

**UNIVERSITATEA "POLITEHNICA" DIN TIMIȘOARA
FACULTATEA DE MECANICĂ**

Ing. MACI CLAUDIU ADRIAN

**CONTRIBUȚII LA CONSTRUCȚIA ȘI UTILIZAREA DIGITIZOARELOR
3D ÎN PROIECTAREA ASISTATĂ DE CALCULATOR ȘI PROGRAMAREA
ROBOȚILOR INDUSTRIALI**

TEZĂ DE DOCTORAT

**Conducător științific
Prof. Dr. Ing. Dr. h. c. KOVACS FRANCISC VILIAM**

**TIMIȘOARA
2002**

Prefață

Odată cu răspândirea explozivă a utilizării calculatoarelor în toate domeniile de activitate a apărut necesitatea de a dispune de modele geometrice ale obiectelor existente fizic. Pentru culegerea unor informații despre dimensiunile și respectiv forma obiectelor se utilizează digitizarea 3D. Domeniul digitizării 3D, de mare actualitate, permite mărirea productivității într-o serie de activități ingineresti și nu numai. În lucrare, autorul a încercat să își aducă contribuția la lămurirea unor aspecte teoretice legate de digitizoarele 3D și de aplicațiile acestora.

Pentru încrederea acordată încă de la început și până în momentul finalizării lucrării, pentru sfaturile, încurajările și sprijinul acordat pe tot parcursul elaborării tezei mulțumesc distinsului meu conducător științific Prof. Dr. Ing. Dr. h. c. Kovacs Francisc.

Mulțumesc conducerii Universității "Politehnica" Timișoara, facultății de Mecanică și departamentului Mecatronică pentru condițiile acordate pregătirii în cadrul doctoratului. Un cald cuvânt de mulțumire colegilor din cadrul departamentului Mecatronică cu care am colaborat pe parcursul perioadei de pregătire la doctorat și în cadrul pregătirii tezei și în special drd. ing. Ciupe Valentin și drd. ing. Nueleanu Alina.

Aș dori de asemenea să adresez mulțumiri colegilor de la S.C. "berg COMPUTERS" S.R.L. (în special lic. inf. Adrian Pop) care prin dinamism, profesionalism și pasiunea pentru nou au reprezentat un exemplu pentru mine. Mulțumesc de asemenea conducerii firmei S.C. "berg COMPUTERS" S.R.L. pentru înțelegerea și sprijinul acordat pe parcursul elaborării tezei de doctorat.

Doresc să adresez un cuvânt de mulțumire referenților științifici ai prezentei lucrări pentru grija cu care au analizat teza și pentru sugestiile făcute în legătură cu continuarea cercetărilor în domeniu.

Mulțumesc prietenului ing. Dinu Bercea care m-a ajutat la inițierea în domeniul programării calculatoarelor și lui Victor Rubinstein care mi-a fost consultant la utilizarea aplicației 3DStudioMax pentru realizarea unor animații folosite la prezentării pe parcursul pregătirii lucrării.

Adresez calde mulțumiri ing. Mihaela Lengyel, care a fost alături de mine în prima parte a elaborării lucrării și o rog să mă ierte pentru timpul acordat tezei în detrimentul ei.

Mulțumesc familiei mele care a fost întotdeauna aproape prin încurajări, prin sfaturi și prin înțelegerea cu care am fost înconjurat.

CUPRINS

CONTRIBUȚII LA CONSTRUCȚIA ȘI UTILIZAREA DIGITIZOARELOR 3D ÎN PROIECTAREA ASISTATĂ DE CALCULATOR ȘI PROGRAMAREA ROBOȚILOR INDUSTRIALI

Prefață

Cuprins.....	2
--------------	---

Capitolul 1. Introducere

1.1. Definirea termenilor specifici utilizați în teză	5
1.2. Cum s-a ajuns la tema tezei.....	8

Capitolul 2. Stadiul actual al cercetărilor și realizărilor în domeniul construcției și utilizării digitizoarelor 3D în domeniul proiectării asistate de calculator și al programării roboților industriali

2.1. Clasificarea sistemelor de digitizare 3D după principiul de funcționare	10
2.2. Digitizoare 3D cu palpator cu acționare manuală.....	13
2.2.1. Structura digitizoarelor 3D cu palpator	13
2.2.2. Modelul geometric direct al digitizoarelor 3D cu palpator	13
2.2.3. Construcția digitizoarelor 3D cu palpator.....	17
2.2.4. Procesul de digitizare cu digitizoare 3D cu palpator, acționate manual.	20
2.2.5. Performanțele digitizoarelor 3D cu palpator	23
2.2.6. Aplicații ale digitizoarelor 3D cu palpator	27
2.2.7. Mărirea spațiului de lucru al digitizoarelor 3D	32
2.3. Sisteme soft pentru digitizarea 3D	34
2.4. Tendințe actuale în construcția și utilizarea digitizoarelor 3D.....	40

Capitolul 3. Modelarea suprafețelor

3.1. Introducere	43
3.2. Curbe de formă liberă.....	43
3.3. Obținerea de suprafețe pornind de la curbe	44

Capitolul 4. Stabilirea obiectivelor tezei de doctorat

4.1. Digitizorul 3D, o noutate tehnică cu multiple aplicații	49
4.2. Stabilirea relațiilor dintre construcția robotului și construcția digitizorului...49	
4.3. Contribuții la fundamentarea modalităților de utilizare a digitizoarelor 3D în proiectarea asistată de calculator	49
4.4. Contribuții la fundamentarea modalităților utilizării digitizoarelor 3D în programarea roboților	50
4.5. Contribuții la fundamentarea utilizării digitizoarelor 3D în alte domenii tehnice sau mixte	50
4.6 Verificarea celor prezentate anterior prin aplicații concrete	51

Capitolul 5. Contribuții teoretice

5.1. Determinarea erorilor de măsurare a coordonatelor punctelor prelevate cu digitizorul 3D cu palpator	52
5.1.1. Modelarea preciziei digitizoarelor 3D	52
5.1.2. Determinarea volumului de toleranță	52
5.1.3. Elaborarea hărții de precizie	56
5.1.4. Determinarea frontierei convexe a unui nor de puncte	57
5.1.5. Calibrarea digitizorului cu palpator cu acționare manuală	58
5.2. Estimarea rezoluției necesare a traductoarelor incrementale de rotație pentru atingerea unei precizii impuse de măsurare a coordonatelor punctelor digitizate..	62
5.3. Calibrarea de ordinul zero. Determinarea matricii de transformare între două sisteme de referință pe baza coordonatelor a trei puncte exprimate în cele două sisteme de referință	64
5.4. Modelarea suprafețelor pornind de la informații despre coordonatele unor șiruri de puncte 3D obținute prin digitizare.....	67
5.4.1. Introducere	67
5.4.2. Descrierea algoritmilor utilizați.	67

Capitolul 6. Realizări constructive ale unor digitizoare 3D

6.1. Introducere	75
6.2. Realizarea constructivă a unui digitizor RRR cu palpator	76
6.2.1. Modelul matematic al digitizorului 3D	76
6.2.2. Subsistemul mecanic	77
6.2.3. Subsistemul electronic	80
6.2.4. Subsistemul informatic	81
6.2.5. Încercări experimentale	84
6.3. Proiectarea constructivă a unui digitizor cu fire	87
6.3.1. Prezentare generală	87
6.3.2. Modelul matematic al digitizorului 3D cu fire	88
6.3.3. Proiectarea părților componente ale sistemului	97

Capitolul 7. Contribuții la realizarea unor aplicații practice ale digitizoarelor 3D

7.1. Introducere	104
7.2 RobSim - aplicație pentru programarea off-line a roboților în mediu virtual cu interfață pentru digitizoare 3D.....	105
7.2.1. Prezentarea generală a aplicației RobSim.....	105
7.2.2. Construirea modelului programabil al robotului pornind de la modelul geometric 3D.....	106
7.2.3. Programarea robotului virtual și desfășurarea sarcinii de lucru.....	109
7.2.4. Interfața robotului virtual cu robotul real	115
7.2.5. Utilitatea interfațării RobSim cu digitizoare 3D	124
7.2.6. Calibrarea robot-digitizor 3D	124
7.2.7. Precizarea situațiilor efectorului final cu digitizorul 3D	126

7.2.8. Realizarea modelului geometric al mediului robotului cu dig. 3D	129
7.2.9. Teleoperare cu digitizorul 3D	129
7.3. Generarea modelului geometric pe baza informațiilor prelevate cu digitizorul 3D.....	131
7.4. Proiectarea asistată de calculator a mobilierului de ședere în industria mobilei pe baza datelor obținute prin digitizare (aplicație).....	135
7.4.1 Prezentare generală	133
7.4.2 Tehnologia de grup pentru fabricarea pieselor de mobilier	136
7.4.3. Prezentarea algoritmilor pentru realizarea modelelor geometrice ale unor tipuri de piese pornind de la șiruri de puncte obținute prin digitizare	139
7.4.4. Generarea automată a desenelor de execuție a șabloanelor și a sculelor pentru realizarea pieselor de revoluție profilate pornind de la date obținute prin digitizare	156
7.5. Sculptură virtuală cu digitizorul 3D	161
7.5.1. Introducere	161
7.5.2. Generarea semifabricatelor inițiale	161
7.5.3. Extragerea de material cu ajutorul sculei virtuale condusă cu digitizorul 3D.....	167
7.5.4. Facilități auxiliare ale aplicației de sculptură virtuală	172
7.5.5. Direcții posibile de dezvoltare a aplicației	173
7.6. Compararea unui obiect existent fizic cu modelul geometric “ideal” al acestuia utilizând un digitizor 3D cu palpator	174
7.6.1. Calibrarea pentru “suprapunerea” obiectelor	174
7.6.2. Compararea geometriei modelului existent fizic cu modelul geometric	174

Capitolul 8. Contribuții personale. Concluzii finale

8.1. Contribuții personale.....	178
8.1.1. Introducere	178
8.1.2. Contribuții în plan teoretic	178
8.1.3. Contribuții practice	179
8.1.4. Contribuții la realizarea unor aplicații ale digitizoarelor 3D	179
8.1.5. Direcții de cercetare viitoare	180
8.2. Concluzii finale	180

Bibliografie	181
---------------------------	-----

Anexe

Anexa 1. Modele de digitizoare 3D si firme producătoare.....	186
Anexa 2. Cod sursă pentru generarea suprafețelor pornind de la șiruri de puncte	192
Anexa 3. Încercări experimentale ale digitizorului 3D RRR	201
Anexa 4. Arhitectura aplicației RobSim	203
Anexa 5. Cod sursă din cadrul aplicației DigiCAD	209
Anexa 6 Elemente de geometrie analitică utilizate în cadrul aplicației DigiMob..	219
Anexa 7. Descrierea aplicației DigiMob și cod sursă din aplicație	222

1 Introducere

1.1 Definirea termenilor specifici utilizați în teză

O caracteristică pregnantă a societății actuale este informatizarea accentuată a tuturor sectoarelor de activitate. În industrie, datorită concurenței determinate de faptul că oferta depășește cererea, rezistă doar organizațiile care reușesc să se adapteze rapid la cerințele clienților și care își prezintă produsele într-un mod atrăgător. Grafica 3D deține un loc important atât în inginerie prin aportul adus la realizarea produselor cât și în activitățile comerciale.

La realizarea fizică a produselor, factorul principal care asigură avantajul competitiv este flexibilitatea asigurată prin utilizarea sistemelor de fabricație flexibilă servite de roboți industriali. În măsura în care complexitatea produselor crește iar mărimea seriilor de fabricație pentru produse scad, timpul ocupat de proiectarea produsului și realizarea prototipului pentru validarea proiectului are o pondere tot mai însemnată în timpul total de realizare a produsului. De aceea, orice îmbunătățire care duce la creșterea eficienței proceselor situate în aval de realizarea fizică a produsului în condiții de serie are o contribuție definitorie la scăderea costurilor produselor și a succesului pe piață. Pentru a crește eficiența acestor procese s-au impus o serie de tehnici și concepte : prototip digital, realizarea rapidă a prototipurilor (Rapid Prototyping) , execuția rapidă de scule (Rapid Tooling), ingineria inversă, digitizarea 3D (3D digitizing), renderingul, animația, realitatea virtuală.

Tendința actuală este de a transfera cât mai multe procese din faza de proiectare și realizare a prototipului în domeniul virtual. Proiectul produsului se realizează utilizând un software CAD (de regulă 3D) interfațat cu sisteme de baze de date, iar utilizarea sistemelor Intranet și Internet permite lucrul în echipă a specialiștilor situați în diverse arii geografice. Chiar și în aceste condiții, oricât de performantă ar fi echipa care a participat la proiectare , în drumul către omologare produsul trebuie să facă dovada calităților sale. Pentru eliminarea ciclului lung și costisitor pe care îl impune realizarea mai multor variante și testarea reală a acestora, precum și pentru punerea în evidență a erorilor conceptuale, *prototipul digital (Digital Mock-up)* este analizat cu ajutorul unor programe specializate, sub o multitudine de aspecte pornind de la analize statice și mergând până la analize de comportare la ciocniri. Prin rularea unor aplicații de optimizare, proiectul produsului poate fi îmbunătățit pe baza criteriilor impuse ceea ce duce la reducerea la minim a necesității de modificări ulterioare în faza de fabricație.

Pentru realizarea unui produs novator, proiectantul care definește forma produsului nu trebuie constrâns de factorul tehnologic pentru a-și putea manifesta pe deplin imaginația creatoare. Implicarea echipelor de tehnologi în proiect cât mai devreme permite realizarea modificărilor necesare pentru ca produsul să poată fi obținut la un cost de fabricație acceptabil. Realizarea modelului produsului în format digital permite suprapunerea parțială a unor faze și actualizarea în timp real a informațiilor despre modificări, ceea ce duce la scăderea duratei proiectării.

O altă tehnologie revoluționară care permite scăderea intervalului de timp de la idee la produs este "*Rapid Prototyping*-ul". Sistemele de realizare rapidă a prototipului fizic utilizează tehnologii ce presupun polimerizarea sub acțiunea unui fascicol laser a unei rășini sau pulberi fotosensibile, depunerea coordonată prin calculator a unor materiale termoplastice sau stratificarea unor pachete de hârtie decupate cu laser pe contur. Procesul de execuție a prototipului fizic are ca date de intrare prototipul digital al produsului. Prototipul fizic rezultat poate fi utilizat pentru verificarea modului în care piesa proiectată corespunde unor cerințe.

Când calitatea furnizată de Rapid Prototyping nu este satisfăcătoare, gradul de finisare a suprafeței trebuind să fie net superior, soluția impune utilizarea tehnologiei "*Rapid Tooling*". În sens larg "*Rapid Tooling*" definește orice tehnologie care permite realizarea rapidă de scule. Scopul acestei tehnologii este realizarea unei pregătiri de fabricație (spre exemplu matrițe) ieftine pentru o serie foarte scurtă. O metodă de obținere a unei matrițe prin această tehnologie pornește de la realizarea unei piese dintr-un material cu o duritate scăzută pe o mașină de frezat CNC care primește date direct de la sistemul CAD. Se continuă cu definirea unui strat de separație și depunerea unui strat de latex special. După polimerizare, latexul se demulează și se depune pe un pat de gips care să îi asigure rigiditatea iar zona interioară se metalizează printr-o metodă specială. Cu matrița astfel realizată se poate obține într-un timp foarte scurt o serie mică de piese de o calitate superioară.

Ingineria inversă ("Reverse Engineering") este denumirea dată procesului de realizare a proiectului unui produs pornind de la repere deja existente. Are scopul de reutilizare (modificare, îmbunătățire) a acelor componente în activitatea de proiectare. Ingineria inversă este frecvent utilizată deoarece numeroase produse noi se bazează pe produse existente. În cazul în care se dorește reproducerea unui model existent fizic este necesară în prima fază obținerea modelului digital al acestuia. Când suprafața este compusă din primitive geometrice este utilizată măsurarea elementelor (puncte, muchii) caracteristice cu o mașină de măsurat în coordonate. Când suprafețele obiectului sunt de formă liberă și este necesară descrierea precisă a suprafeței este utilizată digitizarea 3D.

Digitizarea 3D este procesul de obținere a unui model CAD prin prelucrarea informațiilor referitoare la forma și dimensiunile suprafețelor unui obiect existent fizic. Deci prin digitizarea 3D se realizează procesul invers față de cel implicat în fabricația asistată de calculator. Digitizarea 3D se realizează cu două componente majore: digitizoarele 3D și sistemele software care "construiesc" suprafața în sistemul CAD din datele obținute cu digitizoarele 3D.

Echipamentele utilizate pentru măsurarea coordonatelor punctelor necesare pentru suficienta descriere a suprafeței se numesc *digitizoare 3D*. Informațiile de ieșire a digitizoarelor 3D sunt coordonatele unor puncte individuale sau a unor șiruri de puncte situate pe suprafața obiectului. Viteza de digitizare este aria suprafeței digitizate în unitatea de timp.

Sistemele software utilizate pentru construirea suprafețelor CAD din datele punct obținute în urma digitizării sunt bazate pe tehnici de generare a suprafețelor. În timpul procesului de digitizare, din punctele obținute se generează polilinii care sunt

afișate pe monitor pentru asigurarea feed-back-ului vizual. Poliliniile sunt entități geometrice formate din linii și arce ca subelemente iar punctul comun a două subentități succesive se numește vertex. Digitizarea 3D este un domeniu multidisciplinar la constituirea căruia concură: Mecanica, Robotica, Informatica, Teoria mecanismelor, Electronica, Știința calculatoarelor, Statistica (Fig. 1.1.1)

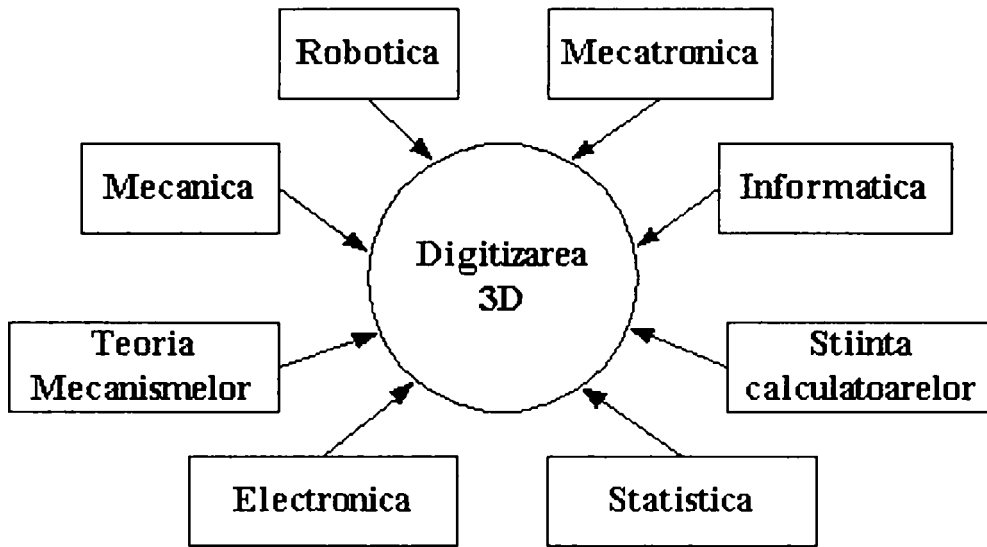


Fig. 1.1.1 Digitizarea 3D, domeniu de știință multidisciplinar

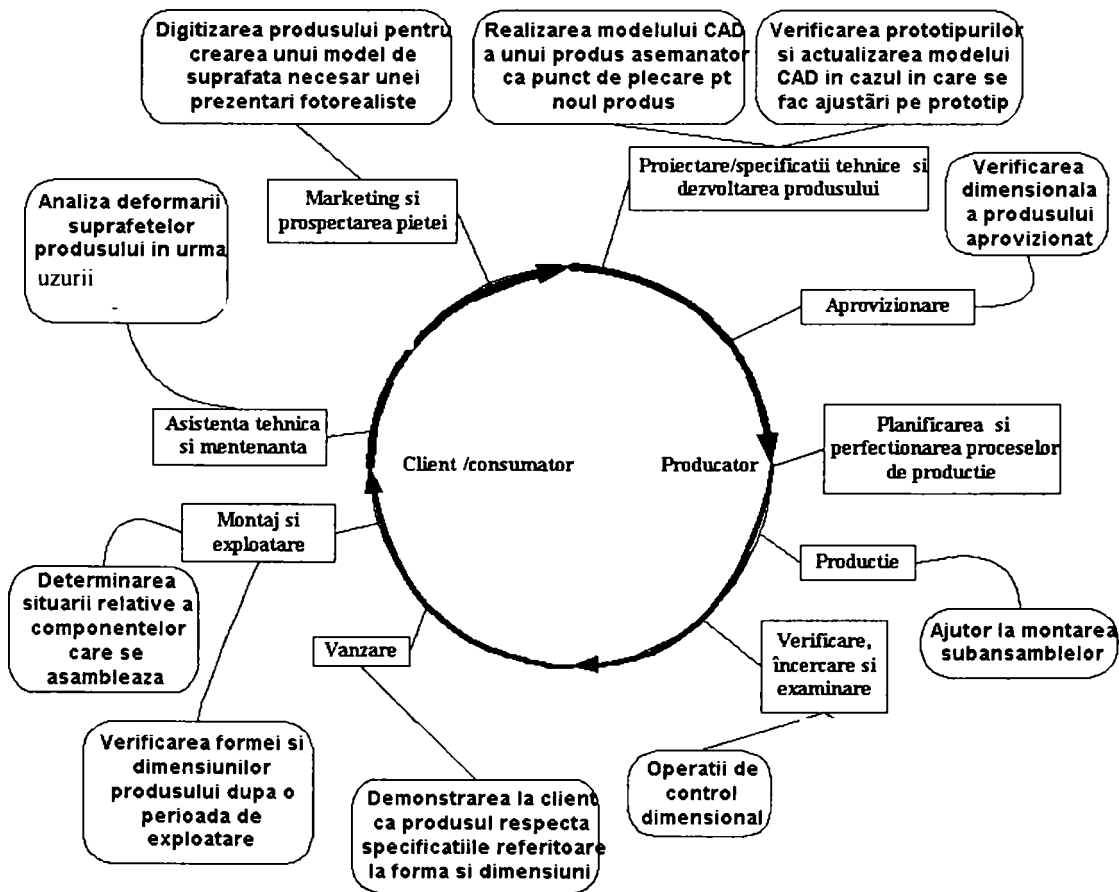


Fig.1.1.2 Utilizarea digitizoarelor 3D pe parcursul ciclului de viață al produsului

Utilizarea digitizoarelor 3D este benefică în toate fazele ciclului de viață al produsului, începând cu ajutorul dat în realizarea modelului și continuând cu verificarea produsului în timpul procesului de fabricație, la testele finale precum și în operațiile de mentenanță (fig. 1.1.2).

Pentru vizualizarea formei produselor și a comportării lor se utilizează tehnici de rendering și respectiv animații. Ambele tehnici pot utiliza modele realizate prin digitizare.

Renderingul permite vizualizarea fotorealistică a formei unui model CAD prin utilizarea unei palete largi de tehnici de umbrire utilizând diferite tipuri de iluminări a modelului. Abilitatea de a comunica vizual utilizând computerul, bazat pe grafica 3D are o mare importanță la ora actuală. Ca mijloc de comunicare, un model fotorealistic 3D poate influența aprobarea produsului de către client, poate facilita discuțiile și luarea de decizii pe durata proiectării și poate fi folosit în locul fotografiilor în cadrul cataloagelor produselor și în cazul activităților de marketing desfășurate înainte de fabricarea propriu-zisă a produselor.

Pentru prezentarea comportamentului modelului CAD este posibilă utilizarea animației. *Animația* permite translatarea și rotirea modelelor astfel încât se poate ilustra comportarea modelului. La fel ca și renderingul, animația este de mare efect pentru demonstrarea calităților unui proiect.

Odată cu creșterea puterii de calcul, un nou domeniu se impune într-o serie de aplicații : realitatea virtuală. În aplicațiile de realitatea virtuală, operatorul uman este transpus într-un mediu sintetic, generat cu ajutorul calculatorului. Pentru determinarea poziției capului observatorului în vederea generării unei imagini corespunzătoare unghiului de privire al acestuia se utilizează structuri asemănătoare cu cele ale digitizoarelor 3D cu palpator.

Autorul își propune în teză să aducă contribuții la sistematizarea cunoștințelor în domeniul digitizării 3D și își îndreaptă atenția în mod special asupra digitizoarelor 3D cu palpator fără acționare.

1.2 Cum s-a ajuns la tema tezei

Tema tezei a rezultat din intersecția dintre preocupările profesionale ale autorului și pasiunea pentru CAD și tehnică cu pasiunea conducătorului științific, distinsul Prof. Dr. Ing. Dr. h c. Kovacs Francisc pentru ultimele noutăți în domeniul tehnicii. Totodată, teza este o continuare a dezvoltării temei abordate în lucrarea de disertație susținută ca finalizare a studiilor aprofundate în specializarea "Sisteme de fabricație flexibilă" în cadrul Universității "Aurel Vlaicu" din Arad. Autorul, de specialitate T.C.M., era în anul 1998 angajat la Fabrica de mobilă "Pâncotas S.A." ca inginer proiectant. În 1999 lucrează ca șef al atelierului de proiectare al aceleiași întreprinderi, situație în care este pus în fața unor numeroase probleme legate de proiectarea mobilei. Dintre problemele cu care autorul a intrat în contact pot fi amintite: necesitatea realizării desenelor de execuție a unor modele oferite de clienți în vederea producerii lor în întreprindere, modificarea unor modele existente și

realizarea desenelor de execuție ale acestora, verificarea dacă modelul rezultat în procesul de fabricație este suficient de apropiat din punct de vedere dimensional cu modelul oferit de client.

Cursurile urmate de autor la disciplinele “Roboți industriali” și “Sisteme de fabricație flexibilă”, au dus la cristalizarea ideilor în legătură cu soluția la problemele întâlnite în activitatea desfășurată la fabrica de mobilă. Soluția era un dispozitiv asemănător cu mecanismul generator de traiectorie al roboților, dar fără acționare. Prin înlocuirea efectorului final cu un palpator a fost posibilă obținerea coordonatelor 3D ale punctului caracteristic (vârful palpatorului) al echipamentului. Prin conducerea ultimului element al mecanismului de către operator astfel încât vârful palpatorului să vină în contact cu suprafața era posibilă obținerea coordonatelor punctelor de pe suprafață, iar printr-un soft adecvat se putea genera suprafața. Echipamentul care permite descrierea suprafeței prin măsurarea punctelor de pe această suprafață este digitizorul 3D.

Domnul Prof. Dr. Ing. Kovacs Francisc a oferit autorului ocazia pregătirii în domeniul roboticii și a digitizoarelor 3D prin acceptarea conducerii tezei având tema actuală.

2 Stadiul actual al cercetărilor și realizărilor în domeniul construcției și utilizării digitizoarelor 3D în domeniul proiectării asistate de calculator și al programării roboților industriali

2.1 Clasificarea sistemelor de digitizare 3D în funcție de principiul de funcționare

Digitizoarele 3D se împart în două mari categorii: digitizoare 3D care necesită contactul unui element al digitizorului cu suprafața obiectului digitizat și digitizoare fără contact. În figura 2.1.1 este prezentată o clasificare a digitizoarelor 3D.

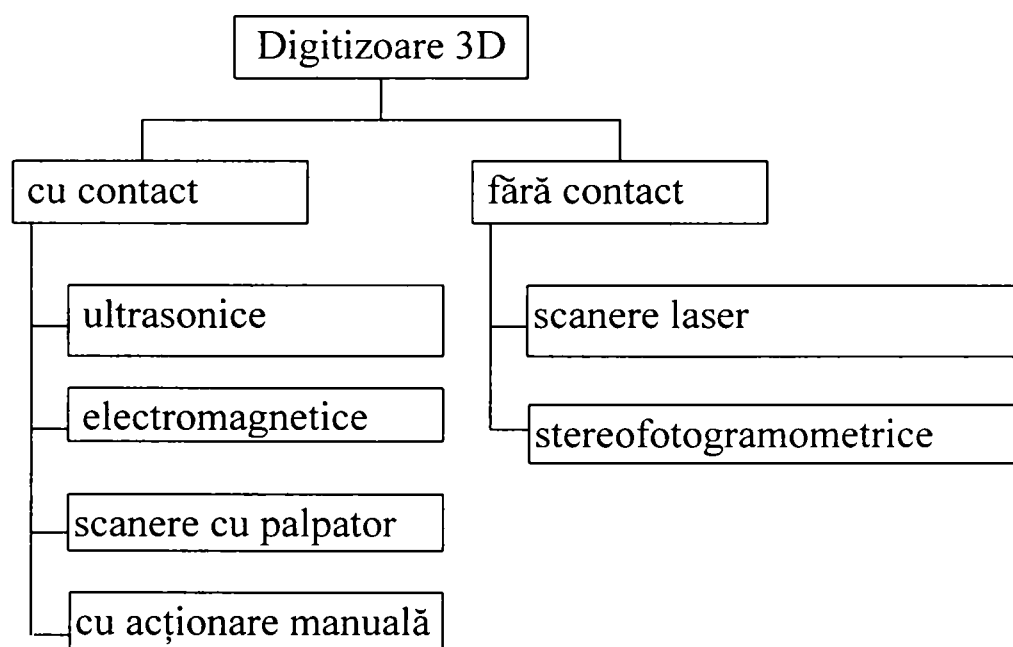


Fig. 2.1.1 Clasificarea digitizoarelor 3D

Digitizoarele 3D care au la bază sisteme ultrasonice au ca principiu de funcționare emiterea unor impulsuri ultrasonice de către un emițător situat într-un mâner și recepționarea acestor impulsuri cu ajutorul unor microfoane. După convertirea energiei acustice în energie electrică cu ajutorul microfoanelor, un sistem cu microcontroller este utilizat pentru prelucrarea spațială în vederea obținerii imaginii pornind de la distribuția câmpului de undă. Digitizoarele realizate pe acest principiu au o precizie scăzută datorită lungimii mari a undei sonore și a suprapunerii undelor reflectate de diferite puncte ale obiectului. Datorită acestui inconvenient, digitizoarele bazate pe sisteme ultrasonice sunt utilizate tot mai rar în aplicații care presupun realizarea modelelor de suprafață a obiectelor existente fizic și sunt folosite preponderent în aplicații de realitate virtuală.

Digitizoarele bazate pe sistemele electromagnetice stabilesc poziția punctului caracteristic prin măsurarea câmpului magnetic produs de o bobină încorporată într-un

mâner având forma unui creion. Alimentarea bobinei se poate realiza cu curent alternativ, caz în care are loc o schimbare a sensului câmpului magnetic sau se poate realiza în impulsuri de curent continuu. Intensitatea câmpului magnetic măsurată de receptori depinde de orientarea mânerului (bobinei) față de receptor, de forma liniilor de câmp și distanța emițătorului față de receptor. Mânerul are un vârf palpator (punctul caracteristic) care este în contact cu suprafața obiectului ce urmează a fi digitizat. În figura 2.1.2 este prezentat un digitizor bazat pe sistem electromagnetic (stânga) și mânerul care conține emițătorul (dreapta).

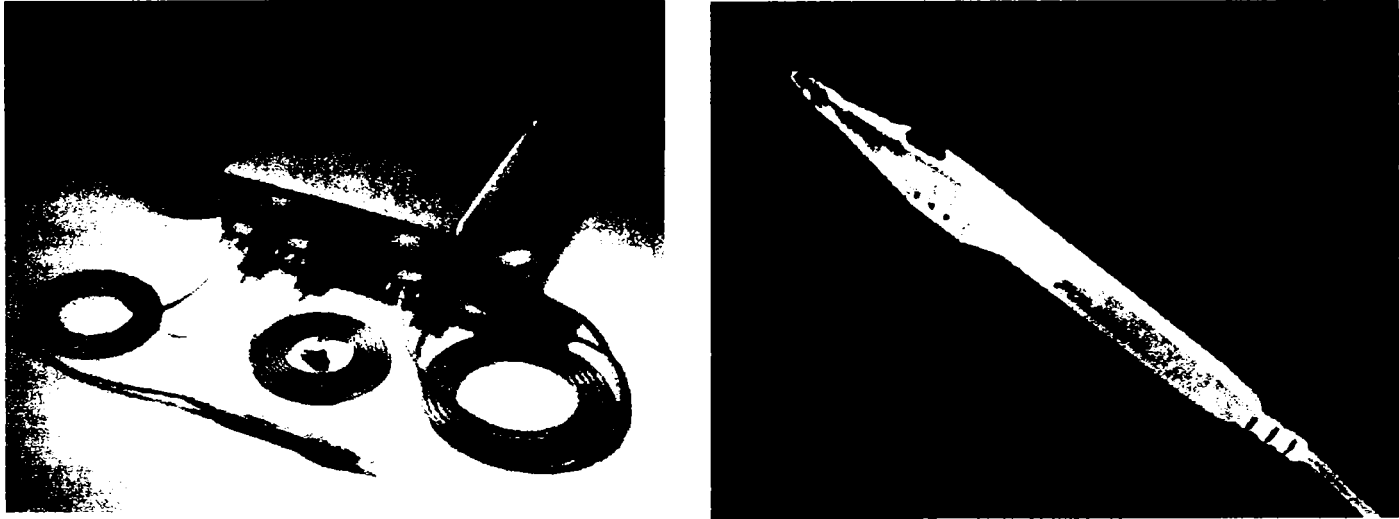


Fig.2.1.2 Digitizorul Fastrack bazat pe sistem electromagnetic

Semnalele analogice provenite de la receptori sunt convertite în semnale digitale și prelucrate cu ajutorul calculatorului. În urma analizei informațiilor provenite de la mai mulți receptori, este stabilită poziția emițătorului și implicit a punctului de pe suprafața obiectului. Datorită principiului de funcționare, digitizoarele bazate pe sistemele electromagnetice pot fi utilizate doar pentru preluarea informațiilor despre forma obiectelor nemetalice. Prezența obiectelor metalice în apropierea sistemului îi poate afecta funcționarea. Deoarece este folosit câmpul magnetic pentru stabilirea situației stiletului, linia de vizare directă între emițători și receptori nu este necesară. Latența în determinarea situației mânerului este de ordinul milisecundelor ceea ce face ca acest tip de sisteme de digitizare să fie preponderent utilizate în aplicații de realitate virtuală.

Digitizoarele care lucrează prin metoda de scanare cu palpator (“probe scanning”) au o structură asemănătoare cu mașinile de măsurare în coordonate dar sunt optimizate din punct de vedere constructiv și al sistemelor software pentru preluarea coordonatelor unor puncte dispuse matricial pe suprafață. Deplasarea palpatorului este realizată cu un pas constant pe două direcții perpendiculare, iar coordonatele punctului suprafeței măsurate sunt înregistrate la contactul dintre vârful palpatorului și suprafață în timpul deplasării pe o direcție perpendiculară pe primele două. Coordonatele 3D ale punctului caracteristic al capului de măsurare al digitizorului sunt înregistrate cu un increment regulat ales de către operator în funcție de precizia de digitizare impusă. Viteza de digitizare atinge 100 puncte/secundă. Pe baza același metode de scanare cu palpator, este posibilă utilizarea pentru digitizarea

suprafețelor a mașinilor de frezat cu comandă numerică, având montate în locul frezei un senzor tactil, dar această variantă asigură o productivitate mult mai scăzută a digitizării.

Digitizoarele 3D cu palpator fără acționare (“digitizing arms”) utilizează pentru aducerea în contact a vârfului palpatorului cu suprafața a cărui model geometric se urmărește a fi realizată deplasarea ultimului element al digitizorului de către operatorul uman. Palpatorul este un senzor de atingere și are un vârf prin care intră în contact cu suprafața care se măsoară.

Scanarea 3D este procedeul de obținere a coordonatelor punctelor situate pe suprafața exterioară a unui obiectelor prin utilizarea triangulației geometrice. Pentru scanarea laser 3D se proiectează un fascicol luminos plan pe obiecte, dintr-o anumită direcție, iar un senzor video detectează imaginea proiecției fascicolului pe obiect dintr-o altă direcție. Pentru scanarea întregului obiect este necesară asigurarea deplasării proiecției fascicolului luminos pe suprafața obiectului. Aceasta poate fi realizată prin deplasarea obiectului în raport cu sistemul de scanare fix, prin deplasarea elementului mobil al sistemului de scanare față de obiectul fix, sau prin deplasarea ambelor. Elementul principal al sistemului de scanare este capul de scanare care conține o sursă de radiație laser în centru și doi senzori CCD situați de o parte și de alta a sursei laser (Fig. 2.1.3). Cu ajutorul unor lentile cilindrice se formează un fascicol luminos plan. Lumina reflectată de obiectul digitizat, care formează un profil pe suprafața obiectului activează pixelii matricei unuia din cei doi senzori situați în capul de scanare. În figura 2.1.4 este prezentat un sistem de scanare laser utilizat pentru digitizarea unor piese de mici dimensiuni.

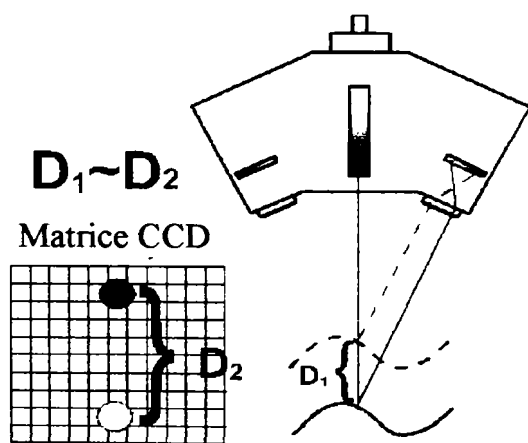


Fig. 2.1.3 Cap de scanare cu doi senzori



Fig. 2.1.4 Echipament de scanare laser

Metoda de digitizare bazată pe stereofotogrammetrie utilizează un principiu asemănător vederii binoculare pentru a extrage informații referitoare la poziția punctului de pe suprafața măsurată.

Există o serie de sisteme de digitizare care nu pot fi încadrate riguros în una din categoriile menționate în clasificarea din figura 2.1.1. Aceste sisteme utilizează combinarea a mai multor principii de funcționare pentru determinarea coordonatelor punctelor situate pe suprafața obiectelor digitizate. De exemplu există sisteme de digitizare care utilizează câmpul electromagnetic în combinație cu scanarea laser sau

sisteme care combină determinarea situației prin utilizarea surselor laser și a digitizoarelor cu palpator fără acționare.

2.2 Digitizoare 3D cu palpator cu acționare manuală

2.2.1 Structura digitizoarelor 3D cu palpator

Digitizoarele 3D cu palpator cu acționare manuală utilizează în majoritatea cazurilor mecanisme cu cuple cinematice de rotație. Alegerea cuplelor cinematice de rotație este datorată în primul rând faptului că permite cu ușurință manevrarea manuală. În unele cazuri, pentru mărirea spațiului de lucru se utilizează o axă de translație lungă. În continuare, digitizoarele 3D cu palpator cu acționare manuală vor fi numite pe scurt digitizoare 3D cu palpator.

Un obiect liber în spațiu are 6 grade de libertate. Deci, pentru ca palpatorul digitizorului să poată ocupa o situație oarecare în spațiu, digitizorul 3D cu palpator trebuie să fie un sistem mecanic mobil cu 6 grade de libertate. Se cunoaște din robotică că în cazul unui lanț cinematic deschis cu cuple cinematice de clasa a V-a gradul de mobilitate al unui lanț cinematic este egal cu numărul cuplelor cinematice sau mai mic cu o unitate ca numărul elementelor lanțului cinematic:

$$M = c_5 = n - 1 \quad (2.2.1)$$

Utilizarea preponderentă a digitizoarelor 3D este la determinarea coordonatelor punctelor de pe suprafețele exterioare ale obiectelor. Pentru ca vârful palpatorului digitizorului 3D să poată fi adus într-un punct oarecare din spațiul de lucru, sistemul mecanic care conduce palpatorul trebuie să posede trei grade de mobilitate. Așa cum se observă din relația 2.2.1, această condiție este îndeplinită de un mecanism cu trei cuple cinematice de clasa a V-a. Acesta este numărul minim de cuple cinematice necesare pentru un digitizor 3D. Pentru ca vârful palpatorului să poată ajunge în contact de pe suprafața obiectului în orice situație relativă sunt necesare mai multe grade de mobilitate, deci mai multe cuple cinematice de clasa a V-a. Numărul uzual de cuple cinematice al digitizoarelor 3D este situat între 5 și 7.

Aducerea vârfului digitizorului în contact cu un punct de pe suprafața obiectului se realizează prin manevrarea de către operator a elementelor mobile ale digitizorului. Prin impunerea situației ultimului element al digitizorului de care este fixat rigid palpatorul sunt anulate toate cele 6 grade de libertate a acestuia, deci mecanismul devine desmodrom. În cazul în care digitizorul are mai mult de 6 cuple cinematice, acesta posedă grade de manevrabilitate. Numărul de grade de manevrabilitate este: $M_a = M - 6$. În acest caz este necesar ca operatorul să acționeze asupra unui al doilea element al lanțului cinematic al digitizorului prin anularea a M_a grade de libertate a acestuia pentru a nu permite deplasarea necontrolată a elementelor digitizorului.

2.2.2 Modelul geometric direct al digitizoarelor 3D cu palpator

Determinarea coordonatelor punctului caracteristic al digitizorului se realizează pe baza modelului geometric direct. În continuare se prezintă pentru exemplificare

modelul geometric direct pentru determinarea coordonatelor punctului caracteristic în cazul modelului de digitizor cu palpator Romer 1000 (fig. 2.2.1) produs de firma Romer INC.

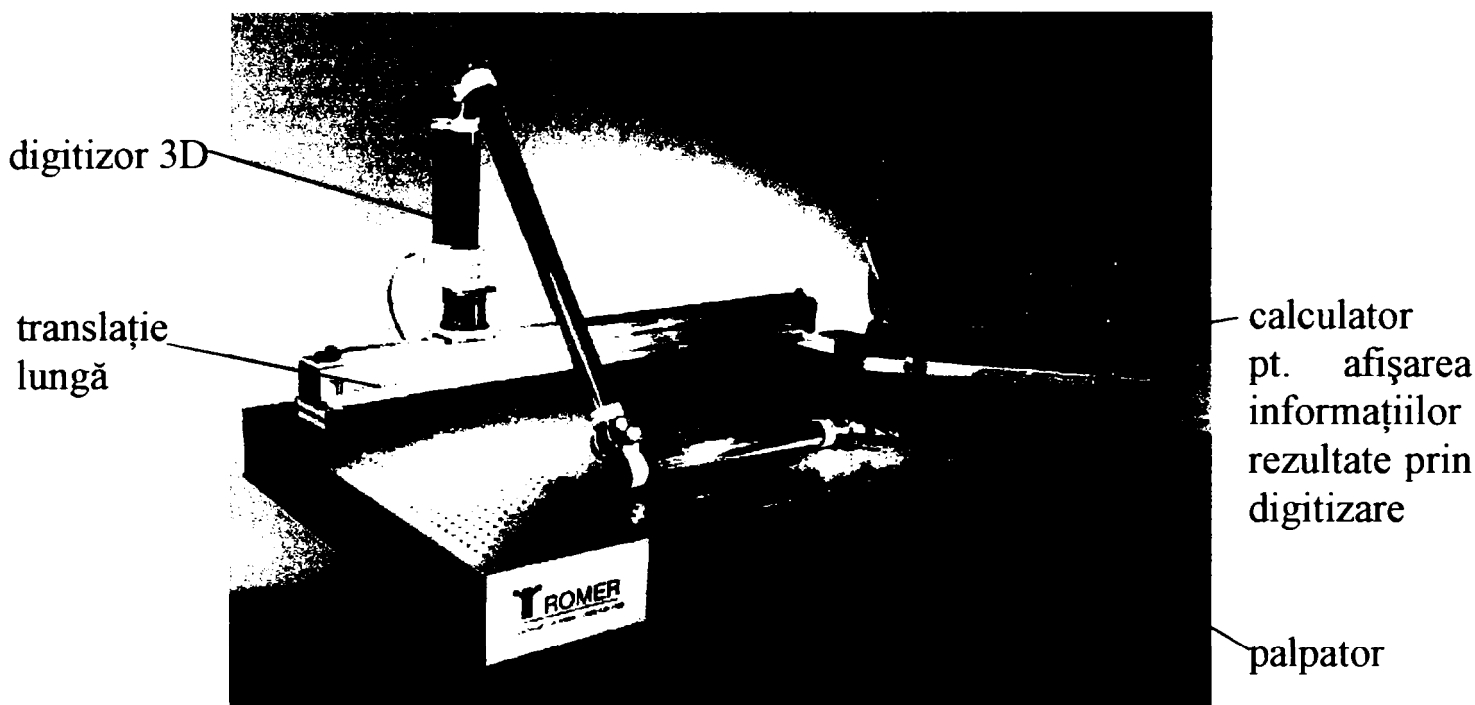


Fig. 2.2.1 Digitizorul Romer 1000

Schema cinematică a digitizorului este prezentată în figura 2.2.2 . Modelul geometric al mecanismului se exprimă prin matricea transformării omogene:

$${}^0T_6 = {}^0A_1 \cdot {}^1A_2 \cdot {}^2A_3 \cdot {}^3A_4 \cdot {}^4A_5 \cdot {}^5A_6 \quad (2.2.2)$$

$$\begin{aligned} {}^0A_1 &= (\text{rot } z_0, \theta_1) \cdot (\text{trans } z_0, d_1) \cdot (\text{rot } x_1, \pi/2) \\ {}^1A_2 &= (\text{rot } z_1, \theta_2) \cdot (\text{trans } y_1, d_2) \cdot (\text{rot } y_1, \pi/2) \cdot (\text{rot } z_1, \pi/2) \\ {}^2A_3 &= (\text{rot } z_2, \theta_3) \cdot (\text{trans } z_2, d_3) \cdot (\text{rot } y_2, -\pi/2) \cdot (\text{rot } x_2, \pi/2) \\ {}^3A_4 &= (\text{rot } z_3, \theta_4) \cdot (\text{trans } x_3, d_4) \cdot (\text{rot } x_3, -\pi/2) \\ {}^4A_5 &= (\text{rot } z_4, \theta_5) \cdot (\text{trans } z_4, d_5) \cdot (\text{rot } x_4, \pi/2) \\ {}^5A_6 &= (\text{rot } z_5, \theta_6) \cdot (\text{trans } y_5, d_6) \cdot (\text{rot } x_5, \pi/2) \end{aligned}$$

Se notează cu $s_i = \sin\theta_i$ și $c_i = \cos\theta_i$ unde $i = 1, 2, \dots, 6$

$${}^0A_1 = \begin{bmatrix} c_1 & -s_1 & 0 & 0 \\ s_1 & c_1 & 0 & 0 \\ 0 & 0 & 1 & d_1 \\ 0 & 0 & 0 & 1 \end{bmatrix} * \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} c_1 & 0 & s_1 & 0 \\ s_1 & 0 & -c_1 & 0 \\ 0 & 1 & 0 & d_1 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

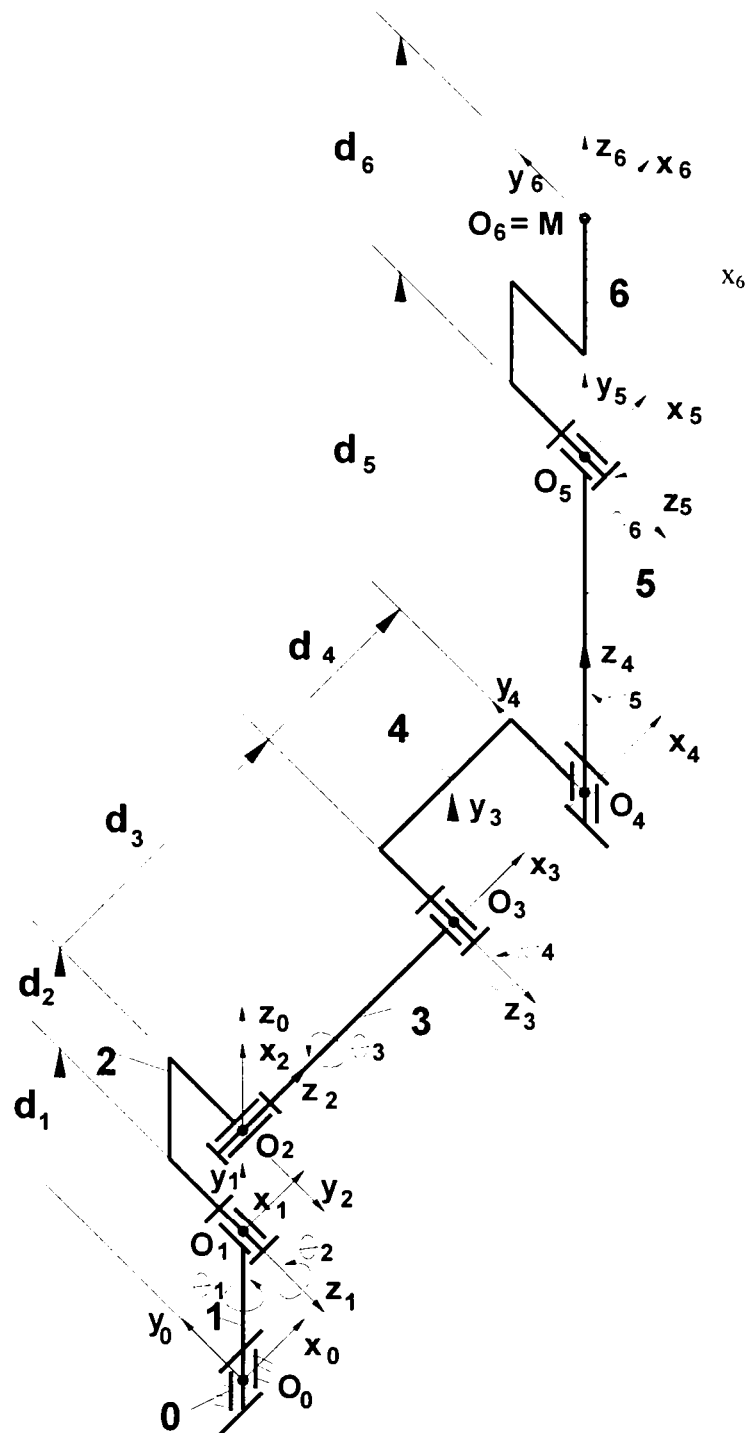


Fig. 2.2.2. Schema cinematică a digitizorului Romer 1000

$${}^1\underline{A}_2 = \begin{bmatrix} c_2 & -s_2 & 0 & 0 \\ s_2 & c_2 & 0 & d_2 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} * \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} * \begin{bmatrix} 0 & -1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} -s_2 & 0 & c_2 & 0 \\ c_2 & 0 & s_2 & d_2 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$${}^2\underline{A}_3 = \begin{bmatrix} c_3 & -s_3 & 0 & 0 \\ s_3 & c_3 & 0 & 0 \\ 0 & 0 & 1 & d_3 \\ 0 & 0 & 0 & 1 \end{bmatrix} * \begin{bmatrix} 0 & 0 & -1 & 0 \\ 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} * \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 0 & c_3 & -s_3 & 0 \\ 0 & s_3 & c_3 & 0 \\ 1 & 0 & 0 & d_3 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$${}^3\underline{A}_4 = \begin{bmatrix} c_4 & -s_4 & 0 & d_4 \\ s_4 & c_4 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} * \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} c_4 & 0 & -s_4 & d_4 \\ s_4 & 0 & c_4 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$${}^4\underline{A}_5 = \begin{bmatrix} c_5 & -s_5 & 0 & 0 \\ s_5 & c_5 & 0 & 0 \\ 0 & 0 & 1 & d_5 \\ 0 & 0 & 0 & 1 \end{bmatrix} * \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} c_5 & 0 & s_5 & 0 \\ s_5 & 0 & -c_5 & 0 \\ 0 & 1 & 0 & d_5 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$${}^5\underline{A}_6 = \begin{bmatrix} c_6 & -s_6 & 0 & 0 \\ s_6 & c_6 & 0 & d_6 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} * \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} c_6 & 0 & -s_6 & 0 \\ s_6 & 0 & c_6 & d_6 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Prin înlocuirea valorilor matricilor de transformare ${}^0A_1, \dots, {}^5A_6$ în relația 2.2.2 se obține matricea transformării omogene a mecanismului :

$${}^0T_6 = \begin{bmatrix} n_x & o_x & a_x & p_x \\ n_y & o_y & a_y & p_y \\ n_z & o_z & a_z & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} \underline{R} & \underline{P} \\ \underline{0} & 1 \end{bmatrix} \text{ unde :}$$

\underline{R} – submatricea de orientare a elementului de care este fixat palpatorul în raport cu sistemul de referință fix;

\underline{P} – vectorul de poziție a originii sistemului de referință atașat elementului de care este fixat palpatorul în raport cu sistemul de referință fix.

În urma calculului cinematic direct rezultă coordonatele punctului caracteristic:

$$P_x = -(s_1c_3c_4 - s_2s_4c_1 - s_3c_1c_2c_4)(d_5 + d_6) + d_2c_1 + d_3c_1c_2 + d_4s_2c_1$$

$$P_y = -(s_1s_2s_4 + s_1s_3c_2c_4 + c_1c_3c_4)(d_5 + d_6) + d_2s_1 + d_3s_1c_2 + d_4s_1s_2$$

$$P_z = -(s_4c_2 + s_2s_3c_4)(d_5 + d_6) + d_1 + d_3s_2 + d_1c_2$$

Lungimile elementelor ($d_1 \dots d_6$) sunt cunoscute din construcție iar valorile sinusurilor și cosinusurilor unghiurilor $\theta_1, \dots, \theta_6$ sunt calculate pe baza informațiilor preluate de la traductorii care măsoară orientarea relativă a elementelor cuplelor cinematice de rotație.

2.2.3 Construcția digitizoarelor 3D cu palpator

Caracteristicile pe care un digitizor 3D trebuie să le îndeplinească pentru a avea succes pe piață sunt: precizie, manevrabilitate facilă, greutate redusă, fiabilitate și un preț de cost cât mai redus. Concurența care există între producătorii de digitizoare 3D cu palpator a dus la perfecționarea continuă din punct de vedere constructiv a acestor digitizoare, odată cu lărgirea gamei de aplicații astfel încât la ora actuală există o mare varietate de soluții constructive.

Principalele firme producătoare a digitizoarelor 3D cu palpator cu acționare manuală sunt Faro Technologies, ROMER Inc. și Immersion Corporation. O schemă bloc a construcției digitizoarelor 3D cu palpator este prezentată în figura 2.2.3.

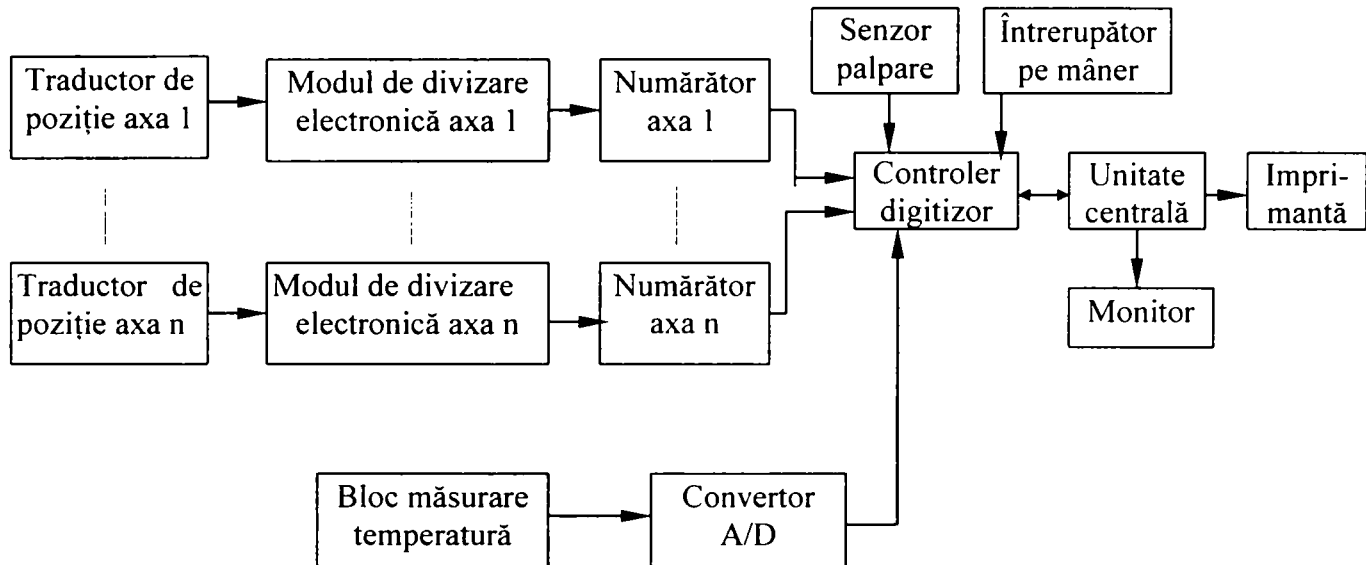


Fig. 2.2.3. Schema bloc a construcției digitizoarelor 3D

În figura 2.2.4. sunt prezentate elementele componente ale unui digitizor Microscribe. S-a încercat păstrarea denumirilor date de firma producătoare. Se observă că denumirile au fost preluate în mare parte din terminologia utilizată în robotică. Principalele elemente constructive a digitizorului 3D sunt suportul, elementele de legătură, palpatorul, traductorii pentru determinarea orientării relative a elementelor, unitatea electronică și sistemul de calcul.

Suportul are rolul de a realiza fixarea digitizorului de un obiect rigid situat în spațiul de lucru. Sistemul de referință fix al digitizorului este atașat de acest element și de aceasta este foarte important ca situarea suportului relativ la piesa digitizată să rămână neschimbată în timpul procesului de digitizare. Un astfel de suport care oferă o mare stabilitate, realizat de firma Faro Technologies se poate observa în figura 2.2.5..

În unele cazuri, printre accesoriile digitizorului este oferită și o masă a cărei elemente de susținere sunt reglabile. Pentru a asigura fixarea digitizorului de suprafețe metalice sunt utilizate suporturi magnetice. Pentru mărirea spațiului de lucru se poate utiliza o axă de translație lungă. În figura 2.2.6. este prezentată o soluție oferită de firma Faro Technologies.

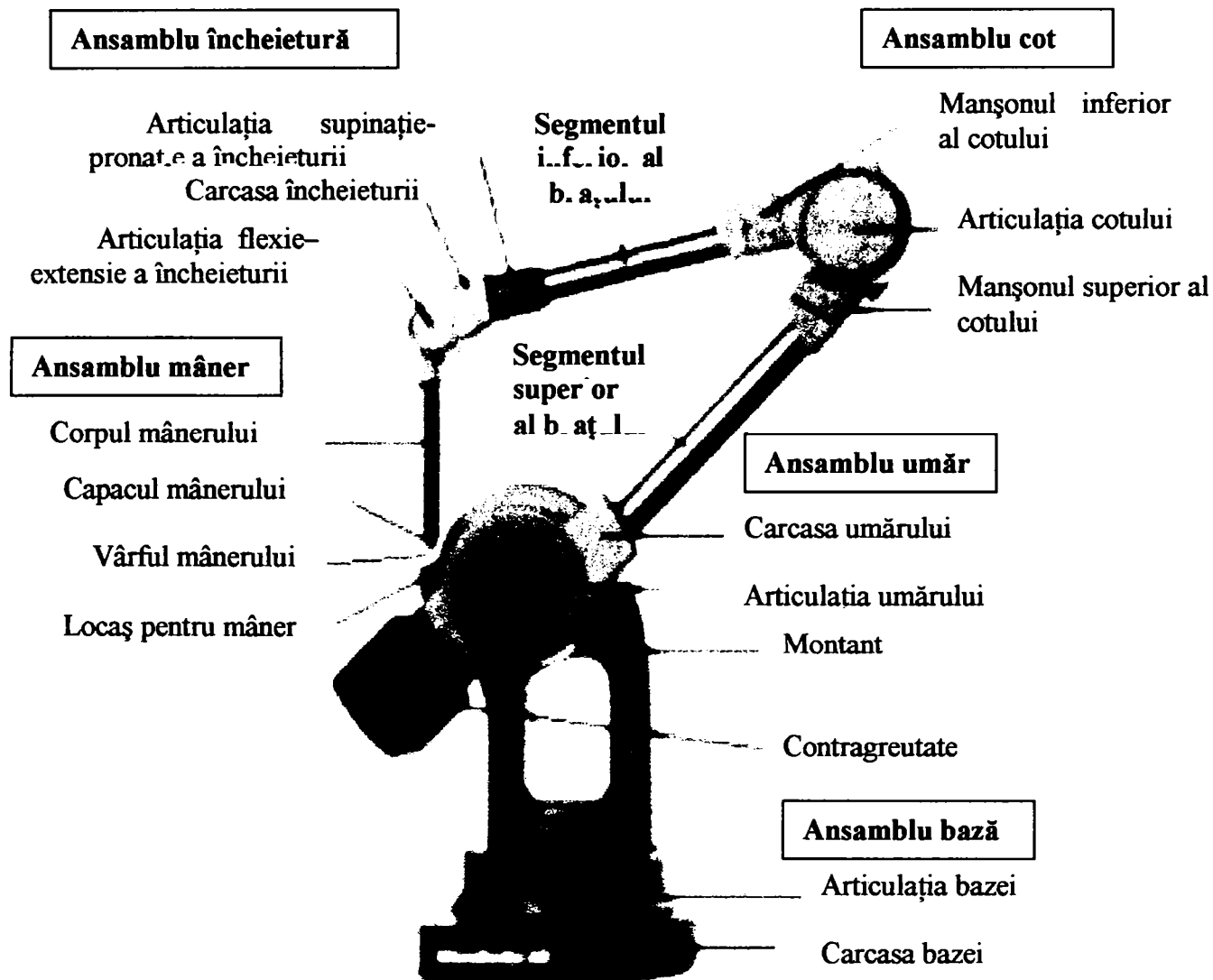


Fig. 2.2.4. Elementele componente ale unui digitizor Microscribe



Fig. 2.2.5 Suport tripod

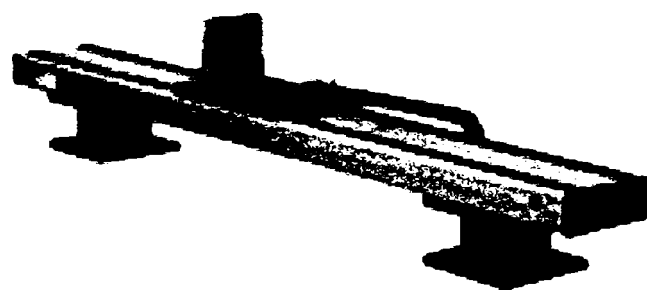


Fig. 2.2.6. Axa de translație lungă

Corpurile elementelor digitizorului, în cele mai multe cazuri de formă tubulară, sunt realizate din materiale ușoare ca de exemplu duraluminiu sau fibră de carbon. În interiorul acestora sunt introduse cablurile care asigură preluarea informațiilor de la traductorii de orientare relativă și palpator de către unitatea electronică de prelucrare primară.

Pe mânerul digitizorului sunt situate comutatoare cu care operatorul poate comanda acțiunile necesare în procesul de digitizare. Alte soluții care ușurează

transmiterea unor semnale de comandă către sistemul de digitizare îl reprezintă comutatoarele de mână, prezentate în două forme constructive în figura 2.2.7. (a) sau pedala de picior dublă 2.2.7. (b).

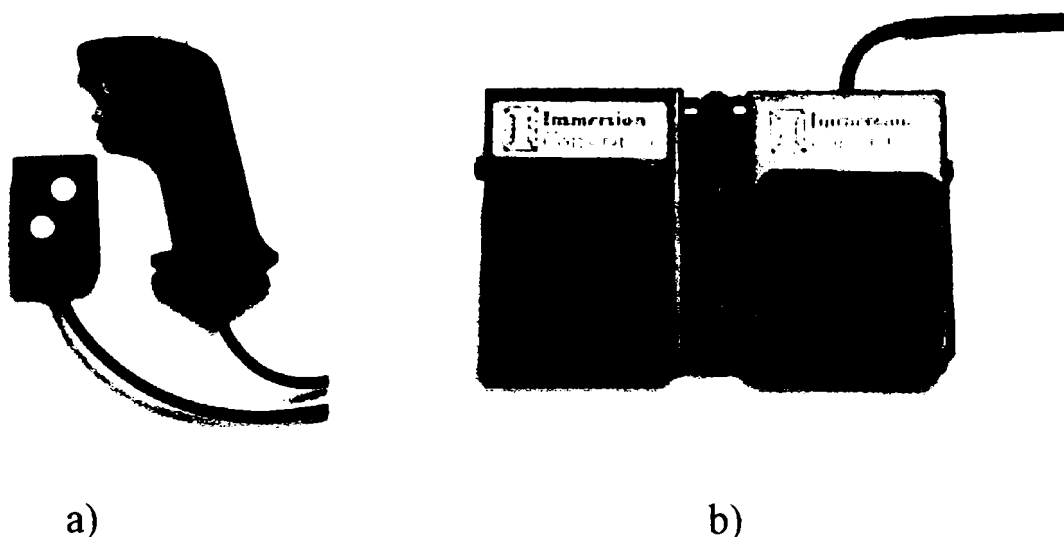


Fig. 2.2.7. Comutatoare de mână a) și pedală dublă b), accesorii ale digitizoarelor Microscribe.

Senzorii tactili (denumiți în limba engleză “probe” în contextul utilizării la digitizoarele 3D) situați în vârful ultimului element al digitizorului, au diferite forme impuse de aplicație. În figura 2.2.8. sunt prezentate câteva tipuri de palpator.

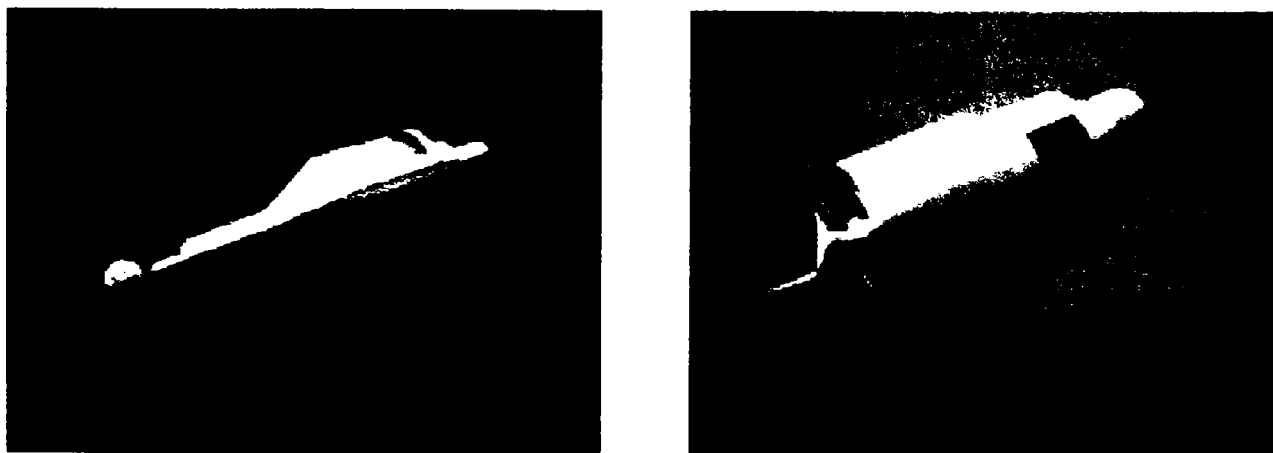


Figura. 2.2.8. Senzori tactili oferți de firma Faro Technologies

Unul dintre cele mai importante elemente ale digitizorului, cu influența cea mai mare în determinarea preciziei acestuia, este traductorul de orientare cu ajutorul căruia se stabilește orientarea relativă a elementelor cuplelor cinematice de rotație. Tipul de traductor preponderent utilizat este traductorul incremental de rotație.

Unitatea electronică are în componența sa echipamentele necesare pentru preluarea informațiilor de la traductorii de rotație și palpator, monitorizarea stării butoanelor de mâner sau a pedalelor de comandă, calculul geometric direct, interfață cu calculatorul. Unitatea electronică este dotată în cea mai mare parte a cazurilor cu

microcontroller pentru realizarea prelucrărilor primare iar pentru interfața cu sistemul de calcul este utilizată comunicația serială.

Calculatoarele utilizate de către sistemele de digitizare 3D cu palpator sunt de tip Laptop. Configurația necesară pentru aceste calculatoare este diferită în funcție de aplicație și de sistemul software utilizat pentru digitizare. Pentru a permite o cât mai bună vizibilitate, aceste calculatoare sunt susținute de standuri ajustabile de tipul celui prezentat în figura 2.2.9.

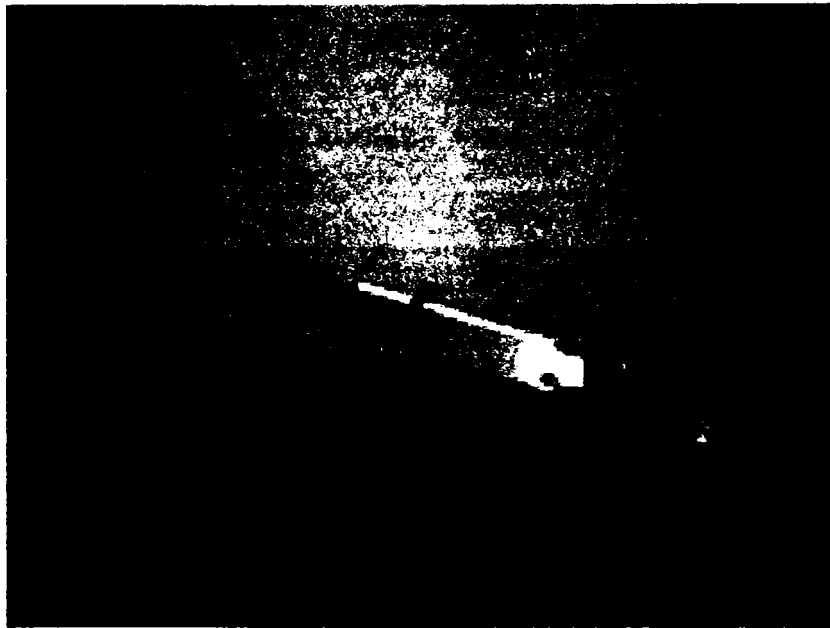


Fig. 2.2.9. Suport pentru calculatorul cu care comunică digitizorul 3D (Faro Technologies)

Există la ora actuală și soluții care utilizează în locul palpatorului un sistem de scanare laser. Acestea combină manevrabilitatea facilă a digitizoarelor 3D cu palpator cu unele avantaje ale metodelor de digitizare non-contact. Odată cu răspândirea digitizoarelor 3D cu palpator în noi sectoare de activitate vor apare noi soluții constructive adaptate unor aplicații concrete.

2.2.4 Procesul de digitizare cu digitizoare 3D cu palpator, acționate manual

Pentru exemplificarea digitizării cu ajutorul digitizoarelor 3D cu palpator, în continuare este prezentat pas cu pas digitizarea unui model de autoturism la scară redusă care în final va fi rasterizat în vederea obținerii unor imagini realiste. Informațiile și imaginile sunt preluate de pe site-ul firmei Imersion Corp iar digitizarea este realizată cu un digitizor Microscribe.

Primul pas constă în realizarea curbilor de profil care vor fi utilizate la obținerea suprafeței (fig. 2.2.10.) . Vârful palpatorului digitizorului 3D este deplasat de-a lungul suprafeței prin conducerea manuală de către operator iar când sunt atinse punctele dorite se acționează butoanele de pe mâner sau pedalele prevăzute în acest scop.

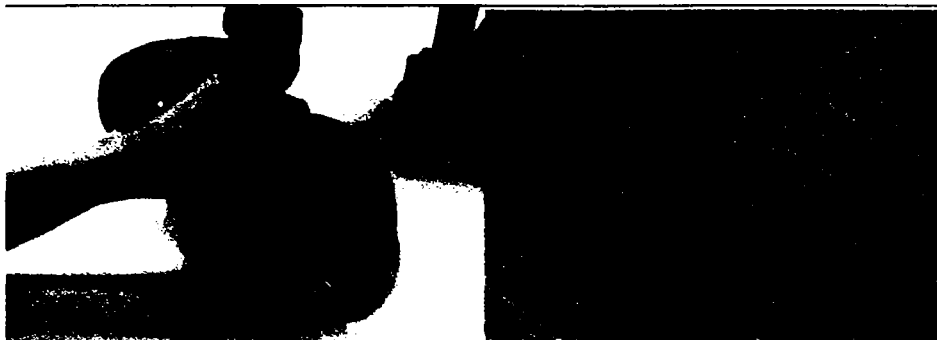


Fig.2.2.10. Realizarea curbilor de profil

După generarea curbilor de profil, acestea sunt selectate și se generează suprafața. (fig. 2.2.11.)



Fig. 2.2.11. Generarea suprafeței pornind de la curbele de profil

Digitizorul poate fi utilizat ca o sculă pentru editare. De exemplu, coordonatele a 3 puncte pot fi preluate cu ajutorul digitizorului Microscribe de pe suprafața obiectului fizic pentru definirea unui cerc. După generarea cercului, acesta poate fi utilizat pentru decuparea suprafeței și desenarea roții (fig. 2.2.12.). Combinația între posibilitatea de generare a suprafețelor de aproximare pornind de la curbe și a entităților geometrice definite analitic pornind de la puncte dă o mare flexibilitate modelării geometrice a obiectelor.



Fig. 2.2.12. Decuparea suprafeței cu ajutorul unui cerc definit prin 3 puncte

În pasul următor (fig. 2.2.13.) digitizorul este utilizat ca o sculă pentru editare ca și în pasul anterior. De această dată suprafața care reprezintă fereastra mașinii este

extrasă din suprafața realizată anterior care reprezintă caroseria și definită ca o suprafață separată. Separarea celor 2 suprafețe se face după o curbă definită prin puncte preluate cu digitizorul 3D.

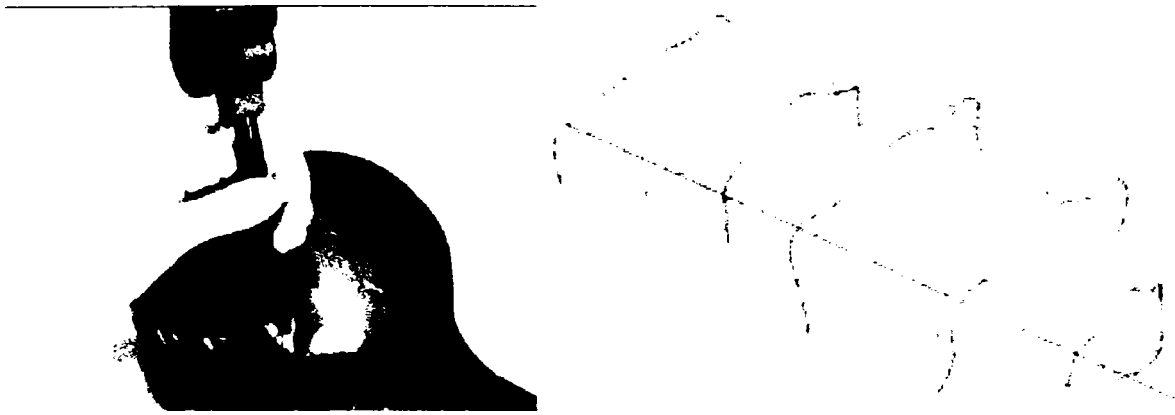


Fig. 2.2.13. Separarea unei suprafețe dintr-o altă suprafață utilizând o curbă definită prin puncte a căror coordonate sunt preluate cu digitizorul 3D

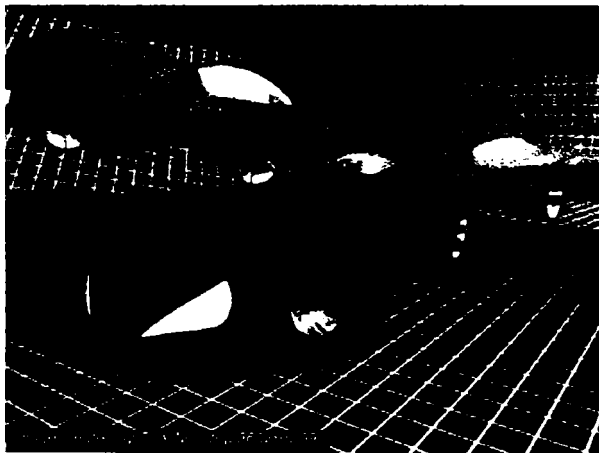


Fig. 2.2.14. Imaginea rezultată în urma randerizării modelului obținut pornind de la date preluate prin digitizare.

Pentru completarea modelului sunt create farurile și roțile utilizând curbe preluate cu digitizorul pe suprafața obiectului. Ultima fază este definirea culorilor și a proprietăților suprafețelor modelului în vederea randerizării (fig. 2.2.14.). Modelul rezultat în urma digitizării și ajustării în sistemul CAD poate fi utilizat pentru obținerea unui model fizic prin frezare pe o mașină de frezat în coordonate (fig. 2.2.15.).



Fig. 2.2.15. Realizarea unui model fizic pornind de la date rezultate în urma digitizării

2.2.5 Performanțele digitizoarelor 3D cu palpator

Privit prin prisma principalului scop, acela de obținere a informațiilor cu privire la forma și dimensiunile obiectelor prin determinarea datelor referitoare la poziția punctelor de pe suprafața acestora, digitizorul 3D poate fi considerat un instrument metrologic. Pentru aprecierea calităților metrologice ale digitizorului 3D sunt luate în considerare caracteristicile sale constructive și performanțele sale în funcționare.

Valorile nominale ale caracteristicilor constructive și cele ale performanțelor fiecărui tip de digitizor sunt indicate de firma producătoare în specificațiile produsului. Prin măsurări și încercări se determină caracteristicile constructive și performanțele reale ale digitizorului. Precizia digitizorului 3D se exprimă prin eroare și are două aspecte : exactitatea și repetabilitatea. Exactitatea reprezintă gradul de corespondență a mărimii nominale și a celei reale un moment dat și se exprimă prin notarea erorii la un moment dat. Valoarea repetabilității este cea mai mare valoare a erorii într-un anumit interval de timp.

Printre operațiile metrologice care se aplică digitizoarelor 3D cu palpator fără acționare este calibrarea. Calibrarea unui mijloc de măsurare este operația de determinare a corespondenței dintre valoarea măsurată (semnalul de ieșire) și valoarea reală a dimensiunii măsurandului. În sens restrâns , prin calibrare se înțelege operația de corectare aplicată digitizorului pentru obținerea unei măsurări precise. Această corectare poate fi realizată în cadrul digitizoarelor 3D atât prin ajustări mecanice cât și prin corecții software. La sistemele de digitizare 3D cu palpator , calibrarea implică două aspecte: calibrarea dispozitivului de ghidare al digitizorului și calibrarea palpatorului.

Calibrarea dispozitivului de ghidare al digitizorului este realizată de obicei de către firma producătoare utilizând instrumente hardware și software specifice. Prin această calibrare de fabrică se determină dimensiunile și schema cinematică reală a digitizorului, se calculează corecțiile necesare și se înscriu în dispozitivul de memorare al sistemului cu microcontroller al digitizorului. Digitizorul își poate pierde calibrarea dacă se produce deformarea elementelor acestuia (formate din tuburi lungi), distrugerea rulmenților tractoarelor de rotație sau deteriorarea sistemului electronic.

Calibrarea palpatorului este necesară deoarece digitizoarele cu palpator acceptă o mare varietate de vârfuri. Pentru vârfurile standard oferite de firma producătoare împreună cu digitizorul, valorile cu care se efectuează corecțiile sunt indicate de producători. Sistemul software cu care digitizorul colaborează pentru îndeplinirea sarcinilor de măsurare asigură rutinele de calibrare a palpatorului. Deoarece vârful palpatorului este în contact cu suprafața piesei, aceasta se uzează în timp. Din acest motiv palpatorul trebuie recalibrat la anumite intervale de timp .

Precizia digitizoarelor 3D este exprimată în termenii deviației standard. Deviația standard exprimă dispersia valorilor față de media acestora :

$$\sigma = \sqrt{\frac{n \sum x^2 - (\sum x)^2}{n(n-1)}} \quad (2.2.3)$$

Pentru o analiză statistică relevantă a datelor referitoare la precizia digitizoarelor 3D trebuie efectuat un număr semnificativ de măsurări. De exemplu firma Faro recomandă un număr de 1000 măsurări / test. Pentru prelucrarea rezultatelor măsurătorilor se consideră repartiția normală (Gauss-Laplace).

Funcția densitate de repartiție este dată de relația :

$$h(x, y, \sigma) = \frac{1}{\sigma\sqrt{2\pi}} \cdot \exp\left[-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2\right] \quad (2.2.4)$$

Valoarea medie ideală, μ , determină poziția repartiției pe axa Ox iar abaterea standard determină forma curbei. Reprezentarea funcției de repartiție în raport cu mărimea $z = (x - \mu) / \sigma$, mărime proporțională cu abaterea rezultatului măsurării față de valoarea medie ideală, este prezentată în fig. 2.2.16.

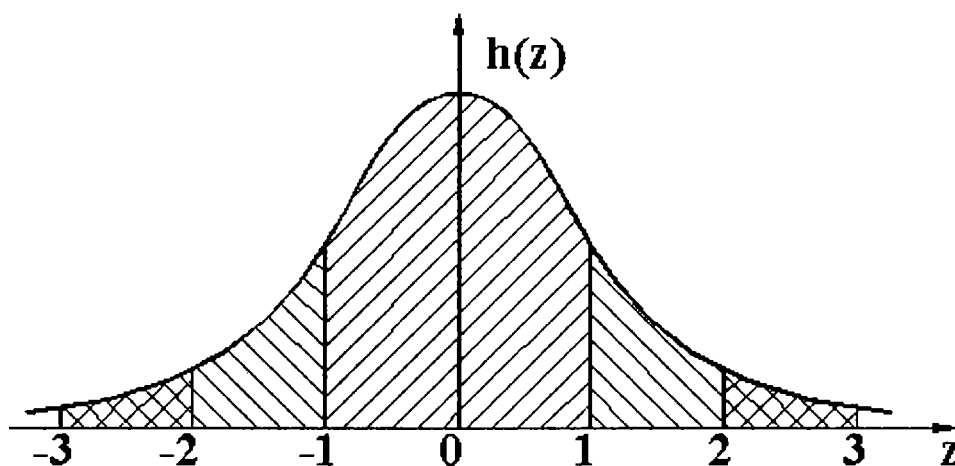


Fig. 2.2.16. Curba de densitate a repartiției normale

Suprafața de sub curba de densitate a repartiției normale (fig. 2.2.5.1) este repartizată astfel [PUR90]: 68.28% din suprafață în intervalul (-1,1) , 95.45% din suprafață în intervalul (-3,3).

Pentru evaluarea preciziei digitizoarelor 3D se parcurg 2 etape. Prima etapă este o etapă preliminară și constă din verificări care au rolul de a permite constatarea funcționării corecte sau incorecte a digitizorului. După constatarea funcționării corecte a digitizorului se trece la etapa de evaluare propriu-zisă a preciziei digitizorului. În etapa preliminară se realizează verificarea stabilității temperaturii, calibrarea palpatorului cu vârf sferic și testul “bară cu bilă”.

Pentru verificarea stabilității temperaturii se măsoară și se înregistrează temperatura unui element al digitizorului la începutul testului, se efectuează operațiile de digitizare o anumită perioadă de timp prestabilită și se măsoară din nou temperatura. Verificarea stabilității temperaturii constă în calculul diferenței valorilor temperaturilor măsurate și compararea cu valoarea admisă pentru această diferență. Calibrarea palpatorului cu vârf sferic implică calcularea corecțiilor necesare și realizarea setărilor cu ajutorul rutinelor de calibrare.

Testul “Bară cu bilă” constă în determinarea cu ajutorul digitizorului a lungimii reale a unei bare etalon, a cărei lungime nominală se cunoaște, calculul deviației pentru valorile măsurate și compararea cu valorile admise. În figura 2.2.17 este prezentat testul bară cu bilă realizat cu un digitizor Faro.

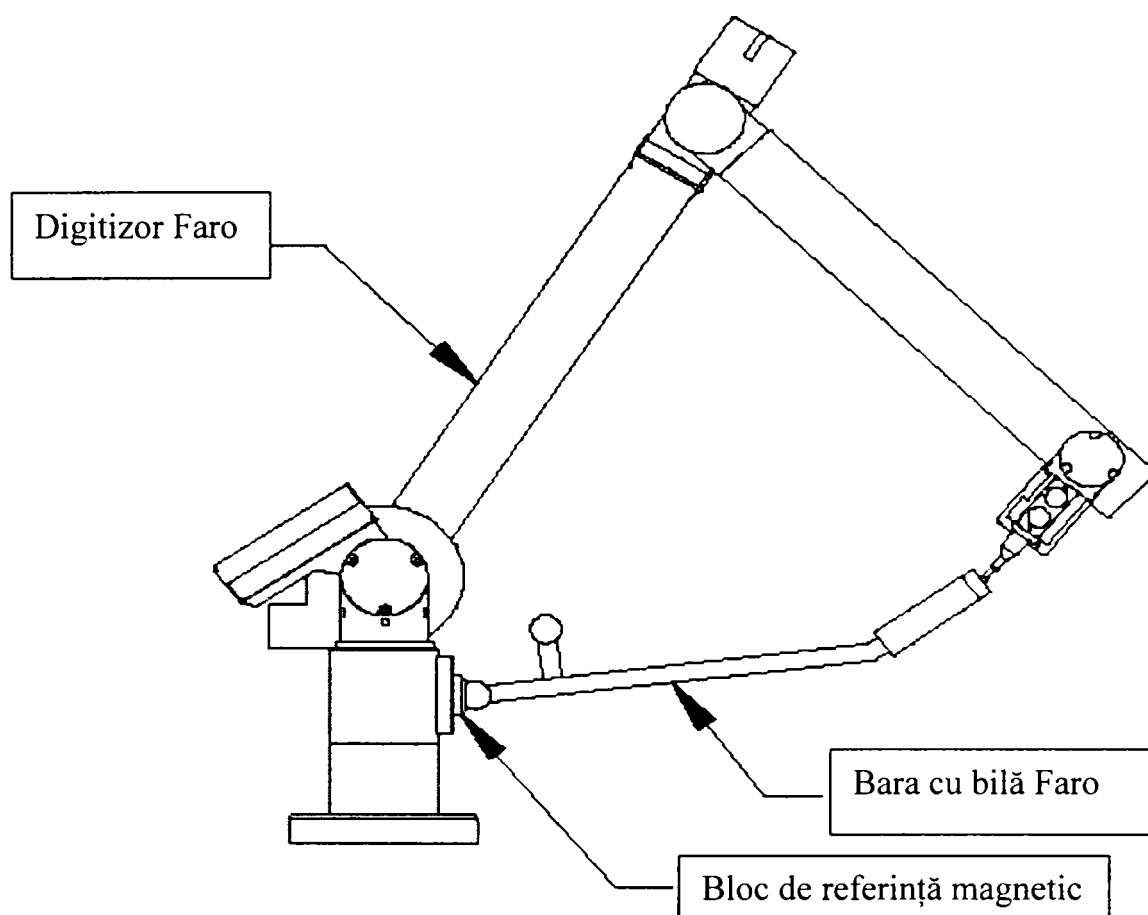


Fig. 2.2.17. Testul bară cu bilă

Bară cu bilă formează o cuplă sferică cu blocul de referință magnetic și o cuplă sferică cu vârful palpatorului. Blocul de referință magnetic are un alezaj conic în care se introduce capătul sferic al barei cu bile și este solidar cu elementul fix al digitizorului. Celălalt capăt al barei are un alezaj conic din material magnetizat care formează cupla sferică cu vârful palpatorului. Distanța între centrul bilei solidară cu bara și punctul caracteristic al digitizorului are o valoare cunoscută. De asemenea, se cunoaște poziția centrului bilei barei introdusă în blocul de referință.

Pentru efectuarea testului se deplasează bara astfel încât să ocupe situații în întregul spațiu de lucru al digitizorului 3D și se înregistrează coordonatele punctului caracteristic al palpatorului și valoarea calculată pentru lungimea barei. După înregistrarea unui număr suficient de puncte, se elimină valorile aberante datorate neglijenței operatorului și se calculează valoarea deviației standard. Valoarea obținută se compară cu valoarea limită precizată în specificațiile digitizorului.

Pentru evaluarea preciziei digitizorului, în urma constatării funcționării corecte a acestuia se realizează : testul de repetabilitate, testul cu calibru de pas și testul sferă la sferă.

Testul de repetabilitate constă în măsurarea coordonatelor unui punct singular cu ajutorul digitizorului 3D de un anumit număr de ori. Punctul se va măsura prin introducerea vârfului sferic al palpatorului într-un alezaj conic de forma celui prezentat în figura 2.2.18. Se urmărește ca orientarea palpatorului să fie cât mai diferită în timpul măsurărilor succesive și să se producă deplasarea relativă a elementelor tuturor cuplelor cinematice ale digitizorului 3D. După un număr oarecare de măsurători se calculează deviațiile standard pentru fiecare din coordonatele x , y și z a punctului măsurat și se verifică încadrarea în limitele admise.

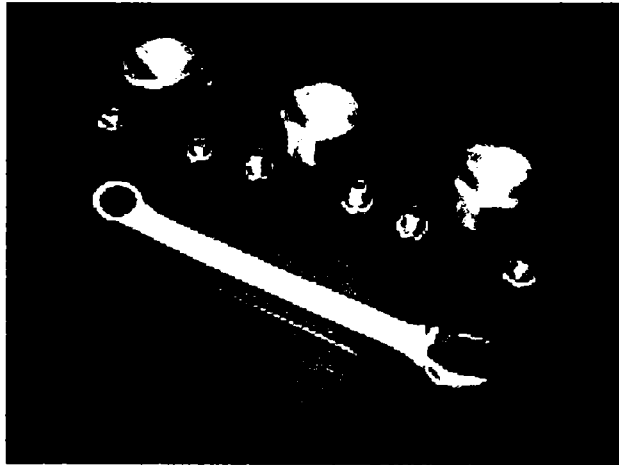


Fig. 2.2.18. Alezaje conice pentru testarea repetabilității punctului singular

Testul cu calibrul de pas urmărește determinarea erorii cumulate și a celei incrementale la măsurarea unor puncte dispuse liniar la un anumit pas. Axa x a sistemului de coordonate al digitizorului se va seta astfel încât axele de simetrie ale alezajelor conice ale calibrului să o intersecteze. Originea sistemului de coordonate se alege astfel încât să fie la intersecția dintre axa x și axa de simetrie a primului alezaj al calibrului.

Testul sferă la sferă constă în determinarea pe baza calculului a distanței dintre centrele a două sfere prin palparea cu ajutorul digitizorului a unor puncte de pe suprafața acestora. Distanța determinată prin măsurare se compară cu valoarea cunoscută și se determină dacă digitizorul asigură precizia impusă.

2.2.6 Aplicații ale digitizoarelor 3D cu palpator

2.2.6.1 Aplicații în industria automobilelor

În industria automobilelor, digitizoarele cu elemente articulate sunt utilizate la verificarea pieselor pe liniile de montaj, la verificarea deformațiilor caroseriilor în urma testelor de ciocnire, verificarea deschiderii ușilor autoturismului, verificarea situării componentelor interioare ale autoturismului. În figura 2.2.19. este prezentată digitizarea caroseriei automobilului pentru determinarea deformațiilor rezultate în urma ciocnirii iar în figura 2.2.20. o aplicație a digitizorului Faro la verificarea și ajustarea alinierii în cazul operațiilor de montaj. Numărul mare de cuple cinematice de rotație a digitizorului permite accesul în zone dificil de atins cu o mașină de măsurare în coordonate clasică.



Fig. 2.2.19. Digitizarea carcasei în urma testului la ciocnire



Fig. 2.2.20. Măsurări la montaj

Pentru inspecțiile finale realizate după asamblarea întregului autoturism este necesar extinderea spațiului de lucru pentru cuprinderea întregului produs, și accesul atât în zonele exterioare cât și a celor interioare. Aceasta trebuie realizată fără sacrificarea preciziei. Un astfel de sistem este realizat de firma Romer și este prezentat mai pe larg în paragraful 2.2.7.

2.2.6.2 Aplicații în industria aeronautică

Industria aeronautică și industria de automobile au fost întotdeauna principalele promotoare a noilor concepte și tehnologii. Pe măsură ce noile tehnologii s-au maturizat, acestea au pătruns și în celelalte sectoare de activitate.

În industria aeronautică, digitizoarele 3D cu palpator sunt utilizate în asociere cu programe software specializate la controlul de calitate al ușilor, ferestrelor, conurilor de ajutoraj, cockpit-ului (fig 2.2.21.) , părților de fuselaj, tubulaturii, etc.

Pentru siguranța aeronavelor este necesară o verificare a toleranțelor de asamblare a elementelor componente. Una din problemele dificile care se pune este determinarea toleranțelor de montaj a motorului aeronavei în interiorul spațiului

prevăzut în acest scop. Măsurarea individuală a miilor de componente ale motorului se realizează cu mașini de măsurare în coordonate clasice, dar pentru întregul motor în stare montată, de mari dimensiuni, acestea nu oferă o soluție satisfăcătoare. Deși în ultimii ani aceste mașini de măsurare în coordonate au fost îmbunătățite pentru a realiza măsurări pe liniile de montaj, operația de corelare a informațiilor rezultate din măsurare cu baza de date CAD nu este rezolvată pe deplin. Acest lucru a determinat o serie de firme să abordeze metode individuale pentru rezolvarea problemei.



Fig.2.2.21. Digitizor Faro utilizat la măsurări în cockpit



Fig. 2.2.22. Digitizor Romer utilizat la măsurări ale elementelor aeronavei

Una din soluțiile eficiente combină utilizarea digitizorului Romer cu un sistem de proiectare grafic. Acest sistem permite integrarea bazei de date CAD cu procesul de măsurare având ca efect scurtarea buclei de feed-back dintre activitatea de măsurare a produsului și procesul de proiectare și îmbunătățirea rezultatelor proiectării. În spațiul restrâns dintre motor și locul în care este amplasat în aeronavă, utilizarea digitizoarelor cu 6 grade de mobilitate permite flexibilitatea necesară pentru măsurare. Acolo unde operatorul nu are o vizibilitate directă a monitorului sistemului de calcul, acesta este ghidat sonor de către sistem prin preluarea informațiilor din baza de date CAD și corelarea cu situarea curentă a palpatorului digitizorului 3D.

2.2.6.3 Alte aplicații industriale

Aplicațiile industriale ale digitizoarelor 3D sunt la montajul subansamblelor, verificarea matrițelor, a pieselor turnate, verificarea deformării pieselor după sudare, a țevilor, etc. În figura 2.2.23. este prezentată utilizarea unui digitizor Faro pentru măsurarea unei piese de mari dimensiuni obținută prin turnare iar în figura 2.2.24. controlul de calitate al unei matrițe.

O gamă foarte largă de piese sunt obținute prin turnarea sub presiune a maselor plastice. Formele în care se vor turna aceste piese sunt proiectate de cele mai multe ori cu ajutorul programelor CAD și realizate prin frezare pe mașini cu comandă numerică. Pentru verificarea acestor forme de turnare în numeroase aplicații se utilizează digitizoarele 3D și mai ales cele cu elemente articulate, care s-au impus

datorită exactității și robusteții lor. Prin digitizare se pot obține date despre zonele care nu corespund cu datele de proiectare și se pot obține indicii pentru corecțiile ce trebuie realizate.



Fig. 2.2.23. Măsurarea pieselor de mari dimensiuni



Fig. 2.2.24. Controlul matrițelor

O situație care apare frecvent, este cea în care firmele care realizează matrițe pentru turnarea sub presiune primesc ca date de intrare pentru realizarea formelor prototipul piesei care va trebui executată în serie. În aceste condiții, procesul tehnologic impune digitizarea acestor obiecte, eventuale corecții ale modelului obținut prin digitizare, proiectarea matrițelor după care urmează obținerea formelor de turnare prin frezare pe mașini cu comandă numerică.

2.2.6.4 Aplicații în proiectarea formei și artă

În continuare, prin design se înțelege proiectarea formei produselor din punct de vedere estetic. Persoanele specializate în proiectarea formei produselor se numesc designeri. Datorită flexibilității și ușurinței de utilizare, digitizoarele 3D și-au extins domeniul de aplicații și în artă. Dintre categoriile de utilizatori se numără animatori, designeri, dezvoltatori de jocuri video, arhitecți. Cele mai multe aplicații în domeniul artistic o au digitizoarele Microscribe produse de firma Imersion. Exemplele prezentate în continuare utilizează acest produs.

În figura 2.2.25 este prezentat un model de suprafață obținut pornind de la date prelevate prin digitizare. Modelul a fost realizat în scopul utilizării în jocul de calculator NASCAR. O problemă care se pune în cazul jocurilor video pe calculator este numărul de fețe ale modelului. Deoarece majoritatea jocurilor de calculator implică renderizare în timp real numărul de fețe trebuie să fie cât mai redus, dar acest lucru nu trebuie să ducă la scăderea expresivității modelului.



2.2.25. Model de mașină realizat cu model de suprafață

Optimizarea între aceste două constrângeri se poate realiza printr-o digitizare inteligentă de către operator a modelului prin preluarea curbelor caracteristice. Având aceste curbe la dispoziție se pot încerca diferite variante până la obținerea unui model satisfăcător din punct de vedere al imaginii și a numărului de fețe. Modelul prezentat în figură conține aproximativ 3000 de poligoane și a fost realizat după un model la scara 1/18 a mașinii care a câștigat cupa NASCAR în 1999.

În figura 2.2.26. este prezentată realizarea unui model pentru animație pornindu-se de la un model din plastelină. Digitizarea a fost realizată utilizând aplicația software MicroScribe-Max . Curbele spline obținute pornind de la punctele digitizate au fost utilizate pentru crearea suprafețelor iar sistemul software Deep Paint 3D s-a utilizat pentru colorarea modelului. Animația a fost realizată cu software-ul 3DS Max – V3.

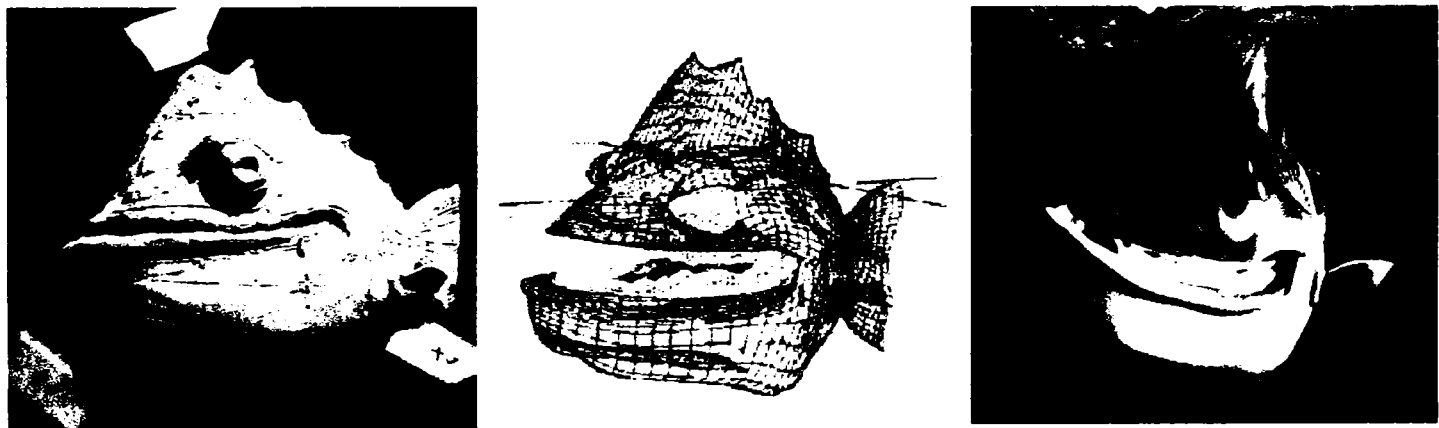


Fig. 2.2.26. Realizarea unui model pentru animație

2.2.6.5 Aplicații de realitate virtuală

Sculele virtuale care ajută cercetători, ingineri, designeri, personalul militar și instructorii/educatorii pentru o mai fluentă conectare cu mediul lor de lucru reprezintă următorul pas în asistarea de către calculator a muncii. Firma Fakespace este principala producătoare de sisteme de realitate virtuală bazată pe determinarea poziției capului operatorului cu ajutorul stabilirii orientării relative a unor elemente articulate. Principiul utilizat de aceste sisteme este identic cu cel al digitizoarelor 3D cu palpator, diferența fiind că în locul palpatorului se atașează casca operatorului. În acest caz impunerea poziției ultimului element al digitizorului se realizează de către operator cu ajutorul capului și nu cu ajutorul mâinii, ca în cazul digitizoarelor.

