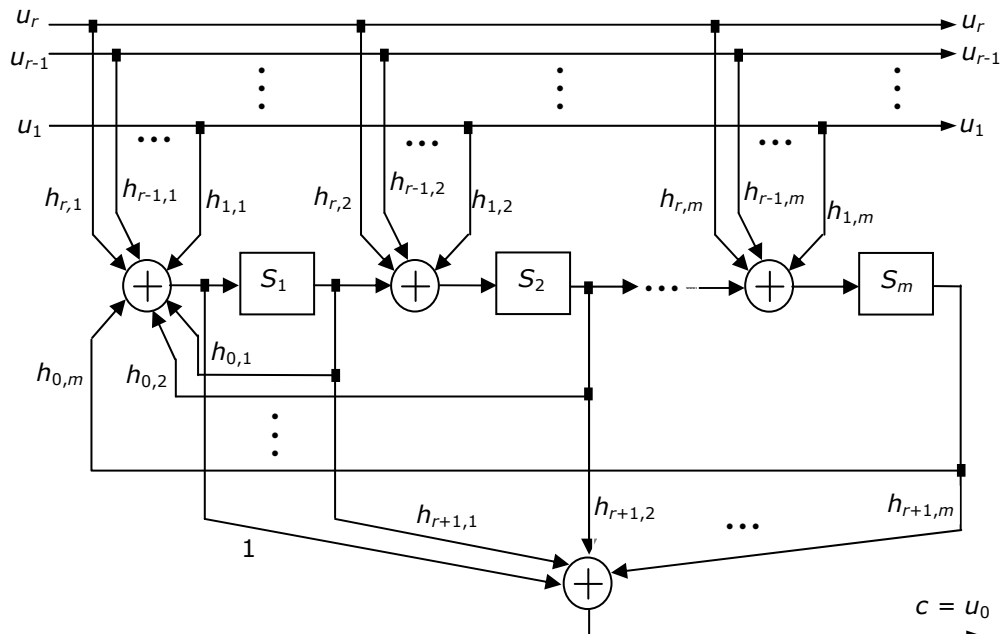


Fig. 5.3 Structura canonică a unui codor RSC multi-intrare, de rată $r/(r+1)$, construit pe baza registrului de deplasare cu reacție cu r intrări.

Cu $\mathbf{S}_t = [s_m^t \dots s_2^t s_1^t]^T$ se notează vectorul stare a codorului de la momentul t , iar cu $\mathbf{U}_t = [u_r^t u_{r-1}^t \dots u_1^t]^T$ se notează vectorul intrare de la momentul t . Operatorul $(\cdot)^T$ semnifică transpunerea. Matricea generatoare completă a codorului, reprezentat în Fig. 5.3, este de forma:

$$\mathbf{H} = \begin{bmatrix} h_{r+1,m} & h_{r,m} & \dots & h_{1,m} & h_{0,m} \\ \dots & \dots & \dots & \dots & \dots \\ h_{r+1,2} & h_{r,2} & \dots & h_{1,2} & h_{0,2} \\ h_{r+1,1} & h_{r,1} & \dots & h_{1,1} & h_{0,1} \end{bmatrix}. \quad (5.10)$$

Eliminând din matricea \mathbf{H} prima și ultima coloană, corespunzătoare vectorului ieșirii, respectiv reacției, se obține matricea generatoare restrânsă \mathbf{H}_0 :

$$\mathbf{H}_0 = \begin{bmatrix} h_{r,m} & \dots & h_{1,m} \\ \dots & \dots & \dots \\ h_{r,2} & \dots & h_{1,2} \\ h_{r,1} & \dots & h_{1,1} \end{bmatrix} \quad (5.11)$$

vectorii de reacție și de ieșire fiind astfel de forma:

$$\begin{aligned} \mathbf{H}_R &= [h_{0,m} \dots h_{0,2} h_{0,1}]^T, \\ \mathbf{H}_{out} &= [h_{r+1,m} \dots h_{r+1,2} h_{r+1,1}]^T. \end{aligned} \quad (5.12)$$

Folosind notațiile anterioare, ecuația principală ce descrie codorul din Fig. 5.3 devine:

$$(\mathbf{S}_{t+1})_{m \times 1} = (\mathbf{H}_0)_{m \times r} \cdot (\mathbf{U}_t)_{r \times 1} + (\mathbf{T})_{m \times m} \cdot (\mathbf{S}_t)_{m \times 1}, \quad (5.13)$$

unde matricea \mathbf{T} este definită astfel:

$$\mathbf{T} = \begin{bmatrix} 0 & 1 & 0 & \dots & 0 & 0 \\ 0 & 0 & 1 & \dots & 0 & 0 \\ \dots & \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & 0 & \dots & 0 & 1 \\ h_{0,m} & h_{0,m-1} & h_{0,m-2} & \dots & h_{0,2} & h_{0,1} \end{bmatrix} = \begin{bmatrix} \mathbf{0}_{m-1 \times 1} & \mathbf{I}_{m-1} \\ & \mathbf{H}_R \end{bmatrix}. \quad (5.14)$$

Ieșirea redundantă este egală cu:

$$\mathbf{c}^t = \mathbf{H}_{out} \cdot \mathbf{S}_t + \mathbf{W} \cdot \mathbf{S}_{t+1}, \quad (5.15)$$

unde vectorul $\mathbf{w} = [0 \ 0 \ \dots \ 0 \ 1]_{1 \times m}$.

Din cele menționate mai sus rezultă că interleaver-ele pentru TCMB-uri trebuie să fie capabile să opereze asupra tablourilor de biți și nu asupra vectorilor. În continuare se prezintă un studiu asupra problematicii construirii unor astfel de interleaver-e și asupra performanțelor BER obținute cu cele propuse.

Construcția interleaver-elor pentru TCMB-uri, analizată în cele ce urmează, s-a făcut pornind de la tipurile de interleaver-e deja cunoscute pentru TC-uri unibinare, care au fost prezentate succint în paragraful 3.4.

5.1 Interleaver-e pentru turbo coduri multi-binare

În construcția interleaver-elor pentru turbo codurile multi-binare, pe care am realizat-o în referința [KOB05], am avut în vedere configurația matricială a datelor unui bloc (ce trebuie întrețesut). În plus, funcție de modul de închidere a trellis-urilor codurilor componente ale turbo-codului (neînchis, doar unul închis, ambele închise) și funcție de modul de concatenare (paralel sau serial), blocul ce trebuie întrețesut poate să conțină doar informația, cu sau fără eventualii biți de închidere ai trellis-ului, sau poate să includă și biții de control rezultați dintr-o primă codare.

La fel ca și pentru TC-urile unibinare și pentru TCMB-uri se pot adopta diferite soluții de închidere a trellis-urilor. Modul de închidere al trellis-urilor TCMB-urilor alături de numărul de intrări per codor, de memoria codorului precum și de numărul de biți per bloc turbo codat sunt factorii de care depinde forma interleaver-ului. În Fig. 5.4 se prezintă structurile posibile ale blocurilor turbo codate. Semnificația notațiilor din Fig. 5.4 este:

- r , reprezintă numărul de intrări în codor;
- N , reprezintă lungimea unui bloc codat (de un singur codor multi-component);
- M , este memoria codoarelor componente;
- INF, sunt biții de informație;
- RIT, sunt biți redundanți pentru închiderea trellis-ului;
- C, cod component;
- COD, sunt biți rezultați prin codare, biți de control.

Întrețeserea se face fie exclusiv asupra blocului INF (cazurile a) sau c)), fie asupra blocurilor INF+RIT (discuția se limitează asupra concatenării paralele, corespunzătoare unui turbo cod pur, la concatenarea serială întrețeserea operează și asupra biților COD1). Așadar, blocul ce trebuie întrețesut are o structură matricială cu dimensiunile $r \times N$ sau $r \times (N-M)$. Această structură matricială impune întrețeserii o cerință în plus față de cazul TC-urilor unibinare. Anume, este de dorit ca biții ce s-au aflat pe aceeași coloană înainte de întrețesere să nu se regăsească pe o aceeași coloană și după întrețesere (împrăștierea coloanei). În cele ce urmează se consideră că blocul pentru întrețesere are dimensiunea $r \times N$.

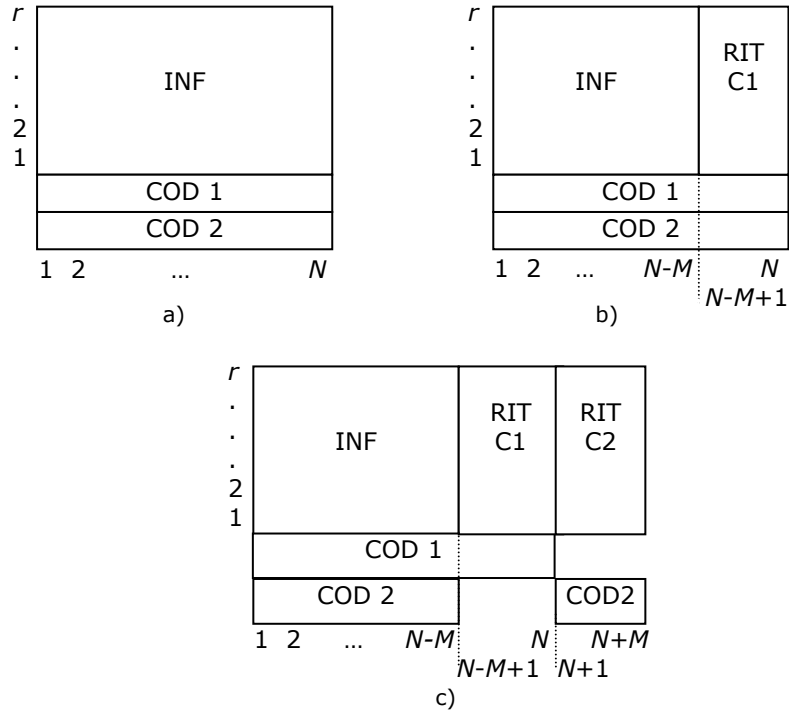


Fig. 5.4. Structura blocului turbo codat funcție de modul de închidere al trellis-urilor TCMB-ului: a) trellis-uri neînchise sau circulare, b) trellis C1 închis, trellis C2 neînchis și c) trellis-uri închise.

Întreșerea unui astfel de bloc poate fi gândită în trei strategii distincte:

A. Întreșerea blocului $r \times N$ transformat linie $1 \times rN$

După transformarea blocului de date din forma matricială, Fig. 5.4 a), în forma vectorială, Fig. 5.5, sub impunerea ca lungimea interleaver-ului să fie multiplu de r , se pot aplica oricare din metodele clasice. Dezavantajul metodei constă în nerealizarea împrăștierii coloanei. Pentru a satisface această cerință, transformarea matricei în coloană trebuie făcută prin multiplexarea liniilor, sau altfel spus, printr-o prealabilă întreșere coloană-linie.

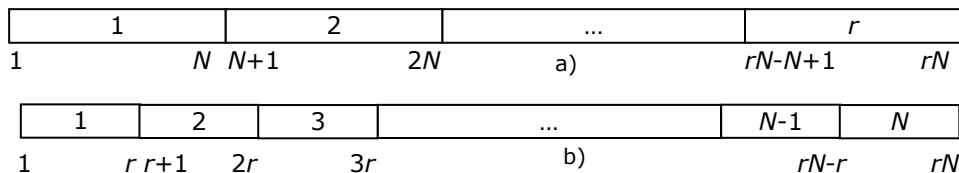


Fig. 5.5. Blocul de date transformat linie (vector): a) concatenarea liniilor, b) multiplexarea liniilor.

B. Întreșeserea blocului $r \times N$ prin metode specifice matricei

Prin „metode specifice” matricei se înțeleg acele funcții de întreșesere ce implică doi indici (i, j). Altfel spus, pozițiile înainte și după întreșesere depind de doi indici. Un exemplu tipic este întreșeserea bloc sau rectangulară, dată de relația (3.9).

Aplicarea directă, însă, a întreșeserii rectangulare asupra unui bloc de dimensiune $r \times N$, r de obicei mic, va conduce la o distanță minimă de întreșesere de valoare $r+1$. Pentru a elimina acest neajuns se propune o operație de ciclare diferențiată a blocului de date, înaintea întreșeserii, operație sugerată în Fig. 5.6. Ciclarea diferențiată constă într-o operație de permutare ciclică a liniilor, cu pași diferiți de la linie la linie.

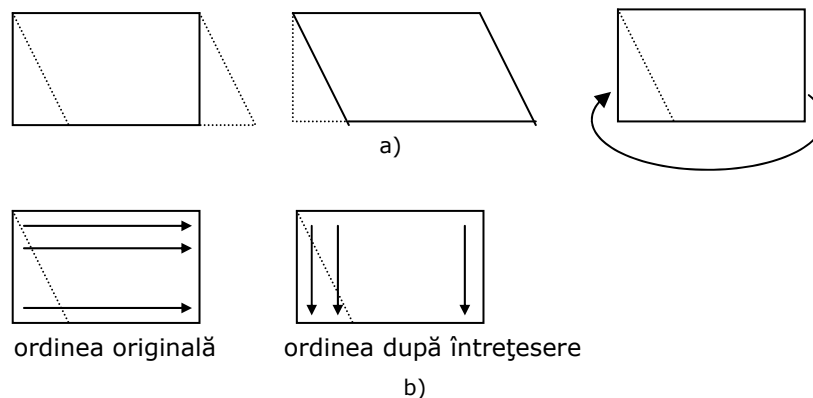


Fig. 5.6. Întreșeserea bloc cu ciclare diferențiată: a) ciclare diferențiată, b) întreșeserea bloc (rectangulară).

C. Întreșeserea independentă a liniilor blocului

Întreșeserea independentă a liniilor blocului se poate face cu oricare din metodele prezentate mai sus. Dacă se utilizează exact aceeași funcție interleaver pentru fiecare linie, întreșeserea se face de fapt asupra simbolurilor (nu se împrăștie coloana). Avantajul este utilizarea unui interleaver cu lungimea N/r . Pentru a realiza împrăștieră coloanei se poate apela de asemenea la o ciclare diferențiată prealabilă.

Pentru simulări am utilizat un TCMB de memorie 3. Codul component recursiv și sistematic are matricea generatoare $G=[1, 1, 13/15, 11/15]$, așa cum am specificat în [KOB05, BAK05a]. S-a utilizat algoritmul MAP, 15 iterații cu un criteriu de oprire a iterațiilor pe baza LLR-ului, [BDK06]. Canalul s-a presupus AWGN iar modulația utilizată a fost BPSK. Rezultatele obținute le-am prezentat în diagramele din Fig. 5.7, utilizând interleaver-ele descrise în Tabelul 5.1.

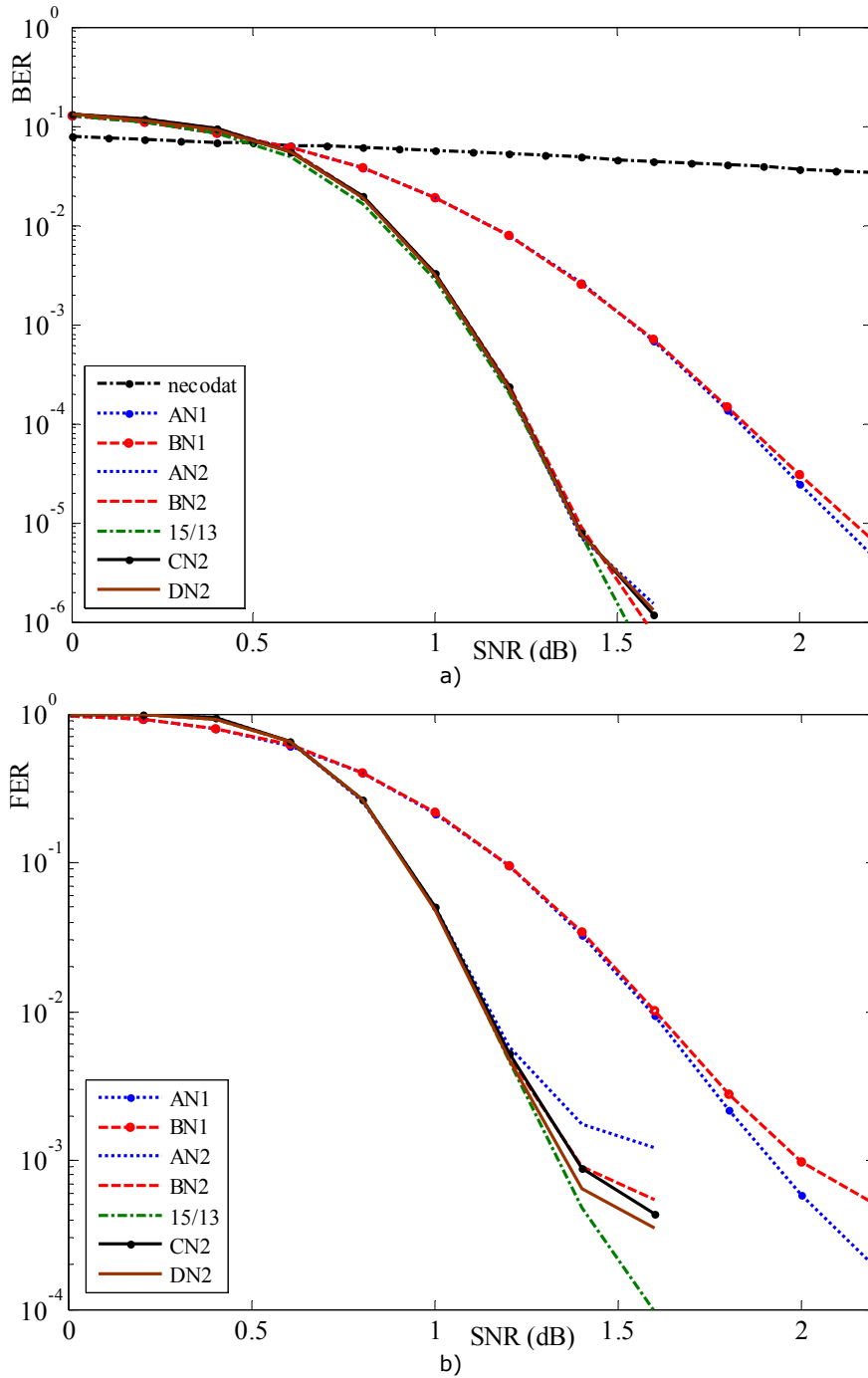


Fig. 5.7 Performanțele: a) BER și b) FER, ale TCMB-urilor, de rată 1/2 pentru codorul 13/11/15, folosind interleaver-ele menționate în Tabelul 5.1 și al TC-ului (puncturat), de rată 1/2, pentru codorul 15/13.

Tabelul 5.1 Interleaver-ele utilizate.

Interleaver	Lungimea blocului	Codul component	Metode de întrețesere
AN1	448	13 11 15	C (ciclare diferențiată) interleaver S de 224 biți
BN1	448	13 11 15	A (multiplexare linii) interleaver S de 448 biți
AN2	1792	13 11 15	C (ciclare diferențiată) interleaver S de 896 biți
BN2	1784	13 11 15	A (multiplexare linii) interleaver S de 1784 biți
CN2	1784	13 11 15	A (concatenareare linii) interleaver S de 1784 biți
DN2	1784	13 11 15	A (concatenareare linii) interleaver CCSDS
1513	1784	15 13 clasic puncturat	Interleaver S de 1784 biți

Dintre cele trei metode de întrețesere descrise anterior pentru două am construit interleaver-e, ale căror performanțe (în asociere cu un TCMB de memorie 3) sunt prezentate în Fig. 5.7 și comparate cu cel mai performant TC clasic puncturat de aceeași rată de codare, 15/13. Diagramele din Fig. 5.7 arată că performanțe la BER sunt foarte apropiate de cele ale TC-ului clasic. Diferența este, însă, esențială în performanța FER. Cel mai performant interleaver, din punct de vedere al BER este BN2 cel obținut prin metoda A, cu multiplexare a liniilor.

În paragraful următor voi face o analiză a performanțelor BER, respectiv, FER, ale TC-urilor puncturate și ale TCMB-urile puncturate, în canale AWGN.

5.2 Performanțele turbo codurilor și a turbo codurilor multi-binare puncturate

În analiza performanțelor se consideră TC-urile inițiale ce sunt constituite din concatenarea în paralel a două codoare RSC, de memorie 4 ($m=4$) și de rată 1/2 [CRO00, GCH99] și TCMB-urile ce sunt constituite din concatenarea în paralel a două codoare RSC de memorie 4 și de rată 2/3, [DOB05]. Pentru a obține o eficiență spectrală mai mare, se iau în considerare rate de codare de 1/3, 2/3 și 3/5 pentru ambele TC-uri și TCMB-uri, prin utilizarea operațiilor de puncturare, [KBDN07].