În figura 2.2.27. este prezentat sistemul BOOM HF care asigură o rezoluție de 1280x1024 pixeli și un câmp de vedere orizontal între 30 și 140 °. Sistemul este proiectat pentru aplicații în domeniul simulărilor de vehicule, modelarea cockpit-ului aeronavelor, comandă militară, centre de control. Deoarece sistemul lasă libere operatorului ambele mâini, acesta poate interacționa direct cu mediul real sau prin intermediul mănușilor cu senzori cu mediul virtual. În figura 2.2.28. este prezentată o aplicație care integrează sistemul Fakespace cu sistemul de proiectare asistată CATIA V5 pentru manipularea, asamblarea și dezasamblarea unor obiecte mecanice virtuale.

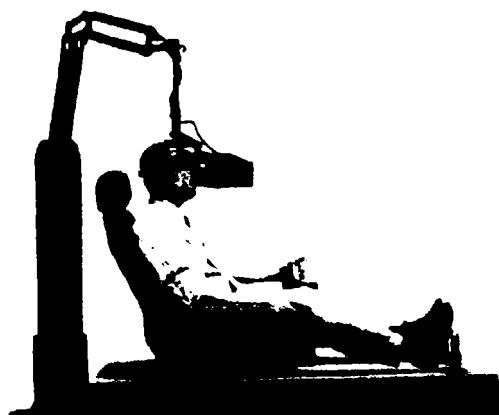


Fig.2.2.27. sistemul BOOM HF

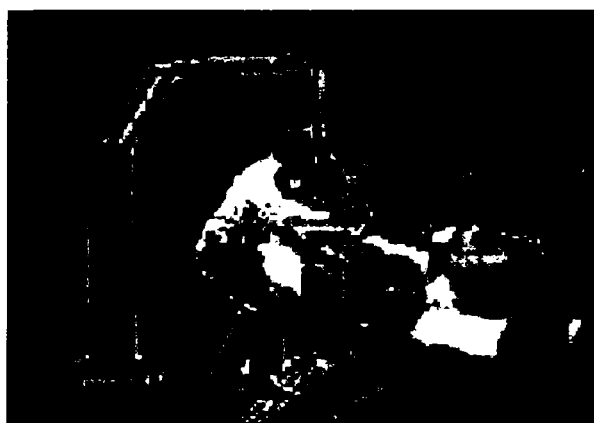


Fig. 2.2.28. Interacțiunea cu mediul virtual

2.2.6.6 *Aplicații de sculptură virtuală*

O sculptură virtuală este o sculptură realizată în mediul virtual. Sculptura virtuală poate fi afișată pe ecran sau poate fi materializată cu ajutorul mașinilor de realizare rapidă a prototipurilor. De-a lungul istoriei, tehnologia a influențat direcțiile de dezvoltare a sculpturii. Prin sculptura virtuală sunt înlăturate o serie de constrângeri care limitează materializarea imaginației creatoare a artistului. Exemple de constrângeri sunt gravitatea și proprietățile materialului. Sculptura virtuală are ca dezavantaje o interacțiune insuficientă om-calculator și o conexiune nu întotdeauna suficientă a intuiției umane cu modul de lucru a sistemului soft de realizare a sculpturii virtuale. Cercetările actuale sunt îndreptate în direcția menținerii înlăturării constrângerilor fizice și rezolvarea problemelor de interacțiune om-calculator.

2.2.7 Mărirea spațiului de lucru al digitizoarelor 3D

Într-o serie de aplicații mărirea obiectelor care trebuie digitizate depășește spațiul de lucru al digitizorului 3D cu palpator. Pentru eliminarea acestei deficiențe s-au impus două soluții. Prima implică fixarea unor puncte de referință și utilizarea calibrării după deplasarea relativă a elementului fix față de obiect iar a doua presupune determinarea continuă a situației elementului fix al digitizorului față de un sistem de referință. Cele două soluții vor fi exemplificate prin produsele oferite de Immersion respectiv de Romer Inc.

În continuare este descrisă soluția de utilizare a digitizorului 3D Microscribe pentru digitizarea unor obiecte a căror dimensiuni nu se încadrează spațiul de lucru al digitizorului. Sistemul software care lucrează împreună cu digitizorul este Rhino. După pornirea digitizorului se indică de către utilizator 3 puncte pentru origine, direcția axei X și respectiv a axei Y. În acest moment, digitizorul este conectat cu sistemul software.

În al doilea pas se definesc trei puncte de referință care vor fi utilizate pentru calibrare. Aceste puncte este necesar să fie marcate în prealabil sau să fie fizic repetabilă atingerea lor cu vârful digitizorului. Poziția ideală pentru punctele de referință este la intersecția a trei suprafețe care creează un colț concav. Lângă punctele alese pe modelul fizic se va nota indexul punctului. Cu ajutorul sistemului software se trasează o polilinie indicând vertexurile în ordinea în care au fost preluate coordonatele celor 3 puncte de referință. Pentru stabilirea corespondenței cu punctele de referință alese pe modelul fizic, se notează cu etichete aceste vertexuri în modelul din calculator (fig. 2.2.29). Polilinia și etichetele vertexurilor sunt trecute într-un strat special al modelului. Straturile modelului sunt asemănătoare unor foi de calc suprapuse. Utilizatorul poate opta pentru “ascunderea” unor straturi. Acest lucru avantajul că permite vizualizarea doar a geometriei relevante pentru o anumită fază de lucru.

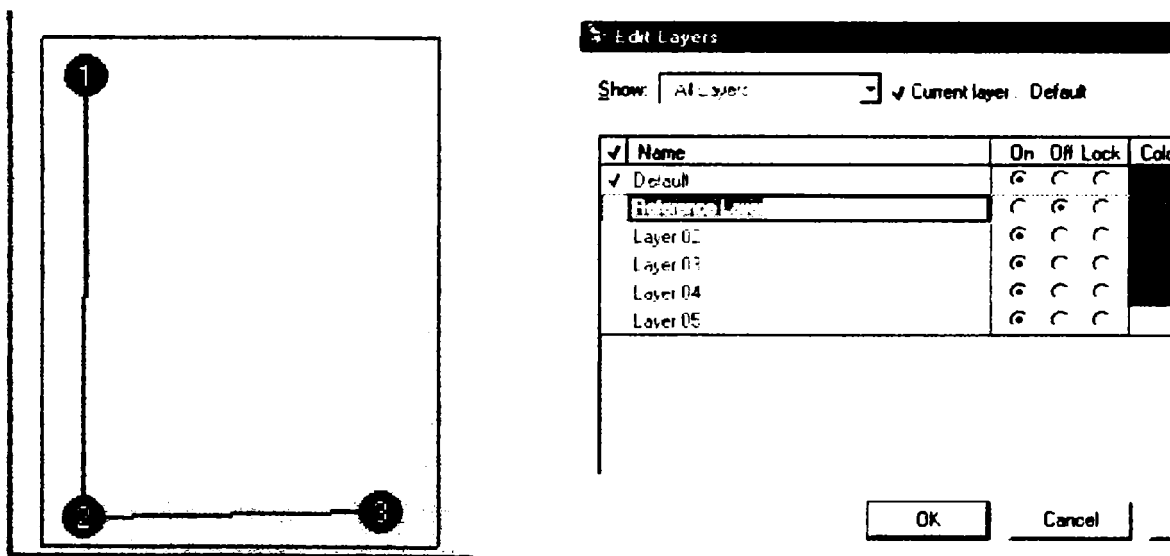


Fig. 2.2.29. Realizarea unei polilinii cu punctele de referință și trecerea în stratul desenului prevăzut în acest scop

După realizarea operațiilor descrise anterior este posibilă deplasarea digitizorului sau a modelului fizic și recalibrarea. Această recalibrare este necesară și dacă a fost oprit și repornit sistemul software. Este esențial ca după deplasarea piesei față de digitizor aceste puncte să rămână în spațiul de lucru al digitizorului adică să poată fi atinse de vârful digitizorului. Se realizează închiderea tuturor straturilor în afara de stratul care conține punctele de referință după care se selectează polilinia și se inițiază secvența de calibrare. Se indică pe rând punctele 2, 3 și 1 pentru origine, direcția axei x și respectiv a axei y. De câte ori este necesar pentru digitizarea întregului model se repetă procedura descrisă anterior.

O altă soluție pentru mărirea spațiului de lucru a digitizoarelor 3D cu palpator este oferită de firma Romer sub numele ELVIS (Extensibile Large Volumetric Inspection Systems). Sistemul utilizează tehnologia triangulației laser pentru a defini cu precizie un volum mare în care un digitizor articulată Romer din seria 2000 special echipat poate fi urmărit cu acuratețe. ELVIS utilizează 4 senzori laser pentru a determina situarea platformei mobile. Situarea sistemului de referință atașat elementului fix al digitizorului 3D față de sistemul de referință atașat platformei mobile este cunoscut prin construcție. Acest fapt permite determinarea situației palpatorului digitizorului 3D într-un spațiu de lucru de până la $6.1 \times 6.1 \times 2.13 \text{ m}^3$ (fig. 2.2.30.).

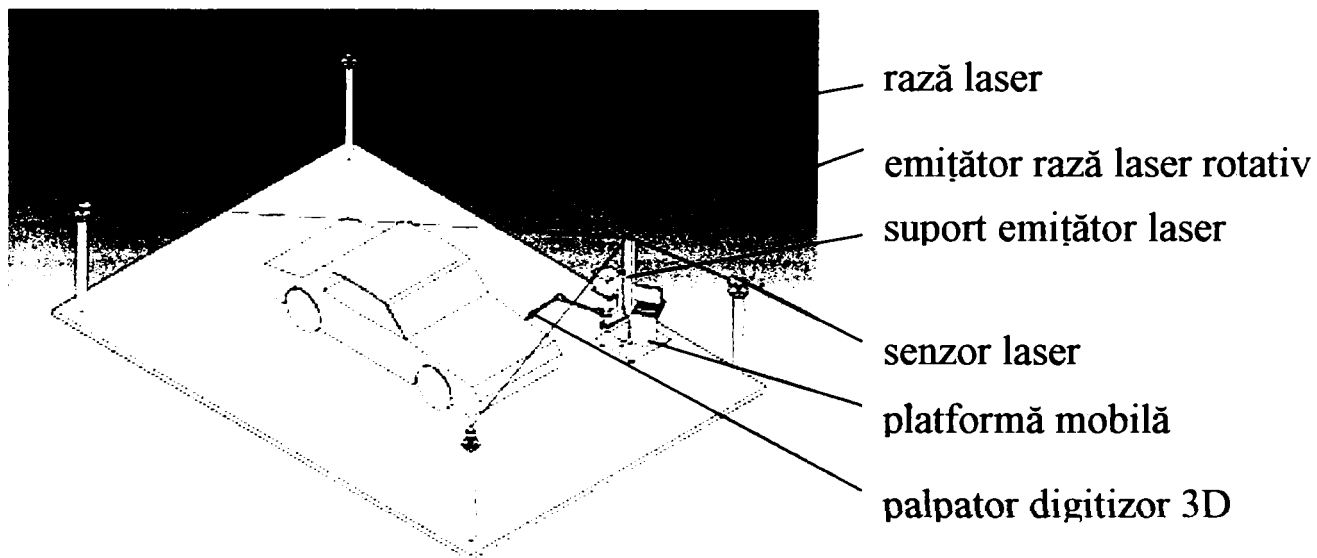


Fig. 2.2.30. Extinderea spațiului de lucru la digitizoarele ROMER

Un model Romer 2500 AACMM este montat pe o platformă de lucru mobilă echipată cu pernă de aer pentru a se putea deplasa cu ușurință și fără efort în interiorul spațiului de măsurare. Sistemul ELVIS are un soft bazat pe algoritmi speciali de poziționare, dezvoltati de Romer Inc., care permite utilizarea triangulației laser în timp real pentru a menține cu precizie situarea de referință absolută în timpul măsurării. Erorile nu se cumulează deoarece este realizată poziționarea absolută. Această soluție permite obținerea unei precizii mai ridicate decât soluția curent utilizată de digitizare pe porțiuni asociată cu recalibrare. Sistemul poate fi utilizat pentru a măsura piese de dimensiuni mari sau chiar vehicule întregi și necesită un singur operator.

2.3. Sisteme soft pentru digitizarea 3D

Eficiența procesului de digitizare depinde în mare măsură de gradul în care componentele hardware ale digitizorului comunică și interacționează cu sistemele software. În cursul procesului de digitizare, digitizoarele 3D cooperează cu un sistem de calcul asociat. Sistemul de calcul asociat este de obicei un calculator personal de tip portabil. Digitizarea 3D constă într-o fază de preluare a informațiilor despre forma și dimensiunile obiectului urmată de o fază de reconstrucție a suprafețelor (Fig. 2.3.1).

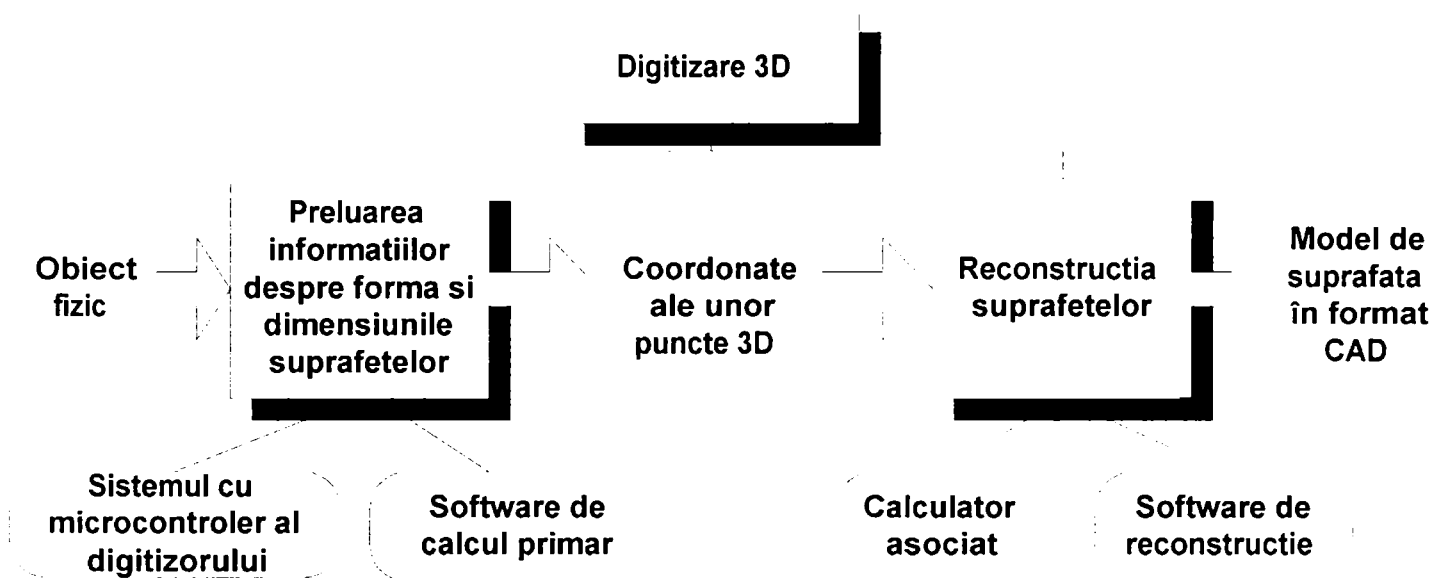


Fig. 2.3.1 Fazele digitizării 3D

În prima fază, cu ajutorul digitizorului 3D sunt palpate puncte de pe suprafața obiectului fizic sau se realizează scanarea suprafeței obiectelor. Ieșirile din această fază sunt constituite din coordonatele unor puncte 3D.

La digitizoarele 3D cu palpator fără acționare, calculele cinematice pentru determinarea coordonatelor punctului palpat pe baza informațiilor provenite de la traductoarele de poziție se realizează de obicei cu sisteme bazate pe microcontrollere. La acest tip de digitizoare, transmiterea informațiilor între sistemul cu microcontroler al digitizorului și calculatorul PC asociat se realizează prin comunicație serială. Datele, sub forma coordonatelor unor puncte individuale sau a unor șiruri de puncte sunt stocate pe hard-disk-ul calculatorului asociat.

În cazul scannerelor laser 3D, determinarea coordonatelor punctelor 3D de pe suprafața obiectului scanat necesită o putere de calcul mai mare. Pe lângă calculele cinematice necesare stabilirii situației capului de scanare față de sistemul de referință fix al sistemului, în cazul scanării trebuie efectuate și calculele de triangulație. Datele vehiculate în unitatea de timp este mult mai mare decât în cazul digitizoarelor cu palpator. Transferul datelor de la scanner la sistemul de calcul asociat se realizează prin comunicație paralelă. Datele, sub forma vertexurilor unor polilinii 3D sau a norilor de puncte sunt stocate pe hard-disk-ul calculatorului asociat. Capacitatea de stocare trebuie să fie substanțial mai mare decât la digitizoarele 3D cu palpator.

Faza de reconstrucție a suprafețelor utilizează ca date de intrare coordonatele 3D stocate în faza anterioară sub forma punctelor individuale, șiruri de puncte sau nori de puncte. În măsura în care capacitatea sistemului de calcul o permite, faza de reconstrucție se poate suprapune în timp cu faza de prelevare a informațiilor de formă. Faza de reconstrucție este o fază de procesare internă bazată pe algoritmi de calcul și are ca rezultat obținerea unor suprafețe în formate convenționale CAD. Modelul de suprafață rezultat poate fi sub formă de seturi de poligoane sau sub forma unor suprafețe de interpolare de tip NURBS, Bezier, etc. Tratarea din punct de vedere matematic a modelării suprafețelor este prezentată în capitolul 3.

Facilitățile oferite de sistemele software pentru digitizare diferă în funcție de tipul digitizorului. Majoritatea sistemele software pentru digitizare specializate pentru digitizoare 3D cu palpator fără acționare permit: vizualizarea datelor digitizate în timpul procesului de digitizare, ajutor pentru calibrare, utilizarea digitizorului ca mouse 3D. Sistemele software care lucrează cu scanere laser 3D oferă posibilitatea feed-back-ului vizual, setării parametrilor digitizării, înregistrarea mișcărilor capului de scanare în vederea scanării unor piese similare, etc .

Feed-back-ul vizual este de o mare importanță deoarece ajută operatorul uman să preia informații despre întreaga suprafață a obiectului. Pe măsură ce datele punct sunt capturate, acestea sunt afișate pe monitorul calculatorului asociat. Operatorul poate observa zonele care nu au fost încă digitizate. Acest lucru este util în special când procesul de digitizare este întrerupt din anumite motive și reluat după un anumit timp.

Parametri digitizării sunt setați de obicei înaintea începerii procesului de digitizare. Pentru digitizoarele cu palpator acești parametri pot fi de exemplu numărul de puncte 3D preluate în unitatea de timp sau distanța minimă pentru care se înregistrează punctele în cazul preluării șirurilor de puncte. Pentru scanerile 3D laser parametrii scanării pot fi densitatea scanării, eșantionarea în timp, numărul de culori în care se realizează scanarea.

Calibrarea este necesară în special pentru digitizoarele 3D cu palpator și se referă la calibrarea palpatorului în cazul utilizării unor palpatoare cu vârfuri diferite de cele standard pentru digitizorul respectiv, sau pentru compensarea uzurii vârfului. Pentru a verifica dacă palpatorul este corect instalat, înainte de fiecare secvență de digitizare se apelează o funcție de detectare automată a palpatorului. Dacă se digitizează piese a căror dimensiuni nu se încadrează în spațiul de lucru al digitizoarelor 3D cu palpator este de asemenea necesară o calibrare după schimbarea situării relative a elementului fix al digitizorului față de piesă. La scanerile laser, calibrarea se referă în principal la sistemul optic.

Digitizorul 3D cu palpator se poate utiliza ca mouse 3D, deci ca periferic de calculator. Această facilitate permite specificarea unor coordonate 3D în cadrul unor programe care necesită acest lucru, de exemplu programe CAD. De asemenea digitizorul 3D poate fi utilizat pentru activarea unor comenzi prin intermediul iconurilor sau meniurilor. Acest lucru asigură o libertate sporită operatorului uman deoarece necesitatea utilizării tastaturii poate produce discontinuități în procesul de

digitizare. O altă facilitate a utilizării digitizorului ca mouse 3D permite navigarea în cadrul modelului digital.

Utilizarea digitizoarelor 3D cu palpator nu se limitează doar la preluarea formei suprafețelor. Sisteme software cu care lucrează aceste digitizoare permit utilizarea lor la verificări ale dimensiunilor pieselor sau a modului de asamblare. Figura 2.3.2 ilustrează utilizarea unui digitizor Romer la verificarea limitelor de deschidere a ușii automobilului și vizualizarea datelor pe monitorul calculatorului asociat. În cazul în care există un model CAD iar locațiile punctelor care trebuie măsurate sunt specificate anterior este posibilă ghidarea sonoră a operatorului uman. Facilitatea este importantă mai ales în cazul în care operatorul se află într-un spațiu restrâns și nu are posibilitatea de vizualizare a monitorului calculatorului asociat. Măsurarea ghidată de datele teoretice pentru inspectarea unor puncte predefinite oferă posibilitatea comparării modelului real cu cel teoretic. De asemenea, este posibilă măsurarea ghidată pe secțiuni pentru inspectarea profilurilor. Măsurarea ghidată pe secțiuni permite utilizarea planurilor de secționare paralele, radiale sau cu orientări definite de operator. Măsurarea ghidată a situației este utilizată pentru operații de asamblare și permite determinarea mărimii necesare pentru deplasările relative a elementelor în vederea unei asamblări corecte. Sistemele software asigură facilități de creare a macrocomenzilor pentru operații de măsurare repetitive și operații de prelucrare a datelor. Rezultatele inspecțiilor sunt comparate cu datele teoretice și pot fi afișate pe monitor, listate la imprimantă sau salvate în fișiere în format DMIS (Dimensional Measuring Interface Specifications) sau SPC (Statistical Process Control).

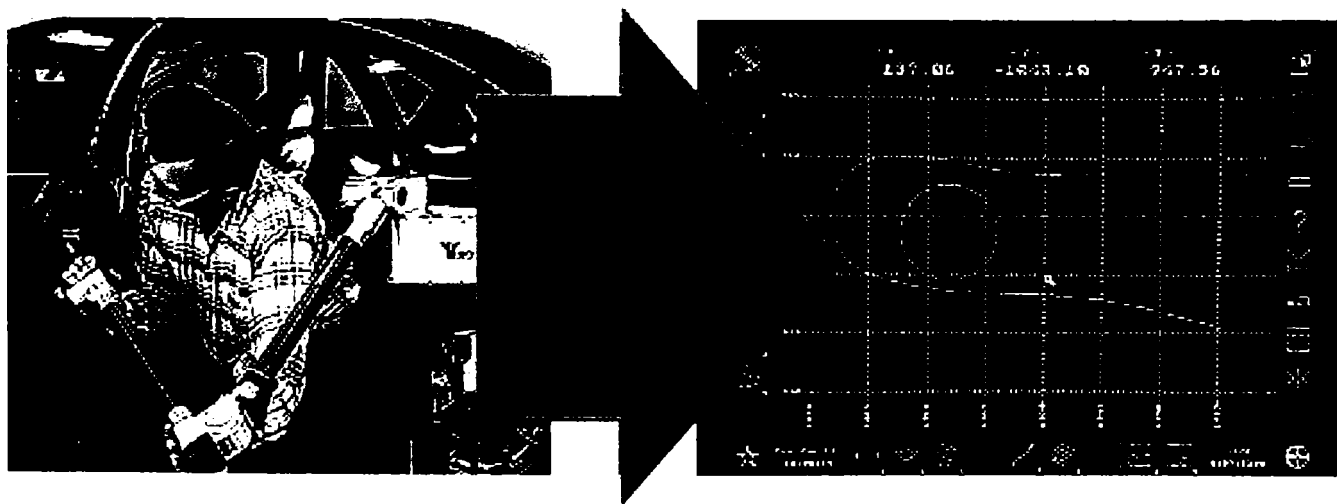


Fig. 2.3.2 Captură de ecran a sistemului software Suprastuff

Digitizorul 3D cu palpator permite indicarea unor puncte în spațiu care nu sunt situate pe suprafața unui obiect existent fizic. Cu ajutorul unor sisteme software adecvate este posibilă realizarea proiectării constructive a unor piese. Entitățile geometrice standard care pot fi create și manipulate sunt informațiile despre seturi de puncte, normale în seturi de puncte, punct singular, linie, plan, cerc, sferă, cilindru, curbă, unghi, lungime.

Sistemele software care realizează reconstrucția suprafețelor se bazează pe algoritmi de: generare a curbelor de interpolare și aproximare din șiruri de puncte, generare a suprafețelor strâmbe în spațiu pornind de la curbe, realizare a suprafețelor fațetate cu triunghiuri sau poligoane. De asemenea, majoritatea sistemelor software de reconstrucție permit acoperirea zonelor incomplet digitizate și corectarea unor zone incorect digitizate.

Sistemele de digitizare 3D cu palpator permit operatorului preluarea unor puncte sau curbe (specificate prin șiruri de puncte) cu ajutorul cărora se pot construi cu ușurință suprafețele. În majoritatea cazurilor suprafețele sunt definite prin indicarea unor profiluri sau a unor curbe care închid o zonă de suprafață. Reconstrucția suprafețelor se poate realiza chiar în timpul procesului de digitizare. Facilitățile necesare pentru preluarea informațiilor despre punctele care aparțin suprafețelor obiectului digitizat și pentru generarea suprafețelor sunt integrate în sisteme programe adiționale (plugin-uri) care rulează în interiorul altor aplicații software. În figura 2.3.3 este prezentat programul adițional MicroScribe-Max, care rulează cu sistemul software 3D Studio Max. Sistemul software 3DStudio Max este un produs al firmei Autodesk specializat în grafică tridimensională și utilizat în special la prezentări realiste și crearea unor secvențe cinematografice.

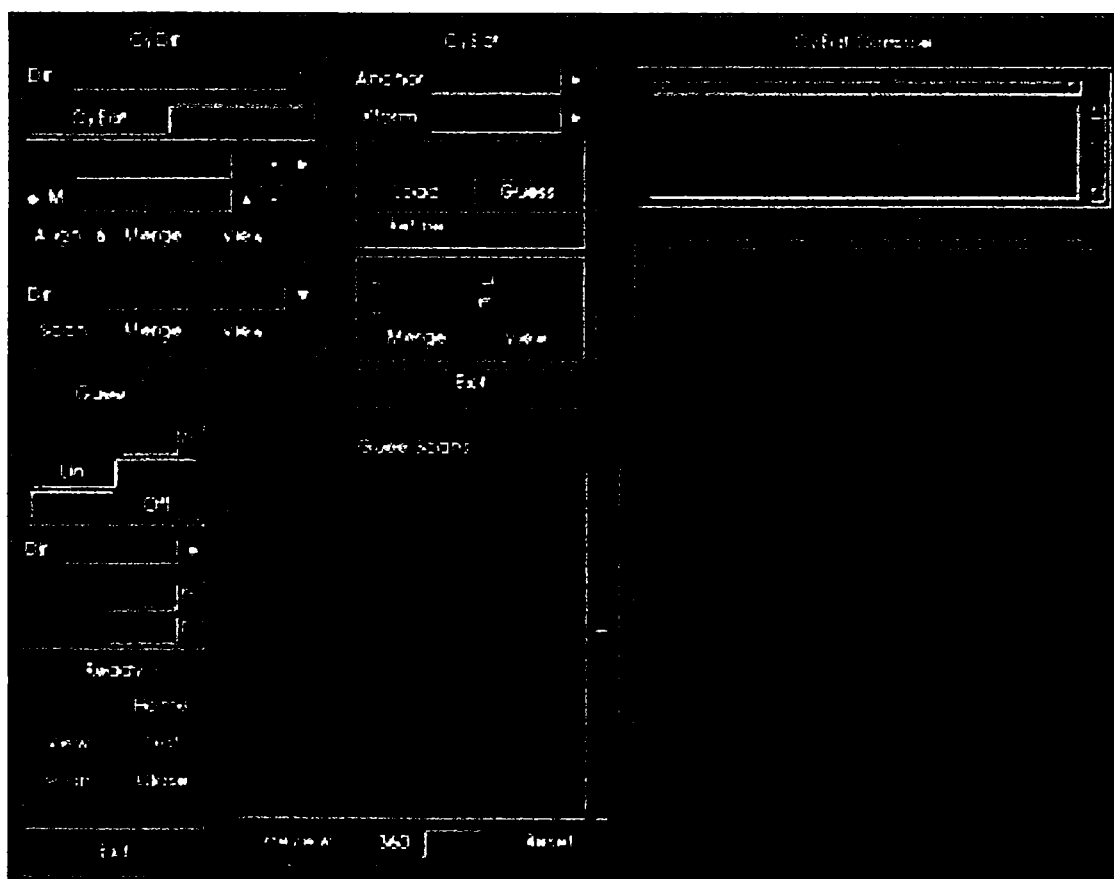


Fig. 2.3.3 Captură de ecran din timpul rulării pachetului soft MicroScribe-Max

Digitizarea modelului cu MicroScribe-Max utilizează o tehnică de subdivizare a suprafețelor. Se creează curbe spline prin palparea unor puncte de pe suprafața

obiectului. După ce toate curbele spline au fost create, se formează petece de suprafață generate din trei sau patru curbe spline apoi se unesc petecele adiacente.

Reconstrucția suprafețelor este necesară în special când coordonatele punctelor 3D de pe suprafața obiectului a fost obținută prin scanare. În acest caz, suprafața trebuie reconstruită pornind de la nori de puncte ale căror număr poate atinge ordinul milioane. Exemplificarea facilităților oferite de sistemele software din acesta categorie este realizată prin prezentarea modulului de editare a datelor digitizate integrat în pachetul software CATIA V5. Acest modul este o aplicație puternică care permite importul, exportul și procesarea datelor sub forma norilor de puncte.

Modulul lucrează cu o serie de termeni care prezintă generalitate în cadrul aplicațiilor software care lucrează cu informații despre nori de puncte: nori de puncte, sub-nori, activare, aliniere, filtrare, vecinătate, eșantionare, fațetare. Așa cum a fost prezentat în capitolul 1, norii de puncte sunt definiți ca un set de puncte 3D. Numărul de puncte poate varia de la un singur punct la câteva milioane. Punctele pot fi rezultatul digitizării sau a unor operații de calcul. Sub-norii reprezintă porțiuni ale norilor de puncte. Funcția de activare permite selectarea unei zone din norul de puncte pentru diferite operații. Prin aliniere se înțelege facilitatea de re poziționare a unor sub-nori de puncte relativ la alți sub-nori în vederea reconstruirii întregului obiect. Filtrarea norilor de puncte este o metodă de ascundere a unor puncte din nor pentru a permite desfășurarea mai rapidă a unor operații. Punctele care au fost ascunse pot fi vizualizate ulterior desfășurării operațiilor. O serie de operații de editare asupra norilor de puncte sunt relative la situarea punctelor în spațiu. Vecinătatea reprezintă distanța maximă la care sunt luate în considerare punctele în jurul unui punct dat în timpul operațiilor de editare. În aceste operații distanța de vecinătate este propusă de program dar poate fi setată de operator. Alegerea unei valori prea mari pentru distanță poate încetini mult desfășurarea operațiilor datorită măririi numărului de calcule necesare. Eșantionarea este o metodă care poate fi aplicată la importul norilor de puncte. Operatorul poate alege importarea doar a unui anumit procent din norul de puncte. Prin fațetare se creează poligoane cu trei sau patru laturi din norul de puncte. Cu ajutorul modelului fațetat se poate constata vizual calitatea digitizării. Modelele fațetate pot fi exportate în majoritatea aplicațiilor CAD.

Dintre facilitățile oferite de modulul de editare a norilor de puncte se enumerează: importul datelor din diferite formate, posibilitatea utilizării funcțiilor de filtrare, activare și ștergere pentru mărirea vitezei de realizare a operațiilor, posibilități de aliniere și unire a norilor de puncte, crearea de secțiuni plane prin norii de puncte, funcții de afișare și analiză, realizarea fațetării, exportarea modelului în diverse formate.

Crearea secțiunilor plane permite obținerea unor curbe pornind de la norii de puncte (fig. 2.3.4). Aceste curbe pot fi utilizate ulterior pentru crearea suprafețelor prin lofting. Avantajul este posibilitatea de definiție analitică a suprafețelor create în acest mod.

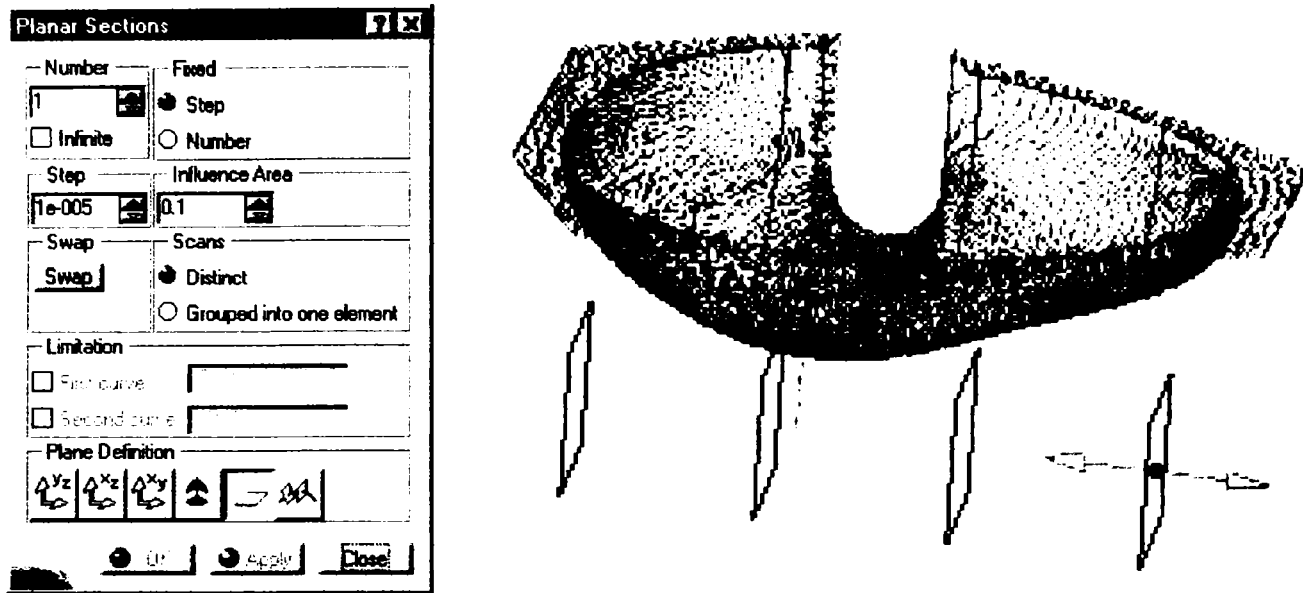


Fig. 2.3.4 Realizarea secțiunilor planare prin norii de puncte

Funcțiile de analiză permit determinarea distanței dintre punctele componente ale unui nor de puncte și o suprafață sau dintre doi nori de puncte. Vizualizarea acestei distanțe se poate face cu ajutorului unui cod de culori (fig. 2.3.5).

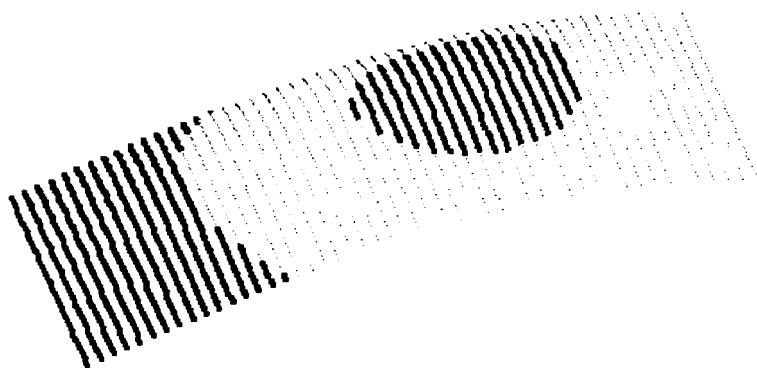


Fig. 2.3.5 Reprezentarea grafică a distanței dintre un nor de puncte și o suprafață

Norii de puncte rezultați prin scanări succesive a diferitelor zone sunt aduși în situațiile relative necesare pentru recrearea întregii suprafețe a obiectului scanat utilizând funcțiile de aliniere. După aducerea norilor de puncte în situarea corectă se utilizează funcțiile de unire a norilor de puncte, în urma acestei operații rezultând un singur nor. Pentru identificarea norilor de puncte, în figura 2.3.6. au fost reprezentate cu cadru de sârmă paralelipipedele care încadrează punctele componente a fiecărui nor.

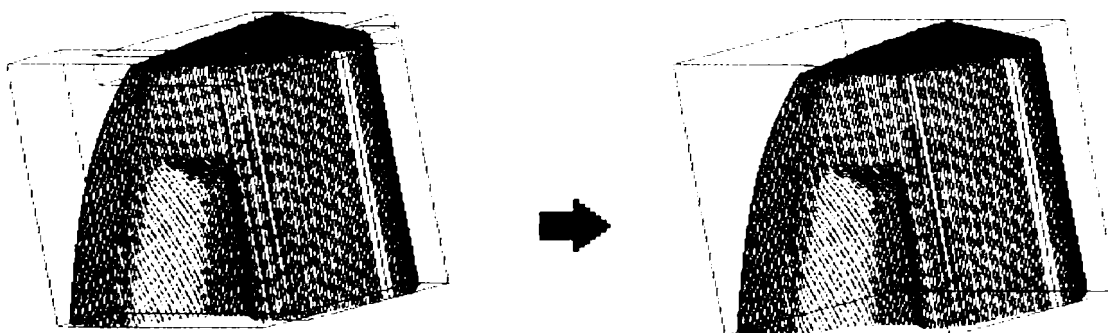


Fig. 2.3.6. Formarea unui nor de puncte prin unire

Datorită ariei largi a aplicațiilor digitizoarelor 3D numărul sistemelor software specializat pentru digitizarea 3D și utilizarea datelor digitizate este foarte mare. În măsura în care a existat o cerere suficientă pentru un tip specific de aplicație, iar beneficiile aplicării unui sistem software specializat pentru aplicația respectivă a depășit costurile necesare pentru realizarea lui, noi sisteme software au fost create. Ca exemplificare, în figura 2.3.7. este prezentată interfața unei aplicații software DigiSize specializate în realizarea modelului digital al corpului uman. Datele rezultate prin scanarea laser pot fi utilizate în vederea realizării unor articole de îmbrăcăminte care să se potrivească exact la forma și dimensiunile persoanei, studii antropologice, aplicații medicale, studii de ergonomie, jocuri de calculator. Dimensiunile necesare pentru realizarea articolelor de îmbrăcăminte sunt extrase automat de către sistemul software.

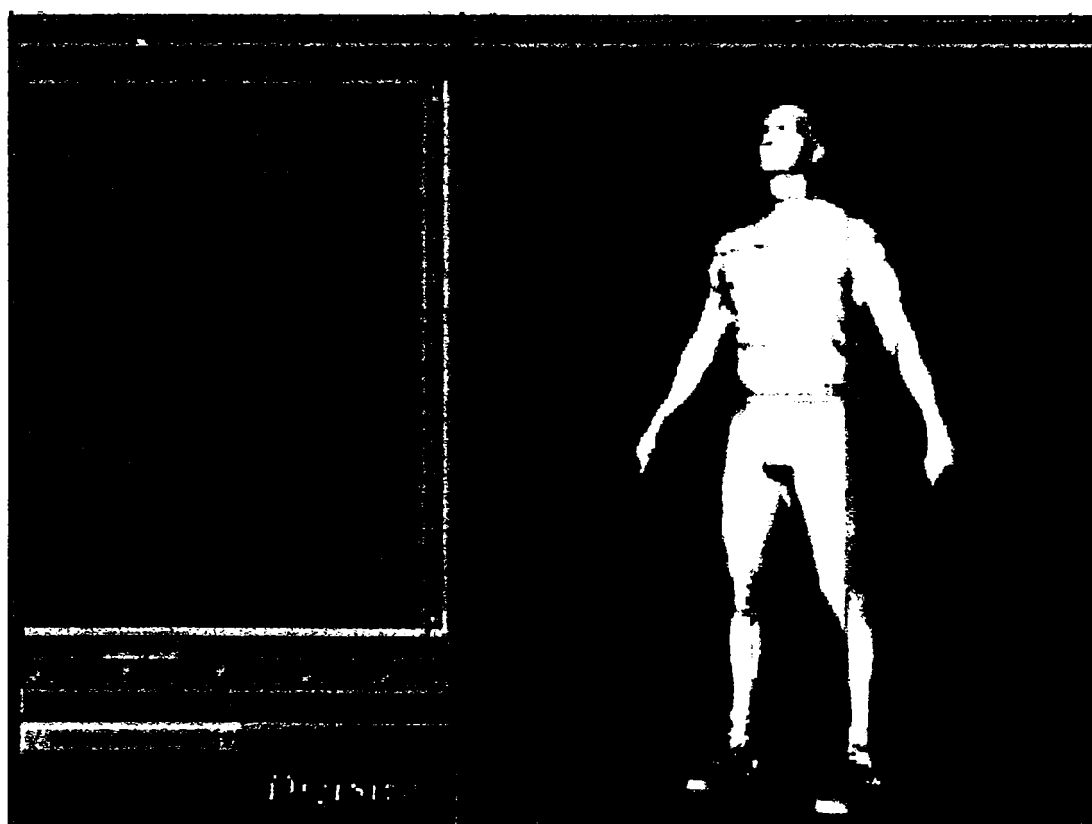


Fig. 2.3.7. Captură de ecran al aplicației software DigiSize

Odată cu înțelegerea de către factorii de decizie a beneficiilor utilizării digitizării 3D, dezvoltarea domeniului va primi o dinamică și mai accentuată. Autorul dorește să aducă contribuția în teză la realizarea unor aplicații software specifice pentru industria mobilei.

2.4. Tendințe actuale în construcția și utilizarea digitizoarelor 3D

Principala aplicație a sistemelor de digitizare 3D este crearea unui model digital al unui obiect existent fizic. Acest model digital poate fi randat și tipărit pentru comunicarea informațiilor referitoare la forma obiectului, pentru extragerea

dimensiunilor în vederea afișării sau pentru realizarea unei copii a obiectului inițial utilizând sisteme de “rapid prototyping” și CNC.

Sistemele de digitizare 3D există la ora actuală într-o mare varietate de soluții constructive bazate pe diferite principii. Sistemele de digitizare 3D au ca rezultat obținerea coordonatelor 3D ale unor puncte de pe suprafața obiectelor care, prin utilizarea unor aplicații software adecvate pot fi utilizate la generarea suprafețelor.

În funcție de modul de utilizare a modelului obținut prin digitizare, precizia de măsurare cerută pentru punctele de pe suprafața obiectului este diferită. Atunci când modelul este utilizat doar pentru vizualizare, cum este cazul în industria cinematografică, nu se impune o precizie deosebită, important fiind ca imaginea obiectului să “arate bine”. În fabricație, mașinile de măsurare în coordonate sau sistemele de “rapid prototyping” au nevoie de informații complete și precise. Dacă o zonă a suprafeței obiectului este incompletă sau lipsește, este imposibilă realizarea acelei părți a obiectului.

Majoritatea digitizoarelor 3D utilizează metode de determinare a coordonatelor punctului de pe suprafața obiectului digitizat prin metode non-contact sau cu ajutorul unor palpatoare.

Sistemele de digitizare 3D cu palpator au cele mai scăzute prețuri, dar impun acționare manuală. Aceasta poate constitui un dezavantaj, dar metoda prezintă și numeroase avantaje, printre care pot fi menționate: posibilitatea preluării doar a datelor geometrice caracteristice ale obiectului, insensibilitatea la gradul de reflexie al suprafeței, inexistența “conurilor de umbră”. Conurile de umbra sunt zone care nu pot fi scanate datorită imposibilității de proiectare a spotului luminos pe acea zonă de suprafață sau datorită faptului că nu este posibilă preluarea imaginii spotului luminos de către senzor. Ca soluții constructive se pot aminti digitizoare 3D cu structura mecanismului generator de traiectorie de tipul TTT derivate din mașinile de măsurat în coordonate și cele cu structura mecanismului generator de traiectorie format din cuple cinematice de rotație derivate din mecanismul generator de traiectorie al roboților. În timp ce digitizoarele cu cuple cinematice de translație au în majoritatea cazurilor acționare cu motoare electrice, digitizoarele cu cuple cinematice de rotație au acționare preponderent manuală. Digitizoarele din ultima categorie pot avea în unele cazuri o cuplă cinematică de translație pentru mărirea spațiului de lucru.

Sistemele de digitizare non-contact permit preluarea coordonatelor 3D ale unui număr mare de puncte de pe suprafața obiectului cu o viteză ridicată, până la ordinul sutelor de mii pe secundă, la o înaltă rezoluție. Dintre sistemele de digitizare non-contact, cele mai răspândite sunt cele bazate pe scanare laser. Calculul coordonatelor punctelor 3D de pe suprafața obiectului digitizat se realizează prin triangulație. O diodă laser iluminează suprafața dintr-o anumită direcție și poziția spotului luminos este determinată cu ajutorul unui senzor dintr-o altă direcție. Precizia determinării coordonatelor punctelor 3D prin această metodă depinde de mărimea suprafeței spotului luminos, precizia sistemului optic și a părții mecanice a sistemului și caracteristicile obiectului digitizat. Caracteristicile obiectului digitizat care influențează precizia digitizării sunt: gradul de reflexie al suprafeței, culoarea obiectului, muchiile ascuțite, etc. Datorită dezvoltării puternice a sistemelor de

scanare laser, prețul acestora este în continuă scădere. Totodată, cererea mare pe piață a acestor sisteme determină apariția a numeroase firme care produc aceste sisteme, într-o mare gamă constructivă.

Din studiul dinamicii domeniului digitizării 3D pe parcursul elaborării tezei se constată o dezvoltare accelerată a construcției digitizoarelor 3D și a aplicației acestora. În anexa 1 se observă gama diversificată a tipurilor de digitizoare 3D și numărul mare de firme producătoare. Dacă inițial digitizoarele 3D au fost utilizate în domenii de vârf precum industria aerospațială și cea a construcției de automobile, actual se constată o tendință de pătrundere a aplicațiilor în domenii ale industriei ușoare și în sectoare neindustriale.

3. Modelarea suprafețelor

3.1 Introducere

Modelul este o reprezentare simplificată a caracteristicilor esențiale ale unor obiecte sau procese. Modelarea geometrică 3D servește la obținerea unei imagini în trei dimensiuni a obiectului modelat. Modelarea geometrică 3D se realizează cu modele de sârmă, de suprafață sau de volum în funcție de scopul urmărit.

Modelul de suprafață definește atât muchiile obiectului cât și suprafețele sale și este util la când nu este necesar nivelul de detaliere a proprietăților fizice oferit de modelul de volum, dar sunt totuși necesare funcțiile de ascundere, umbrire și prezentare realistă pe care structurile de sârmă nu le furnizează. Modelul de suprafață este utilizat pentru generarea programelor pentru realizarea fizică a obiectelor utilizând mașini de prelucrare cu CNC. Utilizarea preponderentă a digitizoarelor 3D este cea de obținerea informațiilor referitoare la forma și dimensiunile obiectelor. Pe baza informațiilor furnizate de digitizoarele 3D, concretizate în coordonate 3D ale unor puncte individuale, șiruri de puncte sau nori de puncte, programe soft adecvate realizează modelul de suprafață al obiectului existent fizic.

3.2 Curbe de formă liberă

Majoritatea metodelor de generare a suprafețelor pornesc de la curbe. În cazul curbilor de formă liberă, forma curbei este aproximată printr-un set de puncte aflate pe curbă iar un program calculează celelalte puncte necesare pentru trasarea curbei. Acest set de puncte poate fi obținut cu ajutorul digitizorului 3D. Metodele de modelare a curbilor de formă liberă sunt cele bazate pe interpolare liniară și cele bazate pe netezirea poligonului format de un set de puncte date.

În cazul curbilor bazate pe interpolare liniară, se caută o funcție continuă și netedă care în anumite puncte (numite puncte de control sau “de sprijin”) ia valori date [SAV97]. Suplimentar, se introduce condiția ca două porțiuni de curbă succesive să realizeze un anumit grad de netezime. Netezimea se exprimă prin continuitatea funcțiilor care definesc două porțiuni de curbă succesive. Din punct de vedere matematic se pune condiția ca primele n derivate a celor două funcții ce se întâlnesc să fie continue în acel punct. Continuitatea de ordinul 0 asigură continuitatea pozițiilor. În cazul în care derivatele de ordinul întâi a celor două funcții sunt continue într-un punct se asigură continuitatea pantei, iar pentru continuitatea derivatelor de ordinul doi se asigură continuitatea curburii. Pentru majoritatea aplicațiilor sunt utilizate curbe spline cubice la care polinoamele care descriu curba între două puncte succesive sunt de grad maxim trei, indiferent de numărul punctelor. În cazul curbilor spline realizate pornind de la puncte obținute prin digitizare, cu cât numărul de puncte este mai mare reprezentarea este mai exactă.

Curbele bazate pe “netezirea” poligonului format de un set de puncte date sunt curbe de aproximare. În acest caz, curba nu trece prin toate punctele date, aceste puncte având rolul de a caracteriza forma și poziția curbei.

Curbele Bézier sunt curbe de aproximare la care se pune condiția ca doar primul și ultimul punct al curbei să coincidă cu punctele date, celelalte puncte servind la controlul formei curbei. Dezavantajul curbelor Bézier este faptul că la creșterea numărului de puncte de control crește și gradul polinoamelor necesare. Această situație poate apare frecvent în cazul obținerii punctelor de control a curbei prin digitizare.

Curbele B-spline îmbină avantajele curbelor spline cubice cu cele ale curbelor Bézier. Ca și în cazul curbelor Bézier, curbele B-spline sunt curbe de aproximare descrise prin puncte de control. Diferența constă în faptul că descrierea curbelor se face prin funcții polinomiale definite pe porțiuni, ceea ce le conferă proprietatea de control local. Segmentele de curbă B-spline sunt descrise prin polinoame de grad 2 sau 3, gradul fiind independent de numărul punctelor de control.

Forma curbei B-spline este definită de relația [SAV97]:

$$p(t) = p_0 \cdot B_{0,k}(t) + p_1 \cdot B_{1,k}(t) + \dots + p_n \cdot B_{n,k}(t)$$

unde: p_0, \dots, p_n sunt punctele de control ale curbei;

$B_{0,k}(t), \dots, B_{n,k}(t)$ sunt funcțiile de bază B-spline, definite recursiv:

$$B_{i,1}(t) = 1 \text{ dacă } t_i < t < t_{i+1} \text{ (} t_n < t < t_{n+1} \text{) dacă } i = n$$

$$B_{i,k}(t) = 0 \text{ în caz contrar;}$$

$$B_{i,k}(t) = \frac{(t-t_i) \cdot B_{i,k-1}(t)}{t_{i+k-1} - t_i} + \frac{(t_{i+k} - t) \cdot B_{i+1,k-1}(t)}{t_{i+k} - t_{i+1}};$$

$$t_i = 0 \text{ dacă } i < k;$$

cu: $t_i = i - k + 1$ dacă $k \leq i \leq n$; $i = 0, \dots, n + k$.

$$t_i = n - k + 2 \text{ dacă } i > n;$$

Ordinul k al curbei este independent de numărul n al punctelor de control. Aproximarea cu ajutorul curbelor B-spline este implementată în majoritatea sistemelor CAD. Dacă nodurile sunt uniform distribuite pe curbă se obțin curbe B-spline uniforme iar dacă se alege o altă distribuție se obțin curbe B-spline neuniforme. Curbele B-spline neuniforme permit o flexibilitate sporită în obținerea formei curbelor. În cazul în care vectorii punctelor de control sunt exprimate ca rapoarte de polinoame se obțin curbele B-spline raționale. Prin aplicarea simultană a unei distribuții neuniforme a nodurilor și exprimarea vectorilor de control în coordonate omogene se obțin curbele B-spline neuniforme raționale (NURBS). Modelul NURBS este în prezent una din formele matematice cel mai des utilizate pentru descrierea suprafețelor în interiorul modelatoarelor geometrice.

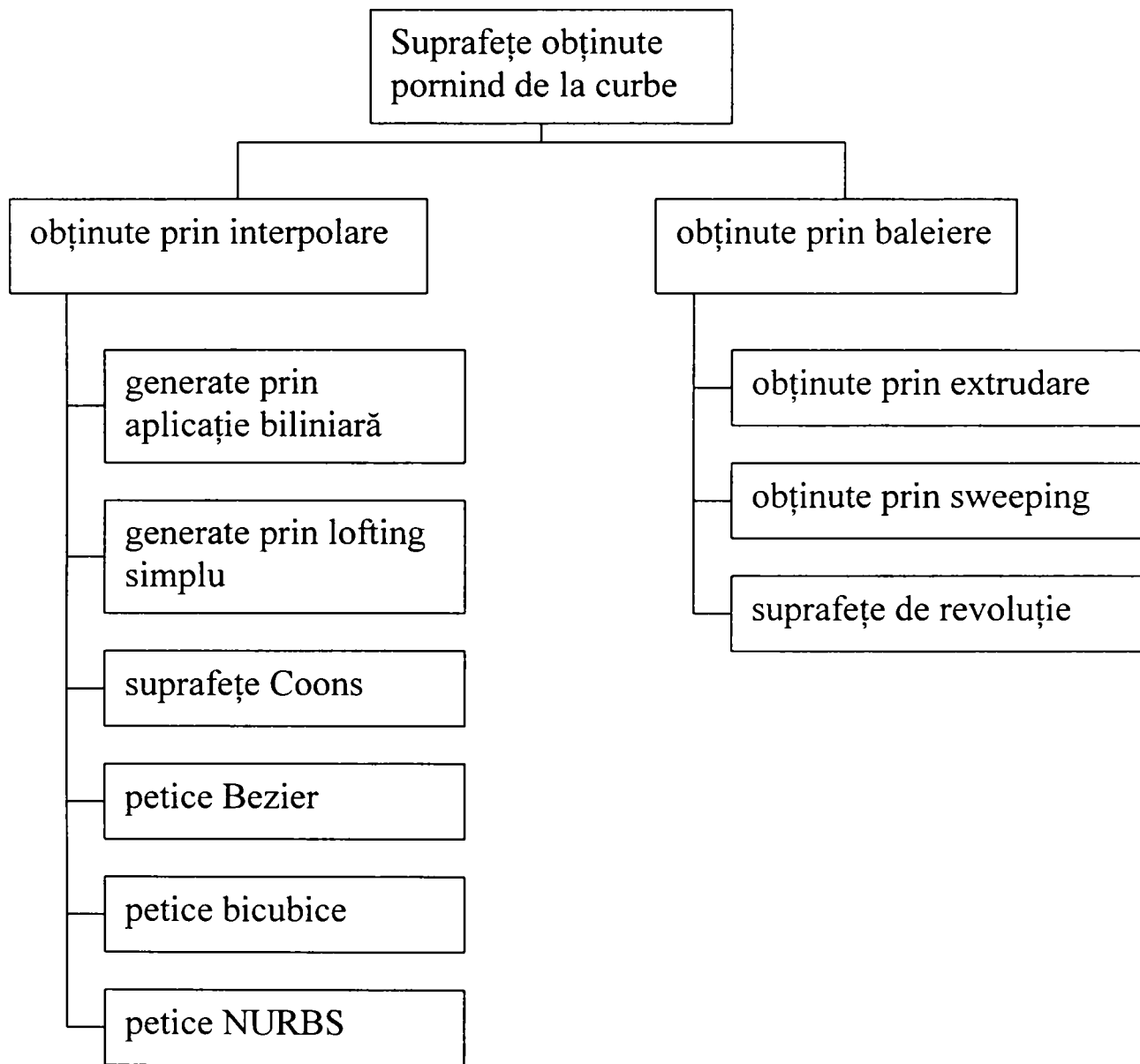
3.3 Obținerea de suprafețe pornind de la curbe

În proiectarea asistată de calculator sunt utilizate două clase de suprafețe și anume suprafețe fațetate și petice. Modelele fațetate utilizează poligoane plane inclusiv pentru reprezentarea unor suprafețe curbe. Aceste suprafețe curbe sunt discretizate în modele fațetate poligonale cu un grad de aproximare acceptat. Pentru realizarea unei aproximări cât mai bune sunt utilizate de obicei poligoane (fațete)

triunghiulare. Suprafețele obținute pe baza informațiilor obținute prin scanare laser sunt în majoritatea cazurilor suprafețe fațetate.

Peticele (în limba engleză “Patches”) realizează o descriere geometrică a unor suprafețe strâmbe în spațiu. Peticele sunt descrise de obicei parametric, fiecare coordonată fiind tratată ca o funcție de doi parametri.

O clasificare a suprafețelor din punct de vedere a modului de generare este prezentată în figura 3.3.1. Pentru ca suprafețele să fie realizabile fizic, de exemplu prin prelucrări de așchiere, este necesară respectarea unor condiții de coerență, evitarea autointersectării și continuitate. Coerența implică orientarea corespunzătoare a normalelor la suprafață de-a lungul muchiilor comune.



3.3.1 Clasificarea suprafețelor din punct de vedere a modului de generare

Suprafața biliniară este o suprafață obținută pornind de la 4 puncte. Punctele Q ale peticului rezultat utilizând punctele $P_{00}, P_{01}, P_{10}, P_{11}$ pot fi calculate parametric pe baza ecuației:

$$Q(u,v) = \begin{Bmatrix} 1-u & u \end{Bmatrix} \cdot \begin{bmatrix} P_{00} & P_{01} \\ P_{10} & P_{11} \end{bmatrix} \cdot \begin{Bmatrix} 1-v \\ v \end{Bmatrix}$$

Un exemplu de suprafață biliniară este prezentată în figura 3.3.2.

Prin lofting se realizează potrivirea unei porțiuni de suprafață pe două curbe parametrice limitatoare (frontiere) utilizând interpolarea între două funcții parametrice opuse utilizate pentru sprijinirea pe cele două frontiere [SAV97]. Forma suprafeței rezultate prin lofting între două curbe depinde de modul de parametrizare a celor două curbe. Operația de lofting permite trecerea lină de la o curbă limită la cealaltă (fig. 3.3.3). Prin generalizare, operația de lofting se poate aplica la generarea unei suprafețe care interpolează o familie de curbe parametrice paralele. Metoda se aplică de exemplu la definirea formei exterioare a cocii vaselor maritime sau la definirea profilului fuselajului aeronavelor. Indicarea curbelor profilelor poate fi realizată cu ușurință cu ajutorul unui digitizor 3D cu palpator.

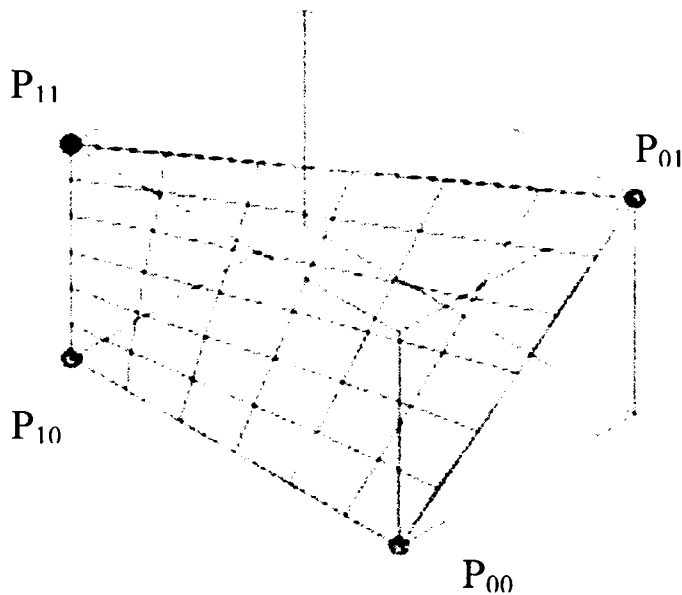


Fig. 3.3.2. Suprafață generată prin aplicație biliniară

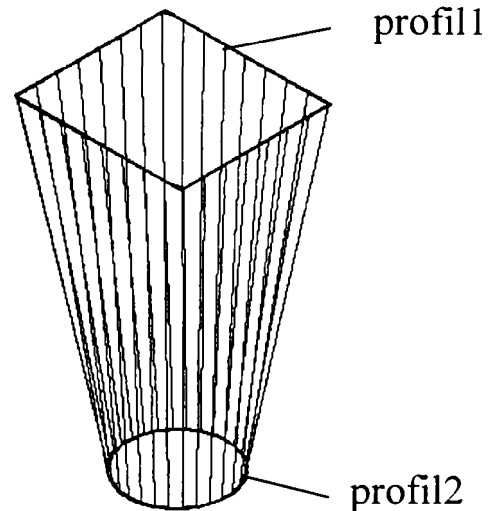


Fig. 3.3.3. Suprafață generată prin lofting

Suprafețele Coons se bazează pe un cadru construit din două familii de curbe care se intersectează (fig 3.3.4). Pentru o mai mare generalizare se lucrează cu curbe parametrice, distingând începutul și sfârșitul curbelor din curbele transversale prin definirea lor în termeni de parametri u respectiv v . Punctele Q ale peticului rezultat utilizând perechile de curbe limită $C(0, u)$, $C(1, u)$, $C(v, 0)$ și $C(v, 1)$ se pot calcula parametric din ecuația :

$$Q(u,v) = \begin{Bmatrix} 1-u & u & 1 \end{Bmatrix} \begin{bmatrix} -P_{00} & -P_{01} & C(0,v) \\ -P_{10} & -P_{11} & C(1,t) \\ C(s,0) & C(s,1) & 0 \end{bmatrix} \begin{Bmatrix} 1-v \\ v \\ 1 \end{Bmatrix}$$

Suprafețele Coons sunt utilizate în special în cazurile în care caracteristicile critice ale suprafeței sunt curbele limitatoare iar în interior se permite o anumită flexibilitate a formei suprafeței.

Suprafețele Bézier sunt suprafețe de aproximare și sunt definite printr-un poliedru caracteristic. Vârfurile poliedrului determină două familii de curbe Bézier care aproximează suprafața.

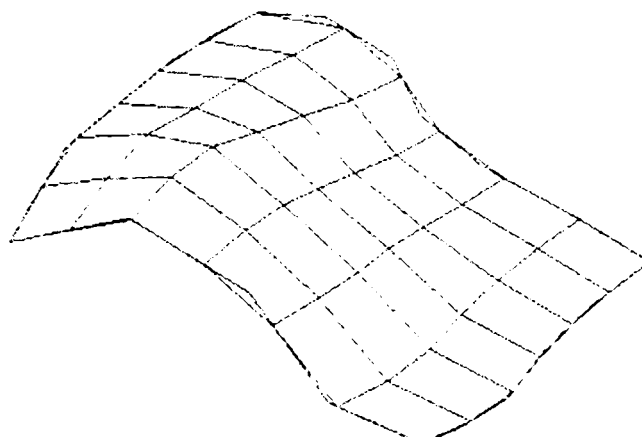


Fig. 3.3.4 Petic de suprafață Coons

Ecuția parametrică de definire a unei suprafețe Bézier este de forma:

$$Q(u, v) = \sum_{i=0}^m \sum_{j=0}^n P_{ij} B_{im}(u) B_{jn}(v), \quad 0 \leq u, v \leq 1$$

unde P_{ij} sunt vârfurile poliedrului caracteristic iar $B_{im}(u)$ și $B_{jn}(v)$ sunt polinoame Bernstein. Pentru a defini o suprafață Bézier se utilizează 4 puncte de control corespunzătoare valorilor $(0,0)$, $(0, 1)$, $(1, 0)$, $(1,1)$ ale parametrilor u și v și în plus alte 12 puncte de control prin intermediul cărora se precizează tangentele la suprafață, rezultând 16 puncte de control. Suprafețele Bézier trec prin cele patru puncte de colț (P_{00} , P_{01} , P_{10} , P_{11}) și în general nu trec prin celelate puncte de control. Problema continuității a 2 suprafețe Bézier conexe de-a lungul unei margini se rezolvă prin asigurarea coliniarității pentru 4 perechi de câte 3 puncte .

Peticele bicubice folosesc aproximări cubice (de tip spline) iar punctele Q se calculează pe baza ecuației:

$$Q(u, v) = \sum_{i=0}^3 \sum_{j=0}^3 a_{ij} u^i v^j$$

unde coeficienții a_{ij} depind de coordonatele vârfurilor peticului, de tangente și de curburile în vârfuri.

Principala utilizare a peticelor B-spline raționale (NURBS) constă în reprezentarea unitară a cvadricelor și suprafețelor strâmbe, în scopul omogenizării structurii datelor ce reprezintă entitățile geometrice în baza de date a unui modelator geometric. Reprezentarea exactă a cercurilor și arcelor de cerc cu ajutorul funcțiilor B-spline raționale permit, prin combinarea generatoarelor B-spline raționale cvadrice neuniforme cu toate celelalte feluri de directoare, modelarea cu exactitate a suprafețelor de revoluție.

Operația de extrudare este o baleiere rectilinie caracterizată de o formă profil. Profilul reprezintă o curbă plană, deschisă sau închisă. Traectoria de baleiere este un segment de dreaptă perpendicular pe planul care conține profilul. Extrudarea este una

din cele mai simple modalități de generare a suprafețelor. Dacă profilul original este închis, rezultatul va fi o suprafață închisă lateral.

Obținerea suprafețelor prin sweeping reprezintă o generalizare a operației de extrudare (baleiere) de-a lungul unei curbe arbitrare în loc de o dreaptă normală. O altă generalizare, ilustrată în figura 3.3.5 este cea în care profilul poate varia în funcție de poziția pe linia de sweeping.

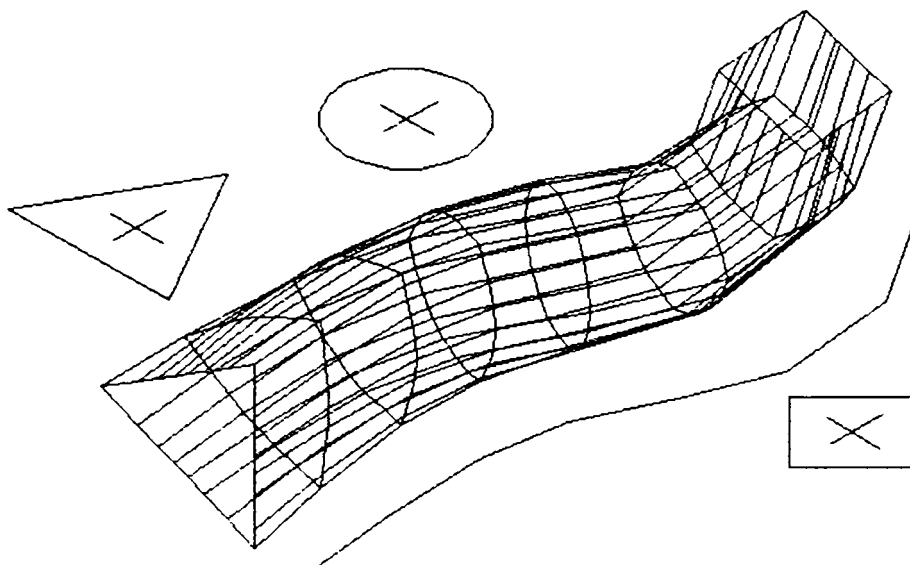


Fig. 3.3.5 Suprafață obținută prin sweeping generalizat

Suprafețele de revoluție se obțin prin baleiere circulară. Operația necesită specificarea unei curbe profil și a unei axe. Unghiul de rotire a profilului poate avea valori mai mici de 360° .

Un alt tip de suprafețe cu largi utilizări în proiectarea asistată de calculator sunt suprafețele offset. Acest tip de suprafețe este foarte important în cazul operațiilor de prelucrare prin așchiere, realizarea pieselor din mase plastice prin injecție și în cazul digitizării cu ajutorul digitizoarelor 3D cu palpator. În cea mai simplă formă, suprafața offset este translația unei suprafețe cu o valoare dată pe direcția normalei. În cazul digitizoarelor 3D cu vârful palpatorului de formă sferică punctul caracteristic al palpatorului se deplasează în timpul digitizării pe o suprafață situată la o distanță egală cu raza vârfului față de suprafața digitizată.

4. Stabilirea obiectivelor tezei de doctorat

4.1 Digitizorul 3D, o noutate tehnică cu multe aplicații

În urma cercetărilor bibliografice se constată că domeniul digitizării 3D are o dinamică deosebită. Tehnica digitizării 3D, utilizată pe scară largă doar în ultimul deceniu, își lărgeste continuu sfera aplicațiilor. Tendința de trecere în mediul virtual a unei părți însemnate din partea de concepție și pregătire a fabricației impune crearea și manipularea unor modele geometrice 3D a viitoarelor produse. O situație des întâlnită e cea de reproiectare a unui produs care există ca model fizic dar pentru care nu există un model în format CAD. În ambele cazuri, utilizarea digitizoarelor 3D crește eficiența proceselor. Deoarece digitizarea 3D este o tehnică nouă, aplicațiile preponderente în momentul actual sunt în industriile de vârf, aerospațială și a construcției de automobile. Asemeni multor noutăți tehnice, utilizarea digitizării 3D difuzează din aceste industrii în alte domenii.

Datorită actualității domeniului digitizării 3D, materialul bibliografic este relativ insuficient în comparație cu cel despre tehnici consacrate. Descrieri ale diferitelor tipuri de digitizoare 3D se găsesc în abundență pe Internet, dar o sistematizare foarte clară și prezentarea aspectelor teoretice și a principiilor constructive care stau la baza construcției digitizoarelor 3D nu este disponibilă.

4.2 Stabilirea relațiilor dintre construcția robotului și construcția digitizorului

Structura digitizoarelor 3D este în mare parte asemănătoare cu cea a roboților. Digitizorul 3D cu palpator cu acționare manuală are structura asemănătoare cu cea a roboților cu topologie serială. Principala diferență este faptul ca digitizorul nu posedă sisteme de acționare proprii. Autorul își propune sa-și aducă contribuția la realizarea unor modele matematice a digitizoarelor 3D cu palpator utilizând metode și tehnici specifice roboticii. Pe baza acestor modele matematice se va încerca tratarea problemelor legate de precizia digitizoarelor. Deoarece digitizoarele 3D sunt utilizate ca instrumente metrologice, precizia este unul dintre condițiile cele mai importante. Prin determinarea erorilor sistematice și efectuarea corecțiilor necesare se poate îmbunătăți precizia acestora. Autorul își propune fundamentarea unei metode de determinare a valorilor necesare pentru realizarea corecțiilor pe baza măsurării cu ajutorul digitizorului a unor puncte a căror poziție este determinată anterior prin alte mijloace.

4.3 Contribuții la fundamentarea modalităților de utilizare a digitizoarelor 3D în proiectarea asistată de calculator

Datele prelevate cu ajutorul digitizoarelor 3D, sub forma coordonatelor unor puncte 3D, a unor șiruri de puncte sau unor nori de puncte, pot fi utilizate pentru generarea modelelor geometrice a unor obiecte existente fizic. Elaborarea unor algoritmi eficienți pentru crearea modelelor de suprafață pornind de la curbe definite

prin puncte obținute prin digitizare este un domeniu în care se mai pot aduce îmbunătățiri.

Industria producătoare de mobilă este un domeniu în care digitizarea 3D este deocamdată utilizată în mică măsură. Autorul își propune să aducă contribuții la infuzia mai rapidă a tehnicii digitizării 3D în acest sector cu tradiție în România. Prin aceasta autorul valorifică experiența dobândită anterior pregătirii în cadrul doctoratului prin activarea la o întreprindere producătoare de mobilă. Se încearcă elaborarea unei aplicații software care să mărească eficiența unor procese implicate în realizarea produselor de mobilier și totodată să asigure realizarea unor prezentări realiste a acestora în scopuri comerciale. Procesele vizate se referă la : realizarea modelului geometric 3D și a desenelor de execuție a unui produs oferit de client ca model, elaborarea desenelor de execuție a șabloanelor și sculelor speciale pentru diferite piese ce intră în componența mobilierului, verificarea formei și dimensiunilor produselor pe parcursul ciclului de fabricație și la final, realizarea unor modele de suprafață a produselor în vederea prezentării pe Internet și pentru campanii de marketing. Pentru creșterea eficienței de modelare generarea modelelor CAD a produselor se va aborda prin prisma unei tehnologii de grup orientată spre modelare.

Prelevarea informațiilor despre forma și dimensiunile modelului fizic se consideră a fi obținute cu ajutorul un digitizor 3D cu palpator.

4.4 Contribuții la fundamentarea modalităților utilizării digitizoarelor 3D în programarea roboților

Aplicația cea mai ușor de implementat este cea de utilizare a digitizorului 3D ca lanț de programare. Dar posibilitatea de implementare pe scară largă este limitată de faptul că necesită o structură identică pentru robot și digitizor.

Programarea off-line poate beneficia de serviciile digitizoarelor 3D prin indicarea cu ajutorul acestora a situațiilor succesive care trebuie atinse de robot în timpul sarcinii de lucru și stocarea acestora . Un ajutor indirect pentru programarea off-line poate fi oferit de utilizarea digitizorului 3D pentru preluarea informațiilor dimensionale și de formă referitoare la periferia robotului. Autorul își propune abordarea ambelor aspecte din acest aliniat.

4.5 Contribuții la fundamentarea utilizării digitizoarelor 3D în alte domenii tehnice sau mixte

Utilizarea digitizoarelor 3D în domeniul artistic este rezultatul convergenței celor două direcții. Atât arta cât și digitizarea 3D lucrează cu forme. Utilizarea tehnicilor de digitizare, eliberează artistul de unele limitări fizice ajutându-l la materializarea imaginației sale. Autorul își propune să aducă contribuții la fundamentarea cerințelor și a posibilităților de realizare a unei aplicații de sculptură virtuală cu ajutorul digitizorului 3D.

4.6 Verificarea celor prezentate anterior prin aplicații concrete

Pentru verificarea aspectelor prezentate în paragrafele 4.1 – 4.5 la nivel conceptual teoretic teza are ca obiectiv realizarea unui digitizor 3D cu palpator cu acționare manuală și realizarea unor aplicații software pentru:

- calibrarea digitizorului 3D;
- programarea off-line a roboților cu ajutorul digitizorului 3D;
- modelarea elementelor de mobilier pe baza datelor obținute prin digitizare
- sculptură virtuală.

5. Contribuții teoretice

5.1 Determinarea erorilor de măsurare a coordonatelor punctelor prelevate cu digitizorul 3D cu palpator

5.1.1 Modelarea preciziei digitizoarelor 3D

Utilizarea digitizoarelor ca instrumente metrologice și pentru modelarea geometriei suprafețelor impune o precizie ridicată de realizare a acestora. Totuși, funcționarea digitizoarelor 3D este afectată de erori. Deoarece digitizoarele 3D cu palpator cu acționare manuală sunt acționate de operatorul uman iar deplasarea relativă a elementelor cuplelor cinematice se realizează cu viteze și accelerații reduse, erorile dinamice se pot neglija.

Problema preciziei digitizorului se pune în mod diferit pentru digitizorul aflat în faza de proiectare sau de optimizare față de un digitizor existent. În cel de-al doilea caz, creșterea preciziei se realizează prin calibrare (vezi paragraful 5.1.5). Principalele erori care afectează măsurarea coordonatelor punctelor de pe suprafața obiectelor sunt erorile de incrementare a traductoarelor de rotație, erorile datorate impreciziilor de execuție și montaj a elementelor și erorile datorate deformațiilor termice a elementelor.

Determinarea distribuției erorilor în raport cu spațiul de lucru este utilă pentru determinarea zonelor de eroare minimă pentru situarea obiectului care se digitizează. Prin determinarea distribuției erorilor în spațiul de lucru se poate genera o “hartă” de precizie și se pot determina suprafețele cu aceeași valoare a erorii în spațiul de lucru. Pentru un set de valori a parametrilor variabili care definesc situarea relativă a elementelor digitizorului q_1, \dots, q_n sau pentru o anumită situație a palpatorului se pot estima erorile de măsurare. În cadrul metodologiei propuse pentru estimarea erorilor de măsurare într-un anumit punct din spațiul de lucru se consideră principalii factori care contribuie la mărimea erorii și anume erorile de incrementare în traductor și abaterile de lungime ale elementelor. Pentru un anumit punct din spațiul de lucru se determină volumul de toleranță. Metoda este utilizată în faza de proiectare și optimizare a digitizorului.

5.1.2 Determinarea volumului de toleranță

Volumul de toleranță reprezintă spațiul în care se poate afla punctul caracteristic al digitizorului 3D pentru un anumit set de parametrii variabili care definesc poziția relativă a elementelor cuplelor cinematice. Pentru determinarea volumului de toleranță se propun două variante. Prin utilizarea primei variante se determină volumul de toleranță cunoscându-se pozițiile relative ale elementelor cuplelor cinematice ale digitizorului iar cu ajutorul variantei a doua se determină volumul de toleranță cunoscându-se poziția punctului caracteristic al digitizorului.

Prima variantă utilizează modelul geometric direct iar a doua variantă modelul geometric invers. În a doua variantă se determină prin intermediul modelului geometric invers parametrii variabili q_1, \dots, q_n apoi se apelează la prima variantă

pentru determinarea volumului de toleranță. Metodologia propusă presupune realizarea prealabilă a modelului geometric direct al digitizorului. Pentru determinarea cu ușurință a relațiilor de calcul care definesc modelul geometric direct se poate utiliza aplicația utilitară ModPeSiR realizată de autor. Aplicația ModPeSiR este bazată pe conceptul de perechi de sisteme de referință PeSiR [KOV2001].

Se descrie în continuare metodologia de estimare a volumului de toleranță a cărei schemă logică este prezentată în fig. 5.1.1.

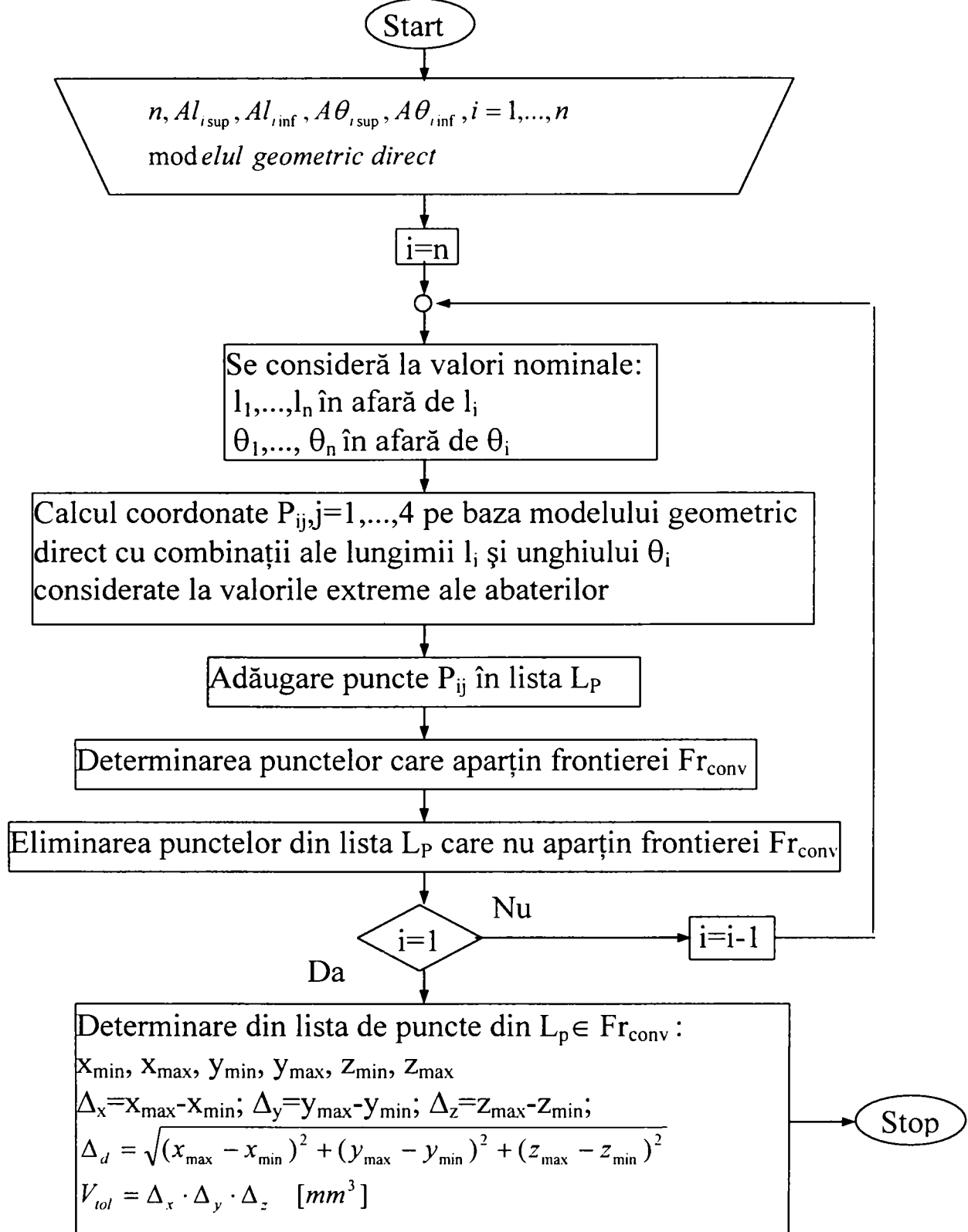


Fig. 5.1.1 Schema logică a algoritmului pentru estimarea volumului de toleranță

Se consideră pentru exemplificarea metodei un digitizor cu cinci cuple cinematice de rotație. Se cunosc lungimile nominale ale elementelor notate cu l_i și valorile unghiurilor θ_i cu $i = 1, \dots, 5$. De asemenea, se cunosc abaterile superioară Al_{isup} la și inferioară Al_{iinf} față de lungimea nominală pentru fiecare element respectiv abaterile de la valoarea nominală a lui θ_i datorate incrementului unghiular al traductoarelor de rotație $A\theta_{isup}$ și $A\theta_{iinf}$. Fie n numărul cuplelor cinematice ale digitizorului. În exemplul prezentat $n = 5$. Se consideră toate elementele $n-1$ ca având lungimea egală cu valoarea nominală în afară de elementul n a cărei lungime variază în limitele extreme $l_n - Al_{ninf}$ și $l_n + Al_{nsup}$. Toate unghiurile θ_i cu $i=1, \dots, n-1$ se consideră la valoarea nominală iar unghiul θ_n se consideră la limitele extreme $\theta_n - A\theta_{ninf}$ și $\theta_n + A\theta_{nsup}$.

Din modelul geometric direct se determină vectorul de poziție al punctului palpat a cărei expresie este o funcție de lungimi de elemente și unghiuri deci $\vec{r}_p = \vec{r}_p(l_i, \theta_i)$. Prin considerarea limitelor extreme a lungimii elementului n și a valorilor extreme a unghiului θ_n rezultă patru combinații posibile (Tab. 5.1.1).

Lungime	Unghi	Punct	Vector de poziție
$l_n - Al_{ninf}$	$\theta_n - A\theta_{ninf}$	P_1	\vec{r}_1
$l_n - Al_{ninf}$	$\theta_n + A\theta_{nsup}$	P_2	\vec{r}_2
$l_n + Al_{nsup}$	$\theta_n - A\theta_{ninf}$	P_3	\vec{r}_3
$l_n + Al_{nsup}$	$\theta_n + A\theta_{nsup}$	P_4	\vec{r}_4

Tab. 5.1.1 Combinații ale extremelor lungimilor și unghiurilor pentru cupla cinematică n

Pentru fiecare din aceste patru combinații se poate calcula câte un vector de poziție rezultând patru puncte $P_j, j=1, \dots, 4$ (fig. 5.1.2). Coordonatele celor patru puncte P_j se stochează într-o listă notată L_p . Deoarece lungimile arcelor P_1P_2 și respectiv P_3P_4 au valori foarte mici în raport cu razele $O_5P_1=O_5P_2$ respectiv $O_5P_3=O_5P_4$ se poate aproxima suprafața reprezentată hașurat în fig. 5.5.1 cu un patrulater cu colțurile $P_j, j=1, \dots, 4$. Se observă că această aproximare este rațională doar în cazul determinării volumului de toleranță și nu este acceptabilă în cazul determinării spațiului de lucru al digitizorului. Deci metoda propusă nu se poate aplica pentru determinarea spațiilor de lucru a digitizoarelor sau a robotilor.

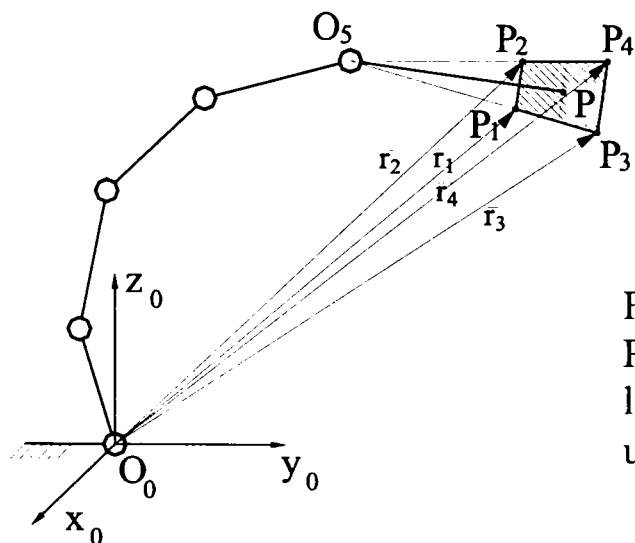


Fig.5.1.2.Poziția vârfului palpatorului P_i când se consideră extremele lungimilor elementului n și extremele unghiului θ_n

În continuare se consideră că toate lungimile elementelor au valori nominale în afară de elementul $n-1$ și toate unghiurile au valorile nominale în afară de unghiul care definește poziția relativă a elementelor cuplei cinematice $n-1$. Analog cu raționamentul pentru cupla cinematică n vor exista 4 combinații posibile ale extremelor lungimii elementului $n-1$ și a unghiului θ_{n-1} . Pentru fiecare din punctele P_j din lista L_P se aplică matricea de transformare corespunzătoare rotirii în jurul axei cuplei cinematice $n-1$ în care se utilizează cele patru combinații ale extremelor lungimii elementului $n-1$ și a unghiului θ_{n-1} . Se observă că numărul punctelor determinate este de patru ori numărul punctelor aflate în lista L_P (fig. 5.1.3).

Se înțelege prin frontieră polinilia2D sau suprafața care limitează norul de puncte. Se notează cu Fr_{conv} frontiera convexă care încadrează norul de puncte. Se observă că în cazul în care norul de puncte este situat în același plan frontiera convexă va fi o polilinie închisă plană, în caz contrar frontiera va fi o suprafață formată din fațete. Algoritm pentru determinarea frontierei convexe a unui nor de puncte este prezentat în paragraful 5.1.4.

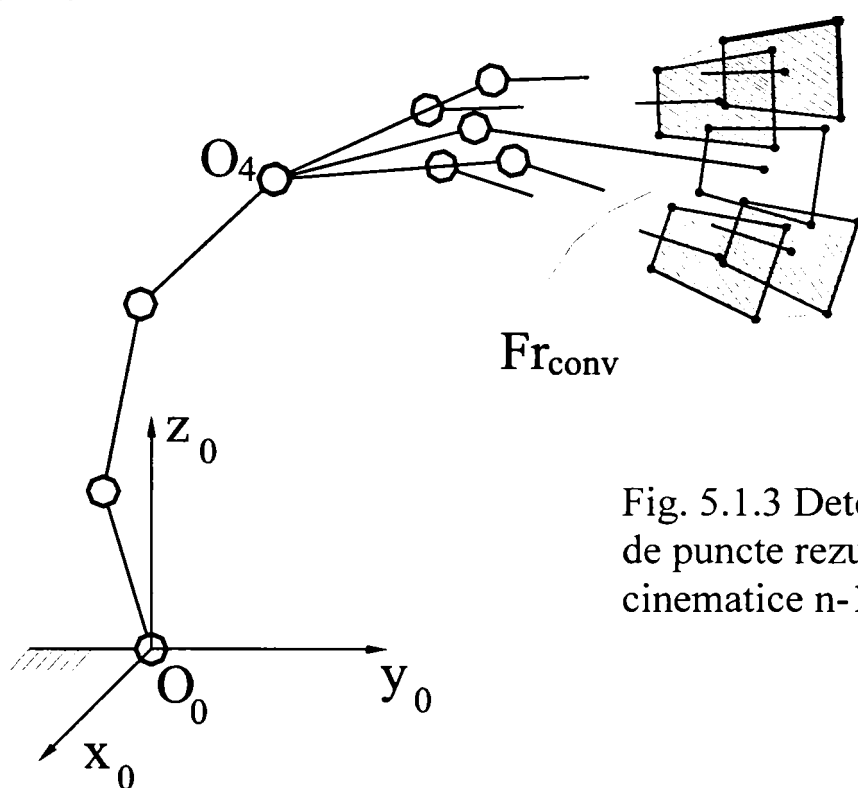


Fig. 5.1.3 Determinarea limitelor mulțimii de puncte rezultate prin considerarea cuplei cinematice $n-1$

Prin repetarea aceluiași raționament pentru toate cele n cuplele cinematice se observă că numărul punctelor aflate în final în lista L_P este: $NrP_l = 4^n$. Acest număr poate fi redus în cazul în care după fiecare considerare a erorilor pentru o cuplă cinematică se aplică un algoritm de eliminare a punctelor care nu aparțin frontierei norului de puncte din lista L_P . Utilizarea acestei eliminări este utilă deoarece scade considerabil timpul de determinare a volumului încadrat de frontieră în urma luării în considerare a tuturor cuplelor cinematice. Deci în cazul a cinci cuple cinematice, volumul de toleranță va fi exprimat printr-un nor de puncte format din maxim $NrP_l = 4^5 = 1024$ puncte. Pentru determinarea valorii numerice a volumului de toleranță V_{tol} , una din metode este secționarea suprafeței care reprezintă frontiera volumului de toleranță cu plane paralele la o distanță dată. Prin aproximarea volumului fiecărei zone limitată de

frontieră și de două plane consecutive și însumarea acestora se poate obține o aproximare bună a valorii numerice a volumului de toleranță.

O variantă care estimează volumul de toleranță cu o marjă mai mare de eroare determină paralelipipedul în care se încadrează norul de puncte. Prin considerarea laturilor paralelipipedului paralele cu axele sistemului de coordonate fix atașat digitizorului problema devine foarte simplă. Este suficientă sortarea punctelor după valorile coordonatelor x , y și respectiv z . Valorile minime ale celor 3 coordonate și valorile maxime ale coordonatelor formează două puncte care sunt capetele unei diagonale a paralelipipedului. Utilizând dimensiunile paralelipipedului se poate reprezenta vizual într-o forma intuitivă printr-un cod de culori valoarea abaterilor punctului măsurat corespunzător unui anumit set de valori pentru parametri variabili θ_i , $i=1, \dots, n$. Prin reprezentarea cu ajutorul culorilor a volumului calculat al paralelipipedului pentru diferite seturi de parametri variabili utilizatorul poate să-și formeze o imagine de ansamblu a distribuției erorilor în spațiul de lucru al digitizorului. Metodele prezentate în acest paragraf pot fi utilizate cu succes la determinarea volumului de toleranță și în cazul roboților cu topologie serială.

5.1.3 Elaborarea hărții de precizie

Harta de precizie este o reprezentare a distribuției erorilor de măsurare a digitizorului în raport cu spațiul de lucru al acestuia. Pentru elaborarea hărții de precizie se consideră doar mecanismul generator de traiectorie care, datorită lungimilor mai mari a elementelor în raport cu mecanismul de orientare are o contribuție mai însemnată la valoarea totală a erorii.

Metoda prezentată presupune realizarea prealabilă a modelului geometric invers al mecanismului generator de traiectorie al digitizorului. Algoritmul a cărei schemă logică este prezentată în fig. 5.1.4 începe prin indicarea de către utilizator a două puncte C_1 și C_2 care reprezintă colțurile diagonale opuse ale unui paralelipiped care acoperă spațiul de lucru al digitizorului. Se indică valoarea pașilor p_x , p_y , p_z de divizare a laturilor paralelipipedului pe fiecare direcție. Se obține o matrice tridimensională de puncte. Pentru fiecare punct se aplică modelul geometric invers și se determină parametrii variabili care definesc poziția relativă a elementelor cuplelor cinematice ale digitizorului. Dacă există soluții raționale, înseamnă că punctul caracteristic se găsește în spațiul de lucru și se aplică metoda prezentată la paragraful anterior pentru determinarea volumului de toleranță.

Dimensiunea volumului de toleranță măsurat în mm^3 se stochează într-o matrice cu 3 dimensiuni cu câte un element pentru fiecare punct caracteristic considerat în spațiul de lucru al digitizorului. Această matrice poate fi reprezentată grafic cu un cod de culori pentru a da o imagine asupra distribuției erorilor în spațiul de lucru. De asemenea, este utilă reprezentarea bidimensională a unor secțiuni plane prin spațiul de lucru sau reprezentarea în 3 dimensiuni a suprafețelor formate din puncte pentru care dimensiunea volumului de toleranță V_{tol} măsurat în mm^3 are o valoare cuprinsă într-un interval dat.

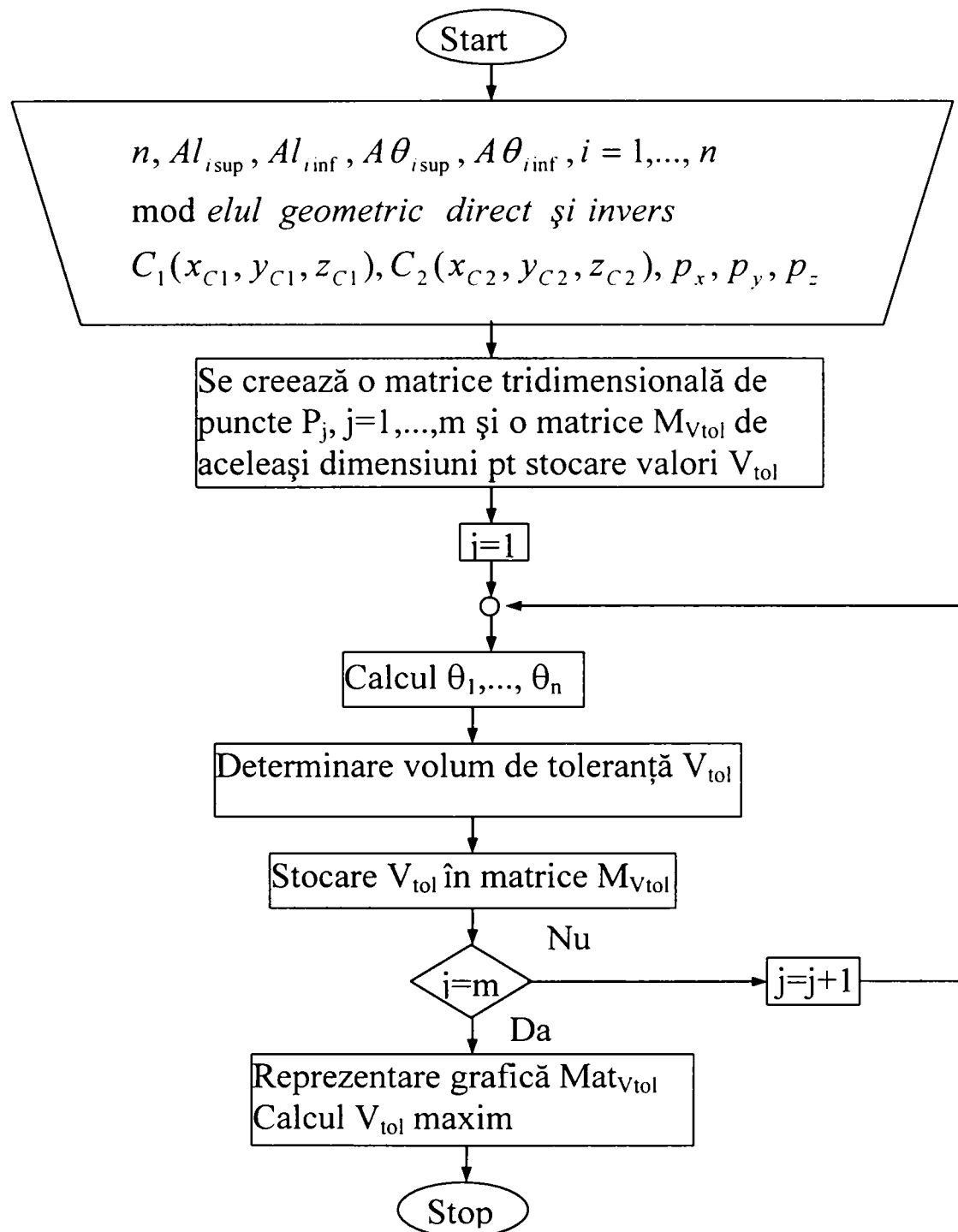


Fig. 5.1.4. Schema logică a algoritmului pentru generarea hărții de precizie

5.1.4 Determinarea frontierei convexe a unui nor de puncte

Metoda propusă utilizează curbele de nivel pentru determinarea suprafeței de frontieră a norului de puncte. Se sortează punctele din norul de puncte în funcție de coordonata z rezultând o valoare z_{\min} și o valoare z_{\max} . Se notează cu h diferența $z_{\max} - z_{\min}$. Această distanță se împarte într-un număr de diviziuni indicate de utilizator sau propuse implicit de către aplicația care implementează algoritmul. Se notează cu n numărul de diviziuni și cu p_z diferența dintre 2 diviziuni succesive. Se consideră un număr n de plane perpendiculare pe direcția z (fig. 5.1.5) care intersectează axa z în punctele P_i , $i=1, \dots, n$. Coordonatele z ale punctelor P_i se calculează cu formula : $z_{P_i} = z_{\min} + i \cdot p_z$.

Pentru fiecare din plane se formează o mulțime M_i care conține punctele situate între planul ce trece prin P_i și planul ce trece prin P_{i+1} .

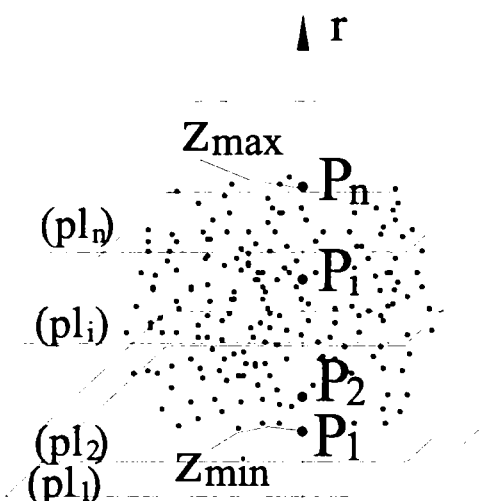


Fig. 5.1.5. Planele separatoare a punctelor din norul de puncte

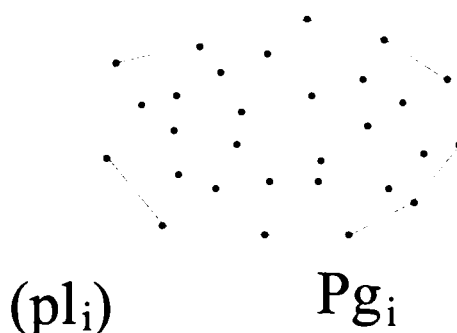


Fig 5.1.6. Poligonul care limiteaza proiecția punctelor în planul pl_j

Fiecare mulțime M_i de puncte se proiectează în planul ce trece prin P_i prin considerarea coordonatelor z tuturor punctelor din M_i coordonata z a punctului P_i notată z_{P_i} . Pentru fiecare mulțime M_i se determină un poligon de arie minimă care limitează punctele situate în M_i (fig. 5.1.6). Fiecare din poligoanele P_{g_i} obținute se divide într-un număr m egal de puncte. Suprafața de frontieră este cu aproximație suprafața care interpolează cele i poligoane (fig. 5.1.7). Metoda prezentată anterior poate aproxima frontiera în cazul în care distribuția punctelor este aproximativ uniformă în interiorul norului de puncte.