Particularizând forma canonică a codorului multi-binar din Fig. 5.3, rezultă schemele codoarelor RSC pentru TC-uri și pentru TCMB-uri, ce au polinomul de reacție egal cu $(23)_8$ și polinomul de redundanță egal cu $(35)_8$ și matricile generatoare $\mathbf{H}=[11\ 1\ 12]_{10}$, respectiv $\mathbf{H}=[11\ 11\ 1\ 12]_{10}$ (modul în care este scrisă această matrice, \mathbf{H} , va fi prezentat mai detaliat în paragraful următor), așa cum am prezentat în [KDBN06]. Aceste scheme sunt reprezentate în Fig. 5.8, unde cu S_m , cu $m=1\div 4$, am notat starea codorului, ce pot fi realizat cu 4 registre de deplasare. Cu u , u_1 și u_2 am notat secvențele de biți de la intrarea în codor, iar cu c am notat secvența biților redundanți, de la ieșirea din codor. Diagramele de stări ale codorului RSC ce intră în componența TC-urilor și a TCMB-urilor conțin 2^m noduri, cu $m=4$ la fiecare durată de timp, ceea ce conduce la $2^m=16$ stări posibile; 2^r ramuri sunt conectate la fiecare nod ce corespunde la 2^r vectori de intrare posibili ce verifică ecuațiile codorului. Astfel, diagrama conține 2^{m+r} ramuri sau tranziții. Fiecare tranziție este asociată la o durată a simbolului. La o durată de timp dată, sunt considerați r biți de intrare, cu scopul de a genera $r+1$ biți la ieșirea decodorului. În cazul TC-urilor, există doar o singură intrare binară, $r=1$, așa cum este reprezentată în Fig. 5.8 a), iar în cazul TCMB-urilor s-au considerat 2 biți de intrare, $r=2$, așa cum este reprezentat în Fig. 5.8 b).

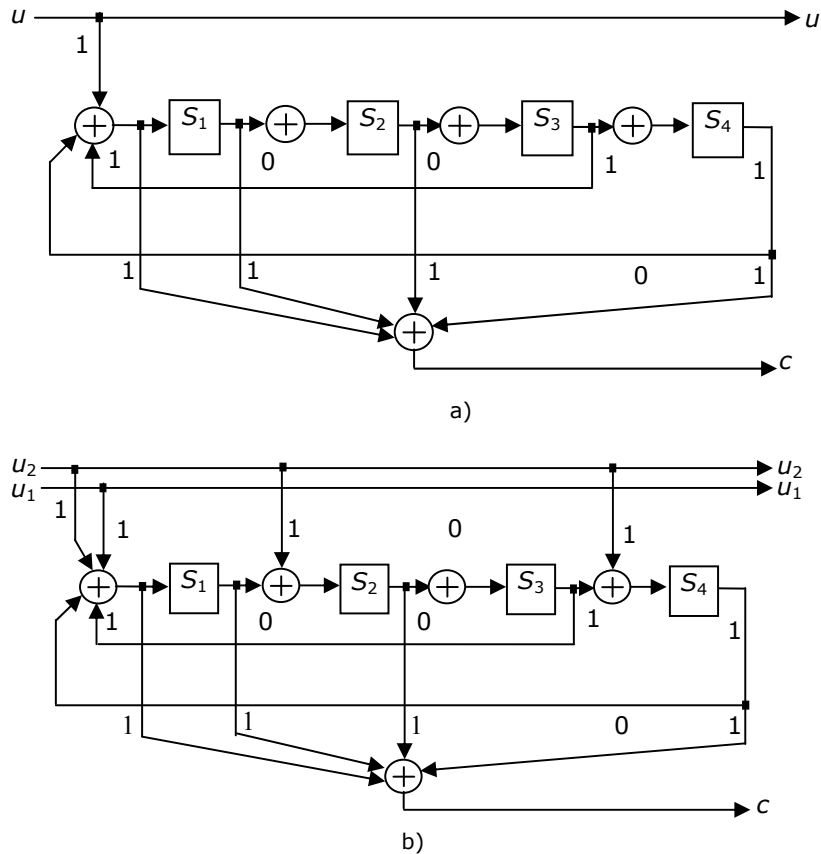


Fig. 5.8 a) Codor component RSC pentru TC; b) Codor component RSC pentru TCMB. Ambele codoare au polinomul de reacție $= (23)_8 = (10011)_2$ și polinomul de redundanță $= (35)_8 = (11101)_2$.

În Fig. 5.9 am prezentat codorul TC și codorul TCMB, ce sunt constituite din conectarea în paralel a câte două codoare RSC cu 16 stări, și de rată $r/r+1$, prezentate deja în figura anterioară. Trebuie precizat faptul că biții de informație sunt întrețesuți înainte de a intra în cel de al doilea decodor. În toate simulările din acest paragraf s-a considerat ca întrețeserea să fie realizată de un interleaver S propus de [CRO00]. Interleaver-ele S sunt interleaver-e semi-aleatoare, ce prezintă performanțe excelente deoarece au distanțe minime foarte mari chiar și pentru blocuri de date de lungimi moderate. De exemplu pentru blocuri de lungimi egale cu 752 și 1504 biți, interleaver-ele S prezentate în [DOD95, KBN05] produc distanțe minime egale cu 19, respectiv 27.

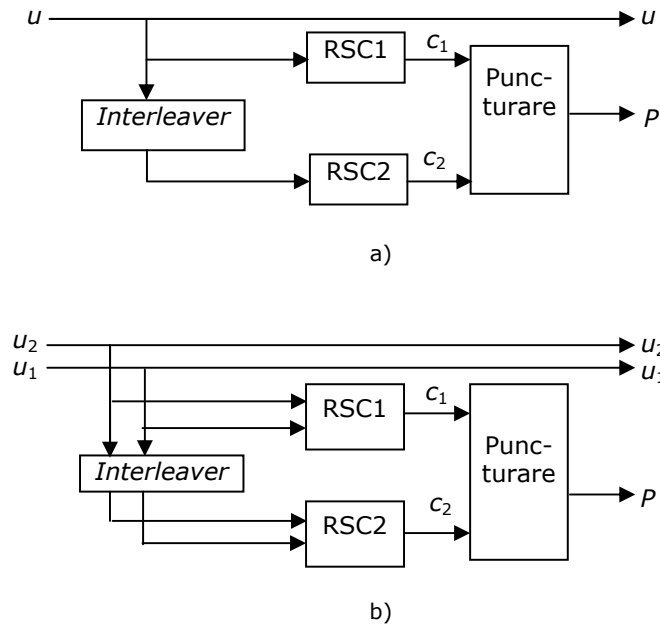


Fig. 5.9 a) Turbo codor cu codoare RSC de rată 1/2; b) Turbo codor multi-binar cu codoare RSC de rată 2/3.

Aceste distanțe minime pot fi crescute așa cum s-a văzut în [CRO00]. Acest fapt se bazează pe o selecție aleatoare cu următoarea constrângere:

$$d(i, j) = |\pi(i) - \pi(j)| + |i - j| \geq S, \quad (5.16)$$

unde π reprezintă funcția de permutare complet aleatoare și $d(i, j)$ reprezintă distanța de întretesere dintre pozițiile i și j , cu $i, j=1, \dots, N$ (în care N reprezintă lungimea cuvântului de cod). Așadar, considerându-se această metodă, interleaver-elor construite li s-au ales distanțele minime egale cu 22, pentru blocuri de 752 biți, respectiv 31, pentru blocuri de 1504 biți.

Ratele de codare ale TC-urilor și ale TCMB-urilor sunt egale cu $r/(r+2)$, adică în cazul de față sunt egale cu 1/3, respectiv 1/2. Cu ajutorul blocului de puncturare se pot obține rate de codare mai mari, prin puncturarea, de exemplu, a biților de paritate c_1 și c_2 . Biții de informație nu sunt puncturați. În Tabelul 5.2 am prezentat matrici optime de puncturare, P , pentru TC-uri și TCMB-uri cu rate de cod $R=1/2, 3/5$ și $2/3$, [TOR02]. Din tabel se remarcă faptul că pentru a obține rate de codare mai mari decât $r/(r+2)$ în cazul TCMB-urilor doar câteva simboluri redundante trebuie să fie ignorate, comparativ cu TC-urile clasice. În consecință, abilitatea de corecție a codului constituent va fi mai puțin degradată.

Tabelul 5.2 Matrici de puncturare optime pentru rate 1/2, 3/5 și 2/3.

Coduri corectoare	Rate de codare		
	$R=1/2$	$R=2/3$	$R=3/5$
TC	$P = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$	$P = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}$	$P = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}$
MBTC	$P = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$	$P = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$	$P = \begin{bmatrix} 1 & 1 & 0 \\ 0 & 1 & 1 \end{bmatrix}$

Pentru a obține rezultatele experimentale prezentate în continuare am mai considerat următoarele situații. Atât în cazul TC-urilor cât și TCMB-urilor, trellis-ul primului codor a fost închis la 0, iar trellis-ul celui de al doilea a rămas neînchis. Lungimea secvențelor codate este egală cu 188 byte=2·752 biți. Am considerat canalul AWGN, iar ca și algoritm de decodare am folosit algoritmul Max-Log.MAP, [VOF00, DOB05], datorită complexității sale reduse de calcul și păstrării performanțelor aproape de cele optimale. Factorul de scalare a informației extrinseci este egal cu 0.75 [KBDN07], iar numărul de iterații de la decodor a fost ales egal cu 15.

În Fig. 5.10÷5.15 sunt arătate performanțele BER și FER ale TCMB-urilor și ale TC-urilor, la diferite rate de codare (1/2, 2/3 și 3/5).

În Fig. 5.10, pentru $BER < 10^{-2}$, TCMB-urile au un câștig de aproximativ 0.1 dB comparativ cu TC-urile. Un câștig similar se poate observa la $FER < 10^{-4}$, așa cum rezultă din Fig. 5.11. Pentru un $FER = 5 \cdot 10^{-5}$, TCMB-urile se apropie la 0.55 dB de limita Shannon. Pentru $BER \geq 10^{-6}$, performanțele BER ale ambelor coduri de rată 2/3 sunt similare. Sub această valoare TCMB-urile au ușoară îmbunătățire a performanțelor față de TC (Fig. 5.12). Interesant este faptul că în ceea ce privește performanțele FER câștigul este mai mare, astfel pentru $FER < 10^{-4}$, este în jur de 0.1 dB, în cazul utilizării TCMB-urilor. De exemplu, pentru $FER = 5 \cdot 10^{-5}$, acest câștig este egal cu 0.15 dB, așa cum rezultă din Fig. 5.13, și se apropie la 0.5 dB de limita Shannon. O altă trăsătură importantă ce poate fi observată în Fig. 5.13 este aceea că *error floor* pare să fie mai puțin importantă în cazul TCMB-urilor decât în cazul TC-urilor.

În Fig. 5.14 sunt comparate performanțele BER dintre cele două tipuri de coduri corectoare, ambele având rata egală cu 3/5. Ambele coduri au performanțe similare pentru BER mai mare decât 10^{-4} . Pentru valori mai mici ale lui BER, MBTC-urile au performanțe ușor îmbunătățite în comparație cu TC-urile. Pentru $FER < 10^{-4}$, se obține un câștig egal sau mai mare decât 0.1 dB. De exemplu, pentru $FER = 5 \cdot 10^{-5}$, acest câștig este egal cu 0.11 dB (Fig. 5.15), și se apropie la 0.55 dB de limita Shannon. Ca și în cazul ratei de codare de 2/3, se pare că *error floor* este mai puțin importantă în cazul TCMB-urilor decât în cazul TC-urilor.

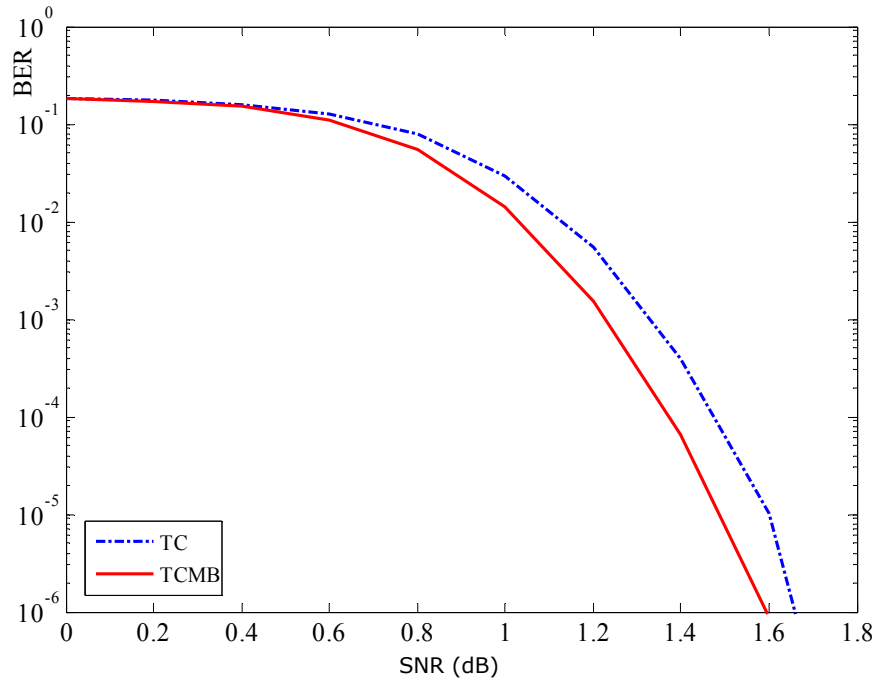


Fig. 5.10 Performanța BER la rata 1/2 pentru TC, $\mathbf{H}=[11\ 1\ 12]$, și TCMB, $\mathbf{H}=[11\ 11\ 1\ 12]$, în cazul unei transmisii pe canal AWGN, funcție de SNR (dB).

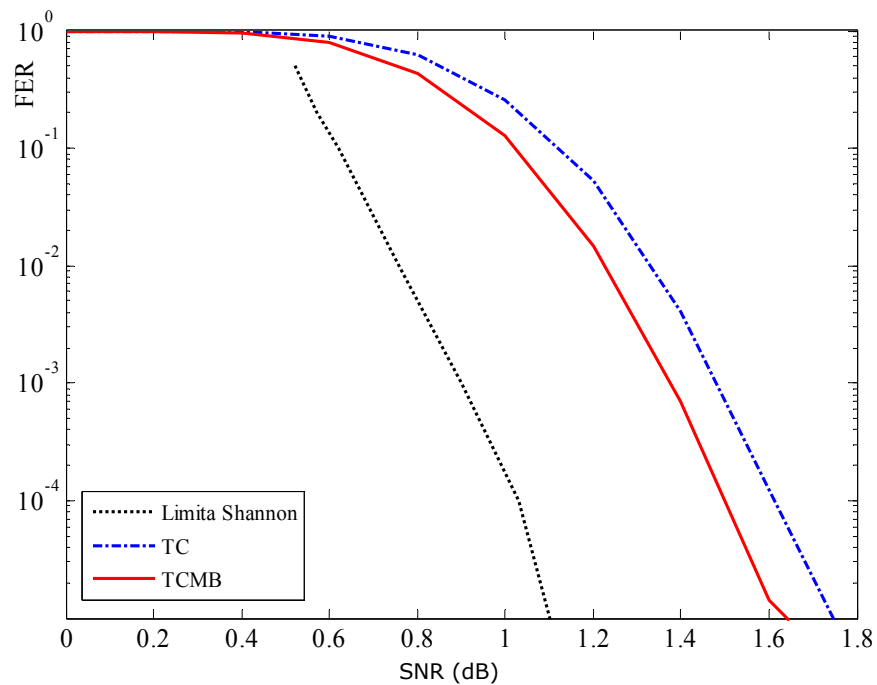


Fig. 5.11 Performanța FER la rata 1/2 pentru TC, $\mathbf{H}=[11\ 1\ 12]$, și TCMB, $\mathbf{H}=[11\ 11\ 1\ 12]$, în cazul unei transmisii pe canal AWGN, funcție de SNR (dB).

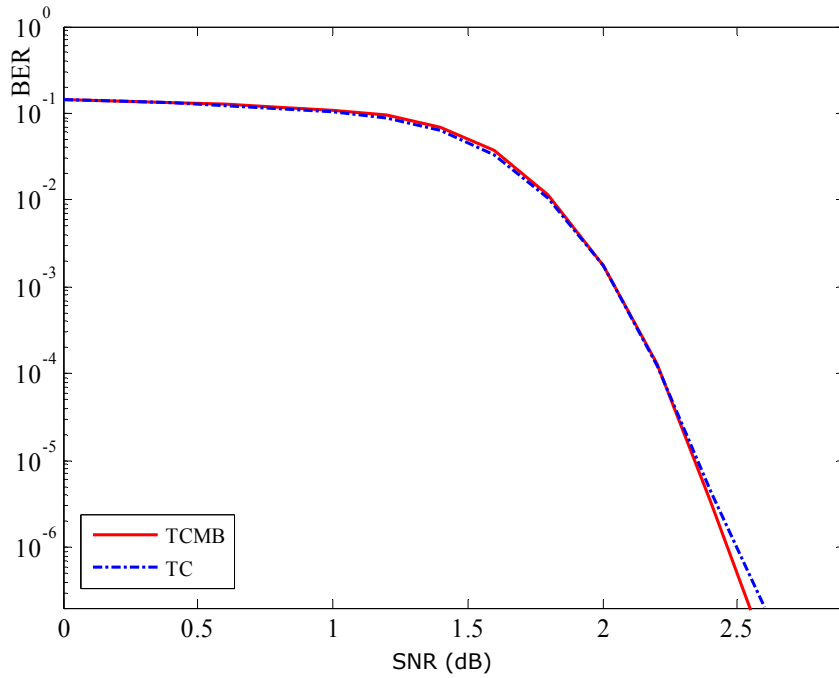


Fig. 5.12 Performanța BER la rata 2/3 pentru TC, $\mathbf{H}=[11\ 1\ 12]$, și TCMB, $\mathbf{H}=[11\ 11\ 1\ 12]$, în cazul unei transmisii pe canal AWGN, funcție de SNR (dB).

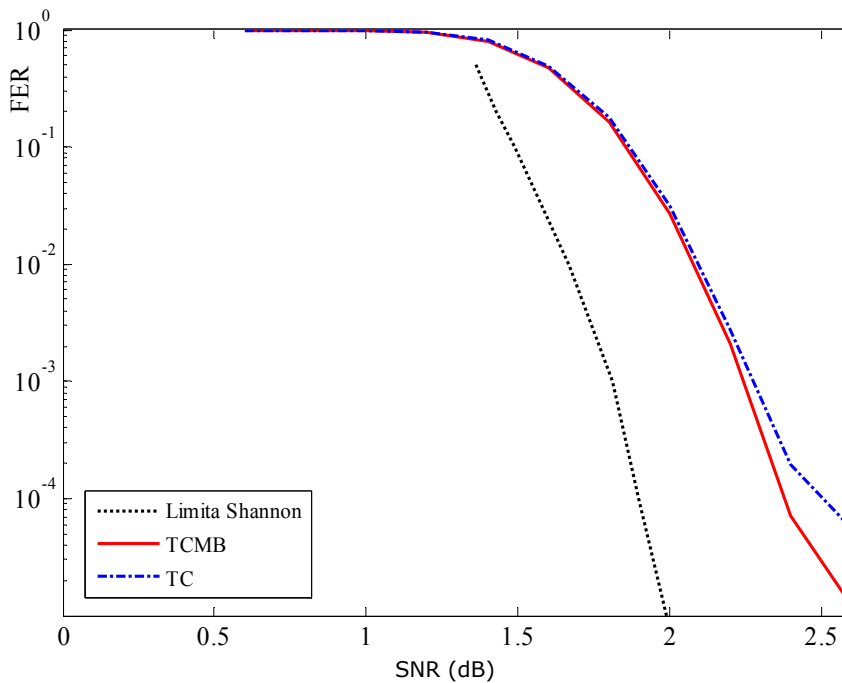


Fig. 5.13 Performanța FER la rata 2/3 pentru TC, $\mathbf{H}=[11\ 1\ 12]$, și TCMB, $\mathbf{H}=[11\ 11\ 1\ 12]$, în cazul unei transmisii pe canal AWGN, funcție de SNR (dB).

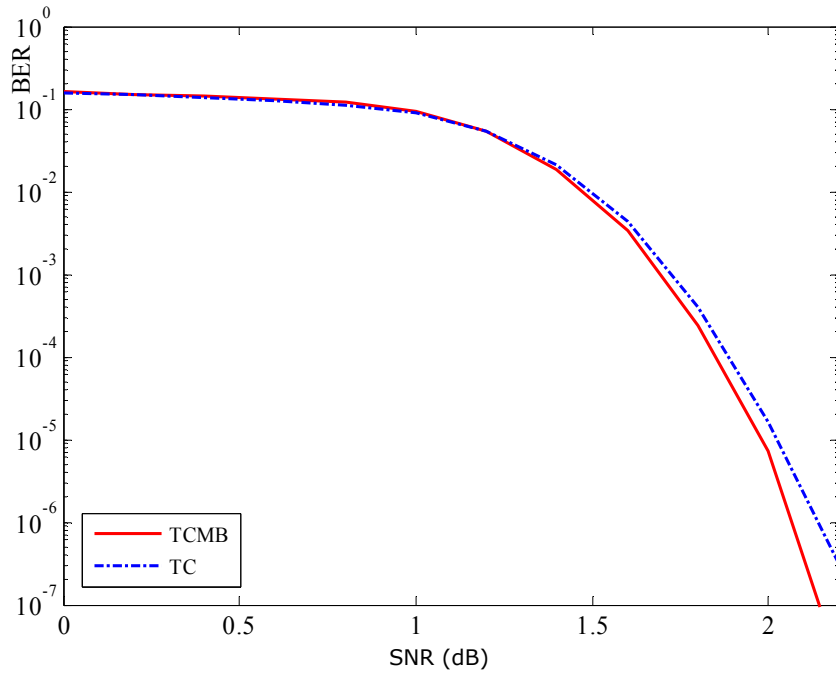


Fig. 5.14 Performanța BER la rata 3/5 pentru TC, $\mathbf{H}=[11\ 1\ 12]$, și TCMB, $\mathbf{H}=[11\ 11\ 1\ 12]$, în cazul unei transmisii pe canal AWGN, funcție de SNR (dB).

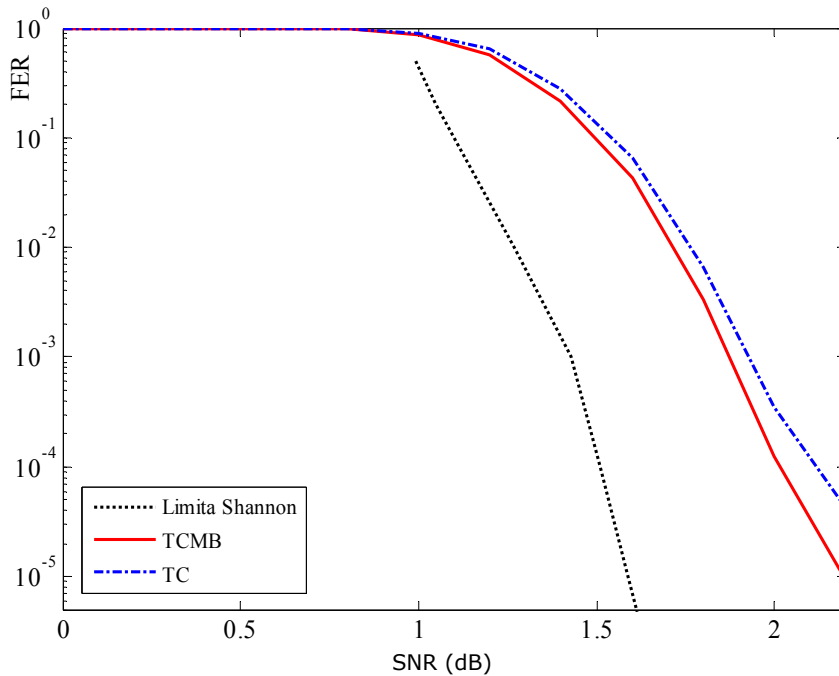


Fig. 5.15 Performanța FER la rata 3/5 pentru TC, $\mathbf{H}=[11\ 1\ 12]$, și TCMB, $\mathbf{H}=[11\ 11\ 1\ 12]$, în cazul unei transmisii pe canal AWGN, funcție de SNR (dB).

Câștigul de aproximativ 0.2 dB pe care l-am obținut comparativ cu [DOB05] s-a datorat faptului că am folosit un număr de 15 iterații în loc de 8, cum s-au folosit în [DOB05] și datorită faptului că am folosit un alt interleaver decât cel specificat în referința amintită anterior, și anume interleaver-ul S .

În continuare analizez performanțele BER și FER ale TCMB-urilor în cazul unei transmisiuni în canale cu fading plat.

5.3 Performanțele turbo codurilor multi-binare în canalele cu fading plat

5.3.1 Fading plat de tip Nakagami

TCMB-urile propuse recent de către C. Douillard și C. Berrou [DOB05], depășesc performanțele TC-urilor clasice, inventate de către C. Berrou în 1993 [BGT93]. Astfel în [DOB05] s-a arătat faptul că, prin concatenarea în paralel a codurilor RSC binare cu r intrări rezultă câteva avantaje față de TC-uri, pe canale AWGN, în special acela că au un efect de *error floor* mai mic.

În continuare analizez performanțele BER și FER ale TCMB-urilor pe canale cu fading ne-selectiv în frecvență. TCMB-urile propuse sunt alcătuite din două coduri RSC duo-binare de rată $2/3$ conectate în paralel. Deasemenea, consider modelul canalului cu distribuție Nakagami- m (prezentată în paragraful 4.5), deoarece acesta se potrivește bine la canalele cu fading ale sistemelor de transmisie fără fir.

Pe baza schemei generale a unui codor convoluțional cu intrări multiple din Fig. 5.3, în Fig. 5.16 prezint schema unui codor duo-binar cu 16 stări, având matricea generatoare $\mathbf{H}=[11\ 11\ 1\ 12]_{10}$, schemă ce este prezentată și în Fig. 5.8 b), unde elementele matricii \mathbf{H} sunt înlocuite cu valorile prezentate în ecuațiile următoare.

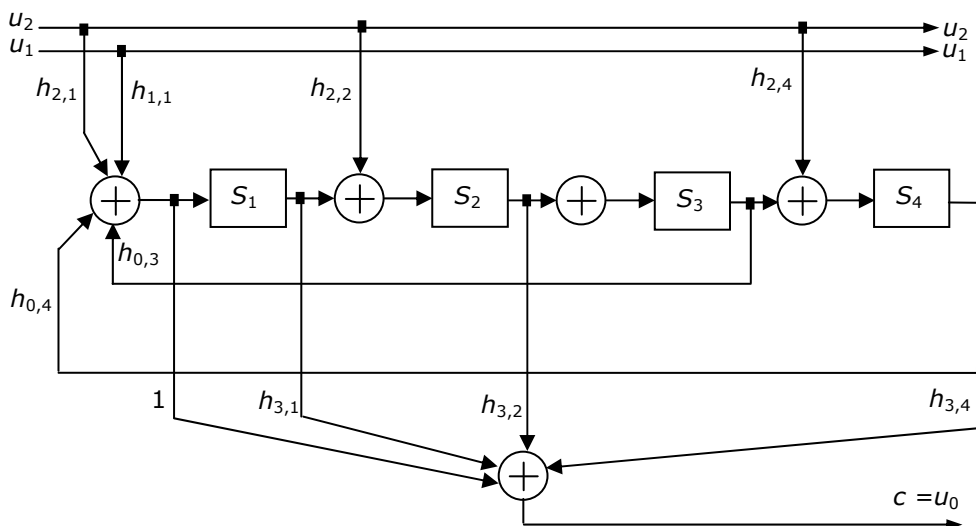


Fig. 5.16 Schema unui codor duo-binar cu 16 stări, cu $\mathbf{H}=[11\ 11\ 1\ 12]_{10}$.

Pentru codul particular reprezentat în Fig. 5.16, vectorii \mathbf{S}_t și \mathbf{U}_t definiți la începutul capitolului 5, sunt egali cu:

$$\mathbf{S}_t = [s_4^t \ s_3^t \ s_2^t \ s_1^t]^T \text{ și } \mathbf{U}_t = [u_2^t \ u_1^t]^T, \quad (5.17)$$

iar relația (5.10) devine, exprimând matricea \mathbf{H} într-o formă compactă, unde fiecare coloană din \mathbf{H} este reprezentată de o valoare zecimală, corespunzătoare vectorului său coloană scris în binar:

$$\begin{aligned} \mathbf{H} &= \begin{bmatrix} h_{r+1,m} & h_{r,m} & \dots & h_{1,m} & h_{0,m} \\ \dots & \dots & \dots & \dots & \dots \\ h_{r+1,2} & h_{r,2} & \dots & h_{1,2} & h_{0,2} \\ h_{r+1,1} & h_{r,1} & \dots & h_{1,1} & h_{0,1} \end{bmatrix} = \\ &= \begin{bmatrix} h_{3,4} & h_{2,4} & h_{1,4} & h_{0,4} \\ h_{3,3} & h_{2,3} & h_{1,3} & h_{0,3} \\ h_{3,2} & h_{2,2} & h_{1,2} & h_{0,2} \\ h_{3,1} & h_{2,1} & h_{1,1} & h_{0,1} \end{bmatrix} = \\ &= \begin{bmatrix} 1 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 \\ 1 & 1 & 0 & 0 \\ 1 & 1 & 1 & 0 \end{bmatrix} = [11 \ 11 \ 1 \ 12]_{10} \end{aligned} \quad (5.18)$$

Rezultă matricea generatoare restrânsă \mathbf{H}_0 și vectorii de reacție și de ieșire de forma:

$$\mathbf{H}_0 = \begin{bmatrix} h_{2,4} & h_{1,4} \\ h_{2,3} & h_{1,3} \\ h_{2,2} & h_{1,2} \\ h_{2,1} & h_{1,1} \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 0 \\ 1 & 0 \\ 1 & 1 \end{bmatrix}, \quad (5.19)$$

$$\mathbf{H}_R = [h_{0,4} \ h_{0,3} \ h_{0,2} \ h_{0,1}]^T = [1 \ 1 \ 0 \ 0]^T,$$

$$\mathbf{H}_{out} = [h_{3,4} \ h_{3,3} \ h_{3,2} \ h_{3,1}]^T = [1 \ 0 \ 1 \ 1]^T.$$

Relațiile (5.13), (5.14) și (5.15) pot fi rescrise sub forma:

$$\begin{aligned} (\mathbf{S}_{t+1})_{4 \times 1} &= (\mathbf{H}_0)_{4 \times 2} \cdot (\mathbf{U}_t)_{2 \times 1} + (\mathbf{T})_{4 \times 4} \cdot (\mathbf{S}_t)_{4 \times 1}, \\ \mathbf{T} &= \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 1 & 0 & 0 \end{bmatrix} \end{aligned} \quad (5.20)$$

$$\mathbf{c}^t = \mathbf{H}_{out} \cdot \mathbf{S}_t + \mathbf{W} \cdot \mathbf{S}_{t+1}$$

unde vectorul $\mathbf{W} = [0 \ 0 \ 0 \ 1]_{1 \times 4}$.

Interleaver-ul utilizat în construcția TCMB-urilor este cel propus în [DOB05]. Funcția sa de permutare $i=\pi(j)$, este realizată în doi pași. Pentru $j=1\div N-1$:

Pasul 1: Permutarea intra-simbol schimbă valorile $r_{j,1}$ și $r_{j,2}$, dacă $j \bmod 2=0$. Altfel, nu.

Pasul 2: Permutarea inter-simbol este dată de următoarea relație:

$$i = (P \times j + Q(j) + 3) \bmod N, \text{ cu:}$$

$$\begin{aligned} Q(j) &= 0 && \text{dacă } j \bmod 4 = 0 \\ Q(j) &= Q_1 && \text{dacă } j \bmod 4 = 1 \\ Q(j) &= 4Q_0 + Q_2 && \text{dacă } j \bmod 4 = 2 \\ Q(j) &= 4Q_0 + Q_3 && \text{dacă } j \bmod 4 = 3 \end{aligned} \quad (5.21)$$

cu: $P=35$, $Q_0=1$, $Q_1=4$, $Q_2=4$, $Q_3=12$.

Algoritmul de decodare pe care l-am utilizat în analiza performanțelor BER și FER este algoritmul Max Log MAP [VOF00]. În Fig. 5.17 este prezentată performanța FER a unui TC duo-binar cu 16 stări, funcție de SNR, atât în cazul utilizării algoritmului de decodare MAP cât și a algoritmului de decodare Max-Log-MAP.

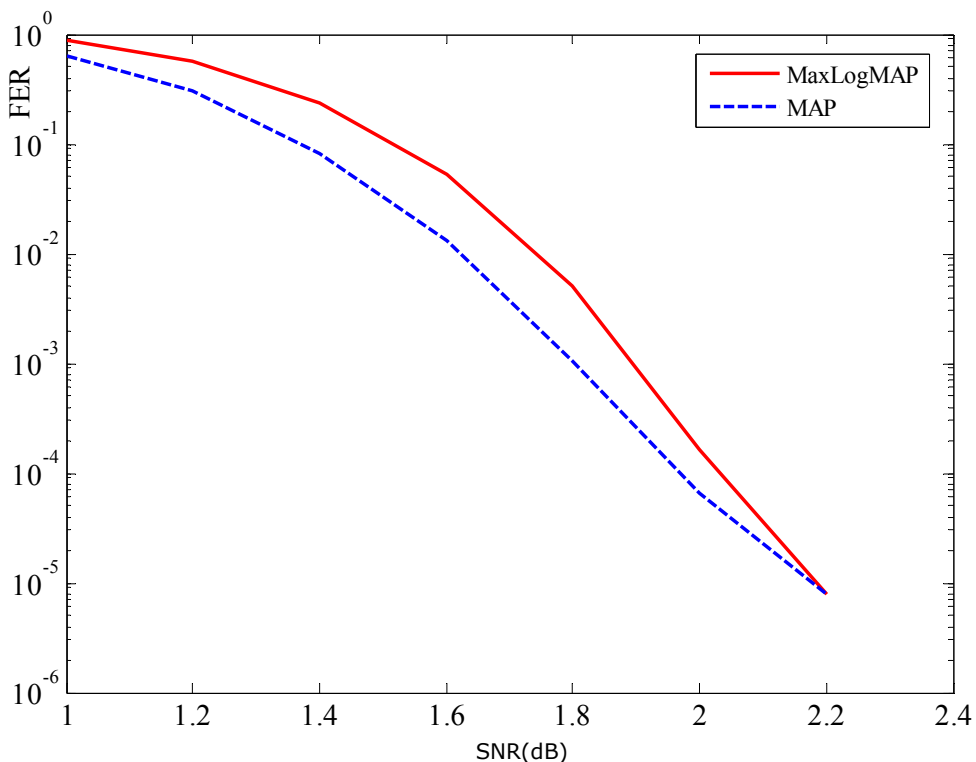


Fig. 5.17 O comparație între două implementări ale algoritmului de decodare la TCMB pentru un canal cu fading Nakagami-5: algoritmul MAP și aproximarea Max-Log-MAP.

Se consideră lungimea blocului de informație egală cu 188 byte. Se observă că la un SNR mic, pierderea nu depășește 0.17 dB. Interesant este faptul că, la un FER mic (mai mic decât 10^{-5}), pierderea devine neglijabilă.

Simulările din acest paragraf cât și cele din [KDBN06] au fost realizate considerând modulația QPSK, și numărul maxim de iterații egal cu 15. În cadrul simulărilor se compară performanțele TCMB-ului propus anterior cu limita Shannon.

În Fig. 5.18 se observă că TCMB-urile se comportă bine în canalele selective în timp, chiar și la lungimi moderate ale cuvântului de cod (lungimea cuvântului de cod este de 1504 de biți). Pierderea în performanțe de la $m=\infty$ (canal AWGN, fără fading) până la $m=1$ (fading Rayleigh) nu depășește 3 dB, pentru oricare FER și BER, în timp ce pierderea teoretică dată de limitele Shannon este aproximativ de 1.6 dB. Această pierdere moderată suplimentară se datorează faptului că se folosesc cuvinte de cod mici.

Se mai observă că, performanțele unui sistem de transmisie codat, cu un TCMB, ca și în cazul folosirii unui TC (paragraful 4.5), în canalele cu fading plat de tip Nakagami, cu $m > 1$, sunt mărginite superior de performanțele fading-ului plat de tip Rayleigh ($m=1$) și mărginite inferior de canalele fără fading ($m=\infty$).

În [DOB05] se arată, prin intermediul simulărilor, că TCMB-urile depășesc în performanță TC-urile clasice în cazul unui canal AWGN. Am făcut o comparație similară (Fig. 5.19), dar luând în considerare, de data aceasta, canalul Nakagami.

Pentru simulările ale căror rezultate sunt prezentate în Fig. 5.19, am considerat că pentru toate TC-urile lungimea blocului de date N este de 1504 biți, rata de codare $R=1/2$ și parametrul fading-ului $m=5$. Datorită criteriului de oprire al iterațiilor, numărul mediu de iterații este aproximativ de 3 dintr-un număr maxim de 15 iterații. Pentru $10^{-5} < FER < 5 \cdot 10^{-3}$ performanțele TC-urilor și a TCMB-urilor sunt similare. Interesant este că pentru valori foarte mici ale FER (mai mici decât 10^{-5}), TCMB-urile nu manifestă efectul de *error floor* datorită distanțelor lor minime mari, în comparație cu TC-urile clasice. Această caracteristică este interesantă în special în sistemele viitoare fără fir cu rate de date mari, unde este necesară o calitate mare a serviciilor.

Într-o abordare viitoare va fi interesant să evaluez impactul erorii de estimare a canalului asupra performanțelor.

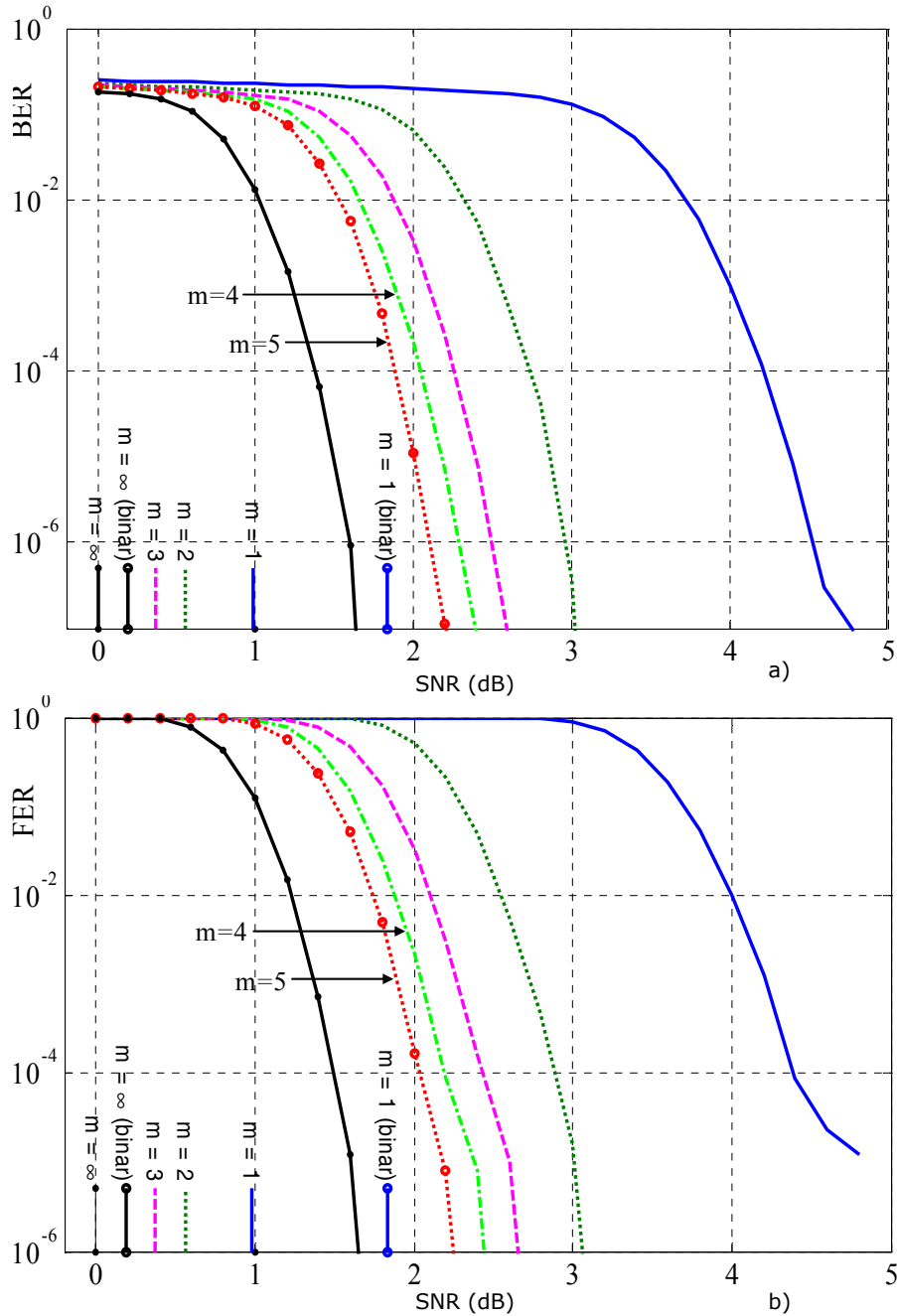


Fig. 5.18 Performanțele TCMB-ului în cazul transmisiei pe un canal cu fading plat de tip Nakagami, utilizând algoritmul de decodare Max-Log-MAP: a) BER, b) FER, sunt trasate în funcție de SNR pentru diferite valori ale parametrului m . Deasemenea, pentru câteva dintre aceste valori ale parametrului m , sunt reprezentate pragurile corespunzătoare limitei Shannon pe canale cu fading plat de tip Nakagami (pentru intrări gaussiene, cât și pentru intrări binare ($m=1$ și $m=\infty$)).