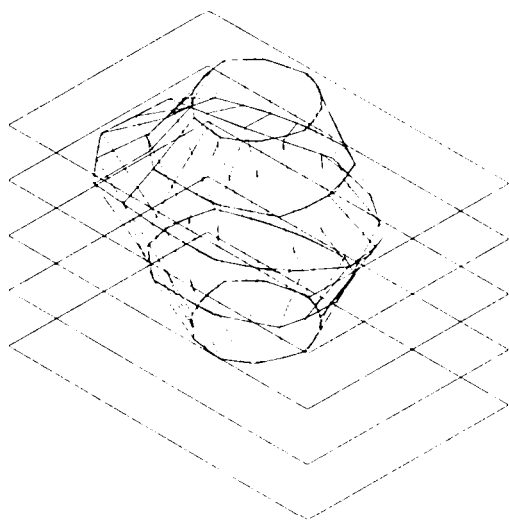


Fig. 5.1.7. Suprafața de frontieră a norului de puncte

5.1.5 Calibrarea digitizorului cu palpator cu acționare manuală

Calibrarea digitizorului se referă la identificarea experimentală a corelațiilor dintre citirile traductoarelor cu ajutorul cărora se determină poziția relativă a elementelor cuplurilor cinematice conducătoare și poziția reală a vârfului palpatorului. Calibrarea digitizorului implică determinarea diferențelor dintre geometria nominală și cea reală referitoare la lungimea elementelor și situarea relativă a axelor succesive,

determinarea diferenței de temperatură față de valoarea nominală. Pe baza datelor geometrice se corectează schema cinematică a digitizorului pentru a reflecta geometria reală și se implementează algoritmul de calcul al poziției vârfului palpatorului bazat pe modelul geometric direct. În programul de calcul a poziției vârfului palpatorului, lungimile elementelor sunt considerate ca variabile și pe baza diferenței dintre temperatura nominală și cea de la momentul respectiv se calculează lungimea reală a elementelor digitizorului. Deoarece vârful palpatorului se poate schimba în majoritatea cazurilor în funcție de aplicația concretă pentru a permite o măsurare precisă este necesară calibrarea palpatorului după fiecare schimbare a vârfului. Calibrarea palpatorului se referă la determinarea distanței de la vârful palpatorului la axa ultimei cuple cinematice.

În continuare se descrie metoda de determinare a geometriei reale a digitizorului datorată erorilor de fabricație. Se consideră un digitizor Dig_1 cu n cuple la care toate cuplele sunt de rotație. Pentru a determina orientarea relativă a axelor cuplelor cinematice în spațiu se imobilizează pe rând toate cuplele cinematice în afară de una. Se notează cupla cinematică care nu se imobilizează cu c_i . Pentru determinarea poziției vârfului palpatorului se utilizează un alt digitizor Dig_2 calibrat și de o clasă de precizie superioară celui care se calibrează. Vârful palpatoarelor digitizorului care se calibrează respectiv digitizorului cu care se măsoară poziția vârfului sunt menținute în contact prin intermediul unui cuplaj fix (fig. 5.1.8). Prima axa a cărei orientare se determină este axa n . Se împiedică deplasarea relativă a elementelor cuplelor cinematice c_1, c_2, \dots, c_{n-1} prin aplicarea unor constrângeri de natură mecanică. Se rotește ultimul element al digitizorului Dig_1 în jurul axei cuplei cinematice c_n și se înregistrează cu ajutorul digitizorului Dig_2 3 poziții ale punctului P care reprezintă punctul de contact a vârfurilor celor două palpatoare aparținând digitizoarelor Dig_1 și respectiv Dig_2 .

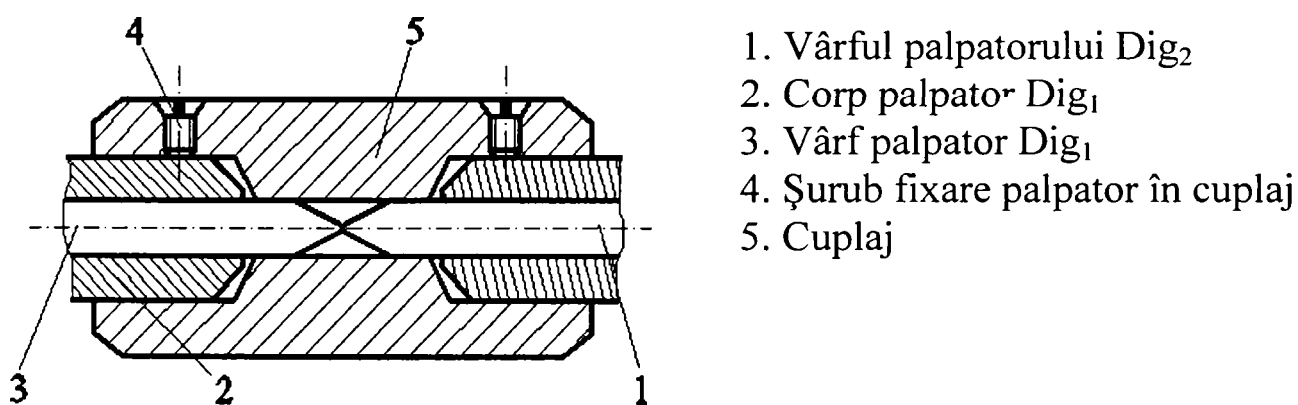


Fig. 5.1.8. Vârful palpatoarelor digitizoarelor aduse în contact

Cupla c_i fiind o cuplă de rotație, cele trei poziții ale punctului P notate P_{1i}, P_{2i}, P_{3i} se situează pe un cerc. Dreapta perpendiculară pe planul care conține cercul determinat de cele trei puncte și care trece prin centrul cercului este dreapta suport a axei cuplei cinematice și se notează în continuare cu d_{ci} (fig. 5.1.10).

La digitizoarele 3D axa cuplei cinematice n este perpendiculară pe direcția axei palpatorului (fig. 5.1.9) Distanța de la vârful palpatorului la dreapta suport a cuplei

cinematice c_n este egală cu raza cercului determinat de P_{1i}, P_{2i}, P_{3i} și poate fi utilizată pentru calibrarea palpatorului. Se notează cu $O_i(x_i, y_i, z_i)$ centrul cercului ce trece prin punctele P_{1i}, P_{2i} , și P_{3i} .

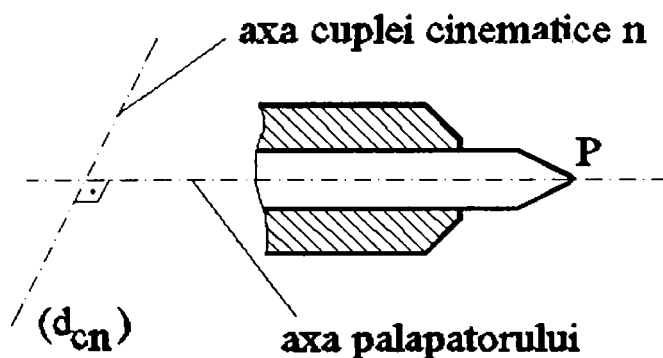


Fig. 5.1.9 Axa cuplei cinematice n perpendiculară pe direcția axei palpatorului

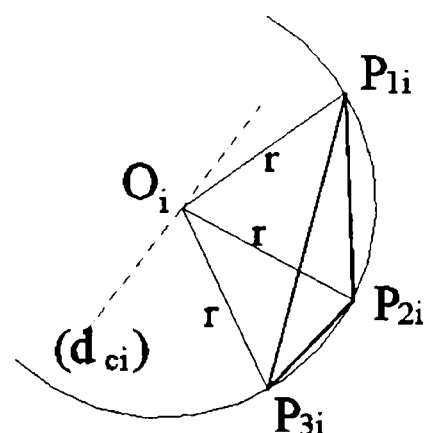


Fig. 5.1.10. Determinarea centrului și razei cercului ce trece prin P_1, P_2, P_3

Dacă O_i este centrul cercului atunci punctele P_{1i}, P_{2i} și P_{3i} sunt situate la distanța r_i de O_i unde prin r_i s-a notat raza cercului ce trece prin punctele P_{ij} , $j=1, \dots, 3$. Deci coordonatele punctului O_i vor satisface sistemul de ecuații:

$$\begin{cases} (x_{P_{1i}} - x_i)^2 + (y_{P_{1i}} - y_i)^2 + (z_{P_{1i}} - z_i)^2 - r_i^2 = 0 \\ (x_{P_{2i}} - x_i)^2 + (y_{P_{2i}} - y_i)^2 + (z_{P_{2i}} - z_i)^2 - r_i^2 = 0 \\ (x_{P_{3i}} - x_i)^2 + (y_{P_{3i}} - y_i)^2 + (z_{P_{3i}} - z_i)^2 - r_i^2 = 0 \end{cases} \quad (5.1.1)$$

Prin împărțirea fiecărei ecuații a sistemului cu 2 și înmulțirea cu -1 sistemul devine:

$$\begin{cases} x_{P_{1i}} \cdot x_i + y_{P_{1i}} \cdot y_i + z_{P_{1i}} \cdot z_i - d = 0 \\ x_{P_{2i}} \cdot x_i + y_{P_{2i}} \cdot y_i + z_{P_{2i}} \cdot z_i - d = 0 \\ x_{P_{3i}} \cdot x_i + y_{P_{3i}} \cdot y_i + z_{P_{3i}} \cdot z_i - d = 0 \end{cases} \quad (5.1.2)$$

unde s-a notat:

$$d = \frac{x_i^2 + y_i^2 + z_i^2 + x_{P_{1i}}^2 + y_{P_{1i}}^2 + z_{P_{1i}}^2 - r_i^2}{2} \quad (5.1.3)$$

Prin extragerea expresiei pentru d în fiecare ecuație a sistemului 5.1.2 și combinarea a câte 2 ecuații se obține:

$$\begin{cases} (x_{P_{2i}} - x_{P_{1i}})x_i + (y_{P_{2i}} - y_{P_{1i}})y_i + z_i(z_{P_{2i}} - z_{P_{1i}})z_i = 0 \\ (x_{P_{3i}} - x_{P_{1i}})x_i + (y_{P_{3i}} - y_{P_{1i}})y_i + z_i(z_{P_{3i}} - z_{P_{1i}})z_i = 0 \\ (x_{P_{3i}} - x_{P_{2i}})x_i + (y_{P_{3i}} - y_{P_{2i}})y_i + z_i(z_{P_{3i}} - z_{P_{2i}})z_i = 0 \end{cases} \quad (5.1.4)$$

Se notează cu D determinantul sistemului:

$$D = \begin{vmatrix} x_{P_{2i}} - x_{P_{1i}} & y_{P_{2i}} - y_{P_{1i}} & z_{P_{2i}} - z_{P_{1i}} \\ x_{P_{3i}} - x_{P_{1i}} & y_{P_{3i}} - y_{P_{1i}} & z_{P_{3i}} - z_{P_{1i}} \\ x_{P_{3i}} - x_{P_{2i}} & y_{P_{3i}} - y_{P_{2i}} & z_{P_{3i}} - z_{P_{2i}} \end{vmatrix} \quad (5.1.5)$$

Prin regula lui Cramer se determină coordonatele x_i, y_i și z_i a punctului O_i .

Raza cercului ce trece prin cele trei puncte P_{1i}, P_{2i} și P_{3i} este egală cu distanța de la centrul cercului la unul din puncte:

$$r_i = \sqrt{(x_i - x_{P_{1i}})^2 + (y_i - y_{P_{1i}})^2 + (z_i - z_{P_{1i}})^2} \quad (5.1.6)$$

Valoarea r_i poate fi utilizată pentru calibrarea palpatorului.

Se determină orientarea dreptei d_{ci} . Dreapta suport a axei c_i este dreapta care trece prin O_i și are direcția perpendiculară pe planul determinat de dreptele concurente $P_{2i}P_{1i}$ și $P_{2i}P_{3i}$. Se notează cu $\bar{v}_1(l_1, m_1, n_1)$ vectorul director al dreptei $P_{2i}P_{1i}$ și cu $\bar{v}_2(l_2, m_2, n_2)$ vectorul director al dreptei $P_{2i}P_{3i}$. Componentele vectorilor directori se determină cu ajutorul coordonatelor punctelor care definesc dreptele:

$$l_1 = x_{2i} - x_{1i}; \quad m_1 = y_{2i} - y_{1i}; \quad n_1 = z_{2i} - z_{1i} \quad (5.1.7)$$

$$l_1 = x_{2i} - x_{3i}; \quad m_1 = y_{2i} - y_{3i}; \quad n_1 = z_{2i} - z_{3i} \quad (5.1.8)$$

Se determină vectorul director $\bar{u}(u_x, u_y, u_z)$ al dreptei d_{ci} . Dreapta d_{ci} este perpendiculară pe dreptele definite de vectorii directori $\bar{v}_1(l_1, m_1, n_1)$ și respectiv $\bar{v}_2(l_2, m_2, n_2)$ deci are ca vector director produsul vectorial al dreptelor definite de cei doi vectori directori: $\bar{u} = \bar{v}_1 \times \bar{v}_2$. Componentele versorului sunt:

$$u_x = m_1 \cdot n_2 - n_1 \cdot m_2 \quad (5.1.9)$$

$$u_y = n_1 \cdot l_2 - l_1 \cdot n_2 \quad (5.1.10)$$

$$u_z = l_1 \cdot m_2 - m_1 \cdot l_2 \quad (5.1.11)$$

Ecuțiile canonice ale dreptei d_{ci} determinate de punctul $O_i(x_i, y_i, z_i)$ și de vectorul director $\bar{u}(u_x, u_y, u_z)$ sunt:

$$\frac{x - x_i}{u_x} = \frac{y - y_i}{u_y} = \frac{z - z_i}{u_z} \quad (5.1.12)$$

În urma determinării ecuației dreptei suport a cuplei cinematice i , se imobilizează deplasarea relativă a tuturor elementelor cuplelor cinematice în afară de elementele cuplei cinematice c_{i-1} și se determină ecuația dreptei suport d_{ci-1} a axei cuplei c_{i-1} utilizând relațiile de calcul prezentate anterior. Se procedează analog pentru toate cuplele cinematice ale digitizorului. Se consideră câte 2 cuple cinematice succesive și se determină unghiul în spațiu și ecuația perpendicularei comune. Dacă se constată diferențe față de schema cinematică nominală se realizează corecțiile schemei cinematice și se reface modelul geometric direct. Pe baza noului model geometric direct se modifică programul de determinare a poziției vârfului palpatorului. În cazul în care există abateri la orientarea relativă a mai multor axe a cuplelor cinematice a digitizorului se are în vedere creșterea complexității programului de calcul a poziției vârfului palpatorului și implicit a timpului de calcul. Dacă timpul de calcul devine prea mare pentru a permite digitizarea în timp real soluția este corectarea construcției mecanice a digitizorului.

Temperatura la care se realizează digitizarea poate să varieze în timpul măsurării, de exemplu datorită conducerii manuale a palpatorului de către operatorul uman, variațiilor temperaturii mediului de lucru sau frecărilor în cuple. Majoritatea sistemelor de digitizare 3D realizează o măsurare continuă a temperaturii elementelor în câteva puncte și realizează o corecție a lungimii elementelor. Dacă temperatura elementului variază de la temperatura nominală t_0 la $t_0 + \Delta_t$, atunci variația lungimii la temperatura t va fi: $\Delta_{l_i} = \alpha \cdot l_{0i} \cdot \Delta_t$, unde s-a notat cu l_{0i} lungimea elementului la temperatura nominală iar α reprezintă coeficientul de dilatare liniară al materialului

din care este realizat elementul digitizorului. Valoarea $l_{oi} + \Delta_{li}$ este utilizată de către programul de calcul a coordonatelor punctului situat pe vârful palpatorului bazat pe modelul geometric direct. Când digitizorul se proiectează fără sistem pentru compensarea temperaturii, la determinarea erorilor prin metoda de la paragraful 5.1.2, volumul de toleranță se determină prin adăugarea la abaterile de construcție a abaterilor datorate diferențelor de temperatură.

5.2. Estimarea rezoluției necesare a traductoarelor incrementale de rotație pentru atingerea unei precizii impuse de măsurare a coordonatelor punctelor digitizate

Estimarea rezoluțiilor necesare a traductoarelor este utilă în faza incipientă de proiectare a digitizorului când se încearcă diferite variante. Se urmărește ca pentru o anumită structură să poată fi determinate rapid rezoluțiile aproximative a traductoarelor incrementale de rotație necesare și eventual să se verifice soluțiile alese de utilizator. Proiectul fiind în faza incipientă nici dimensiunile elementelor nu sunt deja stabilite și este util ca odată cu traductoarele incrementale de rotație să se poată încerca diverse combinații de lungimi de elemente.

Metoda propusă se bazează pe definirea anterioară a structurii și existența modelului geometric direct. Deoarece în cazul digitizării 3D interesează în special poziția vârfului palpatorului și mai puțin orientarea dreptei caracteristice, din modelul geometric direct se vor considera relațiile care definesc poziția punctului caracteristic. Influența erorilor de incrementare a traductoarelor cuplurilor cinematice a mecanismului generator de traiectorie are o influență mai mare asupra preciziei de măsurare și ca urmare în metoda propusă pentru estimarea automată a rezoluției maxime a traductoarelor se vor considera doar aceste erori.

Utilizatorul precizează erorile maxime de măsurare pe fiecare direcție $(\Delta_x, \Delta_y, \Delta_z)$ sau eroarea maximă de măsurare $\Delta_p = \sqrt{\Delta_{px}^2 + \Delta_{py}^2 + \Delta_{pz}^2}$. Este creată o matrice tridimensională de puncte în spațiu cu un pas constant pe direcții paralele cu axele care acoperă spațiul de lucru al robotului (vezi paragraful 5.1.3). Pentru fiecare punct P_c din matricea tridimensională, considerat punct caracteristic al digitizorului se alege un punct P_e aflat la distanța Δ_p de punctul P_c . Dacă au fost indicate valorile $\Delta_x, \Delta_y, \Delta_z$ atunci punctul P_e se alege la distanța dată față de P_c pe fiecare din cele 3 direcții: $x_{Pe} = x_{Pc} + \Delta_{px}$; $y_{Pe} = y_{Pc} + \Delta_{py}$; $z_{Pe} = z_{Pc} + \Delta_{pz}$.

Pentru fiecare din cele 2 poziții ale punctului caracteristic P_c și respectiv P_e se calculează pe baza modelului geometric invers parametrii variabili $\theta_1, \theta_2, \dots, \theta_n$. Se notează cu θ_{ic} valoarea parametrului variabil corespunzător cuplei cinematice i pentru punctul P_c și cu θ_{ie} valoarea parametrului variabil corespunzător cuplei cinematice i pentru punctul P_e . Pentru fiecare cuplă cinematică se determină rezoluția maximă: $rez_{\max} = \theta_{ie} - \theta_{ic}$.

Determinarea rezoluției maxime fiecare cuplă cinematică se repetă pentru toate punctele din matricea tridimensională a punctelor din spațiul de lucru al digitizorului.

Rezoluția maximă necesară pentru fiecare traductor incremental de rotație este valoarea maximă a rezoluției corespunzătoare cuplei cinematice rezultată în urma parcurgerii tuturor punctelor P_c din spațiul de lucru al digitizorului. Se alege o rezoluție mai mică decât cea calculată deoarece în calcul nu s-a ținut cont de erorile de incrementare corespunzătoare mecanismului de orientare și de celelalte surse de erori care influențează precizia de măsurare a digitizorului 3D.

Rezoluția traductoarelor incrementale de rotație corespunzătoare cuplelor cinematice a mecanismului generator traiectorie se aleg pe baza datelor furnizate automat de algoritm. Pentru mecanismul de orientare utilizatorul poate să aleagă traductoarele incrementale apoi se poate verifica dacă soluția aleasă satisface restricția de precizie minimă impusă. Se dau valori pentru toate variabilele care definesc poziția relativă a elementelor întregului mecanism și prin utilizarea metodei descrise la paragraful 5.1.2 se determină norul de puncte care definește volumul de toleranță. Se verifică dacă laturile patrulaterului care încadrează norul de puncte are laturile mai mici decât valorile Δ_x, Δ_y și Δ_z .

5.3. Calibrarea de ordinul zero. Determinarea matricii de transformare între două sisteme de referință pe baza coordonatelor a trei puncte exprimate în cele două sisteme de referință

Conlucrarea digitizorului 3D cu alte echipamente presupune corelarea sistemelor de referință atașate digitizorului și respectiv a echipamentelor în cauză. Calibrarea de ordinul zero urmărește corelarea parametrilor de situație relativă a componentelor. Folosirea calibrării de ordinul 0 este utilă de exemplu în cazul digitizării unui obiect al cărui volum nu se încadrează în spațiul de lucru al digitizorului fiind necesară deplasarea elementului fix al digitizorului față de obiect. Calibrarea de ordinul 0 este utilă și în cazul în care digitizorul 3D este utilizat pentru indicarea situațiilor efectorului final ale unui robot sau la realizarea modelului geometric al mediului robotului în vederea programării robotului în mediu virtual. Metoda propusă utilizează pentru determinarea matricii de transformare între două sisteme de referință (matricea de calibrare) coordonatele a trei puncte necoliniare exprimate în cele două sisteme. Se utilizează pentru determinarea matricii de calibrare facilitățile oferite de aplicația de proiectare asistată AutoCAD.

Un “bloc” în AutoCAD este o colecție de obiecte individuale, combinate într-un obiect mai mare. O caracteristică importantă a blocurilor din punct de vedere al metodei propuse este că fiecare bloc are atașat un sistem de referință propriu. Blocurile sunt generate în memoria sistemului de calcul. O instanță a blocului este o reprezentare în model a blocului din memorie. Când asupra instanțelor blocurilor se aplică transformări de translație sau rotație sistemul de referință atașat instanței blocului suferă aceleași transformări. Printr-o funcție AutoLisp (care este limbaj de programare în mediul AutoCAD) se poate obține matricea de situație a sistemului de referință atașat instanței blocului față de sistemul de referință al modelului AutoCAD.

Se notează cu SR_1 și respectiv SR_2 cele 2 sisteme de referință între se determină matricea de calibrare. Se cunosc coordonatele a trei puncte P_{i1} , cu $i = 1 \div 3$ exprimate în SR_1 și a trei puncte P_{i2} , cu $i = 1 \div 3$ exprimate în SR_2 . Punctul P_{i2} exprimat în SR_2 corespunde punctului P_{i1} exprimat în SR_1 . Schema logică a algoritmului este prezentată în figura 5.3.3.

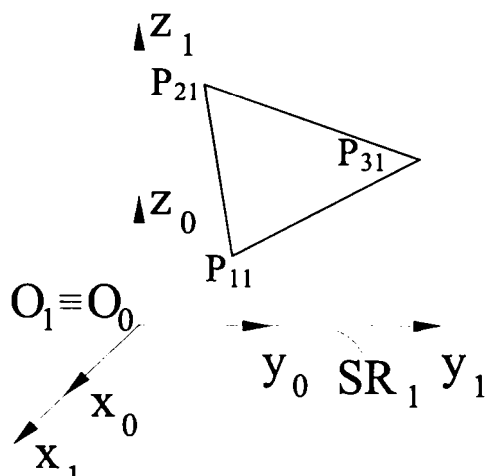


Fig. 5.3.1. Elementele geometrice care intră în componența bl_1

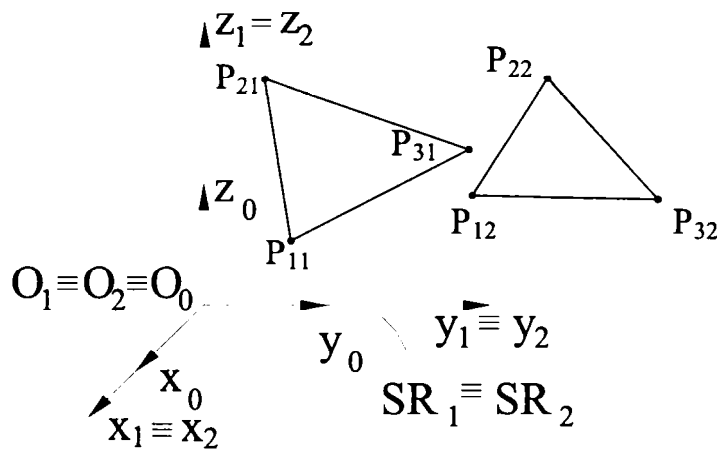


Fig. 5.3.2. Instanțierea blocurilor bl_1 și bl_2

Pentru construirea primului bloc se generează trei “linii” AutoCAD având ca și puncte de capăt $P_{11}P_{21}, P_{21}P_{31}$ și $P_{31}P_{11}$. Aceste trei linii formează un triunghi (fig. 5.3.1.). Se generează un bloc bl_1 constituit din cele trei linii iar ca punct de inserție al blocului se consideră originea sistemului de referință al modelului. Sistemul de referință SR_1 este atașat blocului bl_1 .

În mod similar este generat un al doilea bloc bl_2 pe baza coordonatelor punctelor P_{i2} cu $i = 1 \div 3$ căruia îi este atașat sistemul de referință SR_2 . Se instanțiază ambele blocuri cu indicarea punctului de inserție în originea sistemului de coordonate al modelului (fig. 5.3.2).

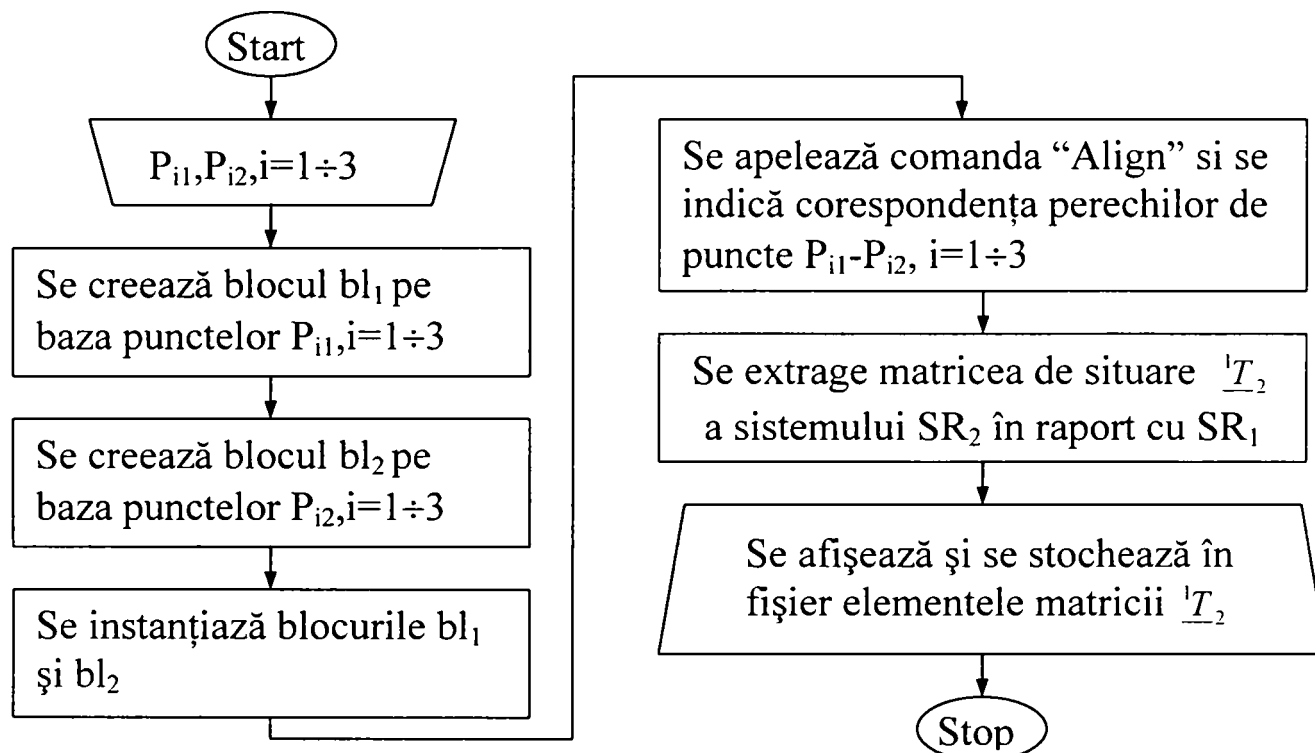
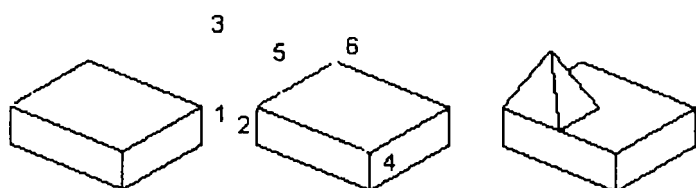


Fig. 5.3.3. Schema logică a algoritmului pentru determinarea matricii de calibrare

Există în AutoCAD comanda “Align” care permite alinierea a două obiecte prin indicarea corespondenței a 3 perechi de puncte. În figura 5.3.4 este prezentat un exemplu preluat din documentația programului AutoCAD versiunea 2000 pentru ilustrarea modului de funcționare al comenzii “Align” :

ALIGN Using Three Pairs of Points

- Specify first source point: *Specify a point (1)*
- Specify first destination point: *Specify a point (2)*
- Specify second source point: *Specify a point (3)*
- Specify second destination point: *Specify a point (4)*
- Specify third source point: *Specify a point (5)*
- Specify third destination point: *Specify a point (6)*



obiectul selectat șase puncte specificate rezultatul aplicării comenzii

Fig. 5.3.4. Ilustrarea modului de funcționare a comenzii “Align” preluat din documentația AutoCAD 2000

Alinierea obiectului prin utilizarea corespondenței celor trei perechi de puncte presupune următoarea succesiune de operații (notații din fig. 5.3.4): translatarea obiectului selectat din punctul 1 în punctul 2, rotirea obiectului selectat astfel ca dreapta ce trece prin punctele 1 și 3 să se suprapună cu dreapta ce trece prin punctele 2 și 4, rotirea obiectului selectat astfel ca dreapta ce trece prin punctele 3 și 5 să se alinieze cu dreapta ce trece prin punctele 4 și 6.

Se lansează comanda “Align” apoi se selectează instanța blocului bl_2 și se indică corespondența celor trei perechi de puncte. Prin aplicarea comenzii, triunghiul $P_{12}P_{22}P_{32}$ se suprapune peste triunghiul $P_{11}P_{21}P_{31}$ odată cu deplasarea sistemului de coordonate SR_2 atașat blocului bl_2 (fig. 5.3.5). Prin funcția AutoLisp “nentselp” se preia matricea 1T_2 de situare a sistemului de referință SR_2 față de sistemul de referință al modelului, această matrice de situare fiind matricea de calibrare a cărei determinare s-a urmărit.

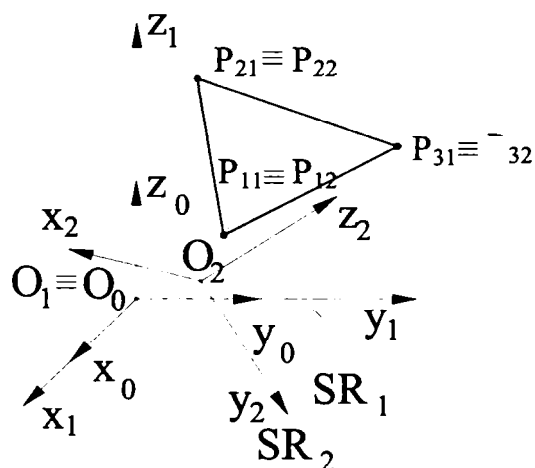


Fig. 3.3.5. Situația sistemului de referință SR_2 în urma utilizării comenzii “align”

Aplicația care implementează algoritmul prezentat în figura 5.3.3 utilizează limbajul VisualBasic și o subrutină AutoLisp. Ca date de intrare se specifică coordonatele celor 3 perechi de puncte în casete de text iar aplicația realizează automat toate operațiile necesare și afișează matricea de calibrare. Pe lângă afișare, componentele matricii de calibrare sunt salvate într-un fișier text pentru a fi utilizate de alte aplicații care necesită determinarea matricii de transformare între două sisteme de referință pe baza coordonatelor a trei puncte exprimate în cele două sisteme de referință.

5.4. Modelarea suprafețelor pornind de la informații despre coordonatele unor șiruri de puncte 3D obținute prin digitizare

5.4.1 Introducere

Digitizarea cu ajutorul digitizoarelor 3D cu palpator cu acționare manuală are avantajul unei flexibilități sporite în preluarea coordonatelor unor curbe formate din puncte situate pe suprafața obiectului și în controlul ulterior al formei suprafeței generate pornind de la aceste curbe. Digitizoarele 3D cu palpator au două moduri de lucru principale: preluarea informațiilor despre coordonatele unor puncte izolate respectiv preluarea informațiilor despre coordonatele unor șiruri de puncte. Prin șir de puncte se înțelege în continuare o mulțime de puncte rezultate prin înregistrarea continuă a coordonatelor carteziene a vârfului palpatorului la contactul acestuia cu suprafața care este digitizată. În cazul prelevării informațiilor despre coordonatele unor șiruri de puncte, problema care se pune este generarea unor suprafețe pornind de la aceste șiruri. Generarea suprafețelor pornind de la curbe limitatoare este o problemă rezolvată de către majoritatea sistemelor CAD și are la bază interpolarea Coons. În continuare, prin curbe limitatoare se vor înțelege polilinii 3D având vertexurile formate din șirurile de puncte rezultate prin digitizare. Vertexurile poliliniilor 3D pot fi considerate puncte de control ale curbelor. Contribuția autorului constă în conceperea algoritmilor și implementarea rutinelor necesare pentru procesarea informațiilor despre coordonate 3D a șirurilor de puncte și pregătirea acestora în formatul acceptat de sistemele CAD profesionale pentru generarea petecelor de suprafață.

5.4.2 Descrierea algoritmilor utilizați

5.4.2.1 Generalități

Algoritmul conceput de autor utilizează ca date de intrare șiruri de coordonate a unor puncte 3D și are ca date de ieșire curbele limitatoare pentru fiecare petec al suprafeței. Pentru îndeplinirea scopului este necesară parcurgerea următoarelor etape: identificarea punctelor de apropiere la distanță minimă ale curbelor, modificări fine asupra coordonatelor vertexurilor aflate la distanță minimă pentru a deveni intersecții reale, determinarea vertexurilor poliliniilor 3D care limitează fiecare petec.

Pentru facilitarea înțelegerii algoritmilor utilizați se impun unele precizări referitoare la modul de prelevare a datelor de intrare. Digitizorul utilizat pentru prelevarea coordonatelor punctelor de pe suprafața obiectului a cărui model de suprafață urmează a fi realizat este un digitizor 3D cu palpator care are în componență cuple de rotație și este condus prin impunerea situației manuale a ultimului element. Operatorul uman deplasează vârful palpatorului astfel încât acesta să păstreze contactul cu suprafața obiectului digitizat. Prin acționarea unui întrerupător situat pe mânerul digitizorului este comandată selectarea în modul de înregistrare continuă a coordonatelor. La începerea operației de digitizare, utilizatorul specifică distanța

minimă d_{\min} între 2 puncte 3D pentru ca acestea să fie înregistrate ca puncte distincte. Alegerea valorii d_{\min} depinde de precizia care se urmărește a corespondenței dintre modelul construit din datele obținute prin digitizare și suprafața obiectului real. Prin alegerea unei valori corespunzătoare pentru d_{\min} se evită stocarea unui număr prea mare de puncte apropiate care poate avea ca și consecință un timp crescut de efectuare a calculelor fără a aduce îmbunătățiri semnificative din punct de vedere al preciziei. Un șir de puncte este considerat o mulțime de puncte înregistrate între acționarea întrerupătorului pentru preluare continuă și eliberarea acestuia. În timpul digitizării efective se înregistrează indicele șirului de puncte împreună cu coordonatele pentru fiecare punct. În faza de digitizare efectivă este preferabil ca prin modul de prelevare a informațiilor despre coordonatele șirurilor de puncte să se urmărească realizarea unui caroiaj astfel încât fiecare petec să fie delimitat de puncte aparținând a patru șiruri, deoarece majoritatea sistemelor CAD utilizează petice Coons pentru generarea peticelor de suprafață pornind de la 4 curbe (sau polilinii 3D) limitatoare deci 4 puncte de intersecție (fig.5.4.1).

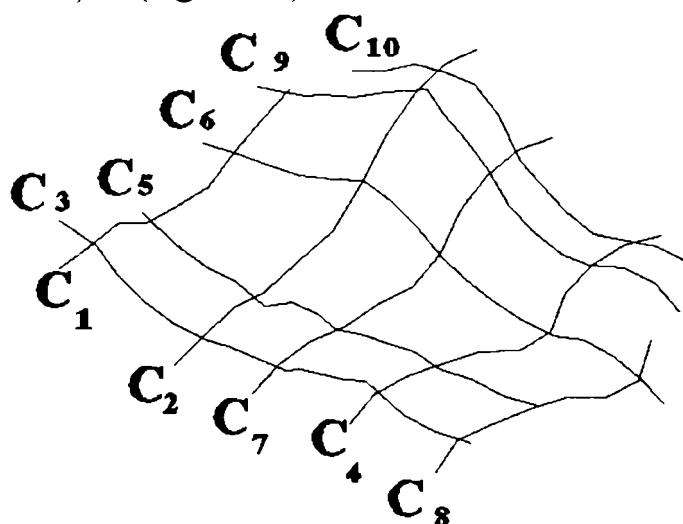


Fig. 5.4.1 Polilinii 3D generate din șiruri de puncte în timpul digitizării

În scopul asigurării unui feed-back corespunzător pentru acțiunile operatorului uman, în timpul digitizării, pe baza șirurilor de puncte digitizate sunt generate polilinii 3D care se pot vizualiza pe ecranul calculatorului. Poliliniile 3D obținute în urma procesului de digitizare nu au vertexuri comune din cauza imperfecțiunilor procesului de digitizare și a modului de dispunere a vertexurilor poliliniilor. Fiecare petec trebuie să aibă limite care se intersectează două câte două și de aceea este necesară utilizarea unui algoritm care să creeze punctele de intersecție reale.

5.4.2.2 Generearea punctelor de intersecție dintre poliliniile 3D formate pe baza informațiilor despre coordonatele șirurilor de puncte obținute prin digitizare

a) Necesitatea optimizării algoritmului

Pentru stabilirea poziției punctelor de intersecție este necesară să fie determinate vertexurile pentru care distanța dintre poliliniile este mai mică decât o distanță d_{\min} setată de utilizator. Două poliliniile pot să se intersecteze în mai multe puncte. Se consideră două poliliniile C_i și C_j având m și respectiv n vertexuri. Soluția cea mai simplă pentru determinarea mulțimii perechilor de vertexuri $V_k V_t$ cu $V_k \in C_i$ și $V_t \in C_j$ care se află la distanța $d, < d_{\min}$, ar fi calcularea acestei distanțe pentru fiecare pereche $V_k V_t$ cu $k = 1 \dots m-1$ și $t = 1 \dots n-1$. Se observă că numărul de distanțe care trebuie calculate în acest caz este $N_c = m \cdot n$. De exemplu dacă fiecare din cele 2 poliliniile are 1000 vertexuri numărul de distanțe care trebuie calculate este 10^6 . Dacă se ține cont că pentru formarea peticelor de suprafață este necesară determinarea intersecțiilor dintre oricare 2 poliliniile formate din șiruri de puncte distincte rezultă un număr inadmisibil de mare de calcule.

b) Reducerea numărului de calcule necesare pentru determinarea vertexurilor a două poliliniile aflate la o distanță mai mică decât o distanță d_{\min} dată

Una din metodele de reducere a numărului de calcule necesare este bazată pe luarea în considerare a distanței minime d_{\min} specificată de utilizator înainte de începerea procesului de prelevare a coordonatelor de pe suprafața obiectului digitizat. Schema logică a algoritmului este prezentată în figura 5.4.3. Se consideră determinarea vertexurilor $V_k \in C_i$ situate la o distanță $d < d_{\min}$ față de vertexul $V_t \in C_j$ (fig. 5.4.2). Pentru poliliniile C_i și C_j se va forma o listă notată $LdMin$ care conține perechi de vertexuri aparținând poliliniilei C_i și respectiv C_j pentru care distanța este mai mică decât d_{\min} .

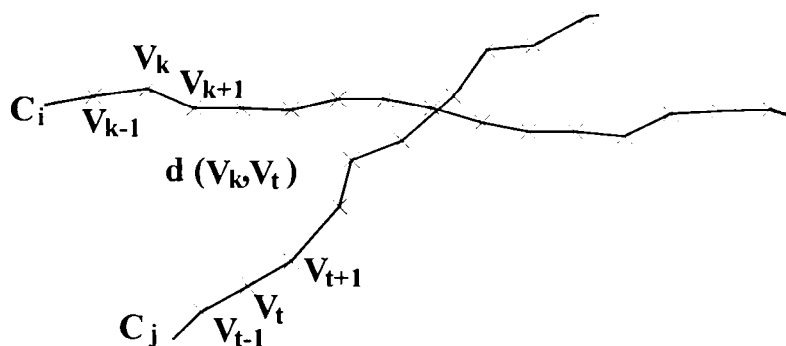


Fig. 5.4.2 Determinarea distanțelor mai mici decât d_{\min} între 2 poliliniile

Se notează cu $d(V_t, V_k)$ distanța între vertexurile $t \in C_j$ și $k \in C_i$. Se calculează distanța $d(V_t, V_k)$ și se notează cu $NrInt$ partea întreagă a raportului $d(V_t, V_k) / d_{\min}$. Dacă $d(V_t, V_k) > d_{\min}$ atunci se observă că următorul vertex aparținând poliliniilei C_i pentru care distanța față de vertexul V_t poate să fie mai mică decât d_{\min} este $V_{k+NrInt}$. Pornind de la acest considerent se observă că pentru un mare număr de vertexuri aparținând lui C_i nu se mai calculează distanțele ceea ce duce la creșterea vitezei de calcul. În momentul în care se găsește un vertex V_k aparținând poliliniilei C_i pentru care distanța $d(V_t, V_k) \leq d_{\min}$ se formează o listă temporară $LTemp$ în care sunt stocate vertexurile care îndeplinesc această condiție. În momentul în care se găsește primul vertex

aparținând lui C_i care nu mai îndeplinește condiția, din lista temporară LTemp se selectează vertexul aparținând lui C_j cu distanța minimă față de V_t și se trece în lista LdMin împreună cu V_t .

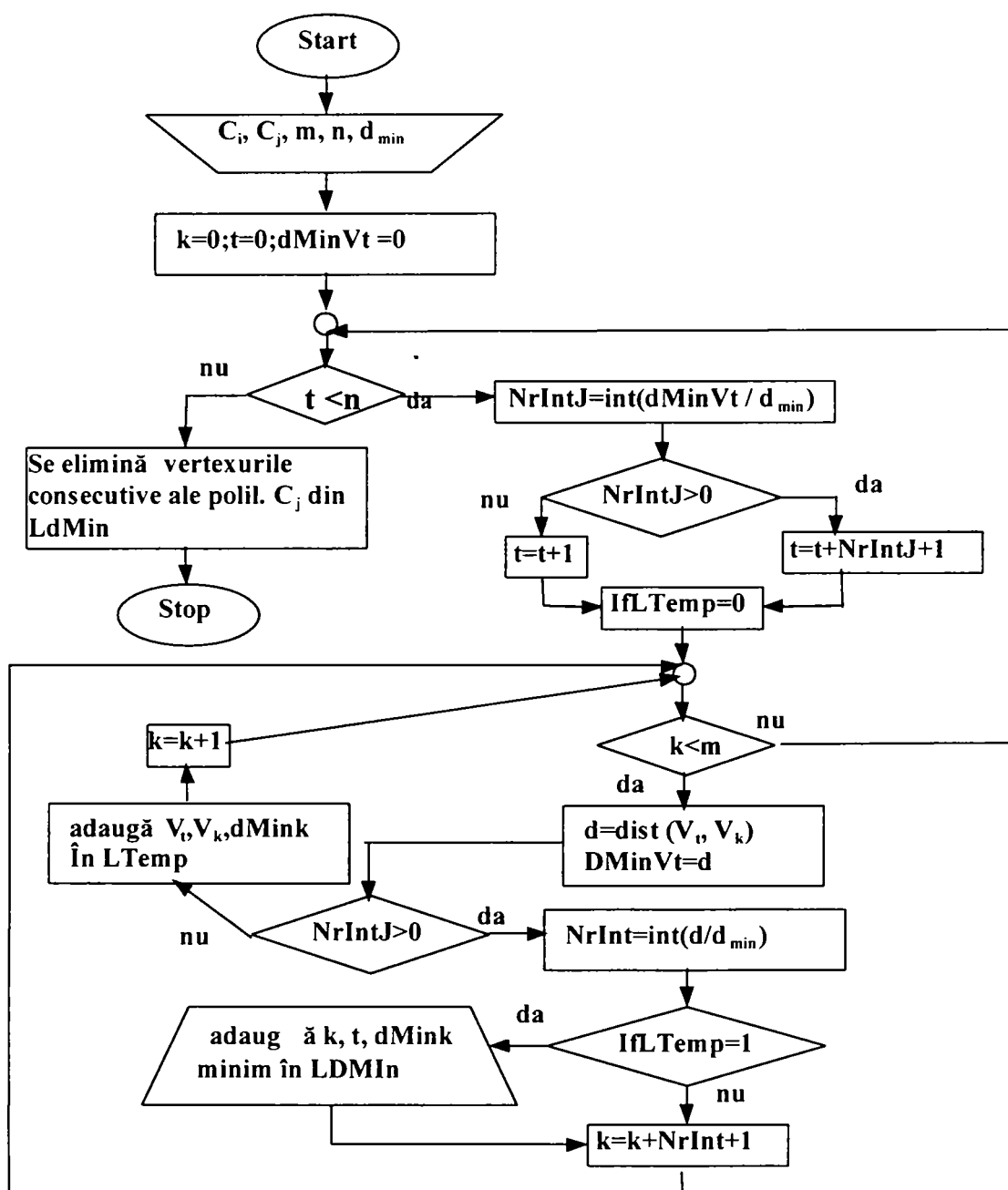


Fig. 5.4.3 Schema logică a algoritmului pentru determinarea intersecției dintre 2 polilinii

Bazat pe un raționament similar se poate reduce și numărul de vertexuri aparținând polilinieii C_j pentru care sunt necesare calculele distanțelor față de vertexurile polilinieii C_i . Dacă se notează cu $dMinV_t$ distanța minimă dintre V_t și oricare din vertexurile polilinieii C_j și cu $NrIntJ$ partea întreagă a raportului $dMinV_t / d_{min}$ atunci se observă că următorul vertex al polilinieii C_j care se ia în considerare la determinarea distanțelor minime între polilinii este $V_{t+NrIntJ}$.

La terminarea vertexurilor polilinieii C_j , în lista LdMin există vertexuri consecutive ale polilinieii C_j . Pentru fiecare grup de vertexuri consecutive ale polilinieii C_j se alege unul singur pentru care distanța dintre vertexurile aparținând poliliniilor C_i

și respectiv C_j este minimă rezultând în final o listă cu unul sau mai multe elemente cu structura : index vertex polilinie C_i , index vertex polilinie C_j .

Prin reducerea numărului de vertexuri aparținând poliliniilor C_i și C_j care se iau în considerare la calculul distanțelor minime între cele 2 polilinii, viteza de calcul crește semnificativ. Utilizarea doar a acestei optimizări nu este însă suficientă pentru a permite determinarea intersecțiilor în timp real, pe măsură ce sunt înregistrate coordonatele șirurilor de puncte 3D rezultate prin digitizare.

c) Optimizare pentru determinarea în timp real a intersecțiilor între poliliniile rezultate pe baza informațiilor despre șirurile de puncte obținute prin digitizare

Pentru determinarea în timp real a intersecțiilor dintre polilinia curentă C_j care se construiește la momentul respectiv din șirul de puncte digitizate și poliliniile generate anterior se urmărește optimizarea din punctul de vedere al vitezei de calcul prin limitarea zonei în care se caută posibilele intersecții. Această limitare se poate realiza prin utilizarea unei structuri de elemente volumice. Elementele volumice au formă paralelipipedică. Intersecțiile dintre polilinia curentă care se generează în timp real din datele preluate cu digitizorul 3D și celelalte polilinii se va căuta doar în elementele volumice care conțin polilinia curentă. Schema logică a algoritmului este prezentată în figura 5.4.5.

Inițial utilizatorul specifică colțurile diagonal opuse $B_1(x_{b1}, y_{b1}, z_{b1})$ și respectiv $B_2(x_{b2}, y_{b2}, z_{b2})$ ale unui paralelipiped în care se încadrează suprafața care urmează a fi digitizată. Aceste 2 puncte pot fi indicate cu ajutorul digitizorului 3D. Sunt specificate de către utilizator valoarea pasului de divizare a paralelipipedului pentru fiecare direcție: p_x , p_y și p_z . Se apelează o rutină care determină numărul de elemente volumice pe fiecare direcție și se alocă o matrice tridimensională cu dimensiunile egale cu numărul elementelor volumice pe direcțiile x, y și respectiv z (fig. 5.4.4).

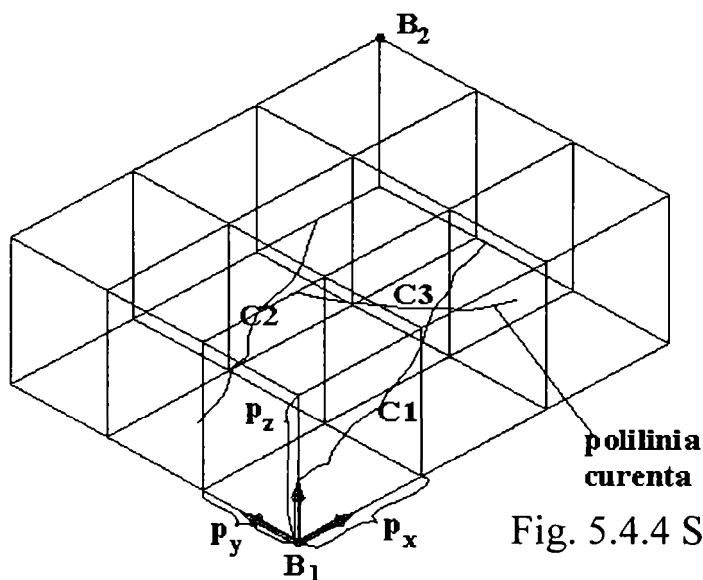


Fig. 5.4.4 Structură de elemente volumice

Fiecare element volumic are asociat o structură care conține adresa zonei de memorie în care este stocată o listă care are ca elemente grupuri de trei componente : adresa zonei de memorie la polilinia care are vertexuri în elementul volumic, indicele vertexului de intrare în elementul volumic, și indicele vertexului de ieșire din

elementul volumic. Dacă vertexul V_t este vertexul curent și este cuprins într-un element volumic VoxB iar vertexul V_{t-1} este cuprins într-un element volumic VoxA atunci se consideră vertexul V_{t-1} ca vertex de ieșire din VoxA și vertexul V_t ca vertex de intrare în VoxB.

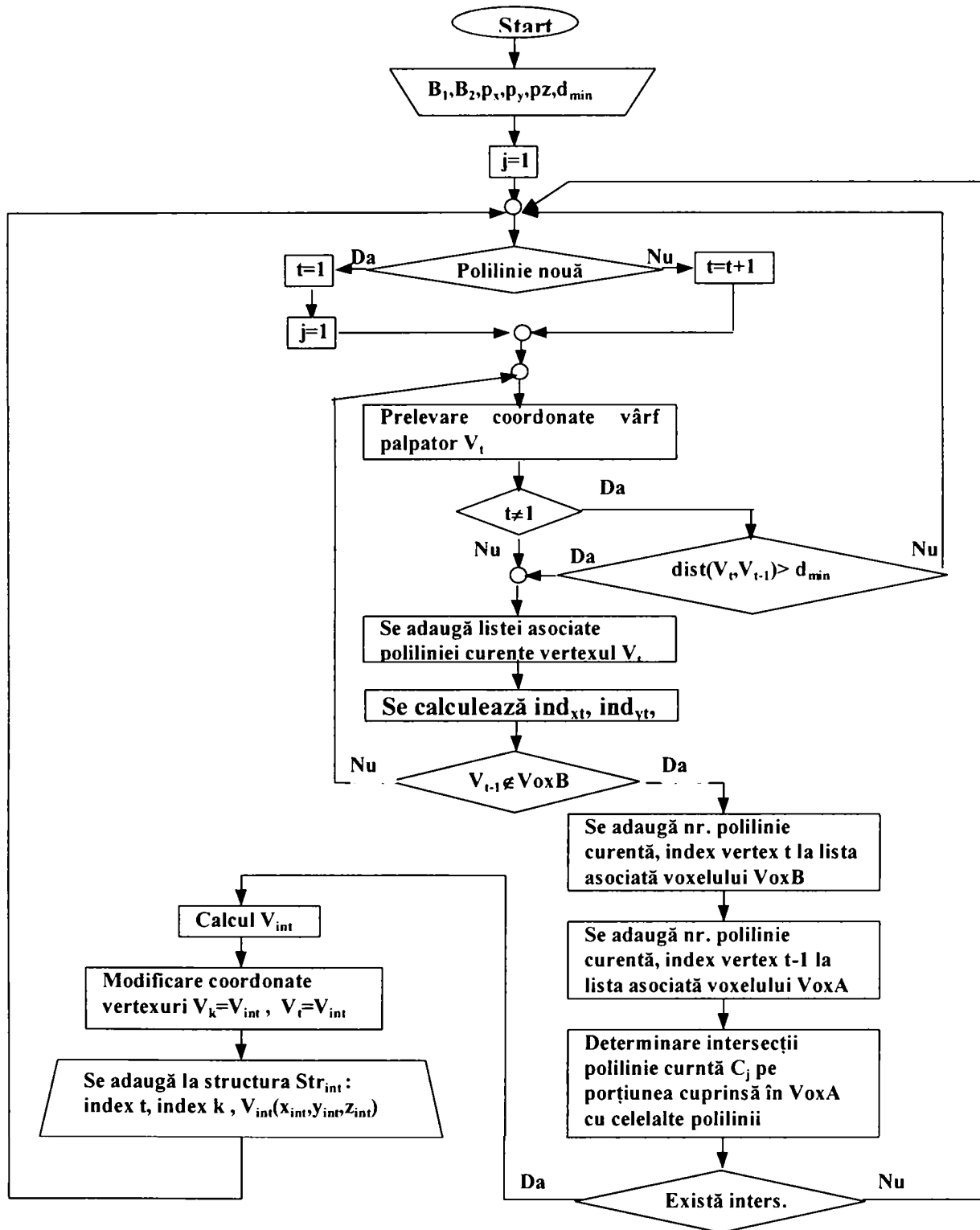


Fig. 5.4.5 Algoritm pentru optimizarea determinării în timp real a intersecțiilor între poliliniile rezultate pe baza informațiilor despre șirurile de puncte obținute prin digitizare.

În timp ce se generează polilinia 3D, pentru fiecare vertex $V_i(x_{vi}, y_{vi}, z_{vi})$ se determină indexul elementului volumic în care se încadrează acel vertex pe baza formulelor:

$$\text{ind}_{xt} = \text{int}[(x_{vt} - x_{bl}) / p_x]$$

$$\text{ind}_{yt} = \text{int}[(y_{vt} - y_{bl}) / p_y]$$

$$\text{ind}_{zt} = \text{int}[(z_{vt} - z_{bl}) / p_z]$$

De fiecare dată când vertexul V_t este cuprins într-un alt element volumic decât vertexul V_{t-1} se setează vertexurile de intrare și ieșire din elementul volumic, și se apelează rutina de căutare a intersecțiilor pentru elementul volumic din care s-a ieșit. Rutina de căutare a intersecțiilor determină intersecțiile poliliniilor curente într-un anumit element volumic V_{ct} numit în continuare element volumic curent cu alte poliliniile care au vertexuri în elementul volumic respectiv. Dacă vertexul V_k aparținând poliliniilor C_i este situat la o distanță mai mică decât $d_{\text{Min}V_t}$ față de un vertex V_t aparținând poliliniilor C_j atunci se vor schimba coordonatele celor 2 vertexuri ca fiind egale cu punctul situat la mijlocul V_{int} al segmentului care unește vertexurile 3D V_k și V_t (fig. 5.4.6).

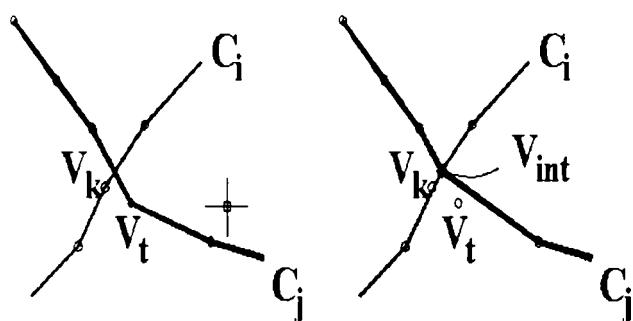


Fig. 5.4.6 Deplasarea vertexurilor în punctul de “intersecție”.

Se creează o structură de date Str_{int} care conține pentru fiecare polilinie o listă cu elemente de forma : *index t al vertexului $V_t \in C_j$, index k al vertexului $V_k \in C_i$, $V_{\text{int}}(x_{\text{int}}, y_{\text{int}}, z_{\text{int}})$* . La fiecare determinare a unei intersecții se completează Str_{int} . Pentru determinarea muchiilor care limitează fiecare petec de suprafață se apelează o rutină bazată pe algoritmul prezentat în paragraful următor care utilizează ca date de intrare structura Str_{int} și lista cu coordonatele vertexurilor pentru fiecare polilinie. Apelarea acestei rutine se poate realiza după generarea unei polilini 3D. În principiu este posibilă apelarea rutinei și în timpul preluării coordonatelor șirului de puncte de pe suprafața obiectului digitizat dar aceasta necesită resurse hardware considerabile din punct de vedere puterii de calcul.

d) Determinarea muchiilor care delimitează petecele de suprafață

Pentru a genera suprafața cu ajutorul petecelor Coons, este necesar să se furnizeze pentru fiecare petec 4 muchii care să se intersecteze două câte două.

Deoarece poliliniile au fost preluate într-o ordine oarecare este necesară utilizarea unui algoritm pentru gruparea nodurilor (adică a “intersecțiilor”) care determină un petec. Pentru aceasta se pornește de la un nod considerat de grad I (de exemplu primul din listă) și se caută nodurile alăturate (de grad II), apoi nodurile alăturate acestora (numite noduri de grad III).

Dacă se găsește un nod de gradul III identic pornind pe două ramuri diferite, muchiile parcurse pe cele două ramuri vor constitui un petec. Prezentarea de principiu a modului de identificare a nodurilor ce determină un petec este dată în figura 5.4.7

Din lista cu câte 4 noduri ce reprezintă fiecare petec și lista cu coordonatele vertexurilor fiecărei polilinii se determină coordonatele vertexurilor care formează fiecare muchie a petecului.

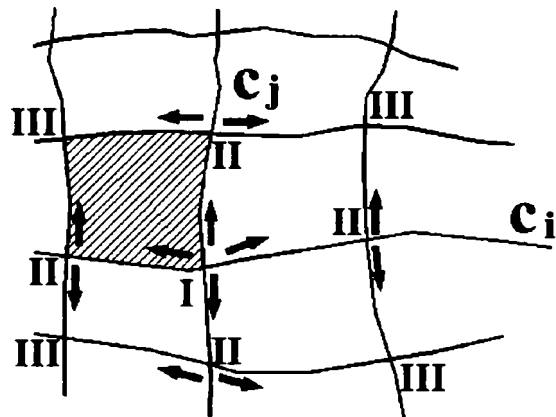


Fig 5.4.7 Identificarea nodurilor care determină un petec

5.4.2.3 Particularități ale utilizării aplicației în diferite sisteme CAD

Metoda prezentată pentru modelarea suprafețelor pornind de la informații despre coordonatele unor șiruri de puncte 3D obținute prin digitizare este generală. Determinarea vertexurilor poliliniilor limitatoare pentru fiecare petec poate fi realizată în absența unui sistem CAD. Pentru vizualizarea în timp real a poliliniilor care se formează din șirul de puncte digitizate și pentru generarea suprafețelor pornind de la curbele limitatoare este necesară utilizarea unui sistem CAD.

În figura 5.4.8 este prezentată o suprafață realizată în mediul AutoCAD pe baza datelor rezultate în urma prelucrării unor șiruri de puncte obținute prin digitizare 3D.

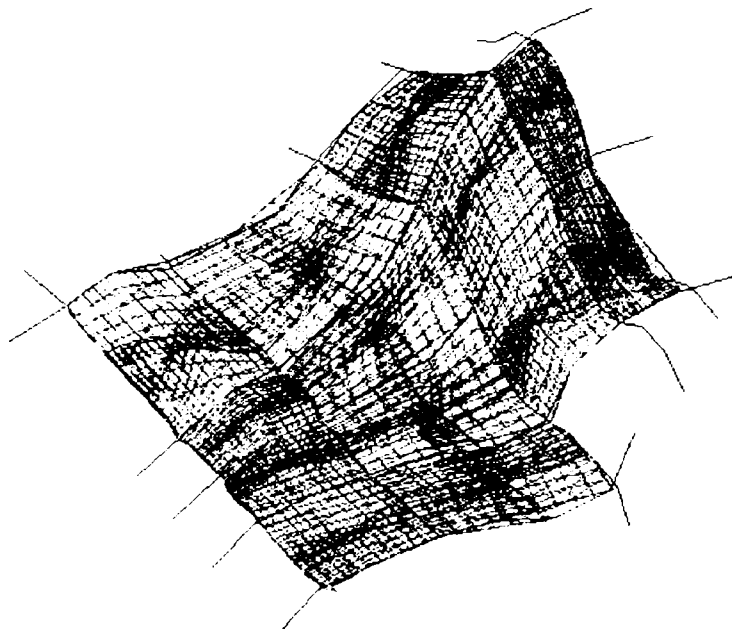


Fig 5.4.8 Suprafața realizată în mediul AutoCAD pornind de la șiruri de puncte digitizate

6. Realizări constructive ale unor digitizoare 3D

6.1. Introducere

În acest capitol se prezintă construcția a două digitizoare 3D cu palpator. S-a pornit de la ideea realizării unor digitizoare de construcție cât mai simplă dar care să permită demonstrarea principiului funcționării unui digitizor 3D cu palpator. S-au realizat două tipuri de digitizoare care sunt denumite în continuare *digitizor 3D RRR* și respectiv *digitizor 3D cu fire*. În ambele cazuri, digitizorul 3D este constituit din 3 subsisteme : subsistemul mecanic, subsistemul electronic și subsistemul informatic (fig.6.1.1).

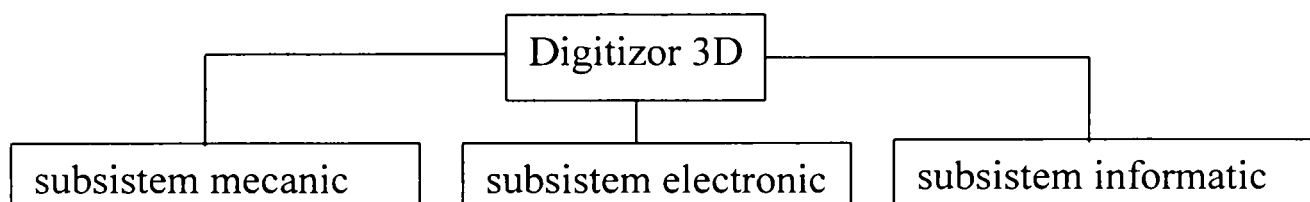


Fig.6.1.1. Subsistemele digitizoarelor 3D

Construcția subsistemelor mecanic și electronic a fost realizată cu costuri minime. Digitizoarele de precizii ridicate sunt utilizate în special în industria auto și în cea aeronautică unde costul echipamentelor se amortizează într-o perioadă relativ redusă. Într-o serie de aplicații nu este necesară o precizie deosebit de ridicată, de exemplu în cazuri în care digitizorul 3D se utilizează pentru obținerea unor modele geometrice de suprafață ale unor obiecte existente fizic în scopul vizualizării sau reclamei pe Internet. Cele 2 digitizoare realizate de autor sunt destinate acestui scop și în același timp au un scop educațional. Componentele subsistemului electronic sunt preluate din dispozitivele de interacțiune în mod grafic cu calculatorul PC (mouse), care la costuri foarte reduse, oferă facilitățile necesare pentru utilizarea la digitizoarele realizate. Datorită preluării acestor componente fără modificări majore descrierea acestor subsisteme se va face sumar în subcapitolele destinate fiecăruia din cele 2 digitizoare.

De câte ori a fost posibil s-a urmărit alegerea unei soluții mecanice cât mai simple, chiar în cazul creșterii volumului de calcul necesar pentru rezolvarea modelelor geometrice. Digitizoarele realizate sunt concepute să lucreze cu calculatoare PC fără necesități hardware deosebite. Un calculator "Pentium" cu frecvența procesorului de 200MHz este suficient pentru acoperirea necesităților de calcul a coordonatelor vârfului palpatorului pe baza modelelor geometrice în timp real (peste 20 puncte/sec). Pentru asigurarea comunicației între digitizoare și sistemul de calcul, calculatorul PC a fost dotat cu 2 porturi seriale suplimentare. În funcție de complexitatea aplicațiilor care utilizează datele furnizate de digitizor necesitățile hardware pentru sistemul de calcul pot să crească.

6.2. Realizarea constructivă a unui digitizor RRR cu palpator

6.2.1. Modelul matematic al digitizorului 3D

Modelul matematic are la baza modelul geometric direct. Schema cinematică este prezentă în figura 6.2.1. și este identică cu cea a mecanismului generator de traiectorie a robotului antropomorf [KOV92].

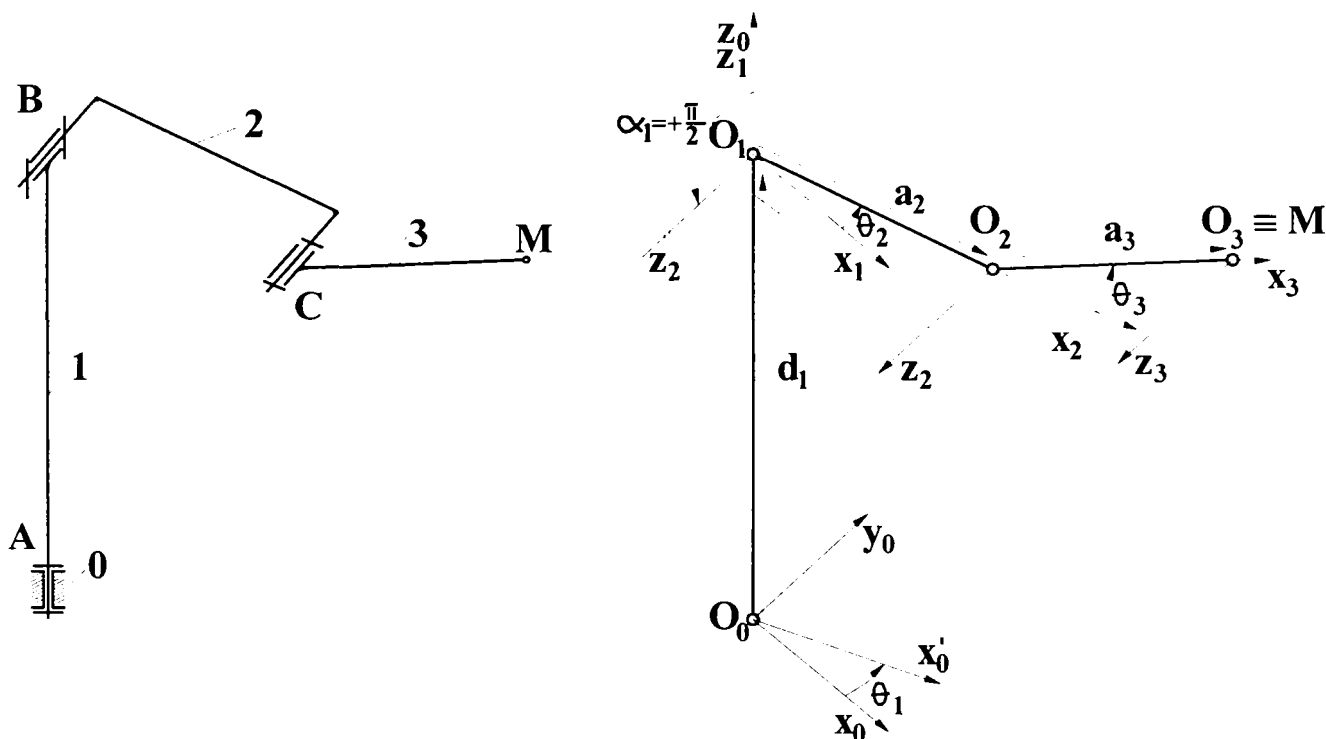


Fig. 6.2.1. Schema cinematică a digitizorului 3D

Se notează : $c_1 = \cos\theta_1$ $s_1 = \sin\theta_1$
 $c_2 = \cos\theta_2$ $s_2 = \sin\theta_2$
 $c_3 = \cos\theta_3$ $s_3 = \sin\theta_3$

Matricea de trecere de la sistemul de referință atașat elementului fix la sistemul de referință atașat ultimului element este :

$${}^0T_3 = \begin{bmatrix} C1C2C3 - C1S2S3 & -C1C2S3 - C1S2C3 & S1 & C1C2C3a3 - C1S2S3a3 + C1C2a2 \\ S1C2C3 - S1S2S3 & -S1C2S3 - S1S2C3 & -C1 & S1C2C3a3 - S1S2S3a3 + S1C2a2 \\ S2C3 + C2S3 & -S2S3 - C2C3 & 0 & S2C3a3 + C2S3a3 + S2a2 + d1 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Deoarece digitizorul dispune doar de mecanism generator de traiectorie interesează doar coordonatele punctului caracteristic care sunt date de relațiile :

$$p_x = C1 (C2C3a3 - S2S3a3 + C2a2)$$

$$p_y = S1 (C2C3a3 - S2S3a3 + C2a2)$$

$$p_z = S_2 C_3 a_3 + C_2 S_3 a_3 + S_2 a_2 + d_1$$

Valorile dimensiunilor d_1 , a_2 și a_3 sunt cunoscute prin construcție iar valorile unghiurilor θ_1 , θ_2 și θ_3 se determină cu ajutorul traductoarelor incrementale de rotație. O reprezentare schematică a funcționării digitizorului 3D este prezentată în figura 6.2.2.

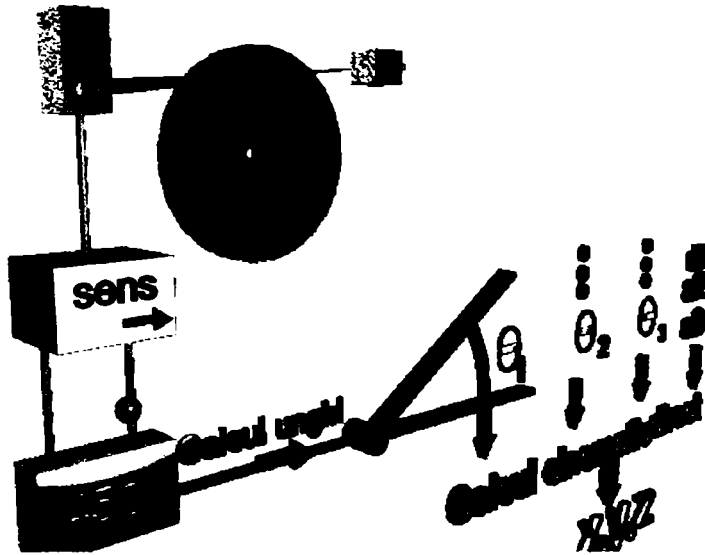
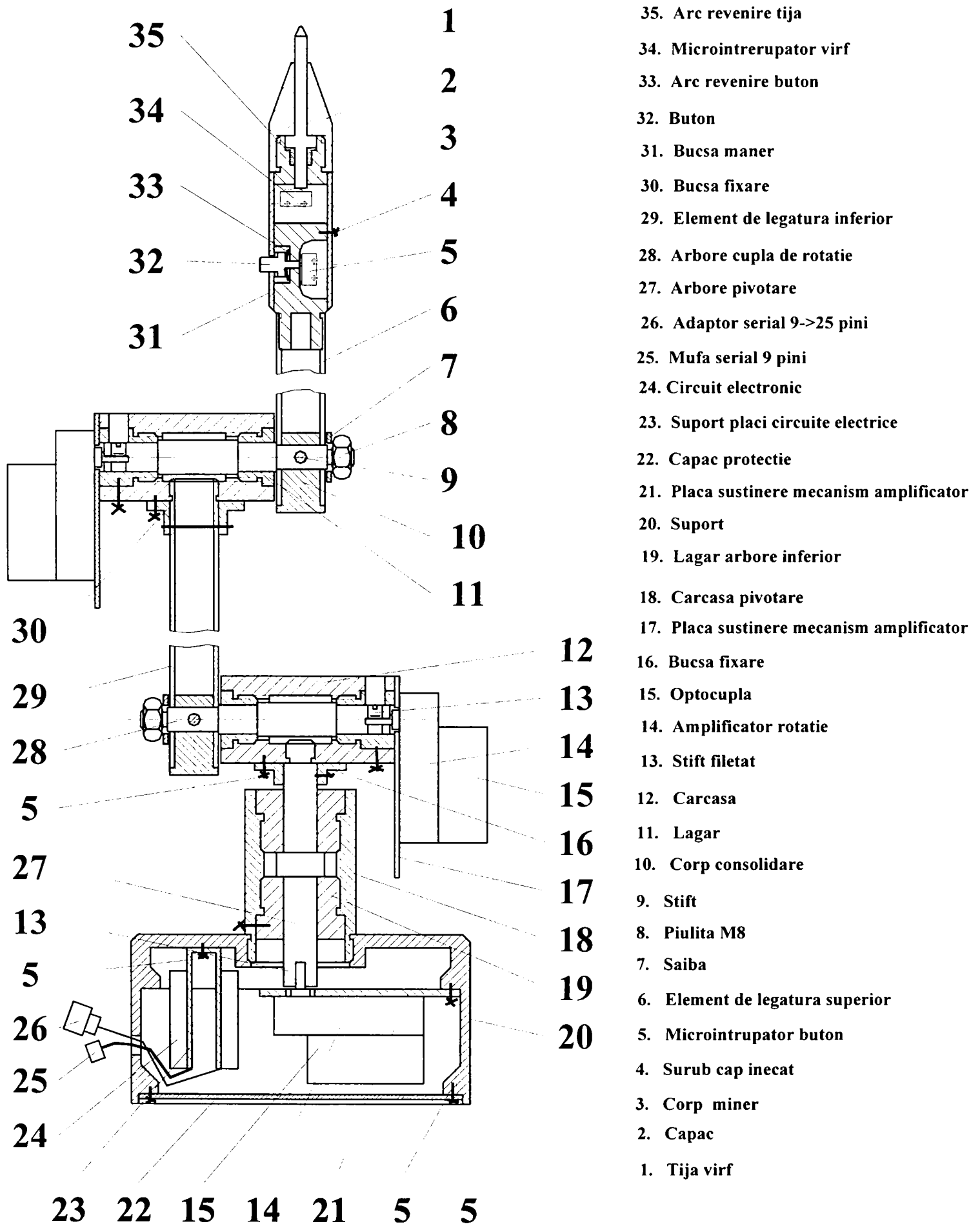


Fig. 6.2.2 Reprezentare schematică a funcționării digitizorului 3D

La construcția digitizorului 3D se utilizează pentru traductorul incremental un disc cu număr relativ redus de fante iar pentru generarea unui număr suficient de impulsuri se utilizează un mecanism cu roți dințate care joacă rol de amplificator. Alegerea acestei soluții constructive are dezavantajul că prezintă jocuri. S-a utilizat un mecanism cu roți dințate de serie mare care intră în componența unui aparat de utilizare casnică cu un cost redus dar care nu dispune de compensarea jocurilor. Pentru obținerea unei precizii suficiente, jocul care apare la schimbarea de sens de rotație s-a modelat prin tratarea ca fenomen de histereză. Prin software se detectează schimbarea de sens și se realizează corecțiile necesare pentru compensarea fenomenului de histereză. Se notează cu j_i numărul de impulsuri “pierdute” la schimbarea sensului de rotație relativă a elementelor cuplei cinematice i . La fiecare schimbare de sens numărul de impulsuri devine: $imp_i = imp_i \pm j_i$. Semnul operației depinde de poziția inițială.

6.2.2. Subsistemul mecanic

Sistemul mecanic a fost astfel conceput încât să materializeze structura unui dispozitiv de ghidare RRR. Desenul de ansamblu al digitizorului și elementele componente sunt prezentate în figura 6.2.3. Elementul fix a primei cuple cinematice de rotație a fost materializată printr-o carcasă (18) solidară cu suportul (20) prin îmbinare cu filet. Mișcarea de pivotare este realizată de către ax, axul fiind susținut de două bușe din bronz (19) de dimensiuni identice. În partea inferioară axul este cuplat cu mecanismul amplificator (14). În partea superioară, de acest ax este solidarizată carcasa celei de-a doua cuple cinematice (12), tot prin intermediul unui filet. Axul este ghidat prin intermediul unor bușe din bronz.



- 35. Arc revenire tija
- 34. Microintrerupator virf
- 33. Arc revenire buton
- 32. Buton
- 31. Bucsa maner
- 30. Bucsa fixare
- 29. Element de legatura inferior
- 28. Arbore cupla de rotatie
- 27. Arbore pivotare
- 26. Adaptor serial 9->25 pini
- 25. Mufa serial 9 pini
- 24. Circuit electronic
- 23. Suport placi circuite electrice
- 22. Capac protectie
- 21. Placa sustinere mecanism amplificator
- 20. Suport
- 19. Lagar arbore inferior
- 18. Carcasa pivotare
- 17. Placa sustinere mecanism amplificator
- 16. Bucsa fixare
- 15. Optocupla
- 14. Amplificator rotatie
- 13. Stift filetat
- 12. Carcasa
- 11. Lagar
- 10. Corp consolidare
- 9. Stift
- 8. Piulita M8
- 7. Saiba
- 6. Element de legatura superior
- 5. Microintrupator buton
- 4. Surub cap inecat
- 3. Corp miner
- 2. Capac
- 1. Tija virf

Fig. 6.2.3. Desenul de ansamblu al digitizorului și elementele componente ale digitizorului 3D

La un capăt al axului este cuplat mecanismul amplificator (14), iar la celălalt capăt este fixat prin intermediul unui șurub brațul elementului doi. S-a optat pentru o construcție modulară, cupla a treia de rotație fiind materializată identic cu cea de-a doua cuplă. În figura 6.2.4 este prezentată o reprezentare explodată utilizând facilitatea de rendering a unei cuple orizontale, corespunzătoare celei de-a doua și a treia cuple a digitizorului.



Fig. 6.2.4 Vedere explodată a unei cuple orizontale a digitizorului

Ultimul element al digitizorului are prevăzut la vârf o tijă de palpate care de fapt este un senzor de existență ce semnalează contactul dintre vârful digitizorului și suprafața care este modelată. Tot în vârful ultimului element se găsește locașul pentru microîntrerupătorul (5) care comandă înscrierea câte unei linii în fișierul de date primare. Reprezentarea palpatorului în vedere explodată este prezentată în figura 6.2.5.

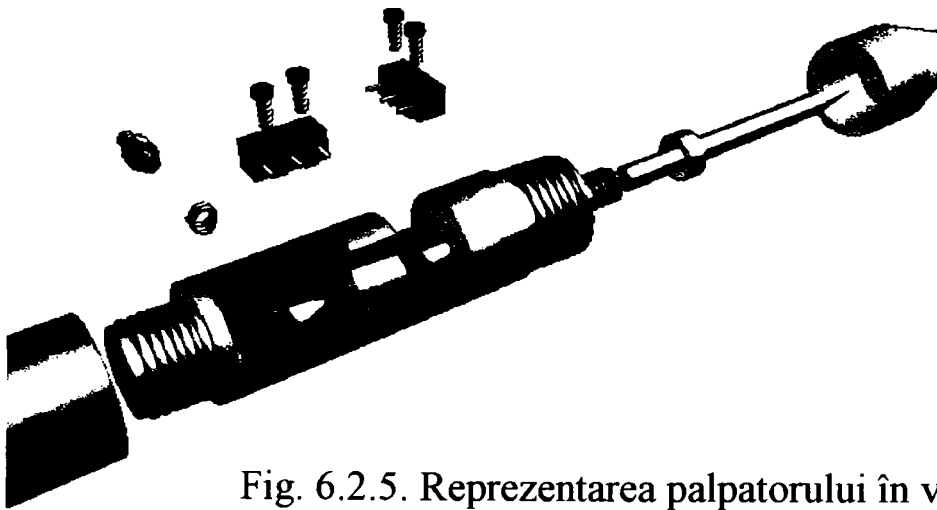


Fig. 6.2.5. Reprezentarea palpatorului în vedere explodată

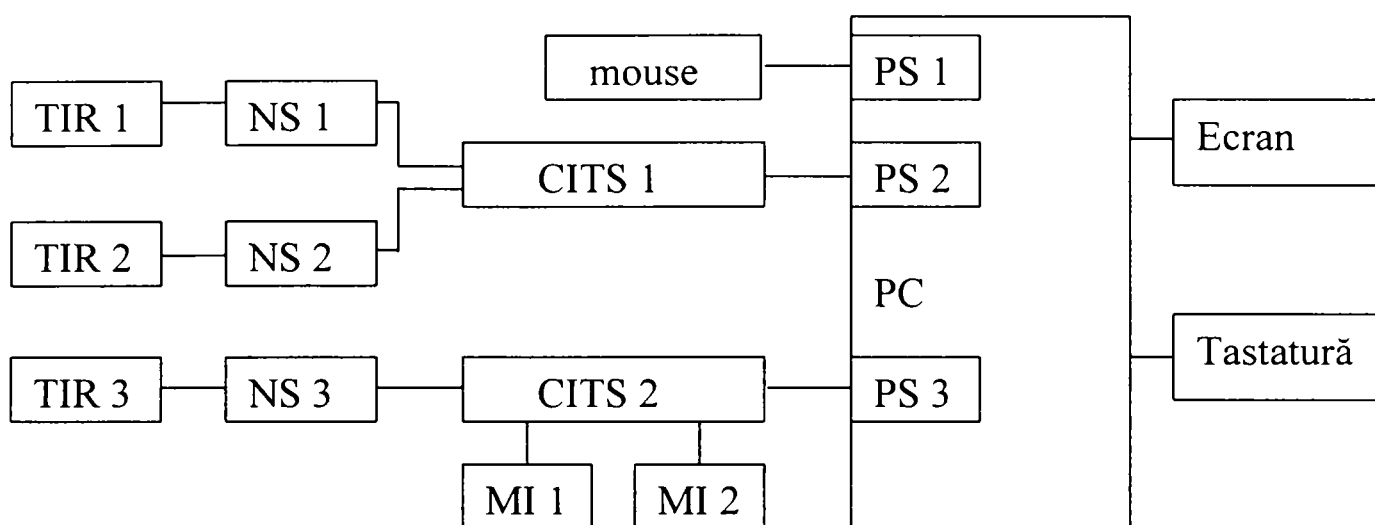
Adoptarea unei soluții modulare pentru cuplele cinematice a doua și a treia și pentru cele 3 subansamble formate din perechi mecanisme cu roți dințate - optocuple are avantajul unei simplități constructive și posibilității de depanare a digitizorului în caz de avarii. Deoarece nu sunt turații mari, pentru ghidare s-au utilizat bușe din bronz. Digitizorul este prevăzut cu trei opritoare, câte unul pentru fiecare cuplă de rotație, în

scopul stabilirii punctului de “acasă”. Lungimile brațelor nu au fost alese de dimensiuni mari deoarece ar fi dus la creșterea erorii de stabilire a punctului caracteristic. Această eroare este dependentă de lungimile brațelor, precizia traductorului incremental de rotație, de jocurile axelor în lagăre și de jocurile transmisiei cu roți dințate utilizate la amplificarea mișcării de rotație. În corpul de susținere au fost realizate locașuri pentru fixarea circuitelor electronice (24) a celor două mouse-uri. Plăcuța electronică pe care sunt fixate fototranzistorul și emițătorul în infraroșu este fixată de corpul amplificatorului de rotație (14), prin intermediul unui element din tablă de îndoită la 90° . Atât pe carcasa amplificatorului, cât și pe suportul de susținere din tablă, sunt prevăzute canale pentru reglajul fin, poziționarea fototranzistorului și emițătorului față de discul cu fante necesitând o precizie ridicată. În scopul protejării circuitului electronic, acesta este prevăzut cu o carcasă care se solidarizează prin șuruburi cu carcasa mecanismului amplificator de rotație.

Digitizorul i se poate atașa solidar o masă rotativă, aceasta fiind considerată ca axă externă a digitizorului. Această masă trebuie să fie prevăzută cu bacuri acționate prin mecanism șurub-piuliță pentru fixarea rigidă a corpului ce urmează a fi digitizat. Utilizarea mesei rotative permite măsurarea obiectelor de dimensiuni mai mari. Pentru utilizarea digitizorului la introducerea formei în calculator a unor suprafețe de dimensiuni și mai mari, se poate utiliza atârnarea digitizorului în poziție răsturnată pe o placă având orificii de fixare pentru elementul fix al digitizorului într-o poziție și cu o orientare determinată univoc. Pentru determinarea coordonatelor unor puncte situate pe aceste suprafețe se va ține cont de deplasarea digitizorului de la un orificiu la altul, precizată în aplicația software. Cursa tijei din vârful ultimului element al digitizorului (1) are o valoare redusă pentru a nu introduce erori. Punctul situat pe intersecția suprafeței sferice a vârfului palpatorului cu axa acestuia este considerat punctul caracteristic al digitizorului și poziția acestuia este determinată prin calcul geometric direct.

6.2.3. Subsistemul electronic

Subsistemul electronic al digitizorului 3D, se compune din optocuplele și electronica aferentă a două dispozitive de interacțiune cu interfața grafică a calculatorului. O schemă bloc a subsistemului electronic este prezentată în figura 6.2.6. Comunicația între electronica digitizorului 3D și calculatorul personal pe care sunt implementate aplicațiile software pentru procesarea datelor este realizată prin interfața serială RS232. În porturile PS2 și PS3 informațiile despre poziția relativă a elementelor digitizorului de la digitizorul 3D ajung sub formă de număr de impulsuri. Microîntrerupătorul MI 1 este utilizat pentru sesizarea contactului dintre vârful palpatorului și suprafața digitizată iar microîntrerupătorul MI 2 este utilizat pentru comanda înregistrării continue a coordonatelor.



Legendă:

- TR 1,...,3 - traductor incremental de rotație
- NS 1,...,3 - numărător și circuit detectare sens
- MI 1,2 - micro-întrerupător
- CITS 1,2 - circuit pentru transmitere serială date
- P 1,...,3 - porturi seriale a calculatorului PC

Fig. 6.2.6. Schema bloc a subsistemului electronic

6.2.4. Subsistemul informatic

6.2.4.1 Introducere

Pentru utilizarea digitizorului 3D, pentru lucrul la nivel apropiat de hardware s-au elaborat programe în limbajul C. Deoarece în versiunea actuală s-a dorit utilizarea datelor furnizate de digitizorul 3D în AutoCAD, unele rutine sunt integrate în acest mediu fiind realizate cu ajutorul bibliotecilor ObjectArx. În scopul îndeplinirii sarcinii de determinare a coordonatelor punctului caracteristic al digitizorului cu fire și preluarea continuă a unor șiruri de puncte au fost realizate o serie de rutine pentru rezolvarea unor probleme punctuale:

6.2.4.2 Rutină pentru indicarea poziției de acasă a digitizorului (inițializarea contoarelor) și preluarea datelor din fișierul de inițializare.

Dimensiunile elementelor d_1 , a_2 a_3 și numărul de impulsuri furnizate de fiecare traductor la o rotație completă sunt stocate într-un fișier tip *.ini. De asemenea, în acest fișier este stocat numărul de impulsuri pentru compensarea jocurilor (j_i). O sesiune de lucru cu digitizorul se consideră perioada de timp consumată de la activarea rutinei de inițializare până la dezactivarea conexiunii aplicației software care utilizează digitizorul 3D. La începerea unei sesiuni de lucru cu digitizorul, prima acțiune este aducerea digitizorului în poziția de acasă prin rotirea elementelor cuplelor cinematice astfel încât să atingă opritoarele. Pentru a ușura această operație, zona care

trebuie sa vină în contact cu opritorul este marcată. În momentul în care digitizorul a fost adus în poziția de acasă se indică acest lucru aplicației software prin acționarea unei taste.

6.2.4.3. *Determinarea numărului de impulsuri curent și a stării butoanelor*

Pentru inițializarea contoarelor și pentru citirea numărului de impulsuri curent corespunzător fiecărui traductor s-a creat o aplicație ObjectArx cu funcțiile principale *digi_setPos(impContor1,impContor2,impContor3,impContor4)* și respectiv *digi_GetPos(imp₁, imp₂, imp₃, imp₄, b11,b12,b21,b22)*. Parametrii *impContorI* cu $i=1, \dots, 4$ reprezintă numărul de impulsuri corespunzător fiecărei axe a digitizorului. A patra axă disponibilă din punct de vedere software și electronic nu este utilizată de digitizor fiind disponibilă pentru o axă externă ce poate aparține de exemplu unei mese de poziționare rotative sau a unei translații lungi a bazei digitizorului 3D. Parametrii *imp_i* cu $i = 1, \dots, 4$ reprezintă numărul de impulsuri curent ca numere întregi iar *b11, b12, b21, b22* reprezintă starea butoanelor. Variabila *b11* reprezintă starea virfului palpatorului (1 pt contact cu suprafața, 0 fără contact) iar variabila *b12* reprezintă starea microîntrerupătorului pentru indicarea preluării șirurilor de coordonate 3D. Variabilele *b21, b22* nu sunt utilizate fiind disponibile pentru 2 microîntrerupătoare.

6.2.4.4. *Rutină pentru calcul geometric direct în vederea obținerii coordonatelor punctului caracteristic al palpatorului.*

Pe baza datelor preluate la punctul 6.2.4.2 și a numărului de impulsuri determinat prin citirea buferelor porturilor 2 și 3 ale calculatorului se poate determina în orice moment coordonatele punctului caracteristic. Această rutină implementează formulele prezentate la paragraful 6.2.1.

6.2.4.5. *Determinarea distanței între 2 puncte indicate prin palparea cu vârful digitizorului*

Prin palparea succesivă a două puncte de către palpator rutina calculează distanța în 3D și afișează coordonatele punctelor și distanța pe monitorul calculatorului.

6.2.4.6. *Determinarea coordonatelor unor șiruri de puncte cu salvare în fișier*

Preluarea continuă a coordonatelor unor șiruri este comandată prin acționarea microîntrerupătorului situat pe mânerul digitizorului 3D. La apăsarea rutinei, utilizatorul este solicitat să specifice distanța minimă d_{\min} pentru ca punctele să fie considerate ca puncte distincte și numele fișierului în care se salvează coordonatele punctelor palpate. Ca și condiție suplimentară se poate impune preluarea coordonatelor punctelor doar dacă palpatorul digitizorului este în contact cu suprafața obiectului. În figura 6.2.7. este prezentată schema logică pentru această rutină. Se observă că rutina utilizează funcții oferite de rutinele anterioare.

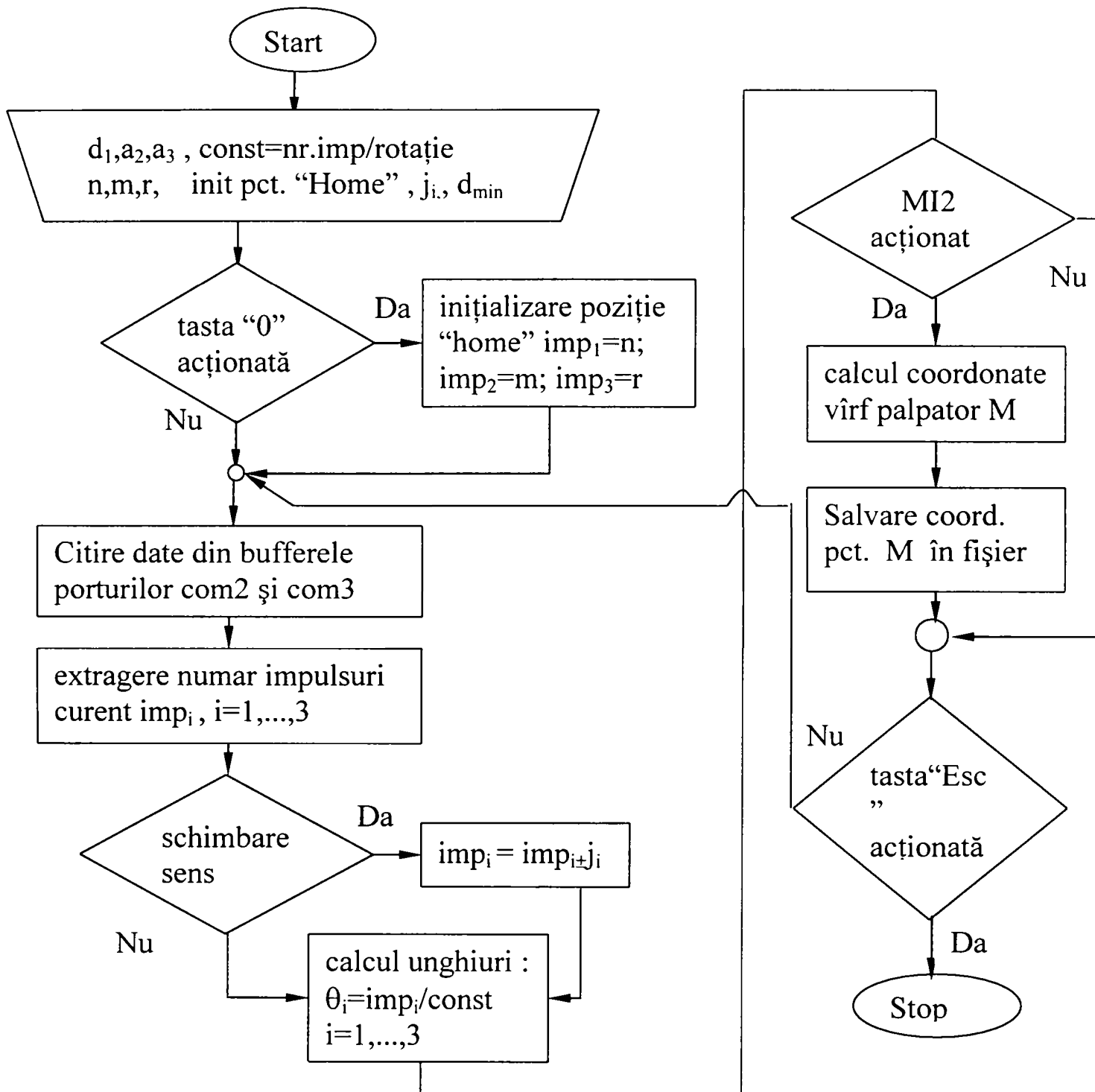


Fig.6.2.7. Schema logică a algoritmului pentru determinarea coordonatelor unor șiruri de puncte cu salvare în fișier

6.2.4.7. Determinarea coordonatelor unor șiruri de puncte cu generarea în timp real a unor polilinii 3D în AutoCAD

Este în mare parte asemănătoare cu rutina de la punctul 6.2.4.6. dar în loc de scrierea coordonatelor în fișier generează polilinii 3D din șirurile de puncte palpate. Generarea acestor polilinii în timp real permite utilizatorului să realizeze corecțiile. Aceste corecții sunt necesare de exemplu pentru a obține grilaj cu o distribuție aproximativ uniformă a poliliniilor palpate pe suprafața care este digitizată.

6.2.4.8. Interfață pentru programul de generare a suprafețelor pornind de la șiruri de puncte

Sunt puse la dispoziție funcții pentru preluarea șirurilor de puncte, indicarea cu ajutorul digitizorului a colțurilor paralelipedului în care se încadrează suprafața care urmează a fi digitizată. Poliliniile generate din punctele palpate se vizualizează pe ecranul calculatorului și în același timp în paralel rulează programul care generează petecele de suprafață.

6.2.5. Încercări experimentale

6.2.5.1. Introducere

Pentru determinarea preciziei de măsurare a digitizorului realizat (fig. 6.2.8) se efectuează trei tipuri de încercări experimentale : determinarea repetabilității și testul cu calibru de pas. Salvarea datelor obținute prin măsurători se realizează în fișiere Excel. Aplicația care realizează automatizarea preluării datelor și a procesării lor este realizată în VisualBasic 6.



Fig. 6.2.8. Digitizorul 3D RRR

6.2.5. 2. *Determinarea repetabilității digizorului*

Testul de repetabilitate constă în măsurarea coordonatelor unui punct singular cu ajutorul digitizorului 3D de un anumit număr de ori. Se consideră pentru încercarea experimentală măsurarea a 10 puncte din spațiul de lucru al digitizorului fiecare punct fiind măsurat de 10 ori. Între fiecare 2 măsurători se deplasează palpatorul digitizorului în spațiul de lucru astfel încât să existe deplasări relative la toate elementele cuplurilor cinematice. Se calculează deviațiile standard pentru fiecare din coordonatele x, y și z a celor 10 puncte măsurate. În figura 6.2.8. este prezentată o captură de ecran din rularea aplicației de măsurare a repetabilității. Înregistrările măsurătorilor sunt prezentate în anexa 3.

Testul de repetabilitate										
Nr.Punct	Nr. Masurare	x	y	z	dx	dy	dz	dist. Pt. anterior		
1	1	135.43627	54.71535	5.301255						
2		135.74611	55.11829	5.541952	-0.30983	-0.40094	-0.2407	0.646139193		
3		135.63402	54.79524	5.286461	0.112088	0.321047	0.255491	0.467659971		
4		135.64432	54.7994	5.556374	-0.0103	-0.00416	-0.26991	0.011861713		
5		135.43627	54.71535	5.301255	0.208046	0.084049	0.255119	0.239607656		
								deviatie max		
Valoare max					-0.30983	-0.40094	-0.26991	0.646139193		
Nr.Punct	Nr. Masurare	x	y	z	dx	dy	dz	dist. Pt. anterior		
1	2	84.273234	117.4775	5.410024						
2		84.353573	117.1573	5.426307	-0.08034	0.320244	-0.01628	0.459963808		
3		83.979961	116.6384	5.476271	0.373612	0.518904	-0.04996	0.823473759		
4		83.776276	116.7848	5.476271	0.203685	-0.14638	0	0.290420903		
5		83.645406	116.6023	5.225836	0.130869	0.182433	0.250635	0.289292741		
								deviatie max		
Valoare max					0.373612	0.518904	0.250635	0.823473759		

Fig. 6.2.9. Captură de ecran a aplicației de determinare a repetabilității digizorului 3D

6.2.5. 3 *Determinarea preciziei de măsurare în spațiul de lucru*

Încercarea cu calibru de pas urmărește determinarea erorii cumulate și a celei incrementale la măsurarea unor puncte dispuse liniar la un anumit pas. Se utilizează o riglă care este fixată în spațiul de lucru al digitizorului. Rigla are marcaje cu un pas constant de 20 mm. Se palpează succesiv cu vârful palpatorului punctele marcate pe riglă. Aplicația determină erorile de măsurare a pasului. În figura 7.2.10 se prezintă o captură de ecran din desfășurarea aplicației. Înregistrările datelor sunt prezentate în anexa 3.