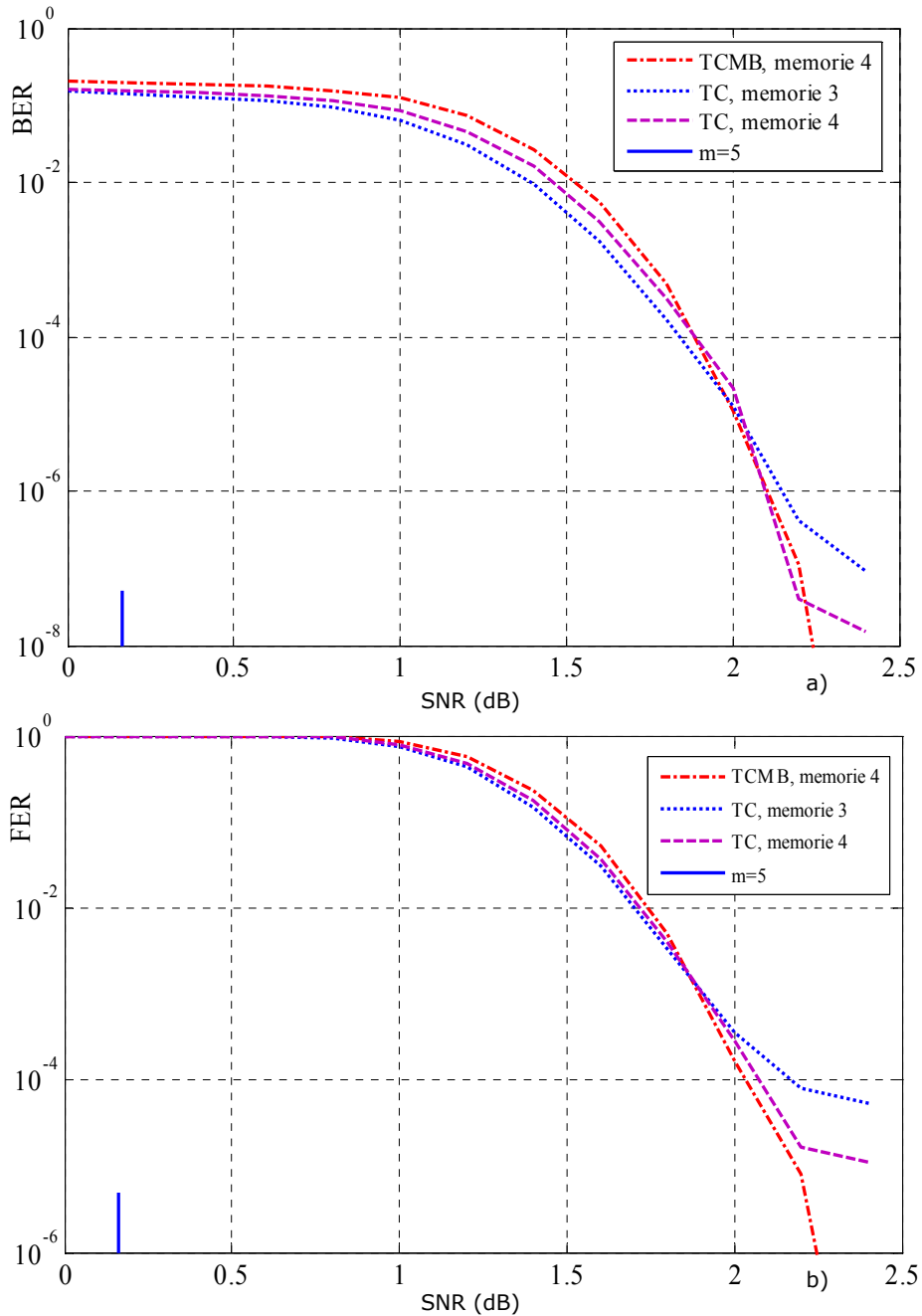


Fig. 5.19 Performanțele a) BER și b) FER pentru TCMB-urile de memorie 4 și pentru TC-urile clasice de memorie 3 (13/15) și de memorie 4 (23/31), pe canale cu fading plat de tip Nakagami (parametrul fading-ului $m=5$), utilizând algoritmul de decodare MAP. De asemenea este reprezentat pragul corespunzător limitei Shannon cu intrări gaussiene pe canale cu fading plat de tip Nakagami.

5.3.2 Fading plat de tip Rayleigh

Pentru transmisiunile *wireless* în canalele cu fading, codarea canalului este un instrument important în îmbunătățirea fiabilității. Mulți cercetători au studiat performanțele diferitelor coduri, cum ar fi: codurile bloc [BAB04], turbo codurile [HAW98, YUS02, CCD03], în canale cu fading de tip Rayleigh. S-a arătat că TCMB-urile prezentate în [DOB05], au performanțe apropiate de limita capacității canalului, în canale AWGN.

În acest paragraf cât și în [KBN06] am analizat performanțele BER și FER ale unui sistem turbo codat multi-binare în canalul cu fading plat de tip Rayleigh, al cărui model l-am prezentat în capitolul 4.

Simulările au fost realizate pentru TCMB-uri, de rată 1/2, care au ca și coduri constitutive codurile convoluționale RSC, de memorie 3 cu matricea generatoare $\mathbf{H}=[6 \ 7 \ 1 \ 5]_{10}$, și de memorie 4 cu matricea generatoare $\mathbf{H}=[11 \ 11 \ 1 \ 12]_{10}$. Schema unui codor duo-binare cu 16 stări, cu $\mathbf{H}=[11 \ 11 \ 1 \ 12]_{10}$ este prezentată în Fig. 5.16. În continuare, voi prezenta schema unui codor duo-binare cu 8 stări, cu $\mathbf{H}=[6 \ 7 \ 1 \ 5]_{10}$, pe baza schemei generale a unui codor multi-intrare din Fig. 5.3.

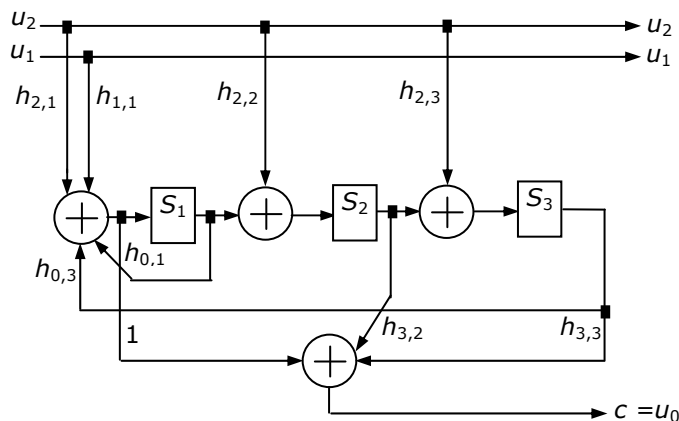


Fig. 5.20 Schema unui codor duo-binare cu 8 stări, cu $\mathbf{H}=[6 \ 7 \ 1 \ 5]_{10}$.

Pe baza Fig. 5.20, vectorul stare și vectorul intrare, la momentul t , sunt $\mathbf{S}_t = [s_4^t \ s_3^t \ s_2^t \ s_1^t]^T$ și $\mathbf{U}_t = [u_2^t \ u_1^t]^T$. Ecuțiile (5.10)÷(5.15) devin:

$$\begin{aligned}
\mathbf{H} &= \begin{bmatrix} h_{r+1,m} & h_{r,m} & \dots & h_{1,m} & h_{0,m} \\ \dots & \dots & \dots & \dots & \dots \\ h_{r+1,2} & h_{r,2} & \dots & h_{1,2} & h_{0,2} \\ h_{r+1,1} & h_{r,1} & \dots & h_{1,1} & h_{0,1} \end{bmatrix} = \\
&= \begin{bmatrix} h_{3,3} & h_{2,3} & h_{1,3} & h_{0,3} \\ h_{3,2} & h_{2,2} & h_{1,2} & h_{0,2} \\ h_{3,1} & h_{2,1} & h_{1,1} & h_{0,1} \end{bmatrix} = \\
&= \begin{bmatrix} 1 & 1 & 0 & 1 \\ 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 1 \end{bmatrix} = [6 \ 7 \ 1 \ 5]_{10}
\end{aligned}$$

$$\mathbf{H}_0 = \begin{bmatrix} h_{2,3} & h_{1,3} \\ h_{2,2} & h_{1,2} \\ h_{2,1} & h_{1,1} \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 1 & 0 \\ 1 & 1 \end{bmatrix} \quad (5.22)$$

$$\mathbf{H}_R = [h_{0,3} \ h_{0,2} \ h_{0,1}]^T = [1 \ 0 \ 1]^T$$

$$\mathbf{H}_{out} = [h_{3,3} \ h_{3,2} \ h_{3,1}]^T = [1 \ 1 \ 0]^T$$

$$(\mathbf{S}_{t+1})_{3 \times 1} = (\mathbf{H}_0)_{3 \times 2} \cdot (\mathbf{U}_t)_{2 \times 1} + (\mathbf{T})_{3 \times 3} \cdot (\mathbf{S}_t)_{3 \times 1}$$

$$\mathbf{T} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 1 \end{bmatrix}$$

$$\mathbf{c}^t = \mathbf{H}_{out} \cdot \mathbf{S}_t + \mathbf{W} \cdot \mathbf{S}_{t+1}$$

unde vectorul $\mathbf{w} = [0 \ 0 \ 1]_{1 \times 3}$.

Interleaver-urile pe care le-am folosit sunt cele propuse în [DOB05], permutarea realizându-se în două etape. În prima etapă se realizează o permutare intra-simbol, iar în a doua etapă se realizează o permutare inter-simbol, conform modelelor particulare date în [DOB05]. Blocurile de date au lungimi egale cu 188 bytes.

Deasemenea, ca algoritmi de decodare am considerat algoritmi MAP și Max-Log-MAP. S-a considerat modulația QPSK și un număr de 15 iterații, cu un criteriu de oprire a iterațiilor bazat pe distribuția APP.

În Fig. 5.21 rezultă câștiguri de zeci de dB, în performanțele BER ale TCMB-urilor, pe canale de transmisie cu fading plat de tip Rayleigh, în comparație cu cazul necodat, așa cum am arătat și în [KBN06].

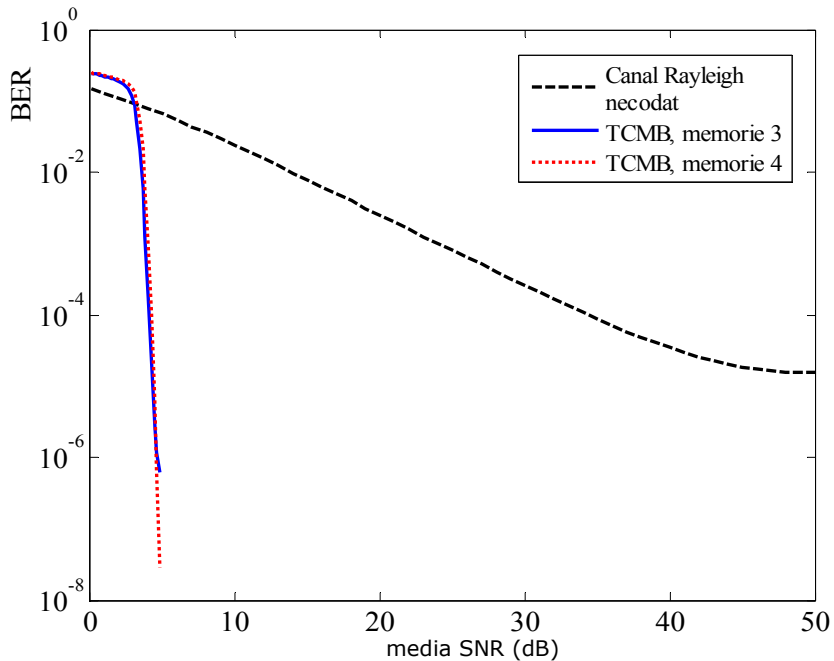


Fig. 5.21 Performanțele BER pentru cazul necodat și codat, cu TCMB-uri de memorie 3, respectiv de memorie 4, pe canale cu fading plat de tip Rayleigh. BER este reprezentat în funcție de media SNR (dB). Algoritmul de decodare implementat este algoritmul Max-Log-MAP.

În tabelul 5.3 și în tabelul 5.4 prezint cum sunt influențate performanțele BER și FER prin estimarea valorii SNR-ului, estimare dată de factorul L_c .

Tabelul 5.3 BER·10⁻⁹ în funcție de f și în funcție de starea TCMB-ului.

TCMB	SNR (dB)	$f = L_c / (4 \cdot R \cdot B)$				
		$f=0.4$	$f=0.5$	$f=0.6$	$f=0.7$	$f=0.8$
		BER	BER	BER	BER	BER
8-stări	4.2	1201433	8525	6013	5097	6057
16-stări	4.2	15727910	15780	2881	5836	28147

Tabelul 5.4 FER·10⁻⁶ în funcție de f și în funcție de starea TCMB-ului.

MBTC	SNR (dB)	$f = L_c / (4 \cdot R \cdot B)$				
		$f=0.4$	$f=0.5$	$f=0.6$	$f=0.7$	$f=0.8$
		FER	FER	FER	FER	FER
8-stări	4.2	22044	577	555	577	688
16-stări	4.2	200422	222	66	88	555

Valoarea parametrului L_c este dată de relația: $L_c = 4 \cdot R \cdot B \cdot f$, unde R este rata turbo codului duo binar și B este valoarea absolută a SNR-ului. Rezultatele obținute în cele două tabele, în funcție de BER și FER, la un SNR=4.2 dB pentru TCMB-uri de memorie 3, sunt mai bune pentru $f=0.7$. Pentru TCMB-urile de memorie 4 cele mai

bune rezultate se obțin la $f=0.6$. Această concluzie este verificată în Fig. 5.22. Este evident faptul că cele mai bune performanțe BER și FER sunt obținute în cazul utilizării unui TCMB de memorie 4, având $f=0.6$.

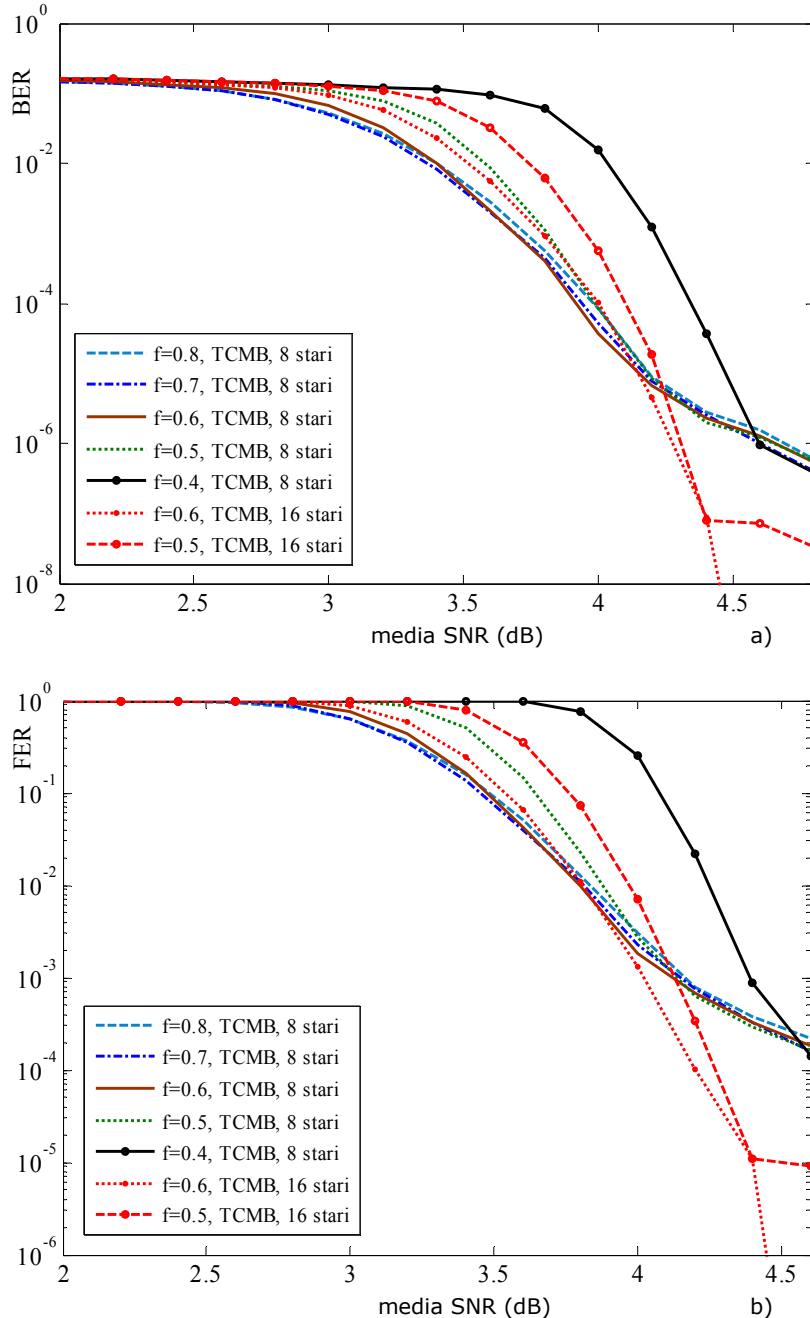


Fig. 5.22 Performanțele a) BER și b) FER ale TC-urilor duo-binare cu 8 stări și cu 16 stări pentru diferite valori ale factorului f . S-a folosit algoritmul de decodare MAP și modulația QPSK.

În Fig. 5.23 am reprezentat performanțele BER și FER ale TC-urilor duobinare (cu 8 stări și cu 16 stări) de rată 1/2, în funcție de raportul semnal pe zgomot, utilizând algoritmi de decodare MAP și Max-Log/MAP.

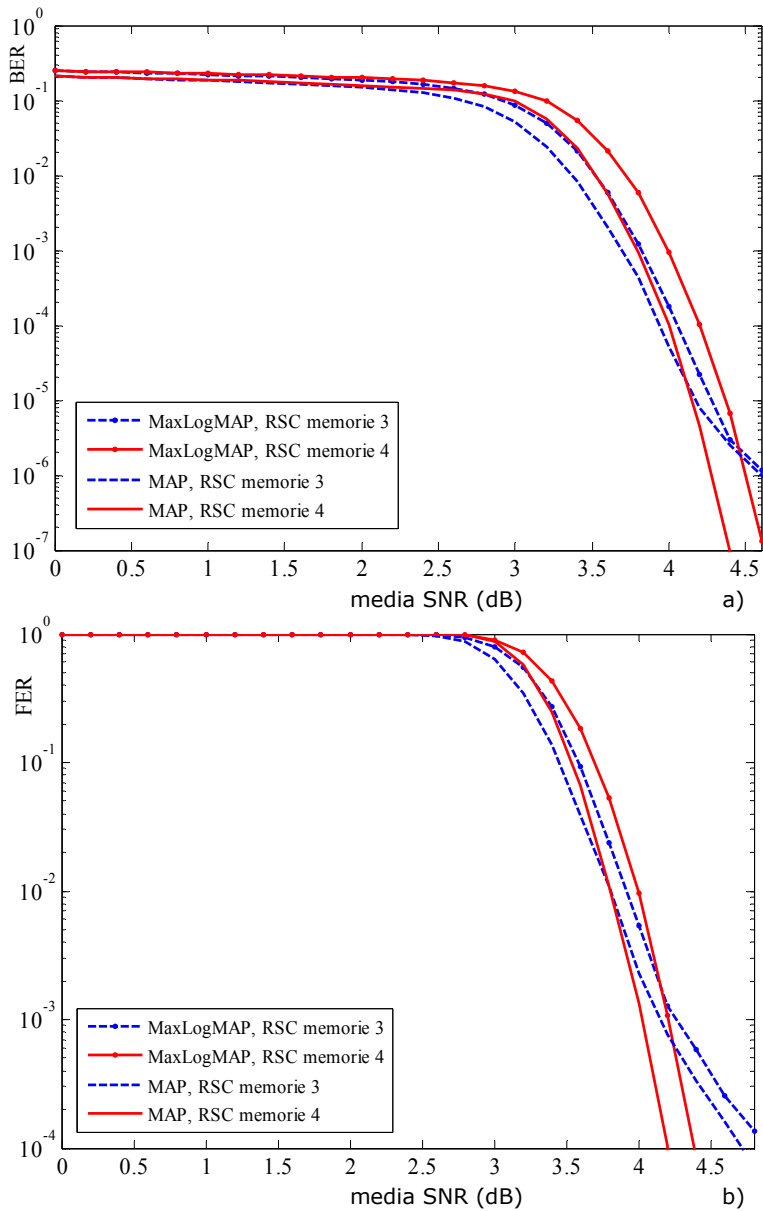


Fig. 5.23 Performanțele a) BER și b) FER. O comparație între două implementări ale algoritmului de decodare ale TCMB-urilor (cu $r=2$), pe canal cu fading plat de tip Rayleigh: algoritmul MAP și aproximarea sa Max-Log-MAP. TCMB-urile sunt bazate pe codurile RSC de memorie 3 (linie întreruptă) și de memorie 4 (linie continuă). Lungimea cuvântului de cod este egală cu $2 \cdot 752 = 1504$ biți.