Testul cu calibru de pas

Nr. Punct	x	y	z	pas. Exp	pas calibru	eroare	pas increm
1	328.00106	154.47432	55.998989				
2	309.34414	157.93985	50.931484	19.64103091	20	-0.358969087	
3	289.11338	168.25298	46.860276	23.06987498	20	3.069874979	
4	269.18917	181.55327	43.638992	24.17123687	20	4.171236865	
5	249.85498	193.09053	40.75262	22.69912574	20	2.699125737	
6	228.86466	206.04829	36.981927	24.95426391	20	4.954263913	
7	209.81097	218.19696	35.670968	22.63518559	20	2.635185595	
8	188.0005	228.88717	33.587068	24.37867677	20	4.378676769	
9	167.34872	242.1431	32.345147	24.57149264	20	4.571492641	
10	148.04309	254.46829	31.854372	22.90978692	20	2.909786917	
					media		3.225630481
					eroare max.		4.954263913

Fig 6.2.10. Captură de ecran a rezultatelor încercării cu calibru de pas

6.3. Proiectarea constructivă a unui digitizor cu fire

6.3.1 Prezentare generală

Se prezintă un sistem de digitizare cu palpator la care determinarea poziției vârfului palpatorului și a orientării axei palpatorului se bazează pe calcularea lungimii unor fire. La unul din capete firul este fixat de un palpator iar pentru determinarea lungimii firului acesta se înfășoară pe o rolă solidară cu discul cu fante a unui traductor incremental de rotație optoelectronic. Prin lungimea firului se înțelege distanța între punctul prin care firul este în contact cu palpatorul și punctul de contact cu rola de sprijin. Există 5 module pentru măsurarea lungimii firului de construcție identică (fig. 6.3.1).

Scopul realizării digitizorului este desfășurarea unor aplicații prin care să se verifice programele software pentru digitizarea unor piese de mobilier, determinarea erorilor de situare a roboților cu digitizoare 3D și interacțiunea cu realitatea virtuală prin digitizoare 3D. S-a dorit totodată experimentarea soluției constructive pentru digitizoare 3D fără pretenții deosebite în vederea preciziei dar care să acopere un spațiu de lucru suficient de mare pentru digitizarea de exemplu a mobilierului de ședere. S-a ales soluția unui digitizor cu fire deoarece permite utilizarea unor traductoare incrementale de rotație de rezoluții mai scăzute și implicit cu costuri mai reduse în condițiile asigurării unui spațiu de lucru de volum mediu pentru digitizoare 3D.

În acest subcapitol se tratează problemele teoretice și proiectarea constructivă, aplicațiile fiind prezentate în capitolele în care se descriu programele software realizate de autor care utilizează digitizorul 3D cu fire.

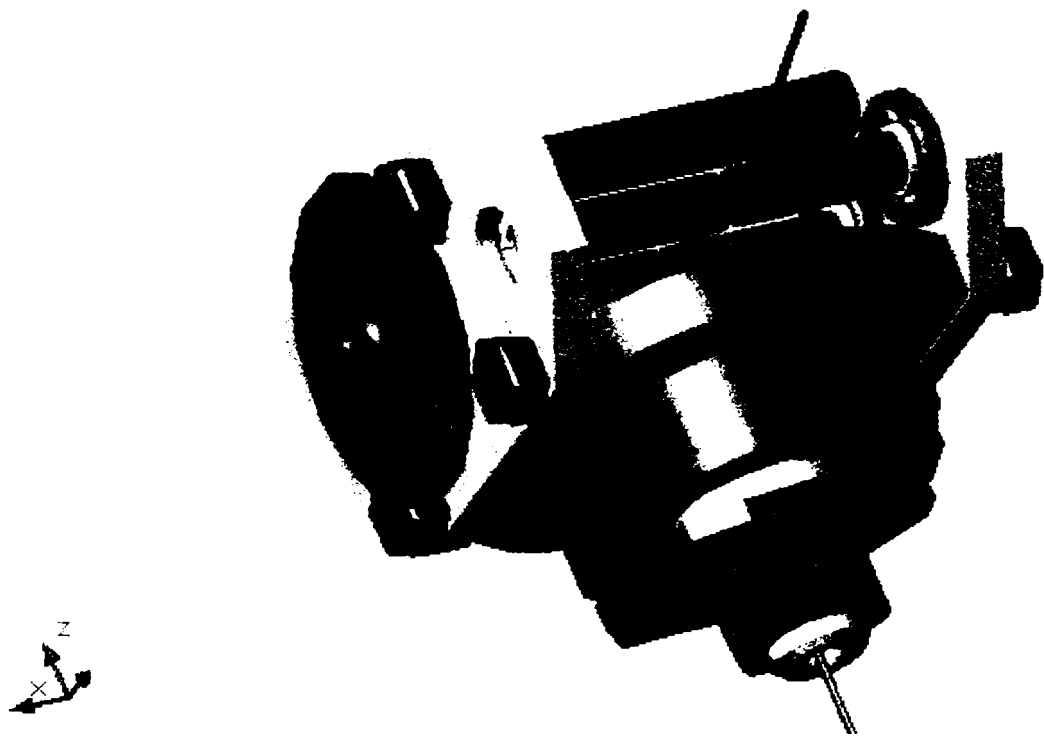


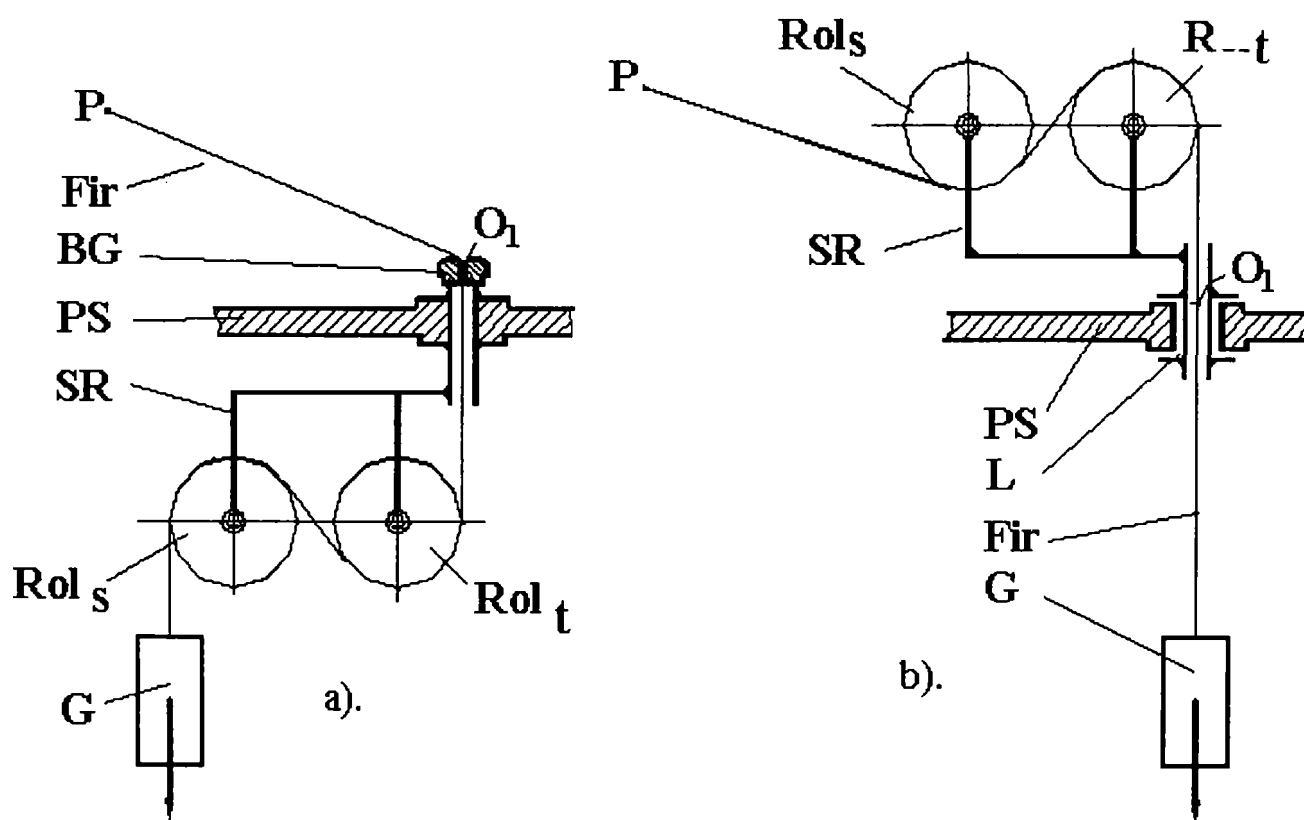
Fig. 6.3.1 Modul al sistemului de digitizare cu fire

6.3.2 Modelul matematic al digitizorul 3D cu fire

6.3.2.1. Introducere

Se prezintă două variante de model matematic a sistemului de digitizare cu fire. Soluțiile constructive care materializează cele 2 variante de modele matematice utilizează aceleași componente doar modul de montare a unora dintre aceste componente este diferit. În figura 6.3.2. este prezentat schematic această diferență a modului de montare pentru un modul.

Prima variantă, prezentată în fig 6.3.2. a) utilizează o bucășă (BG) pentru ghidarea firului. Bucășă are un orificiu de diametru aproximativ egal cu diametrul firului prin care acesta alunecă. Firul este înfășurat pe rola R_{ol_t} solidară cu discul cu fante al traductorului incremental de rotație și pe rola de sprijin R_{ol_s} . Întinderea firului și forța de apăsare pe role este realizată de către greutatea G . Suportul rolor SR este fixat solidar de placa suport PS . punctul P reprezintă punctul în care firul este în contact cu palpatorul digitizorului. Modelul matematic în această variantă este relativ simplu deci resursele hardware necesare pentru calculele de determinare a poziției vârfului palpatorului în timp real pot să fie modeste. Varianta prezintă dezavantajul existenței frecării de alunecare între fir și orificiul bucășei de ghidare ceea ce determină necesitatea utilizării unei greutăți pentru întinderea firelor de masă mai mare.



- P - punct pe palpator
- BG - bucășă de ghidare
- PS - placă de susținere
- R_{ol_s} - rolă de sprijin
- R_{ol_t} - rolă solidară cu discul cu fante al traductorului

- G - greutate pentru întinderea firului
- L - lagăr cuplă cinematică de rotație
- SR - suport role

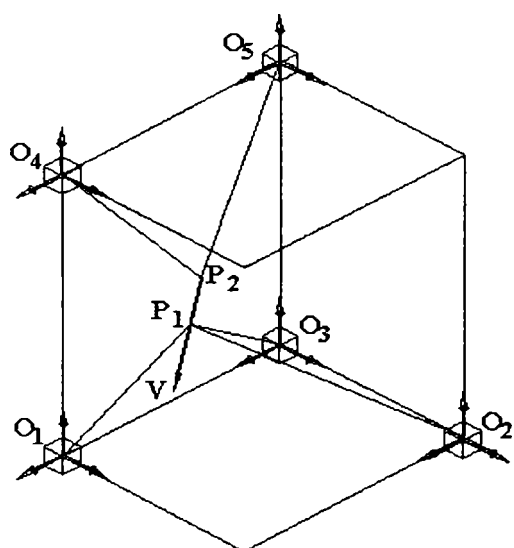
Fig. 6.3.2. Cele două variante de montare a modului de măsurare a lungimii firului.

În cea de a doua variantă , prezentată în figura 6.3.2. b), suportul rozelor are o mișcare de rotație în jurul unei axe verticale. Firul este înfășurat pe rolele R_{ol_T} și R_{ol_S} și este întins cu ajutorul greutății G . Modelul matematic în această variantă este mai complex decât la prima varianta datorită rotirii modulului în jurul axei verticale.

În ambele modele matematice se consideră ca ipoteză simplificatoare faptul ca deplasarea palpatorului este sesizată prin intermediul unui “fir” inextensibil, fără masă și perfect flexibil.

6.3.2.2 Modelul matematic pentru prima variantă de montare

În figura 6.3.3 este prezentată o schemă de principiu a sistemului de digitizare. Prin $O_1 \dots O_6$ s-au notat originile sistemelor de referință atașate fiecărui modul, P_1 reprezintă punctul de intersecție a firelor care leagă palpatorul de modulele 1-3, P_2 reprezintă punctul de intersecție a firelor care prin care palpatorul este legat de modulele 4 și 5 iar V este vârful palpatorului.



6.3.3 Schema de principiu a sistemului de digitizare cu fire

Vârful palpatorului se găsește pe dreapta determinată de punctele $P_1(x_{P1}, y_{P1}, z_{P1})$ și $P_2(x_{P2}, y_{P2}, z_{P2})$ la o distanță d_v cunoscută prin construcție față de P_1 . Fie d_{12} distanța dintre punctele P_1 și P_2 . Se notează cu k raportul în care P_1 împarte segmentul VP_2 deci $k = d_v / d_{12}$. Cunoscând coordonatele punctelor P_1 și P_2 se determină coordonatele vârfului palpatorului care vine în contact cu suprafața digitizată:

$$\begin{aligned} x_v &= x_{P1} + k \cdot (x_{P2} - x_{P1}) \\ y_v &= y_{P1} + k \cdot (y_{P2} - y_{P1}) \\ z_v &= z_{P1} + k \cdot (z_{P2} - z_{P1}) \end{aligned} \tag{6.3.1}$$

Punctul P_1 se găsește la distanțele r_1, r_2, r_3 de punctele O_1, O_2 și respectiv O_3 . Poziția punctelor O_i cu $i = 1, \dots, 3$ nu variază în cursul digitizării. Locul geometric al unui punct în spațiu la o distanță dată față de un punct fix este o sferă. Fiind situat la intersecția celor 3 fire care leagă primele 3 module de palpator, punctul P_1 se găsește în același timp pe sferele S_i cu centrele în O_i și de raze R_i . Deci punctul P_1 se găsește la intersecția a trei sfere de centre O_1, O_2 și O_3 și raze egale cu distanțele O_1P_1, O_2P_1 și respectiv O_3P_1 . Este un caz particular a poziției relative a trei sfere. Punctul P_2 se

găsește la intersecția sferelor de centre O_4, O_5 și P_1 cu razele egale cu distanțele O_4P_2, O_5P_2 și respectiv d_{12} .

Ecuția sferei de centru $I(a, b, c)$ și rază R este :

$(x-a)^2 + (y-b)^2 + (z-c)^2 = R^2$ iar ecuația generală a sferei este:

$x^2 + y^2 + z^2 - 2 \cdot a \cdot x - 2 \cdot b \cdot y - 2 \cdot c \cdot z + d = 0$ unde $d = a^2 + b^2 + c^2 - R^2$

Se consideră 3 sfere de centre $O_1(a, b, c), O_2(a', b', c')$ și $O_3(a'', b'', c'')$ având razele r_1, r_2 și respectiv r_3 . Se determină coordonatele punctului $P(x_p, y_p, z_p)$ de intersecție a celor trei sfere. Ecuțiile celor trei sfere sunt:

$$S = x^2 + y^2 + z^2 - 2 \cdot a \cdot x - 2 \cdot b \cdot y - 2 \cdot c \cdot z + d = 0$$

$$S' = x^2 + y^2 + z^2 - 2 \cdot a' \cdot x - 2 \cdot b' \cdot y - 2 \cdot c' \cdot z + d' = 0 \quad (6.3.2)$$

$$S'' = x^2 + y^2 + z^2 - 2 \cdot a'' \cdot x - 2 \cdot b'' \cdot y - 2 \cdot c'' \cdot z + d'' = 0$$

Intersecția sferelor S și S' este cercul c_1 de ecuații:

$$\begin{cases} x^2 + y^2 + z^2 - 2 \cdot a \cdot x - 2 \cdot b \cdot y - 2 \cdot c \cdot z + d = 0 \\ 2(a-a') \cdot x + 2(b-b') \cdot y + 2(c-c') \cdot z + d'-d = 0 \end{cases} \quad (6.3.3)$$

iar a sferelor S și S'' cercul c_2 de ecuații:

$$\begin{cases} x^2 + y^2 + z^2 - 2 \cdot a \cdot x - 2 \cdot b \cdot y - 2 \cdot c \cdot z + d = 0 \\ 2(a-a'') \cdot x + 2(b-b'') \cdot y + 2(c-c'') \cdot z + d''-d = 0 \end{cases} \quad (6.3.4)$$

Punctul de intersecție a celor trei sfere este punctul de intersecție a cercurilor c_1 și c_2 deci coordonatele punctului P satisfac sistemul de ecuații:

$$\begin{cases} x^2 + y^2 + z^2 - 2 \cdot a \cdot x - 2 \cdot b \cdot y - 2 \cdot c \cdot z + d = 0 \\ 2(a-a') \cdot x + 2(b-b') \cdot y + 2(c-c') \cdot z + d'-d = 0 \\ 2(a-a'') \cdot x + 2(b-b'') \cdot y + 2(c-c'') \cdot z + d''-d = 0 \end{cases} \quad (6.3.5)$$

Cea de a doua și a treia ecuație a sistemului reprezintă ecuațiile a două plane. Problema intersecției celor trei sfere se poate reduce la determinarea intersecției dintre sfera a cărei ecuație este prima ecuație a sistemului 6.3.5. și dreapta (dr) determinată de intersecția planelor date prin ultimele 2 ecuații ale sistemului 6.3.5..

Parametrii directori ai dreptei (dr) sunt :

$$l = 4[(b-b')(c-c'') - (c-c')(b-b'')] \quad (6.3.6)$$

$$m = 4[(a-a')(c-c'') - (a-a'')(c-c')] \quad (6.3.7)$$

$$n = 4[(a-a')(b-b'') - (b-b')(a-a'')] \quad (6.3.8)$$

Se determină coordonatele unu punct $P_0(x_0, y_0, z_0)$ care aparține dreptei care verifică ultimele două ecuații a sistemului 6.3.5.. Fie acest punct intersecția dreptei cu planul xOy (planul de ecuație $z=0$). Punctul P_0 verifică sistemul de ecuații:

$$\begin{cases} 2(a-a') \cdot x_0 + 2(b-b') \cdot y_0 + d'-d = 0 \\ 2(a-a'') \cdot x_0 + 2(b-b'') \cdot y_0 + d''-d = 0 \end{cases} \quad (6.3.9)$$

Se extrage expresia lui y din cea de a doua ecuație a sistemului:

$$y_0 = \frac{a-a''}{b''-b} \cdot x_0 + \frac{d''-d}{2(b''-b)} \quad (6.3.10)$$

Se notează: $e = \frac{a-a''}{b''-b}$ și $f = \frac{d''-d}{2(b''-b)}$

$$y_0 = e \cdot x_0 + f \quad (6.3.11)$$

Prin înlocuirea expresiei necunoscutei y_0 în prima ecuație a sistemului (6.3.9) rezultă:

$$x_0 = \frac{2(b'-b) \cdot f + d - d'}{2[(a-a') + e(b-b')]}$$
 (6.3.12)

Valoarea coordonatei y_0 se obține înlocuind x_0 în expresia (6.3.11.).

Se urmărește determinarea intersecției dintre dreapta (dr) cu parametrii directori l, m, n și care trece prin punctul $P_0(x_0, y_0, z_0)$ cu sfera care verifică prima ecuație a sistemului (6.3.5.). Un punct $L(x, y, z)$ aflat pe dreapta (dr) la distanța ρ de P_0 intersectează sfera dacă verifică ecuațiile:

$$\begin{cases} x = x_0 + \rho \cdot \cos \alpha \\ y = y_0 + \rho \cdot \cos \beta \\ z = z_0 + \rho \cdot \cos \gamma \end{cases} \quad (6.3.13)$$

Pentru ca punctul $L(x, y, z)$ să se intersecteze cu sfera este necesar să verifice ecuația sferei. Prin înlocuirea expresiilor pentru x, y, z din (6.3.13) în ecuația sferei rezultă o ecuație de gradul 2 în necunoscuta ρ :

$$\rho^2 + 2[(x_0 - a) \cdot \cos \alpha + (y_0 - b) \cdot \cos \beta + (z_0 - c) \cdot \cos \gamma] \cdot \rho + N_0 = 0 \quad (6.3.14)$$

În relația de mai sus s-a notat cu N_0 expresia:

$$N_0 = (x_0 - a)^2 + (y_0 - b)^2 + (z_0 - c)^2 - r_1^2$$

Valorile cosinusurilor directoare ai dreptei (dr) se obțin cu :

$$\cos \alpha = \frac{l}{\sqrt{l^2 + m^2 + n^2}}; \cos \beta = \frac{m}{\sqrt{l^2 + m^2 + n^2}}; \cos \gamma = \frac{n}{\sqrt{l^2 + m^2 + n^2}} \quad (6.3.15)$$

Dacă se notează cu $k = 2[(x_0 - a) \cdot \cos \alpha + (y_0 - b) \cdot \cos \beta + (z_0 - c) \cdot \cos \gamma]$ determinantul ecuației (6.3.14) este :

$$\Delta = k^2 - 4 \cdot N_0$$

În cazul în care $\Delta > 0$ există două soluții pentru ρ :

$$\rho_{1,2} = \frac{-k \pm \sqrt{\Delta}}{2}$$

deci vor rezulta 2 puncte de intersecție între (dr) și sferă:

$$A[x_0 + \rho_1 \cdot \cos \alpha, y_0 + \rho_1 \cdot \cos \beta, x_0 + \rho_1 \cdot \cos \gamma] \quad (6.3.16)$$

$$B[x_0 + \rho_2 \cdot \cos \alpha, y_0 + \rho_2 \cdot \cos \beta, x_0 + \rho_2 \cdot \cos \gamma]$$

Din cele două puncte se alege doar unul datorită soluției constructive. Când se determină coordonatele punctului P_1 rezultat din intersecția firelor corespunzătoare primelor 3 module se alege punctul care are valoarea minimă pentru coordonata z . Obiectul care urmează a fi digitizat se așează pe o placă astfel ca orice punct aparținând suprafeței obiectului să aibă coordonata z mai mică ca oricare din centrele sferelor determinate de firele primelor 3 module. La determinarea coordonatelor punctului P_2 de intersecție a sferelor determinate de firele ultimelor două module cu sfera de rază P_1P_2 și centru P_1 dintre cele două puncte de intersecție se alege cel cu valoarea coordonatei z mai mică.

Dacă $\Delta = 0$ există o singură soluție ρ pentru ecuația (6.3.14) și punctul de intersecție între sfere este:

$$T[x_0 + \rho \cdot \cos \alpha, \quad y_0 + \rho \cdot \cos \beta, \quad x_0 + \rho \cdot \cos \gamma]$$

Relațiile de calcul prezentate anterior se utilizează de două ori. Prima dată pentru determinarea punctului P_1 pe baza informațiilor furnizate de primele trei module și a doua oară pentru determinarea punctului P_2 pe baza coordonatelor punctului P_1 și a informațiilor oferite de modulele 4 și 5. Cu relațiile 6.3.1 se determină coordonatele vârfului V al palpatorului care este punctul caracteristic al digitizorului 3D. Orientarea axei palpatorului este dată de linia care unește punctele P_1 și P_2 și reprezintă dreapta caracteristică a digitizorului 3D cu fire. Ecuația dreptei caracteristice (dc) este ecuația unei drepte care trece prin puncte $P_1(x_1, y_1, z_1)$ și $P_2(x_2, y_2, z_2)$:

$$P_1, P_2): \quad \frac{x - x_1}{x_2 - x_1} = \frac{y - y_1}{y_2 - y_1} = \frac{z - z_1}{z_2 - z_1}$$

Parametrii directori ai dreptei caracteristice sunt:

$$l_{dc} = x_2 - x_1; \quad m_{dc} = y_2 - y_1; \quad n_{dc} = z_2 - z_1$$

iar cosinusurile directoare ale dreptei caracteristice sunt :

$$\cos \alpha_{dr} = \frac{l_{dr}}{\sqrt{l_{dr}^2 + m_{dr}^2 + n_{dr}^2}}; \quad \cos \beta = \frac{m_{dr}}{\sqrt{l_{dr}^2 + m_{dr}^2 + n_{dr}^2}}; \quad \cos \gamma = \frac{n_{dr}}{\sqrt{l_{dr}^2 + m_{dr}^2 + n_{dr}^2}} \quad (6.3.17)$$

6.3.2.3 Modelul matematic pentru varianta a doua de montare a modulului

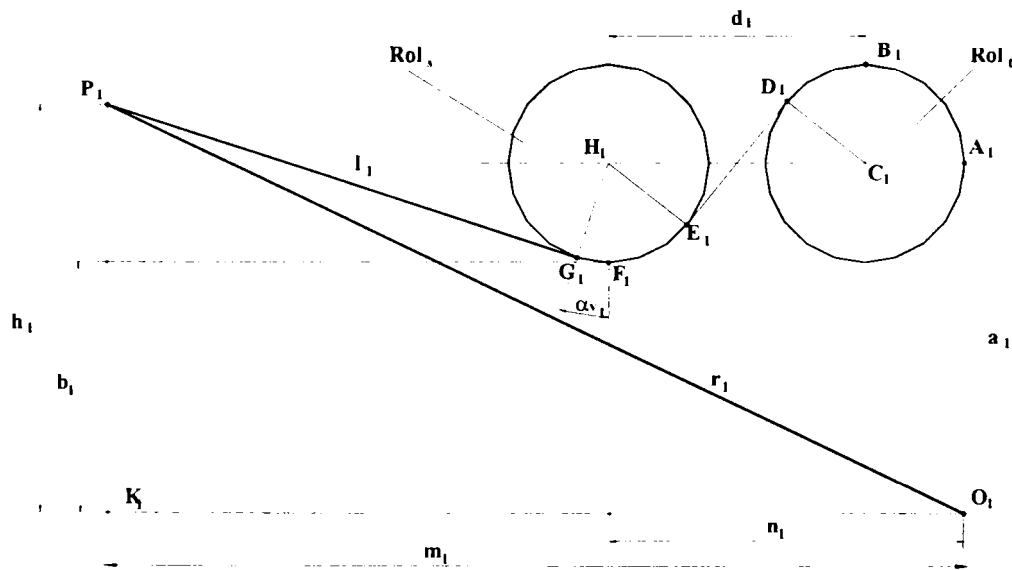


Fig. 6.3.4 Secțiune simplificată printr-un modul de măsurare a lungimii firului

Dacă se cunosc distanțele O_1P_1 , O_2P_1 , O_3P_1 și pozițiile punctelor O_1, O_2, O_3 în raport cu sistemul de referință fix atașat digitizorului se poate determina poziția punctului P_1 ca intersecție a trei sfere pe baza modelului de calcul prezentat la începutul subcapitolului. Lungimea firului pe porțiunea $F_1G_1P_1$ se determină prin contorizarea impulsurilor furnizate de traductorul incremental de rotație solidar cu rola Rol_1 . La începerea operației de digitizare se aduce palpatorul digitizorului în poziția de acasă. Prin această operație, punctul P_1 ajunge într-o poziție cunoscută pentru care se

cunoaște lungimea firului pe porțiunea $F_1G_1P_1$. Se notează cu Nr_t numărul de impulsuri furnizat de traductor la o rotație completă a rolei și cu r_{rol} raza rolei.

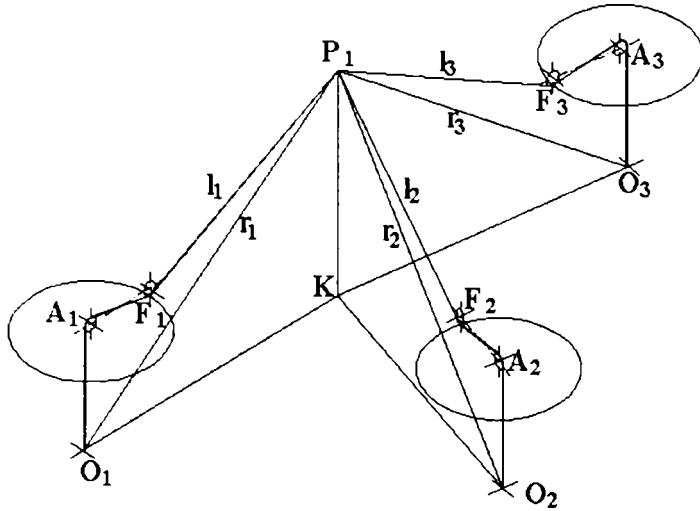


Fig. 6.3.5. Determinarea distanțelor $r_i, i=1, \dots, n$

Cu palpatorul în poziția de acasă se inițializează contorul pentru numărarea impulsurilor la valoarea întreagă notată Nr_{impAc1} :

$$Nr_{impAc1} = \text{int}\left(\frac{l_1}{2 \cdot \pi \cdot r_{rol} \cdot Nr_t}\right) \quad (6.3.18)$$

Din acest moment, deplasarea punctului P_1 cauzată de deplasarea palpatorului implică deplasarea firului și rotirea rolei pe care este înfășurat. Prin contorizarea numărului de impulsuri cauzate de rotirea rolei se cunoaște în orice moment lungimea firului pe porțiunea $F_1G_1P_1$. Problema care se pune este determinarea distanțelor O_1P_1, O_2P_2, O_3P_3 pe baza cunoașterii lungimii firelor $F_1G_1P_1, F_2G_2P_2, F_3G_3P_3$ și a pozițiilor punctelor O_1, O_2, O_3 (fig.6.3.5) Se determină relațiile între coordonatele punctului P și lungimile l_i pentru un modul (fig. 6.3.6).

Din triunghiul $F_iA_iO_i$ se poate scrie relația : $\frac{\pi}{2} - \varphi_i = \text{arctg} \frac{\rho_i}{a_i}$ deci

$$\varphi_i = -\text{arctg} \frac{\rho_i}{a_i} + \frac{\pi}{2} \quad (6.3.19)$$

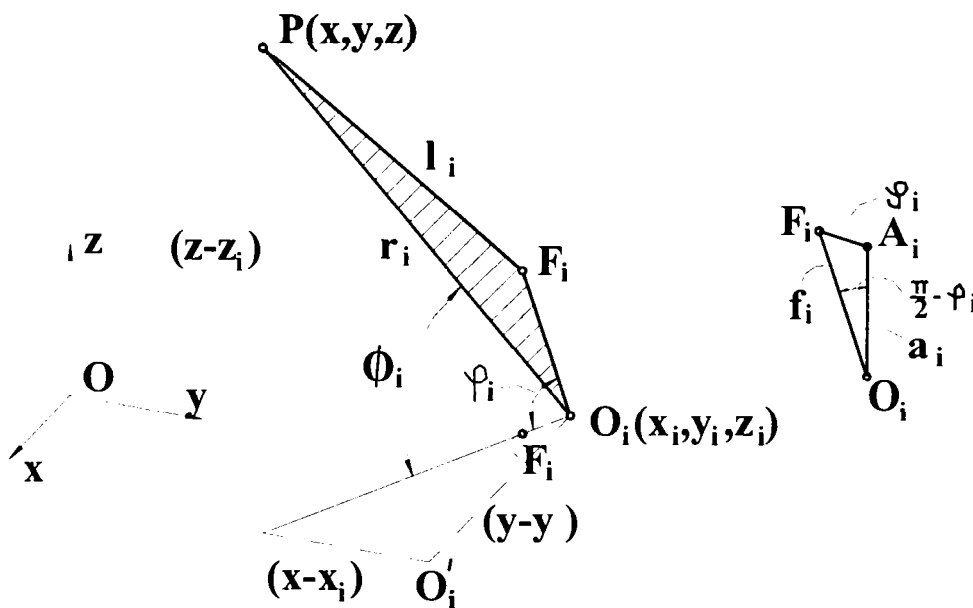


Fig. 6.3.6 Schema simplificată reprezentând un modul

În relația (6.3.19) valorile ρ_i și a_i se cunosc prin construcție, deci unghiul φ_i are o valoare constantă. Se notează cu r_i distanța dintre punctele P și O_i :

$$r_i^2 = (x - x_i)^2 + (y - y_i)^2 + (z - z_i)^2 \quad (6.3.20)$$

Dacă se notează cu ϕ_i unghiul pe care dreapta O_iP îl face cu planul în care sunt situate cele trei puncte O_i avem:

$$\phi_i = \arctg \frac{z - z_i}{\sqrt{(x - x_i)^2 + (y - y_i)^2}} \quad (6.3.21)$$

Prin aplicarea teoremei lui Pitagora generalizată în triunghiul PO_iF_i rezulta relația:

$$l_i^2 = r_i^2 + f_i^2 - 2 \cdot r_i \cdot f_i \cdot \cos(\phi_i - \varphi_i) \quad (6.3.22)$$

Se înlocuiește expresia distanței r_i din 6.3.20 în 6.3.22 și rezultă :

$$l_i^2 = (x - x_i)^2 + (y - y_i)^2 + (z - z_i)^2 + f_i^2 - 2 \cdot \sqrt{(x - x_i)^2 + (y - y_i)^2 + (z - z_i)^2} \cdot f_i \cdot \cos(\phi_i - \varphi_i) \quad (6.3.23)$$

Valorile $f_1 = f_2 = f_3$ sunt cunoscute și se notează cu f . Unghiurile $\varphi_1 = \varphi_2 = \varphi_3$ sunt egale prin construcție și se notează cu φ .

Se formează un sistem de trei ecuații corespunzătoare celor 3 module cu necunoscutele x, y, z

$$\begin{cases} l_1^2 = (x - x_1)^2 + (y - y_1)^2 + (z - z_1)^2 + f^2 - 2 \cdot \sqrt{(x - x_1)^2 + (y - y_1)^2 + (z - z_1)^2} \cdot f \cdot \cos(\phi_1 - \varphi) \\ l_2^2 = (x - x_2)^2 + (y - y_2)^2 + (z - z_2)^2 + f^2 - 2 \cdot \sqrt{(x - x_2)^2 + (y - y_2)^2 + (z - z_2)^2} \cdot f \cdot \cos(\phi_2 - \varphi) \\ l_3^2 = (x - x_3)^2 + (y - y_3)^2 + (z - z_3)^2 + f^2 - 2 \cdot \sqrt{(x - x_3)^2 + (y - y_3)^2 + (z - z_3)^2} \cdot f \cdot \cos(\phi_3 - \varphi) \end{cases} \quad (6.3.24)$$

Se observă că termenul ϕ_i reprezintă o funcție trigonometrică care are drept parametru o expresie care conține necunoscutele.

Pentru rezolvarea sistemului (6.3.24) se atribuie o valoare constantă expresiilor ϕ_i . Se rezolvă sistemul 6.3.24 și prin metoda inversă considerându-se un punct $P(x, y, z)$ se determină lungimilor corespunzătoare pentru l_i notate l_{ci} . Se calculează eroarea $\Delta l_i = l_i - l_{ci}$ corespunzătoare punctului ales. În funcție de valoarea erorii se consideră un nou punct P situat în vecinătatea primului punct ales pentru care se calculează din nou l_{ci} . Se consideră determinat punctul P în momentul în care erorile Δl_i pentru $i = 1, \dots, 3$ au o valoare mai mică decât o valoare de referință stabilită anterior.

Se prezintă în continuare relațiile de calcul și algoritmul pentru modelul geometric invers în care se cunoaște poziția punctului P și se determină lungimile firelor pe porțiunea liniară PG_1 și pe arcul de cerc G_iF_i . Notațiile utilizate sunt cele din fig. 6.3.7.. Se consideră un sistem de referință ortogonal cu axa z paralelă cu dreapta O_iA_z . Se determină coordonatele punctului H_i . Proiecția în planul xOy a punctului H_i notat cu F_i' este situat pe dreapta care unește proiecția punctului P în același plan (punctul K) cu punctul O_i la distanța n_i de punctul O_i . Raportul în care punctul F_i' împarte segmentul O_iK este dat de relația :

$$k = \frac{n_i}{\sqrt{(x_p - x_{O_i})^2 + (y_p - y_{O_i})^2} - n_i} \quad (6.3.25)$$

și coordonatele punctului H_i sunt :

$$x_{H_i} = \frac{x_{O_i} + k \cdot x_p}{1 + k}; \quad y_{H_i} = \frac{y_{O_i} + k \cdot y_p}{1 + k}; \quad z_{H_i} = a_i \quad (6.3.26)$$

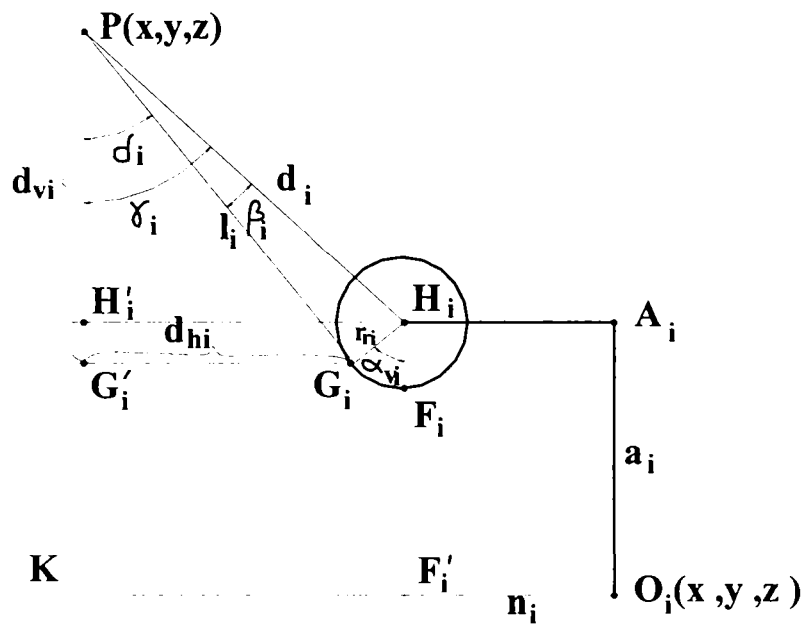


Fig. 6.3.7 Reprezentarea elementelor esențiale ale unui modul pentru elaborarea modelului geometric invers

Dreapta PG_i este tangentă la cercul cu centrul H_i și rază r_i care reprezintă rola deci $PG_i \perp H_iG_i$. Din triunghiul dreptunghic PG_iH_i rezultă distanța PG_i :

$$l_i = \sqrt{(x_p - x_{H_i})^2 + (y_p - y_{H_i})^2 + (z_p - z_{H_i})^2 - r_i^2} \quad (6.3.27)$$

S-a notat cu H_i' proiecția punctului H_i pe dreapta verticală PK . Din triunghiul dreptunghic PH_iH_i' :

$$\gamma_i = \text{atn} \left(\frac{\sqrt{(x_p - x_{H_i})^2 + (y_p - y_{H_i})^2}}{z_p - z_{H_i}} \right) \quad (6.3.28)$$

În triunghiul dreptunghic PH_iG_i se exprimă unghiul dintre dreptele G_iP și PH_i :

$$\beta_i = \text{atn} \left(\frac{r_i}{l_i} \right) \quad (6.3.29)$$

Unghiul $\delta_i = \gamma_i - \beta_i$ deci $d_{hi} = l_i \cdot \sin \delta_i$ și $d_{vi} = l_i \cdot \cos \delta_i$ (6.3.30)

Proiecția punctului G_i pe planul xOy este situat pe dreapta KO_i la distanța d_{hi} de punctul K . Se notează cu λ_i raportul în care proiecția punctului G_i pe planul xOy împarte segmentul KO_i :

$$\lambda_i = \frac{d_{hi}}{\sqrt{(x_p - x_{O_i})^2 + (y_p - y_{O_i})^2}} \quad (6.3.31)$$

Coordonatele punctului G_i , utilizat pentru simularea modelului geometric invers sunt:

$$x_{G_i} = \frac{x_p + \lambda_i \cdot x_{O_i}}{1 + \lambda_i}; \quad y_{G_i} = \frac{y_p + \lambda_i \cdot y_{O_i}}{1 + \lambda_i}; \quad z_{G_i} = z_p - d_{vi} \quad (6.3.32)$$

$$\alpha_{vi} = \frac{\pi}{2} - \delta_i, \text{ deci } G_i F_i \text{ are lungimea } l_{G_i F_i} = \frac{\alpha_{vi}}{2 \cdot \pi} \cdot r_{ri} \quad (6.3.33)$$

Lungimea firului între punctele P și F_1 este dată de relația :

$$l_{PF_i} = l_i + l_{G_i F_i} = \sqrt{(x_P - x_{Hi})^2 + (y_P - y_{Hi})^2 + (z_P - z_{Hi})^2} - r_i^2 + \frac{\alpha_{vi}}{2 \cdot \pi} \cdot r_{ri} \quad (6.3.34)$$

Pentru ca la rezolvarea sistemului 6.3.24 să se pornească de la o valoare pentru ϕ , cât mai apropiată de cea reală este necesară realizarea unei baze de date care să cuprindă coordonatele punctului P corespunzătoare lungimilor l_{PF_i} , $i = 1, \dots, 3$. Pentru completarea bazei de date se utilizează modelul geometric invers descris anterior. Se generează o matrice tridimensională de puncte P în spațiul de lucru al digitizorului, cu pas constant pe cele 3 direcții a sistemului de coordonate. Cu cât pasul ales este mai mic cu atât la determinare punctului P corespunzător lungimii celor 3 fire se pornește de la o valoare mai apropiată de valoarea reală și numărul de iterații este mai mic. Dar alegerea unui pas foarte mic duce la creșterea nepermis de mult al numărului de înregistrări din baza de date. O înregistrare conține coordonatele punctului P corespunzătoare lungimii celor 3 fire. Deci trebuie aleasă o valoare optimă din punct de vedere al consumului de memorie necesar pentru baza de date și timpul de extragere a înregistrării corespunzătoare pe de o parte și a numărului de iterații necesare pe de altă parte.

O metodă de determinare a coordonatelor punctului P pe baza cunoașterii lungimilor celor trei fire (de fapt a distanțelor PG_i , cu $i = 1, \dots, 3$) se bazează exclusiv pe calculul geometric invers. Prin utilizarea metodologiei prezentate anterior a fost completată baza de date cu lungimile firelor corespunzătoare punctelor P alese în spațiul de lucru al digitizorului 3D. Se notează cu l_{ii} valoarea distanței PG_i din baza de date și cu l_{ir} distanța reală a segmentului PG_i obținută prin contorizarea impulsurilor generate de traductoare. Pentru cele trei distanțe PG_i se alege din baza de date punctul P pentru care suma $S = \sum_{i=1}^3 |l_{ii} - l_{ir}|$ are valoarea minimă. În jurul acestui punct P se

consideră puncte la o distanță egală cu jumătate din valoarea pasului de divizare a spațiului de lucru în 6 direcții paralele cu axele de coordonate și alese în sensul pozitiv și respectiv negativ al fiecăreia din axele de coordonate. Pentru fiecare din aceste puncte se calculează lungimile corespunzătoare a firelor și suma S. Procedeu se repetă pentru punctul pentru care suma S este minimă. Se procedează în mod asemănător până la atingerea unei valori pentru S mai mică decât o valoare maximă admisibilă. Algoritmul poate fi optimizat ținând cont că probabilitatea cea mai mare este ca punctul să se deplaseze pe o direcție indicată aproximativ de linia care unește ultimele 2 puncte date. La inițializarea digitizorului se specifică pentru preluarea coordonatelor unor șiruri de puncte o distanță minimă d_{\min} pentru ca punctele să fie considerate ca două puncte distincte. Deci poziția punctului curent, în cazul determinării coordonatelor unor șiruri de puncte va fi situată la o distanță aproximativ egală cu d_{\min} față de poziția punctului anterior.

6.3.3 Proiectarea părților componente ale sistemului

6.3.3.1. Subsistemul mecanic

Subsistemul mecanic este constituit din următoarele trei elemente principale: structură de rezistență, module pentru măsurarea lungimii firului și palpator. Se reamintește ca prin măsurarea lungimii firelor se înțelege determinarea pe baza traductoarelor incrementale de rotație a lungimii firului pe porțiunea $F_iG_iP_i$ unde $i=1, \dots, 5$. La proiectarea și realizarea subsistemului mecanic s-a pornit de la modulele de măsurare a lungimii firului. Proiectarea asistată de calculator sistemului de digitizare s-a realizat prin utilizarea aplicației AutoCAD versiunea 2000. Modulele de măsurare a lungimii firului sunt notate în continuare prin Mod_f . În figura 6.3.8 este prezentat un modul în mod umbrat. Din punct de vedere al construcției mecanice, cele cinci module Mod_f sunt identice.

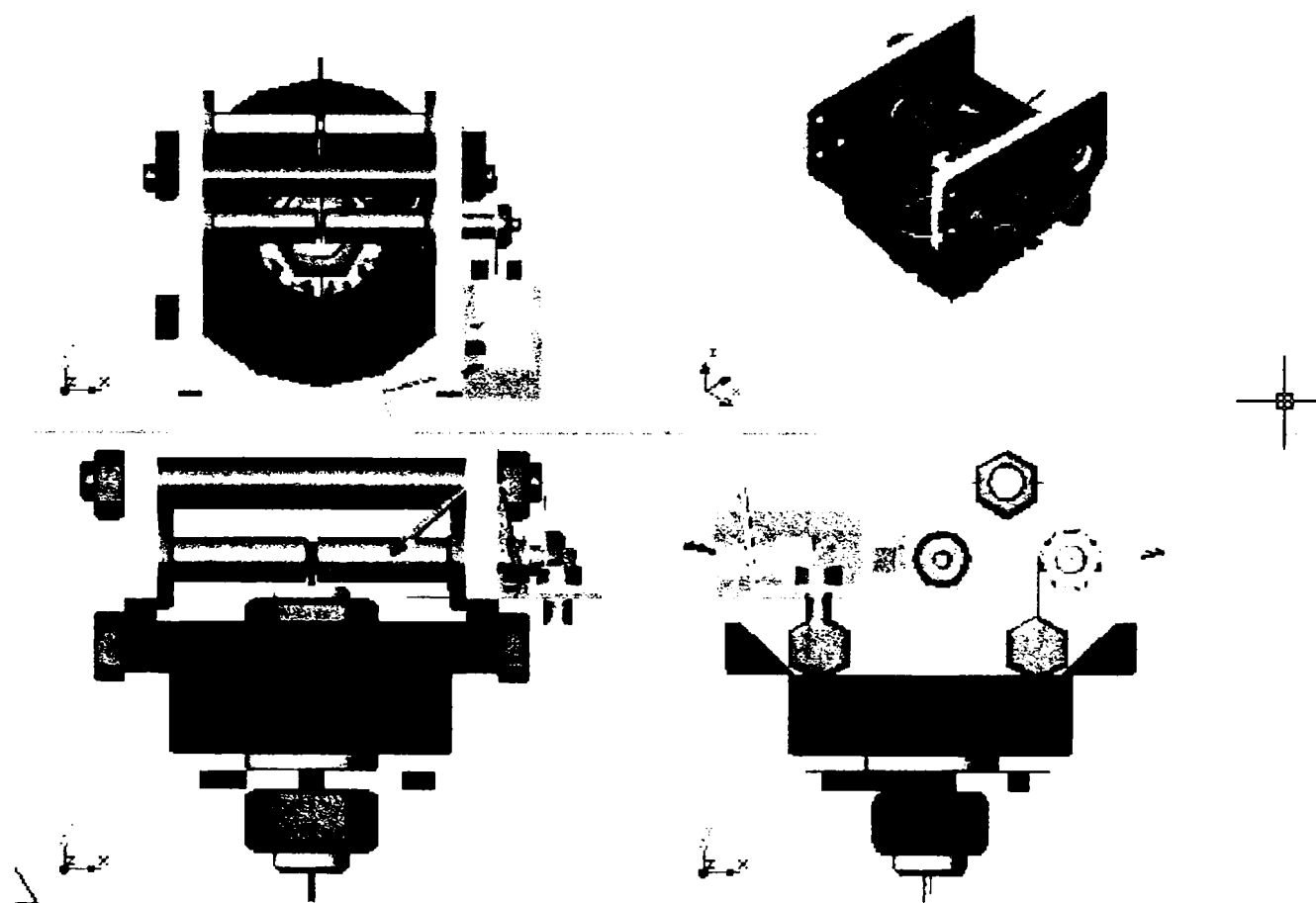


Fig. 6.3.8 Modul de măsurare a lungimii firului în reprezentare umbrată

Lagărele tuturor cuplelor de rotație sunt de rostogolire pentru minimizarea frecărilor. Cu cât frecarea în lagăre are o valoare mai ridicată, cu atât unghiul dintre proiecția dreptelor P_iF_i și O_iF_i în planul xOy are o valoare mai mare și implicit eroarea de măsurare a lungimii firului este mai mare.

În figura 6.3.9 se prezintă o vedere de sus a modului de măsurare, în fig. 6.3.10 se prezintă vederea din lateral a modului, în figura 6.3.11 o secțiune printr-un plan vertical care conține axa rolei solidară cu traductorul incremental de rotație iar în figura 6.3.12 se prezintă un detaliu în vedere umbrită a modului de fixare a optocuplei.

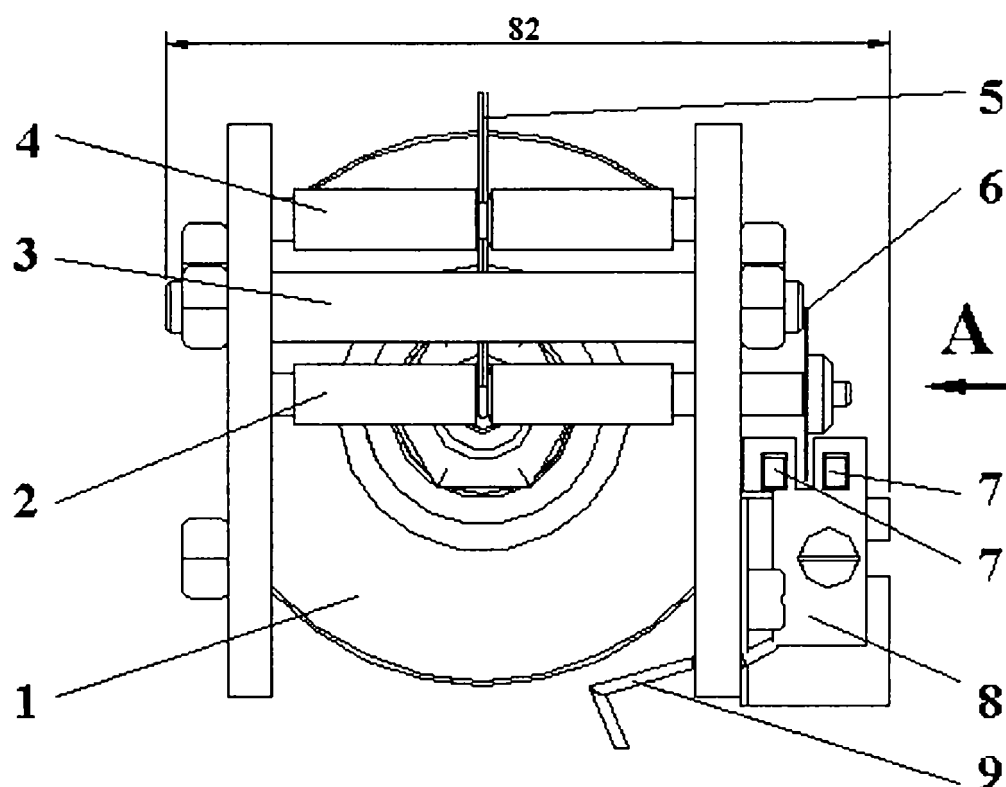


Fig. 6.3.9. Vedere de sus a modului de măsurare

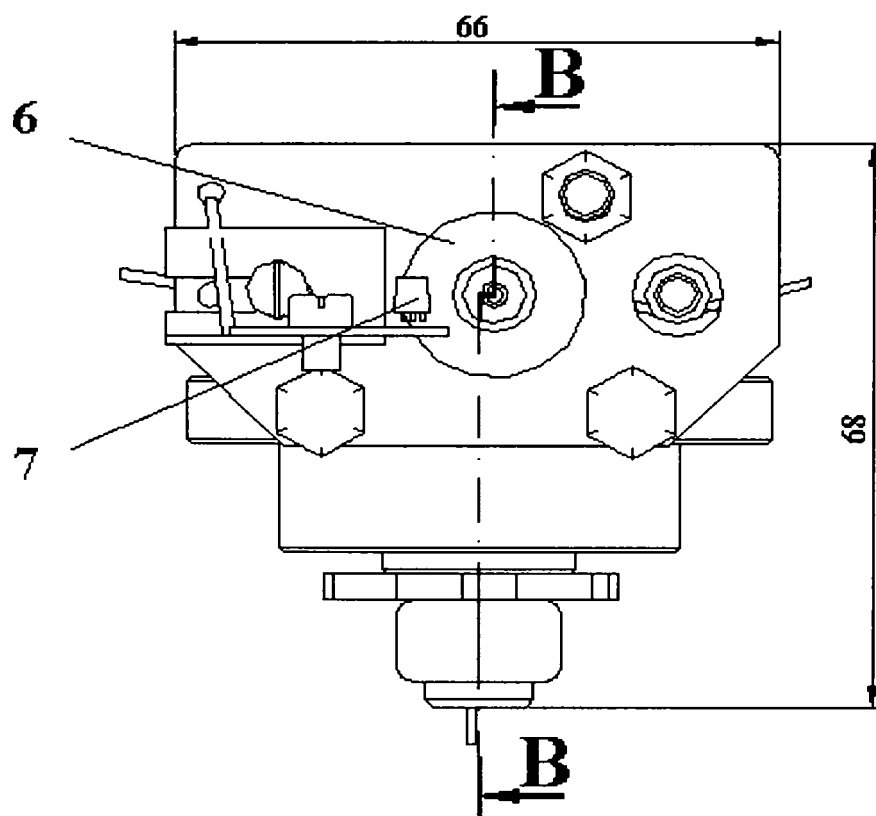


Fig. 6.3.10 Vedere laterala rotită cu 90 grade a modului de măsurare

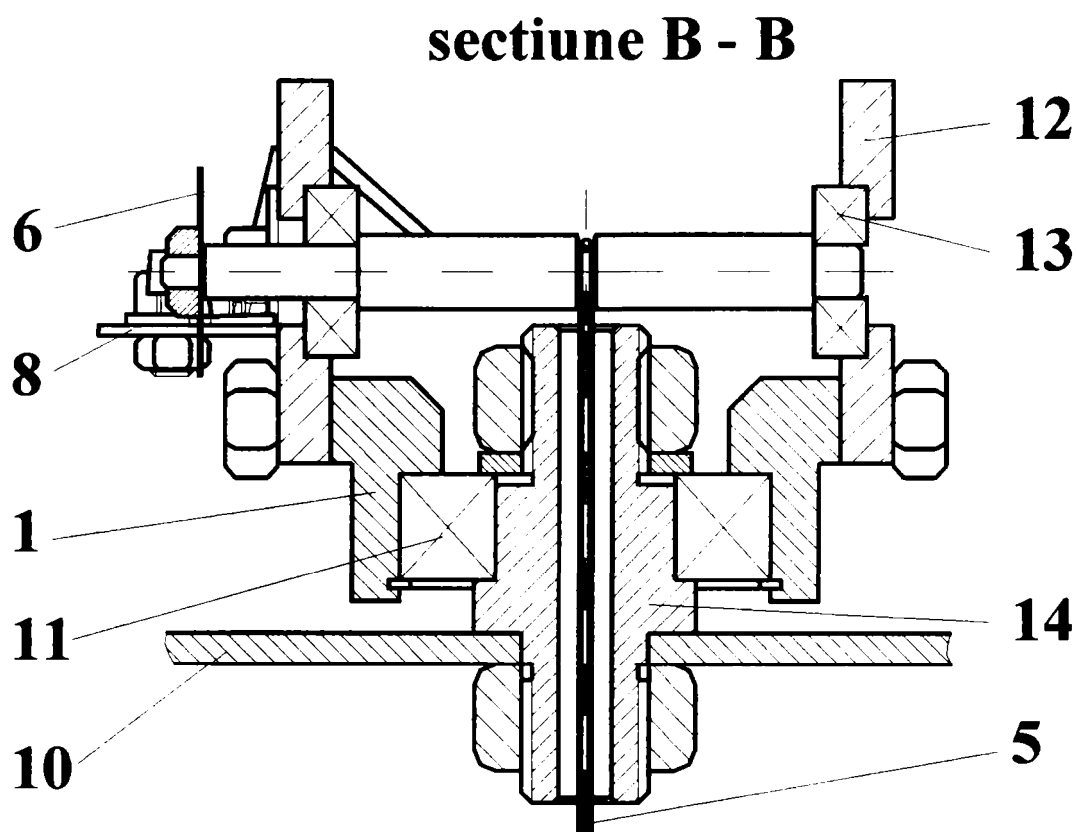


Fig. 6.3.11 Secțiune prin modulul de măsurare

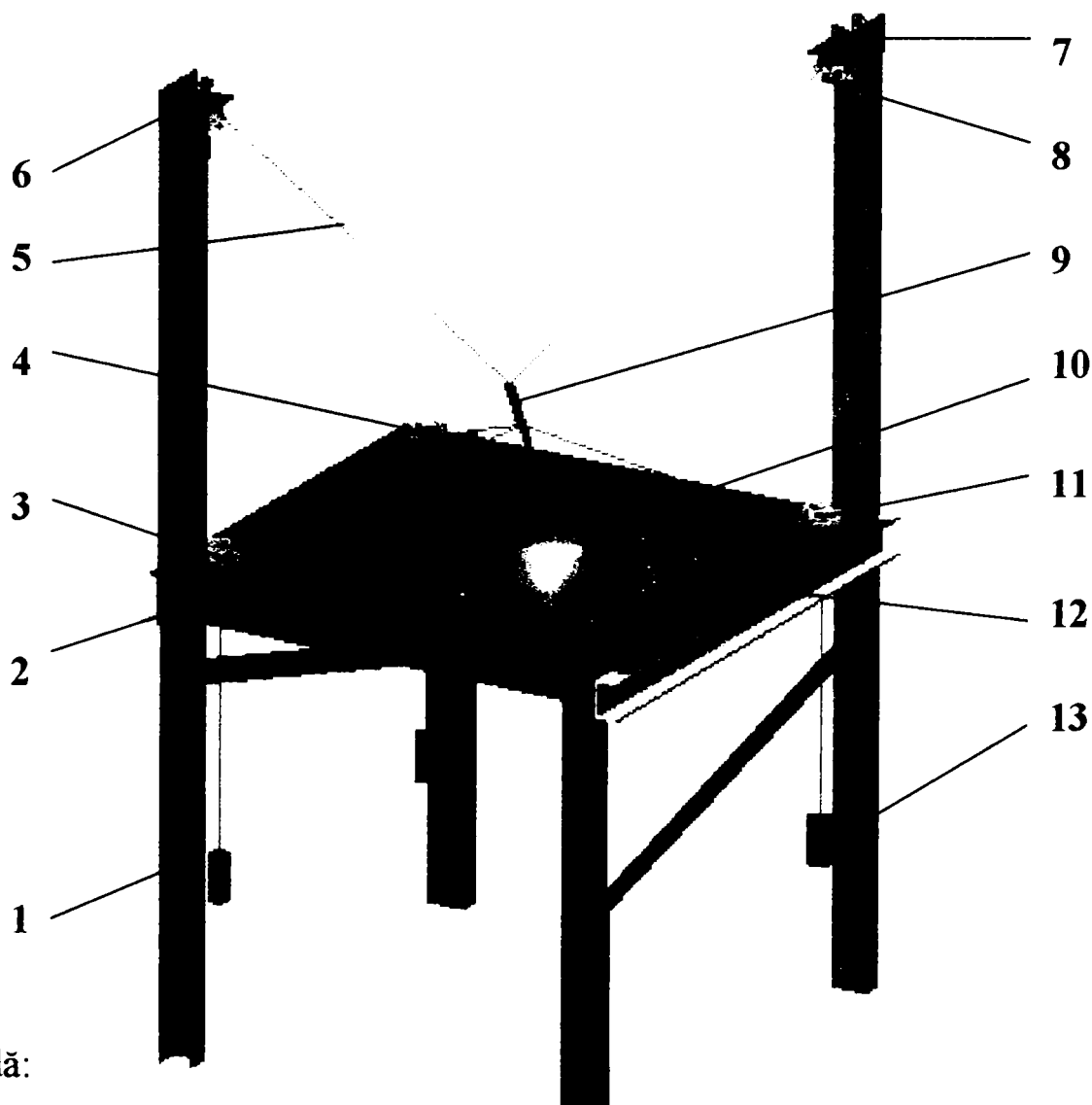
Elementele componente principale ale modulului de măsurare a lungimii firului :

1. - corp rotație verticală
2. - axul de susținere a discului cu fante al traductorului incremental de rotație
3. - element de rigidizare
4. - ax de sprijin fir
5. - fir
6. - discul cu fante al traductorului
7. - element optocuplor
8. - placa susținere elemente semiconductoare optocuplă
9. - conductor electric pentru legatura optocuplei cu cicuitul electronic
10. - placă susținere modul măsurare
11. - rulment radial
12. - placă laterală
13. - rulment radial
14. - ax pivotare



Fig 6.3.12 Detaliu al modului de fixare a optocuplei

Digitizorul 3D cu fire în a doua variantă de montare a modulelor Mod_f este prezentat în figura 6.3.13. Vârful palpatorului (9) este adus în contact cu suprafața obiectului care se digitizează (10) de către operator. Deplasarea palpatorului produce deplasarea poziției punctelor în care firele sunt fixate de palpator, ceea ce determină rotirea rotelor modulelor de măsurare a lungimii firelor: (3), (4), (6), (8) și (11). Digitizorul are în componență o masă fixă (2) pentru așezarea obiectelor. În cazul în care obiectele care urmează a fi digitizate au o suprafață complexă acestea sunt fixate pe masa rotativă indexabilă 12. Masa poate fi blocată în 8 poziții distincte. Schimbarea poziției mesei este indicată de către operator aplicației software care asigură preluarea datelor primare de la digitizor. Greutățile 13 au rolul de a asigura tensionarea firului pentru înfășurarea pe rotele modulelor de măsurare a firelor.



Legendă:

- | | |
|--------------------------|------------------------------------|
| 1. element de susținere | 8. modul nr. 4 |
| 2. placa fixă | 9. palpator |
| 3. modul nr.1 | 10. obiectul care se digitizează |
| 4. modul nr.2 | 11. modul nr. 3 |
| 5. fir | 12. masă rotativă |
| 6. modul nr. 4 | 13. greutate pentru tensionare fir |
| 7. suport modul superior | |

Fig. 6.3.13 Vedere de perspectivă a digitizorului cu fire

În fig. 6.3.15. este prezentată o secțiune prin palpator. Firele sunt fixate de palpator prin intermediul elementelor (3) și (6). Aceste elemente pot avea o mișcare de rotație liberă în jurul axei palpatorului, ceea ce împiedică înfășurarea firelor la rotirea mânerului în jurul axei sale. Această rotire, neintenționată, poate apărea în timpul digitizării. Vârful palpatorului (1) și elementul de ghidare (7) sunt solidarizate de mâner prin intermediul știfturilor filetate 4. La acest digitizor, contactul cu suprafața nu este sesizată prin intermediul palpatorului. Comanda preluării coordonatelor unui punct singular sau a unor șiruri de puncte este dată prin intermediul unor microîntrerupătoare situate pe un panou de comandă fixat de structura de rezistență a digitizorului cu fire.

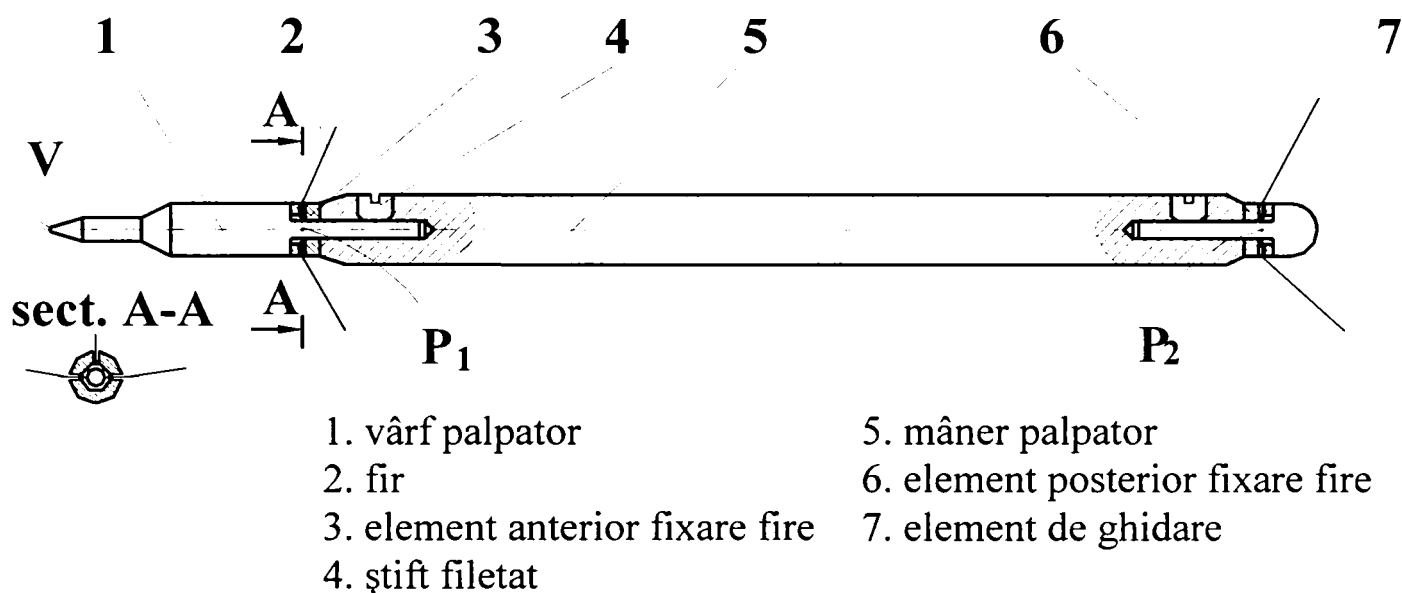


Fig. 6.3.14 Secțiune prin palpator

6.3.3.2. *Subsistemul electronic*

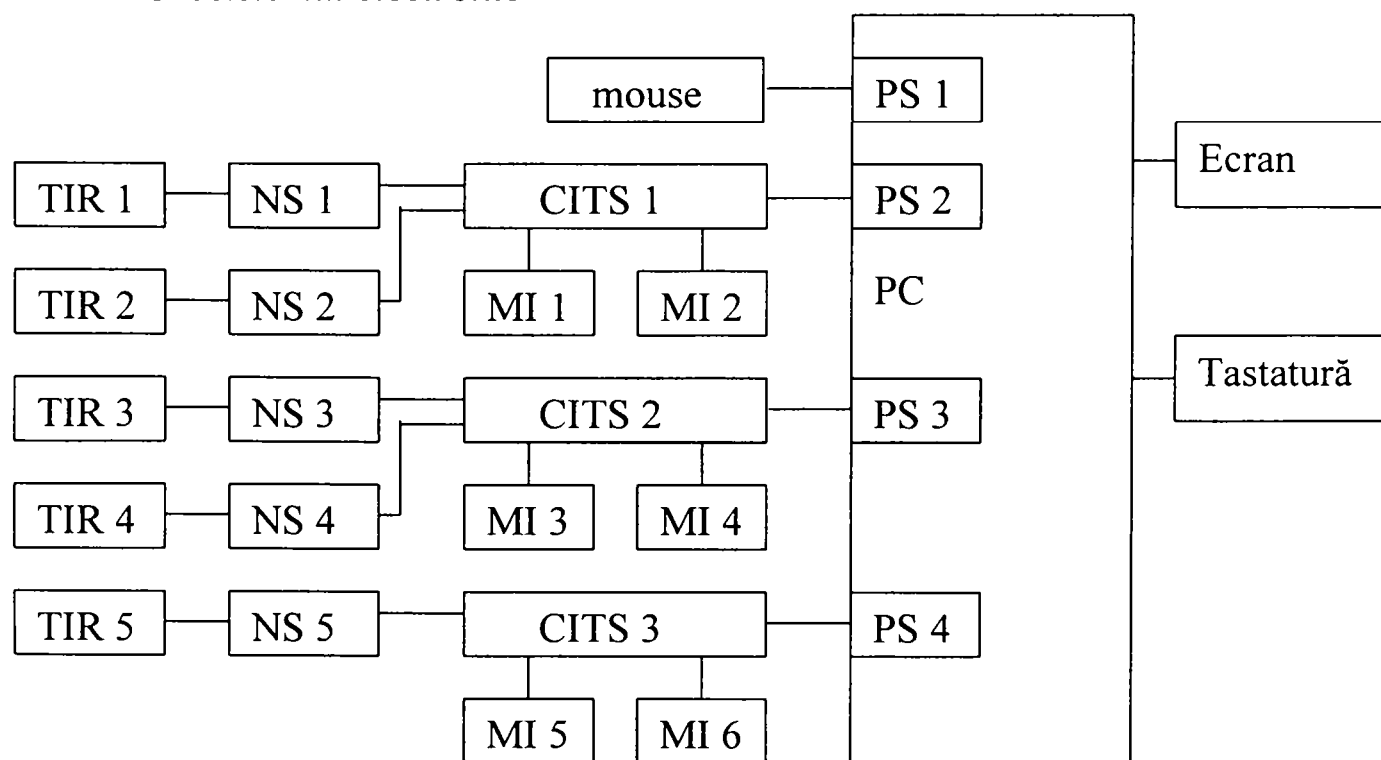


Fig. 6.3.15. Schema bloc a subsistemului electronic

Legendă fig. 6.3.15.:

- TR 1,...,5 - traductor incremental de rotație
- NS 1,...,5 - numărător și circuit detectare sens
- MI 1,...,6 - micro-întrerupător
- CITS 1,...,3 - circuit pentru transmitere serială date
- PS 1,...,4 - porturi seriale a calculatorului PC

Subsistemul electronic al digitizorului cu fire, a cărei schemă bloc este prezentată în figura 6.3.15. este realizat prin utilizarea componentelor electronice a mouseului utilizat pentru interacțiunea în mod grafic cu calculatorul. La alegerea acestei soluții s-a avut în vedere realizarea preciziei impuse și costul redus al componentelor datorită fabricației de masă.

Electronica e similară cu cea de la RRR iar modul de interfațare cu PC-ul este prin porturi seriale cu placă de extensie.

6.3.3.3. Subsistemul informatic

Pentru îndeplinirea sarcinii de determinare a coordonatelor punctului caracteristic al digitizorului cu fire și preluarea continuă a unor șiruri de puncte au fost realizate o serie de subrutine pentru rezolvarea unor probleme punctuale. Pentru realizarea subrutinelor a fost utilizat pe cât posibil cod sursă de la rutinele realizate pentru digitizorul 3D RRR cu brațe articulate prezentate la punctul 2.2.5.

a). Indicarea poziției de acasă a digitizorului cu inițializarea contoarelor corespunzătoare fiecărui modul

Se utilizează rutina descrisă la 6.2.4.2. dar se utilizează alt fișier de tip *.ini în care sunt stocate datele caracteristice digitizorului cu fire.

b) Preluarea informațiilor de la porturile seriale a calculatorului PC despre numărul de impulsuri corespunzătoare fiecărui traductor incremental de rotație și a stării microîntrerupătoarelor. Se utilizează rutina descrisă la punctul 6.2.4.3.

c). Calculul geometric invers (pentru varianta a doua);

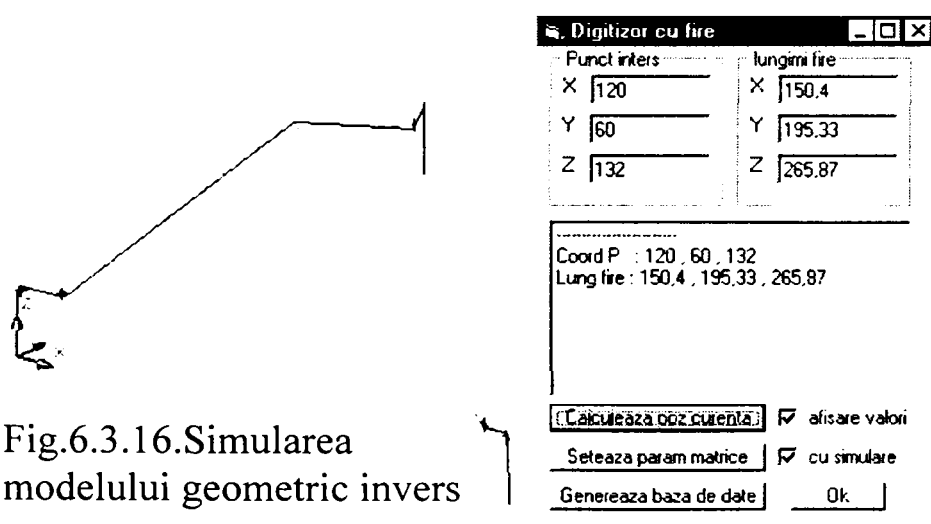


Fig.6.3.16.Simularea modelului geometric invers

În fig. 6.3.16. se prezintă o captură de ecran din rularea aplicației software care implementează algoritmul modelului geometric invers. Aplicația este utilizată pentru completarea bazei de date cu lungimile firelor corespunzătoare unei mulțimi de poziții a punctului caracteristic al digitizorului în spațiul de lucru. Modelul matematic și algoritmul este prezentat la punctul 6.3.2.

c). Calculul geometric direct (pentru prima varianta de model matematic) utilizează o subrutină pentru determinarea intersecției a trei sfere pentru care se cunosc pozițiile centrelor și raza pe baza relațiilor prezentate la punctul 6.3.2. Această subrutină se aplică de două ori. Prima dată se aplică pentru determinarea poziției punctului P1 al digitizorului pe baza informațiilor obținute de la primele 3 module. Apoi se aplică pentru determinarea poziției punctului P2 care împreună cu P1 dă orientarea dreptei caracteristice. Pentru determinarea punctului P2 se utilizează informațiile oferite de modulele 4 și 5 împreună cu coordonatele punctului P1 și distanța cunoscută dintre P1 și P2. Determinarea poziției vârfului palpatorului se realizează pe baza relației 6.3.1. Pentru verificarea modelului matematic s-a utilizat simularea în AutoCAD cu ajutorul unei aplicații VisualBasic. O captură de ecran din preluată în timpul rulării aplicației este prezentată în fig. 6.3.17.

modul 1		modul 2		modul 3		modul 4		modul 5	
X	0	X	1000	X	0	X	0	X	1000
Y	0	Y	0	Y	1000	Y	1000	Y	0
Z	1000	Z	1000	Z	1000	Z	2000	Z	2000
r1	617.64	r1	762.55	r3	813.31	r4	903.59	r5	858.18

coordonate punct caracteristic P1			dist P1P2				
X	400	Y	360	Z	1303.1		200
coordonate P2			dist P1V				
X	465	Y	425	Z	1480.8		50
Coordonatele virfului palpatorului			<input checked="" type="checkbox"/> cu simulare in AutoCAD				
X	383.8	Y	343.8	Z	1258.7	<input type="button" value="Calculeaza"/>	<input type="button" value="Ok"/>
cosinusurile directoare a dreptei caracteristice							
a	0.32	b	0.32	g	0.89		

Fig.6.3.17 . Verificarea modelului geometric direct pentru prima variantă

d). Determinarea coordonatelor unor șiruri de puncte

Utilizează modelul geometric direct în cazul primei variante și rutina descrisă la subpunctul c) pentru cea de-a doua soluție constructivă. Există opțiuni pentru salvarea pozițiilor succesive a punctului caracteristic în fișiere text sau Excel. Șirurile de puncte reprezentând pozițiile succesive a punctului caracteristic pot fi reprezentate în timp real în mediul AutoCAD prin polilinii 3D a căror vertexuri sunt punctele din șir.

f). Interfață pentru programul de generare a suprafețelor pornind de la șiruri de puncte
Se utilizează rutina prezentată punctul la 6.2.4.8.

7. Contribuții la realizarea unor aplicații ale digitizoarelor 3D

7.1. Introducere

Așa cum a s-a prezentat în capitolul 2.2.6. “Aplicații ale digitizoarelor 3D cu palpator” , digitizoarele 3D au o serie de aplicații atât în domeniul industrial cât și în artă, medicină, etc.

În prezentul capitol autorul își aduce contribuția la realizarea unor aplicații în domenii “clasice” din punct de vedere al utilizării digitizoarelor 3D și în domenii în care digitizoarele 3D cu palpator cu conducere manuală nu sunt încă frecvent utilizate. Se prezintă aplicații ale digitizoarelor 3D în programarea roboților industriali (subcapitolul 7.2), la generarea modelului CAD pe baza informațiilor prelevate cu digitizorul 3D (subcapitolul 7.3) , la proiectarea asistată de calculator a mobilierului de ședere în industria mobilei (subcapitolul 7.4), pentru sculptură virtuală cu digitizorul 3D și la compararea entităților CAD cu entități geometrice indicate cu digitizorul 3D pe obiecte existente fizic (subcapitolul 7.6).

7.2. RobSim - aplicație pentru programarea off-line a roboților în mediu virtual cu interfață pentru digitizoare 3D

7.2.1. Prezentarea generală a aplicației RobSim

Pentru ilustrarea aplicațiilor de programare a roboților industriali cu ajutorul digitizoarelor 3D autorul a realizat după o concepție proprie aplicația software denumită RobSim. În cursul studiului bibliografic, autorul a constatat că nu există aplicații de programare a roboților care să dispună de interfață pentru digitizoare 3D. În paragrafele 7.2.2. - 7.2.4 se prezintă elementele necesare pentru înțelegerea modului de utilizare a aplicației RobSim, în paragrafele 7.2.5 - 7.2.9 se prezintă interfața aplicației pentru digitizoarele 3D iar în anexa 4 se prezintă arhitectura aplicației software.

Utilizarea aplicației RobSim presupune cunoștințe de modelare în mediul AutoCAD 3D și cunoștințe de robotică. Cu ajutorul aplicației se pot realiza "roboți virtuali" cu topologie serială care pot fi programați pentru simularea în mediu virtual 3D a desfășurării unor sarcini concrete de lucru. Aplicația permite verificarea calculelor teoretice obținute pentru modelul geometric direct de ordinul 0 al robotului. Față de alte medii de programare grafică existente pe piață (ex. COSIMIR, Workspace, etc.), aplicația RobSim are în plus o interfață pentru digitizoare 3D. Totodată, aplicația permite crearea propriilor modele de roboți, facilitate mai rar întâlnite la aplicații livrate de firme producătoare de roboți împreună cu robotul vândut. De exemplu aplicația "Robot Studio" oferită de ABB permite simularea doar a modelelor de roboți ABB. Evident, RobSim nu are o serie de facilități pe care aplicațiile amintite anterior le au, dar, în măsura în care se justifică, aplicația poate fi dezvoltată.

Programarea roboților virtuali se poate realiza prin instruire sau cu ajutorul unui editor de text. Instrucțiunile programului sunt specifice aplicației RobSim. Pentru elaborarea programului în limbajul specific al unui anumit robot este necesar un postprocesor. Ca exemplificare, aplicația dispune de un postprocesor care permite elaborarea instrucțiunilor în vederea programării la nivelul microcontrollerului pentru robotul SCORBOT ER III. Aplicația pentru comanda efectivă a robotului real este realizată de către drd. ing. Ciupe Valentin din cadrul departamentului Mecatronică al Universității "Politehnica" Timișoara.

Pornind de la conceptul de perechi de sisteme de referință [KOV 2002], este posibilă atașarea de perechi de sisteme de referință corespunzătoare cuplelor cinematice și respectiv ofset-urilor. În orice moment este posibilă determinarea situației unui anumit sistem de referință față de sistemul de referință fix atașat robotului prin simpla selectare a blocului AutoCAD care reprezintă "materializarea" sistemului de referință.

Pentru ca desfășurarea unei sarcini de lucru să poată fi simulată în RobSim este necesară parcurgerea următoarelor etape :

- realizarea modelului geometric 3D al robotului în mediul AutoCAD;
- indicarea corespondenței în modelul geometric 3D a axelor, elementelor și componentelor efectorului final al robotului modelat și precizarea unor date specifice;
- realizarea programului corespunzător desfășurării sarcinii prin instruire sau cu ajutorul editorului.

În fig 7.2.1 este prezentată simplificat arhitectura aplicației:

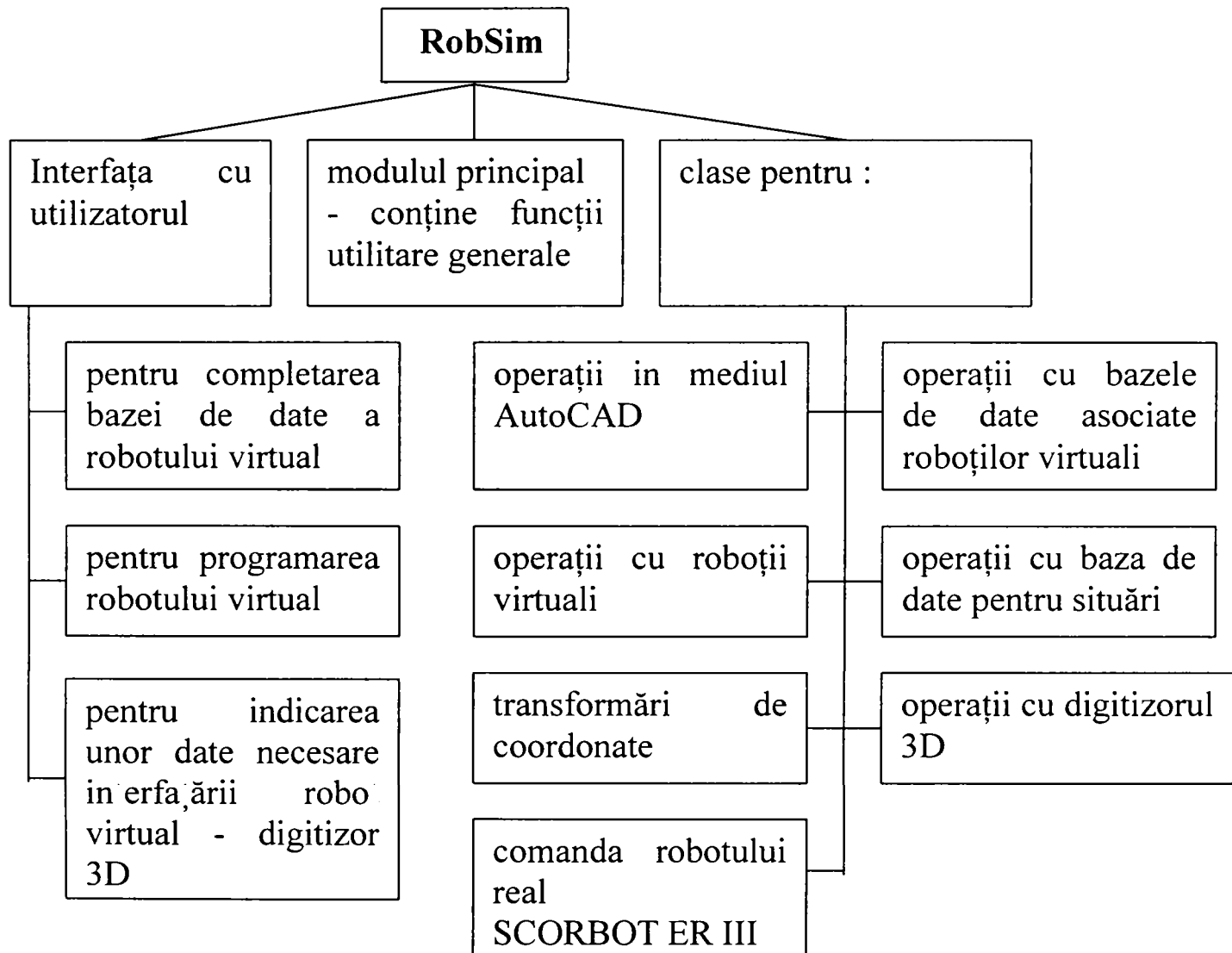


Fig. 7.2.1. Arhitectura aplicației RobSim

7.2.2. Construirea modelului programabil al robotului pornind de la modelul geometric 3D

Modelarea geometrică a componentelor elementelor robotului se realizează prin entități de tip “solid” iar a axelor cuplelor cinematice conducătoare prin entități “line”. Pentru ca modelul robotului să poată deveni programabil este necesară specificarea unor date suplimentare care sunt stocate într-o bază de date Access. Structura bazei de date este comună tuturor roboților dar este utilizată o instanță

pentru fiecare model de robot. În figura 7.2.2 este prezentată structura bazei de date pentru roboții modelați în RobSim.

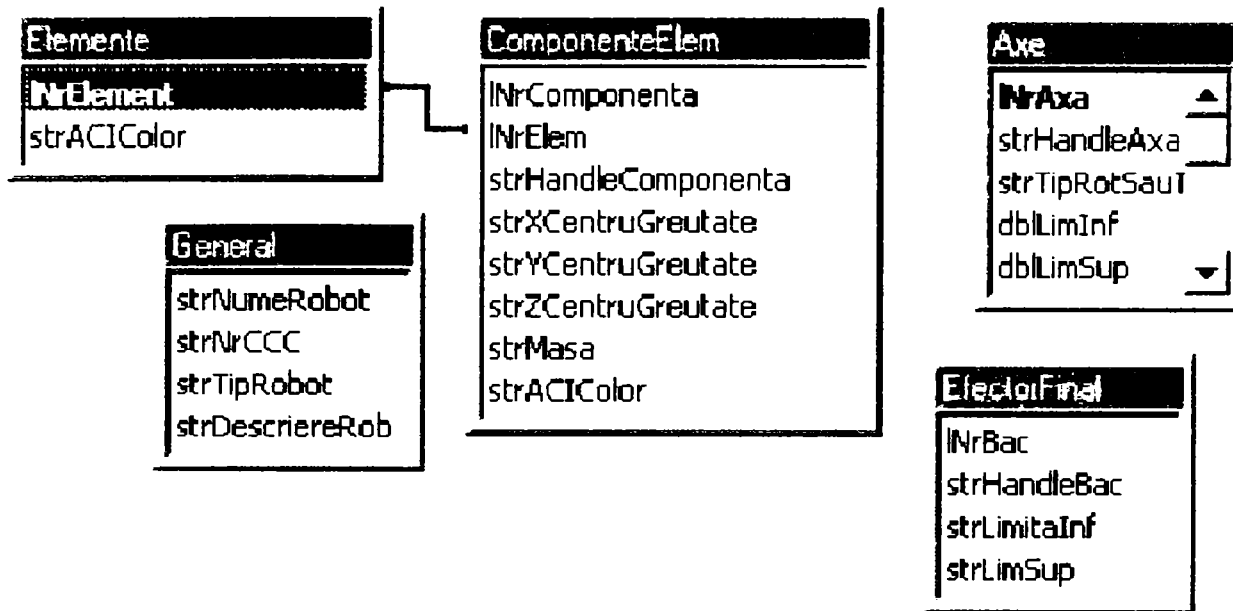


Fig. 7.2.2. Structura bazei de date pentru un robot

Legătura între datele stocate în baza de date și entitățile geometrice este realizată prin “handle” care este un șir de caractere ce referă entitatea respectivă. Acest “handle” se păstrează la închiderea documentului AutoCAD care conține modelul geometric al robotului. Modul de lucru pentru completarea bazei de date a robotului este următorul : se selectează în AutoCAD entitatea geometrică apoi, într-o casetă de dialog de tip Windows se introduc datele referitoare la componenta respectivă din modelul robotului. Ca exemplificare, se prezintă în continuare modul de indicare a datelor pentru o cupla cinematică conducătoare a robotului.

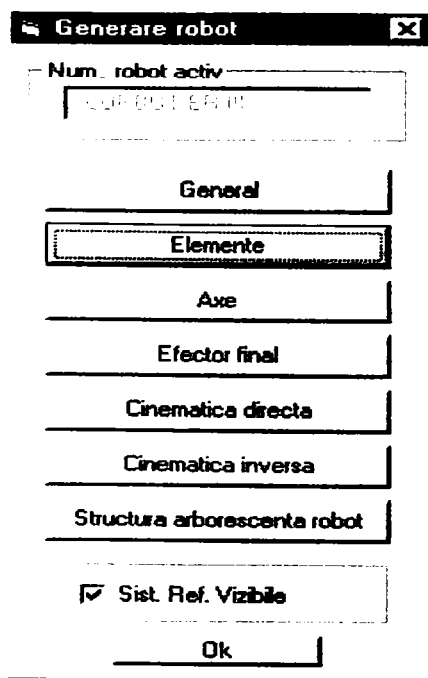


Fig. 7.2.3. Meniul pentru modificări

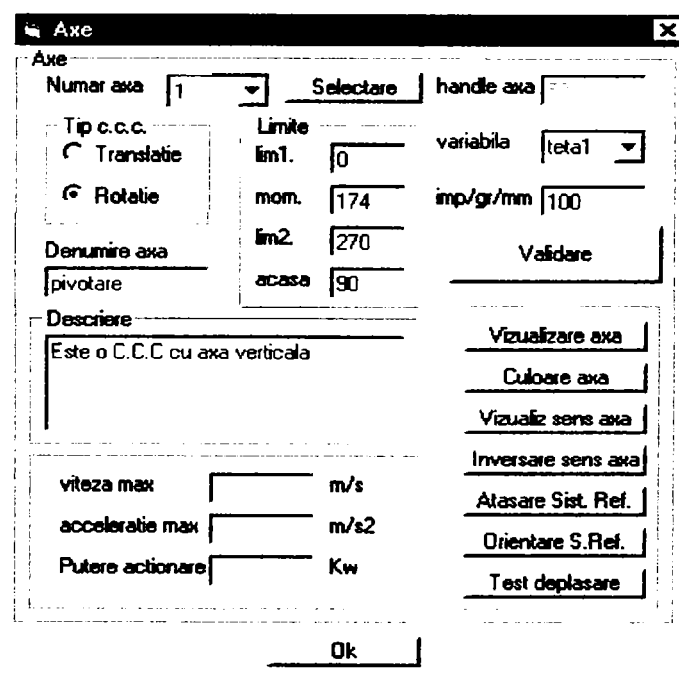


Fig. 7.2.4. Casetă de dialog pentru indicarea datelor referitoare la o axa a robotului

Din caseta de dialog pentru modificări (fig. 7.2.3), se selectează opțiunea “Axe” prin care se deschide caseta de dialog prezentată în figura 7.2.4. În urma acționării butonului de comandă “Selectare”, utilizatorul este solicitat să selecteze în modelul AutoCAD linia care reprezintă axa cuplei cinematice respective. Se selectează tipul axei, apoi se completează limitele de cursă (lim. 1 și lim. 2), situarea relativă a elementelor cuplelor cinematice pentru poziția de “acasă”, și pentru poziția momentană. Poziția momentană este cea corespunzătoare momentului selectării entității în AutoCAD. Numărul de impulsuri pe grad (pentru rotație) sau mm (pentru translație) corespunzător cuplei cinematice conducătoare curente este necesar în cazul în care prin RobSim se comandă un robot real la nivel de microcontroller în coordonatele c.c.c. (cuplelor cinematice conducătoare) . În urma specificării datelor de către utilizator, la acționarea butonului de comandă “validare” se verifică datele și se salvează în baza de date.

La acționarea butonului de comandă “Atașare Sist. Ref” (fig. 7.2.4.) se creează în documentul AutoCAD un bloc care reprezintă sistemul de referință (fig. 7.2.5) având direcția axei z suprapusă peste linia care modelează axa curentă. Utilizatorul este solicitat să specifice elementul de care se atașează sistemul de referință creat. În timpul fazei de instruire și pe parcursul simulării, blocul care reprezintă sistemul de referință se deplasează solidar cu elementul de care este atașat. O facilitate a aplicației este afișarea matricii de transformare corespunzătoare trecerii de la sistemul de referință fix atașat robotului la un sistem de referință selectat de utilizator (utilizatorul selectează blocul AutoCAD ce simbolizează sistemul de referință) . Această facilitate este foarte importantă pentru procesul de instruire a studenților deoarece permite verificarea formulelor modelului cinematic direct de ordinul 0.

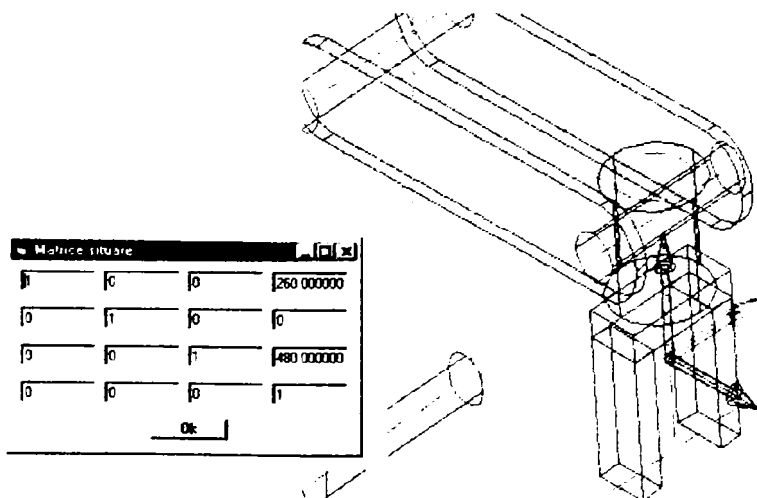


Fig. 7.2.5 Blocul AutoCAD care simbolizează sist. de referință și matricea de situare

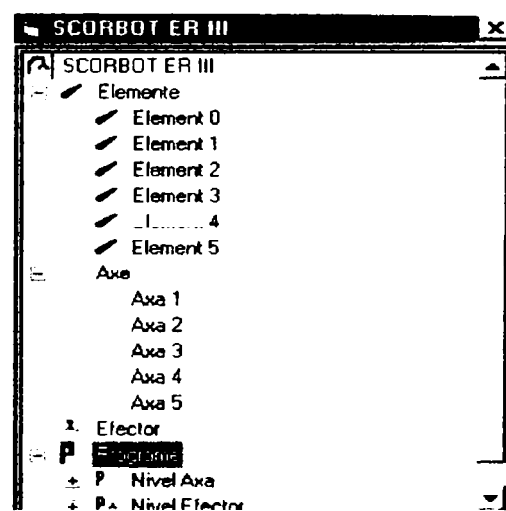


Fig. 7.2.6. Structura arborescentă asociată modelului robotului

Prin înlocuirea în formulele determinate de student pentru elementele matricii de transformare corespunzătoare trecerii de la sistemul de referință fix atașat robotului la sistemul de referință atașat efectorului final a variabilelor cu valorile momentane a pozițiilor corespunzătoare tuturor c.c.c , se obțin valorile numerice pentru matricea de

situare a efectorului final. Această matrice de situare calculată poate fi comparată cu matricea de situare afișată de către aplicație, ajutând la depistarea erorilor de calcul.

În mod asemănător specificării datelor pentru axe se procedează pentru elementele robotului și mecanismul de prehensiune. Aceste date pot fi schimbate și ulterior, accesul la fiecare element sau axă fiind facilitat de structura arborescentă asociată fiecărui model de robot (fig. 7.2.6).

7.2.3. Programarea robotului virtual și desfășurarea sarcinii de lucru

7.2.3.1 Metode de programare puse la dispoziție de aplicația RobSim

În urma completării setului de date minimale pentru c.c.c. , elementele și componentele efectorului final se poate trece la programarea modelului virtual al robotului. Pentru programare există o serie de opțiuni: programare la nivelul c.c.c. în regim de instruire sau cu ajutorul unui editor încorporat al aplicației, programarea la nivel cartezian prin instruire sau cu ajutorul editorului în limbajul SCORBASE, urmărirea de către efectorul final a robotului a piesei virtuale fixate în mecanismul de prehensiune, crearea unei liste de situații cu indicarea situațiilor prin diferite metode. În punctele următoare este descrisă fiecare din aceste facilități.

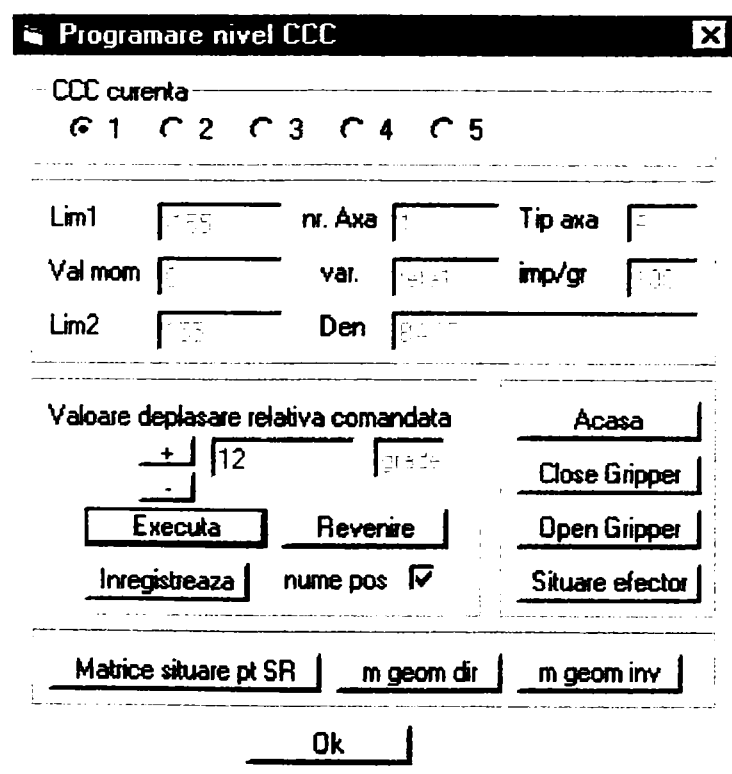


Fig. 7.2.7 Panoul de instruire pentru robot

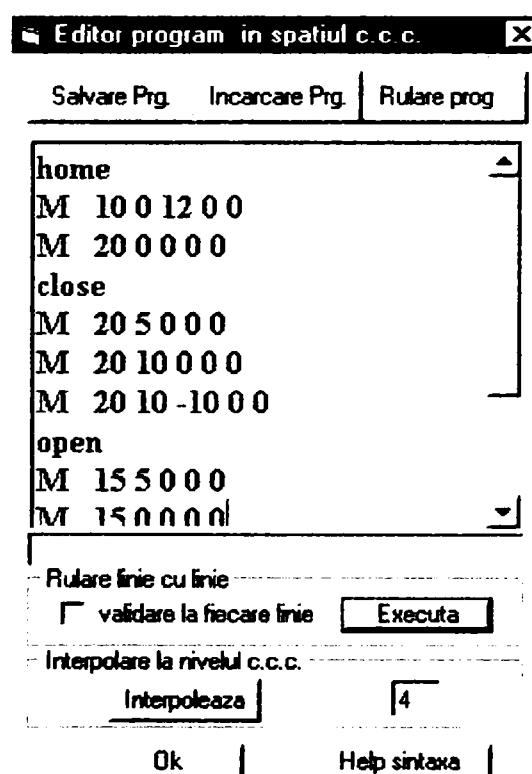


Fig. 7.2.8 Editorul încorporat în RobSim

7.2.3.2 Programare la nivelul c.c.c. în regim de instruire

Pentru programarea la nivelul c.c.c. nu este necesară rezolvarea prealabilă a problemei cinematico-poziționale directe. Programarea prin instruire se realizează prin intermediul unei casete de dialog care modelează panoul de instruire (fig. 7.2.7.).

Modul de programare este foarte simplu. Se selectează cupla cinematică conducătoare curentă din partea superioară a panoului de instruire, apoi se indică valoarea deplasării în caseta de text. Valoarea deplasării se specifică în grade pentru c.c.c. de rotație și în mm pentru c.c.c. de translație și este relativă față de poziția momentană. Pentru realizarea deplasării se acționează butonul de comandă “Execută”. Pentru renunțarea la ultima deplasare comandată se utilizează butonul de comandă “Revenire”. Aplicația permite revenirea la orice poziție anterioară. Specificarea deplasării se poate realiza și cu ajutorul butoanelor de comandă cu etichetele “+” respectiv “-”. La atingerea unei situații dorite, se acționează butonul de comandă “Înregistrează” care înregistrează o linie de program cu pozițiile momentane a tuturor c.c.c. sub forma : $M, \text{ nume situare, pozCCC1, pozCCC2, } \dots, \text{ pozCCCn}$ unde n reprezintă numărul de c.c.c. ale robotului modelat. Numele pentru situație este cerut de aplicație doar dacă este bifată caseta de validare “nume pos.”. Acționarea butonului de comandă “Close Gripper” realizează prehensarea obiectului de lucru virtual cu ajutorul bacurilor mecanismului de prehensiune. Acesta se va deplasa în continuare solidar cu efectorul final până la acționarea butonului de comandă “Open Gripper”. Robotul poate fi adus în poziția de acasă prin acționarea butonului de comandă “Acasă”. În orice moment se poate vizualiza matricea de situație (“Situație efector”) sau o matrice de trecere corespunzătoare c.c.c. sau offset-ului (“Matrice situație pt. SR”).

7.2.3.3 Programare la nivelul c.c.c. în manieră textuală

Aplicația RobSim dispune de un editor simplu pentru elaborarea programelor în manieră textuală (fig. 7.2.8). Editorul permite crearea unui program nou, modificarea unui program realizat anterior prin instruire, salvarea programului, facilități de copiere sau ștergere a unor linii de cod, rularea pas cu pas a câte unei linii de program sau a întregului program. În figura 7.2.9 este prezentată o secvență din desfășurarea simulării unei sarcini de lucru. Facilitatea de interpolare la nivel de axa permite selectarea a două linii succesive de program și realizează o interpolare liniară cu un număr de incremente specificat în caseta de text. Dacă se notează cu n_{inc} numărul de incremente specificat atunci din cele 2 linii de program inițiale rezultă $n_{inc} + 1$ linii de program.

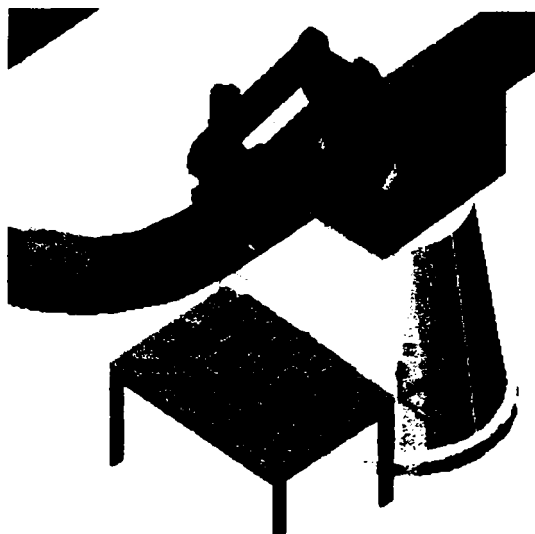


Fig. 7.2.9. Secvență din desfășurarea simulării unei operații de paletizare cu robotul SCORBOT ER III

7.2.3.4 Programare de nivel cartezian prin instruire

Programarea la nivel cartezian prin instruire are o eficiență sporită datorită rulării aplicației într-un mediu CAD deoarece datele necesare pentru specificarea situațiilor efectorului final pot fi obținute direct din mediul CAD prin selectarea directă a entităților geometrice. De exemplu, dacă efectorul final trebuie să deplaseze un știft dintr-o anumită poziție pentru a intra într-un alezaj, poziția punctului caracteristic și orientarea drepte caracteristice necesară poate fi obținută în AutoCAD prin selectarea liniei care materializează axa alezajului.

Programarea la nivel cartezian este implementată actual în aplicația RobSim numai pentru robotul SCORBOT ER III pentru care a fost implementat calculul cinematic invers. Pentru programarea la nivel cartezian prin instruire se utilizează caseta de dialog prezentată în figura 7.2.10. La apelarea programării prin această metodă se cere utilizatorului indicarea numelui listei de poziții în care se vor salva situațiile efectorului final și numele fișierului în care se stochează programul realizat prin instruire.

Fig. 7.2.10 Programarea la nivel cartezian prin instruire

O anumită situație se poate indica atât relativ la situația curentă cât și absolut (față de sistemul de referință fix atașat robotului). Pentru deplasarea de la situația curentă la cea comandată se poate specifica viteza de deplasare. Robotul SCORBOT are 10 trepte de viteză, treapta 0 corespunzând deplasării celei mai rapide și treapta 1 celei mai reduse. Între treapta 1 și 9 viteza de deplasare este crescătoare. În momentul în

care se ajunge la o situatie necesară pentru îndeplinirea sarcinii de lucru a robotului se comandă înregistrarea situații respective sub un nume indicat de utilizator. Situația se salvează în lista de poziții curentă indicată la apelarea programării prin instruire la nivel cartezian iar în program situația este referită prin indicele din lista de poziții. Linia de program care comandă deplasarea efectorului final al robotului în poziția cu indicele 9 din lista de poziții curente cu viteza 3 are forma :

GO POSITION 9 SPEED 3

O facilitate importantă oferită de aplicație este posibilitatea de generare a pozițiilor punctului caracteristic pe traiectorie prin divizarea cu pas constant a unei polilinii 3D selectate de utilizator în mediul CAD. În urma acționării butonului de comandă "Traiectorie" utilizatorul este solicitat să specifice o valoare pentru pasul de divizare a entității care se va selecta sau pentru numărul de diviziuni a traiectoriei "materializată" prin linia selectată.

7.2.3.5 Programare în limbajul SCORBASE cu ajutorul unui editor

Limbajul SCORBASE este unul din cele 2 limbaje în care pot fi programați roboții de tipul SCORBOT ER III. Aplicația RobSim dispune de un editor pentru programarea în limbajul SCORBASE (fig. 7.2.11) și de un "compiler" care transformă instrucțiunile acestui limbaj în forma programului la nivelul cuplelor cinematice conducătoare.

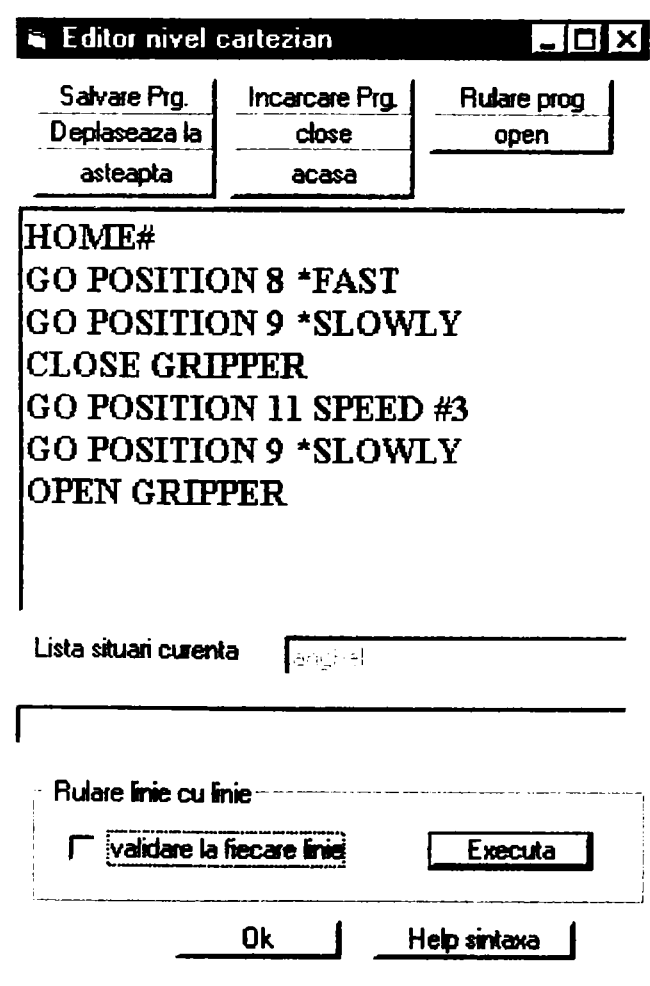


Fig. 7.2.11. Programarea în limbajul SCORBASE

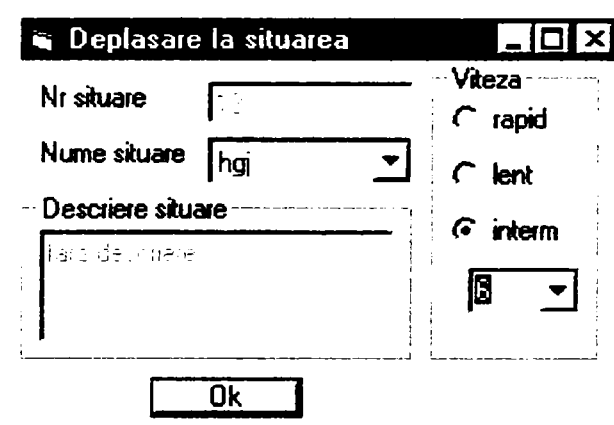


Fig. 7.2.12. Indicarea unei deplasări

Cu ajutorul butoanelor de comandă din partea superioară a formei din figura 7.2.11. se poate edita cu ușurință programul în limbaj SCORBASE fără erori de sintaxă. Pentru indicarea unei deplasări se utilizează o altă casetă de dialog care permite selectarea unei situații din lista de poziții și indicarea vitezei de deplasare (fig. 7.2.12). Pentru verificarea corectitudinii programului θ_i înainte de rularea pe robotul real se poate alege rularea programului în mediu virtual în regim instrucțiune cu instrucțiune. În acest caz, după executarea fiecărei linii de program se așteaptă validarea de către utilizator pentru executarea următoarei linii de program.

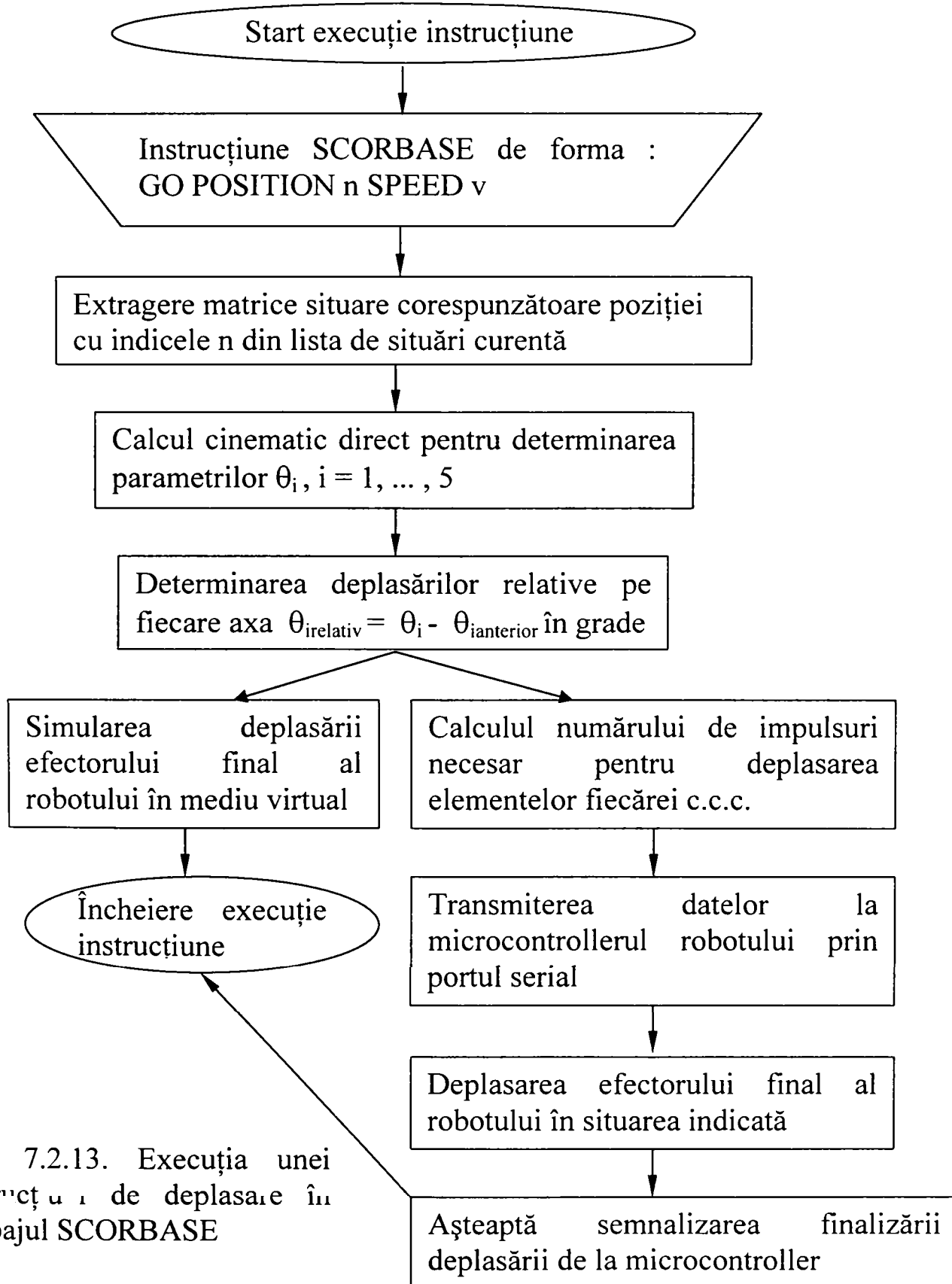


Fig. 7.2.13. Execuția unei instrucțiuni de deplasare în limbajul SCORBASE

Ca mod de implementare, la rularea unei instrucțiuni de deplasare în limbajul SCORBASE la o situație dată cu viteza specificată, pentru deplasarea efectorului final al robotului real se parcurg fazele descrise în organigrama din figura 7.2.13.

7.2.3.6 Urmărirea de către efectorul robotului a deplasării unei piese virtuale prehensate de către robotul virtual

Realizarea aplicației RobSim în mediul AutoCAD permite o bună interacțiune cu aplicația de proiectare asistată. Piese virtuale pot fi prehensate de către modelul robotului. O programare facilă se bazează în cazul în care deplasarea piesei prehensate în mediul AutoCAD generează automat programul la nivel cartezian și eventual realizarea sincronă a deplasărilor piesei reale de către robotul SCORBOT ER III. Pentru acesta se parcurg următoarele faze: deplasarea în mediul AutoCAD a modelului piesei, sesizarea în mediul CAD a evenimentului de modificare a situației entității geometrice care modelează piesa, determinarea situației efectorului final față de sistemul de referință atașat piesei urmată de aplicarea algoritmului descris la punctul 7.2.3.5.

7.2.3.7 Crearea unei liste de situații cu indicarea situațiilor prin diferite metode

Aplicația RobSim dispune de mai multe metode de indicare a situației efectorului final (fig. 7.2.14). Este posibilă crearea, editarea sau ștergerea unei liste de poziții și crearea, editarea sau ștergerea unei poziții din listă.

Fig. 7.2.14. Indicarea unei situații din lista de poziții

Indicarea unei situații a efectorului final se poate realiza ca matrice 4×4 , prin selectarea blocului care reprezintă sistemul de referință atașat efectorului final în AutoCAD sau prin indicarea situației cu ajutorul digitizorului 3D. Indicarea situației efectorului cu digitizorul 3D este prezentată pe larg în paragraful 7.2.7. Afișarea situației sistemului de referință atașat efectorului final este ca matrice 4×4 .

7.2.3.8 Probleme specifice aplicației de simulare a roboților în mediu virtual

Pentru simularea cât mai realistă a desfășurării sarcinii de lucru a fost necesară rezolvarea unor probleme specifice simulării în medii virtuale: afișarea unui anumit număr de situații intermediare între 2 situații comandate în cazul traiectoriilor punct cu punct, coordonarea imaginilor afișate ale mediului virtual cu mișcările reale a robotului în cazul în care prin intermediul RobSim se comandă un robot real, prehensarea obiectului de lucru virtual.

Deoarece frecvența maximă de afișare a cadrelor scenei de lucru în mișcare depinde de viteza procesorului și capacitățile plăcii video ale calculatorului pe care se face simularea, numărul de cadre afișate trebuie să varieze. În cazul în care viteza maximă de deplasare relativă corespunzătoare fiecărei c.c.c. este specificată, este posibilă calcularea timpului necesar pentru trecerea efectorului final dintr-o situație dată în situația următoare și implicit situația efectorului la un moment dat. Se afișează imaginea corespunzătoare momentului de timp la care se apelează funcția de afișare. Dacă capacitățile sistemului de calcul sunt mai modeste, numărul de cadre afișat este mai redus. O soluție rațională în acest caz este afișarea de tip cadru de sârmă, mai puțin consumatoare de resurse. Dacă vitezele nu sunt specificate, numărul de cadre afișate este dat de o constantă propusă de aplicație dar modificabilă de utilizator.

Coordonarea imaginilor afișate cu mișcările robotului real este realizată la încheierea fiecărei deplasări comandate din calculatorul PC prin aplicația RobSim, prin preluarea feed-back-ului de la robotul real. Prehensarea obiectului virtual se bazează pe testarea intersecției dintre bacurile mecanismului de prehensiune și obiectele din spațiul de lucru al robotului virtual.

7.2.4. Interfața robotului virtual cu robotul real

Aplicațiile de programare a roboților cu ajutorul digitizoarelor 3D sunt exemplificate în lucrare cu ajutorul robotului SCORBOT ER III. A fost ales acest tip de robot deoarece există în dotarea departamentului Mecatronică din cadrul Facultății de Mecanică al Universității "Politehnica" Timișoara unde autorul elaborează teza. Digitizorul 3D utilizat este un digitizor 3D cu structură RRR realizat de autor. Implementarea modelului geometric invers este realizată în aplicația RobSim pentru robotul SCORBOT ER III. Ca bază pentru aplicațiile digitizoarelor 3D la programarea roboților se prezintă descrierea robotului SCORBOT ER III, modelului cinematic direct și invers de ordinul 0 al acestui robot.

a) Descrierea robotului SCORBOT ER III

Robotul SCORBOT ER III este un robot educațional cu 5 cuple cinematice conducătoare. Toate cuplele cinematice ale robotului sunt cuple cinematice de rotație. Denumirile cuplelor cinematice ale robotului sunt: 1 – BASE, 2 – SHOULDER, 3 – ELBOW, 4 – PITCH, 5 – ROLL.

Sistemul de operare al robotului permite programarea punct cu punct. Acționarea cuplelor cinematice conducătoare ale robotului este realizată prin motoare pas cu pas. Prima cuplă cinematică permite rotirea robotului în plan orizontal, cuplele SHOULDER, ELBOW și PITCH rotesc elementele robotului în plan vertical, iar a cincea cuplă cinematică permite rotirea efectorului final [Tașcău2001]. Limitele coordonatelor cuplelor cinematice sunt: rotația de bază: 310° ; shoulder: $+130/-35^\circ$; elbow: $\pm 130^\circ$; pitch: $\pm 130^\circ$. În fig. 7.2.15. este prezentată o imagine de perspectivă a robotului.

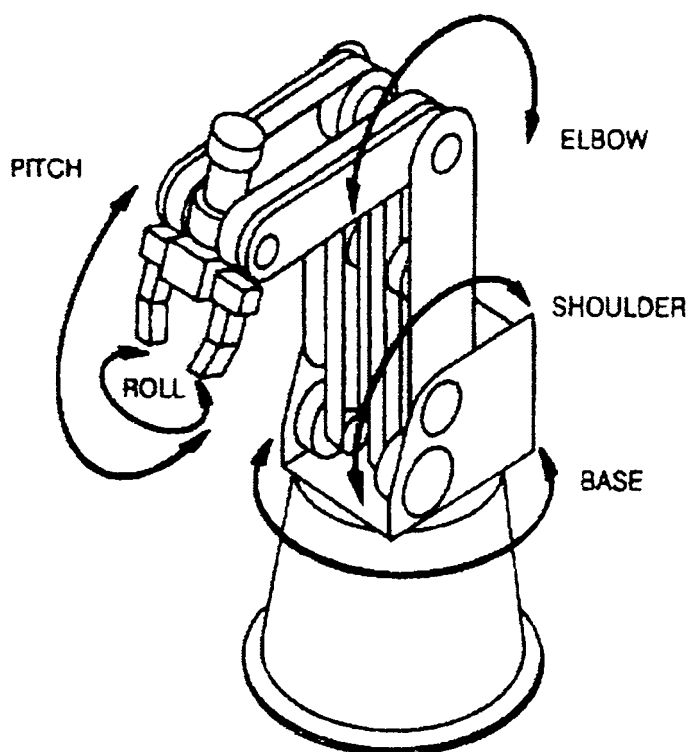


Fig. 7.2.15. Robotul SCORBOT ER III

Una din facilitățile acestui tip de robot este posibilitatea comenzii la nivel de axă prin transmiterea informațiilor la microcontrollerul robotului de la un calculator PC. Această facilitate a fost exploatată pentru comanda robotului prin intermediul aplicației RobSim. Transmiterea informațiilor de la PC către microcontroller și a celor de feed-back de la microcontroller la PC se realizează prin intermediul interfeței seriale RS232. Aplicația care implementează interfața efectivă între aplicația RobSim și robotul virtual este realizată de către drd. ing. Ciupe Valentin din același departament cu autorul.

b) Modelul geometric direct de ordinul 0

Modelul geometric direct exprimă situarea efectorului final în raport cu sistemul de referință fix în funcție de situarea relativă a elementelor fiecărei cuple cinematice conducătoare. Modelul geometric direct este exprimat prin intermediul matricii de situare notată ${}^R T_{EF}$.

$${}^R T_{EF} = \begin{bmatrix} n_x & o_x & a_x & p_x \\ n_y & o_y & a_y & p_y \\ n_z & o_z & a_z & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$n_x = \cos\theta_1 \cdot \cos\theta_5 \cdot \cos(\theta_2 + \theta_3 + \theta_4) - \sin\theta_1 \cdot \sin\theta_5 \tag{7.2.1}$$

$$n_y = \sin\theta_1 \cdot \cos\theta_5 \cdot \cos(\theta_2 + \theta_3 + \theta_4) + \cos\theta_1 \cdot \sin\theta_5 \tag{7.2.2}$$

$$n_z = \cos\theta_5 \cdot \sin(\theta_2 + \theta_3 - \theta_4) \tag{7.2.3}$$

$$o_x = -\cos\theta_1 \cdot \sin\theta_5 \cdot \cos(\theta_2 + \theta_3 + \theta_4) + \sin\theta_1 \cdot \cos\theta_5 \tag{7.2.4}$$

$$o_y = -\sin\theta_1 \cdot \sin\theta_5 \cdot \cos(\theta_2 + \theta_3 + \theta_4) + \cos\theta_1 \cdot \cos\theta_5 \tag{7.2.5}$$

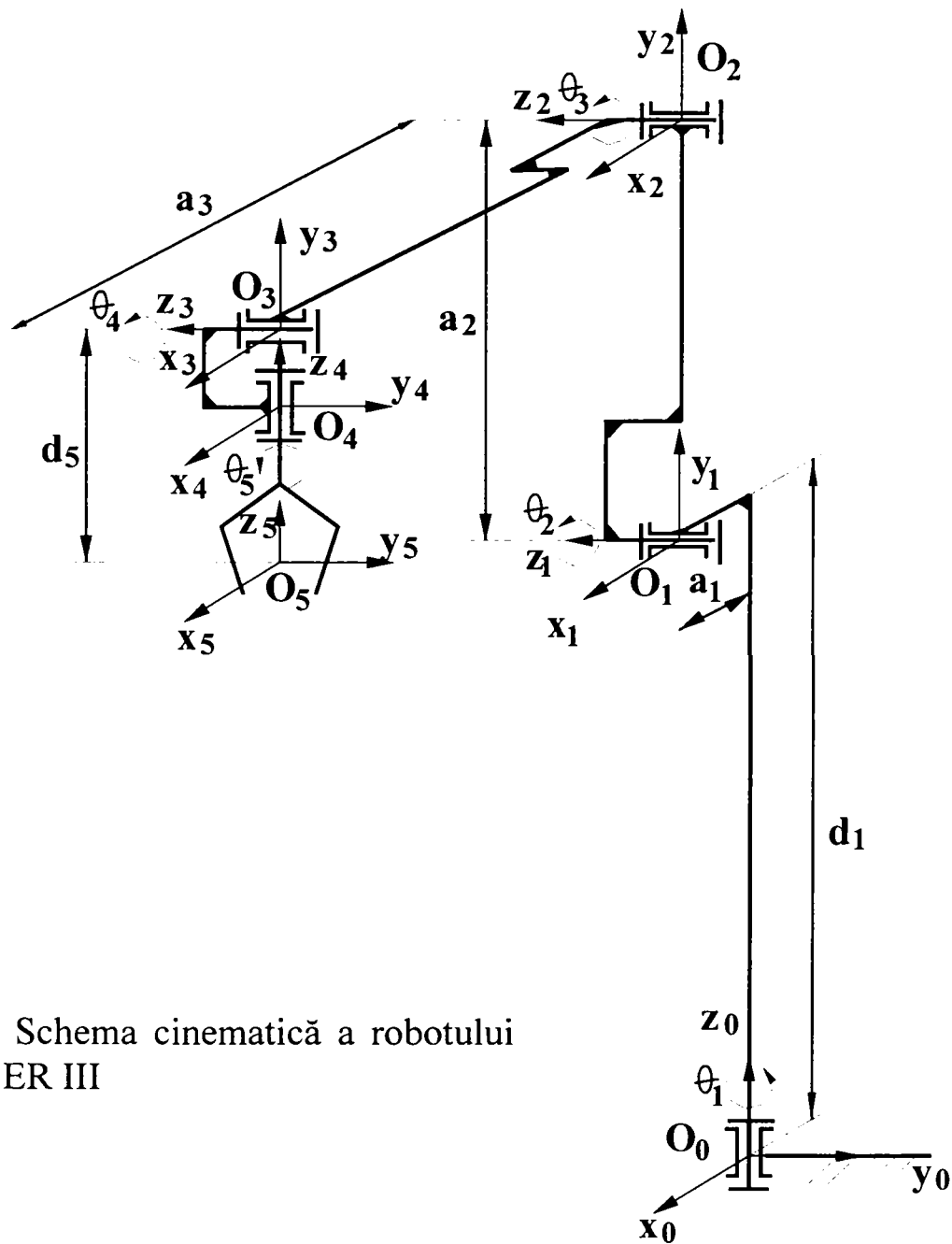


Fig. 7.2.16. Schema cinematică a robotului SCORBOT ER III

$$o_z = \sin \theta_5 \cdot \sin(\theta_2 + \theta_3 - \theta_4) \quad (7.2.6)$$

$$a_x = -\cos \theta_1 \cdot \sin(\theta_2 + \theta_3 + \theta_4) \quad (7.2.7)$$

$$a_y = \sin \theta_1 \cdot \sin(\theta_2 + \theta_3 - \theta_4) \quad (7.2.8)$$

$$a_z = \cos(\theta_2 + \theta_3 + \theta_4) \quad (7.2.9)$$

$$p_x = \cos \theta_1 [d_5 \cdot \sin(\theta_2 + \theta_3 + \theta_4) + a_3 \cdot \cos(\theta_2 + \theta_3) - a_2 \cdot \sin \theta_2 + a_1] \quad (7.2.10)$$

$$p_y = -\sin \theta_1 [d_5 \cdot \sin(\theta_2 + \theta_3 - \theta_4) + a_3 \cdot \cos(\theta_2 + \theta_3) + a_2 \cdot \sin \theta_2 - a_1] \quad (7.2.11)$$

$$p_z = -d_5 \cdot \cos(\theta_2 + \theta_3 - \theta_4) + a_3 \cdot \sin(\theta_2 + \theta_3) + a_2 \cdot \cos \theta_2 + d_1 \quad (7.2.12)$$

c) Modelul geometric invers

În cazul calcului cinematic invers de ordinul 0 se cunosc dimensiunile robotului, distanțele offset și situarea efectorului final și se determină situațiile relative ale elementelor cuplelor cinematice conducătoare exprimate prin unghiurile $\theta_1, \theta_2, \theta_3, \theta_4, \theta_5$. În continuare se prezintă determinarea de către autor a modelului geometric invers. Pentru determinarea modelului geometric invers se utilizează o metodă “geometrică”, considerată de autor mai intuitivă.

Din expresiile (7.2.10) și (7.2.11), în funcție de cadran, rezultă valoarea pentru θ_1 în funcție de cadrantul căruia îi aparține (figura 7.2.17):

$$\left\{ \begin{array}{l} - p_x > 0 \quad p_y > 0 \Rightarrow \theta_1 = \arctg\left(\frac{p_y}{p_x}\right) \\ - p_y > 0 \quad p_x < 0 \Rightarrow \theta_1 = \arctg\left(\frac{p_y}{p_x}\right) \\ - p_y < 0 \quad p_x < 0 \Rightarrow \theta_1 = \arctg\left(\frac{p_y}{p_x}\right) + \pi \\ - p_y < 0 \quad p_x > 0 \Rightarrow \theta_1 = \arctg\left(\frac{p_y}{p_x}\right) + 2 * \pi. \end{array} \right. \quad (7.2.13.)$$

Pentru cazurile în care $p_x = 0$:

$$- p_y > 0 \quad p_x = 0 \Rightarrow \theta_1 = \frac{\pi}{2}$$

$$- p_y < 0 \quad p_x = 0 \Rightarrow \theta_1 = \frac{3 * \pi}{2}$$

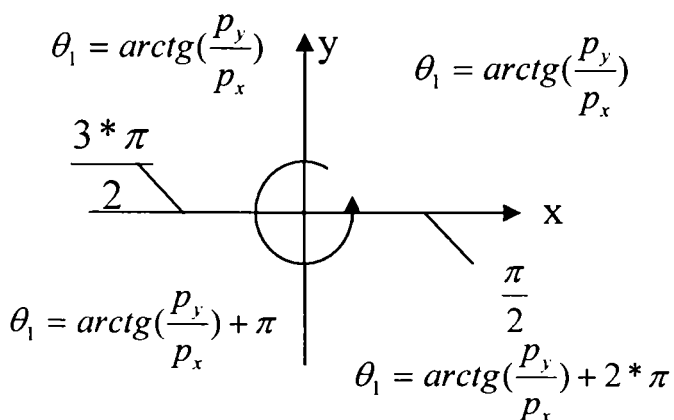


Fig. 7.2.17. Determinarea valorii unghiului θ_1 .

Se notează cu SR_0 sistemul de referință atașat elementului fix al robotului. Se consideră un plan (p) cu următoarele proprietăți: (p) este perpendicular pe planul $x_0O_0y_0$ al sistemului de referință SR_0 , punctul $O_0 \in (p)$ și (p) formează un unghi diedru cu valoarea θ_1 cu planul $x_0O_0z_0$. O secțiune prin schema cinematică a dispozitivului de ghidare al robotului cu planul (p) este ilustrată în figura 7.2.18.

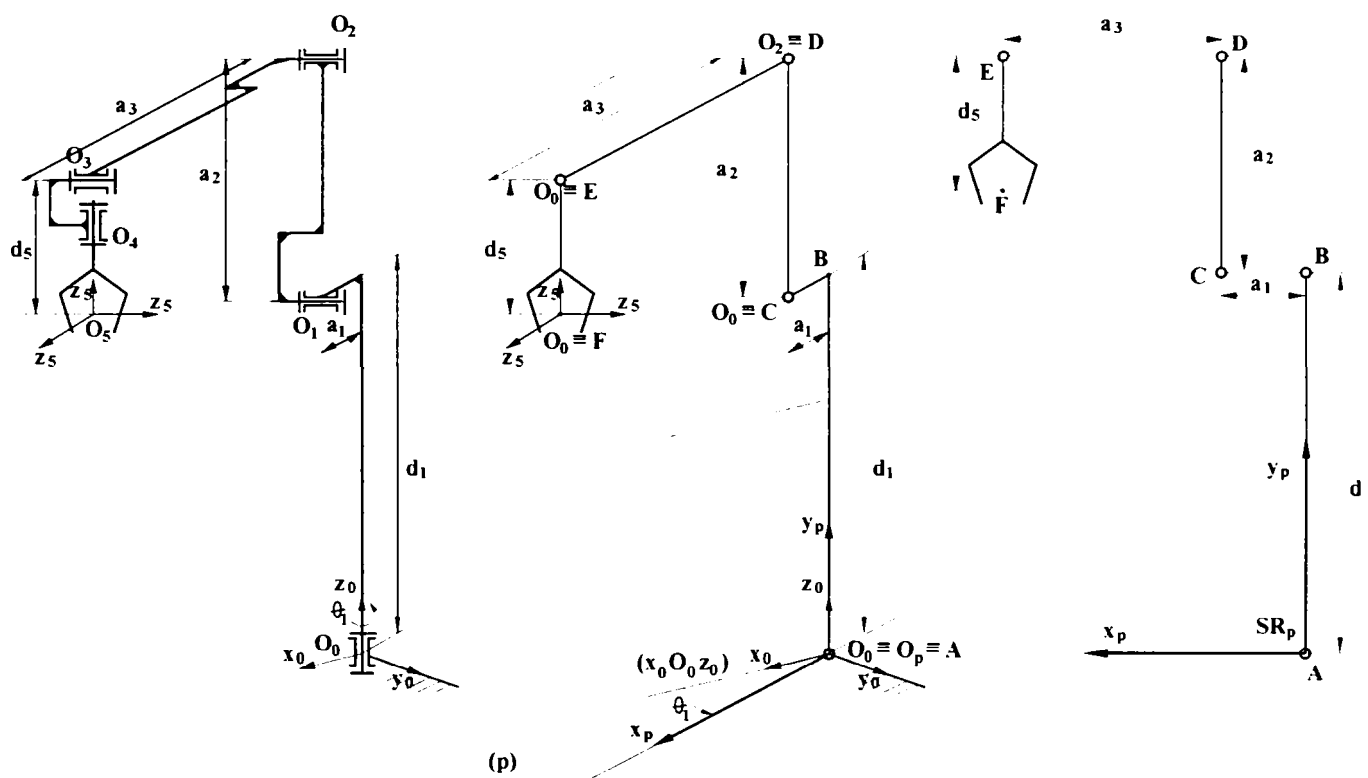


Fig. 7.2.18. Notații utilizate la determinarea modelului geometric invers al robotului

- Se consideră un sistem de referință SR_p cu originea în O_p cu proprietățile:
- axa x_p a sistemului de referință se obține prin aplicarea unei rotiri cu unghiul θ_1 a axei x_0 a sistemului de referință SR_0 în jurul axei z_0 a sistemului de referință SR_0 ;
 - punctul O_p este identic cu originea sistemului de referință SR_0 ;
 - axa y_p a sistemului de referință SR_p este identică cu axa z_0 a sistemului de referință SR_0 .

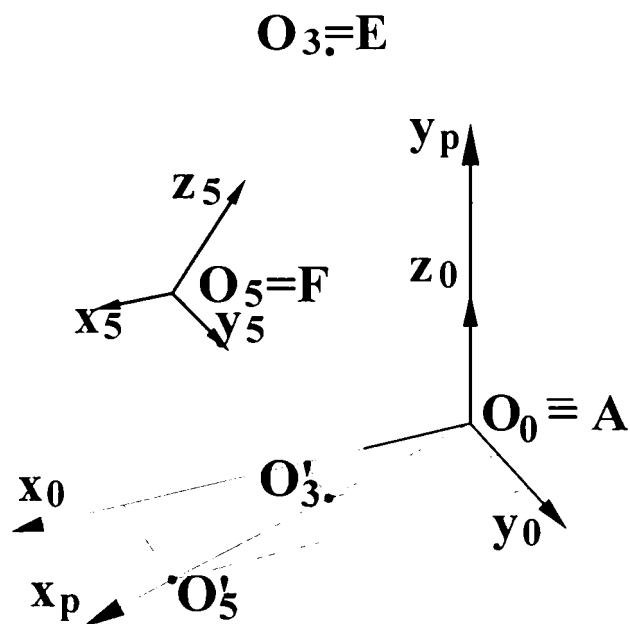
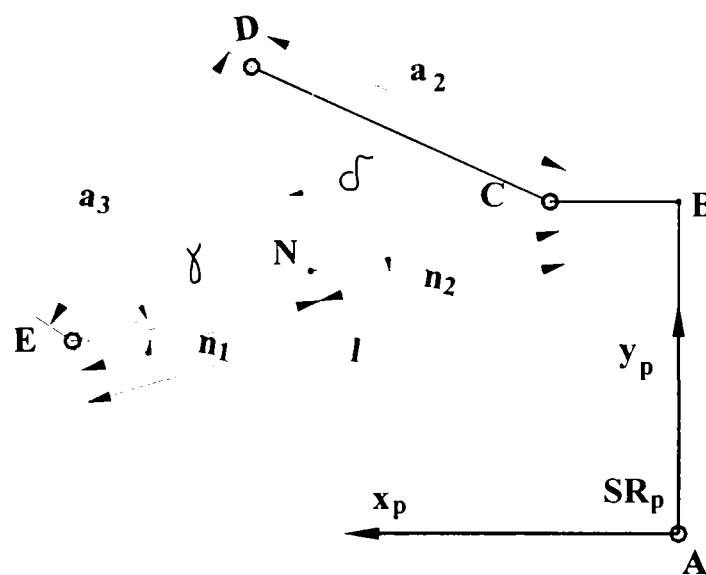
În continuarea acestui paragraf, în cazul în care nu se specifică explicit alt sistem de referință, coordonatele punctelor se consideră în sistemul de referință SR_p . S-au notat cu A, C, D, E, F corespondentele în sistemul SR_p ale punctelor O_0, O_1, O_2, O_3, O_5 din sistemul de referință SR_0 . Din figura 7.2.17 se observă coordonatele punctelor A, B și C: $A(0,0), B(0,d_1), C(a_1,d_1)$. (7.2.14.)

În figura 7.2.19. se ilustrează determinarea coordonatelor punctului O_3 în sistemul de referință SR_0 .

Coordonatele punctului O_3 sunt:

$$\begin{cases} x_3 = x_5 + d_5 \cdot a_x \\ y_3 = y_5 + d_5 \cdot a_y \\ z_3 = z_5 + d_5 \cdot a_z \end{cases} \quad (7.2.15.)$$

În figura (7.2.19) s-a notat cu O_5' proiecția punctului O_5 pe planul $x_0O_0y_0$ a sistemului de referință SR_0 , și cu O_3' proiecția punctului O_3 pe același plan.


 Fig. 7.2.19 Determinarea coordonatelor punctului O_3

 Fig. 7.2.20. Determinarea coordonatelor punctului D

$$O_0 O_3' = \sqrt{x_3^2 + y_3^2} \quad (7.2.16.)$$

$$O_5 O_5' = \sqrt{x_5^2 + y_5^2} \quad (7.2.17.)$$

Coordonatele în sistemul de referință SR_p ale punctelor E și F sunt:

$$x_E = \sqrt{x_3^2 + y_3^2}; \quad y_E = z_3 \quad (7.2.18.)$$

$$x_F = \sqrt{x_5^2 + y_5^2}; \quad y_F = z_5 \quad (7.2.19.)$$

Se determină coordonatele punctului D . Punctul D reprezintă intersecția dintre cercurile de centre $C(x_C, y_C)$, $E(x_E, y_E)$ și raze a_2 respectiv a_3 (fig. 7.2.20).

Se notează cu h lungimea segmentului DN (unde $DN \perp CE$), cu n_1 lungimea segmentului EN și cu n_2 lungimea segmentului CN .

În triunghiurile dreptunghice DNE și CND sunt îndeplinite relațiile :

$$h^2 = a_3^2 - n_1^2 \quad (7.2.20.)$$

$$h^2 = a_2^2 - (l - n_1)^2 \quad (7.2.21.)$$

Din (7.2.20) și (7.2.21) se obține:

$$a_3^2 - n_1^2 = a_2^2 - (l - n_1)^2 \quad (7.2.22.)$$

$$n_1 = \frac{a_3^2 - a_2^2 + l^2}{2 \cdot l} \quad (7.2.23.)$$

$$\text{Se notează: } \lambda = \frac{n_1}{n_2} = \frac{n_1}{l - n_1} \quad (7.2.24.)$$

Coordonatele punctului N sunt:

$$x_N = \frac{x_E + \lambda \cdot x_C}{1 + \lambda} \quad y_N = \frac{y_E + \lambda \cdot y_C}{1 + \lambda} \quad (7.2.25.)$$

$$\gamma = \arctg\left(\frac{y_C - y_E}{x_C - x_E}\right) \quad (7.2.26.)$$

$$\delta = \gamma + \frac{\pi}{2} \tag{7.2.27}$$

Punctul D este situat la distanța h de punctul N pe direcția unei drepte care face unghiul δ cu direcția EC :

$$\begin{cases} x_D = x_N + h \cdot \cos \delta \\ y_D = y_N + h \cdot \sin \delta \end{cases} \tag{7.2.28}$$

Algoritmul de calcul prezentat anterior este utilizat și în cadrul unei aplicații de teleoperare prin Internet a robotului SCORBOT ER III care urmează a fi finalizată în cadrul unui grant la care autorul este membru al colectivului de cercetare.

Indicarea poziției punctului caracteristic al robotului este realizată prin intermediul a două ferestre (fig.7.2.21). În cele 2 ferestre, există o corespondență între coordonatele reale și numărul de pixeli pe fiecare direcție. Prin intermediul primei ferestre se comandă variația unghiurilor $\theta_2, \theta_3, \theta_4$. Cursorul mouse-ului este adus în poziția în care se suprapune cu punctul E reprezentând în acesastă vedere punctul caracteristic al robotului, este acționat butonul stâng al mouse-ului, se deplasează cursorul într-o nouă poziție apoi este eliberat butonul stâng al mouse-ului. În timpul cât este acționat butonul stâng al mouse-ului, reprezentarea punctului caracteristic al robotului urmărește cursorul mouse-ului, iar deplasarea relativă a elementelor este reprezentată pe baza coordonatelor punctelor C, D, E, F. Variația unghiului θ_1 este comandată în mod similar prin intermediul celei de-a doua ferestre reprezentată în figura 7.2.21.. Variația unghiului θ_5 este comandată prin intermediul unui cursor. În momentul în care operatorul validează deplasarea efectorului final al robotului real, se realizează calculul geometric invers și se transmit comenzile robotului.

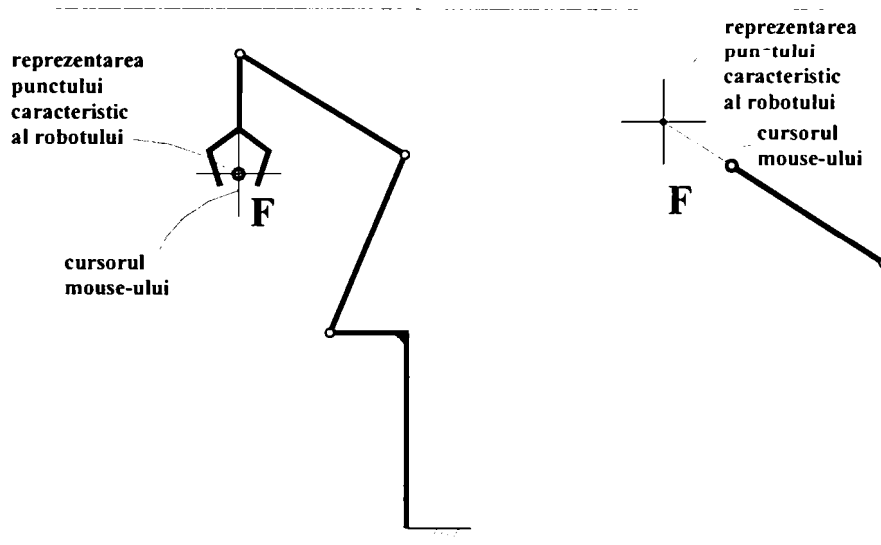


Fig. 7.2.21 Reprezentarea ferestrelor pentru indicarea situației efectorului final în cazul aplicației de teleoperare

Se determină în continuare unghiurile $\theta_2, \theta_3, \theta_4$.

$$\theta_2 = \arctg\left(\frac{x_C - x_D}{|y_C - y_D|}\right) \tag{7.2.29.}$$

Discuția în cazul acestui unghi se face pentru cazurile în care se anulează $|y_C - y_D|$ împărțitorul. Astfel:

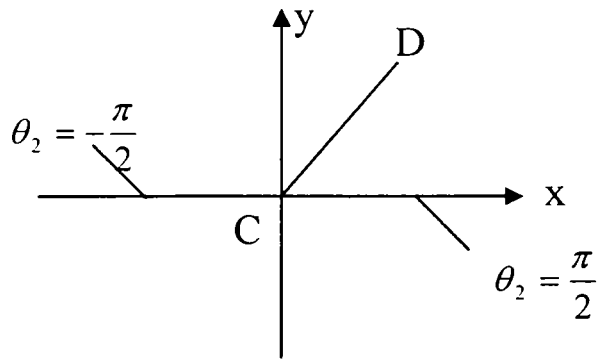


Fig. 7.2.22. Determinarea valorii unghiului θ_2 .

$$- y_D - y_C = 0 \quad x_C < x_D \Rightarrow \theta_2 = -\frac{\pi}{2}$$

$$- y_D - y_C = 0 \quad x_C > x_D \Rightarrow \theta_2 = \frac{\pi}{2}$$

Se notează cu l lungimea segmentului CE și cu α unghiul format de segmentele CD și DE (Fig.7.2.23.).

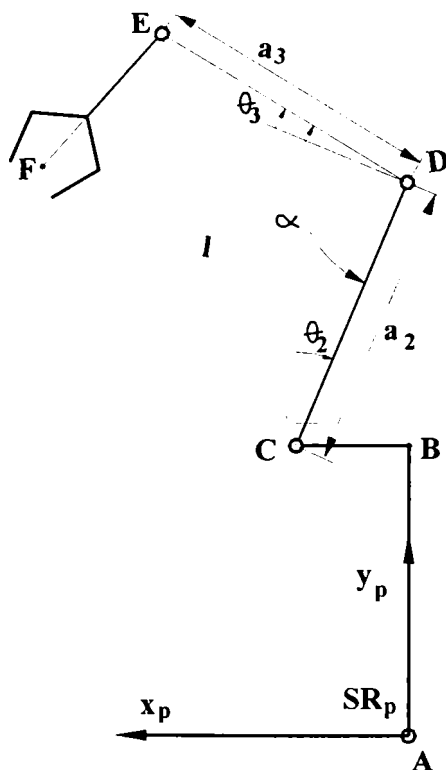


Fig. 7.2.23. Determinarea unghiului θ_3

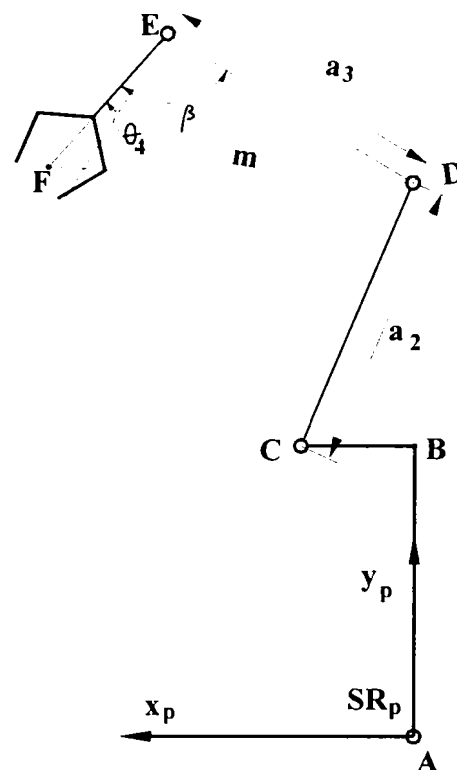


Fig. 7.2.24. Determinarea unghiului θ_4

Se aplică teorema lui Pitagora generalizată în triunghiul CDE :

$$l^2 = a_2^2 + a_3^2 + 2 \cdot \text{sgn}_\alpha \cdot a_2 \cdot a_3 \cdot \cos \alpha \quad (7.2.30)$$

$$\operatorname{sgn}_\alpha = \begin{cases} -1 & \text{daca } l^2 < a_2^2 + a_3^2 \\ 0 & \text{daca } l^2 = a_2^2 + a_3^2 \\ 1 & \text{daca } l^2 > a_2^2 + a_3^2 \end{cases} \quad (7.2.31.)$$

$$\cos \alpha = \frac{l^2 - a_2^2 - a_3^2}{2 \cdot a_2 \cdot a_3 \cdot \operatorname{sgn}_\alpha} \quad (7.2.32.)$$

Determinarea lui α se va face în funcție de intervalul valoric în care se găsește unghiul. Astfel, dacă $\theta_3 \in (-\frac{\pi}{2}, \frac{\pi}{2}) \Rightarrow \alpha \in (0, \pi)$.

Apartenența $\theta_3 \in (-\frac{\pi}{2}, 0)$ e îndeplinită atunci cand $\alpha \in (0, \frac{\pi}{2})$, condiție care este îndeplinită în cazul în care triunghiul determinat de punctele C, D și E este ascuțit, ceea ce este echivalent cu $\operatorname{sgn}_\alpha = -1$. Formula valorică a lui α va fi:

$$\alpha = \operatorname{arctg}\left(\frac{\sqrt{1 - \cos^2 \alpha}}{\cos \alpha}\right) \quad (7.2.33.)$$

Cel de-al doilea caz, $\theta_3 \in (0, \frac{\pi}{2})$ este îndeplinită dacă $\alpha \in (\frac{\pi}{2}, 0)$, echivalent cu condiționarea ca triunghiul CDE să fie obtuz, $\operatorname{sgn}_\alpha = 1$, iar valoarea lui α este:

$$\alpha = \frac{\pi}{2} + \operatorname{arctg}\left(\frac{\sqrt{1 - \cos^2 \alpha}}{\cos \alpha}\right) \quad (7.2.34.)$$

După determinarea lui α se va determina valoarea unghiului căutat, θ_3 cu formula:

$$\theta_3 = \alpha - \frac{\pi}{2} \quad (7.2.35.)$$

Se notează cu m lungimea segmentului DF și cu β unghiul DEF (Fig.7.2.24.). Se aplică teorema lui Pitagora generalizată în triunghiul DEF:

$$m^2 = a_3^2 + d_5^2 + 2 \cdot \operatorname{sgn}_\beta \cdot a_3 \cdot d_5 \cdot \cos \beta \quad (7.2.36.)$$

$$\operatorname{sgn}_\beta = \begin{cases} -1 & \text{daca } m^2 < a_3^2 + d_5^2 \\ 0 & \text{daca } m^2 = a_3^2 + d_5^2 \\ 1 & \text{daca } m^2 > a_3^2 + d_5^2 \end{cases} \quad (7.2.37.)$$

$$\cos \beta = \frac{m^2 - a_3^2 - d_5^2}{2 \cdot a_3 \cdot d_5 \cdot \operatorname{sgn}_\beta} \quad (7.2.38.)$$

$$\operatorname{tg} \beta = \frac{\sqrt{1 - \cos^2 \beta}}{\cos \beta} \quad (7.2.39.)$$

Determinarea lui β se va face în funcție de intervalul valoric în care se găsește unghiul. Astfel, dacă $\theta_4 \in (-\frac{\pi}{2}, \frac{\pi}{2}) \Rightarrow \beta \in (0, \pi)$.

Apartenența $\theta_4 \in (-\frac{\pi}{2}, 0)$ e îndeplinită atunci cand $\beta \in (0, \frac{\pi}{2})$, condiție care este îndeplinită în cazul în care triunghiul determinat de punctele D, E și F este ascuțit, ceea ce este echivalent cu $\text{sgn}_\alpha = -1$. Formula valorică a lui β va fi:

$$\beta = \text{arctg}\left(\frac{\sqrt{1 - \cos^2 \beta}}{\cos \beta}\right). \quad (7.2.40.)$$

Cel de-al doilea caz, $\theta_4 \in (0, \frac{\pi}{2})$ este îndeplinit dacă $\beta \in (\frac{\pi}{2}, 0)$, echivalent cu condiționarea ca triunghiul DEF să fie obtuz, $\text{sgn}_\alpha = 1$, iar valoarea lui β este:

$$\beta = \frac{\pi}{2} + \text{arctg}\left(\frac{\sqrt{1 - \cos^2 \beta}}{\cos \beta}\right) \quad (7.2.41.)$$

După determinarea lui β se va determina valoarea unghiului căutat, θ_4 cu formula:

$$\theta_4 = \beta - \frac{\pi}{2} \quad (7.2.42.)$$

Prin raportul dintre expresiile (7.2.1) și (7.2.4) rezultă:

$$\theta_5 = \text{arctg}\left(\frac{o_x \cdot \cos \theta_1 \cdot \cos(\theta_2 + \theta_3 + \theta_4) + n_x \cdot \sin \theta_1}{o_x \cdot \sin \theta_1 - n_x \cdot \cos \theta_1 \cdot \cos(\theta_2 + \theta_3 + \theta_4)}\right) \quad (7.2.43.)$$

7.2.5. Utilitatea interfațării RobSim cu digitizoare 3D

Prin studiu bibliografic autorul a ajuns la concluzia că aplicațiile soft existente pentru programarea off-line a roboților în medii virtuale nu dispun de interfață pentru digitizoare 3D. Acesta este unul din principalele motive care au dus la realizarea aplicației RobSim.

RobSim permite indicarea cu ajutorul digitizorului 3D a situărilor efectorului final al robotului în decursul sarcinii de îndeplinit, realizarea modelului geometric al mediului robotului pentru facilitarea programării acestuia în mediu virtual și simularea sarcinii de lucru, realizarea unor aplicații de teleoperare pe baza feed-back-ului vizual. Prin descrierea tuturor aplicațiilor enumerate anterior se dorește deschiderea orizontului pentru utilizarea digitizoarelor 3D în domeniul Roboticii.

7.2.6. Calibrarea robot-digitizor 3D

Pentru ca datele sub forma coordonatelor unor puncte individuale, șirurilor de puncte sau a unor matrici de situare să poată fi utilizate de către programele robotului este necesară realizarea unei calibrări robot-digitizor 3D. Calibrarea robot-digitizor constă în determinarea elementelor matricii de trecere de la sistemul de referință fix atașat digitizorului 3D SR_{dig} la sistemul de referință fix atașat robotului SE_{rob} (fig 7.2.25.).

Pentru realizarea calibrării robot-digitizor se palpează cu vârful digitizorului, notat cu V, 3 puncte de pe suprafața ultimului element al robotului. Pentru cele 3 puncte P_i , $i = 1, \dots, 3$ se cunosc coordonatele în raport cu sistemul de referință atașat

efectorului final. Determinarea coordonatelor acestor puncte în SR_{Ef} se realizează anterior calibrării robot-digitizor. Aceste puncte se consideră în elemente geometrice caracteristice (vârfuri ale unor muchii, vârfuri ale unor alezaje conice, etc.) și sunt marcate. Marcarea se poate face de exemplu prin realizarea unor puncte colorate pe suprafața elementului robotului.

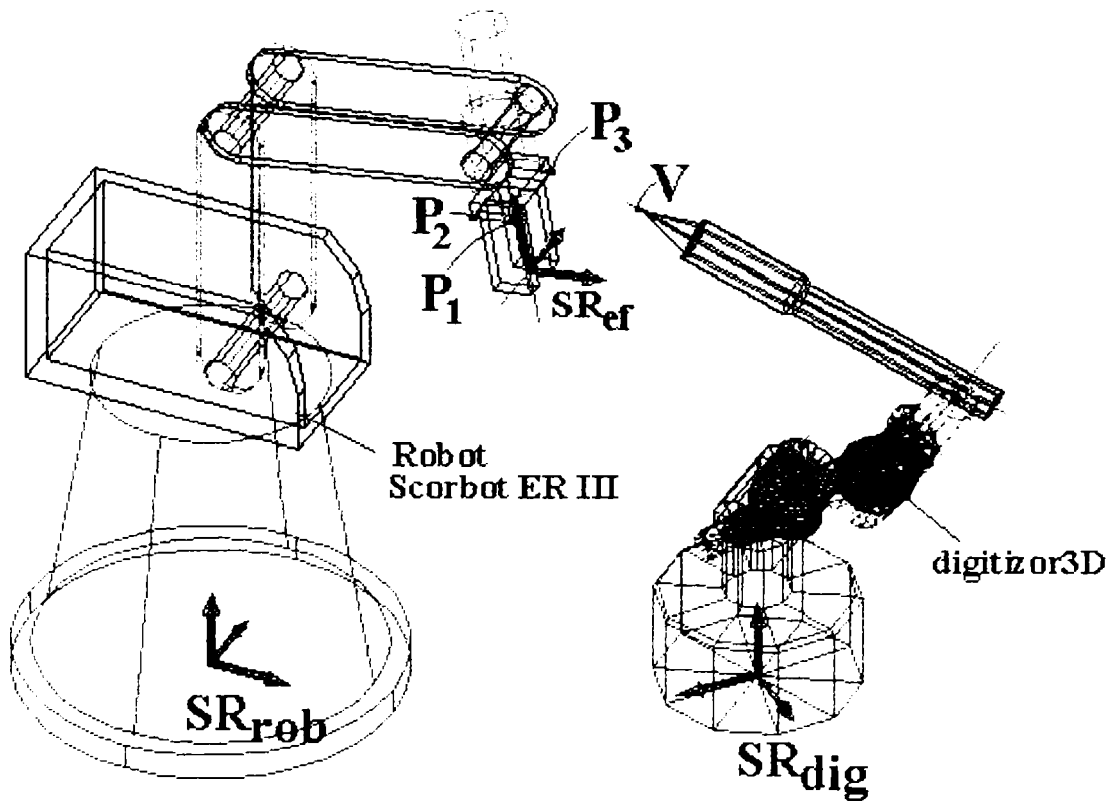


Fig. 7.2.25. Calibrarea robot – digitizor 3D

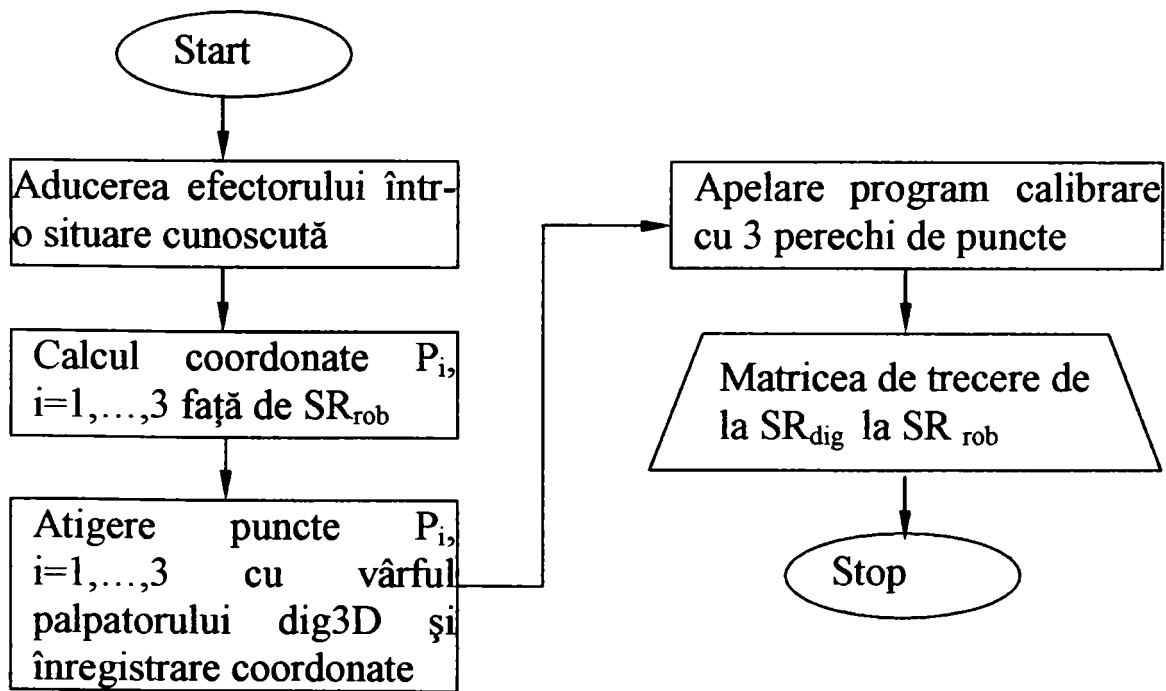


Fig. 7.2.26. Algoritm pentru calibrarea robot – digitizor 3D

Ca mod de realizare a calibrării se parcurg următorii pași: aducerea efectorului final al robotului într-o situație comandată, calculul coordonatelor punctelor P_1 , P_2 și P_3 față de sistemul de referință fix atașat robotului, atingerea succesivă a acestor puncte de pe suprafața elementului robotului cu digitizorul 3D și înregistrarea coordonatelor măsurate cu digitizorul 3D, utilizarea unui program pentru determinarea matricii de trecere de la SR_{dig} la SR_{rob} . Organigrama metodei este prezentată în fig. 7.2.26..

Aducerea efectorului final într-o situație comandată se poate realiza prin instruire. Această situație poate să corespundă poziției de acasă a robotului. Coordonatele punctelor P_i , $i=1, \dots, 3$ în raport cu sistemul de referință fix atașat efectorului final se cunosc, fiind determinate în momentul marcării punctelor pe suprafața ultimului element al robotului. Deoarece situația efectorului final este comandată de către operator, se cunoaște matricea de situație a efectorului final deci prin aplicarea acestei matrici se determină coordonatele punctelor P_i față de sistemul de referință fix atașat robotului. Se notează aceste puncte cu P_{i1} , $i=1, \dots, 3$.

Se aduce vârful palpatorului digitizorului în contact cu fiecare din cele 3 puncte marcate pe suprafața elementului robotului. Pentru fiecare din aceste puncte notate P_{i2} , $i=1, \dots, 3$ se înregistrează coordonatele indicate de către digitizorul 3D.

Utilizând ca date de intrare coordonatele punctelor P_{i1} și P_{i2} , cu $i=1, \dots, 3$ se determină matricea de calibrare adică matricea de trecere de la sistemul de referință SR_{dig} la SR_{rob} cu ajutorul programului care implementează algoritmul descris la paragraful 5.3. Pentru ca robotul să poată utiliza situațiile indicate cu digitizorul 3D, asupra tuturor datelor preluate cu digitizorul se aplică matricea de calibrare.

7.2.7. Precizarea situațiilor efectorului final cu digitizorul 3D

Impunerea situației palpatorului digitizorului 3D prin acționarea manuală a ultimului element al digitizorului permite indicarea cu ușurință și într-un timp scurt a unor situații în spațiul de lucru al digitizorului. Se consideră digitizorul fixat într-o situație relativă față de robot astfel încât spațiile de lucru ale acestora să se suprapună în zona necesară pentru realizarea operațiilor impuse de sarcina de lucru a robotului. În acest caz este posibilă crearea unei liste de situații pentru efectorul final al robotului cu ajutorul digitizorului 3D. Pentru ca acest lucru să fie posibil este necesar ca în prealabil să fie determinată matricea de calibrare robot-digitizor (paragraf 7.2.5.2). Valorile elementelor matricii de calibrare sunt stocate într-un fișier al calculatorului și sunt utilizate pentru transformarea tuturor situațiilor indicate cu ajutorul digitizorului 3D în situații ale efectorului final al robotului.

Utilizarea acestei metode pentru completarea listei de situații în cazul în care numărul de situații distincte necesare este mare consumă timp mai redus față de varianta clasică de realizare a listei de poziții prin instruire. În cazul în care precizia de situație cerută pentru efectorul robotului este foarte mare, este posibil să apară necesitatea unor corecții suplimentare realizate la testarea programului obținut pornind de la situațiile înregistrate cu digitizorul 3D.

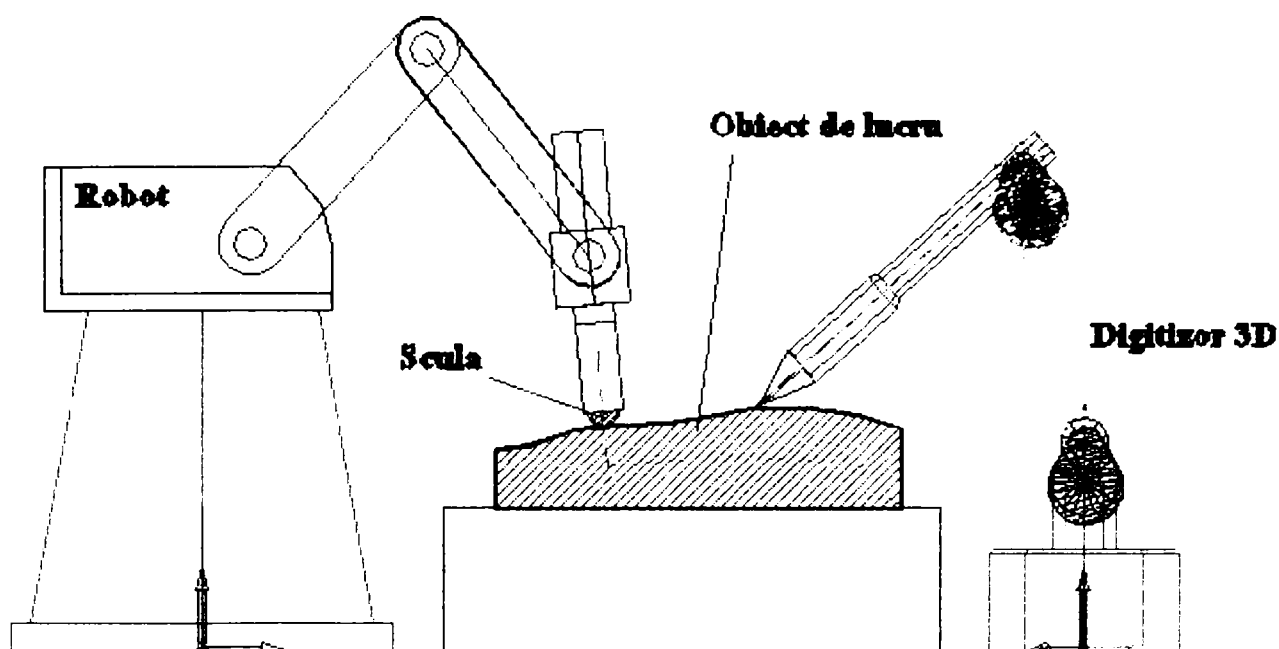


Fig. 7.2.27. Prelucrarea cu ajutorul robotului a unei suprafețe complexe a cărei formă a fost determinată cu ajutorul digitizorului 3D

Un avantaj mai evident al metodei apare în cazul în care efortul final al robotului este reprezentat de o sculă care trebuie să urmărească o traiectorie pe suprafața obiectului de lucru, iar suprafața are o formă complexă. În acest caz se precizează un increment pentru înregistrarea situațiilor cu digitizorul 3D. Se precizează diferența minimă a elementelor a două matrici de situație succesive astfel încât situațiile să fie înregistrate ca situații distincte. Pe baza datelor înregistrate se realizează programul cu puncte “via” (Fig. 7.2.27).

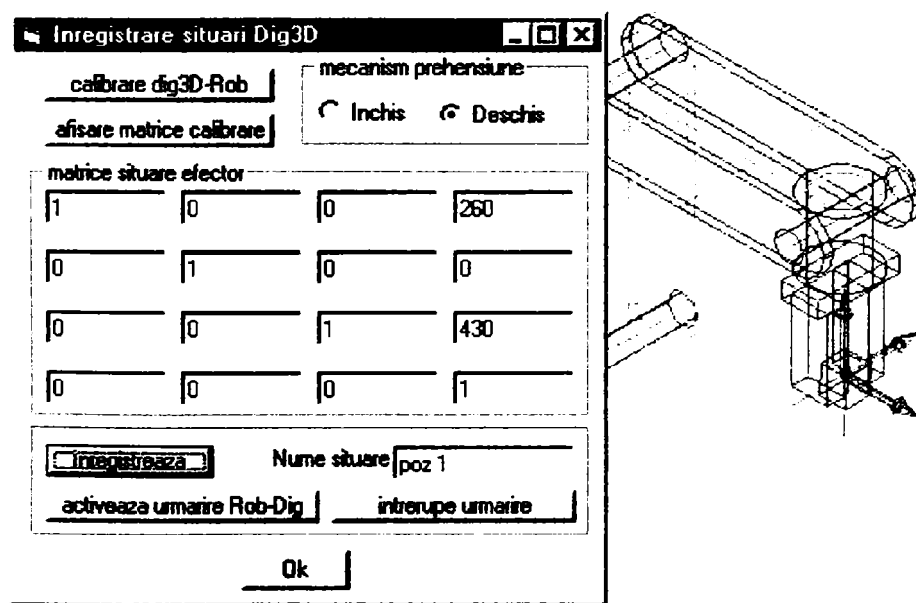


Fig. 7.2.28 Conducerea robotului virtual cu digitizorul 3D și înregistrarea situațiilor efectorului final al robotului

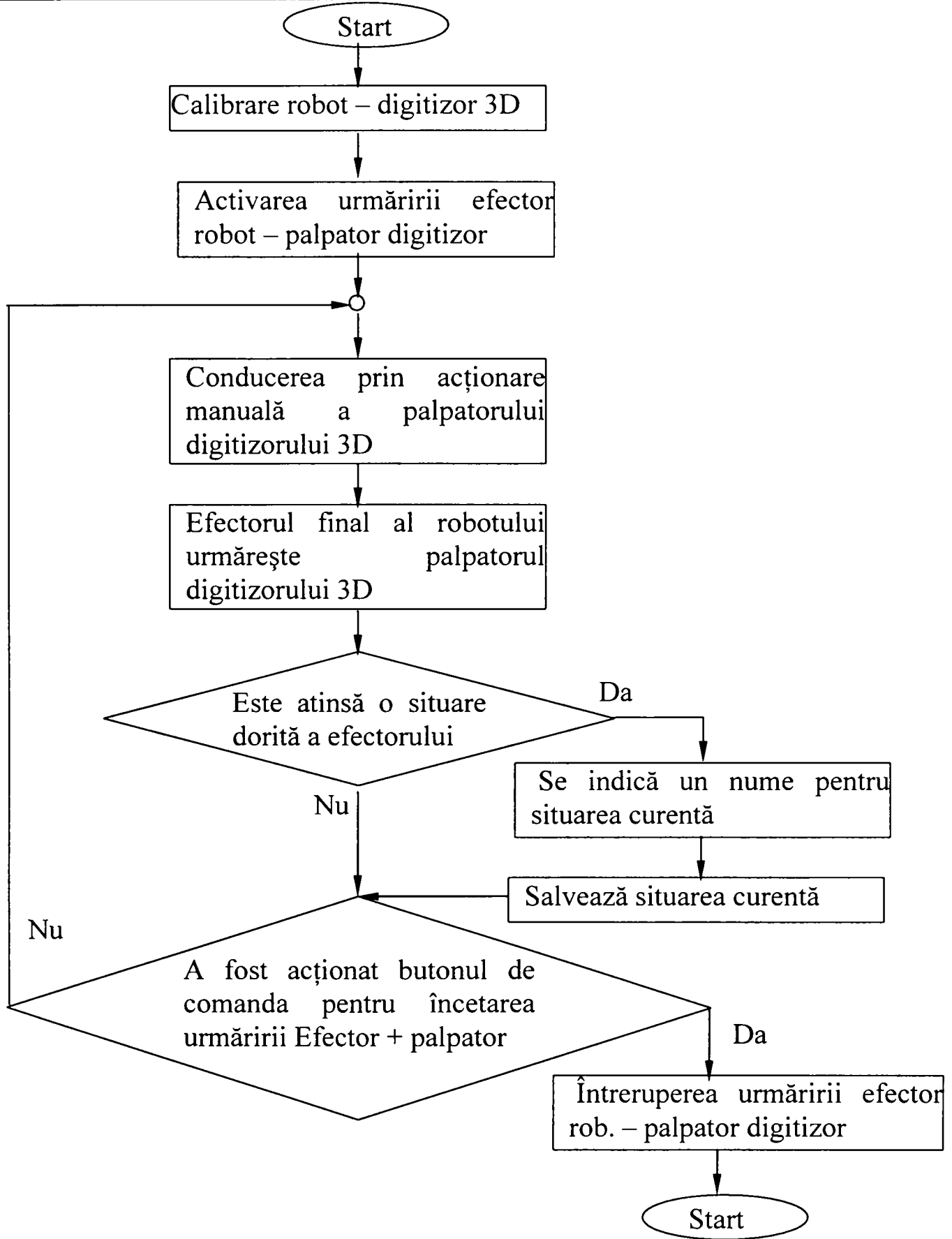


Fig. 7.2.29. Schema logică a algoritmului de formare a listei de poziții pentru robot cu ajutorul digitizorului 3D

Pe baza listei de poziții se poate realiza în manieră textuală programul pentru desfășurarea sarcinii robotului. În cazul în care există un model virtual de robot și este realizat modelul geometric al mediului acestuia, digitizorul 3D se poate utiliza

pentru programarea off-line a robotului. În urma unei prealabile calibrări digitizor – robot virtual, efectorul final al robotului poate fi adus în diferite situații ale spațiului de lucru din mediul virtual prin deplasarea palpatorului digitizorului. În momentul în care utilizatorul vizualizează pe ecranul calculatorului că o anumită situație a fost atinsă de către efectorul final se comandă înregistrarea acelei situații în lista de poziții.

În figura 7.2.28. se prezintă o secvență din programarea prin această metodă iar în figura 7.2.29. se prezintă schema logică care descrie metoda.

7.2.8. Realizarea modelului geometric al mediului robotului cu digitizorul 3D

Pentru programarea off-line a roboților în mediu virtual este necesar ca în prealabil să fie realizat modelul virtual al robotului și modelul geometric al mediului de lucru al acestuia. Prin mediul robotului se înțeleg dispozitivele, instalațiile și obiectele de lucru situate în spațiul de lucru al robotului.

În urma unei calibrări digitizor – robot virtual (paragraful 7.2.6) sunt generate modelele geometrice simplificate ale elementelor din spațiul de lucru al robotului în mediul virtual CAD pornind de la puncte sau șiruri de puncte indicate cu digitizorul 3D cu palpator. Indicarea punctelor se face prin palparea punctelor caracteristice (de exemplu colțuri) ale obiectelor existente fizic. Metodologia și programul de generare a entităților geometrice elementare pe baza datelor obținute prin digitizare este prezentată în paragraful 7.3. În cazul în care modelul geometric al elementelor din mediul de lucru al robotului există deja în format CAD (de exemplu dispozitivul a fost proiectat în firma respectivă) cu ajutorul digitizorului 3D se realizează situarea modelului geometric al dispozitivului în conformitate cu situația reală. De exemplu în mediul AutoCAD se poate utiliza comanda “Align” și prin indicarea corespondenței a trei perechi de puncte din modelul CAD și respectiv de pe suprafața obiectului fizic se realizează deplasarea modelului CAD pentru a reflecta situația reală

7.2.9. Teleoperare cu digitizorul 3D.

În paragraful 7.2.5 s-a prezentat modalitatea prin care efectorul final al robotului virtual poate fi deplasat utilizând aplicația RobSim pentru a urmări situația palpatorului digitizorului 3D deplasat manual de către operatorul uman. În mod identic cu deplasarea robotului virtual se poate realiza deplasarea unui robot real. Standul experimental se compune dintr-un robot SCORBOT ER III, un digitizor 3D cu structură RRR cu palpator și un calculator PC 466 MHz dotat cu 4 porturi seriale (fig 7.2.30.). Feed-back-ul pentru sesizarea deplasării efectorului final al robotului este realizat vizual. Nu sunt utilizați senzori de forță. Cele 4 porturi seriale a calculatorului PC sunt utilizate în modul următor:

Port 1: pentru interacțiunea utilizatorului cu aplicația RobSim;

Port 2: pentru determinarea situației relative a elementelor primelor 2 cuple cinematice ale digitizorului 3D;

Port 3: pentru determinarea situației relative a elementelor celei de-a treia cuple cinematice a digitizorului 3D;

Port 4: Pentru comunicația cu robotul.

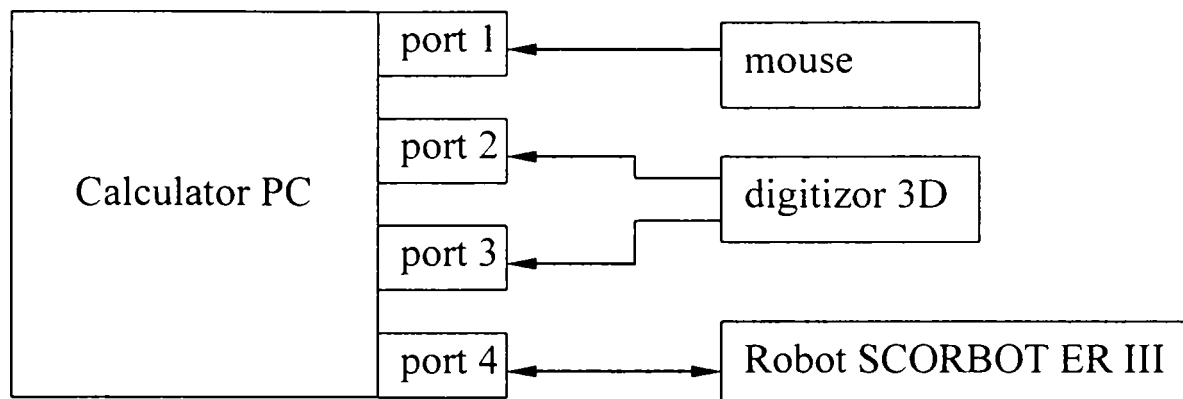


Fig. 7.2.30. Comunicația calculator - echipamente

Poziția punctului caracteristic al digitizorului 3D (vârful palpatorului) este determinată prin calcul cinematic direct pe baza contorizării impulsurilor traductorului incremental de rotație. Calculul corespunzător modelului cinematic invers este realizat de către PC pe baza programului implementat de autor, iar prin intermediul portului serial 4 se transmit comenzile la microcontrolerul robotului.

Sesizarea deplasării efectorului final al robotului este realizată de către operatorul uman prin observare directă a situației robotului în vecinătatea digitizorului. În cazul în care robotul este situat la distanță este vizualizată pe monitor imaginea preluată de o cameră de luat vederi care monitorizează scena de operare a robotului.

7.3. Generarea modelului geometric pe baza informațiilor prelevate cu digitizorul 3D

Generarea într-o aplicație de proiectare asistată a unor entități pe baza unor puncte indicate cu digitizorul 3D pe suprafața unor obiecte reale este diferită față de generarea acelorași entități prin comenzi specifice aplicației CAD. De exemplu în AutoCAD, pentru generarea unei sfere se cere specificarea unui punct pe centru și dimensiunea razei sferei. Pentru a genera aceeași entitate pe baza unor puncte indicate cu digitizorul 3D sunt necesare patru puncte necoplanare. Pe baza acestor patru puncte se calculează coordonatele centrului sferei și raza acesteia apoi se apelează comanda AutoCAD pentru generarea modelului geometric al sferei.

Pentru a realiza procesarea coordonatelor punctelor indicate cu digitizorul 3D, pregătirea acestor date pentru a obține parametri ceruți de comenzile AutoCAD pentru generarea entităților geometrice și apelarea comenzilor corespunzătoare s-a realizat o aplicație software numită DigiCAD. Aplicația DigiCAD este realizată de către autor în mediul VisualBasic6 pentru AutoCAD2000. În figura 7.3.1 este prezentată o captură de ecran a aplicației din momentul generării unei suprafețe biliniare pornind de la 4 puncte. Pentru obținerea datelor necesare generării entităților geometrice sunt utilizate 2 funcții care permit specificarea cu digitizorul a unui punct, respectiv a unui șir de puncte.

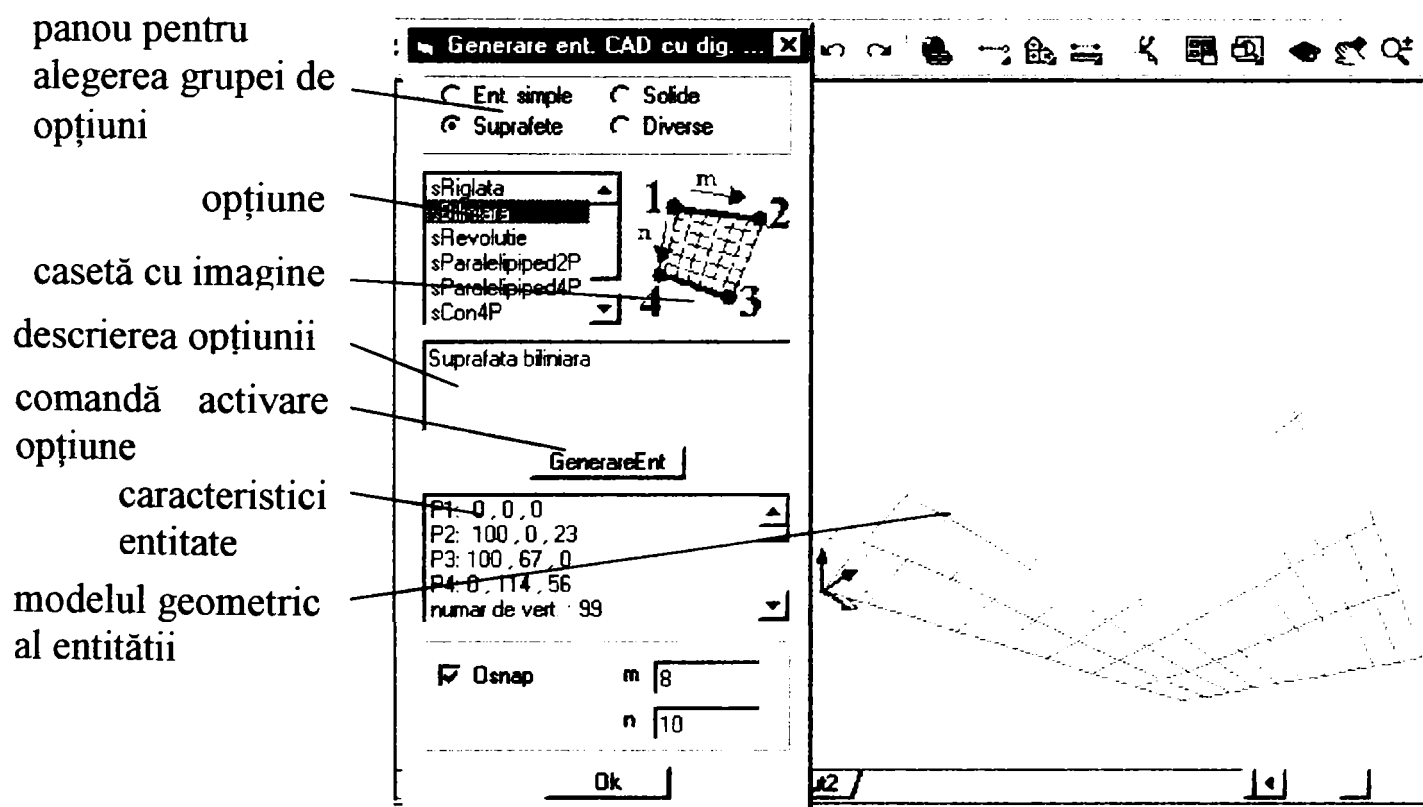


Fig. 7.3.1 Captură de ecran preluată în timpul rulării aplicației DigiCAD

Aplicația are 4 opțiuni principale corespunzătoare pentru generarea entităților simple (punct , linie, polilinie 3D, cerc prin 3 puncte, etc), generarea modelelor solide, generarea modelelor de suprafață și respectiv facilități diverse (vector, matrice de

situare, unghi în 3D, schimbarea direcției de vizualizare, etc.). Se exemplifică modul de lucru pentru câteva din facilitățile implementate.

Pentru generarea unui model geometric cu solide al unui corp cilindric se aplează grupa “Solide” și se alege opțiunea “cilindru4P” (fig. 7.3.2.) .

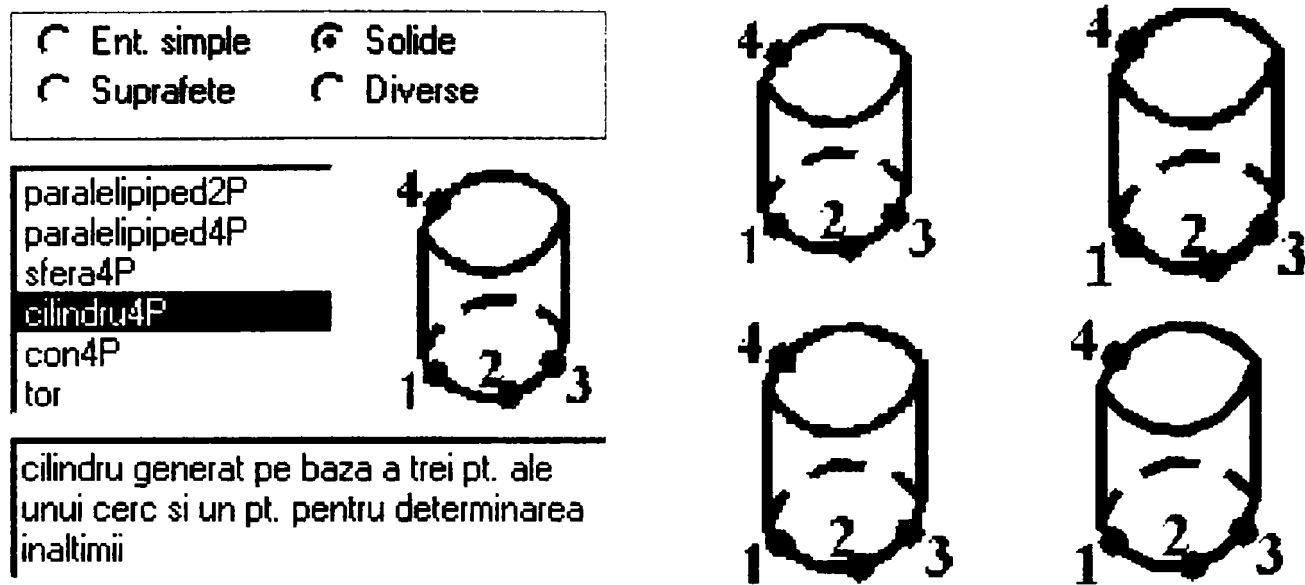
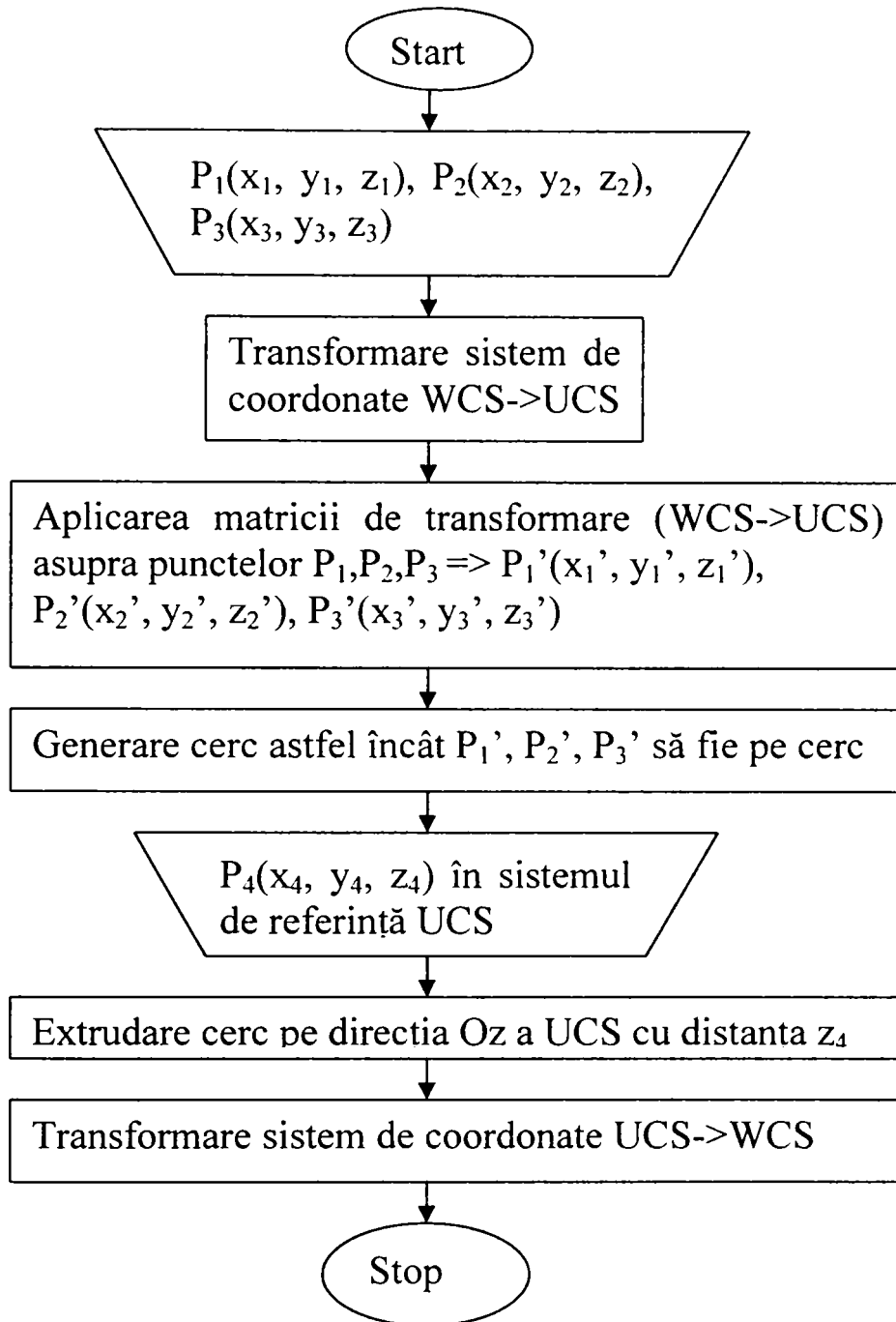


Fig 7.3.2 Aplicația cere utilizatorului indicarea a patru puncte pentru generarea modelului geometric cu solide al cilindrului

Aplicația cere indicarea succesivă a trei puncte pe unul din cercul uneia din baze. În AutoCAD, cercul este generat în planul xOy al sistemului de coordonate curent. Pentru a genera baza cilindrului într-o situare corespunzătoare obiectului real, se utilizează o comandă AutoCAD care permite indicarea unui sistem de coordonate prin 3 puncte: primul punct pentru stabilirea originii, al doilea punct situat pe axa Ox și un alt treilea punct în planul xOy în primul cadran. În urma schimbării sistemului de coordonate al modelului AutoCAD se realizează transformarea coordonatelor punctelor prin aplicarea matricii de trecere de la sistemul de coordonate universal la noul sistem de coordonate. Se apelează comanda AutoCAD “circle” cu opțiunea “3P” pentru generarea cercului prin 3 puncte date. Pentru stabilirea înălțimii cilindrului se utilizează un al patrulea punct situat pe cealaltă bază a cilindrului. Prin transformarea coordonatelor celui de-al patrulea punct din sistemul de coordonate absolut în sistemul de coordonate curent al modelului se obține valoarea înălțimii ca și coordonată z a celui de-al patrulea punct. Modelul geometric al cilindrului se realizează prin extrudarea cercului pe direcția z a sistemului de coordonate curent cu înălțimea determinată. În urma încheierii operației de generare a cilindrului, sistemul de coordonate al modelului revine la sistemul de coordonate absolut. Schema logică a algoritmului pentru generarea modelului solid a unui cilindru pe baza coordonatelor a 4 puncte indicate cu digitizorul este prezentat în figura 7.3.3.



7.3.3. Schema logică a algoritmului pentru generarea unui cilindru pe baza coordonatelor a 4 puncte indicate cu digitizorul 3D

Pentru generarea unei suprafețe biliniare utilizatorul indică cu ajutorul digitizorului 3D coordonatele a patru puncte (fig. 7.3.1). Numărul de vertexuri pe fiecare din direcțiile u și v ale suprafeței de tip rețea sunt specificate în casetele de text cu etichetele “ m ” și respectiv “ n ”. Pe baza acestor parametri se setează variabilele AutoCAD “Surftab1”(pentru direcția u) și “Surftab2”(pentru direcția v). Rețeaua este generată utilizând comenzi specifice AutoCAD iar informațiile despre caracteristicile entității generate sunt afișate (fig. 7.3.1).

Pentru generarea unei polilinii 3D pe baza informațiilor referitoare la coordonatele unui șir de puncte indicat cu ajutorul digitizorului se apelează o funcție de citire continuă a coordonatelor punctului caracteristic al digitizorului 3D. În mod analog se generează și curbele spline. Punctele de control ale curbei spline sunt punctele a două șirul de puncte indicate cu digitizorul 3D.

Din cadrul funcțiilor încadrate la grupa “diverse” face parte și indicarea direcției de vedere a modelului AutoCAD cu ajutorul digitizorului 3D. Pentru indicarea direcției de vedere, aplicația cere utilizatorului specificarea a două puncte corespunzătoare poziției observatorului și respectiv a țintei. Prin realizarea facilității de preluare continuă a coordonatelor punctului caracteristic al digitizorului 3D, se poate realiza o navigare prin mediul virtual realizat în AutoCAD. Tot din cadrul facilităților încadrate la categoria diverse fac parte indicarea cu ajutorul digitizorului 3D a unui vector, determinarea valorii unui unghi în spațiu prin indicarea a 3 puncte, determinarea matricii de transformare de la sistemul de referință absolut al modelului (WCS) la un sistem de referință indicat prin 3 puncte.

În anexa 5 este prezentat codul sursă al unor funcții ale aplicației DigiCAD. Aplicația este utilă atât pentru generarea modelului geometric al unor obiecte disponibile fizic cât și pentru proiectarea unor obiecte noi.

7.4. Proiectarea asistată de calculator a mobilierului de ședere în industria mobilei pe baza datelor obținute prin digitizare (aplicație)

7.4.1. Prezentare generală

Realizarea modelelor geometrice ale unor obiecte existente fizic este cea mai frecventă utilizare a digitizoarelor 3D. În industria mobilei apare deseori situația în care clientul dorește execuția unor elemente de mobilier pentru care oferă modelele existente fizic ale acestora și pentru care nu dispune de un model geometric. Pentru execuția în serie a acestor produse este necesar ca în secțiile de producție să fie distribuite desenele de execuție ale pieselor care intră în componența produsului și a desenului de ansamblu. Aceste desene sunt numite în continuare norme interne. Realizarea desenelor de execuție prin măsurare cu mijloace clasice (ruletă, șubler, etc.) este dificilă deoarece în majoritatea cazurilor obiectele de mobilier au forme complexe rezultate din considerente estetice. Dificultatea este și mai pronunțată în cazul mobilei curbate.

Aplicația prezentată de autor are scopul de a realiza în cât mai mare măsură automatizarea procesului de elaborare a desenelor de execuție a pieselor de mobilier pornind de la informații obținute prin digitizarea obiectului existent fizic. Scăderea timpului de realizare a documentației desenate are ca efect scurtarea perioadei de timp dintre momentul efectuării comenzii de către client și momentul livrării lotului de produse. Aceasta este una din cerințele menținerii pe piață. În cazul utilizării unei aplicații software specializate pe realizarea semiautomată a normei interne cu date furnizate prin digitizare 3D, costul unui digitizor profesional și al aplicației software este amortizat rapid prin timpul economisit.

În momentul actual, pentru testarea pieței sunt trimise posivilor clienți modele de noi produse denumite “mostre”. Aceste mostre sunt realizate de către muncitori cu înaltă calificare denumiți modelişti. Modeliştii, foarte buni executanți și cu simț estetic, execută modele ale viitorului produs pornind de cele mai multe ori de la schițe sumare. Actualmente se realizează mai multe exemplare din aceste modele și sunt trimise potențialilor clienți în unele cazuri situați pe alt continent. Autorul propune realizarea unui singur exemplar, digitizarea acestuia și expedierea la potențialii clienți pentru evaluare a modelului 3D obținut prin digitizare. Programele care permit vizualizarea modelelor 3D (de exemplu în format 3DStudio) sunt la prețuri foarte accesibile.

Aplicația de concepție proprie a autorului, denumită *DigiMob* este destinată unui domeniu al mobilei și anume mobilierului de ședere executat din material lemnos. În cadrul ei este valorificată experiența anterioară a autorului dobândită în timpul desfășurării activității în cadrul atelierului de proiectare al fabricii de mobilă “Pîncotas S.A.” (actual S. C. “C.D. Trade” S.A.) . În timpul dezvoltării aplicației s-a utilizat digitizorul 3D cu fire realizat de autor și prezentat în subcapitolul 6.3.. Aplicația *DigiMob* este realizată ca o extensie a Aplicației AutoCAD versiunea 2002. Codul sursă al aplicației dezvoltate de autor este scris în limbajul VisualBasic6 și

VisualLisp. O descriere a interfeței cu utilizatorul și fragmente din codul sursă al aplicației sunt prezentat în anexa 7.

În vederea elaborării unor algoritmi pentru realizarea modelelor geometrice ale pieselor s-a realizat o grupare a pieselor. Tehnologia de grup, considerată din punct de vedere al modelării geometrice, este prezentată în paragraful 7.4.2. și se suprapune în mare măsură peste gruparea pieselor din punct de vedere a tehnologiei de execuție. Pentru fiecare grup de piese se prezintă în paragraful 7.4.3. algoritmi pentru realizarea modelelor geometrice ale pieselor. Acolo unde a fost posibil s-a urmărit cotarea parțial automată a desenelor de execuție. Algoritmii prezentați au în principal scopul de a pregăti datele într-o formă în care să poată fi utilizate în mod automat comenzi specifice AutoCAD. În paragraful 7.4.4. este prezentată o aplicație care permite realizarea desenelor de execuție ale sculelor și șabloanelor utilizate pentru execuția unei piese pornind de la informații obținute prin digitizare.

7.4.2. Tehnologia de grup pentru fabricarea pieselor de mobilier

7.4.2.1. Clasificarea mobilierului de ședere

Din punct de vedere a tehnologiei de execuție, mobilierul de ședere se clasifică în trei categorii principale: tâmplăresc, colonial și curbat. În figura 7.4.1. este prezentat câte un exemplar din fiecare categorie și denumirile pieselor componente. Produsele din fiecare categorie pot să conțină și un număr redus de piese din celelalte două categorii.