Se observă faptul că pentru TC-uri duo-binare (ale căror curbe sunt reprezentate cu linie întreruptă), la SNR-uri mici, pierderea nu depășește 0.15 dB, iar la valori ale BER mai mici decât 10^{-5} , pierderea devine neglijabilă, spre deosebire de valorile mici ale lui FER unde pierderea rămâne constantă.

5.4 Concluzii

La începutul acestui capitol am analizat beneficiile aduse, din punctul de vedere al performanțelor BER și FER, de către decodarea pe simbol față de decodarea pe bit, în cazul TCMB-urilor. Metoda decodării pe simbol este superioară celei pe bit. De asemenea, se poate observa că atunci când se folosește algoritmul MaxLogMAP cu decodare pe simbol se obțin rezultate mai bune decât dacă se folosește algoritmul MAP cu decodare pe bit, [BKBP09].

O concluzie importantă ce rezultă din paragraful 5.1 este aceea că cel mai performant interleaver, din punct de vedere al BER este BN2 cel obținut prin metoda A, cu multiplexare a liniilor, [KOB05].

În paragraful următor am comparat TC-urile puncturate cu TCMB puncturate. Prin intermediul simulărilor pe care le-am realizat am ajuns la concluzia că pentru un FER egal sau mai mare decât 10^{-4} , aceste coduri se comportă similar. Iar, pentru valori ale FER mai mici, TCMB-urile puncturate au performanțe mai bune decât TC-urile puncturate, în principal datorită valorii *error floor* mai mici. Mai mult se observă că TCMB-urile la orice rată analizată în această abordare, se apropie la aproximativ 0.6 dB de limita Shannon, [KBDN07].

În paragraful 5.3 am analizat comportarea TCMB-urilor de rată 1/2, în canalele cu fading plat de tip Nakagami și Rayleigh. Modelele de canal cu fading plat de tip Nakagami acoperă o scară largă de situații practice, de la cel ne-selectiv în timp (canalul AWGN) până la cel mai selectiv în timp (canalul Rayleigh). Din simulările pe care le-am realizat se poate concluziona faptul că TCMB-urile au o comportare bună în canalele selective în timp, de tip Nakagami și Rayleigh, chiar și la lungimi moderate ale cuvintelor de cod (în toate simulările din acest paragraf am considerat lungimea cuvintelor de cod ca fiind egală cu 188 bytes), [KDBN06], [KBN06].

O altă concluzie importantă este aceea că, pe canale de transmisie cu fading plat de tip Rayleigh, ca și în cazul TC-ului unibinar, am obținut câștiguri de zeci de dB în performanțele BER ale TCMB-urilor, în comparație cu cazul necodat.

6. CONTRIBUȚII ȘI CONCLUZII

Turbo codurile reprezintă o clasă de coduri corectoare de erori, care se apropie cel mai mult de limita teoretică dată de C. Shannon în 1948.

Varianta clasică a unui turbo codor a fost propusă pentru prima dată în 1993, de către C. Berrou ș.a. și constă din conectarea în paralel a două coduri convoluționale, între care a fost intercalat un interleaver. Pe parcursul acestei lucrări s-a folosit această variantă de concatenare.

Ca și contribuții pot fi enumerate următoarele:

- ❖ În paragraful 2.6 am făcut o **analiză a spectrului ponderilor pentru codurile convoluționale**, de rată $R=1/2$ și lungime de constrângere $K=3$: nerecursive și sistematice, recursive și sistematice cât și nerecursive și nesistematice. Au rezultat 3 coduri (RSC[1,7/5], RSC[1,5/7] și NRNSC[5,7]) cu distanță minimă superioară celorlalte coduri analizate. Această distanță indică o superioritate a acestor coduri, din punctul de vedere al capacității de corecție, în special, la raporturi semnal/zgomot mari, unde sunt importante căile de ponderi mici.
- ❖ În capitolul 3, am făcut o prezentare succintă a turbo codului și a algoritmilor de decodare. În cadrul acestui capitol, mai precis în paragraful 3.4.6 **am propus două noi tipuri de interleaver-e: interleaver-ul bloc aleator în linie și interleaver-ul bloc cu linii aleatoare**, alături de câteva tipuri de interleaver-e clasice, deja cunoscute.
- ❖ Pe baza **interleaver-ilor propuse** și prezentate în paragraful 3.4, am făcut o **analiză a curbelor BER** obținute ca rezultat a propriilor simulări, pentru diferite lungimi ale blocului de date. Din analiza rezultatelor obținute, au rezultat următoarele concluzii:
 - Interleaver-ul bloc aleator în linie prezintă performanțe BER și FER apropiate de cele ale interleaver-ului S , cu S maxim, dar avantajul acestui interleaver este construcția sa mai simplă.
 - Interleaver-ul bloc cu linii aleatoare are performanțe similare cu cele ale interleaver-elor bloc cunoscute.
- ❖ În paragraful 3.5.5 **am făcut o analiză a performanțele algoritmilor de decodare** (MAP, Log-MAP și Max-Log-MAP), luând în considerare două tipuri de interleaver-e: interleaver-ul de tip S și interleaver-ul pseudo aleator. Din simulări, a rezultat faptul că algoritmul de decodare Max-Log-MAP este inferior (cu 0.2 dB) celorlalți algoritmi de decodare. De asemenea, la SNR-uri mai mari de 1dB, a rezultat că performanțele algoritmilor de decodare MAP și Log-MAP sunt similare.
- ❖ În capitolul 4 am prezentat pe scurt tipurile de fading ce pot apărea și am făcut o **analiză a performanțelor BER și FER ale turbo codurilor în canalele cu fading plat de tip Rayleigh, Rice și Nakagami**. Din rezultatele pe care le-am obținut se observă **câștiguri de zeci de decibeli obținute cu turbo coduri**

față de varianta necodată. Atât în cazul unui canal cu fading plat de tip Rayleigh, cât și de tip Rice, am făcut simulări cu noul interleaver propus (interleaver-ul bloc aleator în linie) și am constatat că performanțele BER obținute cu acesta sunt similare cu performanțele BER obținute în cazul utilizării interleaver-ului S.

- ❖ În paragrafele 4.3, 4.4 și 4.5, cât și în paragraful 5.3.2, **am realizat o adaptare a algoritmului MAP pentru canalele cu fading de tip Rayleigh, Rice și Nakagami.** Algoritmul MAP a fost elaborat pentru canale nefluctuante (AWGN). Datorită dificultăților de calcul pe care le implică canalul fluctuant adaptarea propriu-zisă s-a realizat în 2 pași. În primul pas am considerat modelul canalului fluctuant, introducând în formula de calcul a coeficientului L_c valoarea medie a amplitudinii de fading (de tip Rayleigh). Ulterior, am ajustat valoarea găsită teoretic pe baza simulărilor. S-a observat faptul că erori în estimare de până la 20% ale factorului L_c , influențează performanța TC-ului, în canalele Rayleigh și Rice, cu aproximativ 0.2 dB. În canalele Nakagami algoritmul de decodare MAP, pe care l-am propus, are rezultate mai bune decât în cazul optimal, pentru valori scalate ale lui L_c cu aproximativ 10% mai mici decât cele teoretice (pentru factorul de scalare egal cu 0.9).
- ❖ În capitolul 5 am **analizat două tipuri de decodare** ce pot fi făcute în turbo codurile multi-binare: **decodarea pe bit și decodarea pe simbol.** Din rezultatele pe care le-am obținut a rezultat, așa cum era de așteptat, că decodarea pe simbol este superioară celei pe bit.
- ❖ În paragraful 5.1 am **propus trei metode de întrețesere care pot fi folosite în structura turbo codurilor multi-binare** și am prezentat rezultatele experimentale pe care le-am obținut în urma simulării a două dintre cele trei metode descrise în cadrul acestui paragraf.
- ❖ În paragraful 5.2 am făcut **o analiză a performanțelor BER și FER ale turbo codurilor puncturate și ale turbo codurilor multi-binare puncturate.** Am arătat că pentru a obține rate de codare mai mari decât $r/(r+2)$, în cazul turbo codurilor multi-binare trebuie ignorate mai puține simboluri redundante, comparativ cu turbo codurile clasice. Pentru $FER < 10^{-4}$, turbo codurile multi-binare puncturate au performanțe mai bune decât turbo codurile puncturate, în principal datorită valorii lor de *error floor* mai mici.
- ❖ În paragraful 5.3 am **simulat performanțele turbo codurilor multi-binare în canale cu fading Nakagami și Rayleigh.** Am observat că, și în cazul utilizării acestor coduri, se obțin câștiguri de zeci de decibeli, în comparație cu cazul necodat. O altă concluzie este aceea că turbo codurile multi-binare se comportă bine în canalele selective în timp, chiar și la lungimi moderate ale cuvântului de cod (188 bytes), unde pentru cazul canalului cu fading de tip Nakagami, pierderea teoretică dată de limitele Shannon este aproximativ de 1.6 dB.

ANEXA A

Exemplu de decodare Viterbi cu decizie soft

Un exemplu de decodare Viterbi cu decizie soft este ilustrat în Fig.1A. Exemplificarea se face considerând codorul convoluțional din Fig.2.2.

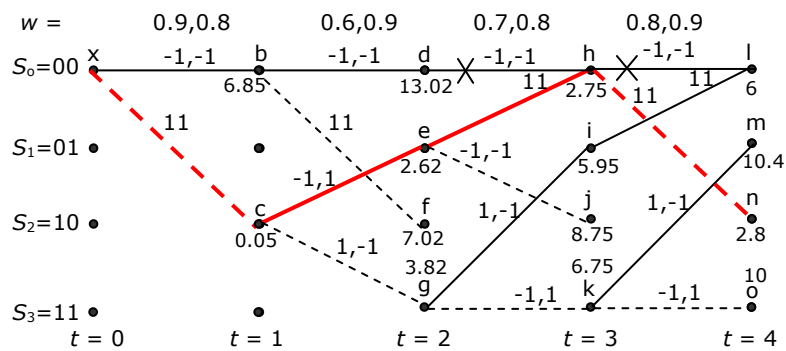


Fig.1A Decodarea cu decizie soft, în cazul codului convoluțional din fig.2.2.

Rezultă următoarele relații pentru:

- cadrul 1:

$$pm[1][b] = pm[0][x] + (0.9 - (-1))^2 + (0.8 - (-1))^2 = 0 + 1.9^2 + 1.8^2 = 3.61 + 3.24 = 6.85$$

$$pm[1][c] = pm[0][x] + (0.9 - 1)^2 + (0.8 - 1)^2 = 0 + 0.1^2 + 0.2^2 = 0.01 + 0.04 = 0.05$$

- cadrul 2:

$$pm[2][d] = pm[1][b] + (0.6 - (-1))^2 + (0.9 - (-1))^2 = 6.85 + 1.6^2 + 1.9^2 = 13.02$$

$$pm[2][e] = pm[1][c] + (0.6 - (-1))^2 + (0.9 - 1)^2 = 0.05 + 1.6^2 + 0.1^2 = 2.62$$

$$pm[2][f] = pm[1][b] + (0.6 - 1)^2 + (0.9 - 1)^2 = 6.85 + 0.4^2 + 0.1^2 = 6.85 + 0.16 + 0.01 = 7.02$$

$$pm[2][g] = pm[1][c] + (0.6 - 1)^2 + (0.9 - (-1))^2 = 0.05 + 0.4^2 + 1.9^2 = 0.05 + 0.16 + 3.61 = 3.82$$

- cadrul 3:

Metrica de cale pentru nodul h :

$$m_d = 13.02 + (0.7 - (-1))^2 + (0.8 - (-1))^2 = 13.02 + 1.7^2 + 1.8^2 = 19.15$$

$$m_e = 2.62 + (0.7 - 1)^2 + (0.8 - 1)^2 = 2.62 + 0.3^2 + 0.2^2 = 2.75$$

$$pm[3][h] = m_e = 2.75;$$

Deci metrica nodului h este 2.75.

Metrica de cale pentru nodul i :

$$m_f = 7.02 + (0.7 - (-1))^2 + (0.8 - 1)^2 = 7.02 + 1.7^2 + 0.2^2 = 9.95$$

$$m_g = 3.82 + (0.7 - 1)^2 + (0.8 - (-1))^2 = 2.62 + 0.3^2 + 1.8^2 = 5.95$$

$$pm[3][i] = m_g = 5.95;$$

Așadar metrica nodului i este 5.95.

Metrica de cale pentru nodul j :

$$m_d = 13.02 + (0.7 - 1)^2 + (0.8 - 1)^2 = 13.02 + 0.3^2 + 0.2^2 = 13.15$$

$$m_e = 2.62 + (0.7 - (-1))^2 + (0.8 - (-1))^2 = 2.62 + 1.7^2 + 1.8^2 = 8.75$$

$$pm[3][j] = m_e = 8.75;$$

Prin urmare metrica nodului j este 8.75.

Metrica de cale pentru nodul k :

$$m_f = 7.02 + (0.7 - 1)^2 + (0.8 - (-1))^2 = 7.02 + 0.3^2 + 1.8^2 = 10.35$$

$$m_g = 3.82 + (0.7 - (-1))^2 + (0.8 - 1)^2 = 3.82 + 1.7^2 + 0.2^2 = 6.75$$

$$pm[3][k] = m_g = 6.75;$$

Rezultă că metrica nodului k este 6.75.

- cadrul 4:

Metrica de cale pentru nodul l :

$$m_h = 2.75 + (0.8 - (-1))^2 + (0.9 - (-1))^2 = 2.75 + 1.8^2 + 1.9^2 = 9.6$$

$$m_i = 5.95 + (0.8 - 1)^2 + (0.9 - 1)^2 = 5.95 + 0.2^2 + 0.1^2 = 6$$

$$pm[4][l] = m_i = 6;$$

Deci metrica nodului l este 6.

Metrica de cale pentru nodul m :

$$m_j = 8.75 + (0.8 - (-1))^2 + (0.9 - 1)^2 = 8.75 + 1.8^2 + 0.1^2 = 12$$

$$m_k = 6.75 + (0.8 - 1)^2 + (0.9 - (-1))^2 = 6.75 + 0.2^2 + 1.9^2 = 10.4$$

$$pm[4][m] = m_k = 10.4;$$

Așadar metrica nodului m este 10.4.

Metrica de cale pentru nodul n :

$$m_h = 2.75 + (0.8 - 1)^2 + (0.9 - 1)^2 = 2.75 + 0.2^2 + 0.1^2 = 2.8$$

$$m_i = 5.95 + (0.8 - (-1))^2 + (0.9 - (-1))^2 = 5.95 + 1.8^2 + 1.9^2 = 12.8$$

$$pm[4][n] = m_h = 2.8;$$

Prin urmare metrica nodului n este 2.8.

Metrica de cale pentru nodul o :

$$m_j = 8.75 + (0.8 - 1)^2 + (0.9 - (-1))^2 = 8.75 + 0.2^2 + 1.9^2 = 12.4$$

$$m_k = 6.75 + (0.8 - (-1))^2 + (0.9 - 1)^2 = 6.75 + 1.8^2 + 0.1^2 = 10$$

$$pm[4][o] = m_k = 10;$$

Deci metrica nodului o este 10.

Așadar se găsește următoarea cale de întoarcere pe trellis :

$$n \Rightarrow h \Rightarrow e \Rightarrow c \Rightarrow x$$

Rezultă că secvența de intrare a codului corespunde lui $i=1001$.

Turbo codurile reprezintă o clasă puternică de coduri corectoare de erori, care se apropie cel mai mult de limita teoretică dată de C. Shannon în 1948.

ANEXA B

Relația lui Bayes

Dacă A și B sunt două evenimente ale aceluiași experiment, se poate scrie relația:

$$P(A/B) \cdot P(B) = P(A \wedge B) = P(B/A) \cdot P(A) \quad (\text{B.1})$$

unde:

- $P(B)$ reprezintă probabilitatea de realizare a evenimentului B;
- $P(A/B)$ reprezintă probabilitatea de realizare a evenimentului A, când s-a realizat evenimentul B;
- $P(A \wedge B)$ reprezintă probabilitatea de a se realiza evenimentele A și B.

O consecință utilă a regulii lui Bayes este următoarea:

$$P(\{A \wedge B\}C) \cdot P(B) = P(A\{B \wedge C\}) \cdot P(B/C) \quad (\text{B.2})$$

ce poate fi derivată din ecuația (B.1). Considerând că $X \equiv A \wedge B$ și $Y \equiv B \wedge C$, se poate scrie:

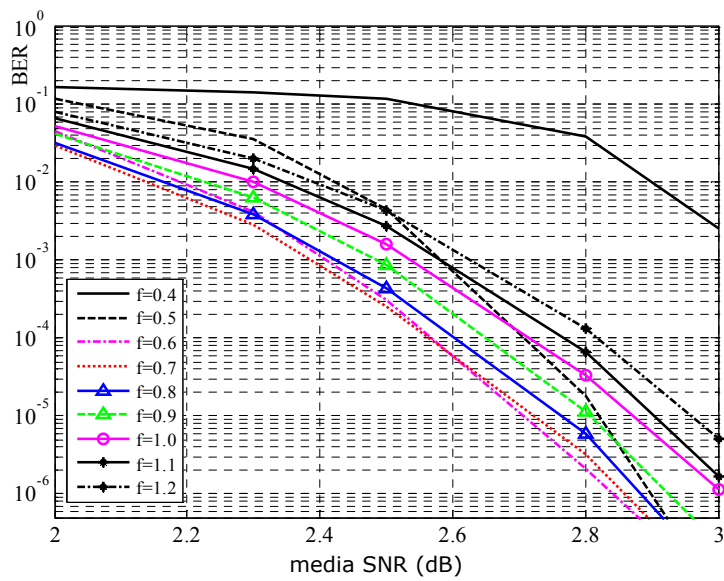
$$\begin{aligned} P(\{A \wedge B\}C) \cdot P(B) &\equiv P(X/C) = \frac{P(X \wedge C)}{P(C)} \\ &= \frac{P(A \wedge B \wedge C)}{P(C)} \equiv \frac{P(A \wedge Y)}{P(C)} \\ &= \frac{P(A|Y)P(Y)}{P(C)} \equiv P(A\{B \wedge C\}) \cdot \frac{P(B \wedge C)}{P(C)} \\ &= P(A\{B \wedge C\}) \cdot P(B/C). \end{aligned} \quad (\text{B.3})$$

ANEXA C

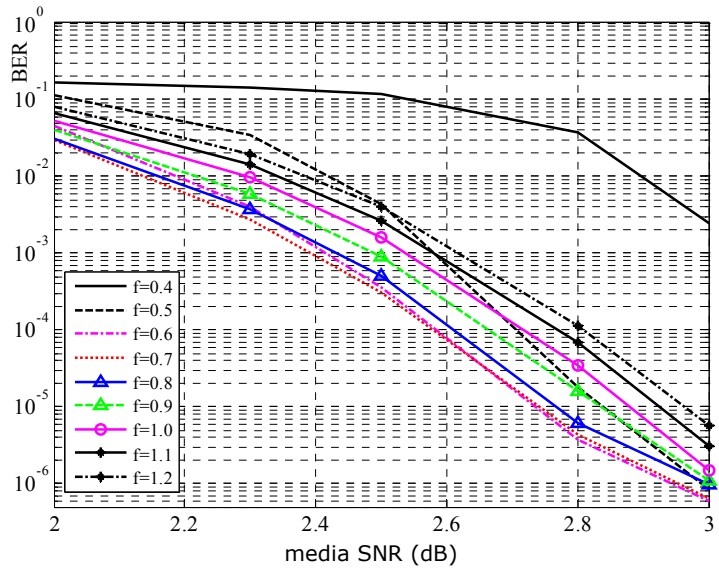
Simulări BER și FER pentru diferite valori ale lui K

S-a utilizat:

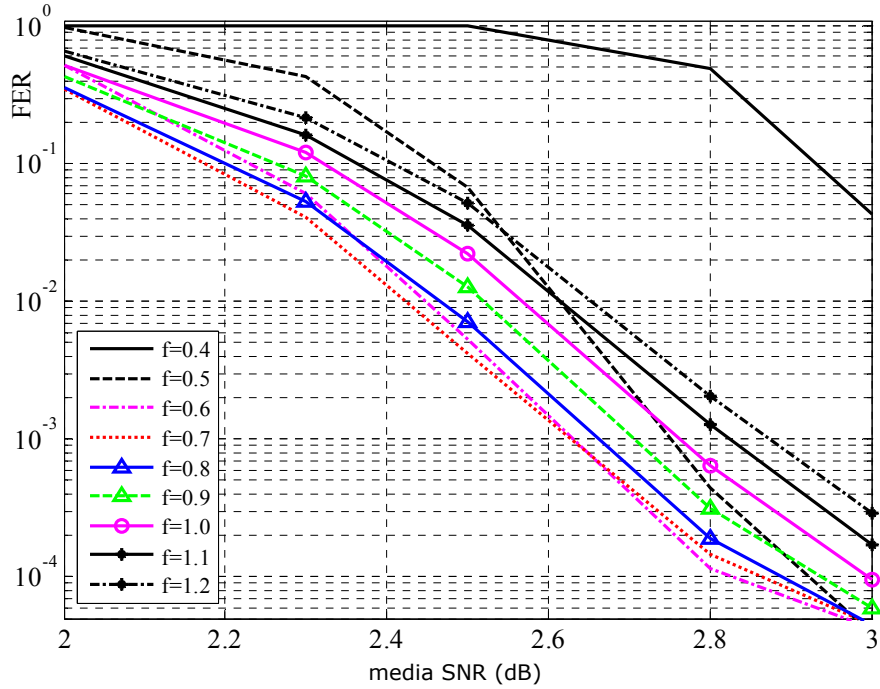
- interleaver-ul S (ilv1784S29): $K=0\%$



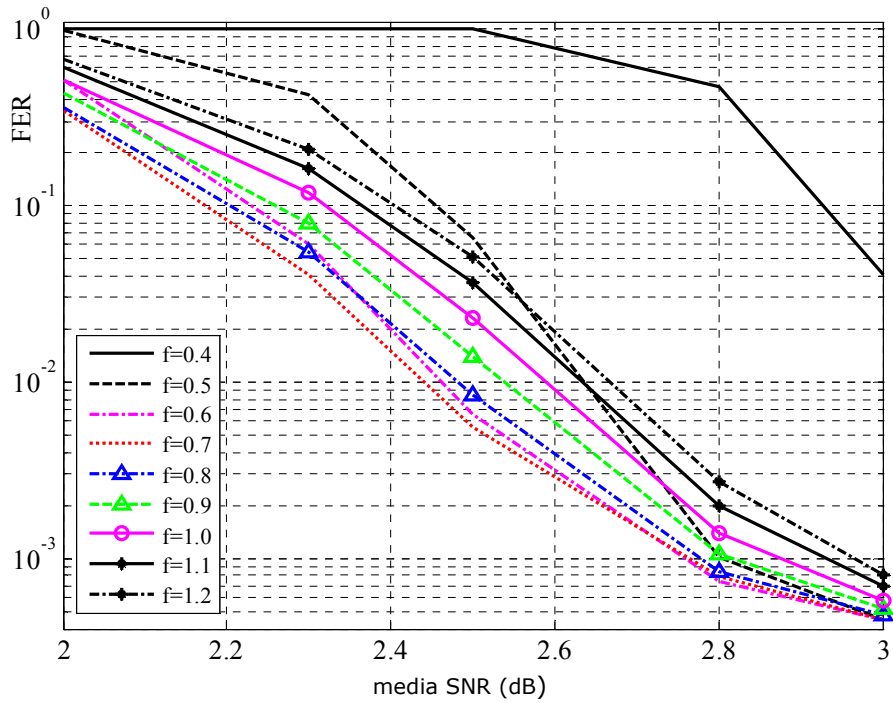
- interleaver-ul bloc aleator în linie:



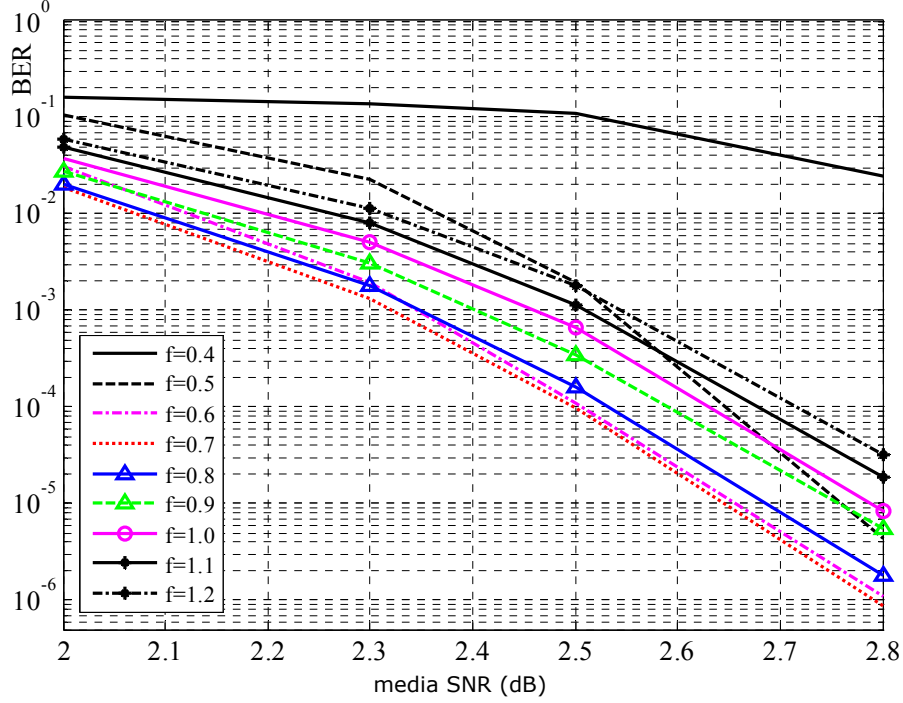
- interleaver-ul S: K=0%



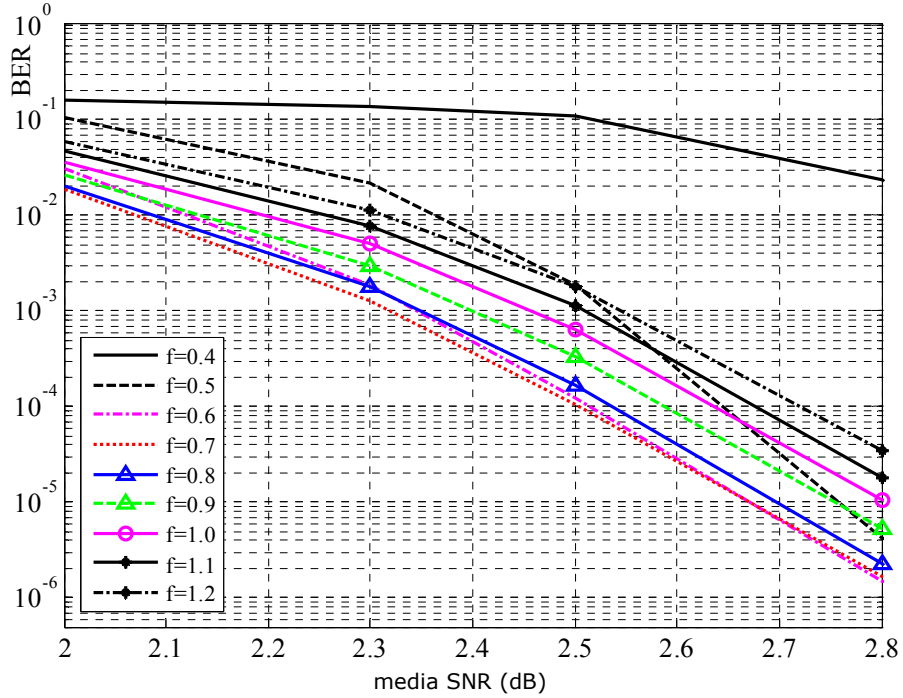
- interleaver-ul bloc aleator în linie:



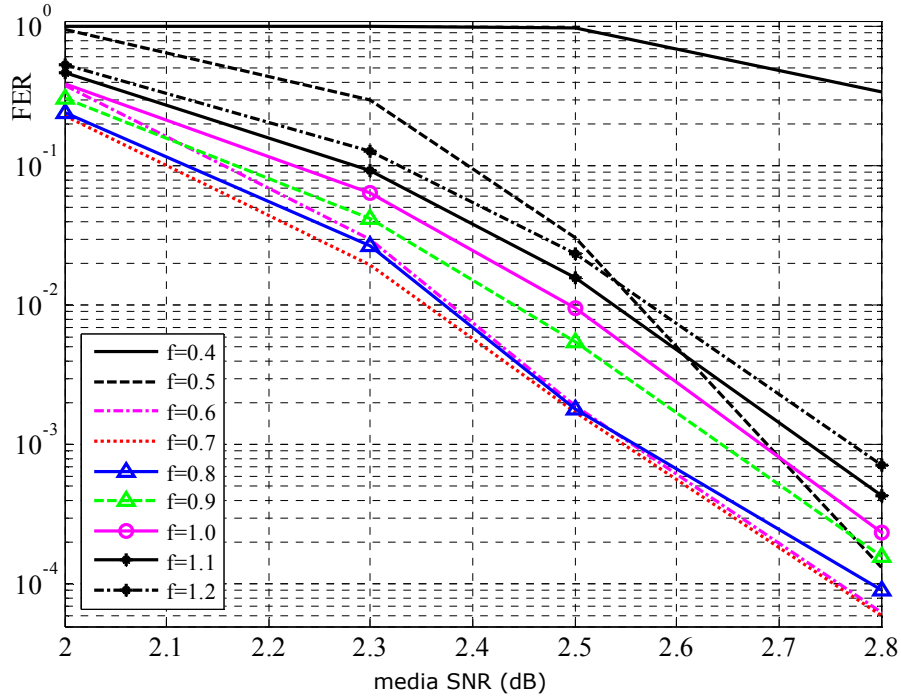
- interleaver-ul S: K=25%



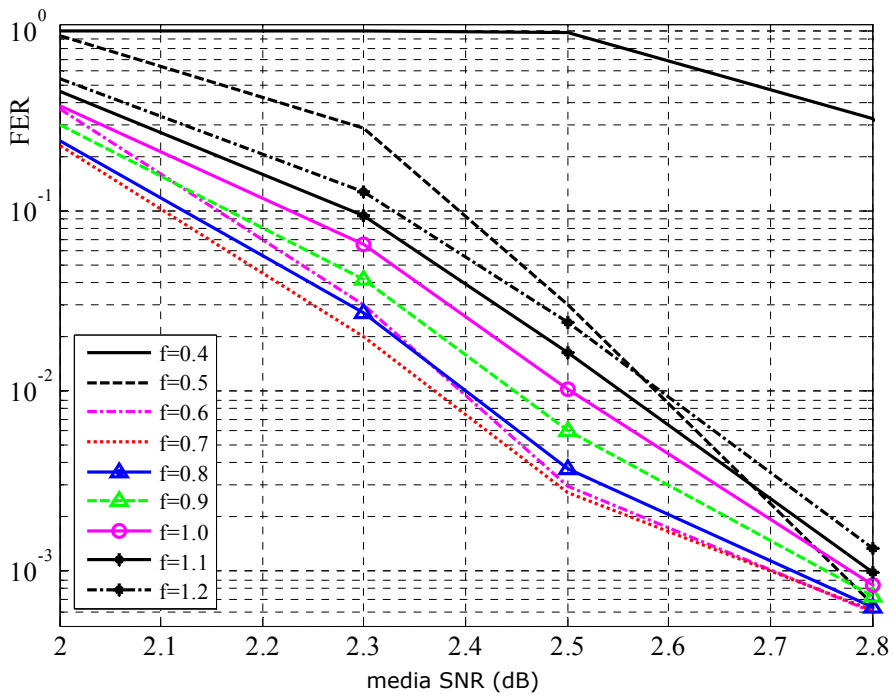
- interleaver-ul bloc aleator în linie:



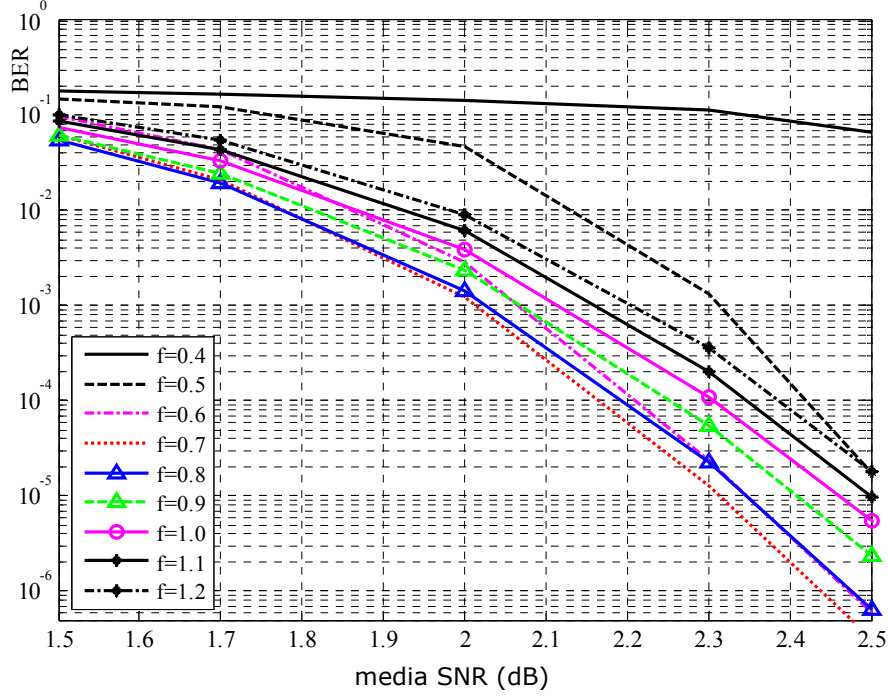
- interleaver-ul S: K=25%



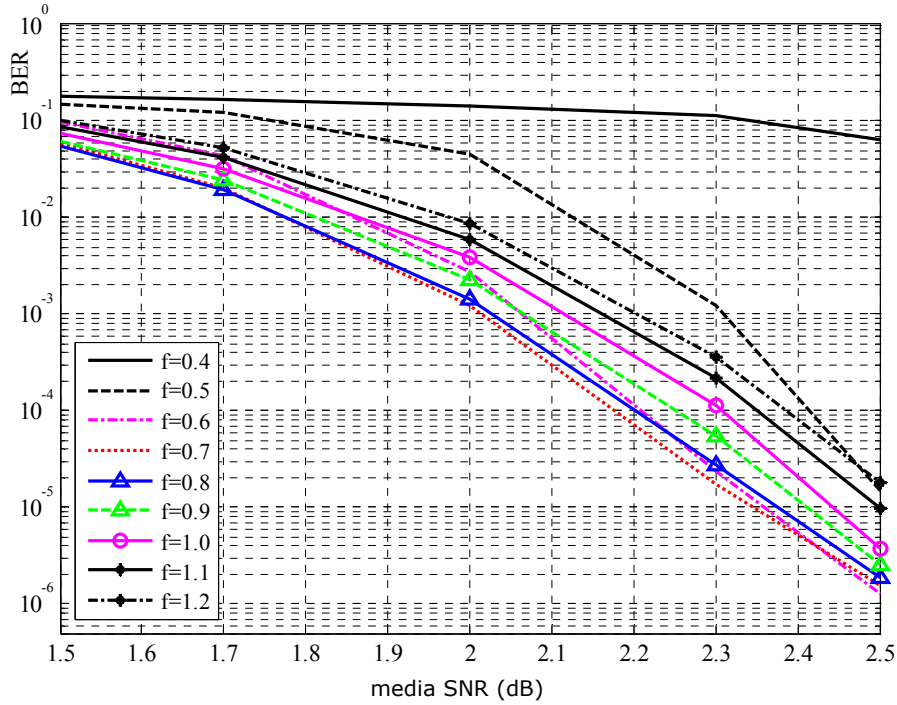
- interleaver-ul bloc aleator în linie:



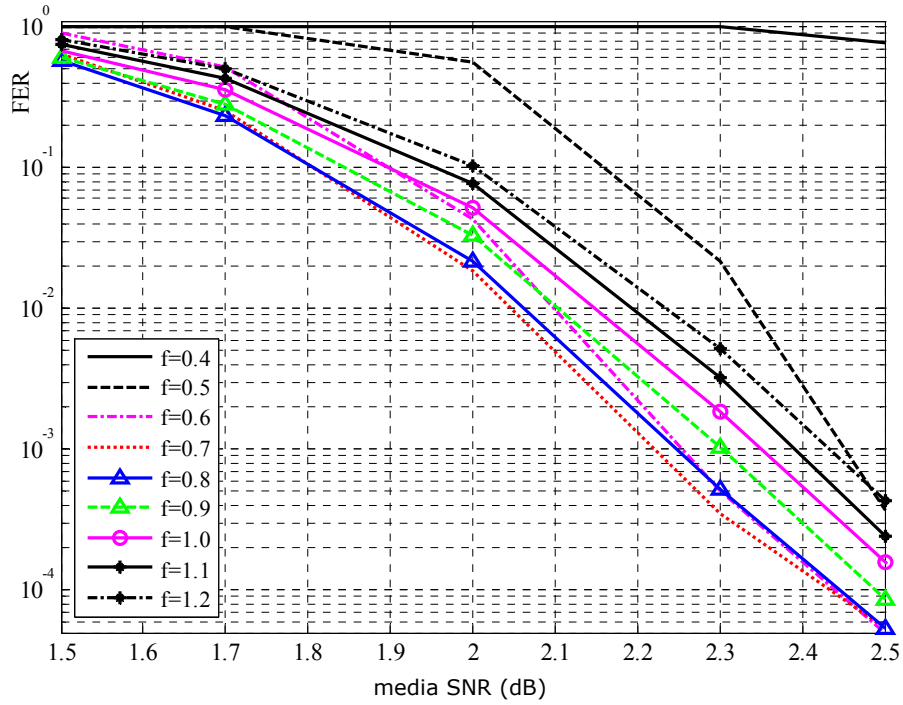
- interleaver-ul S: K=50%



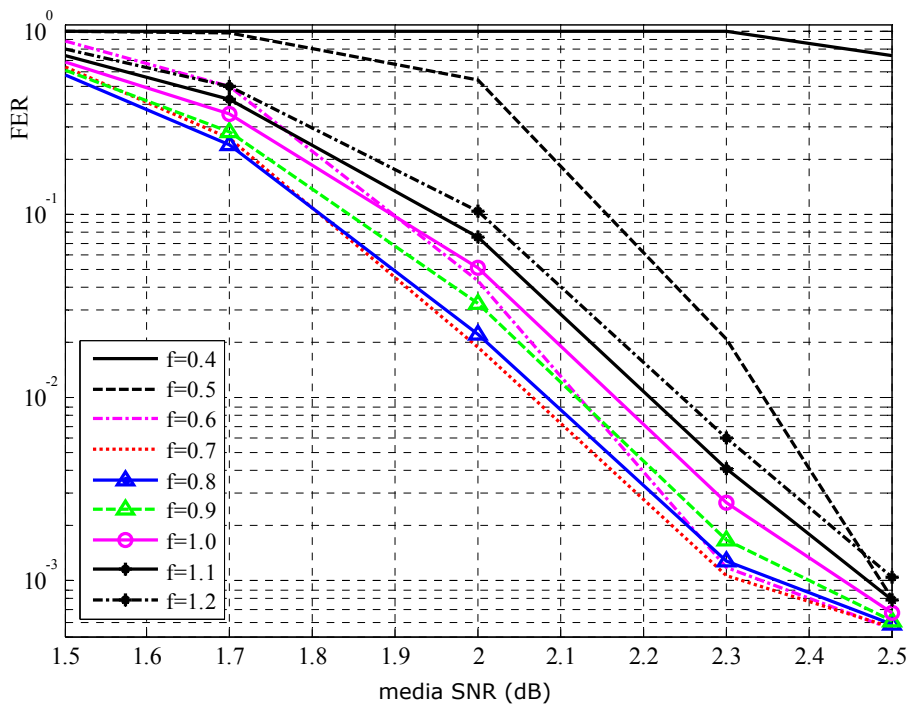
- interleaver-ul bloc aleator în linie:



- interleaver-ul S: K=50%

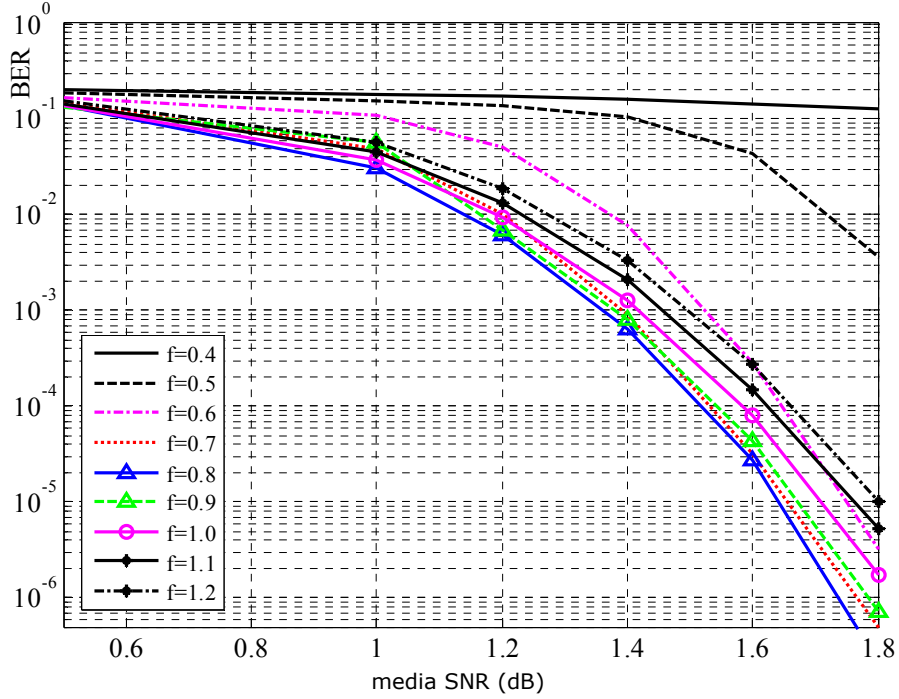


- interleaver-ul bloc aleator în linie:

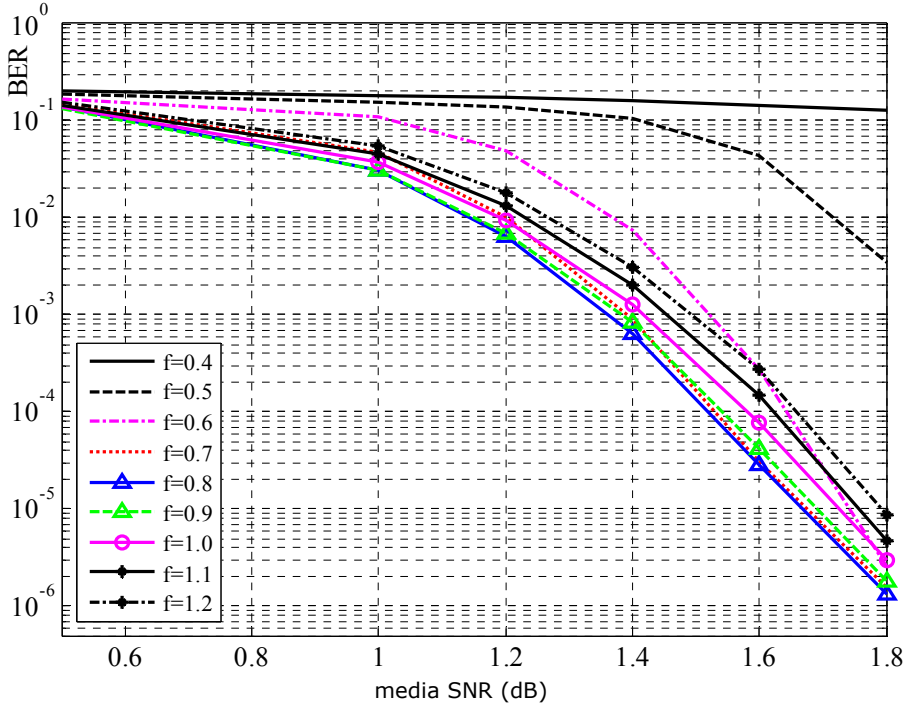


- interleaver-ul S:

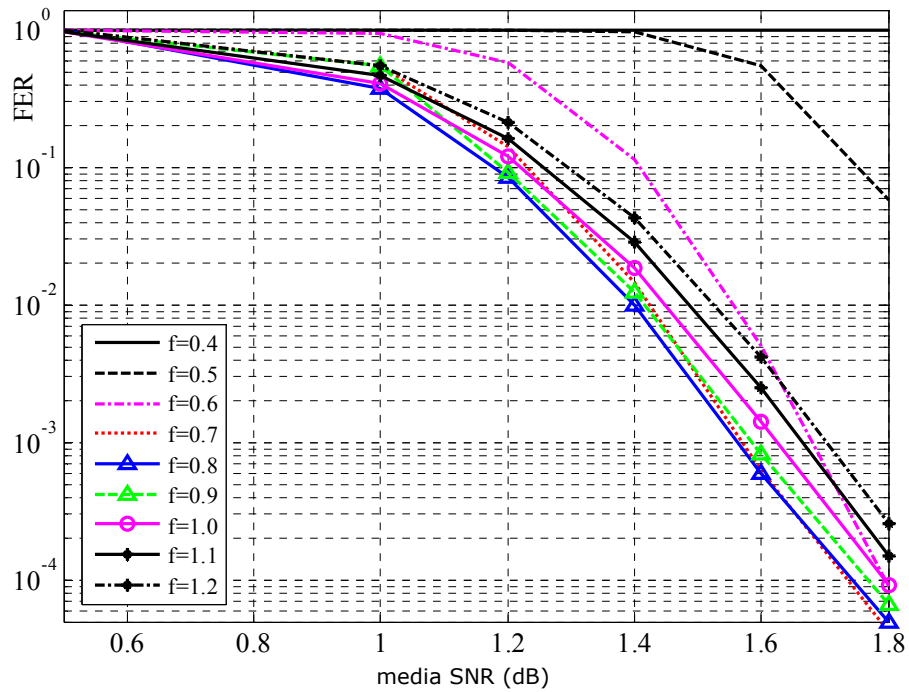
K=75%



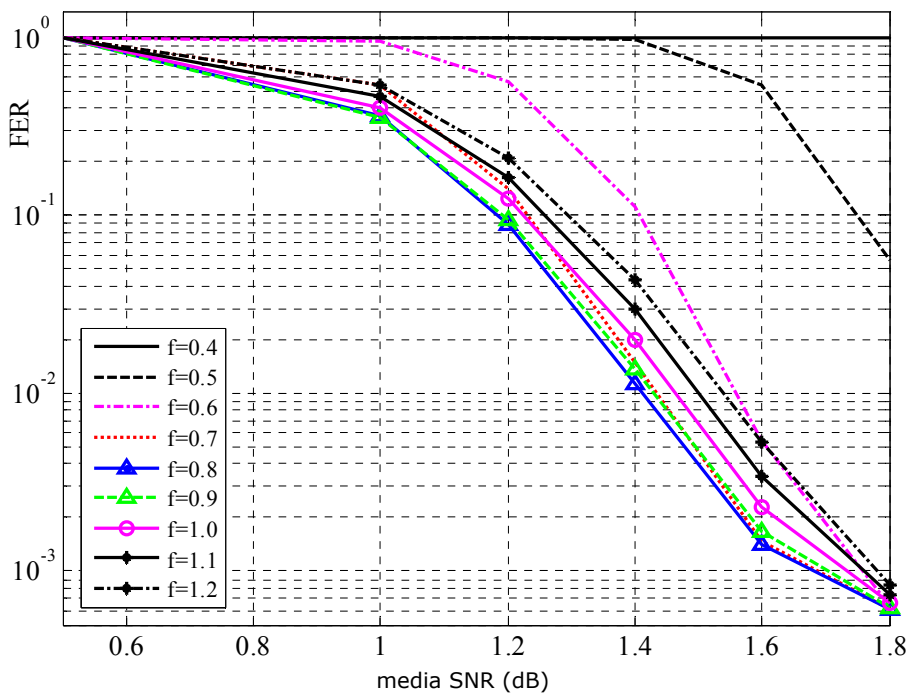
- interleaver-ul bloc aleator în linie:



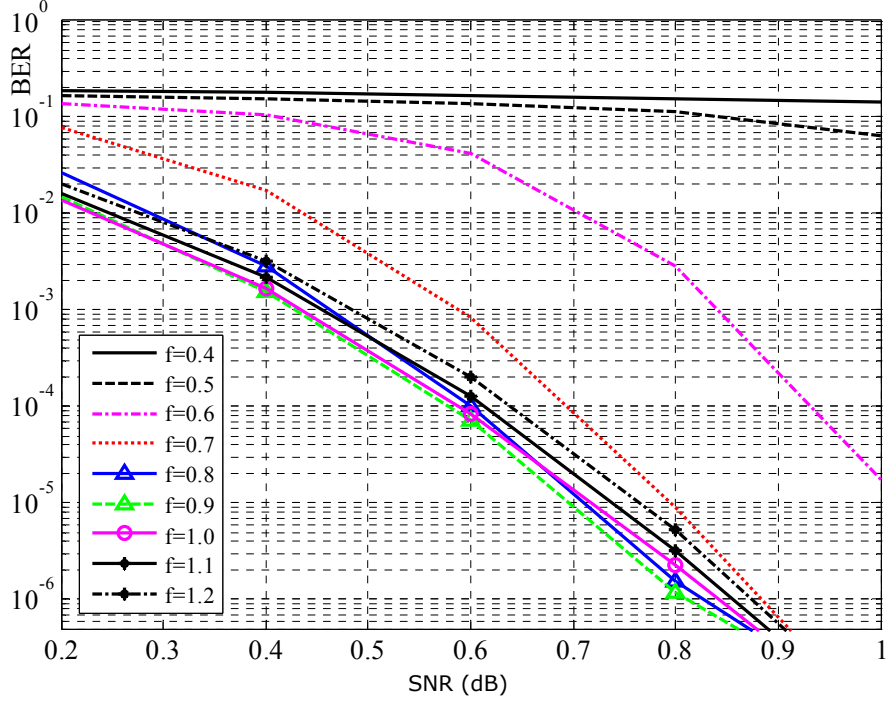
- interleaver-ul S: K=75%



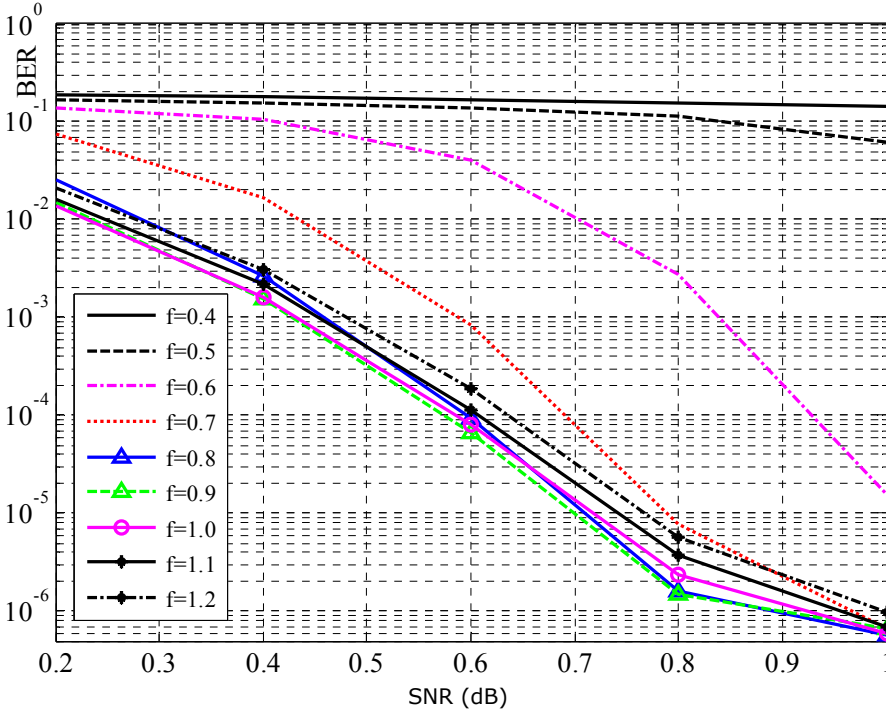
- interleaver-ul bloc aleator în linie:



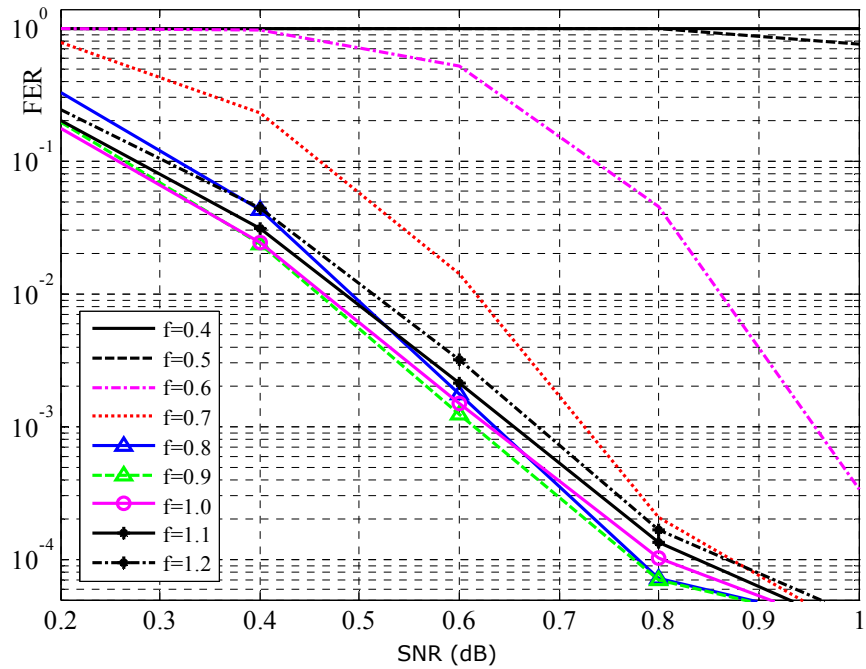
- interleaver-ul S: K=100%



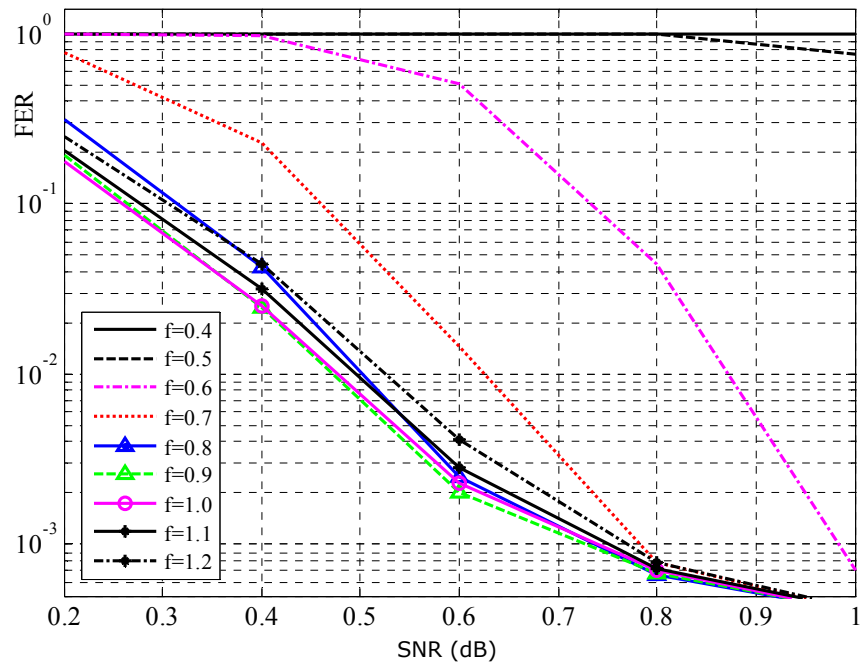
- interleaver-ul bloc aleator în linie:



- interleaver-ul S: K=100%



- interleaver-ul bloc aleator în linie:



ANEXA D

Tabele BER pentru diferite interleaver-e

Tabel D1: BER- 10^{-4} , pentru turbo coduri cu coduri 15/13 RSC.

K [%]	SNR [dB]	$f=Lc/(4 \cdot R \cdot B)$							
		0.4	0.5	0.6	0.7	0.8	0.9	1	1.1
0	2.8	363,4	0,196	0,034	0,046	0,066	0,190	0,382	0,382
25	2.8	235,2	0,046	0,009	0,009	0,026	0,046	0,102	0,169
50	2.5	638,2	0,137	0,011	0,015	0,015	0,024	0,053	0,097
75	1.8	1275,9	32,63	0,024	0,013	0,013	0,016	0,028	0,051
100	0.8	1568,7	1132,2	27,73	0,078	0,020	0,021	0,031	0,040

a) interleaver-ul *pseudo-aleator*, $N=1784$ biți;

K [%]	SNR [dB]	$f=Lc/(4 \cdot R \cdot B)$							
		0.4	0.5	0.6	0.7	0.8	0.9	1	1.1
0	2.8	344,8	0,141	0,108	0,112	0,127	0,171	0,257	0,405
25	2.8	210,1	0,082	0,077	0,078	0,075	0,092	0,101	0,160
50	2.5	633,5	0,120	0,084	0,085	0,088	0,085	0,106	0,109
75	1.8	1275,2	23,55	0,086	0,084	0,087	0,081	0,097	0,106
100	0.8	1569,8	1133,3	17,57	0,113	0,122	0,094	0,107	0,101

b) interleaver-ul *Takeshita-Costello*, $N=2048$ biți.

Tabel D2: BER- 10^{-4} , pentru turbo coduri cu coduri 25/23 RSC.

K [%]	SNR [dB]	$f=Lc/(4 \cdot R \cdot B)$							
		0.4	0.5	0.6	0.7	0.8	0.9	1	1.1
0	2.8	891,3	0,955	0,047	0,046	0,126	0,340	0,997	2,541
25	2.8	720,5	0,215	0,004	0,014	0,031	0,098	0,283	0,689
50	2.5	1147,4	1,189	0,012	0,012	0,014	0,043	0,111	0,313
75	1.8	1499,6	188,08	0,106	0,011	0,014	0,020	0,036	0,125
100	0.8	1691,7	1409,1	150,5	0,309	0,016	0,013	0,014	0,038

a) interleaver-ul BRL, $N=1785$ biți;

K [%]	SNR [dB]	$f=Lc/(4 \cdot R \cdot B)$							
		0.4	0.5	0.6	0.7	0.8	0.9	1	1.1
0	2.8	896,06	0,839	0,024	0,030	0,079	0,362	0,992	2,484
25	2.8	723,1	0,202	0,005	0,006	0,021	0,063	0,247	0,648
50	2.5	1148,8	1,209	0,008	0,007	0,011	0,031	0,0807	0,3044
75	1.8	1499,5	192,4	0,105	0,002	0,004	0,007	0,0238	0,099
100	0.8	1691,7	1408	153,1	0,336	0,015	0,010	0,012	0,029

b) interleaver-ul S , $N=1784$ biți;

K [%]	SNR [dB]	$f=Lc/(4 \cdot R \cdot B)$							
		0.4	0.5	0.6	0.7	0.8	0.9	1	1.1
0	2.8	877,4	0,773	0,012	0,023	0,064	0,287	0,932	2,219
25	2.8	698,01	0,163	0,004	0,010	0,018	0,073	0,211	0,615
50	2.5	1141,1	0,983	0,007	0,007	0,008	0,034	0,101	0,203
75	1.8	1499,1	177	0,067	0,006	0,007	0,009	0,023	0,092
100	0.8	1691,8	1407,8	139,88	0,290	0,017	0,009	0,014	0,031

c) interleaver-ul pseudo-aleator, $N=1784$ biți;

K [%]	SNR [dB]	$f=Lc/(4 \cdot R \cdot B)$							
		0.4	0.5	0.6	0.7	0.8	0.9	1	1.1
0	2.8	873,05	0,317	0,013	0,018	0,047	0,111	0,395	1,041
25	2.8	687,3	0,069	0,016	0,011	0,011	0,029	0,088	0,221
50	2.5	1144,6	0,432	0,020	0,016	0,010	0,016	0,051	0,117
75	1.8	1500,2	150,1	0,055	0,015	0,013	0,009	0,018	0,040
100	0.8	1692,1	1408	112,9	0,131	0,025	0,020	0,018	0,018

d) interleaver-ul *Takeshita-Costello*, $N=2048$ biți.