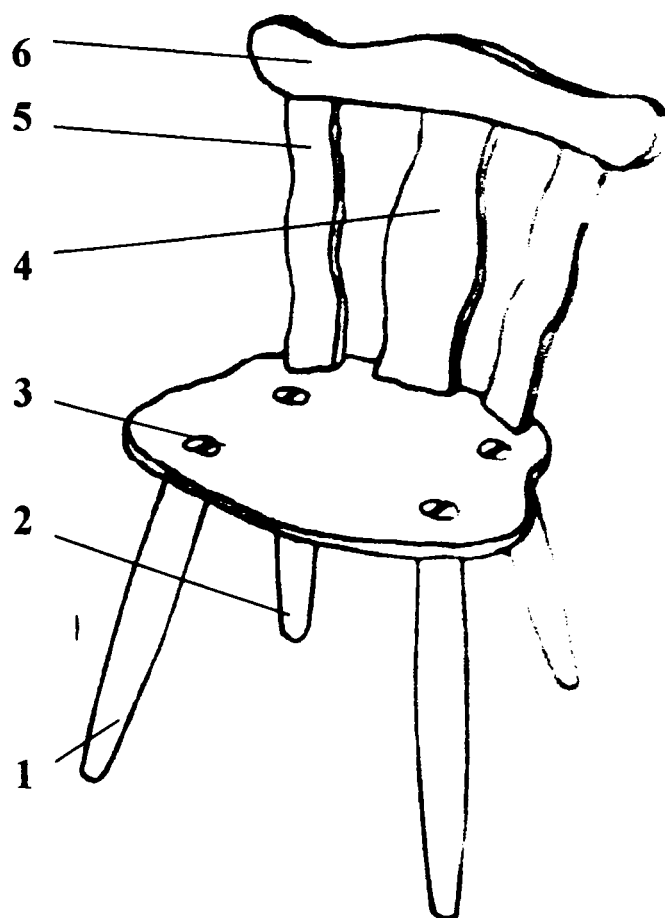


Fig.7.4.1.a) Scaun tâmplăresc

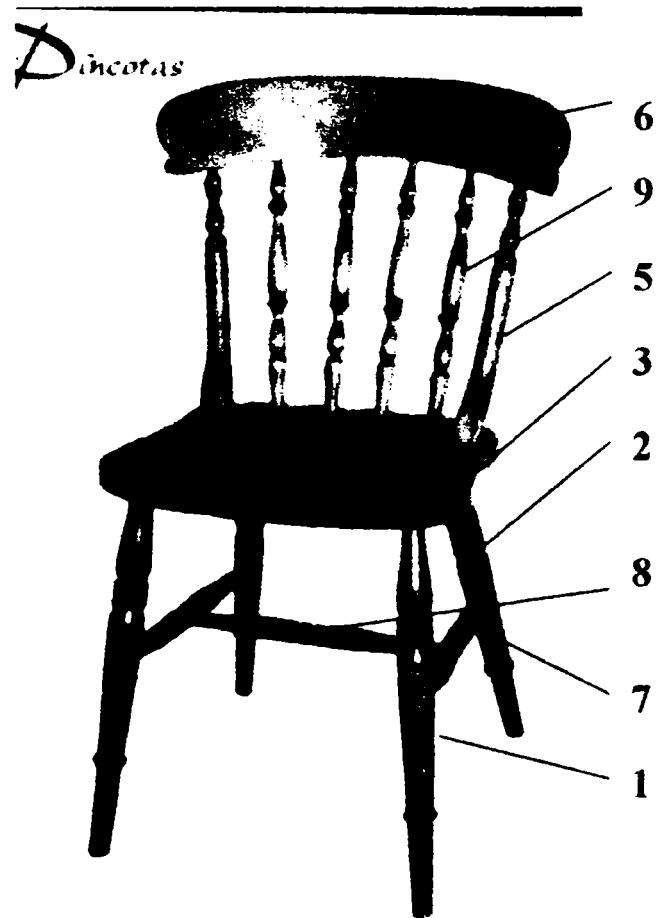
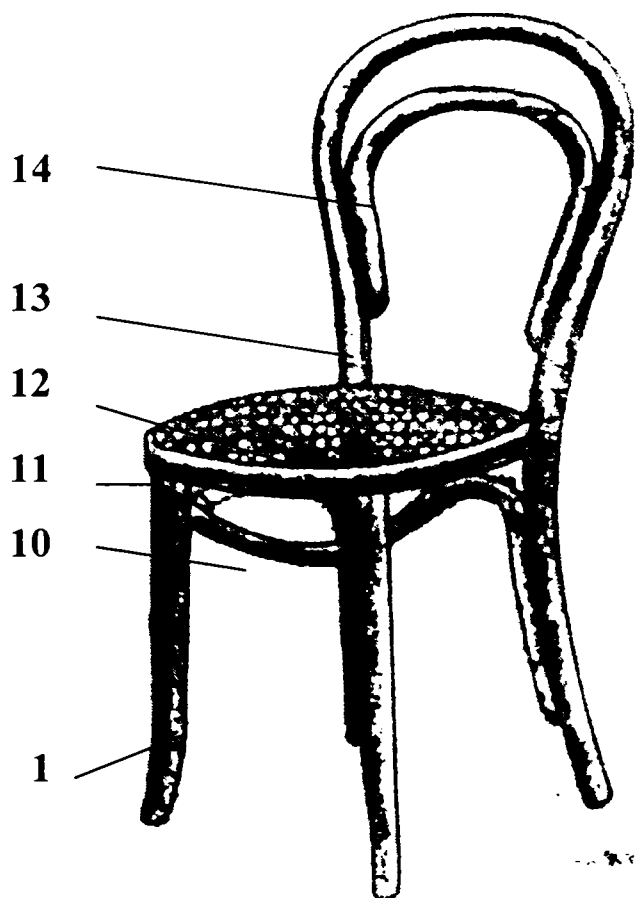


Fig.7.4.1.b) Scaun colonial



Legendă.

1. picior față
2. picior spate
3. șezut
4. liră spătar
5. susținător spătar
6. spătar
7. legătură picior laterală
8. legătură picior transversală
9. balustru
10. arc picior
11. ramă șezut
12. împletitură din trestie
13. rezemător
14. arc spătar

Fig.7.4.1.c) Scaun curbat

În componența scaunelor tâmplărești intră în special elemente de formă paralelipipedică sau piese obținute prin tăiere după contur din semifabricate paralelipipedice. Din punct de vedere al modelării geometrice, ele sunt cel mai ușor de reprezentat și în mare parte desenele de execuție pot fi realizate cu comenzile AutoCAD. Cu ajutorul digitizorului 3D se preiau de la mostre în special informații dimensionale (lungime, lățime, grosime) și coordonatele unor puncte izolate. Coordonate ale unor șiruri de puncte sunt necesare doar pentru definirea conturului în cazul pieselor obținut prin tăiere după contur.

Majoritatea pieselor din componența scaunelor coloniale sunt piese de revoluție. Se execută prin strunjire pe strunguri automate pornind de la semifabricate de secțiune pătrată.

Scaunele curbate sunt formate în special din piese de secțiune rotundă sau dreptunghiulară obținute prin curbare. Din punct de vedere al modelării acestea prezintă cea mai mare dificultate de realizare a desenelor de execuție.

7.4.2.2. Prezentarea pieselor care intră în fiecare din categoriile de mobilier

În tabelul 7.4.1. se prezintă piesele care pot intra în componența fiecărei grupe. Unele piese, deși ca denumire intră în componența mai multor grupe au forme specifice grupei. De exemplu legăturile picior în cazul scaunelor tâmplărești au secțiune dreptunghiulară iar în cazul scaunelor coloniale sunt piese de revoluție. În tabel marcajul semnifică existența piesei aflate în coloana din stânga în grupa respectivă.

Denumire piesă	sc. tâmplărești	sc. coloniale	sc. curbate
șezut	x	x	x
picior față	x	x	x
picior spate	x	x	
legătură picior transversală	x	x	
legătură picior longitudinală	x	x	
spătar	x	x	
balustru spătar	x	x	
susținător spătar	x	x	
brățar	x	x	
balustru brățar	x	x	
susținător brățar	x	x	
traversă subșezut	x		
liră	x	x	x
rezemător			x
arc spătar	x		x
ramă șezut			x
întărituri (bacuri)	x		x
legătură semicerc	x		
arc picior			x

Tabel 7.4.1. Piese care intră în componența celor 3 grupe de mobilier de ședere

Din punct de vedere al modelării geometrice autorul grupează piesele componente ale mobilierului de ședere în următoarele categorii:

- a) piese de revoluție cu secțiune variabilă (RV)
- b) piese curbate de secțiune circulară variabilă (SCV)

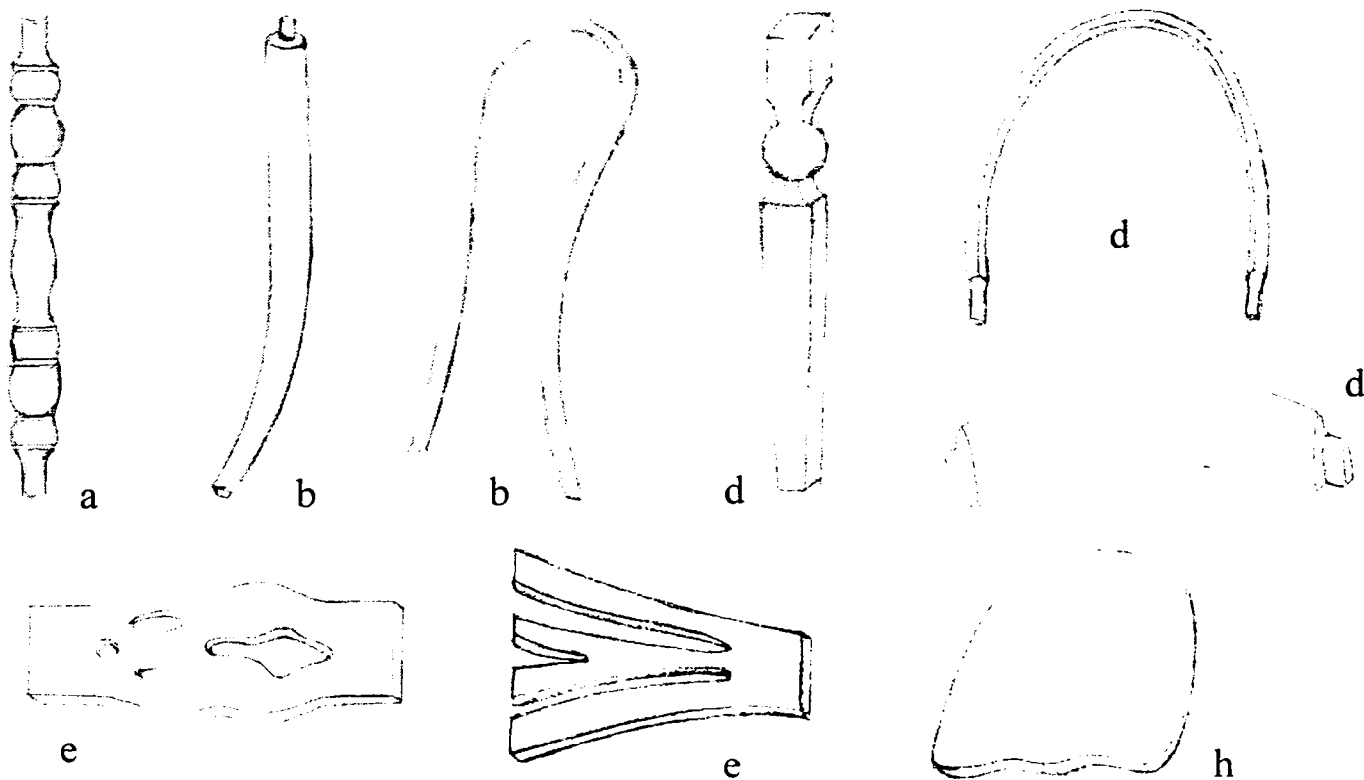


Fig. 7.4.2. Piese din categoriile de modelare

- c) piese curbate cu secțiune oarecare constantă (SC)
- d) piese cu treceri între diferite profile (TP)
- e) piese ce se obțin prin extrudare cu contur drept (ECD)
- f) piese obținute prin extrudare cu contur profilat (ECP)
- g) piese din placaj curbat cu contururi interioare (PCCI)
- h) piese cu mulaj complex (PMC)

În tabelul 7.4.2. se prezintă categoriile posibile din punct de vedere al modelării geometrice pentru fiecare piesă.

Denumire piesă	RV	SCV	SC	TP	ECD	ECP	PCCI	PMC
șezut						x		x
picior față	x	x	x	x	x			
picior spate	x	x	x	x	x			
legătură picior transversală	x			x	x			
legătură picior longitudinală	x			x	x			
spătar			x	x				
balustru spătar	x		x	x				
susținător spătar	x	x	x	x				
brățar		x	x	x				
balustru brățar	x			x	x			
susținător brățar	x			x	x			
traversă subșezut				x	x			
liră				x	x		x	
rezemător		x		x				
arc spătar		x						
ramă șezut			x	x				
arc picior		x						

Tabel 7.4.2. Piesele care intră în componența celor 3 grupe de mobilier de ședere

7.4.3. *Prezentarea algoritmilor pentru realizarea modelelor geometrice ale unor tipuri de piese pornind de la șiruri de puncte obținute prin digitizare*

7.4.3.1 Introducere. Probleme specifice

La realizarea modelelor geometrice ale unor obiecte pornind de la șiruri de puncte ale căror coordonate sunt obținute prin digitizare apar o serie de probleme specifice: necesitatea închiderii poliliniilor de contur, concatenarea șirurilor de puncte și determinarea planului mediu al unui șir de puncte.

a) Închiderea poliliniilor de contur

O serie de comenzi de generare a obiectelor în sistemul CAD au ca date de intrare polilinii închise. Poliliniile închise sunt poliliniile la care ultimul vertex al

polilinieii coincide cu primul vertex. În cazul generării poliliniilor pornind de la șiruri de puncte obținute prin digitizare aceste polilinii nu sunt închise (Fig. 7.4.3.a.). Pentru ca utilizatorul să fie sigur că a preluat coordonatele punctelor de pe întregul contur, pe zona de închidere a conturului se vor prelua redundant o serie de puncte. Astfel, în zona de închidere, va exista o suprapunere aproximativă a unei porțiuni a polilinieii. Pentru determinarea polilinieii închise se realizează prin software o “rețezare” a porțiunii redundante.

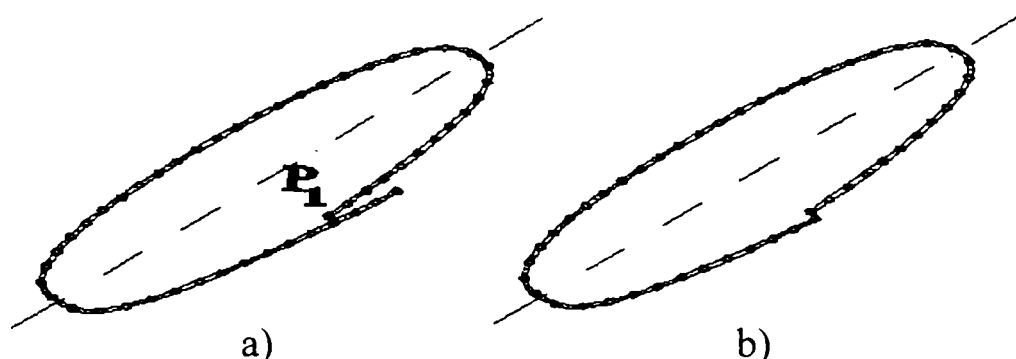


Fig. 7.4.3. Închiderea conturului preluat cu digitizorul

Algoritmul presupune parcurgerea întregului șir de puncte și determinarea distanței față de primul vertex al polilinieii. În cazul în care distanța dintre primul vertex (P_1) și vertexul curent (P_i) din șirul de puncte este mai mică decât o valoare dată d_{min} se elimină vertexurile situate în șirul de puncte în urma vertexului curent (fig.7.4.3.b). Vertexul curent este vertexul pentru care se calculează distanța la un moment dat față de primul vertex. Compararea distanțelor începe doar din momentul în care distanța dintre vertexurile P_i și P_1 devine mai mare decât d_{min} , pentru ca algoritmul să nu se oprească la primele vertexuri ale polilinieii.

b) Concatenarea poliliniilor

Concatenarea poliliniilor formate din șiruri de puncte este necesară atunci când la preluarea informațiilor despre conturul unui obiect, sunt preluate succesiv mai multe șiruri de puncte. Această situație apare când digitizarea continuă a unui șir de puncte trebuie întreruptă datorită imposibilității accesului palpatorului pentru palparea continuă a întregului contur sau deoarece accesul palpatorului este mai facil din mai multe direcții.

Se descrie algoritmul de concatenare utilizând notațiile din figura 7.4.4.. Se consideră în mod arbitrar una din polilinii. Fie această polilinie Pol_1 . Se determină distanța între ultimul vertex al polilinieii Pol_1 notat V_{n1} și vertexurile celorlalte polilinii. În momentul în care un vertex aparținând altei polilinii (Pol_2) se află la o distanță mai mică decât o distanță d_{min} specificată anterior față de V_{n1} se divide polilina Pol_2 și se elimină din aceasta porțiunea cu lungimea mai redusă. Vertexul aparținând polilinieii Pol_2 a cărei distanță față de V_{n1} este mai mică decât d_{min} devine vertexul care succede pe V_{n1} în polilinia Pol_1 , iar ultimul vertex al porțiunii păstrate din Pol_2 devine ultimul vertex al polilinieii Pol_1 . Algoritmul se repetă prin considerarea

tuturor poliliniilor obținându-se o polilinie deschisă. Pentru închiderea polilinie se utilizează algoritmul descris la 7.4.3.1.a).

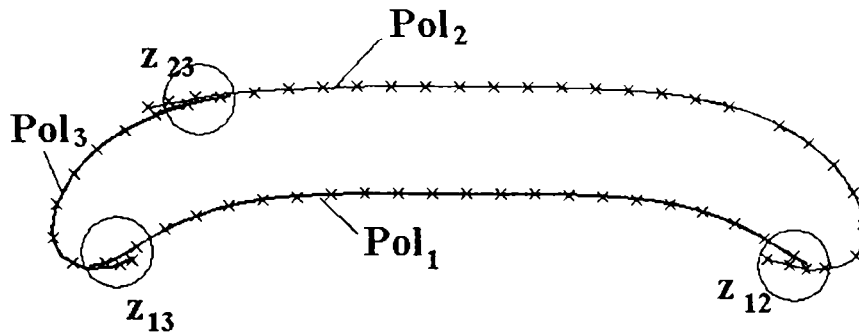


Fig. 7.4.4. Concatenarea poliliniilor

c) Determinarea planului mediu al unui șir de puncte

O serie de comenzi AutoCAD utilizate pentru generarea modelului geometric al pieselor au ca date de intrare polilinii 2D. Poliliniile obținute din șirurile de puncte digitizate sunt polilinii 3D. Chiar în cazul în care vârful digitizorului este deplasat pe o traiectorie plană, punctele din șir nu sunt situate în același plan. Majoritatea punctelor din șirul de puncte deviază cu o anumită distanță față de planul care conține traiectoria. Se propune un algoritm pentru determinarea polilinie 2D obținută prin proiectarea polilinie 3D pe un plan pentru care distanța de la vertexurile polilinie 3D față de plan este cât mai apropiată de minim. Acest plan este denumit în continuare plan mediu. Schema logică a algoritmului este prezentată în fig. 7.4.5.

Se realizează media coeficienților A, B, C și D din toate combinațiile de câte trei puncte din lista LP_{ext} . Planul cu coeficienții având valorile medii se consideră planul mediu. Pentru a da utilizatorului o imagine globală a deviației șirului de puncte obținut cu palpatorul față de planul mediu se calculează și sunt afișate deviația medie și cea maximă. În acest scop se calculează pentru fiecare punct $P_0(x_0, y_0, z_0)$ din șir distanța d față de plan cu relația 7.4.3.

$$d = \frac{|A \cdot x_0 + B \cdot y_0 + C \cdot z_0 + D|}{\sqrt{A^2 + B^2 + C^2}} \quad (7.4.3)$$

Pentru obținerea unei polilinie 2D se proiectează fiecare punct din șirul inițial pe planul mediu. Proiecția punctului $P_0(x_0, y_0, z_0)$ pe plan este punctul de intersecție $P(x, y, z)$ al dreptei ce trece prin punctul P_0 și are parametrii directori l, m, n ai vectorului normal la plan. Parametrii directori ai dreptei se deduc din cosinusurile directe ale normalei la plan :

$$\cos \alpha = \frac{A}{\sqrt{A^2 + B^2 + C^2}}; \quad \cos \beta = \frac{B}{\sqrt{A^2 + B^2 + C^2}}; \quad \cos \gamma = \frac{C}{\sqrt{A^2 + B^2 + C^2}} \quad (7.4.4)$$

Coordonatele punctului de intersecție sunt date de relațiile:

$$\begin{aligned} x &= x_0 - \frac{A \cdot x_0 + B \cdot y_0 + C \cdot z_0 + D}{A \cdot l + B \cdot m + C \cdot n} \cdot l \\ y &= y_0 - \frac{A \cdot x_0 + B \cdot y_0 + C \cdot z_0 + D}{A \cdot l + B \cdot m + C \cdot n} \cdot m \end{aligned} \quad (7.4.5)$$

$$z = z_0 - \frac{A \cdot x_0 + B \cdot y_0 + C \cdot z_0 + D}{A \cdot l + B \cdot m + C \cdot n} \cdot n$$

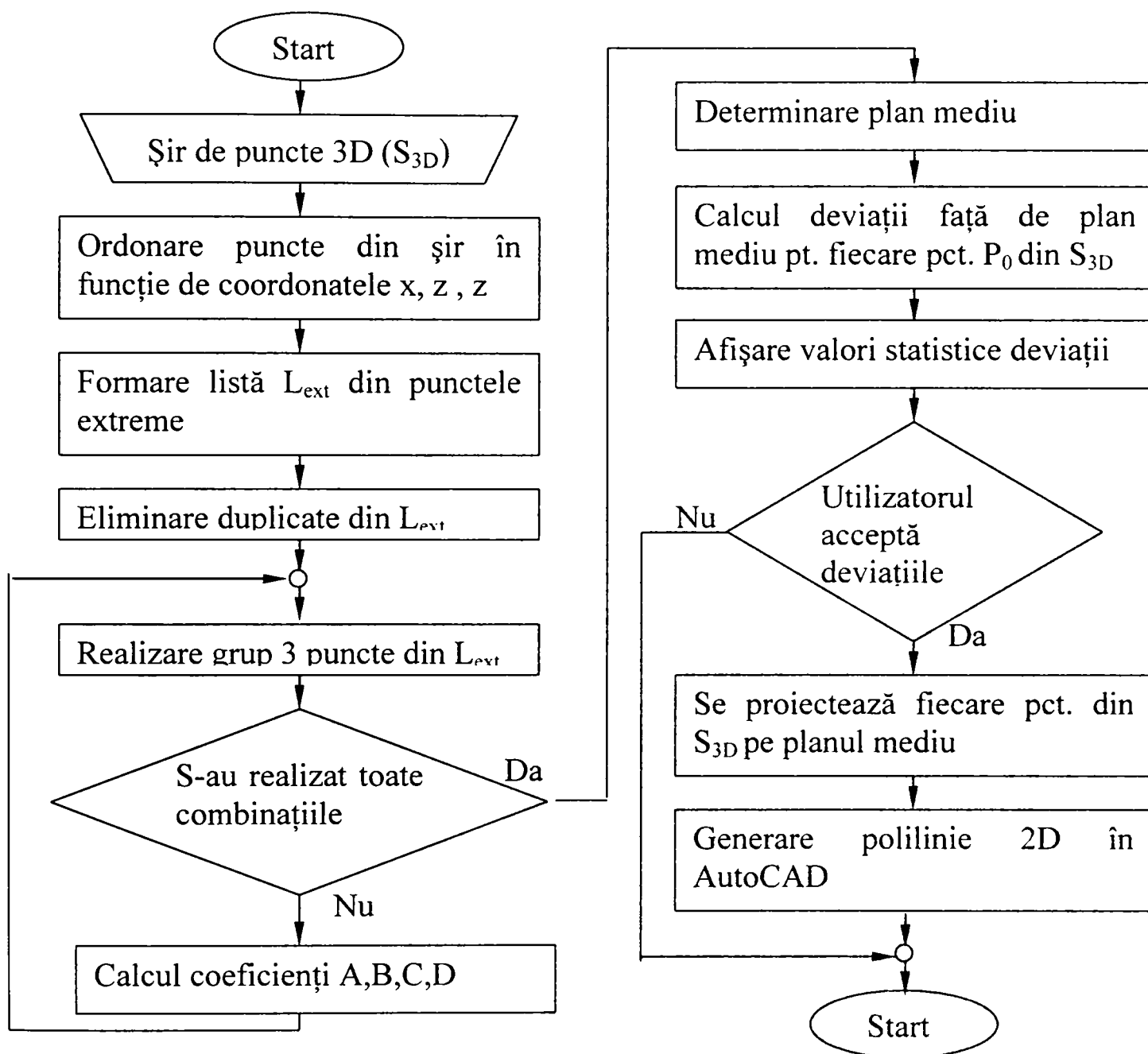


Fig. 7.4.5. Schema logică a algoritmului pentru determinarea polilinieii 2D ce aproximează polilinia 3D obținută din șirul de puncte digitizate

7.4.3.2. Modelarea pieselor de revoluție cu secțiune variabilă (RV)

În sistemul CAD, pentru realizarea modelului geometric al pieselor de revoluție este necesar să fie specificate o linie și o curbă 2D. Cu ajutorul digitizorului se pot obține doar coordonatele unor puncte de pe suprafața piesei. În fig. 7.4.6. este prezentată o piesă de acest tip. Se propune un algoritm pentru determinarea axei piesei și a unei curbe 2D pornind de la coordonatele punctelor 3D digitizate. Schema logică a algoritmului este prezentată în fig. 7.4.8.

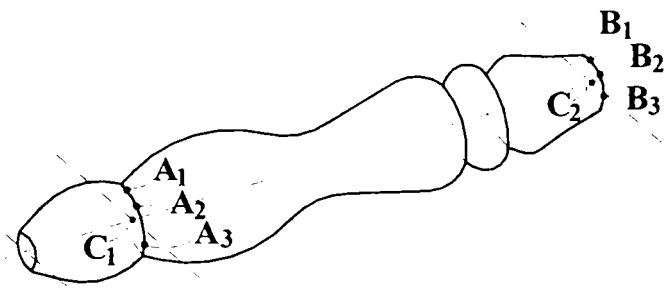


Fig.7.4.6. Palparea punctelor pentru definirea axei piesei

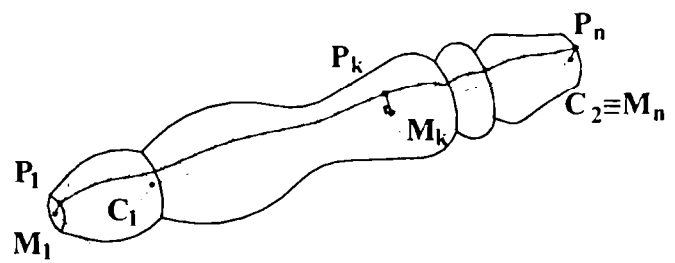


Fig.7.4.7. Determinarea profilului piesei

Pentru determinarea axei se palpează două grupuri de câte 3 puncte de pe suprafața piesei. Se notează cu A_1, A_2, A_3 primul grup de puncte și cu B_1, B_2, B_3 al doilea grup. Este important ca cele 3 punctele din fiecare grup să fie situate într-un plan perpendicular pe axa piesei. Pentru a asigura acest lucru se palpează punctele fie pe muchia de capăt a piesei fie pe o inflexiune concavă a profilului (fig. 7.4.7). Se preferă alegerea celor două grupuri la cele 2 extremități ale piesei sau la o distanță cât mai mare. Cu cât distanța dintre grupurile de puncte este mai mare, eroarea indusă la determinarea axei piesei de către eventuala ovalizare a secțiunii piesei este mai redusă. Se determină planul format de cele 3 puncte A_i . Se determină matricea ${}^{3P}T_{mod}$ de trecere de la sistemul de referință al modelului SR_{mod} la un sistem de referință SR_{3P} având axele Ox și Oy situate în planul ce trece prin punctele A_i . Fie punctele A_i' punctele obținute prin aplicarea matricii ${}^{3P}T_{mod}$ asupra punctelor A_i . Se observă că se obține 0 pentru coordonatele z ale punctelor. Se determină ecuația cercului ce trece prin cele 3 puncte $A_1(x_{A1}, y_{A1}), A_2(x_{A2}, y_{A2})$ și $A_3(x_{A3}, y_{A3})$:

$$\begin{vmatrix} x^2 + y^2 & x & y & 1 \\ x_{A1}^2 + y_{A1}^2 & x_1 & y_1 & 1 \\ x_{A2}^2 + y_{A2}^2 & x_2 & y_2 & 1 \\ x_{A3}^2 + y_{A3}^2 & x_3 & y_3 & 1 \end{vmatrix} = 0 \quad (7.4.6)$$

Ecuația generală a cercului de centru $I(a,b)$ și rază R este :

$$x^2 + y^2 - 2ax - 2by + c = 0 \quad (7.4.7)$$

Din egalarea coeficienților coordonatelor x și y din ecuațiile 7.4.6. și 7.4.7. rezultă coordonatele a și b ale centrului cercului. Se aplică transformarea inversă pentru trecerea de la sistemul de referință SR_{3P} la SR_{mod} asupra punctului $I(a,b,0)$ și se obțin coordonatele 3D ale centrului cercului ce trece prin A_1, A_2 și A_3 . Se notează cu C_1 centrul cercului determinat de punctele A_i și cu C_2 centrul cercului ce trece prin punctele B_i cu $i=1, \dots, 3$. Axa piesei este o dreaptă dr_{cen} ce trece prin punctele C_1 și C_2 .

În vederea determinării poliliniiei 2D care reprezintă profilul, utilizatorul palpează un șir de n puncte P_k pe suprafața piesei, de-a lungul acesteia. Primul punct P_1 al șirului este situat aproximativ pe muchia unei extremități a piesei. Pentru fiecare punct din șir: se determină M_k ca proiecție a punctului P_k pe dreapta C_1C_2 , se calculează distanțele r_k dintre punctele P_k și M_k și d_{1k} dintre punctele M_1 și M_k . Punctele $D_k(d_k, r_k)$ reprezintă profilul 2D al piesei de revoluție. Pentru generarea modelului geometric de suprafață a piesei de revoluție se aplică comanda AutoCAD

“Revolve” cu 2 parametrii: polilinia 2D formată din punctele D_k și axa x a sistemului de coordonate al modelului. În urma generării modelului se aplică acești matricea de trecere care realizează suprapunerea originii sistemului de referință SR_{mod} cu punctul M_1 și a axei Ox a sistemului de referință SR_{mod} cu dreapta C_1C_2 . Prin aceasta modelul piesei ajunge în situația corespondență piesei reale digitizate.

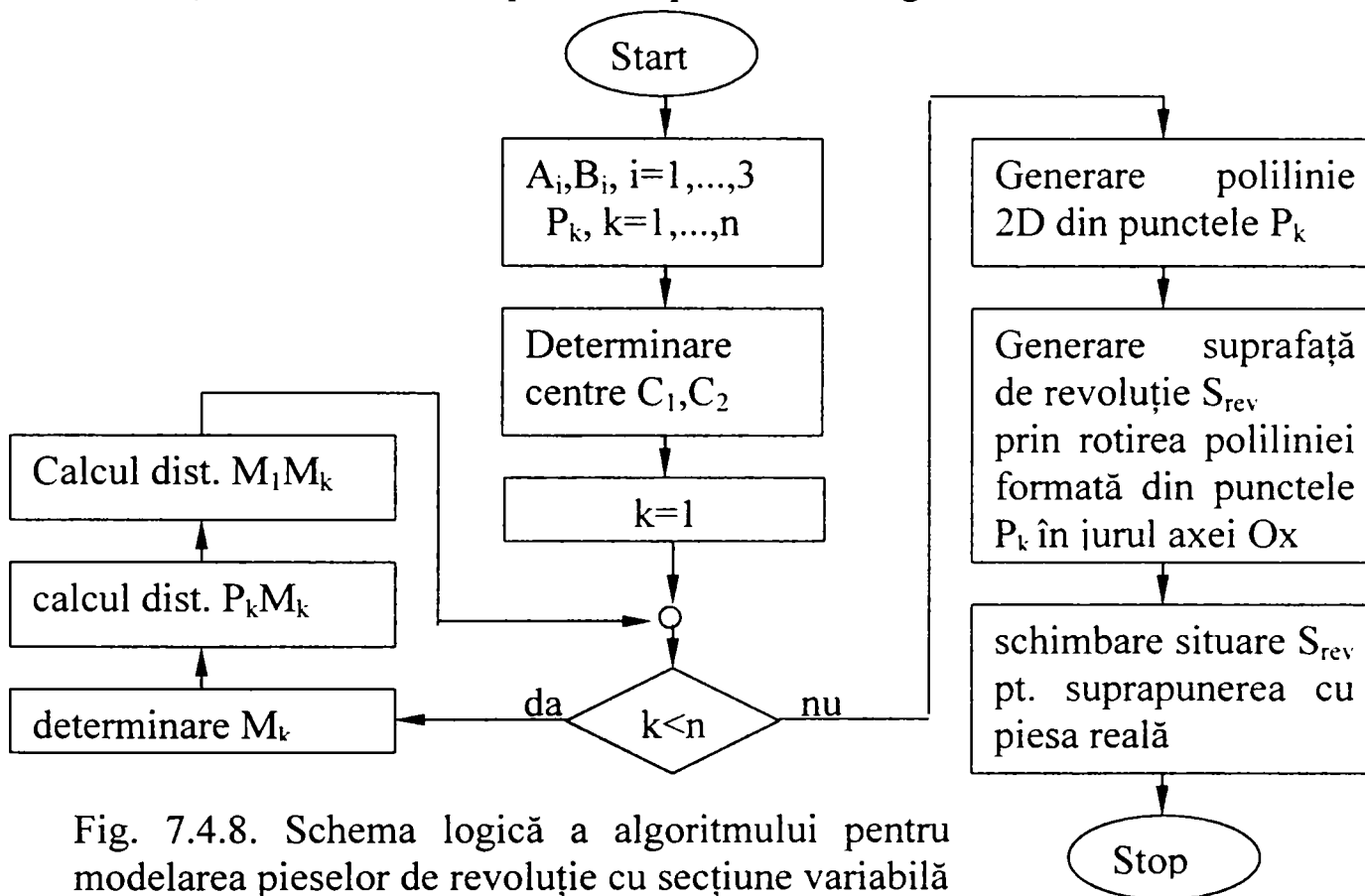


Fig. 7.4.8. Schema logică a algoritmului pentru modelarea pieselor de revoluție cu secțiune variabilă

7.4.3.3. Modelarea pieselor curbate de secțiune circulară variabilă (SCV)

Pentru determinarea curbei strâmbă în spațiu care reprezintă locul geometric al centrelor secțiunilor transversale prin piesă se utilizează un dispozitiv auxiliar care se atașează în locul vârfului palpatorului digitizorului 3D. Schema cinematică a acestui dispozitiv este prezentat în figura 7.4.9. Curba strâmbă în spațiu care reprezintă locul geometric al centrelor secțiunilor transversale prin piesă se aproximează printr-o polilinie 3D denumită în continuare polilinie de sweeping (prin analogie cu “linia de sweeping” din [Sav1997]). Denumirea de “sweeping” provine de la metoda de generare a suprafețelor prin baleierea unui profil de-a lungul unei cube. S-a utilizat termenul de “polilinie de sweeping” deoarece curba de-a lungul căreia se baleiază profilul este reprezentată ca polilinie 3D. Polilinia 3D este obținută pe baza informațiilor rezultate prin digitizare.

Pentru determinarea diametrului secțiunii piesei există mai multe soluții. Una din variante, cu precizie mai redusă, este indicarea diametrului corespunzător pentru câteva secțiuni prin indicarea cu digitizorul a câte 3 puncte. În acest caz razele secțiunilor se determină pornind de la relațiile prezentate la punctul 7.4.3.2.. Totodată se determină punctul de intersecție dintre polilinia de sweeping și planul format de

cele 3 puncte. Se notează acest punct cu O_i . Razele secțiunilor pe porțiunea dintre 2 secțiuni succesive pentru care s-a determinat raza se realizează prin interpolare.

O altă soluție pentru determinarea razei secțiunii este utilizarea dispozitivului a cărei schema cinematică este prezentată în figura 7.4.9. Odată cu determinarea coordonatelor centrului secțiunii se determină și raza acesteia.

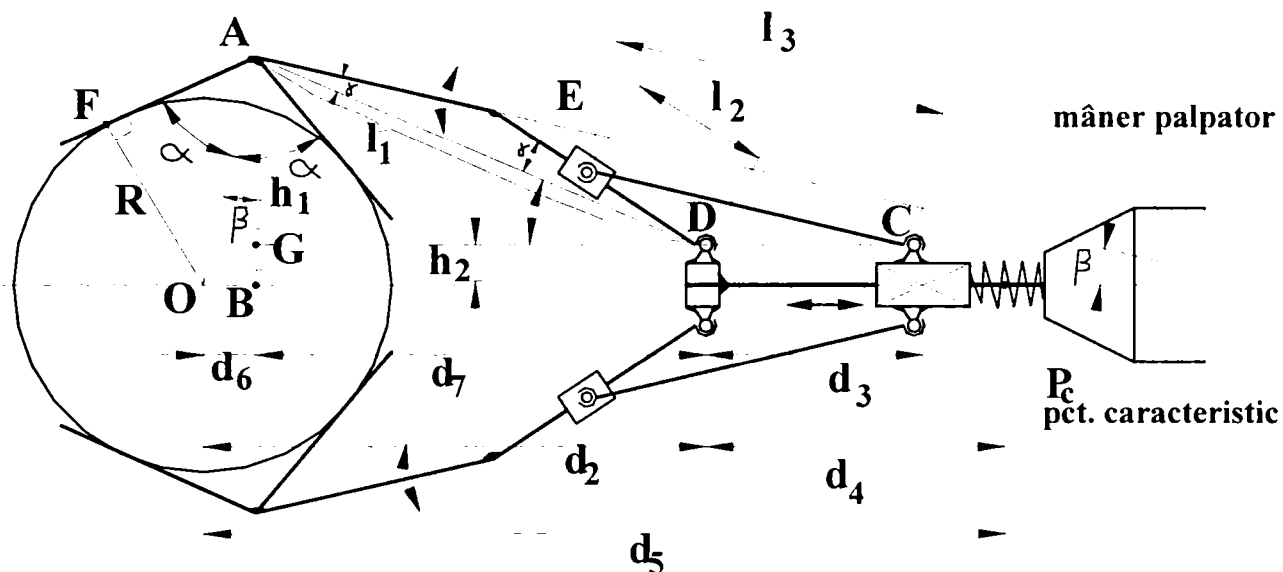


Fig. 7.4.9. Dispozitiv pentru determinarea vertexurilor poliliniei de sweeping și a razei secțiunii

Se cunosc prin construcție unghiurile α, γ și dimensiunile l_1, l_2, l_3, h_2, d_4 . Distanța între punctele C și D, notată cu d_3 este determinată cu ajutorul unui traductor liniar. Se urmărește stabilirea relațiilor de calcul pentru determinarea razei R a secțiunii și distanța d_5 între punctul caracteristic al digitizorului 3D și centrul secțiunii în funcție de distanța d_3 . În triunghiul dreptunghic AFO se respectă relația:

$$\sin \alpha = \frac{d_1}{R} \quad (7.4.8)$$

$$\text{Din triunghiul dreptunghic ABO, } \cos(\beta) = \frac{h_1}{d_1} \text{ deci } d_1 = \frac{h_1}{\cos \beta}. \quad (7.4.9)$$

$$\text{Din triunghiul dreptunghic AGD, } \operatorname{tg} \rho = \frac{h_1 - h_2}{l_1}. \quad (7.4.10)$$

$$\text{Unghiul } \rho = \gamma + \beta. \quad (7.4.11)$$

Din relațiile (7.4.8), (7.4.9), (7.4.10) și (7.4.11) se exprimă raza secțiunii:

$$R = \frac{l_1 \cdot \operatorname{tg} \rho + h_2}{\cos(\rho - \gamma)} \cdot \sin \alpha \quad (7.4.12)$$

$$\text{Din triunghiul dreptunghic AGD, } d_7 = l_1 \cdot \cos \rho. \quad (7.4.13)$$

$$\text{Din triunghiul ABO se obține : } d_6 = d_1 \cdot \sin \alpha. \quad (7.4.14)$$

Din (7.4.14) și (7.4.8) se exprima distanța d_6 :

$$d_6 = \frac{R \cdot \sin \beta}{\sin \alpha} \quad (7.4.15)$$

$$d_2 = d_6 + d_7 = \frac{R \cdot \sin \beta}{\sin \alpha} + l_1 \cdot \cos \alpha \quad (7.4.16)$$

Prin aplicarea teoremei lui Pitagora generalizată în triunghiul CDE se obține:

$$l_3^2 = l_2^2 + d_3^2 - 2 \cdot l_2 \cdot d_3 \cdot \cos(\pi - \rho - \gamma)$$

$$\rho = \pi - \gamma - \arccos\left(\frac{l_2^2 + d_3^2 - l_3^2}{2 \cdot l_2 \cdot d_3}\right) \quad (7.4.17)$$

Distanța dintre punctul caracteristic al digitizorului și centrul secțiunii piesei este:

$$d_5 = d_4 + d_2 \quad (7.4.18)$$

Coordonatele punctului caracteristic al digitizorului $P(x_p, y_p, z_p)$ sunt cunoscute, fiind determinate prin calcul geometric direct. De asemenea, este cunoscută orientarea dreptei caracteristice a digitizorului 3D. Dacă se notează cu l_c, m_c, n_c parametrii directori ai dreptei caracteristice a digitizorului 3D, coordonatele centrului secțiunii sunt:

$$x_0 = x_p + d_5 \cdot l_c; \quad y_0 = y_p + d_5 \cdot m_c; \quad z_0 = z_p + d_5 \cdot n_c \quad (7.4.19)$$

Algoritmul pentru determinarea razei secțiunii și a centrului secțiunii cu dispozitivul a cărei schemă cinematică este prezentată în figura 7.4.10 presupune parcurgerea următorilor pași: determinarea unghiului ρ pe baza relației (7.4.17), stabilirea valorii razei secțiunii pe baza relației (7.4.12), aflarea distanței d_5 pe baza relației (7.4.18) și calculul coordonatelor centrului secțiunii prin utilizarea relațiilor 7.4.19

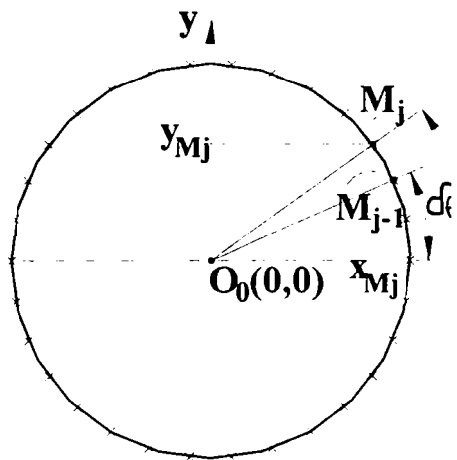


Fig. 7.4.10. Determinarea punctelor M:

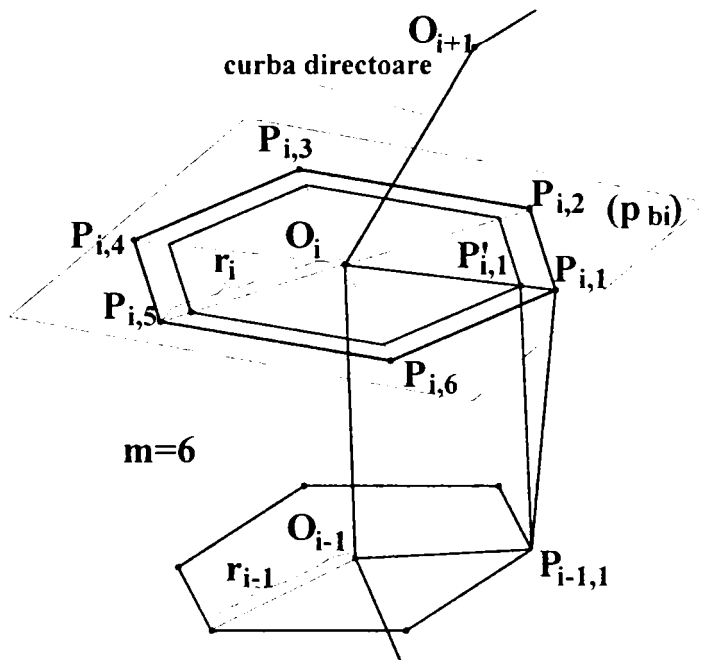
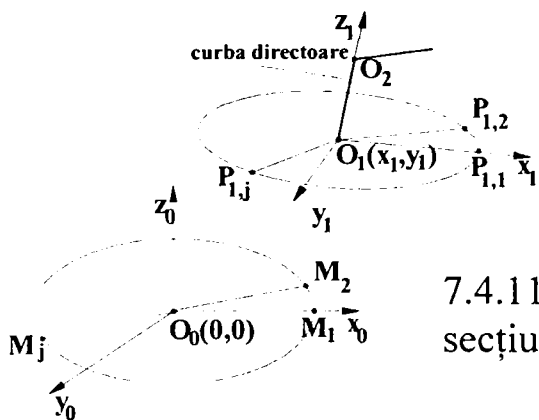


Fig. 7.4.12. Determinarea coordonatelor punctelor corespunzătoare unei secțiuni i



7.4.11. Se deplasează cercul reprezentând prima secțiune într-un plan perpendicular pe traiectorie

Suprafața piesei se realizează în continuare prin sweeping generalizat. Va rezulta o rețea 3D. Se notează cu m numărul de vertexuri corespunzătoare unei secțiuni.

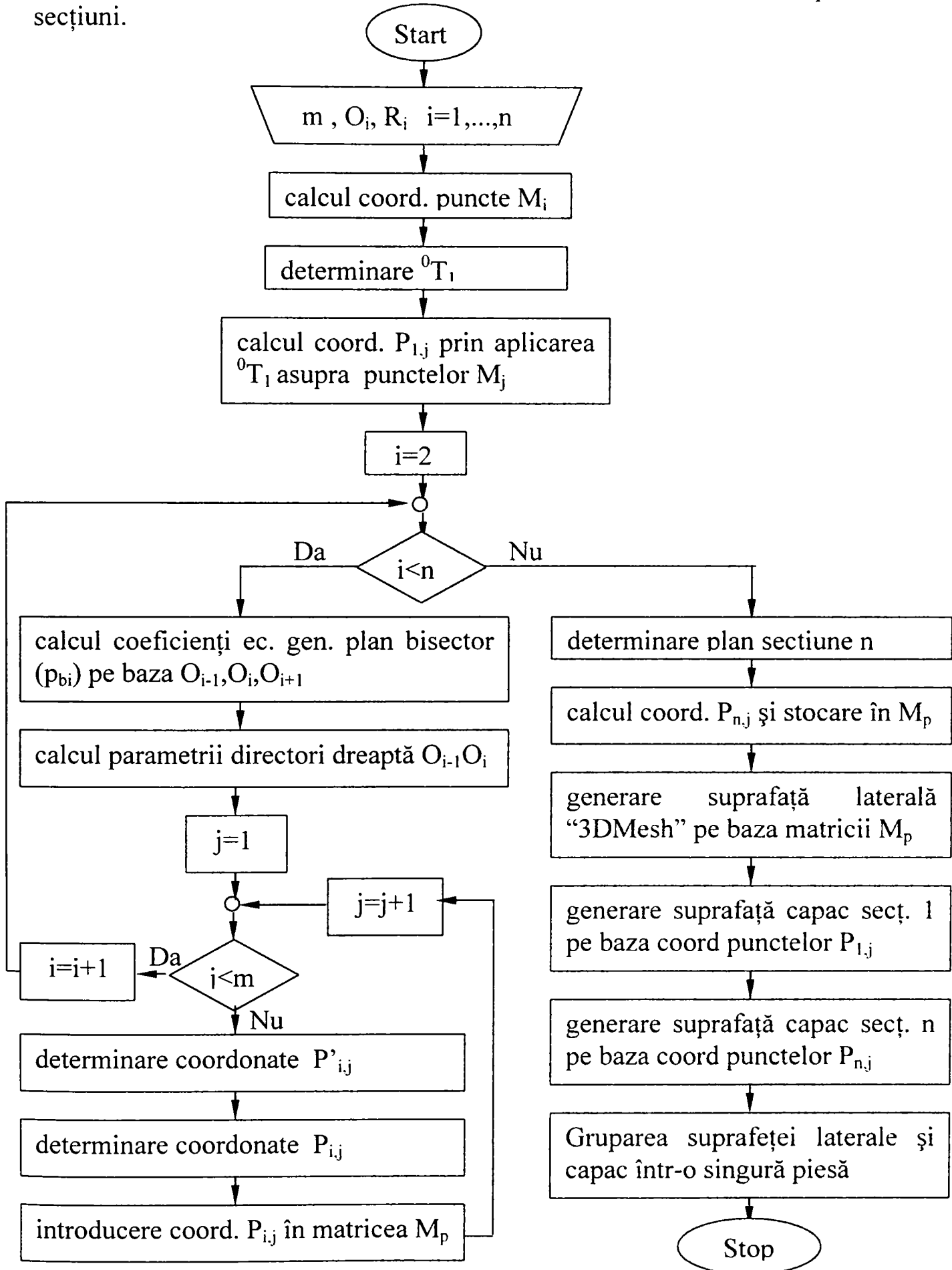


Fig. 7.4.13. Schema logică a algoritmului pentru modelarea pieselor curbate de secțiune circulară variabilă

Se notează cu n numărul de vertexuri al polilinieii de sweeping. Numărul de vertexuri ale rețelei care reprezintă suprafața laterală a piesei va fi $n \cdot m$. Schema logică a algoritmului utilizat pentru generarea suprafeței modelului piesei se prezintă în fig.7.4.13. Utilizatorul precizează numărul m de laturi ale poligonului regulat care aproximează conturul secțiunii circulare. Se consideră cercul cu centrul în originea sistemului de coordonate al modelului, și se determină punctele $M_j(x_{Mj}, y_{Mj})$ pe circumferință (fig.7.4.10). Se notează cu δ_θ unghiul elementar corespunzător diferenței unghiurilor față de axa Ox a două raze corespunzătoare unor puncte succesive pe cerc. Dacă m este numărul de puncte indicat de utilizator atunci $\delta_\theta = 2\pi / m$. Coordonatele punctului M_j pe cerc vor fi:

$$x_{Mj} = r \cdot \cos(j \cdot \delta_\theta); \quad y_{Mj} = r \cdot \sin(j \cdot \delta_\theta) \quad j = 1, \dots, m \quad (7.4.20)$$

Se notează cu 0T_1 matricea de trecere de la sistemul de coordonate al modelului SR_0 la un sistem de coordonate SR_1 la care axa z_1 are direcția și sensul vectorului determinat de punctele O_1O_2 . Punctul O_1 reprezintă primul vertex al polilinieii de sweeping. Asupra punctelor M_j se aplică matricea de trecere 0T_1 obținându-se punctele $P_{1,j}$. Primul indice al punctului P , cu valoarea 1, reprezintă indicele secțiunii și corespunde numărului de ordine al vertexului pe polilinia de sweeping. Indicele j reprezintă numărul vertexului în cadrul secțiunii.

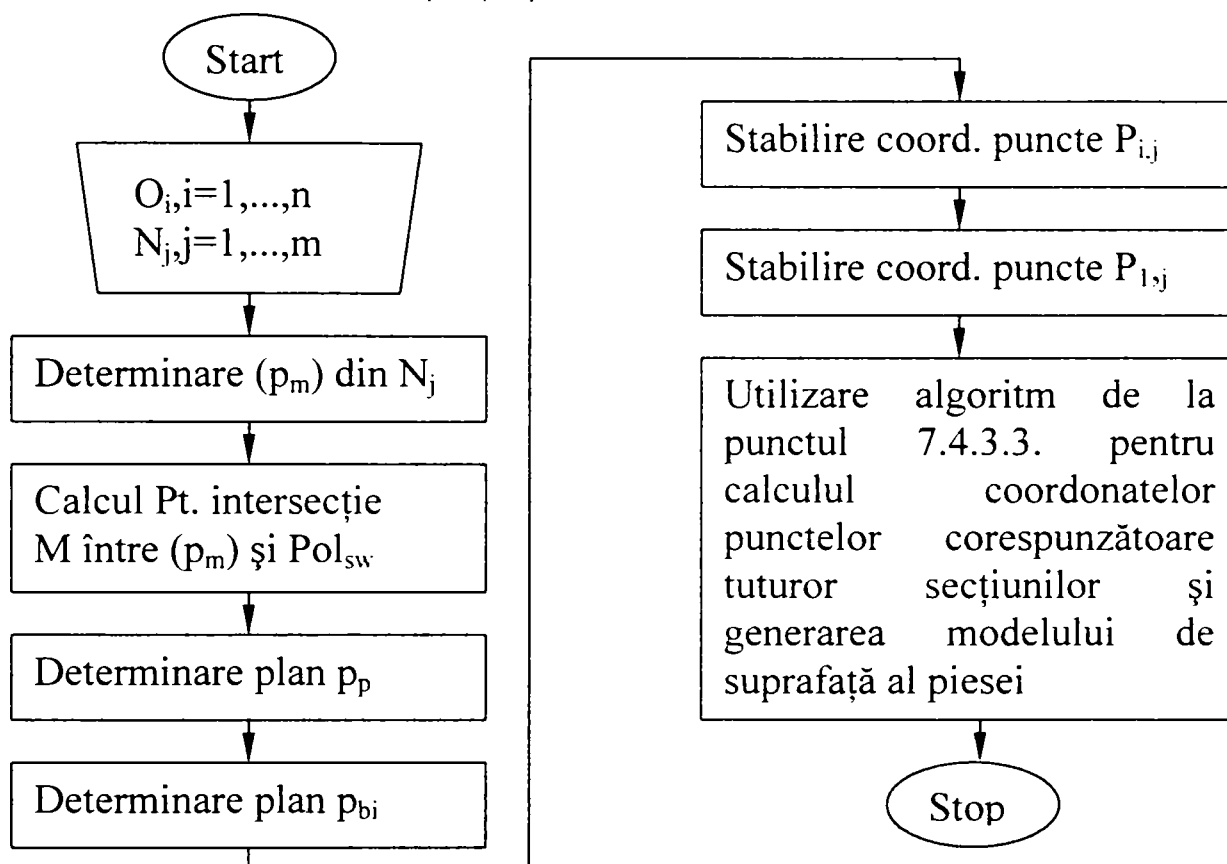
Pentru determinarea coordonatelor punctului $P_{i,j}$ a unei secțiuni corespunzătoare vertexului O_i a polilinieii de sweeping se determină planul secțiunii. Planul secțiunii cu indicele i este planul bisector (p_{bi}) al dreptelor concurente $O_{i-1}O_i$ și O_iO_{i+1} . Coeficienții ecuației generale a acestui plan se determină cu algoritmul prezentat în anexa 6. Pentru ultimul vertex al polilinieii de sweeping, planul secțiunii este perpendicular pe segmentul $O_{n-1}O_n$. Se determină punctul $P'_{i,j}$ ca intersecție cu planul (p_{bi}) a dreptei ce trece prin punctul $P_{i-1,j}$ și are direcția dreptei $O_{i-1}O_i$. Punctul $P_{i,j}$ este punctul situat la distanța r_i de punctul O_i pe direcția $O_iP'_{i,j}$. Se generează o matrice bidimensională M_p , de dimensiuni n respectiv m în care se stochează coordonatele punctelor $P_{i,j}$.

Suprafața laterală a piesei se generează în AutoCAD ca suprafață "3DMesh" pe baza coordonatelor punctelor stocate în matricea M_p . Suprafețele capac de la cele 2 extremități ale piesei se realizează cu comanda AutoCAD "Pface" pe baza punctelor $P_{1,j}$ pentru prima secțiune respectiv $P_{n,j}$ pentru ultima secțiune.

7.4.3.4. Modelarea pieselor cu secțiune de formă oarecare constantă (SC)

Algoritmul pentru generarea modelului pieselor de acest tip este prezentat în fig. 7.4.14. La modelarea pieselor SC se prelevează coordonatele punctelor situate pe o muchie de-a lungul piesei. Din aceste puncte se generează polilinia de sweeping notată Pol_{sw} (fig. 7.4.15). Pentru obținerea profilului se digitizează conturul unei secțiuni oarecare a piesei obținându-se un șir de puncte N_j cu $j=1, \dots, m$. Pentru acest șir de puncte se determină planul mediu (p_m) prin utilizarea algoritmului descris la

punctul 7.4.3.1.c.. Se determină intersecția dintre planul mediu (p_m) și polilinia de sweeping. Se notează cu M punctul de intersecție. Se determină orientarea segmentului polilinie de sweeping (parametrii directori l_p, m_p, n_p) pe care este situat punctul M și coeficienții directori ai planului (p_p) care este perpendicular pe dreapta având parametrii directori l_p, m_p, n_p .



7.4.14. Schema logică a algoritmului pentru generarea modelelor pieselor SC

Dacă punctul M coincide cu un vertex O_i al polilinie de sweeping atunci se consideră ca plan (p_p) planul bisector (p_{bi}). Planul bisector (p_{bi}) se determină cu algoritmul descris în anexa 6 pornind de la coordonatele vertexurilor O_{i-1}, O_i, O_{i+1} .

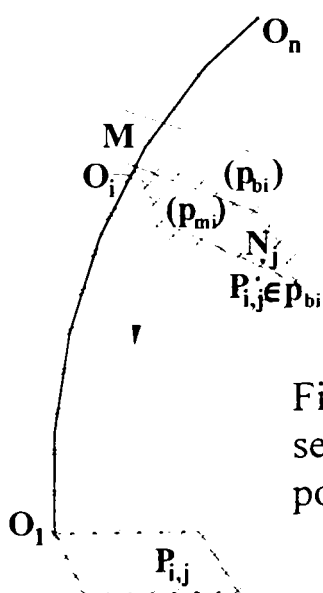
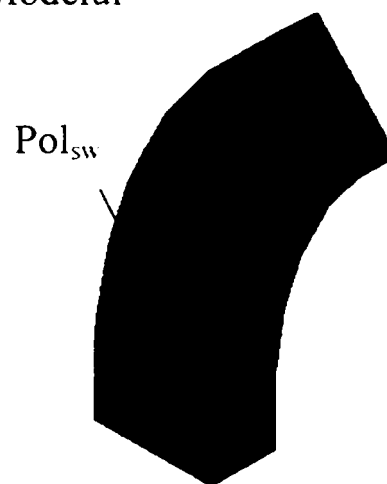


Fig. 7.4.15. Determinarea profilului secțiunii în primul vertex al polilinie de sweeping

Fig. 7.4.16. Modelul piesei SC



În cazul în care distanța dintre oricare două vertexuri succesive ale polilinieii de sweeping sunt situate la o distanță relativ mică (de exemplu sub 0.5 mm) atunci se poate considera ca plan (p_i) planul bisector corespunzător celui mai apropiat vertex al polilinieii de sweeping de punctul M. Se proiectează punctele N_j din șirul de puncte obținut prin digitizare pe planul (p_{bi}) și se obțin punctele $P_{i,j}$ cu $j=1, \dots, m$.

Se determină punctele $P_{1,j}$ care definesc conturul secțiunii în primul vertex al polilinieii de sweeping. Se utilizează un algoritm iterativ care, pornind de la coordonatele punctelor secțiunii corespunzătoare vertexului i al polilinieii de sweeping, determină coordonatele corespunzătoare secțiunii i-1. Punctul $P_{i-1,j}$ este punctul de intersecție a dreptei ce trece prin $P_{i,j}$ și are direcția paralelă cu segmentul $O_i O_{i-1}$ cu planul bisector corespunzător secțiunii i-1. Pentru generarea modelului de suprafață al piesei de tip SC se utilizează în continuare pașii care urmează după stabilirea punctelor secțiunii corespunzătoare primului vertex al polilinieii de sweeping în algoritmul descris la punctul 7.4.3.3. Se consideră secțiunea constantă de-a lungul întregii polilinii de sweeping. În fig. 7.4.16 se prezintă modelul de suprafață realizat în vedere umbrită.

7.4.3.5. Modelarea pieselor cu treceri între mai multe profile (TP)

Suprafața laterală a acestor piese se modelează prin sweeping generalizat în sensul că profilul variază în funcție de poziția pe polilinia de sweeping. Se notează cu Pol_{sw} polilina de sweeping. Vertexurile polilinieii Pol_{sw} pot fi situate pe o dreaptă sau pe o curbă strâmbă în spațiu.

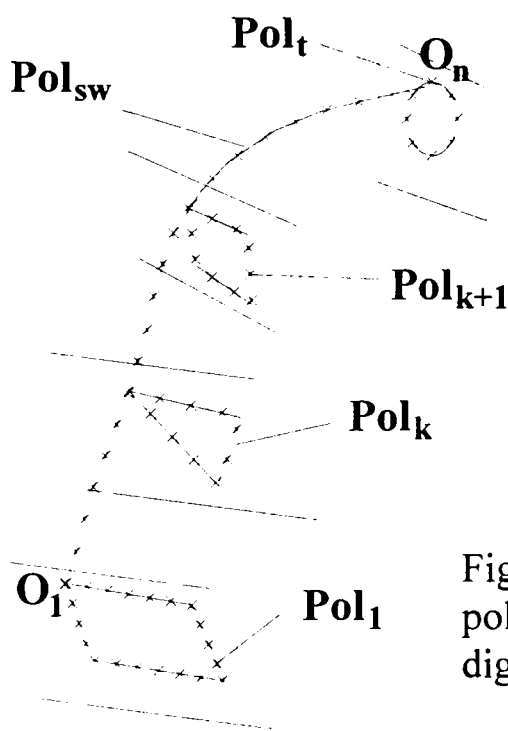


Fig. 7.4.17. Polilinia de sweeping și poliliniile corespunzătoare secțiunilor digitizate

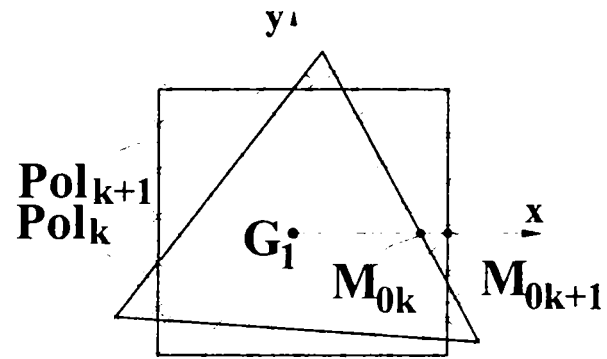
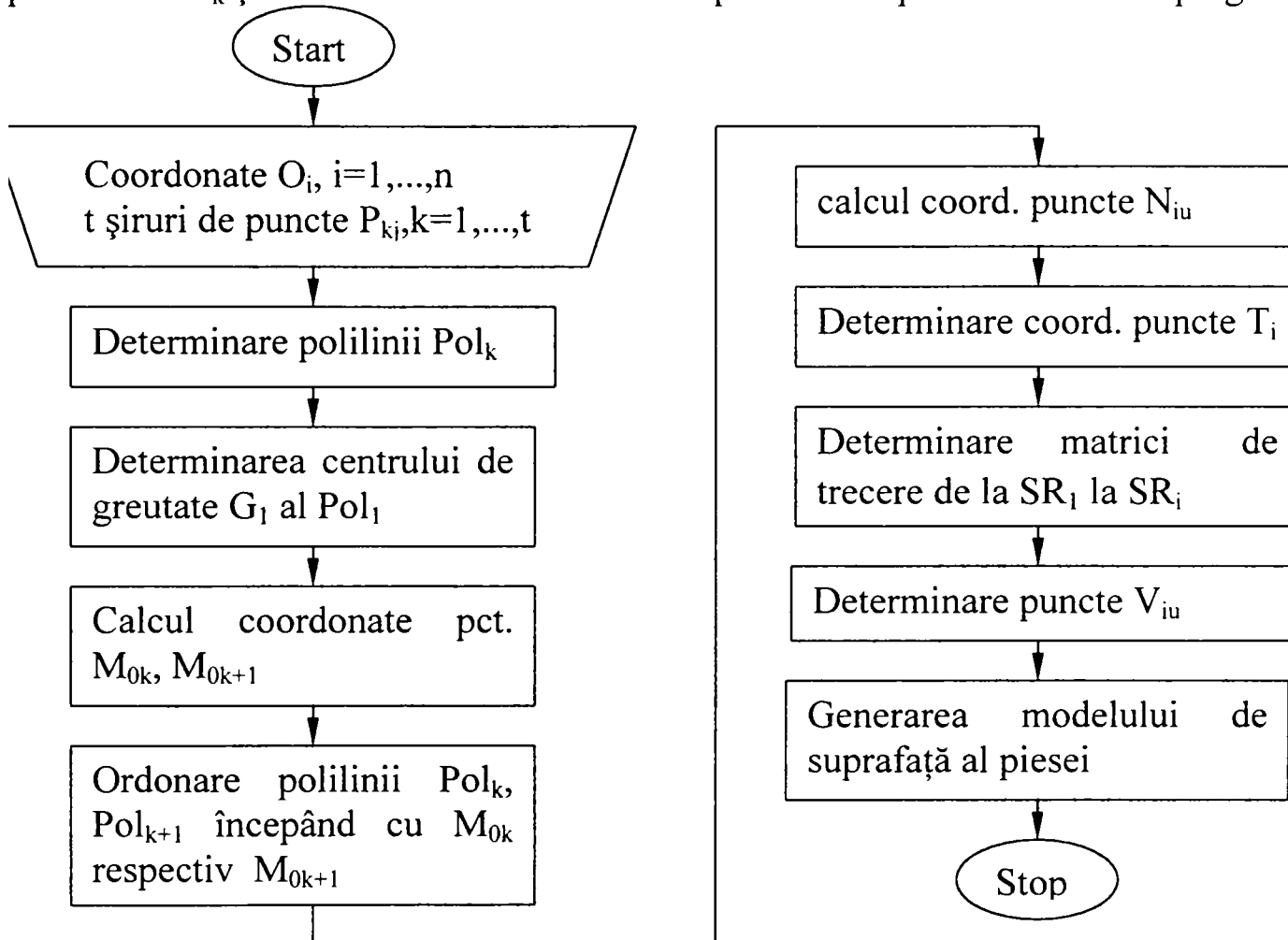


Fig. 7.4.18. Determinarea conturului secțiunilor în vertexul 1 al Pol_{sw}

Se prelevează prin digitizare coordonatele unui șir de puncte pe o muchie de-a lungul piesei. Din acest șir de puncte se generează polilinia de sweeping. Pentru mai multe secțiuni transversale prin piesă se prelevează cu ajutorul digitizorului 3D coordonatele punctelor situate pe suprafața piesei. Numărul de secțiuni pentru care se prelevează informațiile despre contur depinde de precizia cu care utilizatorul dorește să obțină

modelul piesei. Pentru fiecare șir de puncte reprezentând profilele în diferite secțiuni se obține o polilinie 2D notată Pol_k situată în planul care trece prin primul vertex al polilinieii de sweeping pe baza algoritmului prezentat la punctul 7.4.3.4. Se consideră că s-au digitizat t secțiuni prin piesă. Se consideră k indicele secțiunii digitizate, $k=1, \dots, t$. Pentru fiecare secțiune k digitizată se reține șirul de vertexuri P_{kj} al polilinieii Pol_k și indicele i al vertexului corespunzător al polilinieii de sweeping.



Fi . 7.4.19. Schema logică simplificată a algoritmului pentru generarea pieselo TP pornind de la informații obținute prin digitizare 3D

Se prezintă în continuare algoritmul pentru determinarea coordonatelor vertexurilor unei secțiuni corespunzătoare unui vertex oarecare i al polilinieii Pol_{sw} . Se determină pentru vertexul i al polilinieii Pol_{sw} între ce secțiuni digitizate se află. Fie aceste secțiuni cele cu indicii k și respectiv $k+1$ din cele t secțiuni digitizate. Conturul secțiunii corespunzătoare vertexului i al polilinieii Pol_{sw} se determină pornind de la poliliniile Pol_k și Pol_{k+1} (Fig. 7.4.17.). Se notează cu m_k numărul de vertexuri ale polilinieii Pol_k și cu m_{k+1} numărul de vertexuri al polilinieii Pol_{k+1} iar $m = \max(m_k, m_{k+1})$. Se determină centrul de greutate al poligonului având ca și colțuri vertexurile polilinieii Pol_k . Coordonatele centrului de greutate $G_k(x_{Gk}, y_{Gk})$ vor fi:

$$x_{Gk} = \frac{\sum_{j=1}^m x_{kj}}{m}; \quad y_{Gk} = \frac{\sum_{j=1}^m y_{kj}}{m} \tag{7.4.21}$$

Se determină coordonatele punctului de intersecție $M_{0k}(x_{0k}, y_{0k})$ dintre polilinia Pol_k și dreapta (d_g) care trece prin G_k și este paralelă cu axa Ox a sistemului de coordonate cu originea în O_1 și direcția axei Oz suprapusă peste O_1O_2 . De asemenea se determină vertexul $M_{0k+1}(x_{0k+1}, y_{0k+1})$ ca punct de intersecție între polilinia Pol_{k+1} și dreapta (d_g). Se determină sensul celor două polilinii. Se consideră polilinia în sens trigonometric dacă unghiul format de dreapta definită de punctul G_k cu vertexul cu indicele j al polilinieii față de axa Ox este mai mare decât unghiul corespunzător indicelui $j-1$. Această afirmație este validă în cazul în care poligonul definit de vertexurile polilinieii este convex, în caz contrar fiind necesară utilizarea unui algoritm suplimentar pentru detectarea sensului polilinieii. În cazul în care polilinia este în sensul acelor de ceasornic se inversează sensul acesteia. Inversarea semnifică faptul că vertexul cu indicele m_k devine vertexul cu indicele 1, cel cu indicele $m_k - 1$ devine 2, ș.a.m.d.. Se înserează vertexul M_{0k} în polilinia Pol_k respectiv vertexul M_{0k+1} în polilinia Pol_{k+1} apoi se reordonează vertexurile polilinieii astfel încât primul vertex să devină M_{0k} pentru Pol_k și M_{0k+1} pentru Pol_{k+1} (Fig. 7.4.18). Se creează câte m puncte pe fiecare din poliliniile Pol_k și respectiv Pol_{k+1} notate $N_{k,u}$ respectiv $N_{k+1,u}$ cu $u=1, \dots, m$. Pe fiecare polilinie, distanța între 2 puncte N succesive este egală. Se determină coordonatele unui punct $N_{i,u}$ corespunzător secțiunii ce trece prin vertexul cu indicele i al polilinieii Pol_{sw} inițiale (Fig 7.4.20). Se notează cu l_{sw} lungimea polilinieii Pol_{sw} între vertexurile corespunzătoare secțiunii k și respectiv $k+1$. Cu d_N se notează distanța dintre vertexurile $N_{k,u}$ și $N_{k+1,u}$. Lungimea polilinieii Pol_{sw} între vertexul corespunzător secțiunii k și vertexul cu indicele i se notează cu l_i . Distanța la care este situat vertexul $N_{i,u}$ de $N_{k,u}$ pe direcția $N_{k,u}N_{k+1,u}$ este :

$$d_i = \frac{l_i}{l_{sw}} \cdot d_n \tag{7.4.22}$$

Dacă se notează $\lambda = \frac{d_i}{d_N - d_i}$, coordonatele punctului $N_{i,u}$ sunt date de relațiile:

$$x_{i,u} = \frac{x_{k,u} + \lambda \cdot x_{k+1,u}}{1 + \lambda}; \quad y_{i,u} = \frac{y_{k,u} + \lambda \cdot y_{k+1,u}}{1 + \lambda} \tag{7.4.23}$$

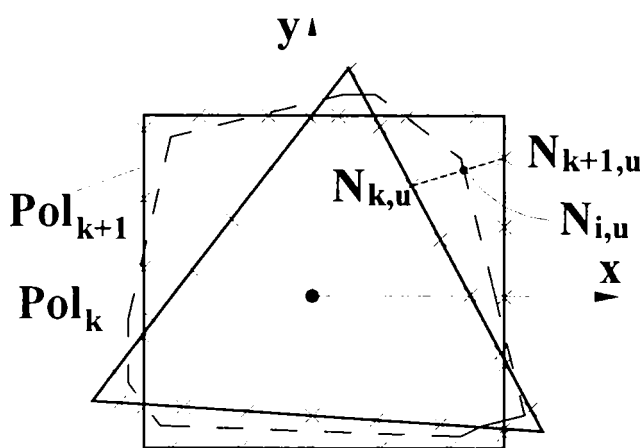


Fig. 7.4.20. Determinarea conturului unei secțiuni i în planul ce trece prin primul vertex al polilinieii Pol_{sw} .

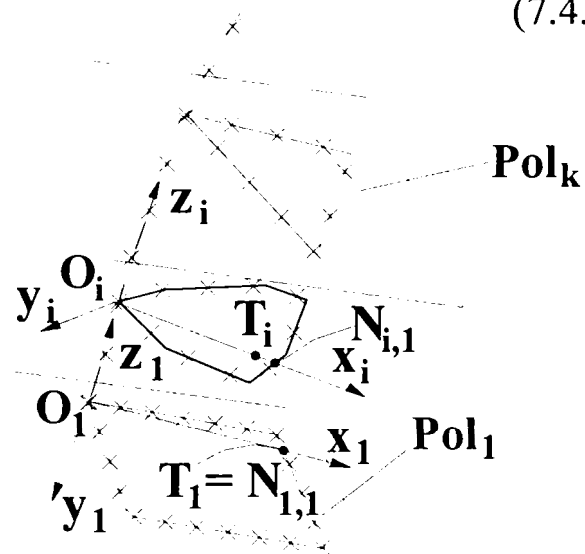


Fig. 7.4.21. Determinarea conturului unei secțiuni i în planul ce trece prin vertexul i al polilinieii Pol_{sw} .

Se repetă algoritmul prezentat anterior până la obținerea coordonatelor punctelor $N_{i,u}$ corespunzătoare tuturor vertexurilor polilinieii Pol_{sw} în planul ce trece prin primul vertex al polilinieii Pol_{sw} . Se consideră punctul $N_{1,1}$ și se determină punctele T_i corespunzătoare fiecărui vertex al polilinieii Pol_{sw} . Punctul $T_1=N_{1,1}$. Punctul T_i se determină ca punct de intersecție al dreptei ce trece prin punctul T_{i-1} și are direcția paralelă cu segmentul $O_{i-1}O_i$ al polilinieii Pol_{sw} cu planul bisector (p_{bi}). Se prezintă modul de determinare a coordonatelor punctelor $V_{i,u}$ corespunzătoare secțiunii i a piesei. Se determină matricea de situare a sistemului de referință SR_i cu originea în punctul O_i , cu axele Ox și Oy situate în planul bisector (p_{bi}) și cu axa Ox pe direcția dreptei O_iT_i . Se notează cu 1T_i matricea de situare a sistemului de referință SR_i față de SR_1 . Pentru determinarea punctelor $V_{i,u}$ se aplică matricea 1T_i punctelor $N_{i,u}$. Punctele $V_{i,u}$ sunt stocate într-o matrice cu două dimensiuni care este utilizată pentru generarea suprafeței laterale a modelului de suprafață al piesei. Cele două capace ale extremităților piesei se realizează cu comanda AutoCAD "Pface" pe baza punctelor $V_{1,u}$ respectiv $V_{n,u}$.

7.4.3.6. Modelarea pieselor ce se obțin prin extrudare cu contur drept (ECD)

Pentru modelarea acestui tip de piesă se pornește de la prelevarea coordonatelor șirurilor de puncte de pe conturul pieselor. Se începe cu digitizarea conturului exterior. Pentru concatenarea și închiderea poliliniilor 3D realizate din șirurile de puncte obținute prin digitizare se utilizează algoritmul descris la subpunctul 7.4.3.1.b). Din șirul de puncte aparținând conturului exterior se determină coeficienții A, B, C, D ai ecuației generale a planului mediu (p_m) prin utilizarea algoritmului descris la 7.4.3.1.c). Pe acest plan se proiectează toate șirurile de puncte reprezentând contururi interioare și se generează în sistemul CAD poliliniile 2D corespunzătoare. Pentru determinarea grosimii g a piesei se cere utilizatorului palparea unui punct $M_0(x_0, y_0, z_0)$ situat pe muchia piesei care nu e conținută în planul mediu (fig. 7.4.22). Se determină grosimea g ca distanță de la punctul M la planul (p_m):

$$g = \frac{|A \cdot x_0 + B \cdot y_0 + C \cdot z_0 + D|}{\sqrt{A^2 + B^2 + C^2}} \tag{7.4.24}$$

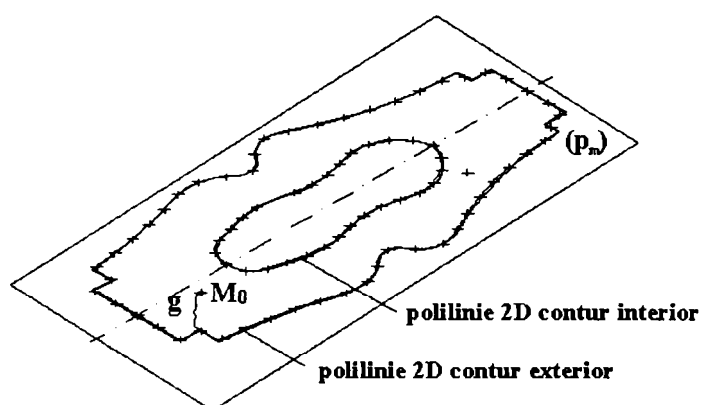


Fig. 7.4.22. Indicarea unui punct pentru determinarea grosimii piesei

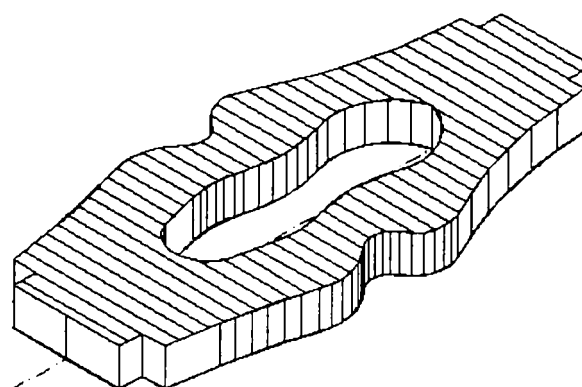


Fig. 7.4.23. Generarea modelului de suprafață al piesei

Suprafețele reprezentând canturile piesei se realizează ca suprafețe riglate prin utilizarea comenzii AutoCAD “Tabsurf”. Prin cantul piesei se înțeleg suprafețele verticale ale piesei. Capacele modelului de suprafață al piesei se obțin ca “Regiuni”. Regiunile sunt suprafețe plane cu sau fără zone interioare, delimitate de polilinii 2D închise. Se realizează regiunea reprezentând suprafața capac din planul (p_m). Apoi se copiază această regiune la o distanță g pe o direcție perpendiculară pe planul (p_m) în semispațiul care conține punctul M_0 . În figura 7.4.23. este prezentat modelul piesei.

7.4.3.7. Modelarea pieselor obținute prin extrudare cu contur profilat (ECP)

Modelarea prin extrudare cu contur profilat necesită realizarea poliliniilor 2D închise pe baza coordonatelor punctelor de pe muchiile de contur exterior și eventual muchiile interioare așa cum este prezentat în figura 7.4.22. de la punctul 7.4.3.6. Pentru fiecare din poliliniile de contur pentru care cantul piesei este profilat se realizează prelevarea informațiilor referitoare la profil. Se exemplifică modul de obținere a suprafeței cantului exterior al piesei. Pentru canturile interioare algoritmul este aproximativ același.

Profilul piesei este reprezentat cu ajutorul unei polilinii deschise (Pol_p) situată într-un plan (p_p) perpendicular pe planul mediu (p_m). Utilizatorul palpează cu vârful digitizorului 3D un șir de puncte situat pe suprafața cantului piesei, pe cit posibil într-un plan perpendicular pe planul mediu. Punctul din șir care este cel mai apropiat de un vertex al polilinieii de contur Pol_{ext} se deplasează în acel vertex. Fie acesta vertexul V_i al polilinieii Pol_{ext} , unde i reprezintă indexul vertexului în cadrul polilinieii. Cu ajutorul vertexurilor V_{i-1} , V_i și V_{i+1} se determină planul (p_p) perpendicular pe planul mediu (p_m) ca plan bisector al unghiului diedru format de planele create din dreptelor concurente $V_i V_{i-1}$ și $V_{i+1} V_i$ și perpendiculara lor comună. Algoritmul pentru determinarea planului bisector este descris în anexa 6.

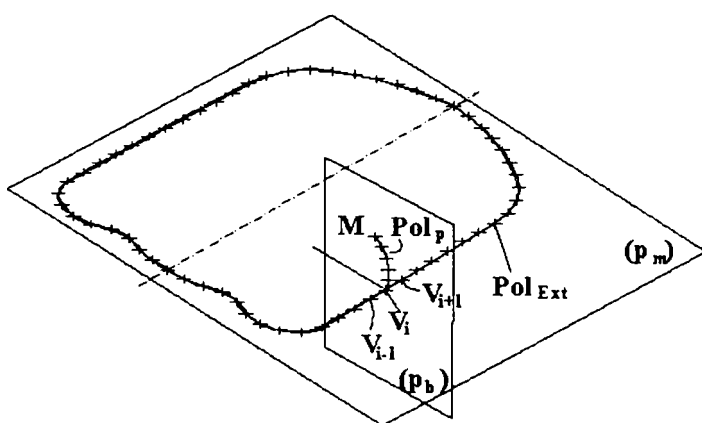


Fig. 7.4.24. Indicarea profilului cantului piesei

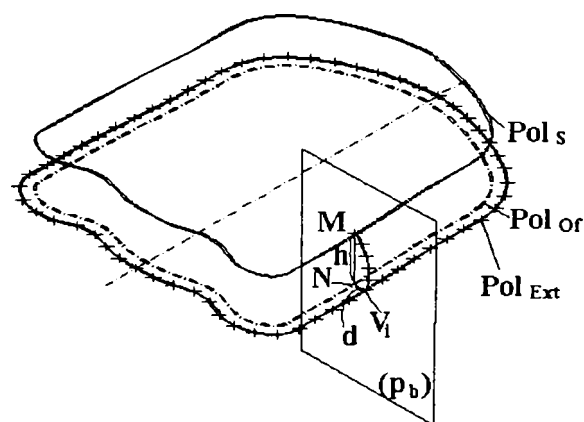


Fig. 7.4.25. Crearea muchiei superioare

Toate punctele din șirul pentru profil se proiectează pe planul (p_p) obținându-se polilina Pol_p (fig. 7.4.24.). Se notează cu M vertexul care aparține polilinieii Pol_p și este situat la capătul opus al polilinieii față de V_i . Proiecția punctului M pe planul (p_m) se notează cu N . Se calculează distanța d între vertexul V_i și punctul N și distanța h

dintre punctele M și N. Se generează polilinia Pol_{Of} situată la o distanță offset egală cu d față de polilinia Pol_{ext} (Fig. 7.4.25.). Pentru poliliniile care reprezintă conturul exterior al piesei polilinia obținută prin offset se află în interiorul conturului iar pentru contururile interioare se situează în exterior. Se deplasează polilinia Pol_{Of} cu distanța h pe direcția NM și se notează cu Pol_s . Polilinia Pol_{ext} se întrerupe în vertexul V_i iar polilinia Pol_s în vertexul M obținându-se 2 polilinii deschise. Se realizează o copie Pol_p' a polilinieii Pol_p suprapusă peste aceasta. Suprafața care reprezintă cantul exterior al piesei se generează ca suprafață Coons delimitată de patru muchii: Pol_{Ext} , Pol_p , Pol_s și Pol_p' . Pentru generarea suprafeței Coons se utilizează comanda AutoCAD "Edgesurf". Suprafețele capac se obțin în mod asemănător modelării ECD (punctul 7.4.3.6).

7.4.3.8. Modelarea pieselor din placaj curbat cu contururi interioare (PCCI)

Acest tip de piese sunt realizate prin curbarea unui material de grosime constantă. O astfel de piesă și poliliniile obținute prin digitizare 3D sunt prezentate în figura 7.4.26.. Pentru generarea modelului de suprafață se pornește de la digitizarea conturului exterior al piesei. Se obțin prin digitizare 4 șiruri de puncte din care se generează poliliniile $Pol_{e1}, \dots, Pol_{e4}$. Deoarece aceste polilinii nu se intersectează într-un punct se utilizează un algoritm pentru a crea puncte de intersecție a două polilinii succesive obținându-se vertexurile C_1, C_2, C_3, C_4 . De exemplu, vertexul C_1 devine primul vertex al polilinieii Pol_{e1} și ultimul vertex al polilinieii Pol_{e4} . Se utilizează comanda AutoCAD "Edgesurf" pentru a genera suprafața Coons delimitată de 4 muchii. Se notează această suprafață cu S_1 .

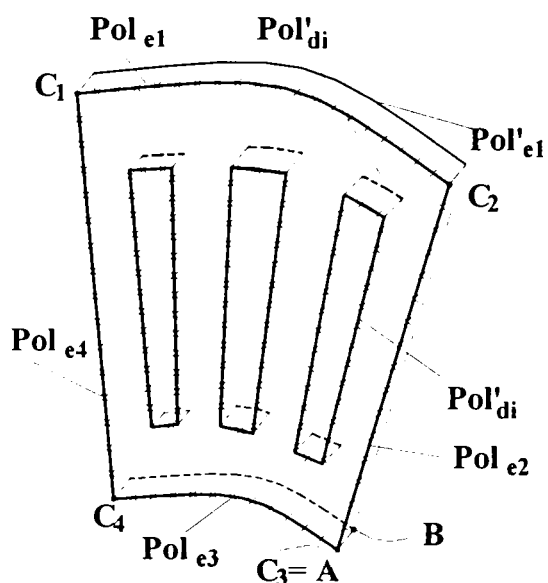


Fig. 7.4.26. Modelarea pieselor PCCI

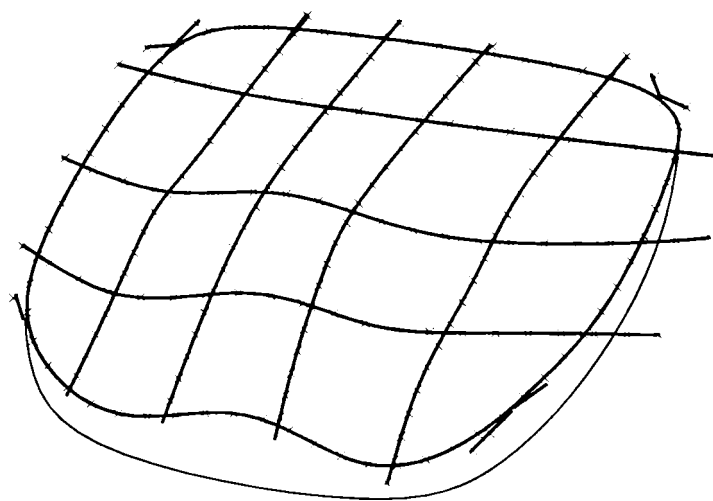


Fig. 7.4.27. Prelevarea poliliniilor 3D pentru generarea suprafeței superioare a piesei

Pentru a obține decupajele se digitizează muchii interioare obținându-se poliliniile Pol_{di} . Dacă se notează cu n numărul decupajelor interioare, atunci $i=1, \dots, n$. Pentru indicarea grosimii piesei și a direcției pentru offset se indică două puncte A și B pe extremitățile unei muchii de cant. Se notează cu g distanța între punctele A și B . Se

decupează din suprafața S_1 suprafețele delimitate de proiecția poliliniilor Pol_{di} închise pe S_1 . Suprafața S_1' , situată de cealaltă parte a piesei se obține prin offset a suprafeței S_1 . Suprafețele de legătură între S_1 și S_2 se obțin ca suprafețe riglate între poliliniile Pol_e și Pol_e' respectiv Pol_{di} și Pol_{di}' prin utilizarea comenzii "Tabsurf" din AutoCAD.

7.4.3.9. Modelarea pieselor cu mulaj complex (PMC)

Din această categorie fac parte de exemplu șezuturile stratificate cu formă anatomică. Suprafața superioară a piesei (Fig 7.4.27.) este o suprafață strâmbă în spațiu. Deoarece generarea suprafeței prin utilizarea doar a muchiilor de contur nu oferă o aproximare suficient de bună a suprafeței reale, este necesar palparea unui carioaj de curbe pe suprafața piesei. Pentru generarea modelului de suprafață pornind de la un carioaj de curbe (aproximate prin polilinii 3D) se utilizează algoritmul prezentat la subcapitolul 5.4. ("Modelarea suprafețelor pornind de la șiruri de puncte obținute prin digitizare"). Pentru generarea celorlalte suprafețe ale piesei se utilizează metodele prezentate în subpunctele anterioare.

7.4.4. Generarea automată a desenelor de execuție a șabloanelor și a sculelor pentru realizarea pieselor de revoluție profilate pornind de la date obținute prin digitizare

7.4.4.1. Introducere

Piese de revoluție profilate în industria mobilei se realizează în special pe strunguri automate sau pe strunguri cu comandă numerică. În cazul unor serii suficient de mari pentru acoperirea costurilor unor scule speciale, piesele de revoluție profilate se execută pe strunguri automate, unul din motivele acestei opțiuni fiind productivitatea ridicată și existența unor astfel de utilaje în majoritatea firmelor producătoare de mobilă. În figura 7.4.28. este prezentată o piesă care intră în componența mobilierului de ședere.

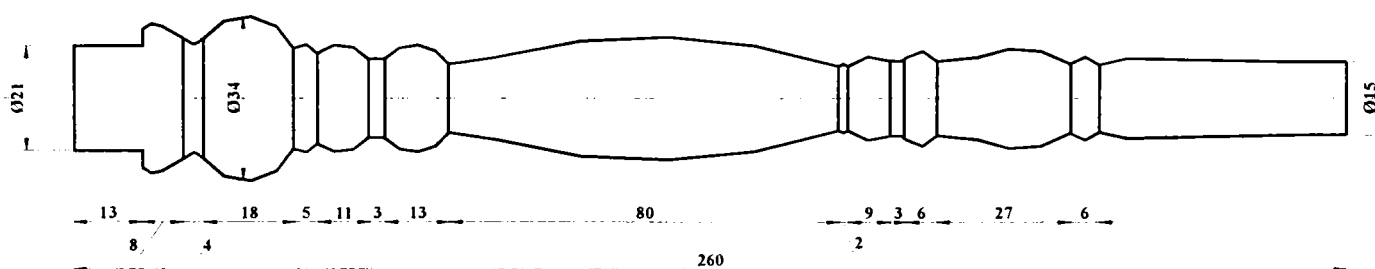


Fig.7.4.28. Piesă de revoluție profilată

Executarea pieselor se face prin strunjire pornind de la un semifabricat de secțiune pătrată. Așchierea se realizează într-o singură trecere cu mai multe scule (Fig. 7.4.29). Prima sculă (sc.1) realizează o prelucrare în urma căreia secțiunea semifabricatului

devine circulară. Această sculă este urmată de un cuțit (sc.2) care produce o aşchiere în urma căreia se obține o formă apropiată de cea a piesei finale.

Deplasarea cuțitului sc.2 este ghidată pe direcție transversală față de piesă prin intermediul unei role aflate în contact cu un șablon de copiere. Șablonul de copiere are rolul unei came iar rola îndeplinește sarcina tachetului. Definitivarea profilului se realizează cu ajutorul cuțitelor profilate.

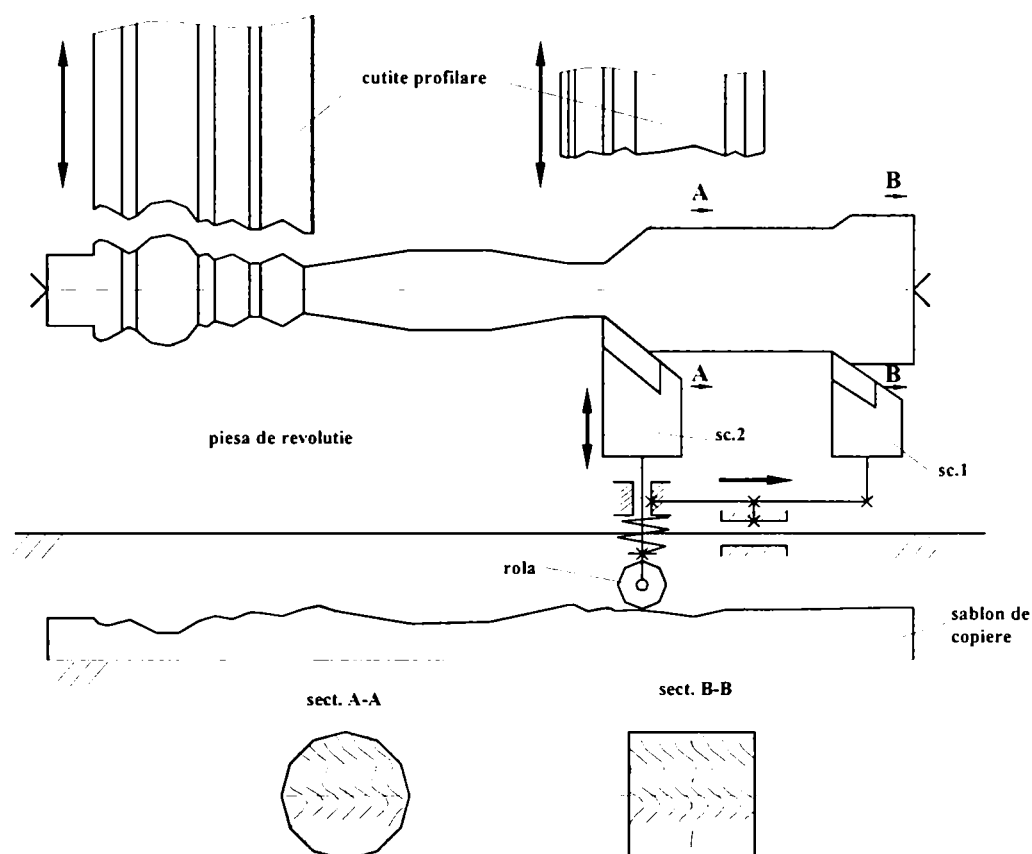


Fig.7.4.29 Tehnologia de execuție a pieselor de revoluție profilate pe strunguri automate

Modalitatea actuală de realizare a desenelor de execuție ale sculelor presupune parcurgerea următoarelor faze: strunjirea manuală a unei piese cât mai apropiate ca formă și dimensiuni cu piesa modelului, secționarea longitudinală a piesei, realizarea desenului de execuție al piesei utilizând piesa secționată, desenarea șablonului de copiere prin executarea racordărilor pe profilul piesei cu raze egale cu raza rolei, identificarea zonelor care nu sunt atinse de rolă și realizarea desenelor de execuție ale cuțitelor de profilare. Sculele (sc.1) și (sc.2) sunt generale și sunt utilizate la toate piesele. Șabloanele de copiere și cuțitele profilate sunt specifice fiecărei piese, și pentru acestea autorul a elaborat un program de realizare automată a modelului 3D și a desenelor de execuție.

7.4.4.2. Descrierea algoritmului soluției propuse de autor

Soluția propusă pentru realizarea desenelor de execuție ale șablonului de copiere și ale cuțitelor profilate presupune parcurgerea următoarelor etape: digitizarea modelului fizic al piesei de revoluție, simularea ghidării cuțitului de tip sc.2 prin intermediul contactului dintre rolă și șablonul de copiere, determinarea profilului

șablonului de copiere ca înfășurătoare a pozițiilor succesive ale rolei și realizarea desenului de execuție al șablonului, determinarea zonelor care nu pot fi finalizate de cuțitul sc.2 și realizarea desenelor de execuție și ale modelor 3D ale cuțitelor profilate. Pentru implementarea realizării automate a etapelor enumerate anterior s-au utilizat limbajele de programare Visual Lisp și Visual Basic în mediul AutoCAD 2000.

Polilinia 2D care reprezintă profilul piesei se obține cu ajutorul unui digitizor 3D prin metoda descrisă la punctul 7.4.3.2. (“Modelarea pieselor de revoluție cu secțiune variabilă”). Înregistrarea punctelor 3D cu o rezoluție de 0.2 mm oferă o precizie suficientă scopului propus. În urma obținerii polilinieii 2D, coordonatele acesteia sunt înregistrate într-un fișier text precizat de utilizator.

Coordonatele punctelor aparținătoare profilului piesei sunt preluate din fișierul text și se generează o polilinie 2D într-un nou document AutoCAD în care se va realiza generarea desenelor de execuție a sculelor și șablonului. În continuare se simulează rostogolirea rolei pe acest profil (polilinie 2D). Pentru fiecare vertex al polilinieii se creează un cerc reprezentând rola, tangent la linia orizontală care trece prin vertexul polilinieii în semiplanul superior (fig.3). La o anumită poziție a rolei, pentru fiecare din vertexurile polilinieii care au coordonata x în intervalul $(x_0 - r, x_0 + r)$ se determină distanța h_i minimă cu care trebuie deplasată rola în sensul crescător al axei Oy astfel încât să nu existe nici un vertex al polilinieii ce reprezintă profilul piesei în interiorul cercului. Rola este reprezentată prin cercul de centru $O(x_0, y_0)$ și raza de lungime r, deci $r^2 = (x_0 - x_2)^2 + (y_0 - y_2)^2$. Punctul $P_1(x_1, y_1)$ aparține polilinieii care reprezintă conturul piesei și este cunoscut. Distanța h cu care trebuie deplasat cercul este :

$$h = \|MN\| = \|ON\| - \|OM\| = \sqrt{r^2 - (x_0 - x_2)^2} - \sqrt{l^2 - (x_0 - x_1)^2}$$

Punctele P_1 și P_2 aparțin dreptei verticale (d_v) deci $x_2 = x_1$. Distanța cu care se deplasează rola este valoarea maximă a distanței h pentru vertexurile din intervalul $(x_0 - r, x_0 + r)$.

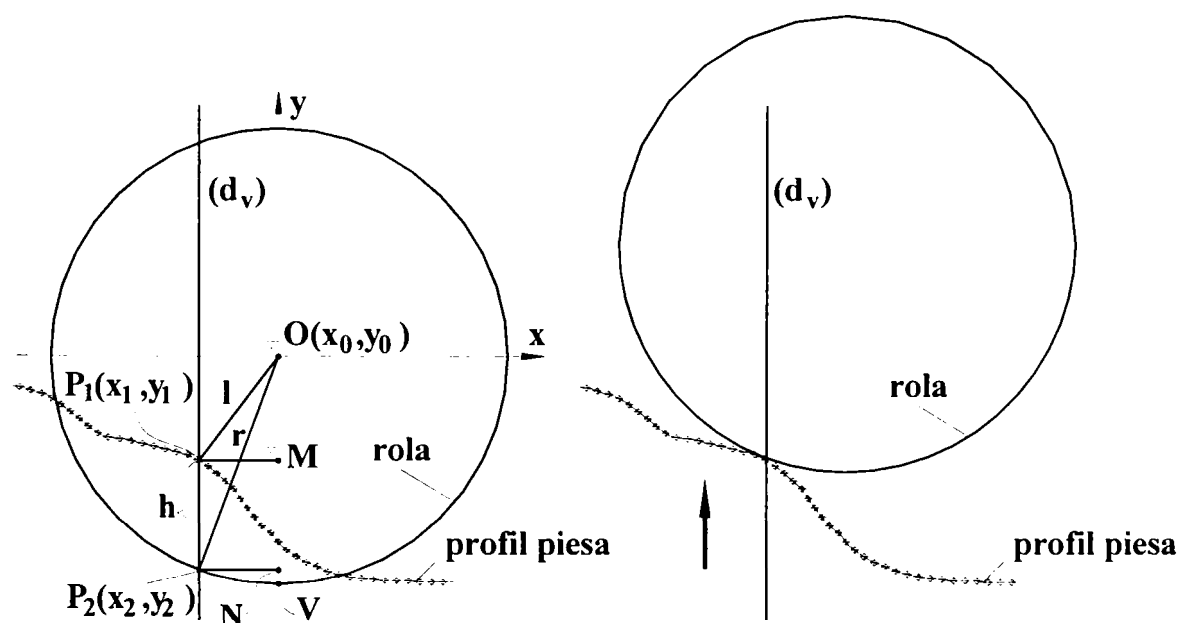


Fig. 7.4.30.. Determinarea poziției momentane a rolei și deplasarea acesteia pe verticală

În urma determinării pozițiilor succesive ale rolei în timpul rostogolirii pe profilul piesei se determină profilul șablonului de copiere ca înfășurătoare a cercurilor reprezentând rola (fig. 7.4.31.).

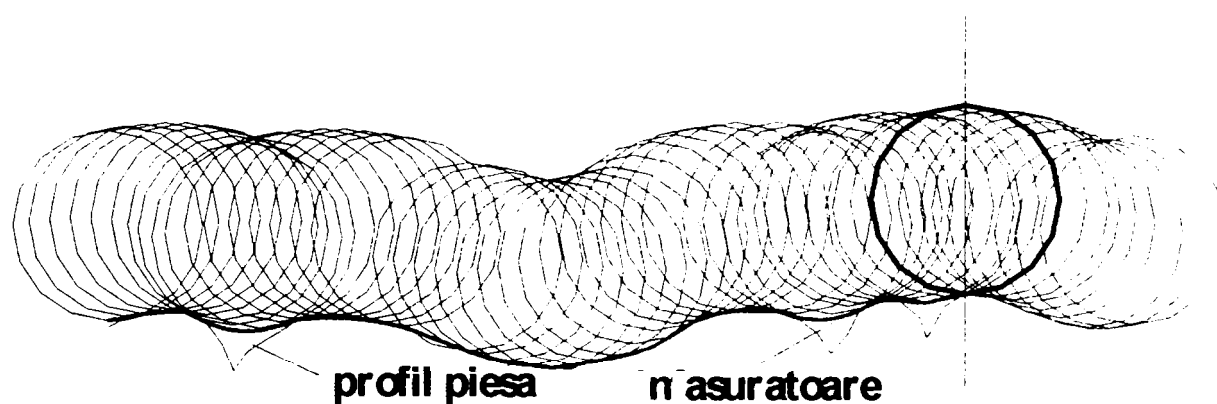


Fig. 7.4.31. Determinarea profilului șablonului de copiere

Algoritmul pentru determinarea înfășurătoarei constă în parcurgerea următoarelor faze: desenarea unei drepte verticale prin fiecare vertex al poliliniei, determinarea cercurilor intersectate de dreaptă, determinarea punctelor de intersecție dintre dreaptă și fiecare cerc și reținerea punctului cu coordonata y minimă ca vertex al poliliniei ce reprezintă înfășurătoarea. Polilinia înfășurătoare este utilizată pentru realizarea desenului de execuție și a modelului 3D al șablonului de copiere (fig. 5). Coordonatele vertexurilor succesive ale poliliniei pentru care $h > 0$ se stochează într-un fișier text în vederea utilizării la generarea cuțitelor profilate, ca segmente de polilinie.

Nu va fi executat un cuțit profilat pentru fiecare din aceste segmente de polilinie. Mai multe astfel de zone alăturate ale piesei vor fi profilate cu același cuțit dacă diferența dintre valoarea coordonatei x a primului vertex al segmentului și coordonata x a ultimului vertex al segmentului alăturat se încadrează în valoarea maximă prescrisă pentru lățimea cuțitului de profilare (fig. 7.4.33.).

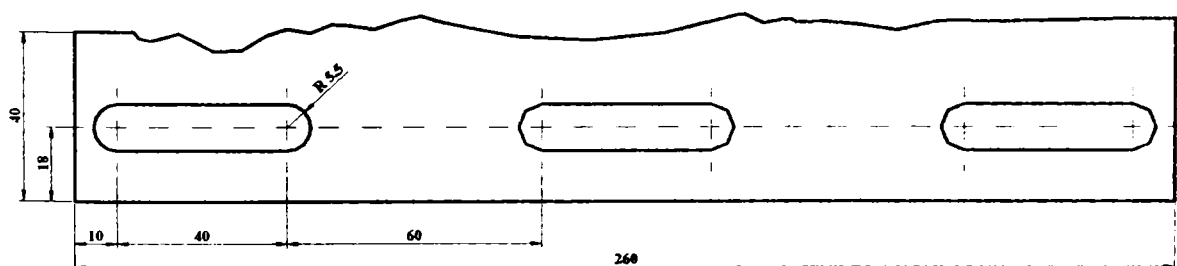


Fig. 7.4.32. Desenul de execuție al șablonului de copiere pentru o piesă

În urma concatenării, profilele rezultate sunt transferate ca parametri unui program Visual Lisp care generează desenele de execuție și modelul 3D al cuțitelor de profilare. Un model pentru un asemenea cuțit este prezentat în figura 7.4.34.. În funcție de adâncimea profilului (diferența între valoarea y min și y max a vertexurilor profilului sculei) programul determină automat grosimea semifabricatului necesar

pentru realizarea cuțitului. În funcție de lățimea cuțitului numărul de orificii utilizate pentru fixarea în portcuțit diferă și este determinat în mod automat de către program.

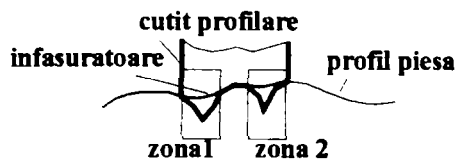


Fig.7.4.33. Concatenarea segmentelor de profil alăturate pentru formarea unui sigur cuțit de profilare

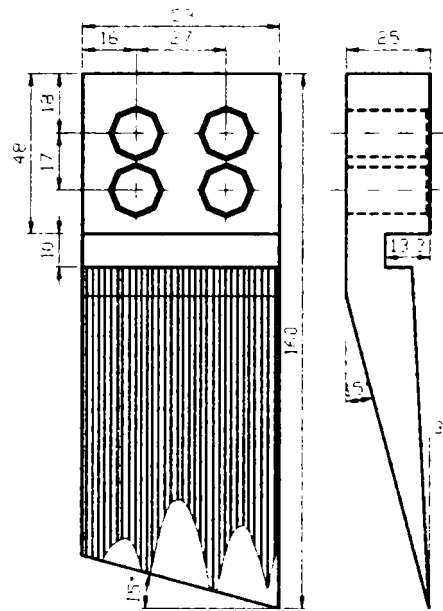
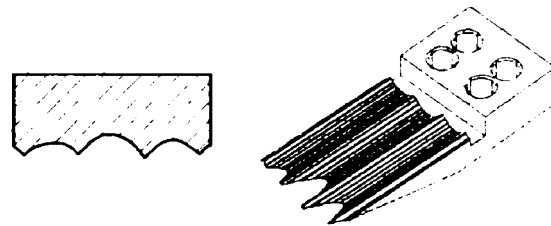


Fig. 7.4.34. Desenul de execuție și modelul 3D pentru un cuțit de profilare



7.4.4.3. Dezvoltări posibile ale aplicației

Deocamdată cotarea secțiunii cuțitului rămâne să fie realizată de către operator, dar prin implementarea regulilor de cotare programul poate fi dezvoltat să realizeze întreaga cotare în mod automat. În cazul în care în întreprindere există mașini de prelucrare cu comandă numerică, pe baza modelelor în format electronic se pot elabora cu ușurință programele pentru realizarea fizică a șabloanelor și sculelor. Realizarea unei baze de date cu profile de piese și cuțite de profilare ar permite utilizarea unor algoritmi de căutare a similitudinii unui profil la piese realizate anterior și pentru care deja sunt realizate sculele necesare prelucrării.

7.5. Sculptură virtuală cu digitizorul 3D

7.5.1. Introducere

Sculptura virtuală constă în realizarea unor modele 3D ale unor obiecte, pornind de la un semifabricat virtual prin “îndepărtarea” de material cu ajutorul unei scule virtuale. Modelul virtual poate fi materializat prin diferite metode, de exemplu prin aşchiere sau realizare rapidă de prototipuri.

Ca aplicații ale sculpturii virtuale se pot menționa: realizarea proiectării formei unor produse cu suprafețe complexe, opere de artă virtuale, divertisment, realizarea unor modele geometrice de suprafață a unor obiecte pentru prezentarea pe Internet, modelarea pentru vizualizare a rezultatului previzibil al unor operații de chirurgie plastică.

Aplicația prezentată de autor are scopul de a permite realizarea unor modele de suprafață prin deformarea unui semifabricat virtual inițial. Aplicația, de concepție proprie a autorului este denumită *DigiSculpt*. Deplasarea sculei virtuale se realizează cu ajutorul digitizorului 3D acționat prin conducere manuală. Ca digitizoare 3D se pot utiliza ambele tipuri de digitizoare prezentate în capitolul 6. Digitizorul cu fire oferă o flexibilitate sporită prin posibilitatea de indicare a orientării dreptei caracteristice și posibilitatea utilizării unor scule virtuale care au o axă de simetrie.

În cadrul subcapitolului se prezintă modul de generare a semifabricatelor virtuale inițiale și algoritmul pentru îndepărtarea de material din semifabricatul virtual cu ajutorul sculei virtuale conduse cu digitizorul 3D

7.5.2. Generarea semifabricatelor inițiale

7.5.2.1. Introducere

Aplicația este concepută să lucreze cu modele de suprafață fațetate. Utilizarea modelelor fațetate a fost aleasă din mai multe considerente: este posibil exportul facil al modelului în format VRML (Virtual Reality Modelling Language), renderingul suprafețelor fațetate este implementat hardware și este facilitată implementarea unor algoritmi pentru deformarea suprafețelor în timp real.

Pentru a permite operarea în timp real se utilizează o structură de elemente volumice, în care sunt încadrate fațetele. Această structură permite determinarea rapidă a fațetelor semifabricatului care sunt în interiorul sculei virtuale în vederea eliminării lor și a relimitării suprafeței semifabricatului. Volumul ocupat de structura de elemente volumice trebuie să cuprindă gabaritul modelului piesei sau semifabricatului în orice fază a modelării. Deoarece este permisă și adăugarea de material este posibil ca gabaritul piesei rezultate în urma modelării să fie mai mare decât gabaritul semifabricatului. În figura 7.5.1. se prezintă notațiile utilizate pentru structurile de elemente volumice. Elementele volumice sunt de formă cubică având latura egală cu pasul de divizare. Pentru definirea structurii de elemente volumice se precizează punctul $O_0(x_0, y_0, z_0)$, pasul de divizare p_d și numărul de elemente volumice pe fiecare din direcțiile axelor de coordonate. Se notează cu l numărul de elemente

volumice pe direcția axei O_0x , cu m numărul de elemente volumice pe direcția axei O_0y și respectiv cu O_0z numărul de elemente volumice pe direcția axei O_0z . Un anumit element volumic v se referă prin cei trei indecși ai săi $v(i,j,k)$ cu $i=1\div l$, $j=1\div m$, $k=1\div n$. Dacă fiecărui element volumic i s-ar asocia o listă de fețe 3D care sunt cuprinse în elementul volumic respectiv, consumul de memorie ocupat de matricea tridimensională care stochează structura de elemente volumice ar fi foarte mare. Deoarece se lucrează cu model de suprafață, un mare număr de elemente volumice nu ar avea în interior fețe 3D. Pentru optimizarea consumului de memorie, se consideră pentru stocarea fețelor 3D o structură bidimensională. Toate fețele care sunt cuprinse în elementele volumice cu primii 2 indecși constanți și indexul $k=1\div n$ se păstrează în aceeași listă. O listă $Lst(i,j)$ stochează informații despre fețele cuprinse într-un volum paralelipedric având ca și colțuri diametral opuse punctele $P_1(x_{i-1}, y_{j-1}, z_1)$ și $P_2(x_i, y_j, z_n)$.

Ca semifabricate se pot utiliza suprafețe sferice, cilindrice, paralelipedice, tronconice, etc. Se descriu algoritmi pentru generarea fațetelor în cazul semifabricatelor de formă sferică și cilindrică.

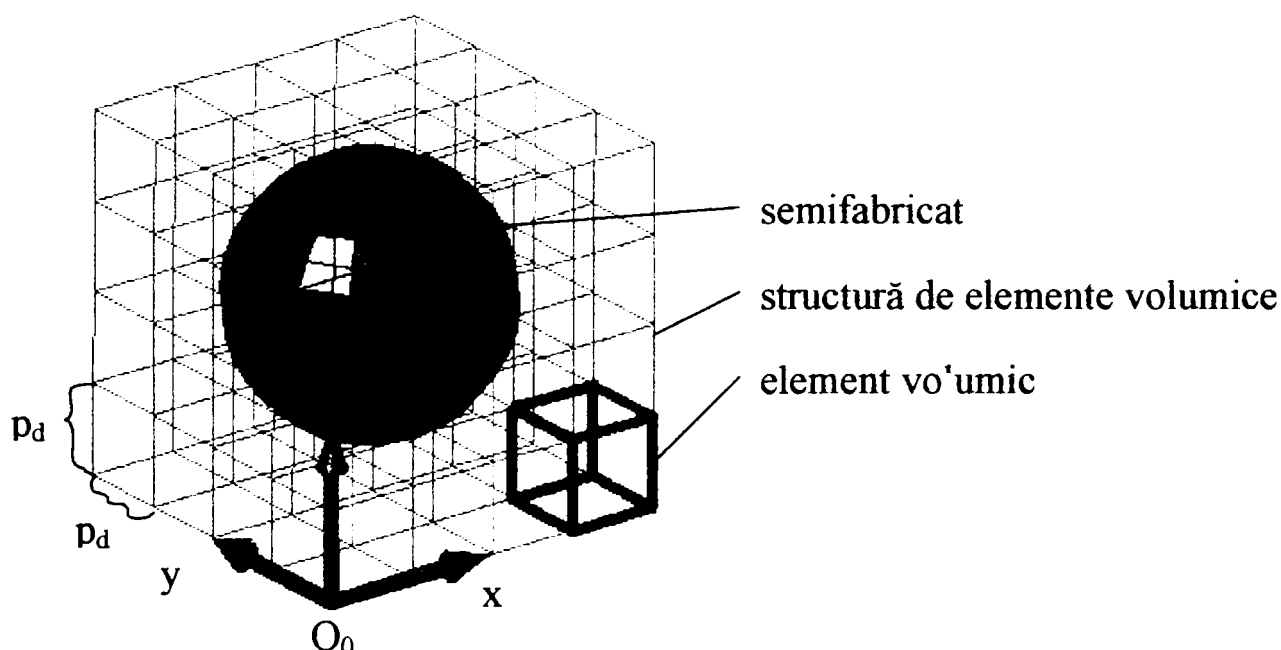


Fig. 7.5.1. Notățiile utilizate pentru structura de elemente volumice

7.5.2.2. Generarea semifabricatului virtual sferic

Se consideră o sferă definită prin centrul $C_{sf}(x_{sf}, y_{sf}, z_{sf})$ și raza r_{sf} . Sunt indicați parametrii structurii de elemente volumice astfel încât modelul piesei finale să se încadreze în volumul ocupat de structură. Se urmărește aproximarea suprafeței sferice prin fațete 3D, și stocarea informațiilor despre aceste fațete în structura matricială. Pentru determinarea coordonatelor fețelor 3D sunt determinate intersecțiile sferei cu drepte paralele cu axele de coordonate, aflate la distanța p_d una de alta. Se notează cu (Dx_{jk}) , (Dy_{ik}) , (Dz_{ij}) dreptele paralele cu O_0x , O_0y și respectiv O_0z . Pentru determinarea punctelor de intersecție dintre sferă și dreptele (Dx_{jk}) și (Dy_{ik}) se determină cercurile rezultate prin intersecția sferei cu plane paralele cu xO_0y la

distanțe $k \cdot d$ de planul xO_0y . În figura 7.5.2. se prezintă determinarea centrului și razei corespunzătoare unui plan paralel cu xO_0y .

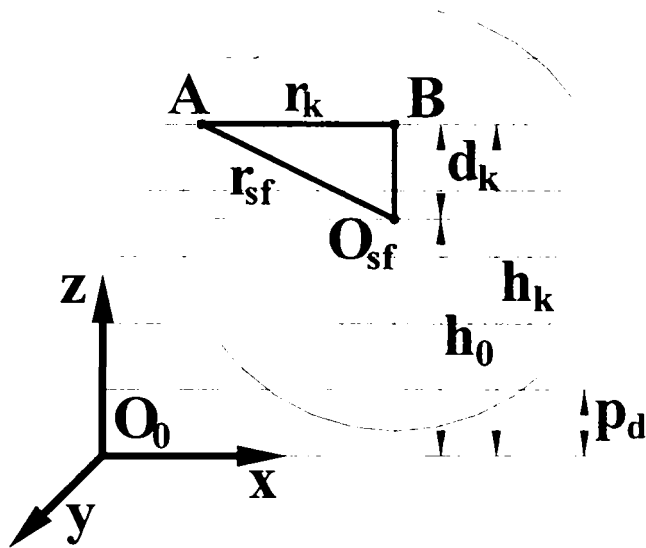


Fig.7.5.2. Determinarea cercului rezultat prin intersecția sferei cu un plan

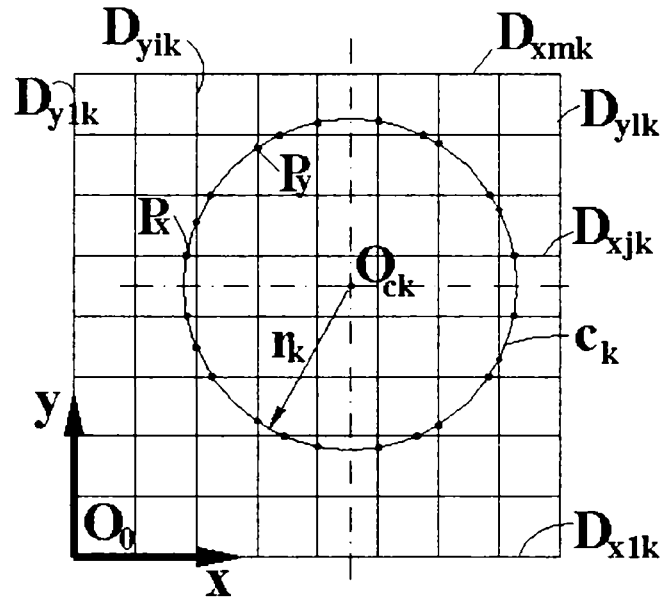


Fig.7.5.3. Determinarea intersecției sferei cu dreptele (D_{xjk}) și (D_{yik})

Distanța dintre centrul O_{sf} al sferei și planul situat la distanța $h_k = p_d \cdot k$ de planul xO_0y se notează cu d_k și are valoarea :

$$d_k = |h_k - h_0| \tag{7.5.1}$$

Din triunghiul dreptunghic $A O_{sf} B$ rezultă valoarea r_k a razei cercului c_k a unei secțiuni paralele cu planul xO_0y la distanța h_k de acesta:

$$r_k = \sqrt{r_{sf}^2 - d_k^2} \tag{7.5.2}$$

Centrul cercului c_k are coordonatele x și y egale cu ale centrului sferei și coordonata z egală cu valoarea h_k deci $O_{ck}(x_{sf}, y_{sf}, h_k)$.

Se determină coordonatele punctelor notate generic $P_x (x_{Px}, y_{Px}, z_{Px})$ de intersecție a cercului cu dreptele (D_{xjk}) :

$$\begin{cases} x_{Px} = p_d \cdot j \\ y_{Px} = y_{ck} \pm \sqrt{r_{ck}^2 - (x_{Px} - x_{ck})^2} \\ z_{Px} = h_k \end{cases} \tag{7.5.3}$$

Coordonatele punctelor P_y rezultate prin intersecția cercului cu dreapta (D_{yik}) se determină în mod analog:

$$\begin{cases} x_{Py} = p_d \cdot i \\ y_{Py} = y_{ck} \pm \sqrt{r_{ck}^2 - (y_{Py} - y_{ck})^2} \\ z_{Py} = h_k \end{cases} \tag{7.5.4}$$

Pentru fiecare element de volum având înălțimea $h_n = p_d \cdot n$ și ca și colțuri diametral opuse punctele $P_1(x_{i-1}, y_{j-1}, z_1)$ și $P_2(x_i, y_j, z_n)$ se generează o listă $L_P(i,j)$ în care se

stochează punctele P_x și P_y . Un punct oarecare de coordonate x_p, y_p se stochează în lista $L_P(i,j)$ dacă sunt îndeplinite simultan condițiile :

$$\begin{cases} p_d \cdot (i-1) \leq x_p \leq p_d \cdot i \\ p_d \cdot (j-1) \leq y_p \leq p_d \cdot j \end{cases} \quad (7.5.5)$$

Pentru determinarea punctelor P_z de intersecție a sferei cu dreptele verticale (Dz_{ij}), paralele cu axa O_0z a sistemului de coordonate se procedează în mod analog cazului dreptelor paralele cu O_{0x} și O_{0y} . Notațiile sunt prezentate în figura 7.5.4. Se determină raza r_i a cercului c_i rezultat prin intersecția sferei cu planul paralel cu yO_0z situat la distanța $h_i = p_d \cdot i$ de acesta:

$$r_i = \sqrt{r_{sf}^2 - |h_i - x_{ci}|^2} \quad (7.5.6)$$

Punctele de intersecție a cercului c_i cu dreapta D_{vij} au coordonatele :

$$\begin{cases} x_{Pz} = p_d \cdot i \\ y_{Pz} = p_d \cdot j \\ z_{Pz} = z_{ck} \pm \sqrt{r_{ci}^2 - (x_{px} - x_{ck})^2} \end{cases} \quad (7.5.7)$$

Coordonatele punctelor P_z se stochează în listele $L_P(i,j)$ pe baza condițiilor (7.5.5).

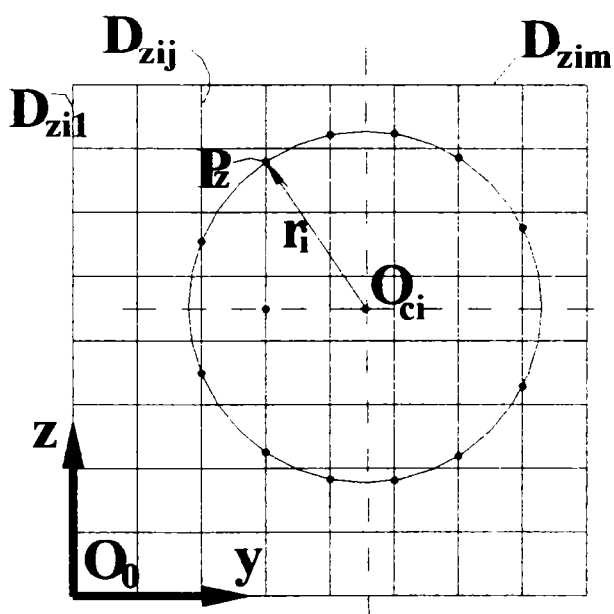


Fig.7.5.4. Determinarea intersecției sferei cu dreptele (Dx_{jk})

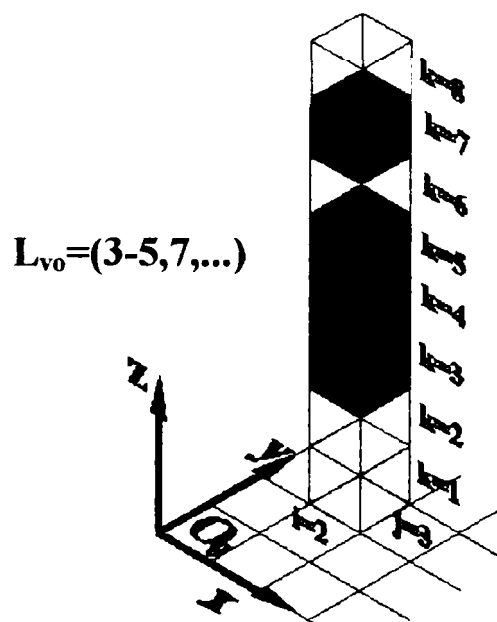


Fig.7.5.5. Indicarea elementelor volumice ocupate de corpul delimitat de fețe 3D

În urma determinării tuturor punctelor de intersecție dintre sferă și dreptele paralele cu axele de coordonate și stocarea coordonatelor punctelor în liste $L_P(i,j)$ se trece la construirea unor fețe 3D din aceste puncte. Pentru fiecare listă $L_P(i,j)$ se ordonează punctele crescător în funcție de valoarea coordonatei z. Fiecare volum paralelipipedic avînd ca și colțuri diametral opuse punctele $P_1(x_{i-1}, y_{j-1}, z_1)$ și $P_2(x_i, y_j, z_n)$ mai are asociată o listă L_{v0} care indică elementele volumice ocupate de către semifabricat. Această listă e de forma $L_{v0}=(3-5,7,...)$ unde 3-5 indică faptul că elementele volumice cu idicele $k=3,4,5$ sunt ocupate de către semifabricat iar cifra 7 indică faptul că și

Cheia de identificare a feței este de obicei un șir care permite regăsirea rapidă a entității geometrice de tip față 3D în modelul CAD. Informațiile referitoare la o față 3D stocate în lista $Lst(i,j)$ sunt păstrate sub forma unor grupuri cu 3 elemente (indicele k al elementului volumic în care se găsește fața 3D, cheia de identificare a feței în CAD și o listă cu coordonatele colțurilor). În figura 7.5.7. sunt prezentate câteva din posibilele situații ale fețelor 3D în interiorul elementelor volumice. Se observă că numărul de colțuri ale suprafeței poligonale compusă din fețe este variabil.

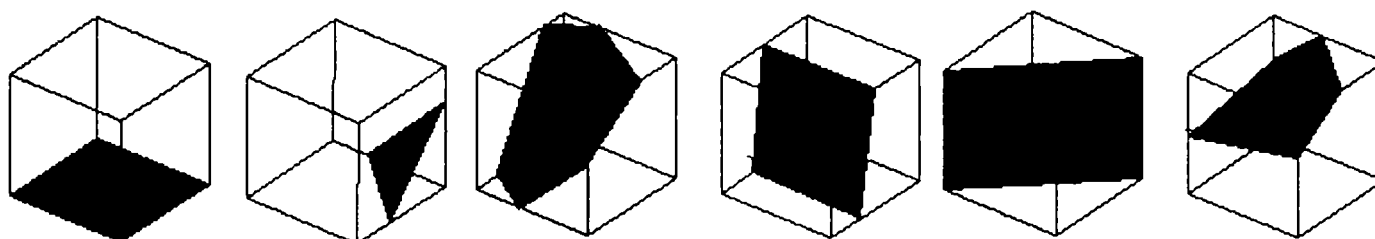


Fig.7.5.7. Exemple de fețe 3D

Schema logică ce sintetizează algoritmul pentru generarea semifabricatului de formă sferică este prezentat în figura 7.5.6.

7.5.2.3. Generarea semifabricatului virtual cilindric

În cazul semifabricatului de formă cilindrică se utilizează un algoritm asemănător cu cel utilizat în cazul semifabricatelor de formă sferică. Diferența constă în modul de generare a punctelor de intersecție a dreptelor paralele cu axele de coordonate cu suprafața cilindrică. Utilizatorul specifică numărul de elemente volumice l, m, n pe fiecare din direcțiile axelor de coordonate, coordonatele centrului $O_c(x_c, y_c, z_c)$ bazei suprafeței cilindrice, raza r_c și înălțimea h_c a cilindrului. Notățiile utilizate sunt ilustrate în figurile 7.5.8 și 7.5.9.

Pentru determinarea punctelor de intersecție P_{ext} dintre suprafața cilindrică laterală și planele paralele cu planul xO_0y se utilizează relațiile de calcul de la punctul 7.5.2.2.. Punctele de intersecție P_{int} dintre cele 2 suprafețe capac cu dreptele verticale sunt punctele a căror coordonate verifică relațiile :

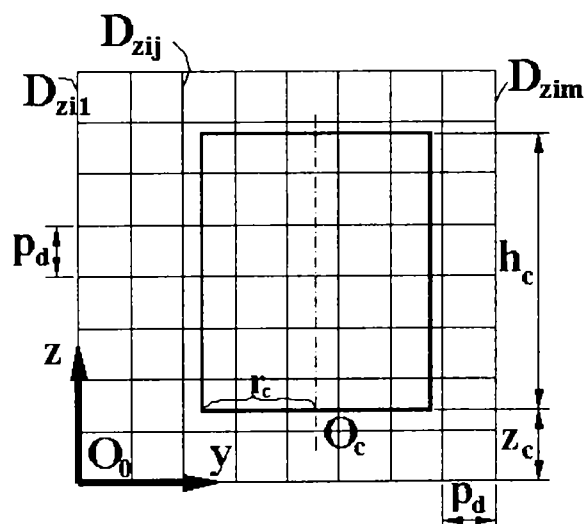


Fig.7.5.8 Vedere laterală a structurii de elemente volumice

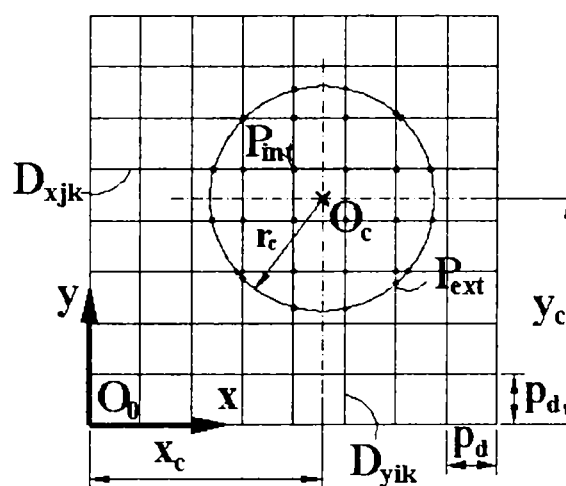


Fig. 7.5.9. Determinarea punctelor de intersecție între suprafața cilindrică și rețeaua 3D

$$\begin{cases} x_{P_{int}} = i \cdot p_d, & i = 1 \div l \\ y_{P_{int}} = j \cdot p_d, & j = 1 \div m \\ x_c - r_c < x_{P_{int}} < x_c + r_c \\ y_c - r_c < y_{P_{int}} < y_c + r_c \\ |x_c - x_{P_{int}}|^2 + |y_c - y_{P_{int}}|^2 \leq r_c^2 \\ (z_{P_{int}} = z_c) \cup (z_{P_{int}} = z_c + h_c) \end{cases} \quad (7.5.8)$$

În continuare se aplică algoritmul prezentat în figura 7.5.6 (doar partea încadrată de poligonul cu linie întreruptă). Zona delimitată de linie întreruptă din schema logică de la figura 7.5.6 se utilizează la generarea semifabricatului virtual, indiferent de forma sa.

7.5.3. Extragerea de material cu ajutorul sculei virtuale condusă cu digitizorul 3D

7.5.3.1. Determinarea situației sculei virtuale cu ajutorul digitizorului 3D

Impunerea situației palpatorului digitizorului 3D prin acționare manuală de către operatorul uman permite definirea unor poziții ale punctului caracteristic și a unor orientări ale dreptei caracteristice a digitizorului 3D. Modelul geometric al digitizorului 3D poate fi vizualizat în orice moment pe ecranul calculatorului. Elementele modelului geometric al digitizorului 3D se deplasează sincron cu elementele digitizorului real acționat de către operator. Scula virtuală este atașată palpatorului modelului geometric al digitizorului 3D. Scula virtuală are atașat un punct caracteristic și o dreaptă caracteristică care definesc situația sculei în raport cu un sistem de referință fix. Pentru a permite sculptura virtuală, matricea de situație a palpatorului digitizorului real este aplicată în orice moment sculei virtuale.

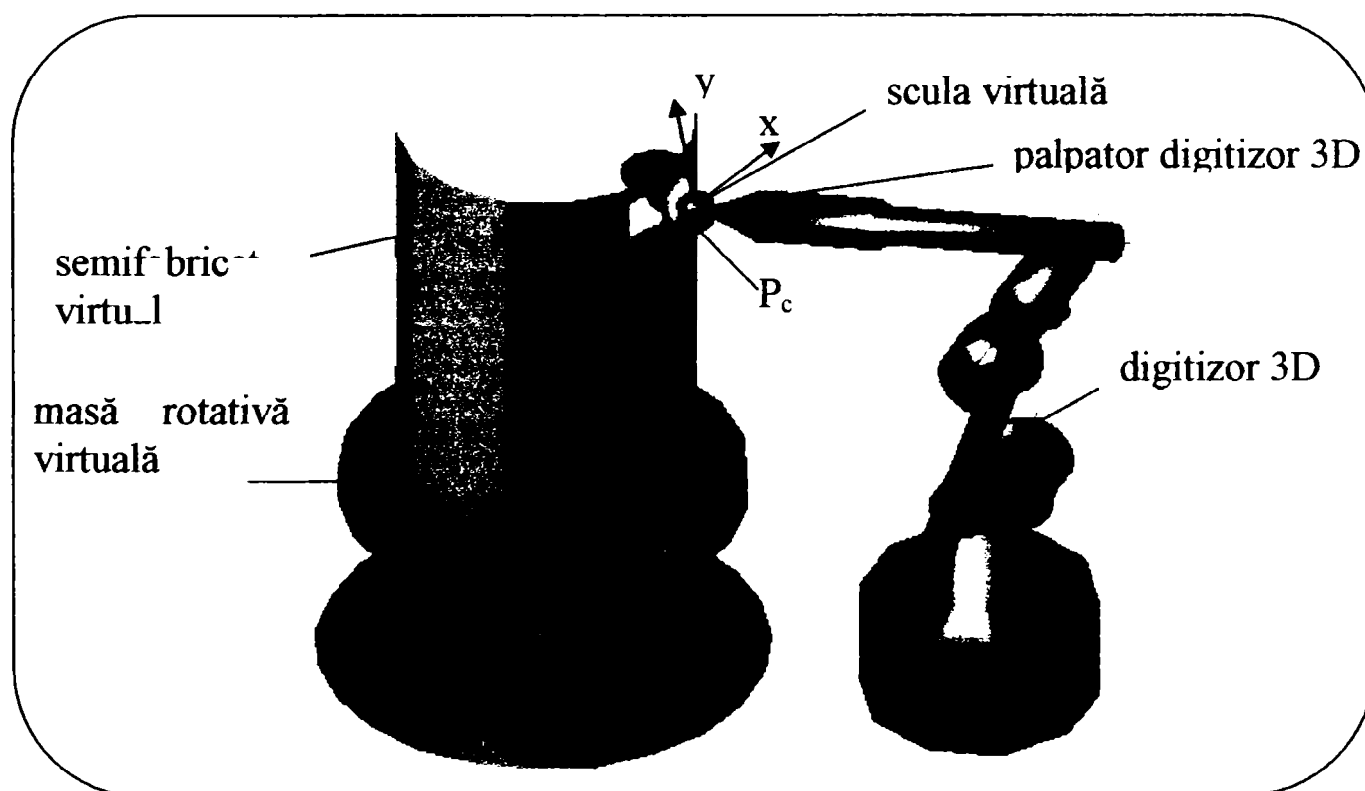


Fig. 7.5.10. Interacțiunea dintre scula atașată de palpatorul digitizorului 3D și un semifabricatul virtual.

În cazul în care scula are o formă sferică, interesează doar poziția punctului caracteristic al digitizorului 3D real în care se va considera centrul sferei ce reprezintă scula virtuală. În figura 7.5.10. se prezintă interacțiunea dintre scula atașată de modelul geometric al palpatorului digitizorului 3D RRR și un semifabricat virtual vizualizate pe monitorul calculatorului.

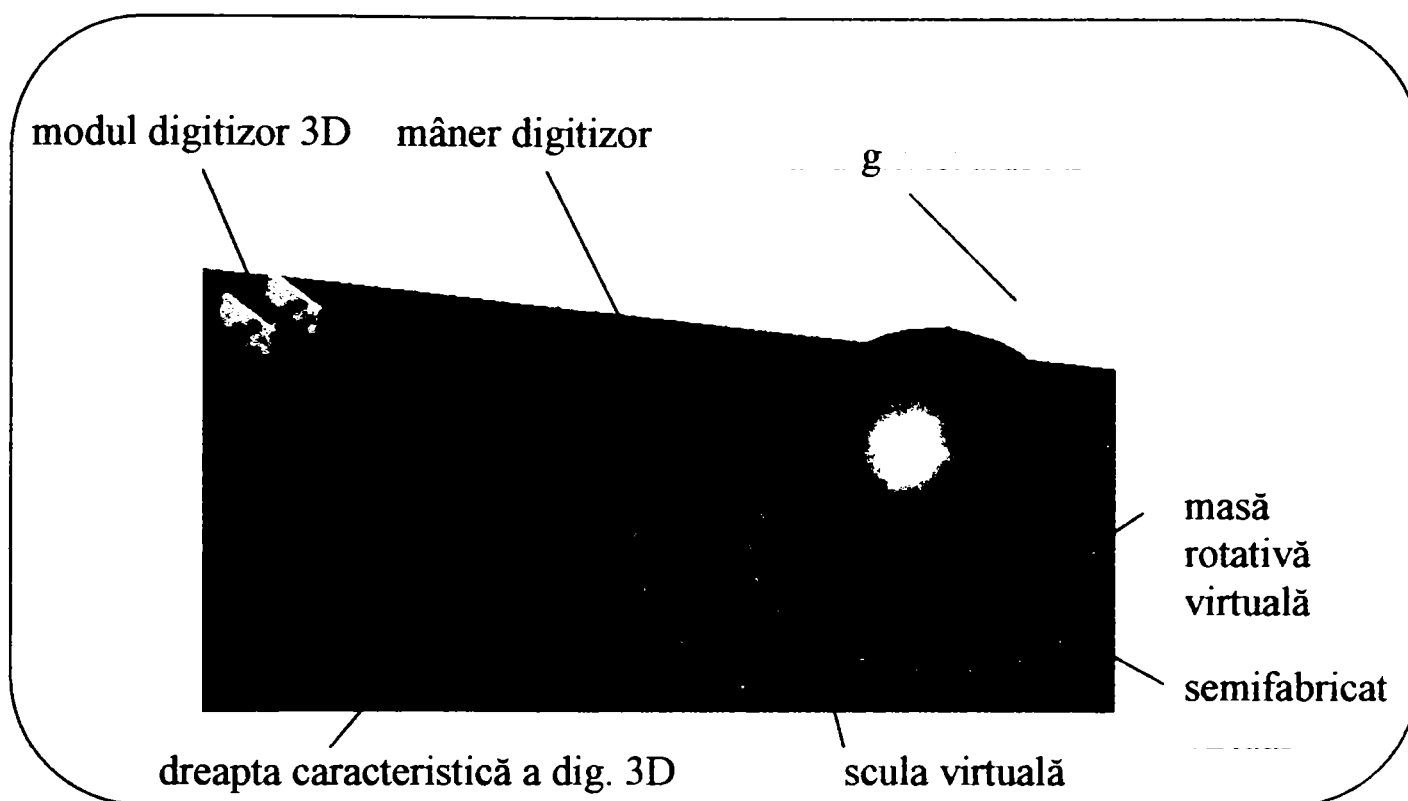


Fig. 7.5.11. Sculptură virtuală cu ajutorul digitizorului 3D cu fire

Spațiul de lucru al digitizorului 3D trebuie să includă volumul ocupat de structura de elemente volumice care stochează fațetele 3D ale semifabricatului virtual. Pentru o situare relativă mai facilă dintre scula virtuală și semifabricat se poate utiliza o masa virtuală rotativă care permite aplicarea unei rotații a semifabricatului în jurul axei mesei rotative. Digitizorul 3D cu fire permite pe lângă indicarea poziției sculei și indicarea orientării dreptei caracteristice. Aceasta facilitează utilizarea unor scule având o axă de simetrie. În figura 7.5.11. este prezentat principiul sculpturii virtuale cu o sculă tronconică cu vârf sferic a cărei situare este comandată cu ajutorul unui digitizor cu fire.

7.5.3.2. Algoritm pentru sculptura virtuală

Algoritm utilizat pentru îndepărtarea de material din semifabricatul virtual presupune existența semifabricatului în forma rezultată în urma algoritmului descris la punctul 7.5.2. Sunt utilizate listele $L_{st}(i,j)$ și L_{vo} . Se numește sesiune de lucru perioada de timp scursă între momentul deschiderii aplicației și închiderea acesteia. Pentru a evita calculele necesare pentru reumplerea listelor $L_{st}(i,j)$ și L_{vo} la fiecare început al unei sesiuni de lucru cu un anumit model, înainte de închiderea aplicației, conținutul listelor $L_{st}(i,j)$ și L_{vo} este stocat într-un fișier al calculatorului. La începutul sesiunii de lucru, prima etapă este preluarea informațiilor stocate în fișier și generarea listelor

$Lst(i,j)$ și L_{vo} . Algoritmul prezentat este în mare parte general, indiferent de forma sculei și a semifabricatului. Ca restricție se impune totuși ca scula să aibă o formă convexă. Se exemplifică algoritmul pentru cazul unei scule de formă sferică. Pentru o anumită situație a sculei relativă la semifabricat se determină elementele volumice din structura de elemente volumice care sunt situate integral în interiorul volumului delimitat de suprafața sculei. În figura 7.5.12. este prezentată o secțiune prin structura de elemente volumice paralelă cu planul xO_0y în momentul declanșării operației de îndepărtare a materialului. În figură s-a notat cu $C_S(x_s, y_s, z_s)$ centrul momentan al suprafeței sferice a sculei virtuale exprimat în sistemul de referință atașat structurii de elemente volumice. Primul pas al algoritmului este delimitarea zonei din structura de elemente volumice în care se vor realiza operații asupra fațetelor semifabricatului (zona hașurată în figura 7.5.12).

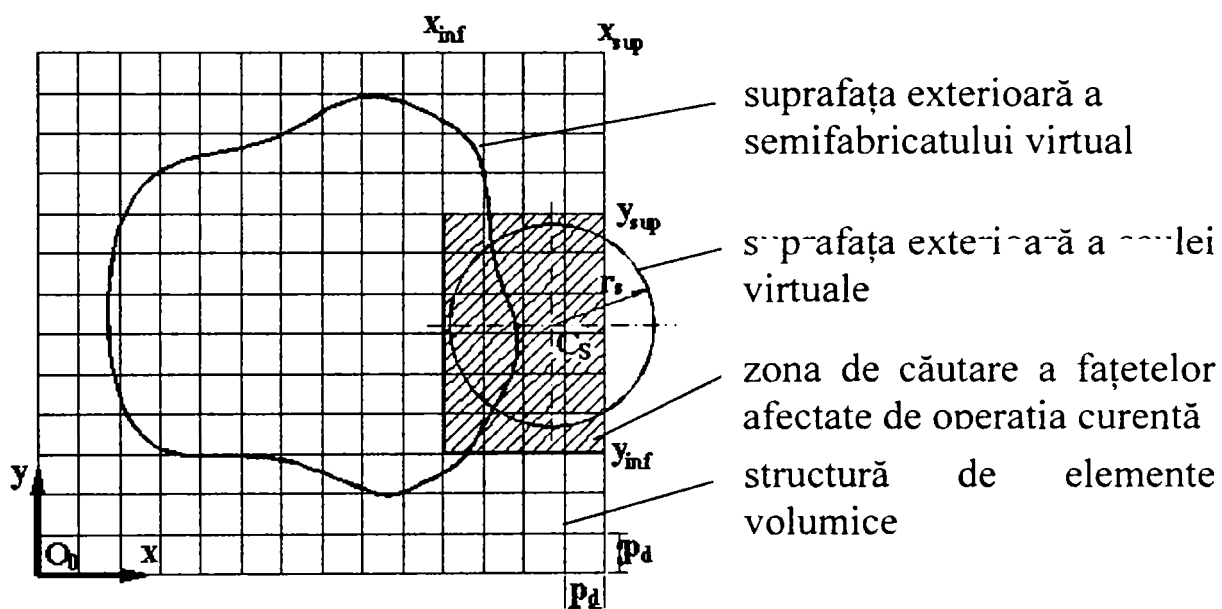


Fig. 7.5.12. Delimitarea zonei de căutare a fețelor 3D posibil afectate de operația curentă de îndepărtare de material

Se determină indicii i_s , j_s și k_s ai elementului volumic care conține centrul sferei ce reprezintă scula.

$$i_s = \text{int}\left(\frac{x_s}{p_d}\right) + 1; \quad j_s = \text{int}\left(\frac{y_s}{p_d}\right) + 1; \quad k_s = \text{int}\left(\frac{z_s}{p_d}\right) + 1 \quad (7.5.9)$$

Valorile limită ale indicilor i , j , k care delimitează elementele volumice care conțin scula virtuală se determină prin relațiile:

$$i_{inf} = i_s - \text{int}\left(\frac{r_s}{p_d}\right) - 1; \quad i_{sup} = i_s + \text{int}\left(\frac{r_s}{p_d}\right) + 1 \quad (7.5.10)$$

$$j_{inf} = j_s - \text{int}\left(\frac{r_s}{p_d}\right) - 1; \quad j_{sup} = j_s + \text{int}\left(\frac{r_s}{p_d}\right) + 1 \quad (7.5.11)$$

$$k_{\text{inf}} = k_s - \text{int}\left(\frac{r_s}{p_d}\right) - 1; \quad k_{\text{sup}} = k_s + \text{int}\left(\frac{r_s}{p_d}\right) + 1 \quad (7.5.12)$$

Este posibil ca o porțiune a sculei să fie în exteriorul structurii de elemente volumice și în aceste cazuri se realizează corecții ale limitelor indicilor. De exemplu pentru indicele corespunzător axei O_0x :

$$\begin{aligned} \text{daca } i_{\text{inf}} < 1 \text{ atunci } i_{\text{inf}} &= 1 \\ \text{daca } i_{\text{sup}} > l \text{ atunci } i_{\text{sup}} &= l \end{aligned} \quad (7.5.13)$$

În toate figurile care reprezintă structuri de elemente volumice s-au utilizat valori reduse pentru parametrii l , m și n ai structurii, deci un număr redus de elemente volumice pe fiecare direcție paralelă cu axele de coordonate. Acesta este doar pentru claritatea figurilor. În realitate, pentru a obține o imagine acceptabilă a sculpturii virtuale, se utilizează un număr mult mai mare de elemente volumice pe fiecare direcție. Pentru a ilustra influența asupra vitezei de procesare a delimitării zonelor de căutare a fețelor 3D care sunt afectate de îndepărtarea de material curentă (pentru poziția momentană a sculei virtuale) se consideră o valoare medie pentru parametrii structurii de elemente volumice: $l = m = n = 400$. Numărul total de elemente volumice ai structurii este $Nr_{\text{TotVox}} = 64 \cdot 10^6$ [voxeli]. Se consideră valoarea pasului $p_d = 1$ [mm] și raza sferei care reprezintă scula $r_s = 10$ [mm]. În cazul în care scula este în întregime în interiorul structurii de elemente volumice, căutarea fațetelor afectate de operația curentă se realizează într-un număr de $Nr_{\text{VoxCaut}} = 22 \cdot 22 \cdot 22 = 10648$. Deci viteza de determinare a fațetelor 3D posibil afectate de îndepărtarea curentă de material crește în urma aplicării algoritmului prezentat anterior de $s_v = \frac{Nr_{\text{TotVox}}}{Nr_{\text{VoxCaut}}} = 6010$ ori și permite operarea în timp real.

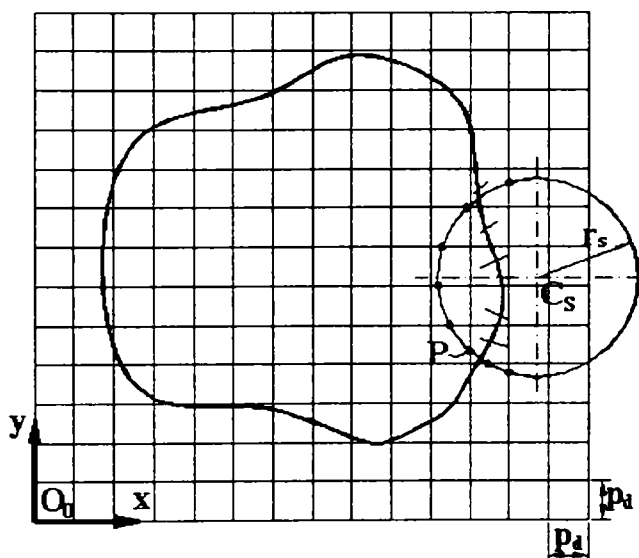


Fig. 7.5.13 Secțiune înainte de extragerea de material din semifabricatul virtual

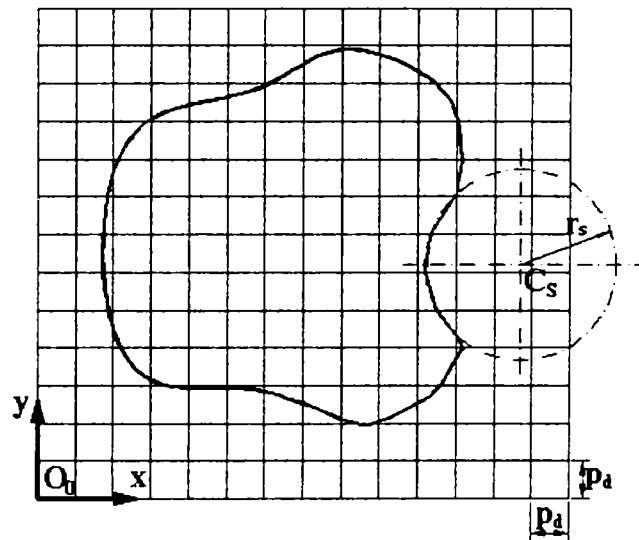


Fig. 7.5.14. Secțiune în urma extragerii de material din semifabricatul virtual

În urma delimitării elementelor volumice în care se realizează căutarea fețelor semifabricatului posibil afectate se elimină fațetele situate integral în interiorul volumului delimitat de scula sferică. Se determină punctele de intersecție P dintre suprafața sferică a sculei și rețeaua de drepte $(D_{x_{jk}})$, $(D_{y_{ik}})$, $(D_{z_{ij}})$, paralele cu O_0x , O_0y și respectiv O_0z pe baza algoritmului prezentat la punctul 7.5.2.2., utilizat pentru generarea semifabricatului sferic. Se generează listele $L_P(i,j)$ în care se stochează punctele P situate în interiorul semifabricatului. Pentru a determina care din punctele P sunt situate în interiorul semifabricatului se utilizează listele cu elementele volumice ocupate de materialul semifabricatului $L_{v_0(i,j)}$. Fațetele 3D care au unul sau mai multe colțuri în interiorul semifabricatului și unul sau mai multe colțuri în exteriorul semifabricatului se elimină, iar punctele care reprezintă colțurile fațetelor se trec în listele $L_P(i,j)$ corespunzătoare. Pentru definitivarea operației de îndepărtare a materialului se generează fațetele 3D pe baza informațiilor conținute în listele $L_P(i,j)$ și se stochează informațiile despre aceste fațete în listele $Lst(i,j)$. În figura 7.5.13 este prezentată imaginea unei secțiuni paralele cu planul xO_0y care trece prin centrul suprafeței sferice a sculei înainte de extragerea de material din semifabricatul virtual. În figura 7.5.14. este prezentată o imagine a aceleiași secțiuni în urma încheierii operației. Schema logică simplificată a algoritmului de îndepărtare de material prin sculptură virtuală este prezentată în figura 7.5.15.

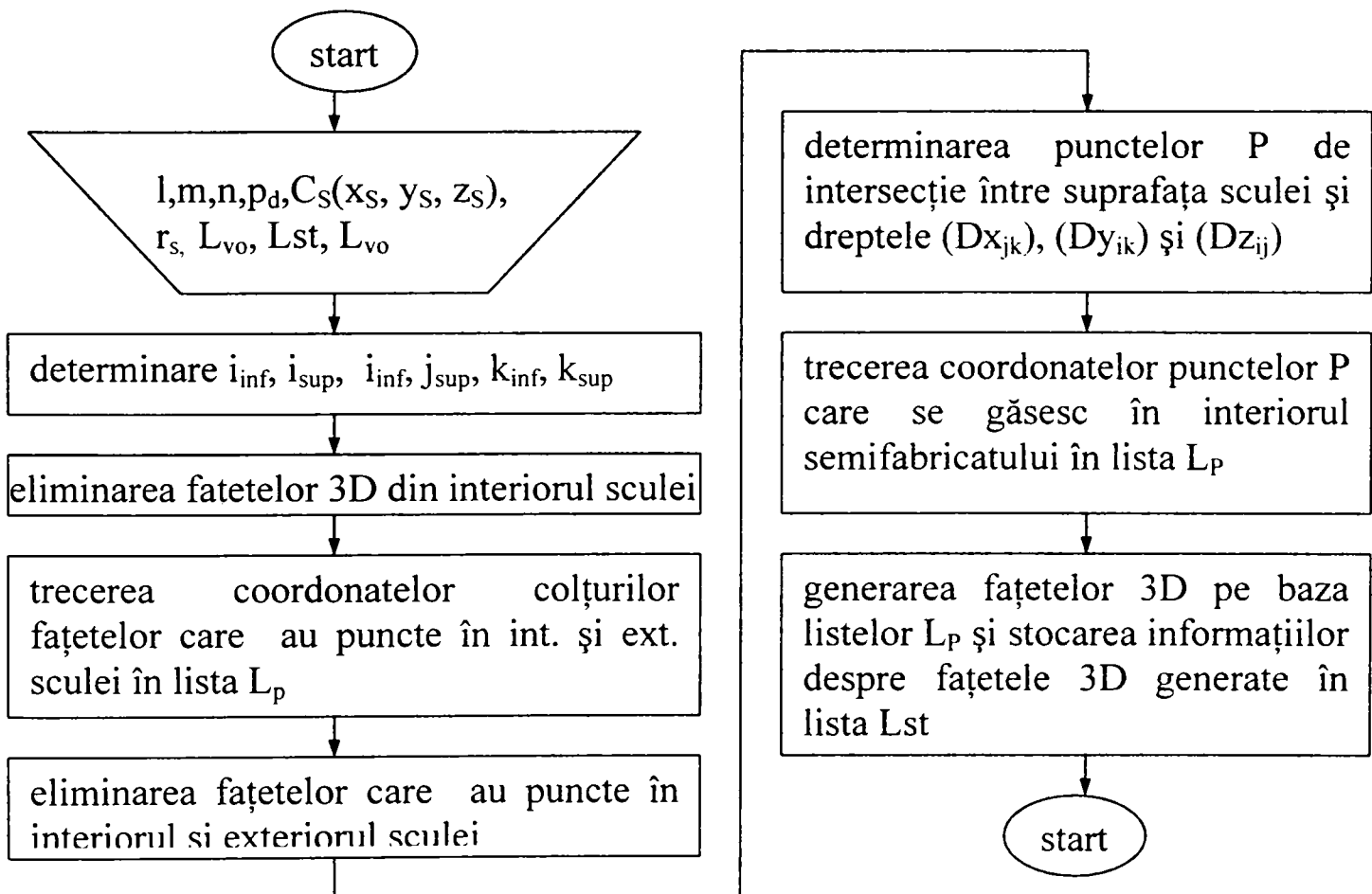
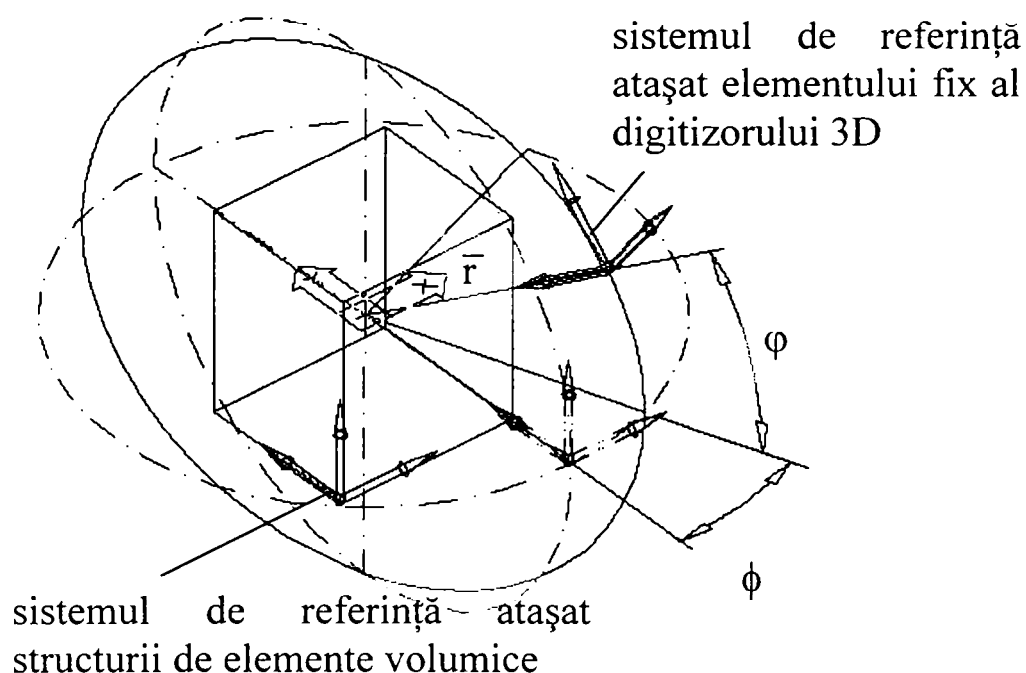


Fig. 7.5.15. Schema logică simplificată a algoritmului de îndepărtare de material prin sculptură virtuală

7.5.4. Facilități auxiliare ale aplicației de sculptură virtuală

În acest paragraf sunt prezentate o serie de facilități care ușurează desfășurarea operației de sculptură virtuală sau care permit obținerea unor modele mai valoroase din punct de vedere artistic.

Pentru accesul facil al sculei virtuale conduse prin intermediul unui digitizor 3D cu un număr redus de grade de mobilitate la suprafața semifabricatului virtual este necesară posibilitatea schimbării situației relative a sistemului de referință fix atașat digitizorului 3D față de sistemul de referință fix atașat structurii de elemente volumice. O variantă ar fi utilizarea unei mese rotative virtuale de suprafața căreia să fie fixată structura de elemente volumice. Rotirea mesei împreună cu structura de elemente volumice care încadrează semifabricatul virtual se poate comanda de către utilizator prin intermediul tastaturii sau a unor microîntrerupătoare ale digitizorului 3D. În cazul în care structura de elemente volumice conține un mare număr de fațete 3D, fiecare rotire a mesei ar implica recalcularea coordonatelor tuturor colțurilor fețelor 3D. Acest lucru este necesar deoarece algoritmi utilizați pentru îndepărtarea de material prin sculptură virtuală presupun ca operațiile să se desfășoare în sistemul de coordonate atașat structurii de elemente volumice. O soluție rațională este schimbarea situației relative dintre structura de elemente volumice și digitizorul 3D prin modificarea situației sistemului de coordonate asociat digitizorului 3D. Schimbarea situației sistemului de referință fix al digitizorului 3D se realizează într-un sistem de coordonate sferic. Centrul sistemului de referință sferic este considerat în centrul de greutate al volumului ocupat de structura de elemente volumice. Prin intermediul tastaturii sau a unor microîntrerupătoare a digitizorului 3D se modifică (cu reactualizarea dinamică imaginii în mediul grafic pentru feed-back vizual) parametrii unghiulari ϕ , φ și deplasarea de-a lungul vectorului r (fig. 7.5.16).



Asigurarea schimbării unghiului de vedere se realizează prin comenzile specifice sistemului CAD în care se implementează aplicația de sculptură virtuală.

Pentru a îmbunătăți calitatea artistică este posibilă colorarea fațetelor 3D cu ajutorul sculei virtuale. Pentru colorare se alege o anumită culoare și se dezactivează eliminarea de material cu ajutorul sculei virtuale. Fațetele 3D situate în interiorul suprafeței delimitate de sculă se colorează în culoarea aleasă.

7.5.5. Direcții posibile de dezvoltare a aplicației

Prin calculul volumului de material din semifabricatul virtual cuprins în interiorul suprafeței exterioare a sculei, determinarea vitezei de deplasare a sculei și alegerea unui anumit material pentru semifabricat se poate determina cu aproximație forța întimpinată la deformarea suprafeței materialului semifabricatului. În cazul în care în locul digitizorului 3D se utilizează un dispozitiv haptic de interacțiune cu realitatea virtuală, utilizatorul poate sesiza forța cu care materialul semifabricatului se opune deformării, senzația de imersiune în mediul virtual fiind substanțial îmbunătățită.

Prin divizarea recursivă a elementelor volumice și stocarea informațiilor într-o structură arborescentă în locul unei structuri matriciale se poate obține o suprafață a modelului cu o dimensiune mai redusă a fațetelor 3D, deci implicit o calitate mai bună a vizualizării fără a crește substanțial cantitatea de memorie ocupată de structura de elemente volumice.

Alegerea unui pas de divizare p_d egal cu grosimea stratului de depunere a unei mașini de realizare rapidă a prototipurilor permite exportul facil al modelului într-un format acceptat de softul care coordonează mașina și permite materializarea sculpturii virtuale.

7.6. Compararea unui obiect existent fizic cu modelul geometric “ideal” al acestuia utilizând un digitizor 3D cu palpator

7.6.1. Calibrarea pentru “suprapunerea” obiectelor

Pentru compararea unor date geometrice ale obiectului real cu echivalentul lor din cadrul modelului din calculator este necesar ca sistemele de referință față de care se definesc aceste date să fie identice.

Realizarea suprapunerii celor 2 sisteme de referință ale obiectului real existent fizic notat cu SR_f cu cel al modelului geometric al acestuia notat SR_c se realizează pe baza algoritmului prezentat la subcapitolul 5.3. Este de mare importanță ca cele 3 puncte pe obiectul real să în fie alese în puncte pentru care utilizatorul știe apriori că diferențele față de corespondentele lor din modelul geometric sunt minime. Punctele alese pot fi, de exemplu, puncte de fixare în dispozitiv la prelucrările mecanice. De asemenea, este de preferat alegerea unor puncte particulare din considerente geometrice ca de exemplu colțuri sau concavități. În cazul în care ulterior este necesară o deplasare a obiectului real față de elementul fix al digitizorului 3D punctele alese pentru calibrare trebuie să fie marcate pe obiect sau să fie suficient de ușor identificabile. Matricea de trecere de la sistemul de referință atașat digitizorului 3D la sistemul de referință atașat obiectului este utilizată pentru exprimarea fiecărui punct măsurat pe suprafața obiectului.

7.6.2. Compararea geometriei modelului existent fizic cu modelul geometric

7.6.2.1 Introducere

Ca mod de lucru se propune indicarea succesivă a entității din modelul geometric și palparea cu ajutorul digitizorului a unor puncte de pe suprafața obiectului fizic urmată de determinarea abaterilor. Prin secvență de măsurare se înțelege în continuare succesiunea entităților geometrice de tip punct, linie, curbă, suprafață din modelul geometric a căror corespondență cu obiectul real se verifică. Secvența de măsurare poate fi înregistrată în timpul măsurării, prin validarea fiecărei entități pentru înregistrare. Rularea unei secvențe de măsurare înregistrată anterior poate fi utilă de exemplu pentru determinarea corespondenței unui lot de piese cu modelul geometric al acestora. Stocarea datelor rezultate în urma comparărilor succesive poate fi utilizată pentru realizarea unor calcule statistice. Pe baza coordonatelor punctelor obținute cu digitizorul 3D, în modelul CAD poate fi vizualizată entitatea geometrică obținută pe baza acestor puncte și poate fi realizată o comparare vizuală.

Măsurarea ghidată poate fi utilă la compararea model geometric – obiect fizic. Utilizatorul poate vizualiza în mod dinamic pe ecranul calculatorului direcția în care trebuie să deplaseze vârful digitizorului pentru ca punctul palpat să se suprapună cu punctul specificat în modelul geometric sau precizat prin coordonatele sale. Prin această facilitate, utilizatorul determină unde trebuia să fie un punct al modelului fizic

pentru a se suprapune cu cel din modelul geometric. Direcția necesară de deplasare a vârfului palpatorului se poate indica printr-o săgeată în modelul geometric și se poate afișa dinamic distanța dintre vârful palpatorului și punctul corespondent din modelul geometric. Pentru a se putea concentra asupra vârfului palpatorului utilizatorul este îndrumat prin mesaje vocale de către aplicație prin mesaje de tipul “sus”, “stânga jos”, “punct atins”, etc. Mesajul “punct atins” este dat când vârful palpatorului este situat la o distanță mai mică decât o distanță limită indicată de utilizator.

7.6.2.2. Compararea punctelor

Pentru compararea punctelor se selectează punctul din modelul geometric și punctul corespunzător pe modelul fizic. Fie $P_c(x_c, y_c, z_c)$ punctul din modelul geometric și $P_o(x_o, y_o, z_o)$ punctul corespunzător de pe suprafața modelului fizic. Compararea celor două puncte se referă la determinarea abaterilor $d_x = x_c - x_o$, $d_y = y_c - y_o$, $d_z = z_c - z_o$ și a distanței dintre cele 2 puncte.

7.6.2.3. Compararea liniilor

Pentru compararea unor linii de lungime finită se vor selecta 2 puncte pe modelul geometric și apoi se palpează 2 puncte pe modelul fizic. Fie punctele $M_1(x_1, y_1, z_1)$ și $M_2(x_2, y_2, z_2)$ capetele liniei din modelul geometric și $M_3(x_3, y_3, z_3)$ respectiv $M_4(x_4, y_4, z_4)$ punctele corespondente palpate pe suprafața obiectului fizic. Se notează cu D1 dreapta suport a liniei din modelul geometric și cu D2 dreapta definită de punctele M_3 și M_4 . Vectorii directori ai dreptelor D1 și D2 se notează cu $\vec{v}_1(l_1, m_1, n_1)$ respectiv $\vec{v}_2(l_2, m_2, n_2)$. Componentele vectorilor directori se pot determina cu ajutorul coordonatelor punctelor care definesc liniile:

$$l_1 = x_2 - x_1; \quad m_1 = y_2 - y_1; \quad n_1 = z_2 - z_1 \quad (7.6.1)$$

$$l_2 = x_4 - x_3; \quad m_2 = y_4 - y_3; \quad n_2 = z_4 - z_3 \quad (7.6.2)$$

Datele rezultate în urma comparării sunt abaterile coordonatelor punctelor de capăt corespondente ale liniilor, abaterile lungimii segmentului, ecuațiile parametrice sau canonice ale celor 2 drepte suport, unghiul în spațiu între linia din modelul geometric și cea obținută prin palparea punctelor pe obiectul fizic. În continuare este utilizat aparatul matematic prezentat în [MUR62].

Ecuațiile parametrice ale dreptei definite de punctele palpate este :

$$\begin{cases} x = x_3 + \lambda(x_4 - x_3) \\ y = y_3 + \lambda(y_4 - y_3) \\ z = z_3 + \lambda(z_4 - z_3) \end{cases} \quad (7.6.3)$$

iar ecuațiile canonice:

$$\frac{x - x_3}{l_2} = \frac{y - y_3}{m_2} = \frac{z - z_3}{n_2} \quad (7.6.4)$$

Unghiul în spațiu al dreptelor D1 și D2 este dat de formula:

$$\cos \varphi = \frac{l_1 \cdot l_2 + m_1 \cdot m_2 + n_1 \cdot n_2}{\varepsilon \cdot \sqrt{l_1^2 + m_1^2 + n_1^2} \cdot \sqrt{l_2^2 + m_2^2 + n_2^2}}, \quad \varepsilon = \pm 1 \quad (7.6.5)$$

Semnul lui ε corespunde celor 2 unghiuri suplimentare formate de dreptele D_1 și D_2 .

7.6.2.4. Compararea curbilor

În cazul comparării curbilor, datele obținute sunt deviațiile unor puncte corespondente de pe curbă și erorile razei de curbură în aceste puncte (fig. 5). Curba de pe modelul fizic este indicată ca polilinie 3D, arc de cerc, cerc sau curbă spline. Curba de pe suprafața obiectului fizic este indicată prin puncte. Pentru comparare, polilinia 3D obținută din punctele palpate pe suprafața obiectului se divide în segmente egale cu numărul de segmente în care a fost divizată curba din modelul geometric rezultând o mulțime de puncte. Pentru fiecare perechi de puncte se calculează abaterea de poziție și eroarea de curbură. Rezultatele se prezintă în formă grafică într-o scală de culori prin vizualizarea în modelul geometric a unor linii care unesc perechile de puncte corespondente de pe cele două curbe. Pentru o anumită lungime de la un capăt al curbei indicat de utilizator se poate determina punctul corespondent pe suprafața modelului fizic și se pot determina abaterile.

7.6.2.5. Compararea suprafețelor

Problemele care se pun la compararea suprafețelor sunt mai complexe deoarece cel puțin una din suprafețe și anume cea obținută prin digitizare este o suprafață complexă, în majoritatea cazurilor polifațetată. De obicei suprafața din modelul geometric corespunzătoare suprafeței digitizate este o suprafață definită parametric. Principalul element care interesează la compararea suprafețelor este distanța dintre cele 2 suprafețe. Pentru suprafețele selectate este posibilă determinarea și afișarea unor caracteristici și anume tipul de suprafață (parametric sau polifațetat), se poate seta distanța minimă dintre suprafețe pentru comparare. La realizarea comparării, pentru fiecare față elementară a suprafeței din modelul geometric se determină normala care trece prin centrul de greutate al feței și se determină intersecția cu suprafața rezultată în urma digitizării.

Un algoritm pur iterativ ar fi mare consumator de resurse și timp de calcul și de aceea în prima fază se vor determina centrele de greutate și normalele pentru fiecare față a suprafeței din modelul geometric. Se extrag coordonatele vertexurilor și fațetele suprafeței digitizate și se construiește o structură paralelipipedică având în componență elemente volumice cu dimensiunea laturii propusă implicit de către program dar setabilă de către utilizator. Corespunzător fiecărui element volumic sunt stocate fațetele ale căror centre de greutate sunt cuprinse în elementul volumic respectiv. Pentru realizarea calculelor de determinare a distanței dintre suprafața din modelul geometric și cea rezultată în urma digitizării se consideră pentru fiecare față a suprafeței modelului geometric fațetele suprafeței rezultate prin digitizare care sunt cuprinse în elementele volumice care încadrează o sferă cu centrul în centrul feței

modelului geometric și cu raza egală cu distanța maximă pentru care este necesară compararea. Pentru fiecare fațetă se determină punctul sau punctele de intersecție dintre normala N și fațetele rezultate în urma digitizării. Intersecția I_i corespunzătoare distanței minime între punctul c_i aparținând suprafeței geometric și I_i aparținând suprafeței digitizate se stochează împreună cu punctul c_i . Aceste date sunt utilizate pentru calcule statistice și la afișarea prin cod de culori a distanțelor cu linii care unesc centrul fiecărei fețe a suprafeței din modelul geometric cu punctul de intersecție corespondent cu suprafața rezultată prin digitizare.

8. Contribuții personale. Concluzii finale

8.1. Contribuții personale

8.1.1 Introducere

Tema abordată în cadrul tezei de doctorat contribuie la dezvoltarea unui domeniu de mare actualitate, cu o dinamică deosebită a dezvoltării.

În cadrul tezei s-au urmărit următoarele obiective:

- sistematizarea bibliografiei referitoare la construcția și aplicațiile digitizoarelor 3D;
- prezentarea unor aplicații software care procesează informațiile prelevate cu ajutorul digitizoarelor 3D;
- stabilirea relațiilor dintre construcția robotului și construcția digitizorului 3D;
- modelarea preciziei digitizoarelor 3D cu palpator;
- realizarea modelelor matematice ale unor tipuri de digitizoare 3D ;
- simularea funcționării unor digitizoare 3D concepute de autor;
- realizarea constructivă a unor variante de digitizoare 3D;
- realizarea unor aplicații ale digitizoarelor 3D în domeniul roboticii;
- prezentarea unor posibile aplicații ale digitizoarelor 3D în domeniul proiectării asistate de calculator;
- formularea unor concluzii privind posibilitățile actuale ale digitizoarelor 3D și ale direcțiilor de dezvoltare.

Contribuțiile originale ale autorului rezultă din urmărirea realizării obiectivelor propuse prin teza de doctorat și pot fi sintetizate pe trei direcții principale:

- în plan teoretic;
- în domeniul contribuțiilor practice;
- în domeniul aplicațiilor concrete.

8.1.2 Contribuții în plan teoretic

În plan teoretic contribuțiile sunt:

- conceperea algoritmului pentru determinarea volumului de toleranță corespunzător unei situații date a palpatorului în spațiul de lucru al digitizorului 3D;
- conceperea algoritmului pentru determinarea hărții de precizie a digitizorului 3D;
- conceperea unei variante de determinare a suprafeței de frontieră a unui nor de puncte;
- elaborarea unei metodologii de calibrare a digitizorului cu palpator cu acționare manuală;
- estimarea rezoluției necesare a traductoarelor pentru atingerea unei precizii impuse de măsurare a coordonatelor punctelor digitizate;
- calcularea matricii de transformare între două sisteme de referință pe baza coordonatelor a trei puncte exprimate în cele două sisteme de referință;

- elaborarea unui algoritm eficient pentru generarea suprafețelor pornind de la șiruri de puncte obținute prin digitizare.

8.1.3 Contribuții practice

Contribuțiile practice se materializează prin următoarele:

- realizarea unui digitizor 3D cu palpator cu structură RRR;
- întocmirea unui program pentru calcul geometric direct în vederea obținerii coordonatelor punctului caracteristic al palpatorului;
- implementarea unui program pentru determinarea coordonatelor unor șiruri de puncte cu generarea în timp real de polilinii 3D în AutoCAD;
- realizarea unor încercări experimentale cu digitizorul 3D RRR;
- întocmirea unor programe pentru procesarea automată a datelor prelevate în cadrul încercărilor experimentale;
- realizarea modelului matematic al unui digitizor 3D cu fire;
- proiectarea constructivă a unui digitizor cu fire;
- întocmirea programului pentru calculul geometric invers în cea de a doua variantă de montare a modulelor de măsurare a lungimii firelor digitizorului 3D;
- întocmirea programului pentru calculul geometric direct în prima variantă de montare a modulelor de măsurare a lungimii firelor digitizorului 3D;

8.1.4 Contribuții la realizarea unor aplicații ale digitizoarelor 3D

- realizarea unei aplicații software pentru programarea off-line a roboților în mediu virtual;
- realizarea modelului geometric invers al robotului SCORBOT ER III;
- colaborare la interfațarea aplicației RobSim cu robotul real;
- implementarea unui subprogram pentru precizarea situărilor efectorului final cu digitizorul 3D;
- întocmirea unui subprogram pentru realizarea modelului geometric al mediului robotului cu digitizorul 3D;
- prezentarea unei aplicații de teleoperare a robotului SCORBOT ER III cu digitizorul 3D;
- realizarea unei aplicații pentru generarea unor entități geometrice în AutoCAD pe baza unor puncte și șiruri de puncte prelevate cu digitizorul 3D;
- elaborarea unei tehnologii de grup pentru modelarea geometrică a pieselor de mobilier;
- prezentarea algoritmilor pentru realizarea modelelor geometrice ale unor tipuri de piese pornind de la șiruri de puncte obținute prin digitizare;
- conceperea algoritmilor pentru o aplicație de sculptură virtuală cu digitizorul 3D;
- conceperea unei metodologii pentru compararea unui obiect existent fizic cu modelul geometric “ideal” al acestuia utilizând un digitizor 3D cu palpator.

8.1.5 Direcții de cercetare viitoare

Lucrarea pune la dispoziția celor interesați un material considerat de autor util pentru formarea unei viziuni de ansamblu asupra digitizării 3D și a aplicațiilor posibile. Totodată, prin aplicațiile originale prezentate este ilustrată posibilitatea extinderii domeniului de utilizare a digitizoarelor 3D. Se deschid noi perspective de aplicații ale digitizoarelor 3D în robotică. Aplicația RobSim, concepută și implementată de autor prezintă o arhitectură deschisă și permite adăugarea cu ușurință a unor noi facilități.

Soluțiile constructive de digitizare 3D realizate de autor pentru validarea aspectelor teoretice prezentate în teză și a testării aplicațiilor software pot fi perfecționate și adaptate unor utilizări specifice.

8.2. Concluzii finale

Teza de doctorat cu titlul “Contribuții la construcția și utilizarea digitizoarelor 3D în proiectarea asistată de calculator și programarea roboților industriali” tratează într-o variantă originală aspecte legate de construcția digitizoarelor 3D și aplicațiile acestora.

Lucrarea este structurată pe 8 capitole incluzând 180 figuri și 7 tabele pe parcursul a 231 pagini. Anexele la prezenta teză cuprind 45 pagini.

Prima parte a lucrării este destinată încadrării digitizării 3D în contextul tehnicilor moderne, prezentării situației actuale a construcției și utilizării digitizoarelor 3D cu palpator cu acționarea manuală a palpatorului de către operatorul uman. S-a realizat o sistematizare a aplicațiilor digitizoarelor 3D în domeniul industrial și în alte domenii de activitate. Sunt prezentate unele aplicații software care utilizează informații obținute prin digitizare 3D.

În capitolul 3 sunt prezentate unele noțiuni legate de modelarea suprafețelor pentru facilitarea înțelegerii aplicațiilor concepute de autor în domeniul proiectării asistate.

Contribuțiile teoretice se axează în special pe aspecte legate de precizia digitizoarelor 3D. Pentru reconstrucția suprafețelor pornind de la șiruri de puncte este elaborat un algoritm original. Prin cele două tipuri de digitizare concepute și realizate se validează aspectele teoretice prezentate în lucrare.

Aplicațiile abordate de autor demonstrează că digitizoarele 3D pot mări substanțial productivitatea în domeniul proiectării asistate de calculator. Aplicația RobSim, cu interfață pentru digitizare 3D ilustrează posibilitățile de utilizare a digitizoarelor 3D în domeniul roboticii.

Autorul consideră că teza de doctorat, prin cercetările efectuate, îndeplinește toate obiectivele propuse prin tema tezei și deschide noi perspective în domeniul digitizării 3D.

Bibliografie

[BOG91] Bogchi, Amit; Beaman, Joseph: Intelligent design and manufacturing for prototyping, PED-vol. 50, edited by The American Society of Mechanical Engineers, New York, 1991

[BOI88] Boissonat, J.D.; Faverjon, B. ș.a.: Techniques de la robotique. Perception et planification, Ed. Hermes, Paris, 1988

[BUR98] Burchard, Bill; Pitzer, David; Soen, Francis ș.a. : Secrete AutoCAD 14, Editura Teora, București, 1998

[BUR94] Burdea G, Coifect Ph. : Virtual Reality Technology, Willey and Sons Publ., New York, 1994

[CRA86] Craig, I. J. (1986) Introduction to Robotics , Addison Wessley Publ., USA, 1986

[DAV01] Davidescu, A. : Metrologie generală, Editura Politehnica, Timișoara, 2001

[DIA00] Diaconu A. Contribuții la analiza și sinteza sistemelor de teleoperare controlate bilateral în poziție.- Teza de doctorat ,Timișoara, 2000.

[DOG88] Dogaru, Dorian – Elemente de grafică 3D, Editura Științifică și Enciclopedică, București, 1988

[DOL97] Dolga, L. : Bazele proiectării asistate de calculator, curs la Universitatea “Politehnica” Timișoara, 1997

[DUM96] Dumitriu, A.; Bucșan, C.;Demian T., : Sisteme senzoriale pentru roboți, Editura Medro, 1996

[FAU79] Faux, I.; Pratt, M.: Computational geometry for design and manufacture, Editura Ellis Horwood, New York, 1979

[GHE97] Gheorghe, I. Gh., : Contribuții la studiul și construcția traductoarelor fotoelectrice incrementale de deplasare și poziție utilizate în mecanica fină și robotică, Teza de doctorat, Timișoara , 1997

[HAM93] Hamit, F. , : Virtual Reality and the Exploration of Cyberspace, Editura Sams, 1993

[ISP99] Ispas, C-tin; Caramihai, S. și alții : Simularea sistemelor integrate de fabricație, Editura Bren, București, 1999

- [ISR00] Preceding of the 3^{1st} International Symposium on Robotics, Montreal, 2000
- [IVA94] Ivănescu, M.: “Roboți industriali”, Editura Universitaria, Craiova, 1994
- [JAL95] Jalobeanu, C; Rașa I: Probleme de calcul numeric și statistic în Mathcad, Editura Albastră, Cluj Napoca, 1995
- [KOV69] Kovacs, F.: Contribuții la elaborarea unei metode unitare de sinteză a mecanismelor”, Teză de disertație pentru obținerea titlului științific de doctor inginer, Timișoara, 1969.
- [KOV82] Kovacs, F., Cojocaru G.: Manipulatoare, roboți și aplicațiile lor industriale, Editura Facla, Timișoara, 1982
- [KOV92] Kovacs, Fr.; Rădulescu, C.: Roboți industriali, Universitatea Tehnică Timișoara, 1992, vol. 1 și 2
- [KOV94] Kovacs, F., Grigorescu S., Rădulescu C.: Sisteme de fabricație flexibilă robotizate, Lito Universitatea “Politehnica”, Timișoara, 1994.
- [KOV99] Kovacs, Fr.; Varga, Șt.; Tusz, Fr.: Fabrica viitorului, Editura Multimedia International, Arad, 1999
- [KOV00] Kovacs, Fr.; Varga, Șt.; Pau, C. V.: Introducere în robotică, Editura Printech, București, 2000
- [KOV01a] Kovacs F. W., Maci C., :Tehnici robotice de interacțiune om-mediul virtual, Revista “Robotică & Management”, vol.6, nr.1, 2001, Reșița-Timișoara.
- [KOV01b]Kovacs, F.V.: General Mathematic Model of Technological Processes Based on Relative Motions Proceedings of LSS 2001 9th IFAC / IFORS / IMACS / IFIP Symposium Bucharest, 2001
- [LEO91] Leon, Jean-Claude : Modelisation et construction de surfaces pour la CFAO, Editura Hermes, Paris, 1991
- [LYC98] Lyche, T., Schumaker, L.: Mathematical methods in computer aided geometric design, Academic Press Inc. , New York, 1989
- [MAC99] Maci, C. : Stadiul actual al cercetărilor și realizărilor privind digitizoarele 3D, Universitatea Politehnică Timișoara , referat doctorat numărul 1, 1999

- [MAC00a] Maci, C. : Modelarea geometrică a suprafețelor pe calculator, Universitatea Politehnică Timișoara , referat doctorat numărul 2, 2000
- [MAC00b] Maci, C. : Programarea roboților și a sistemelor robotizate utilizând digitizoare 3D, Universitatea Politehnică Timișoara , referat doctorat numărul 3, 2000
- [MAC02a] Maci, C. : RobSim - aplicație pentru programarea off-line a roboților în medii virtuale , sesiune de comunicări, Oradea 2002
- [MAC02b] Maci, C. : Applications of 3D digitizers in Robotics , Conferința Națională de Robotică, Craiova 2002
- [MAC02c] Maci, C. : Modelarea 3D a geometriei cuțitului pentru prelucrarea prin așchiere a pieselor de revoluție din industria lemnului , Sesiunea jubiliară de comunicări științifice cu participare internațională “30 ani de învățământ superior de stat în Arad”, Arad 2002
- [MIL01] Milenco, L., : Modelarea și simularea sistemelor mecanice” , Universitatea Politehnică Timișoara, Referat doctorat numărul 3, 2001
- [MIC98] Micșa, Dan : “Metode de generare, conversie, import, export, analiză și fabricație optimizată a suprafețelor discrete”, teză doctorat, 1998, conducător științific prof. dr. ing. George Drăghici
- [MIL98] Milenco, Luchin : Prognoza preciziei (exactității) de situare a robotului industrial în faza de proiectare a acestuia folosind metode de modelare - simulare, Referat doctorat nr.2, Conducător științific prof. dr. ing. Francisc Kovacs, Universitatea Politehnica Timișoara, 1998
- [MOC99] Mocian, I.: Proiectare tehnologică asistată de calculator în construcția de mașini, Editura Universității “Petru Maior”, Târgu Mureș, 1999
- [MOL96] Moldoveanu, Fl.; ș.a. :Grafica pe calculator , Editura Teora, București, 1996
- [MUL94] Mullineaux, Glen : CAD: Computational Concepts and Methods, Macmillan Publishing Company, New York, 1994
- [MUR96] Mureșan, Voichița : Modelator de solide pentru proiectarea asistată de calculator, teză de doctorat, 1996, coordonator științific Prof. dr. ing. George Savii
- [MUR62] Murgulescu, E. s.a. : Geometrie analitică și diferențială, Editura didactică și pedagogică, Bucuresti, 1962

- [NOF92] Nof, S.: Handbook of industrial robotics, Kriegel Publ. Malabar Flo. 1992
- [PAU81] Paul, R. P.: Robot Manipulators. Mathematics. Programming and Control; MIT Press Cambridge, 1981
- [PAU96] Păunescu, D. A.: Sisteme de conducere a roboților industriali Modelarea și planificarea traiectoriei , Universitatea tehnică Gh. Asachi, Iași, 1996
- [PER91] Peroche, B.; Les images de synthese , Ed. Hermes, Paris, 1991
- [PON91] Pontier, P.; La C.A.O. Robotique. Etudes D'Avant - Projects at Programmation Hors-Ligne Des Robots , Centre tehnicque des industries mecaniques, 1991
- [POR70] Portase C., Țăranu G. : Tehologia fabricării mobilei artistice , Editura didactică si pedagogică, București, 1970
- [POZ00] Pozna, C.: Comanda și controlul roboților industriali , Editura CIT, Brașov, 2000
- [PUR83] Purii, G. : Metrologie generală , reprografia Universității Tehnice Timișoara, 1983
- [RAN85] Ranky, P. G. Ho, CY: Robot Modeling. Control and Applications with Software , IFS Publ, UK, Springer V, Berlin;
- [RÎM80] Rîmbu, I. : Tehnologia Prelucrării Lemnului, Editura Tehnică,1980
- [SAV97] Savii, George: Grafica 3D pe calculatoare PC , Presa universitară română, 1997
- [SAV97] Savii, George: Bazele proiectării asistate de calculator , Editura Mirton, Timișoara, 1997
- [SAV00] Savii, G.; Luchin, M.: Modelare și simulare , Editura Eurostampa, Timișoara, 2000
- [SLA96] Slavici, T; Marinceu, D.; Dumitru, E. : Proiectarea asistată de calculator, volumul I, Universitatea Politehnica Timișoara, 1996
- [SLA98] Slavici, T.: Calculatoare personale Aplicatii la conducerea sistemelor tehnologice , Editura Mirton, Timișoara, 1998

[SLA94] Slavici, T. :Contribuții la programarea asistată de calculator a mașinilor–unelte cu comandă numerică în vederea prelucrării entităților geometrice complexe, teză doctorat, 1994, coordonator științific Prof. dr. ing. George Drăghici

[STA96] Stăncescu, C. : AutoLISP Manual de programare, Editura Fast 2000, București, 1996

[STO96] Stoicu-Tivadar, L.: Programarea roboților industriali și a mașinilor unelte cu comandă numerică, Universitatea Politehnica Timișoara, 1996

[TAN89] Tănăsescu, A.; Marinescu, D. s.a. : Grafică asistată Programe FORTRAN pentru reprezentări geometrice , Editura Tehnică, București, 1989

[TAR01] Țarcă, R.: Utilizarea informațiilor cu privire la situarea efectorului final achiziționate prin senzori în vederea conducerii în timp real a roboților, Teză de doctorat, Universitatea “Politehnica” Timișoara, 2001.

[VAS94] Vasilescu, E. ș.a. , Desen tehnic industrial , Editura Tehnică, București 1994

[VAN99] Vântu, M.: Programarea roboților industriali, Editura Aldus, Brașov, 1999

[VRÂ82] Vrânceanu, S.: Desen tehnic și ornamental în industria lemnului, Editura didactică și pedagogică, București, 1982

[WOH95] Wohlers, T. :The Challenge of 3D Digitizing, vol. 18, nr. 11, 1995

[VIN2000] Vince J. : Realitatea virtuală. Trecut, prezent și viitor, Ed. Tehnică, București, 2000.

[ZEI91] Zeid, I.: CAD/CAM Theory and practice , Editura Mc Graw-Hill Inc., New York, 1991

[ZIR96] Zirbel, Jay; Combs, St.: Utilizarea programului AutoCAD versiunea 13 pentru Windows, Editura Teora, București, 1996

*** CAD Report – colecția revistei 1992-1998

*** Fraunhofer magazine – colecția revistei 2000-2002

*** Hello CAD Fans – colecția revistei 1992 - 1998

*** MaxCAD - colecția revistei 2000-2002

Documentație CATIA V5R9 , Dassault.