BIBLIOGRAFIE

- [ASH65] R. B. Ash, "*Information Theory*", Dover Publications, Urbana, Illinois, July, 1965;
- [BAAF93] C. Berrou, P. Adde, E. Angui, S. Faudeil, "A low complexity soft-output Viterbi decoder architecture", Proc.ICC'93, Geneva, Switzerland, May 1993, pp. 737-740;
- [BAB04] F. Babich, "On the Performance of Efficient Coding Techniques Over Fading Channels", IEEE Trans. on Wireless Communications, Vol. 3, no. 1, January 2004, pp. 290-299;
- [BAK04a] H. Baltă, **M. Kovaci**, "A Comparison Between Weight Spectrum of Different Convolutional Code Types", Analele Universității din Oradea, Fascicola Electrotehnică, Secțiunea Electronică, 2004, pp. 174-179;
- [BAK04b] H. Baltă, **M. Kovaci**, "Turbo-Coduri: Construcție, performanțe, perspectivă", Academia forțelor terestre "Nicolae Bălcescu" Sibiu, a VIII-a Sesiune de Comunicări Științifice, 11 iunie, 2004, pp. 27-34;
- [BAK04c] H. Baltă, **M. Kovaci**, "Studiu asupra preciziei curbilor BER construite prin simularea funcționării turbo-codurilor", Academia forțelor terestre "Nicolae Bălcescu" Sibiu, a VIII-a Sesiune de Comunicări Științifice, 11 iunie, 2004, pp. 35-42;
- [BAK04d] H. Baltă, **M. Kovaci**, "A study on turbo decoding iterative algorithms", Buletinul Științific al Universității "Politehnica" din Timișoara, Transactions on Electronics and Communications, Tom 49(63), Fascicola 2, 2004, Oct. 22-23, Timișoara, pp. 33-37;
- [BAK04e] H. Baltă, **M. Kovaci**, "The performances of convolutional codes used in turbo codes", Buletinul Științific al Universității "Politehnica" din Timișoara, Transactions on Electronics and Communications, Tom 49(63), Fascicola 2, 2004, Oct. 22-23, pp. 38-43;
- [BAK05a] H. Baltă, **M. Kovaci**, "A Study on Non-Binary Turbo Codes", The 36-th International Scientific Symposium of Military Equipment & Technologies Research Agency, 2005, mai 26-27, București, România, pp. 214-219;
- [BAK05b] H. Baltă, **M. Kovaci**, "The Turbo-Codes Performances In The (Radio) Rice Flat Fading Channels", The 31st Internationally Attended Scientific Conference "Modern Technologies In The XXI Century", Military Technical Academy, 2005, nov.03-04, București, pp. 10.1-10.6;
- [BAP94] S. A. Bărbulescu, S. S. Pietrobon, "Interleaver design for turbo codes", IEE Electronics Letters, December 1994, pp. 2107-2108;

- [BAP98] S. A. Bărbulescu, S. S. Pietrobon, "Turbo Codes: a tutorial on a new class of powerful error correcting coding schemes. Part I: Code Structures an Interleaver Design", 26 October, 1998;
- [BCJR74] L. R. Bahl, J. Cocke, F. Jelinek, J. Raviv, "Optimal Decoding of Linear Codes for Minimising Symbol Error Rate", IEEE Transactions on Information Theory, Vol. 20, March 1974, pp. 284-287;
- [BCVSV02] J. Boutros, G. Caire, E. Viterbo, H. Sawaya, S. Vialle, "Turbo Code at 0.03 dB from Capacity Limit," Proc. Int'l. Symp. Info. Theory, ISIT 2002, Lausanne, Switzerland, June 30 -July 5 2002, p. 56;
- [BDBK08] H. Baltă, Al. De Baynast, **M. Kovaci**, "On the Encoding of the Multi-Binary Convolutional Codes", Buletinul Științific al Universității „Politehnica” Timișoara, Seria Electronică și Telecomunicații, Tom 53-67 Electronică și Telecomunicații, Fascicola 2, 2008, Timișoara, România, pp. 183-188;
- [BDK06] H. Baltă, C. Douillard, **M. Kovaci** "The Minimum Likelihood APP Based Early Stopping Criterion for Multi-Binary Turbo Codes", Buletinul Științific al Universității „Politehnica” Timișoara, Seria Electronică și Telecomunicații, Tom 51-65 Electronică și Telecomunicații, Fascicola 2, 2006, Timișoara, România, pp. 199-203;
- [BEAD91] N.C. Beaulieu, A.A. Abu-Dayya, "Analysis of equal gain diversity on Nakagami fading channels", IEEE Transactions on Communications, Volume 39, Issue 2, Feb. 1991 pp. 225 - 234;
- [BEC05] N. Beaulieu, C. Cheng, "Efficient Nakagami-m Fading Channel Simulation", IEEE journal on Vehicular Technology, vol. 54, no. 2, March 2005;
- [BEJ99] C. Berrou, M. Jézéquel, "Nonbinary convolutional codes for turbo coding", Electron. Lett., vol. 35, no.1, Jan. 1999, pp. 39-40;
- [BEM96a] S. Benedetto, G. Montorsi, "Design of parallel concatenated convolutional codes", IEEE Transactions on Communications, vol. 44, May 1996, pp. 591-600;
- [BEM96b] S. Benedetto, G. Montorsi, "Unveiling turbo codes: Some results on parallel concatenated coding schemes", IEEE Transactions on Information Theory, vol. 42, March 1996, pp. 409-428;
- [BGT93] C. Berrou, A. Glavieux, P. Thitimajashima, "Near Shannon Limit Error-Correcting Coding and Decoding: Turbo Codes", Proc.ICC'93, Geneva, Switzerland, May 1993, pp. 1064-1070;
- [BJDK01] C. Berrou, M. Jézéquel, C. Douillard, S. Kerouédan, "The advantages of nonbinary turbo codes," in Proc. IEEE Inf. Theory Workshop, Cairns, Australia, Sep. 2001, pp. 61-63;

- [BKBP09] H. Baltă, **M. Kovaci**, C. Botiz, C. Poenaru, "Bit Decoding Versus Symbol Decoding in Multi-Binary Turbo Decoders", The 4-th International Conference on Engineering Technologies-ICET 2009, Novi Sad, April 28-30, 2009;
- [BKDB05] H. Baltă, **M. Kovaci**, Al. De Baynast, "Performance of Turbo-Codes on Nakagami Flat Fading (Radio) Transmission Channels", Proceedings of IEEE ASILOMAR Conference, 25-27 octombrie 2005, USA, pp. 606-610;
- [BKDVL06] H. Baltă, **M. Kovaci**, Al. De Baynast, C. Vlădeanu, R. Lucaciu, "Multi-Non-Binary Turbo-Codes From Convolutional to Reed-Solomon Codes", Buletinul Științific al Universității „Politehnica” Timișoara, Seria Electronică și Telecomunicații, Tom 51-65 Electronică și Telecomunicații, Fascicola 2, 2006, Timișoara, România, pp.113-118;
- [BKN05a] H. Baltă, **M. Kovaci**, M. Naforniță, „A New Method for the Simulation of the Nakagami Flat Fading (Radio) Transmission Channels”, Analele Universității din Oradea, Fascicola Electrotehnică, secțiunea Electronică, 2005, Oradea, România, pp. 21-24;
- [BKN05b] H. Baltă, **M. Kovaci**, M. Naforniță, "A Study on Turbo Coding Systems with $\pi/4$ Shifted DQPSK Modulation", Proceedings of IEEE International Symposium on SCS, ISSCS, July 14-15, 2005, Iași, România, pp. 367-370;
- [BKN06] H. Baltă, **M. Kovaci**, M. Naforniță "Performance of Multi Binary Turbo Codes on Rice Flat Fading Channels", Analele Universității din Oradea, Fascicola Electrotehnică, Secțiunea Electronică, 2006, pp. 17-20;
- [BKT06] H. Baltă, **M. Kovaci**, L. Trifina "A Reduced Memory MAP Algorithm for Turbo Codes", Proceedings of the 8th International Conference on Development and Application Systems, Suceava, 25-27 May, 2006, pp. 164-168;
- [BOR99] M. E. Borda, "Teoria transiterii informației", Editura Dacia, Cluj-Napoca, 1999;
- [BRH00] M. Breiling, L. Hanzo, "Optimum non-iterative decoding of turbo codes", IEEE Transactions on Information Theory, vol. 46, September 2000, pp. 2212-2228;
- [BRH97] M. Breiling, L. Hanzo, "Optimum Non-iterative Turbo-Decoding", Proceedings of PIMRC 97, Helsinki, Finland, September 1997, pp. 714-718;
- [CCD03] L. Conde Canencia, C. Douillard, "Performance estimation of 8-PSK turbo coded modulation over Rayleigh fading channels," 3rd International Symposium on Turbo Codes & Related Topics, Brest, France, Sept. 2003, pp. 567-570;

- [CCSDS02] Consultative Committee for Space Data Systems "Telemetry Channel Coding", CCSDS 101.0-B-6, Blue Book, October, 2002, <http://public.ccsds.org/publications/archive/101x0b6s.pdf>;
- [CFRU01] S.Y. Chung, G. D. Forney, Jr, T. J. Richardson, R. Urbanke, "On the Design of Low-Density Parity-Check Codes within 0.0045 dB of the Shannon Limit," IEEE Communication Letter, vol. 5, no. 2, Feb. 2001, pp. 58-60;
- [CRO00] S. Crozier, "New High-Spread High-Distance Interleavers for Turbo-Codes", 20th Biennial Symposium on Communications, Kingston, Ontario, Canada, May 28-31, 2000, pp. 3-7;
- [CWV98] A.J. Coulson, A.G. Williamson, R.G. Vaughan, "Improved fading distribution for mobile radio Communications", IEE Proceedings Volume 145, Issue 3, June 1998, pp. 197 - 202;
- [DIP95] D. Divsalar, F. Pollara, "Turbo codes for PCS applications", IEEE Int. Conf. Communications, 18-22 June, 1995, pp. 54-59;
- [DOB05] 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;
- [DOD95] S. Dolinar, D. Divsalar, "Weight Distributions for Turbo Codes Using Random and Nonrandom Permutations", TDA Progress Report 42-122, August 15, 1995;
- [ELI54] P. Elias, "Error-free Coding", IRE Trans. Inform. Theory, vol IT-4, 1954, pp. 29-37;
- [EPG94] J. A. Erfanian, S. Pasupathy, G. Gulak, "Reduced complexity symbol detectors with parallel structures for ISI Channels", IEEE Trans. Commun., vol.42, 1994, pp. 1661-1671;
- [ETSI02] ETSI EN 301 958 V1.1.1, "Interaction Channel for Digital Terrestrial Television", European Standard, DVB, March 2002;
- [ETSI03] ETSI EN 301 790 V1.3.1, "Interaction Channel for Satellite Distribution Systems", European Standard, DVB, March, 2003;
- [FAN63] R. M. Fano, "Heuristic discussion of probabilistic decoding", IEEE Transactions on Information Theory, IT-9, pp. 64-73, Apr. 1963;
- [FOR70] Forney, G. "Convolutional codes I: Algebraic structure", IEEE Trans. Information Theory, IT-16(6), 1970, pp. 720-738;
- [GCH99] K. Gracie, S. Crozier, A. Hunt, "Performance of a Low-Complexity Turbo Decoder with a Simple Early Stopping Criterion Implemented on a SHARC Processor", International Mobile Satellite Conference (IMSC'99), Ottawa, Canada, June 16-18, 1999;

- [GGB94] S. L. Goff, A. Glavieux, C. Berrou, "Turbo-codes and high spectral efficiency modulation", in Proceedings of IEEE International Conference on Communications, 1994, pp. 645-649;
- [GHA04] A. Ghayeb, T. Abualrub, "Asymptotic performance comparison of concatenated (turbo) codes over $GF(4)$ ", Int. J. Commun. Syst. 2004, 17, pp. 479-490;
- [GLJ96] A. Glavieux, M. Joindot, "Communications numériques. Introduction", Masson, Paris, 1996;
- [HAG95] J. Hagenauer, "Source Controlled Channel Decoding", IEEE Trans. Commun., vol. 43, Sept. 1995, pp. 2449-2457;
- [HAH89] J. Hagenauer, P. Hoeher, "A Viterbi algorithm with soft-decision outputs and its applications", Proc. of GLOBECOM'89, Dallas, Texas, 1989 pp. 47.1.1-47.1.7;
- [HAW98] Eric K. Hall, Stephen G. Wilson, "Design and Analysis of Turbo Codes on Rayleigh Fading Channels", IEEE journal on selected areas in communications, vol. 16, no. 2, February 1998;
- [HAY01] S. Haykin, "Digital Communication", John Wiley & Sons, 2001;
- [HLY02] L. Hanzo, T.H. Liew, B.L. Yeap, "Turbo Coding, Turbo Equalisation and Space-Time Coding for Transmission over Fading Channels", John Wiley & Sons Ltd, England, 2002;
- [HOP96] J. Hagenauer, E. Offer, L. Papke, "Iterative decoding of binary block and convolutional codes", IEEE Transactions on Information Theory, vol. 42, March 1996, pp. 429-445;
- [HOT04] Harri Holma, Antti Toskala, "WCDMA for UMTS. Radio Access for Third Generation Mobile Communications", Third Edition, John Wiley & Sons Ltd, The Atrium, Southern Gate, West Sussex PO19 8SQ, England, 2004;
- [JBD04] M. Jézéquel, C. Berrou, C. Douillard, "Turbo codes (convolutifs)", Seminar Timișoara, 15-18 Mar. 2004, <http://hermes.etc.upt.ro/cercetare/carti.html>;
- [JSC98] Li-Der Jeng, Yu T. Su and Jung-Tang Chiang, "Performance of Turbo Codes in Multipath Fading Channels", VTC'98, pp. 61-65;
- [KBDN07] **M. Kovaci**, H. Baltă, Al. De Baynast, M. Naforniță, "Performance Comparison of Punctured Turbo Codes and Multi Binary Turbo Codes", Proceedings of IEEE International Symposium on Signal, Circuits and Systems, ISSCS 2007, Iași, România, July, 12-13, 2007, pp. 485-488;
- [KBN05] **M. Kovaci**, H. Baltă, M. Naforniță, "The Performance of Interleavers used in Turbo Codes", Proceedings of IEEE International Symposium SCS, ISSCS'2005, Iași, July, 14-15, 2005, pp. 363-366;

- [KBN06] **M. Kovaci**, H. Baltă, M. Naforniță "Performance of Multi Binary Turbo Codes on Rayleigh Flat Fading Transmission Channels", Analele Universității din Oradea, Fascicula Electrotehnică, Secțiunea Electronică, 2006, pp. 64-67;
- [KBN09] **M. Kovaci**, H. Baltă, M. Naforniță, "On Using Turbo Codes Over Rice Flat Fading Channels", Proceedings of IEEE International Symposium SCS, ISSCS'2009, Iași, July, 09-10, 2009, pp. 461-464;
- [KDBN06] **M. Kovaci**, Al. De Baynast, H. Baltă, M. Naforniță "Performance of Multi Binary Turbo-Codes on Nakagami Flat Fading Channels", Buletinul Științific al Universității „Politehnica” Timișoara, Seria Electronică și Telecomunicații, Tom 51-65 Electronică și Telecomunicații, Fascicula 2, 2006, Timișoara, România, pp. 140-145;
- [KOB05] **M. Kovaci**, H. Baltă, "Non-Binary Turbo Codes Interleavers", The 36-th International Scientific Symposium of Military Equipment & Technologies Research Agency, mai 26-27, București, România, 2005, pp. 208-213;
- [KOB09] **M. Kovaci**, H. Baltă, "On Using Turbo Coding over Rayleigh Flat Fading Channels", The 4-th International Conference on Engineering Technologies- ICET 2009, Novi Sad, April 28-30, 2009;
- [KOB90] W. Koch, A. Baier, "Optimum and sub-optimum detection of coded data disturbed by time-varying inter-symbol interference", IEEE Globecom, Dec.1990, pp. 1679-1684;
- [KOV05] **M. Kovaci**, Referat Nr.2, "Canale de comunicație: Gaussian, Rice-an, Rayleigh-an", Dep. Comunicații, Fac. ETC, UPTimișoara, martie 2005;
- [MCK4] Y. Ould-Cheikh-Mouhamedou, S. Crozier, P. Kabal, "Distance Measurement Method for Double Binary Turbo Codes and a New Interleaver Design for DVB-RCS", Proceedings of the 47th annual IEEE Global Telecommunications Conference (Globecom 2004), Dallas, Texas, USA, Nov. 29-Dec. 3, 2004;
- [OMU69] J. K. Omura, "On the Viterbi Decoding Algorithm", IEEE Transactions on Information Theory, Jan.1969, pp. 177-179;
- [PAR92] J. D. Parson, "The Mobile Radio Propagation Channel", Second Edition, John Wiley & Sons Ltd, 1992;
- [PRO00] J. Proakis, "Digital Communications", 4th Ed., McGraw Hill, New York, 2000;
- [PSC96] L. Perez, J. Seghers, D. Costello, "A distance spectrum interpretation of turbo codes", IEEE Transactions on Information Theory, vol. 42, November 1996, pp. 1698-1709;

- [PYN97] R. Pyndiah, "Iterative decoding of product codes: Block turbo codes", in International Symposium on Turbo Codes and related topics, (Brest, France), September 1997, pp. 71-79;
- [ROW98] P. Robertson, T. Worz, "Bandwidth-Efficient Turbo Trellis-Coded Modulation Using Punctured Component codes", IEEE Journal on Selected Areas in Communications, vol. 16, February 1998, pp. 206-218;
- [RVH95] P. Robertson, E.Villebrun, P.Hoeher, "A Comparison of Optimal and Sub-Optimal MAP Decoding Algorithms Operating in the Log Domain", Proceedings of the International Conference on Communications, Seattle, USA, iunie 1995, pp. 1009-1013;
- [SCE01] C. Schurgers, F. Catthoor, M. Engels, "Memory Optimization of MAP Turbo Decoder Algorithms", IEEE Transactions on VLSI Systems, Vol. 9, Issue 2, April, 2001, pp. 305-312;
- [SHA48] C. E. Shannon, "A mathematical theory of communication", Bell Syst. Tech. Jour., vol. 27, Jul.1948, pp. 379-423, 623-656;
- [SKL88] B. Sklar, "Digital communications: fundamentals and applications", Prentice-Hall, Inc., Upper Saddle River, NJ, 1988;
- [STAND04] IEEE Standard for Local and Metropolitan Area Networks Part 16, "Air Interface for Fixed Broadband Wireless Access Systems", IEEE 802.16-2004;
- [STAND05] IEEE Standard for Local and Metropolitan Area Networks Part 16, "Air Interface for Fixed and Mobile Broadband Wireless Access Systems. Amendment 2: Physical and Medium Access Control Layers for Combined Fixed and Mobile Operation in Licensed Bands and Corrigendum 1. Corrigendum to 802.16-2004", IEEE 802.16e-2005;
- [TAC98] O.Y. Takeshita, D.J. Costello, "New Classes of Algebraic Interleavers for Turbo-Codes", ISIT 1998, August 16-21, USA;
- [TAS00] J. Tan, G. L. Stüber, "A MAP equivalent SOVA for non-binary turbo codes", in Proc. IEEE International Conference on Communications, ICC, New Orleans, L. A., June 2000, pp. 602-606;
- [THI93] P. Thitimajshima, "Les codes Convolutifs Récurrents Systématiques et leur application à la concaténation parallèle", Thèse, 21.12.1993;
- [TIA02] TIA/EIA/IS 2000-2-C, "Physical Layer Standard for cdma2000 Spread Spectrum Systems," May 2002;
- [TOR02] http://www.ieee802.org/16/tga/contrib/C80216a-02_80.pdf;

- [TRB06] L. Trifina, A. Rusinaru, H. Baltă, "Scaling coefficient determination of extrinsic information from the Max-Log-MAP decoding algorithm used in duobinary turbo codes", submitted on Symposium of Electronics and Telecommunications Etc 2006, 21 - 23 September 2006, Timișoara;
- [TRJ04] A. Trofimov, T. Johansson, "A memory-efficient optimal APP symbol decoding algorithm for linear block codes", IEEE Transactions on Communications, 2004, vol. 52, no 9, pp. 1429-1434;
- [TSG99] 3GPP Technical Specification Group, "Multiplexing and Channel Coding (FDD)", TS 25.212 v2.3.0, October 1999;
- [VIT67] A. J. Viterbi, "Error bounds for convolutional codes and an asymptotically optimum decoding algorithm", IEEE Transactions on Information Theory, vol. IT-13, Aprl. 1967, pp. 260-269;
- [VIT71] A. J. Viterbi, "Convolutional codes and their performance in communication systems", IEEE Trans. Commun., vol. COM-19, Oct.1971, pp. 751–772;
- [VMB02] F. Vatta, G. Montorsi, F. Babich, "Analysis and Simulation of Turbo Codes Performance over Rice Fading Channels" ICC 2002, New York, Volume: 3 , 2002 pp. 1506 -1510;
- [VOF00] J. Vogt, A. Finger, "Improving the max-log-MAP turbo decoder," Electron. Lett., vol. 36, no. 23, Nov. 2000, pp. 1937–1939;
- [VUY01] B. Vucetic, J. Yuan, "TURBO CODES Principles and Applications", The Kluwer international series in engineering and computer science, Second Printing 2001;
- [WAD00] G. Wade, "Coding Techniques-An Introduction to Compression and Error Control", Creative Print and Design, Ebbw Vale, Geat Britain, 2000;
- [WON04] Kai-Kit Wong, "Performance analysis of single and multiuser MIMO diversity channels using Nakagami-m distribution", IEEE Transactions on Wireless Communications, Volume 3, Issue 4, July 2004, pp. 1043 - 1047;
- [XIA03] W. Xiang, "Joint Source-Channel Coding for Image Transmission and Related Topics", www.library.unisa.edu.au/adt-root/uploads/approved/adt-SUSA-04112004-175228/public/02whole.pdf;
- [YAS92] Y. D. Yao, A.U.H. Sheikh, "Investigations into cochannel interference in microcellular mobile radio systems", IEEE Transactions on Vehicular Technology, vol 41, May 1992, pp. 114 - 123;
- [YUS02] Dong-Feng Yuan, Xiao-Fei Song, "Turbo Code Performance over Rayleigh Fading Channel Using QPSK Modulation", Proceedings on IEEE TENCON'02, pp. 1056-1059;

- [YUV99] J. Yuan, B. Vucetic, "*Turbo Code Performance on Rician Fading Channels*", 1999 IEEE, pp. 83-87.