<http://www.ideasin3d.com/ideastechn.html>

<http://www.ideasin3d.com/introdig.html>

<http://www.ideasin3d.com/indepth.htm>

<http://www.geomagic.com>

<http://www.cyberware.com/>

<http://www.faro.com/>

<http://www.immersion.com/default.shtml>

<http://www.laserdesign.com>

<http://www.polhemus.com>

<http://www.romer.com>

COMPANY	PRODUCT	TECHNOLOGY	ACCURACY	VOLUME/ ENVELOPE	COLOR	SPEED	URL
3D Digital Corp	3D Digital	Triangulation with laser					http://www.3ddigitalcorp.com/
3D Scanners	Model Marker, Replica, Reversa	Laser stripe/camera					http://www.3dscanners.com/
Aracor	Konospe 160/200	X-Ray/CT cone-beam	0.020mm	200mm diameter max. size	No	1024·1024·1024 scan in 16 hours	http://www.aracor.com/
Aracor	Konospe Micro-CT	X-Ray/CT cone-beam	0,002mm	40mm diameter max. size	No	1024·1024·1024 scan in 16 hours	http://www.aracor.com/
Arius 3D	Arius 3D foundation Scanner	Triangulare cu laser	<25 microns	100·100·100mm to 6000·2000·1400mm	Yes	3250 points per second	http://www.arius3d.com/
Breuckmann GmbH	opto TOP-HE 100	Triangulation with pattern projection	+/-15 um	80·60·50mm	available	1,300,000 points per second	http://www.breuckmann.com/
Breuckmann GmbH	opto TOP-HE 100	Triangulation with pattern projection	+/-20 um	160·125·100mm	available	1,300,000 points per second	http://www.breuckmann.com/
Breuckmann GmbH	opto TOP-HE 400	Triangulation with pattern projection	+/-40 um	320·250·200mm	available	1,300,000 points per second	http://www.breuckmann.com/
Breuckmann GmbH	opto TOP-HE 600	Triangulation with pattern projection	+/-50 um	480·380·300mm	available	1,300,000 points per second	http://www.breuckmann.com/
CGI	CSS1000 Benchtop	Camera/slicing destructive	0.0008inch				http://www.reverse-eng.com/
Cyberware	Desktop 3D Scanner Bundle	Laser video/color	0.002 - 0.008 inch	250·150·75mm	No	14.580 points per second	http://www.cyberware.com/
Cyberware	Head&Face Color 3D Scanner Bundle	Laser video/color	0.004 inch	1000·260·300mm	Yes	4.580 points per second	http://www.cyberware.com/
Cyberware	Mini Model Stop Color 3D Scanner Bundle	Laser video/color	0.002 – 0.008 inch	350·130·150mm	Yes	4.580 points per second	http://www.cyberware.com/
Cyberware	Model 3030 Color 3D Scanhead	Laser video/color	0.0016 – 0.004 inch	130·170·150mm to 260·340·300mm	Yes	14.580 points per second	http://www.cyberware.com/

COMPANY	PRODUCT	TECHNOLOGY	ACCURACY	VOLUME/ ENVELOPE	COLOR	SPEED	URL
Cyberware	Model Shop Color 3D Scanner Bundle	Laser video/color	0.004 inch	1000·260·300mm	Yes	14,580 points per second	http://www.cyberware.com/
Cyberware	Whole Body Color3D Scanner Bundle	Laser video/color	0.019 inch	Cylindrical 1200mm diameter, 2000mm depth	Yes	60,000 points per second	http://www.cyberware.com/
Cyra (Lecia subsidiary)	Cyrax 3D Laser Scanning System	Laser dot/video camera					http://www.cyra.com/
Digibotics	Digibot	Laser/triangulation					http://www.digibotics.com/
Faro Technologies	Faro Arm 10ft	Mechanical probe/encoders	0.0033 in.	n/a	n/a	n/a	http://www.digibotics.com/
Faro Technologies	Faro Arm 12ft	Mechanical probe/encoders	0.0047 in.	n/a	n/a	n/a	http://www.faro.com/
Faro Technologies	Faro Arm 4ft	Mechanical probe/encoders	0.001 in.	n/a	n/a	n/a	http://www.faro.com/
Faro Technologies	Faro Arm 6ft	Mechanical probe/encoders	0.0016 in.	n/a	n/a	n/a	http://www.faro.com/
Faro Technologies	Faro Arm 8ft	Mechanical probe/encoders	0.002 in.	n/a	n/a	n/a	http://www.faro.com/
Genex	EI 3D Digitizer	Projected white-light pattern /CCD camera	0.001 – 0.0098 in.	59·48 · 32mm to 250·200·200mm	Yes	442,368 points in < 1 second	http://www.genextech.com/
Genex	Rainbow 100	Projected white-light pattern /CCD camera	0.0039 in.	100·80·80mm	Yes	442,368 points in < 1 second	http://www.genextech.com/
Genex	Rainbow 25	Projected white-light pattern /CCD camera	0.001 in.	32·52·20mm	Yes	442,368 points in < 1 second	http://www.genextech.com/
Genex	Rainbow 250	Projected white-light pattern /CCD camera	0.0098 in.	250·200·200mm	Yes	442,368 points in < 1 second	http://www.genextech.com/
Genex	Rainbow 50	Projected white-light pattern /CCD camera	0.002 in.	59·48·32mm	Yes	442,368 points in < 1 second	http://www.genextech.com/
GOM mbH	ATOS II	Structured white light /CCD cameras, Photogrammetry	0.005mm to 0.02mm	35·28·20mm to 1200·960·960mm	No	1,300,000 points in 7 second	http://www.gom.com/

COMPANY	PRODUCT	TECHNOLOGY	ACCURACY	VOLUME/ ENVELOPE	COLOR	SPEED	URL
Immersion	LightScribe 3D	Mechanical probe	0.008 in. (0.2mm)	300mm·300mm·300mm	Yes	points input manually	http://www.immersion.com/default.shtml
Immersion	MicroScribe 3D	Mechanical probe	0.015 in. (0.38mm)	50" (1.27m)	No	1000 points per second	http://www.immersion.com/default.shtml
Immersion	MicroScribe 3DL	Mechanical probe	0.017 in. (0.43 mm)	66" (1.67m)	No	1000 points per second	http://www.immersion.com/default.shtml
Immersion	MicroScribe 3DLX	Mechanical probe	0.012in.(0.30 mm)	66" (1.67m)	No	1000 points per second	http://www.immersion.com/default.shtml
Immersion	MicroScribe 3DX	Mechanical probe	0.009 in(0.23mm)	50" (1.27m)	No	1000 points per second	http://www.immersion.com/default.shtml
INTECU	CyLan3D	Triangulation with laser	0.01 mm	10·10·10mm to 375·350·350 mm	No	4,000 points per second	http://www.intecu.de
INTECU	CyLan701	Triangulation with laser	0.01 to 0.05 mm	max.1400·1600·1600mm	No	4,000 points per second	http://www.intecu.de
Kreon Industries	KLS	Laser/camera			No	20,000 points per second	http://www.kreon3d.com
Laser Design, Inc.	Surveyor, AM-Series	Laser line/CCD in probe	+/-25 microns	100mm depth of field	No	Up to 14,000 points per second	http://www.laserdesign.com
Laser Design, Inc.	Surveyor, CS - Series	Laser line/CCD in probe	3.0+3.5 L to 10 +15 L<40	700·550·500mm to 300·1000·1200mm	No	Up to range of 520mm to 750 mm per second	http://www.laserdesign.com
Laser Design, Inc.	Surveyor, D M-Series	Laser line/CCD in probe	0.00038 in.	400·500·350mm	No	Up to 15,000 points per second	http://www.laserdesign.com
Laser Design, Inc.	Surveyor, DS- Series	Laser line/CCD in probe	0.0006 in. to 0.00124 in.	500·500·400mm to 1500·2500·1000mm	No	Up to range of 250mm to 610 mm points per second	http://www.laserdesign.com
Laser Design, Inc.	Surveyor, PS - Series	Laser line/CCD in probe	0.00047 in. to 0.0390 in.	75 mm to 600 depth of field	No	Up to 15,000 points per second	http://www.laserdesign.com

COMPANY	PRODUCT	TECHNOLOGY	ACCURACY	VOLUME/ENVELOPE	COLOR	SPEED	URL
Laser Design, Inc.	Surveyor, RE-Series	Laser line/CCD in probe	0.004 in.	200-250-125mm to 1420-2435-520mm	Yes	Up to range of 60mm to 300 mm points per second	http://www.laserdesign.com
Mensi	MENSI GS100	Time of flight	0.0039 in.	Field of view: 360-60degrees	Yes	1000 points per second	http://www.mensi.com
Mensi	MENSI S10	Triangulation with laser	0.00083 in.	320-46 degrees	Yes	Up to 100 points per second	http://www.mensi.com
Mensi	MENSI S25	Triangulation with laser	0.0024 in.	320-46 degrees	Yes	Up to 100 points per second	http://www.mensi.com
Mertricvision	MV-200-24	Laser dot CCD camera	2-3 meters +/- 50 microns 2 sigma; 3-10 meters +/- 7 PPM 2 sigma; 0-24 meters +/- 10 PPM 2 sigma	24 meter radius from scanner	No	1000 points per second	http://www.metricvision.com
Minolta	VIVID 700	Laser stripe/CCD color	0.75mm	190 - 190mm to 400- 400mm	Yes	40,000 points in 0,6 second	http://www.minolta3d.com
Minolta	VIVID 300	Laser stripe/CCD color	0.25mm	100-100mm to 1100 -1100mm	Yes	40,000 points in 0,6 second	http://www.minolta3d.com
Minolta	VIVID 900	Laser stripe/CCD color	0.05mm	100-80-40mm to 1200-900-400mm	Yes	307,200 points in 2,5 seconds to 0,5 seconds (Dual Mode)	http://www.minolta3d.com
Nextec	Hawk, Vizprobe	Laser dot/LED	Single point: 6 micron at 1 sigma			50 points per second	http://www.nextec-viz.com/index.html
Optimet	Mini-Conoscan 3000	Interferometry	2um to 50 um	125-1250.6 - mm to 125-125 70 - mm	Yes	1,000 points per second	http://www.optimet.com/
Perceptron	ScanWorks Lite	Laser scanner	+/-50 micron, 3 sigma	based on size of CMM arm	No	23,000 points per second	http://www.scanworks.perceptron.com/

COMPANY	PRODUCT	TECHNOLOGY	ACCURACY	VOLUME/ENVE LOPE	COLOR	SPEED	URL
Polhemus	Fast SCAN	Laser stripes, EM probe	0.04in. at 8 in.	n/a	No	15,000 points per second	http://www.polhemus.com
Renishaw	Cyclone	Mechanical probe, laser option					http://www.renishaw.com
Renishaw	Renscan	Mechanical probe, laser option					http://www.renishaw.com
Renishaw	Retroscan	Mechanical probe, laser option					http://www.renishaw.com
Riegl	LMS Z 210	Time of flight	10-25 mm	360 degrees X 300degrees	Yes	10,000 points per second	http://www.riegl.com/scanner_menu.html
Riegl	LMS Z 420	Time of flight	5-20 mm	360 degrees X 300degrees	Yes	3,000 la 8,000 points per second	http://www.riegl.com/scanner_menu.html
Riegl	LPM 100 VSH	Time of flight	25 mm	360 degrees X 300degrees	No	1000 points per second	http://www.riegl.com/scanner_menu.html
Riegl	LPM 25 HA	Time of flight	4-8 mm	360 degrees X 300degrees	No	1000 points per second	http://www.riegl.com/scanner_menu.html
Riegl	LPM 2K	Time of flight	25-75 mm	360 degrees X 300degrees	No	1000 points per second	http://www.riegl.com/scanner_menu.html
Riegl	LPM 300VHS	Time of flight	50mm	360 degrees X 300degrees	No	1000 points per second	http://www.riegl.com/scanner_menu.html
Roland Digital Group	PIX-4/30, PIX-3,LPX-250,MDX-15/20	Piezo mechanical probe					http://www.rolanddga.com/products/3D/scanners/default.asp
Romer, Inc.	1000i,3000i	Mechanical probe /encoders	0.0008 inch				http://www.romer.com
ShapeGrabber	AI2400, Foot Scanner, Tabletop, Portable, Gantry	Laser stripe/CCD camera					http://www.shapegrabber.com/
Steinbichler Optotechnik	COMET C100VZ	White light fringe projection	+/-0.02 mm	100-80 mm	No	Up to 6666 points per second	http://www.steinbichler.de/

COMPANY	PRODUCT	TECHNOLOGY	ACCURACY	VOLUME/ENVE LOPE	COLOR	SPEED	URL
Steinbichler Optotechnik	COMET C250	White light fringe projection	+/-0.06mm	280·180 mm	No	Up to 6666 points per second	http://www.steinbichler.de/
Steinbichler Optotechnik	COMET C250VZ	White light fringe projection	+/-0.04 mm	245·200 mm	No	Up to 6666 points per second	http://www.steinbichler.de/
Steinbichler Optotechnik	COMET C250/400VZ	White light fringe projection	+/-0.04-0.07 mm	245·200 or 420·340 mm	No	Up to 6666 points per second	http://www.steinbichler.de/
Steinbichler Optotechnik	COMET C400	White light fringe projection	+/-0.1 mm	430 ·330 mm	No	Up to 6666 points per second	http://www.steinbichler.de/
Steinbichler Optotechnik	COMET C400VZ	White light fringe projection	+/-0.07 mm	420 340· mm	No	Up to 6666 points per second	http://www.steinbichler.de/
Steinbichler Optotechnik	COMET C50	White light fringe projection	+/-0.02 mm	45·35 mm	No	Up to 6666 points per second	http://www.steinbichler.de/
Steinbichler Optotechnik	COMET C75	White light fringe projection		78·58 mm	No	Up to 6666 points per second	http://www.steinbichler.de/
Steinbichler Optotechnik	COMET T- Scan	Triangulation with laser	0.1-0.2mm	3m depth	No	Up to 6666 points per second	http://www.steinbichler.de/
TC2	3D Body Measurement system	White Light	1mm	1.1 x 1.0x2.0m	No	15,000 points per second	http://www.tc2.com
Vitana corp	PixelINK	Triangulation with laser					http://www.pixelink.com/

Notă : Informațiile din anexa 1 au fost preluate de pe site-ul www.geomagic.com

```

:*****
:Nume fisier: Sup.lsp
:autor: Maci Claudiu
:*****
:Program pentru obtinerea retelelor tip 3D Mesh din poliliniile trasate pe o suprafata cu
:Digitizare 3D
:Verexurile poliliniilor sunt extrase dintr-un fisier unde sunt puse de programul care preia
:datele digitizate sau de functia PreLin . Se cauta intersecțiile dintre poliliniile după care se
:redesenează poliliniile astfel încât să se intersecteze în mod real. Apoi se întrerup
:poliliniile în punctele de intersecție , după care se ordonează în liste de câte 4 segmente
:de polilinie ce se intersectează două câte două și aceste liste sunt transferate funcției
:edgesurf pentru calcularea și desenarea suprafeței în AutoCAD
:*****
:Variabile globale
: G_poli Lista cu subliste conținând vertexurile poliliniilor trasate pe sup List
:*****

:*****Start F_readfis*****
:Functie citire date din fisier cu virgula ca separator de forma :nr_polilinie.nr_vertex.x,y,z
:și formare lista cu elementele ((polilinie1 (x y z) (x y z) ...) (polilinie2 ...) ...)
: Numarul primei polilinii trebuie sa fie
:*****
:Valoare returnată
: G_poli Lista cu subliste conținând vertexurile poliliniilor trasate pe sup List
:*****
(defun F_readfis ( / numefis fis linie_fis lung_sir date_vertex i sir litera element lista_out)

: Introducerea de utilizator a numelui fisierului de unde se preiau coordonatele
: vertexurilor poliliniilor
(setq numefis
(getstring
"\nNume fisier (fara extensie) unde sunt stocate coordonatele vertexurilor : "))
(setq numefis (strcat numefis ".txt"))
(setq fis (open numefis "r"))

: Formarea listei de iesire
(setq lista_out ())
(setq nr_polil_veche 1); Numarul primei polilinii trebuie sa fie 1 !!!
(setq polilinie ())

: Preluarea cite unei linii din fisier
(while (/= (setq linie_fis (read-line fis)) nil)
(setq lung_sir (strlen linie_fis))
(setq date_vertex ())
(setq i 1)
(setq sir ""))

: Preluarea cite unui caracter din linia citita din fisier
(while (<= i lung_sir)
(setq litera (substr linie_fis i 1))
(if (/= litera ",")
(setq sir (strcat sir litera))
(progn
(setq element (atof sir))
(setq date_vertex (append date_vertex (list element)))
(setq sir ""))
)
)
(setq i (+ i 1))
)

(setq date_vertex (append date_vertex (list (atof sir))))

: Se extrage numarul poliliniilor
(setq nr_polil (fix (car date_vertex)))
(setq date_vertex (cdr date_vertex))
(if (= nr_polil nr_polil_veche)

```

```

    (setq polilinie (append polilinie (list date_vertex)))
    (progn
      (setq lista_out (append lista_out (list polilinie)))
      (setq polilinie (list date_vertex))
    )
  )
)
(setq nr_polil_veche nr_polil)
)

:adaugarea ultimei polilinii citite din fisier
(setq lista_out (append lista_out (list polilinie)))

(print "poliliniile trasate pe suprafata G_poli")
(print G_poli)

(setq G_poli lista_out)
):close defun
;*****End F_readfis*****

;*****Start F_inters*****
;   Functie pentru cautarea intersectiilor dintre polilinii pt suprafete deschise
;*****
;Input
;   G_poli      Lista cu subliste continind vertexurile poliliniilor trasate pe
;               suprafata ((polilinie1 (x y z) (x y z) ...) (polilinie2 ...) ...) List
;Output
;   G_inters    Lista cu numarul poliliniilor si coordonatele vertexului
;               de intersectie      de forma
;               ((nr_lin1 nr_vert_pe_lin1 nr_lin2 nr_vert_pe_lin2 (coord inters)..)
;*****
(defun F_inters ( lista_pol / dist_epsilon nr_poli i j prima_pol doua_pol nr_prima_pol
                nr_doua_pol elem_cu_dist_min vertex_prima vertex_doua
                distanta elem_lista_dist m n elem_cu_dist_min coord_inters
                elem_cu_d_min_cord )

  (initget (+ 1 2 ))
  (setq dist_epsilon
    (getreal "\n Distanța de la care se considera ca doua polil se intersect [mm]:"))
  (setq nr_poli (length lista_pol))
  (setq G_inters ())
  (setq i 1)
  (while (<= i nr_poli)
    (setq j 1)
    (while (<= j nr_poli)
      (if (/= i j)
        (progn
          (setq prima_pol (nth (- i 1) lista_pol))
          (setq doua_pol (nth (- j 1) lista_pol))
          (setq nr_prima_pol (length prima_pol))
          (setq nr_doua_pol (length doua_pol))

          :initializare element pt distanta minima
          (setq elem_cu_dist_min (list 0 0 0 0 10000))
          (setq m 1)
          (while (<= m nr_prima_pol)
            (setq n 1)
            (while (<= n nr_doua_pol)

              ;vertex apartinind primei polilinii
              (setq vertex_prima (nth (- m 1) prima_pol))

              ;vertex apartinind celei de-a doua polilinii
              (setq vertex_doua (nth (- n 1) doua_pol))

              ;calculul distantei dintre vertexurile celei doua polilinii
              (setq distanta (distance vertex_prima vertex_doua))

              ;formarea unui element continind numerele de ordine a vertexurilor si distanta
              (setq elem_lista_dist (list i m j n distanta))
            )
          )
        )
      )
    )
  )

```



```

:Se parcurge lista L_noduri
(while (> (length L_noduri) 0)
  (setq Ramuri_curent ())
  (setq curent (nth 0 L_noduri))
  (setq nr_lin_m (nth 0 curent))
  (setq nr_vert_t (nth 1 curent))
  (setq nr_lin_n (nth 2 curent))
  (setq nr_vert_s (nth 3 curent))

: Pentru fiecare nod se apeleaza functia care cauta nodurile vecine cu cel curent
  (setq nod_Lm.Vt+1 (F_caut_nod nr_lin_m nr_vert_t 1 L_noduri))
  (setq nod_Lm.Vt-1 (F_caut_nod nr_lin_m nr_vert_t -1 L_noduri))
  (setq nod_Ln.Vs+1 (F_caut_nod nr_lin_n nr_vert_s 1 L_noduri))
  (setq nod_Ln.Vs-1 (F_caut_nod nr_lin_n nr_vert_s -1 L_noduri))

: Pentru fiecare din nodurile gasite (diferite de nil) se cauta nodurile invecinate
: pe noile linii
  (if (/= nod_Lm.Vt+1 nil)
    (progn
      (setq Lin_a (nth 2 nod_Lm.Vt+1))
      (setq nod_a (nth 3 nod_Lm.Vt+1))
      (setq vec1_nod_Lm.Vt+1 (F_caut_nod Lin_a nod_a 1 L_noduri))
      (if (/= vec1_nod_Lm.Vt+1 nil)
        (setq Ramuri_curent (append Ramuri_curent
          (list (list curent nod_Lm.Vt+1 vec1_nod_Lm.Vt+1))))))
      (setq vec2_nod_Lm.Vt+1 (F_caut_nod Lin_a nod_a -1 L_noduri))
      (if (/= vec2_nod_Lm.Vt+1 nil)
        (setq Ramuri_curent (append Ramuri_curent
          (list (list curent nod_Lm.Vt+1 vec2_nod_Lm.Vt+1))))))
    )
  )
  (if (/= nod_Lm.Vt-1 nil)
    (progn
      (setq Lin_b (nth 2 nod_Lm.Vt-1))
      (setq nod_b (nth 3 nod_Lm.Vt-1))
      (setq vec1_nod_Lm.Vt-1 (F_caut_nod Lin_b nod_b 1 L_noduri))
      (if (/= vec1_nod_Lm.Vt-1 nil)
        (setq Ramuri_curent (append Ramuri_curent
          (list (list curent nod_Lm.Vt-1 vec1_nod_Lm.Vt-1))))))
      (setq vec2_nod_Lm.Vt-1 (F_caut_nod Lin_b nod_b -1 L_noduri))
      (if (/= vec2_nod_Lm.Vt-1 nil)
        (setq Ramuri_curent (append Ramuri_curent
          (list (list curent nod_Lm.Vt-1 vec2_nod_Lm.Vt-1))))))
    )
  )
  (if (/= nod_Ln.Vs+1 nil)
    (progn
      (setq Lin_c (nth 2 nod_Ln.Vs+1))
      (setq nod_c (nth 3 nod_Ln.Vs+1))
      (setq vec1_nod_Ln.Vs+1 (F_caut_nod Lin_c nod_c 1 L_noduri))
      (if (/= vec1_nod_Ln.Vs+1 nil)
        (setq Ramuri_curent (append Ramuri_curent
          (list (list curent nod_Ln.Vs+1 vec1_nod_Ln.Vs+1))))))
      (setq vec2_nod_Ln.Vs+1 (F_caut_nod Lin_c nod_c -1 L_noduri))
      (if (/= vec2_nod_Ln.Vs+1 nil)
        (setq Ramuri_curent (append Ramuri_curent
          (list (list curent nod_Ln.Vs+1 vec2_nod_Ln.Vs+1))))))
    )
  )
  (if (/= nod_Ln.Vs-1 nil)
    (progn
      (setq Lin_d (nth 2 nod_Ln.Vs-1))
      (setq nod_d (nth 3 nod_Ln.Vs-1))
      (setq vec1_nod_Ln.Vs-1 (F_caut_nod Lin_d nod_d 1 L_noduri))
      (if (/= vec1_nod_Ln.Vs-1 nil)
        (setq Ramuri_curent (append Ramuri_curent
          (list (list curent nod_Ln.Vs-1 vec1_nod_Ln.Vs-1))))))
    )
  )

```

```

(setq vec2_nod_Ln.Vs-1 (F_caut_nod Lin_d nod_d -1 L_noduri))
(if (/= vec2_nod_Ln.Vs-1 nil)
  (setq Ramuri_curent (append Ramuri_curent
    (list (list curent nod_Ln.Vs-1 vec2_nod_Ln.Vs-1))))))
)
)

:Eliminare 2 noduri din lista L_noduri (nodurile i->j si j->i )
(setq L_noduri (F_elim_elem curent L_noduri))
(setq L_noduri
  (F_elim_elem (list nr_lin_n nr_vert_s nr_lin_m nr_vert_t (nth 4 curent)) L_noduri))

;cautare noduri comune in sublistele din ramuri_curent si formare liste cu colturi de
;petice
(setq G_nod_petice (append G_nod_petice (F_form_petice ramuri_curent)))
)

(print "G_nod_petice")
(print G_nod_petice)
(print "Numar petice")
(print (length G_nod_petice))

);close defun
;*****End F_caut_petice*****

;*****Start F_form_petice*****
; Functia preia o lista cu subliste continind noduri si verifica daca exista doua
; noduri comune in doua subliste pe ultima pozitie (Deci algoritmul nu e aplicat
; Pentru petice cu numar impar de laturi 3,5,7) .
; Daca sunt gasite doua noduri identice pe ultima pozitie a celei de-a doua subliste
; atunci se formeaza un petic prin lipirea primei subliste cu inversul celei de-a doua
; subliste fara elementul comun . Trebuie verificate toate combinatiile de cite 2 subliste,
; evitindu-se cazul ca cele doua subliste sa fie aceeasi lista
;*****
(defun F_form_petice (lista / i j subl_a subl_b )

  (setq i 0)
  (setq petice ())
  (while (< i (length lista))
    (setq j 0)
    (while (< j (length lista))
      (if (/= i j)
        (progn
          (setq subl_a (nth i lista))
          (setq subl_b (nth j lista))
          (setq ultim_s_a (nth (- (length subl_a) 1) subl_a))
          (setq ultim_s_b (nth (- (length subl_b) 1) subl_b))
          (setq inv_ultim_s_b (list (nth 2 ultim_s_b) (nth 3 ultim_s_b)
            (nth 0 ultim_s_b) (nth 1 ultim_s_b) (nth 4 ultim_s_b)))
          (if (and
              (= (nth 0 ultim_s_a) (nth 0 inv_ultim_s_b))
              (= (nth 1 ultim_s_a) (nth 1 inv_ultim_s_b))
              (= (nth 2 ultim_s_a) (nth 2 inv_ultim_s_b))
              (= (nth 3 ultim_s_a) (nth 3 inv_ultim_s_b))
            )
            (progn
              (setq petice (append petice (list (append subl_a (cdr (reverse subl_b))))))
            )
          )
        )
      (setq j (+ j 1))
    )
    (setq i (+ i 1))
  )
  petice
);close defun
;*****End F_form_petice*****

```

```

*****Start F_caut_nod*****
:   Functia cauta un nod intr-o lista dupa primele doua elemente ( linia si nodul )
*****
:Argumente
:   Nrlin      Numarul liniei                int
:   Nrnod      Numarul de ordine a vertexului in linie          int
:   sens       Sensul de parcurgere al listei                    int
:   Lista      Lista cu vertexuri                                list
:Output
:   nodCoordonatele vertexului                                list
*****
(defun F_caut_nod ( Nrlin Nrnod sens Lista / i out curent mai_mare mai_mic )
  (setq i 0)
  (setq out 0)
  (while (= out 0)
    (setq curent (nth i lista))
    (setq linie (nth 0 curent))
    (setq vertex (nth 1 curent))
    (if (and (= linie Nrlin) (= sens -1) (< vertex Nrnod))
      (progn
        (setq cautat curent)
        (setq out 1)
      )
    )
    (if (and (= linie Nrlin) (= sens 1) (> vertex Nrnod))
      (progn
        (setq cautat curent)
        (setq out 1)
      )
    )
    (setq i (+ i 1))
    (if (and (= i (length lista)) (= out 0))
      (progn
        (setq cautat nil)
        (setq out 1)
      )
    )
  )
  cautat
)
*****End F_caut_nod*****

*****Start F_elim_elem*****
:   Functia elimina un element dat ca argument dintr-o lista data ca argument
*****
;argumente
;   element    Elementul ce trebuie eliminat
;   Lista      Lista din care se elimina elementul
;Output
;   Lista_fara  Lista fara elementul respectiv
*****
(defun F_elim_elem ( element Lista)
  (setq Lista_fara (append (reverse (cdr (member element (reverse Lista))))
    (cdr (member element Lista))))
)
*****End F_elim_elem*****

*****Start F_coord_petice*****
:   Functia preia nodurile ce formeaza un petec din lista G_nod_petice si formeaza
:   o lista cu subliste pentru un petec , fiecare sublista continind la rindul ei
:   patru subliste cu coordonatele vertexurilor cite unei laturi .
*****
:Argumente
:   G_nod_petice ((lista noduri petec1) (lista noduri petec2) ...) list
:Output
:   Sunt generate peticele
*****

```

```

(defun F_coord_petice ( nod_pet / )

  (setq i 0)
  (setq nr_petice (length nod_pet))

  (while (< i nr_petice)

    :extragerea nodurilor unui petec
    (setq nod_pet_curent (nth i nod_pet))

    ;Fomarea muchiilor din cite 2 noduri succesive
    (setq muchie1 (list (nth 0 nod_pet_curent) (nth 1 nod_pet_curent)))
    (setq muchie2 (list (nth 1 nod_pet_curent) (nth 2 nod_pet_curent)))
    (setq muchie3 (list (nth 2 nod_pet_curent) (nth 3 nod_pet_curent)))
    (setq muchie4 (list (nth 3 nod_pet_curent) (nth 4 nod_pet_curent)))

    ;formarea listelor cu coordonatele muchiilor
    (setq coord_muc1 (F_coord_muchie muchie1))
    (setq coord_muc2 (F_coord_muchie muchie2))
    (setq coord_muc3 (F_coord_muchie muchie3))
    (setq coord_muc4 (F_coord_muchie muchie4))

    (print "*****")
    (print coord_muc1)
    (print coord_muc2)
    (print coord_muc3)
    (print coord_muc4)
    (print "*****")

    ;Desenare muchii
    (setq latura1 (F_desen_poli coord_muc1))
    (setq latura2 (F_desen_poli coord_muc2))
    (setq latura3 (F_desen_poli coord_muc3))
    (setq latura4 (F_desen_poli coord_muc4))

    :desenare suprafata petec cu comanda "Edgesurf"
    (command "edgesurf" latura1 latura2 latura3 latura4)

    (setq i (+ i 1))
  )
)
;*****End F_coord_petice*****

;*****Start F_coord_muchie*****
; Funcția Preia o lista a unei muchii formata din doua noduri si returneaza o lista
; continind vertexurile unei muchii a petecului
; Funcția utilizeaza coordonatele tuturor vertexurilor stocate in G_poli de forma
; ((polilinie1 (x y z) (x y z) ...) (polilinie2 ...) ... )
;*****
(defun F_coord_muchie ( nod_muc / nod1 nod2 Linie1_nod1 Linie2_nod1 Linie1_nod2 Linie2_nod2
                    vert_capat1 vert_capat2 vert_incep vert_term muc_fara_capete
                    i vert_muchie)

  (setq list_vert G_poli)

  ;Cautare liniei pe care sunt ambele noduri
  (setq nod1 (nth 0 nod_muc))
  (setq nod2 (nth 1 nod_muc))
  (setq Linie1_nod1 (nth 0 nod1))
  (setq Linie2_nod1 (nth 2 nod1))
  (setq Linie1_nod2 (nth 0 nod2))
  (setq Linie2_nod2 (nth 2 nod2))
  (if (= Linie1_nod1 Linie1_nod2)
    (progn
      (setq linie Linie1_nod1)
      (setq vert_capat1 (nth 1 nod1))
      (setq vert_capat2 (nth 1 nod2))
    )
  )
)

```



```

)
(if (= Linie1_nod1 Linie2_nod2)
  (progn
    (setq linie Linie1_nod1)
    (setq vert_capat1 (nth 1 nod1))
    (setq vert_capat2 (nth 3 nod2))
  )
)
(if (= Linie2_nod1 Linie1_nod2)
  (progn
    (setq linie Linie2_nod1)
    (setq vert_capat1 (nth 3 nod1))
    (setq vert_capat2 (nth 1 nod2))
  )
)
(if (= Linie2_nod1 Linie2_nod2)
  (progn
    (setq linie Linie2_nod1)
    (setq vert_capat1 (nth 3 nod1))
    (setq vert_capat2 (nth 3 nod2))
  )
)

;Formarea listei cu punctele cuprinse intre cele doua noduri fara nodurile capete
(if (< vert_capat1 vert_capat2)
  (progn
    (setq vert_incep vert_capat1)
    (setq vert_term vert_capat2)
  )
  (progn
    (setq vert_incep vert_capat2)
    (setq vert_term vert_capat1)
  )
)

;Formarea Listei de vertexuri fara capete
(setq i vert_incep)
(setq muc_fara_capete ())
(while (< i (- vert_term 1))
  (setq muc_fara_capete (append muc_fara_capete (list (nth i (nth (- linie 1) list_vert))))))
  (setq i (+ i 1))
)

;Formarea listei unde capetele au fost inlocuite cu nodurile calculate
(setq vert_muchie (append (list (nth 4 nod1)) muc_fara_capete (list (nth 4 nod2))))

vert_muchie
)
;*****End F_coord_muchie*****

;*****Start F_desen_poli*****
;Functia preia o lista de vertexuri si deseneaza o polilinie
;*****
;Argument
; Lista_pol Forma : ((coord vert 1) (coord vert 2) ...List
;Output
; Desenarea polilinie
;*****
(defun F_desen_poli ( Lista_pol / i a point)

(setq i 0)
(ENTMAKE)
(setq a '((0 . "POLYLINE") (5 . "35") (100 . "AcDbEntity") (67 . 0) (100 . "AcDb3dPolyline")
(66 . 1) (10 0.0 0.0 0.0) (70 . 8) (40 . 0.0) (41 . 0.0) (210 0.0 0.0 1.0) (71 . 0) (72 . 0)
(73 . 0) (74 . 0) (75 . 0)))

;Crearea entitatii principale in baza de date
(entmake a)

```

```

:adaugare vertexuri
(while (< i (length Lista_pol))
  (setq point (nth i Lista_pol))
  (entmake
    (list
      (cons 0 "vertex")
      (cons 70 32)
      (cons 10 point)
    )
  )
  (setq i (+ i 1))
)

:Semnalizarea incheierii polilinii
(entmake '((0 . "seqend")))
(setq nume_pol (entlast))
)
:*****End F_desen_poli*****

:*****Start Sup*****
:  FUNCTIA PRINCIPALA
:  Functie obtinere rețele tip 3D Mesh din polilinii trasate pe o suprafața
:*****
(defun C:Sup ()

:Formarea listei cu vertexurile citite din fisier
(F_readfis)

:Cautarea intersectiilor
(F_inters G_poli)

:Functie pentru cautarea peticelor
(F_caut_petice G_inters)

:Desenare petice
(F_coord_petice G_nod_petice)

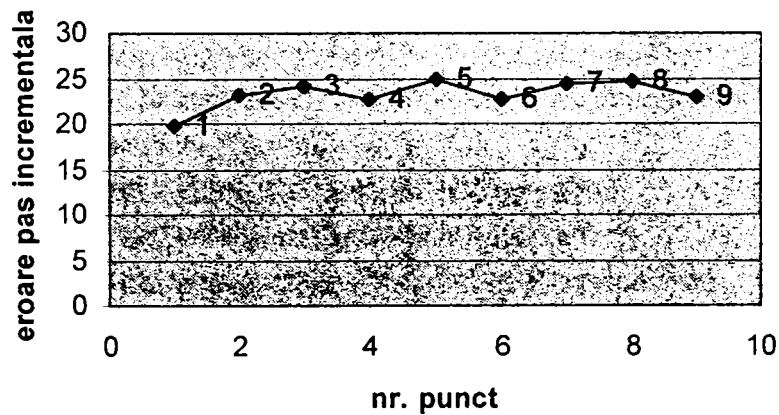
);close defun
:*****End Sup*****

```

Anexa 3 Incercări experimentale ale digitizorului 3D RRR

Testul de repetabilitate								
Nr.Punct	Nr. Masurare	x	y	z	dx	dy	dz	dist. Pt. anterior
1	1	135.43627	54.71535	5.301255				
	2	135.74611	55.11629	5.541952	-0.30983	-0.40094	-0.2407	0.646139193
	3	135.63402	54.79524	5.286461	0.112088	0.321047	0.255491	0.467659971
	4	135.64432	54.7994	5.556374	-0.0103	-0.00416	-0.26991	0.011861713
	5	135.43627	54.71535	5.301255	0.208046	0.084049	0.255119	0.239607656
								deviatie max
			Valoare max		-0.30983	-0.40094	-0.26991	0.646139193
Nr.Punct	Nr. Masurare	x	y	z	dx	dy	dz	dist. Pt. anterior
2	1	84.273234	117.4775	5.410024				
	2	84.353573	117.1573	5.426307	-0.08034	0.320244	-0.01628	0.459963808
	3	83.979961	116.6384	5.476271	0.373612	0.518904	-0.04996	0.823473759
	4	83.776276	116.7848	5.476271	0.203685	-0.14638	0	0.290420903
	5	83.645406	116.6023	5.225636	0.130869	0.182433	0.250635	0.289292741
								deviatie max
			Valoare max		0.373612	0.518904	0.250635	0.823473759
Nr.Punct	Nr. Masurare	x	y	z	dx	dy	dz	dist. Pt. anterior
3	1	141.75736	53.27881	5.809903				
	2	141.58801	52.51129	5.829621	0.169359	0.767515	-0.01972	1.098562783
	3	142.43499	52.68408	5.790931	-0.84699	-0.17279	0.03869	0.881531997
	4	142.92763	52.58292	5.772704	-0.49264	0.101162	0.018227	0.512994431
	5	142.83565	52.83227	5.772704	0.091985	-0.24936	0	0.364443873
								deviatie max
			Valoare max		-0.84699	0.767515	0.03869	1.098562783
Nr.Punct	Nr. Masurare	x	y	z	dx	dy	dz	dist. Pt. anterior
4	1	208.59105	122.3687	12.39505				
	2	207.95006	122.2367	11.84208	0.640987	0.132	0.552966	0.667616713
	3	208.59105	122.3687	12.39505	-0.64099	-0.132	-0.55297	0.667616713
	4	208.41566	122.2658	12.32905	0.175393	0.102893	0.065994	0.227896318
	5	208.48419	122.5507	12.39505	-0.06853	-0.28486	-0.06599	0.408645484
								deviatie max
			Valoare max		0.640987	-0.28486	0.552966	0.667616713
Nr.Punct	Nr. Masurare	x	y	z	dx	dy	dz	dist. Pt. anterior
5	1	136.29553	54.92419	5.783971				
	2	135.89979	54.76472	5.811884	0.395738	0.159474	-0.02791	0.455491402
	3	135.53156	54.4789	5.301255	0.368236	0.285813	0.51063	0.546786877
	4	135.32295	54.39505	5.046528	0.208605	0.083852	0.254727	0.239955318
	5	135.73975	54.56259	5.556374	-0.4168	-0.16754	-0.50985	0.479435809
								deviatie max
			Valoare max		-0.4168	0.285813	0.51063	0.546786877

Testul cu calibru de pas						
Nr. Punct	x	y	z	pas. Exp	pas calibru	eroare pas increm.
1	328.00106	154.47432	55.998989			
2	309.34414	157.93985	50.931484	19.64103091	20	-0.358969087
3	289.11338	168.25298	46.860276	23.06987498	20	3.069874979
4	269.18917	181.55327	43.638992	24.17123687	20	4.171236865
5	249.85498	193.09053	40.75262	22.69912574	20	2.699125737
6	228.86466	206.04829	36.981927	24.95426391	20	4.954263913
7	209.81097	218.19696	35.670968	22.63518559	20	2.635185595
8	188.0005	228.88717	33.587068	24.37867677	20	4.378676769
9	167.34872	242.1431	32.345147	24.57149264	20	4.571492641
10	148.04309	254.46829	31.854372	22.90978692	20	2.909786917
					media	3.225630481
					eroare max.	4.954263913



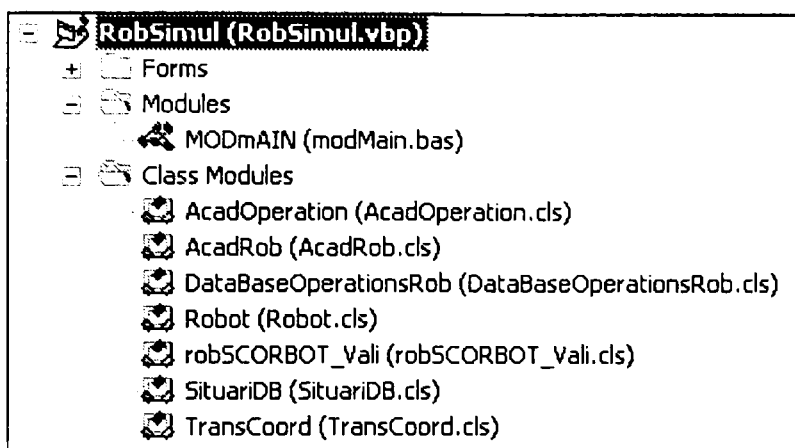
Arhitectura aplicației RobSim

Aplicația RobSim este realizată în limbajul de programare Visual Basic versiunea 6. Prezentarea arhitecturii aplicației este considerată utilă de către autor deoarece cu RobSim se desfășoară lucrări de laborator la disciplina Robotică Avansată iar autorul a pus la dispoziția celor interesați codul sursă al aplicației pentru realizarea unor particularizări sau dezvoltări ulterioare.

Arhitectura aplicației este constituită dintr-o interfață cu utilizatorul constituită din 27 forme, un modul și 8 clase . O captură de ecran a structurii arborescente a proiectului în care se pot observa formele este prezentată în figura de mai jos.



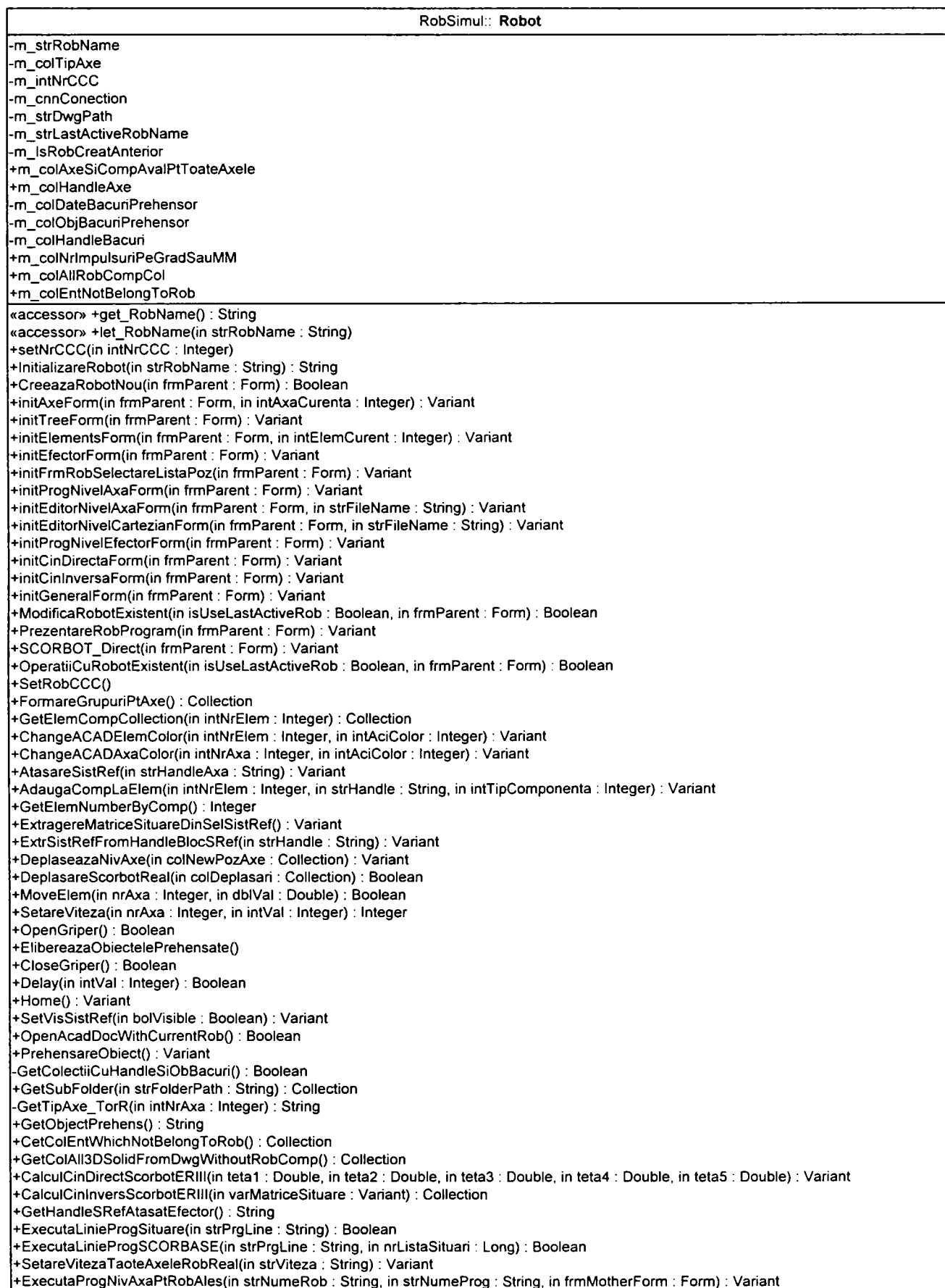
Modulul aplicației RobSim conține referințe la toate formele și clasele aplicației. În acest modul sunt stocate funcții de interes general pentru scrierea și citirea din fișierul de inițializare a aplicației, pentru conversii din șiruri de caractere în colecții, etc. În figura următoare se observă denumirile claselor aplicației.



Variabilele globale ale aplicației și funcțiile de interes general conținute în modul se observă în figura următoare :

«utility» RobSimul:: MODmAIN
<pre> +Pl_ +g_strRobActivName +g_DBOperatii +g_SituariDB +g_TransCoord +g_robSCORBOT +g_AcadOperation +g_Robot +g_AcadRob +g_frmRobot +g_frmRobGenerare +g_frmRobGeneral +g_frmRobOpExist +g_frmRobAxe +g_frmRobElemente +g_frmRobProgNivelEfector +g_frmRobProgNivelAxa +g_frmRobCinDirecta +g_frmRobCinInversa +g_frmRobEfectorFinal +g_frmRobAfisareMatriceSituare +g_frmRobEditorProgNivelAxa +g_frmRobEditorProgNivCartezian +g_frmRobHelp +g_frmRobSelectareListaPoz +g_frmRobEditareListaSituari +g_frmRobEditareSituare +g_frmRobTree +g_frmRobSchemaCinematica +g_frmRobAlegereRobPtModificare +g_frmRobAlegereProgPtEditor +g_frmRobDeplaseazaLa +g_frmRobAlegereListaSituari +g_frmRobPrezRobProgram +g_frmSCORBOT +g_frmDigInregistrareSituari +g_acadApp +g_docAcadDocument +g_cnnSituare +g_CaleDirectorAplicatieVB +g_CaleFisierIniPTVB +g_CaleFisierComandaSCORBOT +g_IsScorbolLegat </pre>
<pre> +Main() +GetPath(in strParameter : String, in intTypeVB1OrLSP2 : Integer) : String +SetParamValue(in strSection : String, in strParam : String, in strValue : String) +GetParamValue(in strSection : String, in strParam : String) : String +copyFolder(in strSource : String, in strDest : String) : Variant +ColToStrCuSep(in colEI : Collection, in chrSeparator : String) : String +StrCuSepToCol(in strSir : String, in chrSeparator : String) : Collection +ScrieLinieInFisier(in strCaleFisier : String, in strLinFis : String) : Variant +StergeFisier(in strFilePath : String) : Variant </pre>

Clasele conțin funcțiile care implementează facilitățile aplicației. În diagramele prezentate, în fiecare diagramă corespunzătoare unei clase dreptunghiul superior conține numele clasei, dreptunghiul din mijloc conține declarațiile de variabile publice iar dreptunghiul inferior conține declarațiile funcțiilor conținute în clasă în care se observa numărul și tipul parametrilor de intrare și tipul valorii returnate.



Clasa *Robot* este clasa principală a aplicației. Deși actual aplicația permite simularea unui singur robot la un moment dat, prin modul de implementare al acestei clase este posibilă dezvoltarea RobSim pentru simularea unui sistem de fabricație flexibilă în care operează mai mulți roboți simultan. Clasa TransCoord implementează transformările de coordonate.

RobSimul: TransCoord
+CalculeazaProodusMat(in colMat : Collection) : Variant +ProdMatrixNumeric(in mat1 : Variant, in mat2 : Variant) : Variant +CorecteeazaMatriceEpsilon(in transMatrix : Variant) : Variant +mat4x4Euler(in varMat4x4 : Variant) : Variant +Eulermat4x4(in varEuler : Variant) : Variant

Clasa robSCORBOT_Vali a fost implementată de drd. ing. Ciupe Valentin și conține funcții pentru comunicația aplicației cu microcontrollerul robotului SCORBOT ER III.

RobSimul: robSCORBOT_Vali
+SeteazaParanmetriiComunicatie(in intNrPort : Integer, in intVitezaComunicatie : Integer) : Variant +InitializeazaPortul() : Variant +InchidePort() : Variant +EmergencyBrake() : Variant +ExecuteCommandByRealScorbot(in command : String) : String +HomeScorbot() : Variant +stop_mot()

Clasele “AcadOperation” și “AcadRob” conțin funcții orientate în special pe interfața aplicației cu mediul de proiectare asistată AutoCAD. În acest mediu are loc construirea modelelor virtuale de roboți și simularea desfășurării sarcinilor de lucru a roboților.

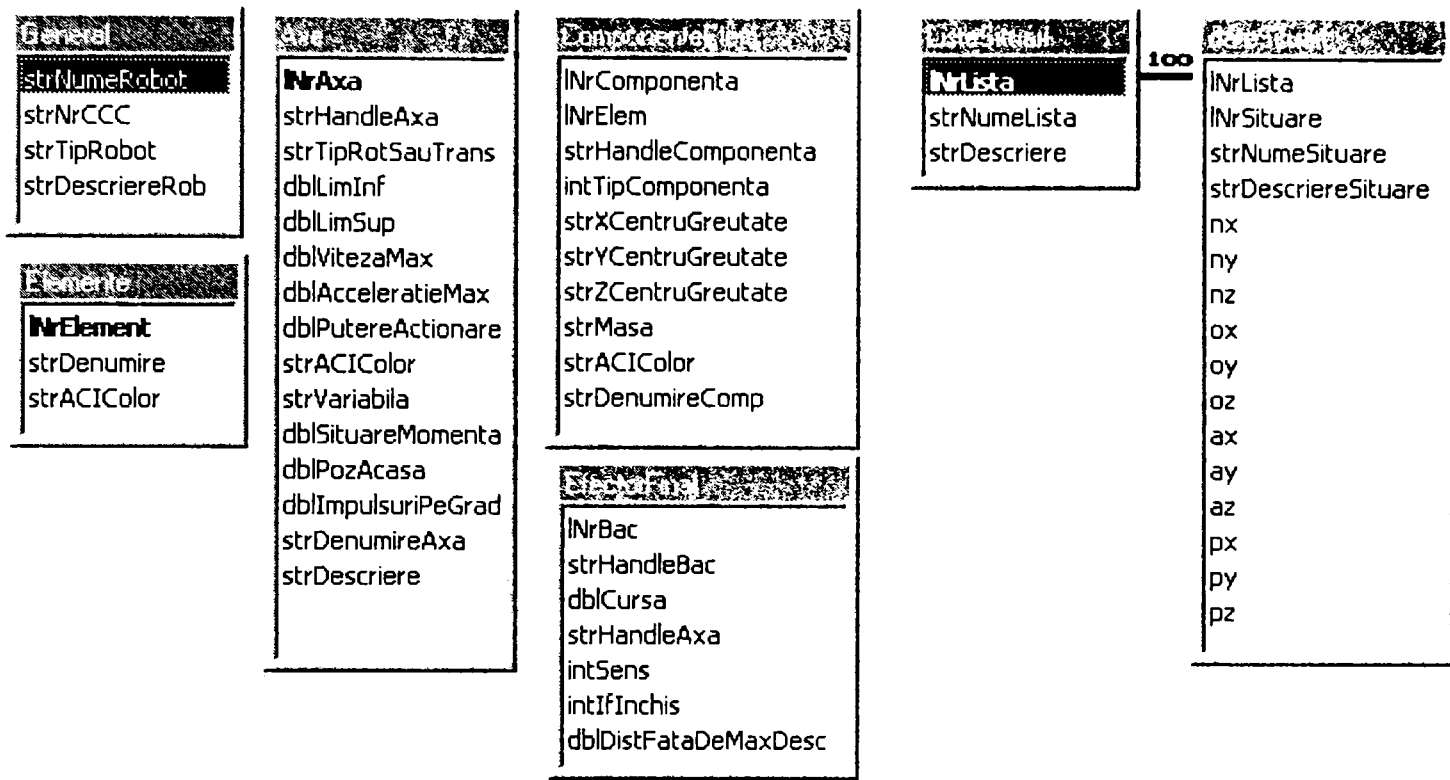
RobSimul: AcadRob
+m_objLucruDepl +m_IsDeplasareInAcad +GetIsDeplasareInAcadStatus() : Boolean +SetIsDeplasareInAcadStatus(in statusDep : Boolean) : Variant +InitObjLucruEvent(in objSolid : AcadBlockReference) : Variant +FinishObjLucruEvent() : Variant -m_objLucruDepl_Modified(in pObject : IAcadObject) +UrmaresteEfeatorulObiectulAcad() +SeselectareEntInAcad() : Collection +SeselectareAxalnAcad() : Collection +SelectareBloclnAcadPtUrmarireEfeatorulEntAcad() : String +Rotire(in intNrAxa : Integer, in dblAngGrade : Double) : Variant +GradeToRad(in dblGrade : Double) : Double +RadToGrade(in dblRad : Double) : Double +Translatie(in intNrAxa : Integer, in dblValoare : Double) : Variant +InsereazaSistRef(in varInsPt : Variant, in varPtDir : Variant) : String +AdaugaDateExtinse(in objAcad : AcadObject, in strParametru : String, in strValoare : String) : Variant +ExtrageDateExtinse(in objAcad : AcadObject, in strParametru : String) : String +GetHomegenousMatrixFromUCS3Pt(in UcsName : String, in UcsOrig : Variant, in UcsXpt : Variant, in UcsYpt : Variant) : Variant +GetSolideElemFromDrawing() : Collection +HighlighElemente(in colElem : Collection, in bolIfHighligh : Boolean) : Variant +GetHandleOffAllSolidFromDwg() : Collection

RobSimul. AcadOperation
+OpenAcad() +OpenAcadDocument(in strFilePath : String) : AcadDocument +CreeazaDreptunghi(in dblWidth : Double, in dblHeigth : Double, in dblLineWidth : Double) : AcadPolyline +CreateVports() : Collection +LoadARX_IfNotAlreadyLOaded(in arxName : String) : Boolean +GetDistanceBetween2Points(in dblX1 : Double, in dblY1 : Double, in dblZ1 : Double, in dblX2 : Double, in dblY2 : Double, in dblZ2 : Double) : Double

Clasa *DataBaseOperationsRob* conține funcțiile pentru lucrul cu bazele de date ale roboților. Fiecare robot virtual are asociată o bază de date care stochează informațiile necesare corespunzătoare pentru robot. Este important ca la închiderea aplicației, baza de date să stocheze informații exacte referitoare la situarea relativă a elementelor robotului. În urma stabilirii corespondenței dintre modelul geometric al elementelor și axelor cuplelor cinematice a robotului și șirul care le identifică în mod unic în baza de date (prin handle) nu mai este permisă ștergerea modelului geometric în AutoCAD.

RobSimul.: DataBaseOperationsRob
-m_cnnConection +InitializeDBO(in strDBPath : String) : Connection +SalvareHandleElemInDB(in intNrElem : Integer, in colComponente : Collection, in cnn : Connection) : Boolean +SalveazaSituareAxa(in cnn : Connection, in intNrAxa : Integer, in dblValMom : Double) -StergereDateAnterioareAxa(in cnn : Connection, in intNrAxa : Integer) : Variant +ElibereazaObiectePrehensate(in cnn : Connection) : Variant -AadaugaComponentePtUnElement(in intNrElem : Integer, in colComponente : Collection, in cnn : Connection) : Boolean -StergeComponenteElementAnterioare(in intNrElem : Integer, in cnn : Connection) : Variant -AadaugaElement(in intNrElem : Integer, in cnn : Connection) : Boolean +GetDataGeneraleRob(in cnn : Connection, in strNumeRob : String) : Recordset +GetElementsRobRecordset(in cnn : Connection, in strNumeRob : String) : Recordset +GetAxisRobRecordset(in cnn : Connection, in strNumeRob : String) : Recordset +GetAxeData(in cnn : Connection, in strNumeRob : String, in intNrAxa : Integer) : Recordset +GetComponentColection(in cnn : Connection, in strNumeRob : String) : Collection +GetColWithCompForAnElement(in cnn : Connection, in intNrElem : Integer) : Collection +GetAllComponentColectionInASimpleCol(in cnn : Connection) : Collection -GetComponentForAnElemRecordset(in cnn : Connection, in intElement : Integer) : Recordset -GetRecWithComponentForAllElem(in cnn : Connection) : Recordset +GetElemCompCollection(in cnn : Connection, in intNrElem : Integer) : Collection +GetAxisType_TorR_col(in cnn : Connection, in strNumeRob : String) : Collection +GetNrImpulsuriPeGradSauMM(in cnn : Connection, in strNumeRob : String) : Collection +GetAxisRobCollection(in cnn : Connection, in strNumeRob : String) : Collection +GetRobPosMomentanaAxeCol(in cnn : Connection, in strNumeRob : String) : Collection +GetRobPosAcasaAxeCol(in cnn : Connection, in strNumeRob : String) : Collection +AadaugaComponenta(in cnn : Connection, in intNrElem : Integer, in strHandleCompNoua : String, in intTipComp : Integer) +GetEINumberByComp(in cnn : Connection, in strCompHandle : String) : Integer -GetAllCompRecordset(in cnn : Connection) : Recordset +GetColWithSistRefHandle(in cnn : Connection) : Collection +GetHandleSistRefEfector(in cnn : Connection, in intNrCCC : Integer) : String +GetMecPrehensRecordset(in cnn : Connection, in intNrBac : Integer) : Recordset +GetGriperStatus(in cnn : Connection) : Integer +GetAxeRecordsetForValue(in cnn : Connection, in strValue : String) : Recordset +GetElementsRecordsetByValue(in cnn : Connection, in strComboValue : String) : Recordset +GetBACRecordsetForValue(in cnn : Connection, in strComboValue : String) : Variant +UpdateComponentalInfo(in cnn : Connection, in colInfoComponenta : Collection) : Variant

În figura următoare (stânga) este prezentată o captură de ecran în care se observă tabelele conținute în baza de date a fiecărui robot. Baza de date este realizată în Acces97. În figura următoare, în dreapta, se prezintă baza de date în care sunt stocate listele de situații. Această bază de date este specifică aplicației, aceeași listă de situații poate fi utilizată de mai mulți roboți, în cazul în care este implementat modelul geometric invers. De asemenea această bază de date conține și situațiile indicate cu ajutorul digitizorului 3D.



În ultima diagramă din anexă este prezentată clasa *Situări DB* care conține funcțiile pentru stocarea în baza de date și extragerea componentelor matricilor de situare a efectorului final al robotului.

RobSimul: SituariDB
+Initialize() : Variant +GetMatrixFromSitName(in strNumeListaSituari : String, in strNumeSituare : String) : Variant +GetMatrrixFromSitNrAndListNr(in INrListaPozitii : Long, in INrSituare : Long) : Variant +SalvareInDB(in INrLista : Long, in strNumeSituare : String, in strDescriereSituare : String, in transMatrix : Variant) : Boolean +GetNumarUltimaSituareDinLista(in INrLista : Long) : Long +OpenADORecordset(in strSql : String) : Recordset +GetDescriptionFromName(in strName : String) : String +GetNrListaFromName(in strNumeLista : String) : Long +GetNrSituareFromName(in strName : String) : Long +GetNameSituareFromNr(in INrStuare : Long) : String +GetDescriereSituareFromNr(in INrStuare : Long) : String +GetColectionWithPosLists() : Collection +GetCollWithNumeAndNr(in INrLista : Long) : Collection

Cod sursă din cadrul aplicației DigiCAD

```

*****
'Genereaza o linie in modelul AutoCAD avand ca si capete
'2 puncte a caror coordonate sunt indicate cu digitizorul 3D
*****
Private Function linie() As AcadEntity

    'Calea directorului unde sunt stocate imaginile afisate în forma
    Dim strPicturePath As String
    strPicturePath = App.Path & "\\Resurse\\ImaginiEnt\\simple\\"

    txtDescription.Text = "Indicati primul punct de capat al liniei"
    picFigModelare.Picture = LoadPicture(strPicturePath & "linie_1.jpg")

    'Indicarea coordonatelor primului punct
    Dim startP As Variant
    startP = IndicaPunctCuDig3D

    'Afiseaza coordonatele primului punct
    AfiseazaCoordPt startP, "P1:"

    txtDescription.Text = "Indicati al doilea punct de capat al liniei"
    picFigModelare.Picture = LoadPicture(strPicturePath & "linie_2.jpg")

    'Indicarea coordonatelor celui de-al doilea punct
    Dim endP As Variant
    endP = IndicaPunctCuDig3D

    'Afiseaza coordonatele celui de-al doilea punct
    AfiseazaCoordPt endP, "P1:"

    'Creeaza linia
    Dim objLine As AcadLine
    Set objLine = g_acadApp.ActiveDocument.ModelSpace.AddLine(startP, endP)

    'Afiseaza lungimea liniei
    caract "lungimea liniei :" & CStr(dist(startP, endP)) & " mm"

    txtDescription.Text = "linie 3D indicata prin punctele de capat"
    picFigModelare.Picture = LoadPicture(strPicturePath & "linie.jpg")

    'Returneaza obiectul ca entitate AutoCAD
    Dim ent As AcadEntity
    Set ent = objLine
    Set linie = ent

End Function

```

```

*****
'Stocheaza sistemul de coordonate curent si comanda ca sistemul de coordonate
'curent al modelului sa devina sistemul de coordonate absolut
*****
Private Sub CitesteUcs()

    'salveaza sistemul de coordonate curent
    Set m_UCS_old = g_acadApp.ActiveDocument.ActiveUCS

    'UCS devine WCS
    g_acadApp.ActiveDocument.SendCommand "UCS" & vbCr & "W" & vbCr

End Sub

*****
'Sistemul de coordonate curent devine sistemul de coordonate stocat de catre
'functia CitesteUCS
*****
Private Sub RefaceUcs()

    g_acadApp.ActiveDocument.ActiveUCS = m_UCS_old

End Sub

*****
'Indicarea matricii de transformare de la WCS la un sistem de coordonate UCS
'indicat prin 3 puncte
*****
Private Function situare()

    'Se citeste UCS initial
    CitesteUcs

    Dim strPicturePath As String
    strPicturePath = App.Path & "\Resurse\ImaginiEnt\diverse\"

    txtDescription.Text = "Indicati originea sistemului de referinta"
    picFigModelare.Picture = LoadPicture(strPicturePath & "situare_1.jpg")

    'Se indica primul punct cu digitiyorul 3D pentru originea UCS
    Dim P1 As Variant
    P1 = IndicaPunctCuDig3D

    'Afiseaza coordonatele primului punct
    AfiseazaCoordPt P1, "origine :"

    txtDescription.Text = "Indicati punctul pentru directia axei Ox a sistemului de referinta"
    picFigModelare.Picture = LoadPicture(strPicturePath & "situare_2.jpg")

    'Se indica punctul care impreuna cu punctul de origine indica directia

```

```
'axei Ox
Dim P2 As Variant
P2 = IndicaPunctCuDig3D

'Afiseaza coordonatele celui de-al doilea punct
AfiseazaCoordPt P2, "pt. directie Ox:"

txtDescription.Text = "Indicati un punct în planul xOy"
picFigModelare.Picture = LoadPicture(strPicturePath & "situare_3.jpg")

'Se indica cu digitiyorul 3D un punct care sa fie continut in planul xOy al UCS
Dim P3 As Variant
P3 = IndicaPunctCuDig3D

'Afiseaza coordonatele celui de-al doilea teilea punct
AfiseazaCoordPt P3, "pt. plan xOy:"

txtDescription.Text = "vector indicat prin origine si punct pt directie"
picFigModelare.Picture = LoadPicture(strPicturePath & "situare.jpg")

'Se shimba UCS pe baza a trei puncte
Dim varMatrix As Variant
varMatrix = SchimbaUcs3P(P1, P2, P3)

'Se citeste matricea UCS si se afiseaza
caract "-----"
caract "Matrice situare :"
Dim i As Integer
Dim j As Integer
For i = 0 To 3
    For j = 0 To 3
        caract "mat(" & CStr(i) & "," & CStr(j) & ")=" & CStr(Round(varMatrix(i, j), 3))
    Next j
Next i

'Se reface UCS initial
RefaceUcs
```

End Function

```
*****
'Schimba UCS prin indicarea a trei puncte si returneaza matricea noului UCS
*****
Private Function SchimbaUcs3P(P1 As Variant, P2 As Variant, P3 As Variant) As Variant
```

```
    Dim strP1 As String
    Dim strP2 As String
    Dim strP3 As String

    strP1 = CStr(P1(0)) & "," & CStr(P1(1)) & "," & CStr(P1(2))
```

```

strP2 = CStr(P2(0)) & "," & CStr(P2(1)) & "," & CStr(P2(2))
strP3 = CStr(P3(0)) & "," & CStr(P3(1)) & "," & CStr(P3(2))

'Apeleaza comanda AutoCAD
Dim strCommand As String
strCommand = "UCS" & vbCr & "3P" & vbCr & strP1 & vbCr & strP2 & vbCr & strP3 & vbCr
g_acadApp.ActiveDocument.SendCommand strCommand

Dim varUcsMatrix As Variant
On Error Resume Next
Dim dblUCSWordMatrix(0 To 3, 0 To 3) As Double

'Determina matricea de transformare de la UCS la WCS
varUcsMatrix = g_acadApp.ActiveDocument.ActiveUCS.GetUCSMatrix

If Err.Number <> 0 Then
    Err.Clear
    'Eroarea apare cind ucs este wcs
    dblUCSWordMatrix(0, 0) = 1
    dblUCSWordMatrix(0, 1) = 0
    dblUCSWordMatrix(0, 2) = 0
    dblUCSWordMatrix(0, 3) = 0
    dblUCSWordMatrix(1, 0) = 0
    dblUCSWordMatrix(1, 1) = 1
    dblUCSWordMatrix(1, 2) = 0
    dblUCSWordMatrix(1, 3) = 0
    dblUCSWordMatrix(2, 0) = 0
    dblUCSWordMatrix(2, 1) = 0
    dblUCSWordMatrix(2, 2) = 1
    dblUCSWordMatrix(2, 3) = 0
    dblUCSWordMatrix(3, 0) = 0
    dblUCSWordMatrix(3, 1) = 0
    dblUCSWordMatrix(3, 2) = 0
    dblUCSWordMatrix(3, 3) = 1
    varUcsMatrix = dblUCSWordMatrix
End If
On Error GoTo 0
SchimbaUcs3P = varUcsMatrix

End Function

'*****
' Genereaza un cilindru pe baza indicarii coordonatelor a patru puncte
'cu digitizorul 3D
'*****
Private Function cilindru4P() As AcadEntity

    CitesteUcs

```

```

Dim strPicturePath As String
strPicturePath = App.Path & "\\Resurse\ImaginiEnt\solide\"

*****
'Preluare coordonate puncte in WCS
*****
txtDescription.Text = "Indicati primul punct pe cerc"
picFigModelare.Picture = LoadPicture(strPicturePath & "cilindru4P_1.jpg")

'Primul punct
Dim P1 As Variant
P1 = IndicaPunctCuDig3D

'Afiseaza coordonatele primului punct
AfiseazaCoordPt P1, "P1: "

txtDescription.Text = "Indicati al doilea punct pe cerc"
picFigModelare.Picture = LoadPicture(strPicturePath & "cilindru4P_2.jpg")

'Al doilea punct
Dim P2 As Variant
P2 = IndicaPunctCuDig3D

'Afiseaza coordonatele celui de-al doilea punct
AfiseazaCoordPt P2, "P2: "

txtDescription.Text = "Indicati al treilea punct pe cerc"
picFigModelare.Picture = LoadPicture(strPicturePath & "cilindru4P_3.jpg")

'Al treilea punct
Dim P3 As Variant
P3 = IndicaPunctCuDig3D

'Afiseaza coordonatele celui de-al doilea teilea
AfiseazaCoordPt P3, "P3:"

'Se schimba ucs prin 3 puncte
Dim varMatrix As Variant
varMatrix = SchimbaUcs3P(P1, P2, P3)

'Transforma punctele in noul ucs
P1 = g_acadApp.ActiveDocument.Utility.TranslateCoordinates(P1, acWorld, acUCS, False)
P2 = g_acadApp.ActiveDocument.Utility.TranslateCoordinates(P2, acWorld, acUCS, False)
P3 = g_acadApp.ActiveDocument.Utility.TranslateCoordinates(P3, acWorld, acUCS, False)

'Deseneaza cercul cu 3 puncte cu SendCommand
Dim strP1 As String
Dim strP2 As String
Dim strP3 As String

'sunt necesare Puncte 2D !!!

```

```

strP1 = CStr(P1(0)) & "," & CStr(P1(1))
strP2 = CStr(P2(0)) & "," & CStr(P2(1))
strP3 = CStr(P3(0)) & "," & CStr(P3(1))

Dim strCommand As String
strCommand = "Circle" & vbCr & "3P" & vbCr & strP1 & vbCr & strP2 & vbCr & strP3 & vbCr
g_acadApp.ActiveDocument.SendCommand strCommand

'Determina entitatea cerc generata in modelul AutoCAD
Dim ent As AcadEntity
Set ent = gasesteUltimaEntitateGenerata

'Creeaza o variabila cu cercul pentru extrudare
Dim strVar As String
strVar = "(setq varCerc (entlast))" & vbCr
g_acadApp.ActiveDocument.SendCommand strVar

On Error Resume Next
'Citeste datele cercului
Dim objCerc As AcadCircle
Set objCerc = ent
Dim centru As Variant
Dim raza As Double
Dim normala As Variant
centru = objCerc.Center
raza = objCerc.Radius
normala = objCerc.Normal

txtDescription.Text = "Indicati al doilea punct pe cerc"
picFigModelare.Picture = LoadPicture(strPicturePath & "cilindru4P_4.jpg")

'Indica un al patrulea punct pentru inaltime
Dim P4Relativ As Variant
P4Relativ = IndicaPunctCuDig3D
P4Relativ = g_acadApp.ActiveDocument.Utility.TranslateCoordinates(P4Relativ, acWorld,
acUCS, False)

txtDescription.Text = "Cilindru indicat prin 4 puncte"
picFigModelare.Picture = LoadPicture(strPicturePath & "cilindru4P.jpg")

'Inaltimea este coordonata z a celui de-al patrulea punct relativ la ucs
Dim dblInaltime As Double
dblInaltime = P4Relativ(2)
caract "inaltime : " & CStr(dblInaltime)

'Realizeaza extrudarea cercului cu inaltimea data
Dim strExtrudeCommand As String
strExtrudeCommand = "Extrude" & vbCr & "!varCerc" & vbCr & vbCr & CStr(dblInaltime) &
vbCr & vbCr
g_acadApp.ActiveDocument.SendCommand strExtrudeCommand

```



```

'Determina entitatea cilindru generat in modelul AutoCAD
Dim entCilindru As AcadEntity
Dim objCilindru As Acad3DSolid
Set entCilindru = gasesteUltimaEntitateGenerata
Set objCilindru = entCilindru

Dim c1 As Variant
Dim c2 As Variant
'Extrage colturile diagonale ale paralelipedului care incadreaza
'cilindrul generat
objCilindru.GetBoundingBox c1, c2
caract "colturi Bounding Box"
AfiseazaCoordPt c1, "c1 :"
AfiseazaCoordPt c1, "c1 :"

If Err.Number <> 0 Then Err.Clear
On Error GoTo 0

RefaceUcs

'Returneaza entitatea AutoCAD ce reprezinta cilindrul
Set cilindru4P = entCilindru

End Function

'*****
'Genereaza o suprafata biliniara avind ca si colturi 4 puncte indicate
'cu digitizorul 3D
'*****
Private Function sBiliniara()

On Error Resume Next

Dim strPicturePath As String
strPicturePath = App.Path & "\Resurse\ImaginiEnt\suprafete\"

txtDescription.Text = "Indicati primul punct pentru prima linie"
picFigModelare.Picture = LoadPicture(strPicturePath & "sBiliniara_1.jpg")

'Primul punct
Dim P1 As Variant
P1 = IndicaPunctCuDig3D

'Afiseaza coordonatele primului punct
AfiseazaCoordPt P1, "P1: "

txtDescription.Text = "Indicati al doilea punct pentru prima linie"
picFigModelare.Picture = LoadPicture(strPicturePath & "sBiliniara_2.jpg")

'Al doilea punct

```

Dim P2 As Variant

P2 = IndicaPunctCuDig3D

'Afiseaza coordonatele celui de-al doilea punct

AfiseazaCoordPt P2, "P2: "

txtDescription.Text = "Indicati primul punct pt a doua linie"

picFigModelare.Picture = LoadPicture(strPicturePath & "sBiliniara_3.jpg")

'Al treilea punct

Dim P3 As Variant

P3 = IndicaPunctCuDig3D

'Afiseaza coordonatele celui de-al doilea teilea punct

AfiseazaCoordPt P3, "P3:"

txtDescription.Text = "Indicati punctul al doilea pt a doua linie"

picFigModelare.Picture = LoadPicture(strPicturePath & "sBiliniara_4.jpg")

'Al patrulea punct

Dim P4 As Variant

P4 = IndicaPunctCuDig3D

'Afiseaza coordonatele celui de-al patrulea punct

AfiseazaCoordPt P4, "P4:"

'Creeaza patru linii ca si muchii

Dim L1 As AcadLine

Dim L2 As AcadLine

Dim L3 As AcadLine

Dim L4 As AcadLine

Set L1 = g_acadApp.ActiveDocument.ModelSpace.AddLine(P1, P2)

g_acadApp.ActiveDocument.SendCommand "(setq muchia_1 (entlast))" & vbCr

Set L2 = g_acadApp.ActiveDocument.ModelSpace.AddLine(P2, P3)

g_acadApp.ActiveDocument.SendCommand "(setq muchia_2 (entlast))" & vbCr

Set L3 = g_acadApp.ActiveDocument.ModelSpace.AddLine(P3, P4)

g_acadApp.ActiveDocument.SendCommand "(setq muchia_3 (entlast))" & vbCr

Set L4 = g_acadApp.ActiveDocument.ModelSpace.AddLine(P4, P1)

g_acadApp.ActiveDocument.SendCommand "(setq muchia_4 (entlast))" & vbCr

'Setari pentru densitatea retelei

g_acadApp.ActiveDocument.SetVariable "SURFTAB1", Int(Val(txtM.Text))

g_acadApp.ActiveDocument.SetVariable "SURFTAB2", Int(Val(txtN.Text))

'Utilizeaza comanda EDGESURF pentru generarea suprafetei retea

Dim strCommand As String

strCommand = "_Edgesurf" & vbCr & "!muchia_1" & vbCr & "!muchia_2" & _
vbCr & "!muchia_3" & vbCr & "!muchia_4" & vbCr

g_acadApp.ActiveDocument.SendCommand strCommand

Dim ent3DMesh As AcadEntity

```

Dim obj3DMesh As AcadPolygonMesh
Set ent3DMesh = gasesteUltimaEntitateGenerata
Set obj3DMesh = ent3DMesh

'Sterge liniile create ca si muchii
Dim strEraseCommand As String
strEraseCommand = "ERASE" & vbCr & "!muchia_1" & vbCr & "!muchia_2" & vbCr &
"!muchia_3" & vbCr & "!muchia_4" & vbCr & vbCr
g_acadApp.ActiveDocument.SendCommand strEraseCommand

Dim coord As Variant
coord = obj3DMesh.Coordinates

'Afiseaza date caracteristice ale suprafetei
Dim nrVert As Integer
nrVert = Int((Val(txtM.Text) + 1) * (Val(txtN.Text) + 1))
caract "numar de vert. : " & CStr(nrVert)
caract "-----"
caract "Coordonatele vertexurilor :"
Dim i As Integer
For i = 0 To nrVert - 1
    caract CStr(i) & " : " & CStr(coord(i)) & " , " & CStr(coord(i + 1)) & " , " & CStr(coord(i + 2))
Next

txtDescription.Text = "Suprafata biliniara"
picFigModelare.Picture = LoadPicture(strPicturePath & "sBiliniara.jpg")

If Err.Number <> 0 Then
    Err.Clear
End If
On Error GoTo 0

Set sBiliniara = ent3DMesh

End Function

*****
'Schimba directia de vedere a modelului pe baza a doua puncte indicate cu digitizorul 3D
*****
Private Function dirVedere()

Dim strPicturePath As String
strPicturePath = App.Path & "\Resurse\ImaginiEnt\diverse\"

txtDescription.Text = "Indicati punctul directie"
picFigModelare.Picture = LoadPicture(strPicturePath & "dirVedere_1.jpg")

'Primul punct
Dim startP As Variant
startP = IndicaPunctCuDig3D

```

'Afiseaza coordonatele primului punct

AfiseazaCoordPt startP, "origine :"

txtDescription.Text = "Indicati punctul tinta"

picFigModelare.Picture = LoadPicture(strPicturePath & "dirVedere_2.jpg")

'Al doilea punct

Dim endP As Variant

endP = IndicaPunctCuDig3D

'Afiseaza coordonatele celui de-al doilea punct

AfiseazaCoordPt endP, "pt. directie :"

txtDescription.Text = "Directie de vedere indicata prin punct observator si punct tinta"

picFigModelare.Picture = LoadPicture(strPicturePath & "dirVedere.jpg")

'calculeaza componente vectorului pentru directia de vedere

Dim dirVed(0 To 2) As Double

dirVed(0) = endP(0) - startP(0)

dirVed(1) = endP(1) - startP(1)

dirVed(2) = endP(2) - startP(2)

'Schimba directia de vedere a modelui AutoCAD

Dim strCommand As String

strCommand = "Vpoint" & vbCr & CStr(dirVed(0)) & "," & CStr(dirVed(1)) & "," &

CStr(dirVed(2)) & vbCr

g_acadApp.ActiveDocument.SendCommand strCommand

End Function

Elemente de geometrie analitică utilizate în cadrul aplicației de proiectare asistată de calculator a mobilierului pornind de la informații obținute prin digitizare.

Determinarea planului bisector pornind de la două drepte concurente care trec fiecare prin 2 puncte date

Se notează cu M_0 punctul de intersecție a două drepte (D_1) și (D_2) . Direcția dreptei (D_1) este definită de punctele M_0 și A iar dreapta (D_2) de punctele M_0 și B . Se notează cu (D_p) dreapta perpendiculară pe dreptele (D_1) și (D_2) . Se consideră un punct M_1 pe dreapta (D_1) la o distanță arbitrară d de punctul M_0 în sensul M_0A și un punct M_2 la aceeași distanță d de M_0 pe dreapta d_2 . Se cunoaște că într-un triunghi isoscel, bisectoarea unghiului format de laturile de dimensiuni egale este și mediană. Deci bisectoarea unghiului format de dreptele (D_1) și (D_2) este dreapta care trece prin M_0 și mijlocul M_{12} al segmentului M_1M_2 . Se notează cu (p_1) planul definit de dreptele concurente D_1 și D_p iar cu (p_2) planul definit de D_2 și D_p . Prin plan bisector se denumește în continuare planul care împarte în două părți egale unghiul diedru format de planele (p_1) și (p_2) . Planul bisector (p_b) coincide cu planul definit de dreptele perpendiculare (D_p) și (D_b) .

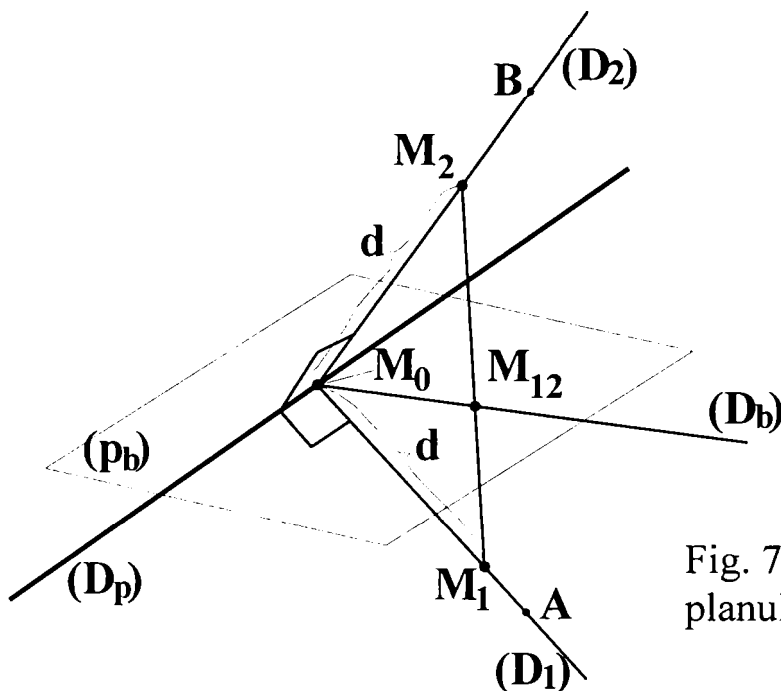


Fig. 7.4.1. Determinarea ecuației planului bisector

Ca date de intrare pentru determinarea planului bisector sunt punctele $M_0(x_0, y_0, z_0)$, $A(x_a, y_a, z_a)$ și $B(x_b, y_b, z_b)$. Se urmărește determinarea coeficienților ecuației generale a planului (p_b) . Pe baza punctelor A , M_0 și B se determină parametrii directori ai dreptelor (D_1) și (D_2) .

$$l_1 = x_a - x_0; \quad m_1 = y_a - y_0; \quad n_1 = z_a - z_0 \quad (\text{A6.1})$$

$$l_2 = x_b - x_0; \quad m_2 = y_b - y_0; \quad n_2 = z_b - z_0 \quad (\text{A6.2})$$

Se determină perpendiculara comună (D_p) a dreptelor (D_1) și (D_2) . Dreapta (D_p) este dreapta de intersecție între planul (p_1) ce trece prin (D_1) și (D_p) și planul (p_2) ce trece prin (D_2) și (D_p) deci verifică sistemul de ecuații [Mur62]:

$$\left\{ \begin{array}{l} (p_1) \equiv \begin{vmatrix} x-x_0 & y-y_0 & z-z_0 \\ l_1 & m_1 & n_1 \\ \begin{vmatrix} m_1 & n_1 \\ m_2 & n_2 \end{vmatrix} & \begin{vmatrix} n_1 & l_1 \\ n_2 & l_2 \end{vmatrix} & \begin{vmatrix} l_1 & m_1 \\ l_2 & m_2 \end{vmatrix} \end{vmatrix} = 0 \\ (p_2) \equiv \begin{vmatrix} x-x_0 & y-y_0 & z-z_0 \\ l_2 & m_2 & n_2 \\ \begin{vmatrix} m_1 & n_1 \\ m_2 & n_2 \end{vmatrix} & \begin{vmatrix} n_1 & l_1 \\ n_2 & l_2 \end{vmatrix} & \begin{vmatrix} l_1 & m_1 \\ l_2 & m_2 \end{vmatrix} \end{vmatrix} = 0 \end{array} \right. \quad (A6.3)$$

Se exprimă în sistemul A6.3 planele (p₁) și (p₂) sub forma generală :

$$\begin{cases} A_1 \cdot x + B_1 \cdot y + C_1 \cdot z + D_1 = 0 & (p_1) \\ A_2 \cdot x + B_2 \cdot y + C_2 \cdot z + D_2 = 0 & (p_2) \end{cases} \quad (A6.4)$$

Se utilizează notațiile :

$$a = \begin{vmatrix} m_1 & n_1 \\ m_2 & n_2 \end{vmatrix}; \quad b = \begin{vmatrix} n_1 & l_1 \\ n_2 & l_2 \end{vmatrix}; \quad c = \begin{vmatrix} l_1 & m_1 \\ l_2 & m_2 \end{vmatrix} \quad (A6.5)$$

Prin desfășurarea expresiilor ecuațiilor A6.3 și echivalarea cu A6.4 se determină coeficienții ecuațiilor generale a planelor (p₁) și (p₂):

$$\begin{aligned} A_1 &= m_1 \cdot c - b \cdot n_1; & B_1 &= a \cdot n_1 - l_1 \cdot c; & C_1 &= l_1 \cdot b - a \cdot m_1; \\ D_1 &= x_0(b \cdot n_1 - m_1 \cdot c) + y_0(l_1 \cdot c - a \cdot n_1) + z_0(a \cdot m_1 - l_1 \cdot b); \\ A_2 &= m_2 \cdot c - b \cdot n_2; & B_2 &= a \cdot n_2 - l_2 \cdot c; & C_2 &= l_2 \cdot b - a \cdot m_2; \\ D_2 &= x_0(b \cdot n_2 - m_2 \cdot c) + y_0(l_2 \cdot c - a \cdot n_2) + z_0(a \cdot m_2 - l_2 \cdot b) \end{aligned}$$

Parametrii directori ai dreptei (d_p) sunt :

$$l_p = \begin{vmatrix} B_1 & C_1 \\ B_2 & C_2 \end{vmatrix}; \quad m_p = \begin{vmatrix} C_1 & A_1 \\ C_2 & A_2 \end{vmatrix}; \quad n_p = \begin{vmatrix} A_1 & B_1 \\ A_2 & B_2 \end{vmatrix} \quad (A6.5)$$

Se consideră o distanță arbitrară d față de punctul M₀. Se determină coordonatele punctului M₁(x₁, y₁, z₁) situat la distanța d de M₀ pe dreapta (D₁):

$$x_1 = x_0 + d \cdot l_1; \quad y_1 = y_0 + d \cdot m_1; \quad z_1 = z_0 + d \cdot n_1 \quad (A6.6)$$

În mod analog, coordonatele punctului M₂(x₂, y₂, z₂) situat la distanța d de M₀ sunt:

$$x_2 = x_0 + d \cdot l_2; \quad y_2 = y_0 + d \cdot m_2; \quad z_2 = z_0 + d \cdot n_2 \quad (A6.7)$$

Coordonatele mijlocului M₁₂ al segmentului M₁M₂ sunt:

$$x_{12} = \frac{x_1 + x_2}{2}; \quad y_{12} = \frac{y_1 + y_2}{2}; \quad z_{12} = \frac{z_1 + z_2}{2} \quad (A6.8)$$

Ecuația planului bisector (p_b) determinat de dreapta (D_p) care trece prin punctul M₀(x₀, y₀, z₀) având parametrii directori l_p, m_p, n_p și punctul M₁₂(x₁₂, y₁₂, z₁₂) este:

$$\begin{vmatrix} x-x_0 & y-y_0 & z-z_0 \\ x_{12}-x_0 & y_{12}-y_0 & z_{12}-z_0 \\ l_p & m_p & n_p \end{vmatrix} = 0 \quad (A6.9)$$

$$\text{Se notează: } e = x_{12} - x_0; \quad f = y_{12} - y_0; \quad g = z_{12} - z_0 \quad (A6.10)$$

Coeficienții ecuației generale A_b · x + B_b · y + C_b · z + D_b = 0 a planului (p_b) sunt:

$$A_b = f \cdot n_p - m_p \cdot g$$

$$B_b = l_p \cdot g - e \cdot n_p$$

$$C_b = e \cdot m_p - l_p \cdot f$$

$$D_b = (m_p \cdot g - f \cdot n_p)x_0 + (e \cdot n_p - l_p \cdot g)y_0 + (l_p \cdot f - e \cdot m_p)z_0$$

Descrierea modului de lucru al aplicației DigiMob

Aplicația DigiMob dispune de o caseta de dialog principală (fig. 1) din care se poate activa opțiunea de generare a modelelor geometrice a pieselor de mobilier (prin acționarea butonului de comandă “Generare piese”). La activarea acestei opțiuni se deschide caseta de dialog prezentată în figura 2 în care se alege tipul de piesa corespunzător piesei reale a cărui model geometric urmează a fi generat pornind de la informații obținute prin digitizare.

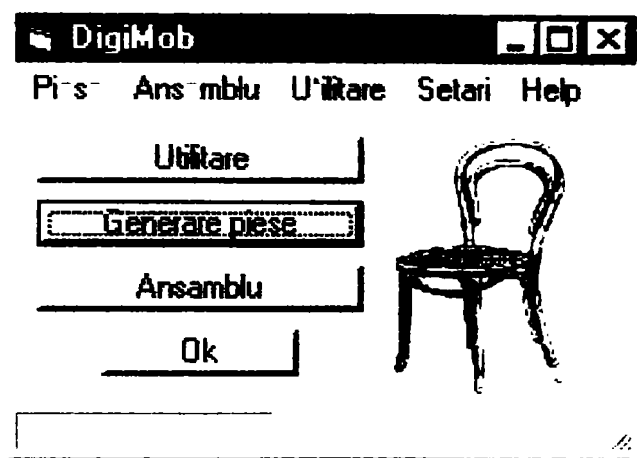


Fig. 1. Caseta de dialog principală a aplicației DigiMob

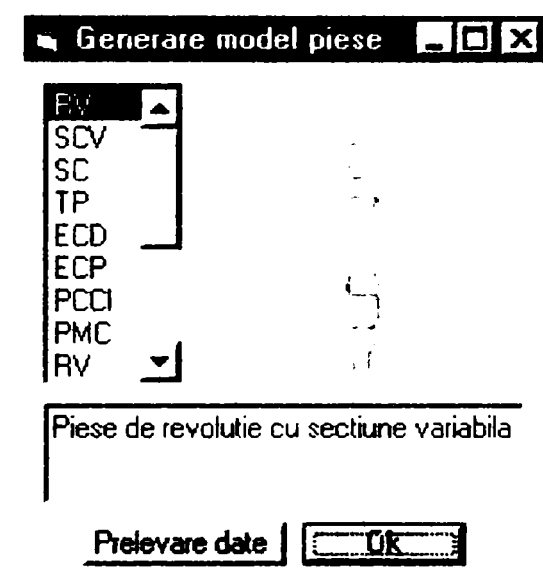


Fig. 2. Imagine a casetei de dialog pentru selectarea tipului de piesă

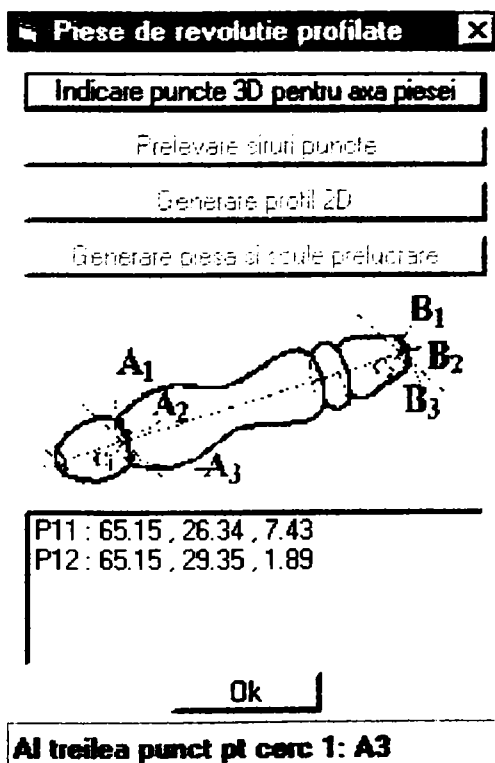


Fig.3. Imagine a casetei de dialog în timpul indicării cu ajutorul digitizorului 3D a punctelor pentru axă

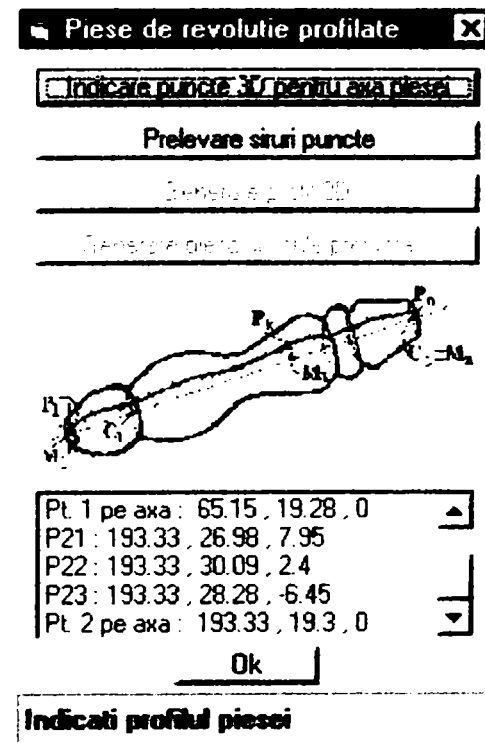


Fig. 4. Utilizatorul este solicitat să preleveze cu digitizorul 3D coordonatele unui șir de puncte de-a lungul piesei

Prin acționarea butonului de comandă “Prelevare date” se deschide caseta de dialog pentru tipul de piesă selectat. Se prezintă în continuare pașii urmați pentru generarea modelului unei piese de revoluție (RV) și fragmentele corespunzătoare din codul sursă. Caseta de dialog utilizată pentru piesele de revoluție profilate este prezentată în figura 3. În urma selectării a două grupe de câte trei puncte, aplicația determină axa piesei (funcția *IndicaPunctePtAxaPiesei* a cărei cod sursă este prezentat în această anexă). Coordonatele celor două puncte care determină axa piesei sunt afișate în caseta de dialog în lista destinată afișării informațiilor. Utilizatorul este solicitat printr-o imagine sugestivă (fig. 4) să deplaseze vârful digitizorului pe suprafața piesei de-a lungul acesteia pentru prelevarea informațiilor despre coordonatele unui șir de puncte. Punctele din șirul de puncte sunt situate pe o curbă strâmbă în spațiu. Aplicația determină polilinia 2D care reprezintă profilul piesei (funcția *DeterminaProfil2DPiesaRev*). În urma generării polilinieii 2D, utilizatorul deschide prin intermediul butonului de comandă “Generare piesă și scule prelucrare” o nouă casetă de dialog a cărei imagine este prezentată în figura 5. În această casetă de dialog sunt specificați parametrii necesari pentru realizarea desenelor de execuție ai SDV-urilor (vezi paragraful 7.7.4.).

Prin activarea butonului de comandă “LANSARE DESENARE AUTOMATĂ” se activează rutinele pentru determinarea profilului șablonului de copiere, determinarea zonelor în care sunt necesare cuțite de profilare, realizarea automată a desenelor de execuție a cuțitelor profilate, șablonului de copiere și a modelului 3D al piesei (fig. 6).

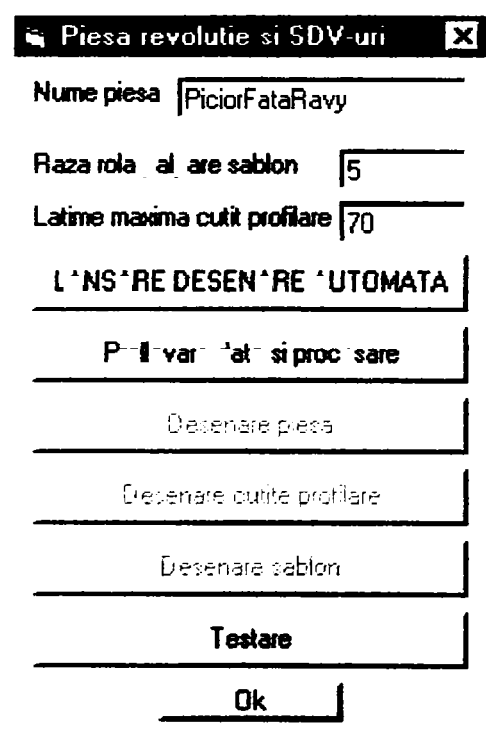


Fig. 5. Indicarea unor parametrii necesari pentru realizarea SDV-urilor.

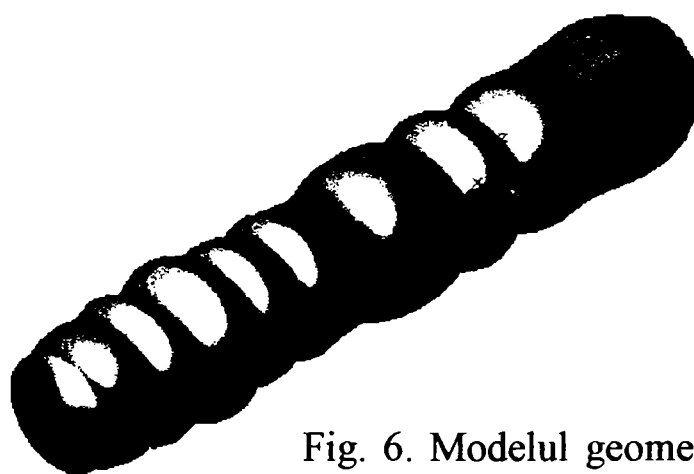


Fig. 6. Modelul geometric al piesei de revoluție în vedere umbrită

Fragmente din codul sursă al aplicației DigiMob

'Se cere utilizatorului sa indice cu ajutorul digitizorului doua grupe de cite 3 puncte
'de pe suprafata piesei. Punctele din fiecare grupa trebuie sa fie situate intr-un
'plan perpendicular pe axa piesei. Din fiecare grup se constrieste un cerc care trece
'prin cele 3 puncte ale grupului si se determina centrul cercului. Acest centru este
'unul din cele 2 puncte care indica directia axei piesei

Private Function IndicaPunctePtAxaPiesei()

Dim p1 As Variant
Dim p2 As Variant
Dim P3 As Variant

Mesaj "---- puncte pentru determinare axa ---"

Mesaj "Primul punct pt cerc 1: A1"
picRevProf.Picture = LoadPicture(App.Path & "\Resources\Img\RV\" & "P1.jpg")
p1 = g_DigiOperations.IndicaPtDig
p1 = Round2(p1)
AfiseazaCoordPt p1, "P11 :"

Mesaj "Al doilea punct pt cerc 1: A2"
picRevProf.Picture = LoadPicture(App.Path & "\Resources\Img\RV\" & "P2.jpg")
p2 = g_DigiOperations.IndicaPtDig
p2 = Round2(p2)
AfiseazaCoordPt p2, "P12 :"

Mesaj "Al treilea punct pt cerc 1: A3"
picRevProf.Picture = LoadPicture(App.Path & "\Resources\Img\RV\" & "P3.jpg")
P3 = g_DigiOperations.IndicaPtDig
P3 = Round2(P3)
AfiseazaCoordPt P3, "P13 :"

Dim varPt1Axa As Variant
varPt1Axa = DeterminaCentruCerc(p1, p2, P3)
varPt1Axa = Round2(varPt1Axa)

'Afiseaza coordonatele centrului cercului 1
AfiseazaCoordPt varPt1Axa, "Pt. 1 pe axa :"

Dim P4 As Variant
Dim P5 As Variant
Dim P6 As Variant

Mesaj "Primul punct pt cerc 2: B1"
picRevProf.Picture = LoadPicture(App.Path & "\Resources\Img\RV\" & "P4.jpg")
P4 = g_DigiOperations.IndicaPtDig
P4 = Round2(P4)
AfiseazaCoordPt P4, "P21 :"

```
Mesaj "Al doilea punct pt cerc 2: B2"
picRevProf.Picture = LoadPicture(App.Path & "\Resources\Img\RV\" & "P5.jpg")
P5 = g_DigiOperations.IndicaPtDig
P5 = Round2(P5)
AfiseazaCoordPt P5, "P22 :"
```

```
Mesaj "al treilea punct pt cerc 2: B3"
picRevProf.Picture = LoadPicture(App.Path & "\Resources\Img\RV\" & "P6.jpg")
P6 = g_DigiOperations.IndicaPtDig
P6 = Round2(P6)
AfiseazaCoordPt P6, "P23 :"
```

```
Mesaj "Indicati profilul piesei"
picRevProf.Picture = LoadPicture(App.Path & "\Resources\Img\RV\" &
"PiesaRevolutieindicareProfil.jpg")
```

```
Dim varPt2Axa As Variant
varPt2Axa = DeterminaCentruCerc(P4, P5, P6)
varPt2Axa = Round2(varPt2Axa)
```

```
'Afiseaza coordonatele centrului cercului 2
AfiseazaCoordPt varPt2Axa, "Pt. 2 pe axa : "
```

```
'Seteaza variabile private in modul care strocheaza coordonatele a doua
'puncte de pe axa
m_P1Axa = varPt1Axa
m_P2Axa = varPt2Axa
```

```
g_acadApp.ActiveDocument.SendCommand "osmode" & vbCr & "0" & vbCr
```

```
cmdPrelevareSiruriPuncte.Enabled = True
```

End Function

```
'-----
'Functia determina profilul 2D al piesei de revolutie pe baza cunoasterii a doua
'puncte pe axa si a punctelor 3D din sirul de puncte palpate cu digitizorul 3D
'pe suprafata piesei de-a lungul acesteia
'-----
```

```
Private Function DeterminaProfil2DPiesaRev() As Collection
```

```
Mesaj "Se genereaza profilul 2D al piesei "
```

```
'Stocheaza sistemul de coordonate al modelului intr-o variabila privata in clasa
CitesteUcs
```

```
'Schimba UCS astfel incit originea sa fie in primul punct si
'axa x pe directia celor 2 puncte indicate cu dig 3D
g_acadApp.ActiveDocument.SendCommand "(setq p1 (list " & CStr(m_P1Axa(0)) & _
```

```
" " & CStr(m_P1Axa(1)) & " " & CStr(m_P1Axa(2)) & ")")" & vbCr
g_acadApp.ActiveDocument.SendCommand "(setq p2 (list " & CStr(m_P2Axa(0)) & _
" " & CStr(m_P1Axa(1)) & " " & CStr(m_P2Axa(2)) & ")")" & vbCr
g_acadApp.ActiveDocument.SendCommand "UCS" & vbCr & "ZA" & vbCr & "!p1" & _
vbCr & "!p2" & vbCr
```

'Pentru fiecare punct 3D se calculeaza distanta la dreapta indicata prin cele 2 puncte

Dim i As Integer

Dim p3DWCS As Variant 'Punct 3D din sirul de puncte indicat cu digitizorul 3D

Dim p3DUCS As Variant 'Punct 3D rezultat prin transformarea punctului p3DWCS in
'coordonatele UCS curent

Dim PtPol2D(0 To 2) As Double 'Punct 2D pentru generarea modelului geometric al
'piesei de revolutie

Dim xPol2D As Double

Dim yPol2D As Double

Dim distPtAxa As Double 'Distanta de la punctul din sirul de puncte la axa

Dim p1AxaUCS As Variant 'Primul punct pe axa exprimat in sistemul de coord. local

Dim p2AxaUCS As Variant 'Al doilea punct pe axa exprimat in sistemul de coord. local

'Se transforma coordonatele celor 2 puncte care determina directia axei din
'sistemul de coordonate absolut in sistemul de coordonate local

```
p1AxaUCS = g_acadApp.ActiveDocument.Utility.TranslateCoordinates(m_P1Axa, _
acWorld, acUCS, False)
```

```
p2AxaUCS = g_acadApp.ActiveDocument.Utility.TranslateCoordinates(m_P2Axa, _
acWorld, acUCS, False)
```

Dim colPtPol2DNeordonate As New Collection 'Punctele apartinand profilului
'2D al piesei

Dim dblP3DWCS(0 To 2) As Double 'Utilizat deoarece functia de transformare
'a coordonatelor suporta doar array de double

For i = 1 To m_colPolProfil3D.Count

p3DWCS = m_colPolProfil3D(i)

dblP3DWCS(0) = p3DWCS(0)

dblP3DWCS(1) = p3DWCS(1)

dblP3DWCS(2) = p3DWCS(2)

'Transformarea coordonatelor punctelor in sistemul de coordonate curent

```
p3DUCS = g_acadApp.ActiveDocument.Utility.TranslateCoordinates(dblP3DWCS, _
acWorld, acUCS, False)
```

'Se calculeaza distanta de la punct la axa

```
distPtAxa = g_UilitiesDigiMob.GetDistPunctLaDreapta(p1AxaUCS, p2AxaUCS, p3DUCS)
```

'Se determina coordonatele pentru profilul planar al piesei

xPol2D = p3DUCS(2)

yPol2D = distPtAxa

PtPol2D(0) = xPol2D

PtPol2D(1) = yPol2D

colPtPol2DNeordonate.Add PtPol2D

Next

```
'Ordoneaza punctele in functie de coordonata indicata de al doilea parametru
Dim colPtProfilOrdonate As New Collection
Set colPtProfilOrdonate = OrdoneazaPuncte(colPtPol2DNeordonate, 0)
```

```
'Reface Ucs
RefaceUcs
```

```
Mesaj "Activati optiunea de generare a modelului piesei si al sculei"
```

```
Set DeterminaProfil2DPiesaRev = colPtProfilOrdonate
```

End Function

```
'-----
'Se lanseaza comanda de desenare automata a piesei de revotie profilate
'si a SDV-urilor pornind de la polilinia 2D ce reprezinta profilul piesei
'-----
```

```
Private Sub cmdLansareDesenareAutomata_Click()
```

```
'Verifica daca utilizatorul a specificat un nume pentru piesa
If txtNumePiesa.Text = "" Then
    MsgBox "Dati un nume piesei"
    Exit Sub
End If
```

```
Dim objPolil As Object
```

```
'Crearea unui director pentru piesa curenta
Dim strNumePiesa As String
strNumePiesa = txtNumePiesa.Text
Dim fso As New FileSystemObject
Dim pathDirPiesa As String
pathDirPiesa = App.Path & "\PieseSiScule\" & strNumePiesa
fso.CreateFolder pathDirPiesa
```

```
'Setari
Dim dblRidicareMinimaPtProfilare As Double
dblRidicareMinimaPtProfilare = 0.5
Dim dblRaza As Double
dblRaza = val(txtRazaRola.Text)
Dim dblLatimeMaxCutit As Double
dblLatimeMaxCutit = val(txtLatimeMaxCutit.Text)
m_clsPoliline.Initializare dblRidicareMinimaPtProfilare, dblLatimeMaxCutit, _
    dblRaza, pathDirPiesa, strNumePiesa
```

```
Me.Hide
```

```
'Selectare polilinie 2D (LWPOLILINE) in AutoCAD
If m_strHandlePolilInitiala <> "" Then
```

```
Dim entPolil As AcadEntity
Set entPolil = g_acadApp.ActiveDocument.HandleToObject(m_strHandlePolilInitiala)
Set objPolil = entPolil
Else
Set objPolil = m_clsPoliline.SelectPolilineOnScreen
End If
```

```
'Procesarea polilinieii. Determinarea infasuratoarei rolei si a zonelor profilate
m_clsPoliline.GenerareAutomataDeseneSculeSiPiesa objPolil
```

```
ActivareOptiuniDesenare
```

```
Me.Show
```

```
End Sub
```

```
'-----
'Generarea automata a modelului piesei de revolutie si a desenelor de executie a sculelor
'-----
```

```
Public Function GenerareAutomataDeseneSculeSiPiesa(objPoliline As Object)
```

```
'Procesarea vertexurilor polilinieii pentru determinarea infasuratoarei si profilelor
ProcesarePolilinePtExtragereInfasSiProfile objPoliline
```

```
'Desenarea cutitelor profilate
DesenareCutiteProfilate m_colProfileCutite
```

```
'Desenarea sablonului
DesenareSablon m_colVertexuriSablonCopiere
```

```
'Desenarea piesei de revolutie
DesenarePiesa m_colPolilineOriginala, m_dblDistDelaPrimPtPolilLaAxa
```

```
End Function
```

```
'-----
'Desenarea piesei de revolutie profilate
'-----
```

```
Private Function DesenarePiesa(colPolilOrig As Collection, dblDistPrimPtProfilAxa As Double)
```

```
'Deschiderea unui nou document
g_acadApp.Documents.Add
```

```
'Deseneaza axa
Dim c1 As Variant
Dim c2 As Variant
c1 = colPolilOrig(1)
```

```

c2 = colPolilOrig(colPolilOrig.Count)
Dim c1_3D(0 To 2) As Double
Dim c2_3D(0 To 2) As Double
c1_3D(0) = c1(0)
c1_3D(1) = c1(1)
c1_3D(2) = 0#
c2_3D(0) = c2(0)
c2_3D(1) = c2(1)
c2_3D(2) = 0#
Dim x1 As Double
Dim x2 As Double
x1 = c1(0)
x2 = c2(0)
Dim p1Axa(0 To 1) As Double
Dim p2Axa(0 To 1) As Double
p1Axa(0) = x1
p1Axa(1) = c1(1) - dblDistPrimPtProfilAxa
p2Axa(0) = x2
p2Axa(1) = p1Axa(1)

'Adauga primul capat ca prim vertex a poliliniei
Dim colCurbaSiCapete As New Collection
colCurbaSiCapete.Add p1Axa
Dim var As Variant
For Each var In colPolilOrig
    colCurbaSiCapete.Add var
Next
'Adauga al doilea capat ca ultim vertex al curbei
colCurbaSiCapete.Add p2Axa

'Creeaza polinia in desenul curent
Dim strHandle As String
strHandle = g_UtilitiesDigiMob.CreeazaPolilDinColectiePuncte(colCurbaSiCapete, 1)
Dim objCurba As Object
Set objCurba = g_acadApp.ActiveDocument.HandleToObject(strHandle)
g_acadApp.ActiveDocument.SendCommand "(setq curba (entlast))" & vbCr
ZoomExtents

Dim p1axa3D(0 To 2) As Double
Dim p2axa3D(0 To 2) As Double
p1axa3D(0) = p1Axa(0)
p1axa3D(1) = p1Axa(1)
p1axa3D(2) = 0#
p2axa3D(0) = p2Axa(0)
p2axa3D(1) = p2Axa(1)
p2axa3D(2) = 0#
Dim axa As AcadLine
g_acadApp.ActiveDocument.ModelSpace.AddLine p1axa3D, p2axa3D
g_acadApp.ActiveDocument.SendCommand "(setq axa (entlast))" & vbCr

'Deseneaza Obiectul de revolutie

```

```

g_acadApp.ActiveDocument.SendCommand "-color" & vbCr & "33" & vbCr
g_acadApp.ActiveDocument.SendCommand "Surftab1" & vbCr & "60" & vbCr
g_acadApp.ActiveDocument.SendCommand "Surftab2" & vbCr & "10" & vbCr
g_acadApp.ActiveDocument.SendCommand "_revsurf" & vbCr & "!curba" & vbCr & "!axa" &
vbCr & "0" & vbCr & "360" & vbCr
g_acadApp.ActiveDocument.SendCommand "Vpoint" & vbCr & "1,1,1" & vbCr
g_acadApp.ActiveDocument.SendCommand "shademode" & vbCr & "G" & vbCr
g_acadApp.ActiveDocument.SendCommand "erase" & vbCr & "!curba" & vbCr & "!axa" &
vbCr & vbCr

```

```

'Salvare documentul cu modelul geometric 3D al piesei
Dim strDocName As String
strDocName = m_strPathDirPiesa & "\" & "Piesa.dwg"
g_acadApp.ActiveDocument.SaveAs strDocName

```

End Function

```

'-----
'Desenarea cutitelor profilate
'-----

```

Private Function DesenareCutiteProfilate(colProfileCutite As Collection)

```

Dim i As Integer
Dim colVertProfilCurent As New Collection
For i = 1 To colProfileCutite.Count

```

```

'Colectia cu vertexurile polilinieii curente pentru polilinie
Set colVertProfilCurent = colProfileCutite(i)

```

```

'Deschiderea formatului A4
Dim strDocA4 As String
strDocA4 = App.Path & "\A4.dwg"
g_acadApp.Documents.Open strDocA4

```

```

'Creeaza polinia in desenul curent
Dim strHandle As String
strHandle = g_UutilitiesDigiMob.CreeazaPolilDinColectiePuncte(colVertProfilCurent, 1)

```

```

'Desenare profil
Dim strCaleLisp As String
strCaleLisp = CaleLispDinCaleVB("e:\Claudiu\cut.lsp") 'App.Path & "\cut.lsp")
Dim strForLoadLispProg As String
strForLoadLispProg = "(load " & """" & strCaleLisp & """" & ")" & vbCr
Dim strNumePiesa As String
strNumePiesa = "cutit " & m_denumirePiesa & " " & CStr(i)
g_acadApp.ActiveDocument.SendCommand "(setq DenumirePiesa " & """" & strNumePiesa
& """" & ")" & vbCr
g_acadApp.ActiveDocument.SendCommand strForLoadLispProg
g_acadApp.ActiveDocument.SendCommand "(setq handlePolil " & """" & strHandle & """" &
")" & vbCr
g_acadApp.ActiveDocument.SendCommand "cut" & vbCr

```



```
ZoomExtents  
DoEvents  
pauza 300000
```

```
g_acadApp.ActiveDocument.SendCommand "Shademode" & vbCr & "h" & vbCr
```

```
'Salvare document cu cutit
```

```
Dim strDocName As String
```

```
strDocName = m_strPathDirPiesa & "\" & "cutitStrunjire_" & CStr(i) & ".dwg"
```

```
g_acadApp.ActiveDocument.SaveAs strDocName
```

```
Next
```

```
End Function
```



Imagine a unei variante funcționale a digitizorului 3D cu fire prezentat în subcapitolul 6.3. “Proiectarea constructivă a unui digitizor cu fire”.