

# **GEOMETRIA TOPOLOGICA A CONSTRUCTIILOR MECANICE**

**Raport anual grant ID\_791  
Contract 738/2009**

**Colectiv:   ș. l. dr. ing. Virgil Gabriel TEODOR  
              prof. dr. ing. Nicolae OANCEA  
              prof. dr. ing. Alexandru EPUREANU  
              asist. dr. ing. Florin Bogdan MARIN  
              cercet. drd. ing. Ionut POPA**

**Anul 2011**

## CUPRINS

<b>CUPRINS</b> .....	2
<b>OBIECTIVUL IV. CONTROLUL GENERĂRII SUPRAFETELOR TOPOLOGICE ÎN SCOPUL COMPENSĂRII ERORILOR</b> .....	3
4.1. Controlul prin identificarea topologica utilizând algoritmi genetici .....	6
4.2. Controlul prin identificarea topologica pe baza rețelelor neuronale .....	7
4.3. Controlul prin extinderea tehnicii de identificare bazata pe circulația parametrilor la identificarea structurilor topologice.....	9
<b>OBIECTIVUL V. APLICAREA INTEGRATA A METODOLOGIILOR DE MODELARE ȘI CONTROL AL GENERĂRII STRUCTURILOR TOPOLOGICE ȘI DEZVOLTAREA UNEI TEORII UNITARE, DENUMITA GEOMETRIA TOPOLOGICA A CONSTRUCȚIILOR MECANICE</b> .....	13
5.1. Aplicarea metodologiilor de modelare a structurilor topologice .....	13
5.2. Aplicarea metodologiilor de control a generării structurilor topologice .....	14
5.3. Teoria geometriei topologice a construcțiilor mecanice.....	47
<b>ANEXE</b> .....	55
Lucrări publicate în cadrul contractului ID-791/2008.....	55
Lucrări transmise spre publicare .....	56
Codurile sursă ale programelor realizate în cadrul contractului de cercetare.....	56

## OBIECTIVUL IV. CONTROLUL GENERĂRII SUPRAFETELOR TOPOLOGICE ÎN SCOPUL COMPENSĂRII ERORILOR

În procesul de măsurare a unui reper, pe mașinile de măsurat în coordonate (CMM), una dintre problemele frecvent apărute este necesitatea de a modifica poziția reperului în timpul măsurării, astfel încât sistemul de măsurare să aibă acces la diversele suprafețe componente. Acest lucru presupune modificarea sistemului de referință al piesei stabilit inițial și necesită fie restabilirea acestui sistem de referință, fie stabilirea unui nou sistem de referință. Uneori, datorită noii poziții a reperului, este imposibil accesul la acele elemente pe baza cărora a fost creat sistemul de referință inițial. Mai mult, poate apare imposibilitatea de a măsura simultan cele două sisteme de referință necesare (cel inițial și un nou sistem de referință la care să fie raportate suprafețele ce urmează a fi măsurate).

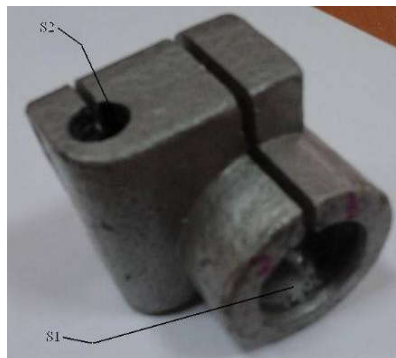
A fost elaborată o metodă care să permită corelarea punctelor de pe suprafețele măsurate, în două etape distincte, respectiv, în două poziții diferite ale aceluiași reper. Metoda este bazată pe determinarea coordonatelor a trei puncte de control în cele două poziții distincte de măsurare și, apoi, stabilirea transformării de coordonate pe baza căreia s-a ajuns la poziția modificată a reperului de măsurat.

Astfel devine posibil controlul suprafețelor generate și se creează premisele pentru compensarea erorilor prin modificarea parametrilor regimului de așchiere sau prin modificarea traiectoriilor sculelor generatoare.

Una dintre problemele frecvent apărute în cadrul procesului de măsurare este necesitatea de a modifica poziția reperului în timpul măsurării, astfel încât palpatorul sistemului de măsurare să aibă acces la diferitele elemente componente ale reperului.

Este probabil ca, datorită noii poziții a reperului, accesul palpatorului la elementele pe baza cărora a fost stabilit primul sistem de referință să fie imposibil.

Un exemplu în acest sens îl constituie cazul unui reper având ca suprafețe funcționale două alezaje cu axe disjuncte (suprafețele  $S_1$  și  $S_2$ ), vezi figura 1. În exemplul, prezentat suprafețele exterioare ale reperului nu au rol funcțional și, deci, nu necesită o prelucrare precisă din punct de vedere dimensional și al calității suprafeței. Evident că stabilirea unui sistem de referință pe baza suprafețelor exterioare ale reperului ar fi extrem de imprecisă și nu s-ar justifica din punct de vedere metrologic.



**Fig. 1.** *Reper cu două suprafețe funcționale*

O soluție a acestei probleme o constituie inspectarea suprafețelor în poziții convenabile de măsurare pentru fiecare dintre elementele respective și, apoi, corelarea poziției elementelor referitor la un unic sistem de referință.

Se consideră un punct  $P$  de pe una dintre suprafețele reperului, având coordonatele  $(x,y,z)$  în sistemul de referință al mașinii. Poziția  $P'$  a aceluiași punct va fi caracterizată de coordonatele  $(X,Y,Z)$  în sistemul de referință al piesei. Deși  $P$  și  $P'$  reprezintă fizic același punct de pe suprafața măsurată, raportarea la sisteme de referință diferite conduce la obținerea de seturi de coordonate distincte.

Înregistrarea constituie procesul de determinare a transformării de coordonate care leagă cele două sisteme de referință. Ecuația acestei transformări de coordonate este

$$P = P' \cdot R + T, \quad (1)$$

în care matricea  $R$  este matricea de rotație iar  $T$  este vectorul de translație.

Problema constă în faptul că  $R$  și  $T$  sunt necunoscute. Algoritmul de înregistrare permite determinarea elementelor  $R$  și  $T$  pe baza suprapunerii pozițiilor a trei puncte.

Dezvoltarea ecuației (1) conduce la o ecuație matriceală de tipul:

$$x = \omega_3(\varphi) \cdot \omega_2(\psi) \cdot \omega_1(\chi) \cdot X + x_0, \quad (2)$$

sau, dezvoltat:

$$\begin{pmatrix} x \\ y \\ z \end{pmatrix} = \begin{pmatrix} \cos \varphi & \sin \varphi & 0 \\ -\sin \varphi & \cos \varphi & 0 \\ 0 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} \cos \psi & 0 & -\sin \psi \\ 0 & 1 & 0 \\ \sin \psi & 0 & \cos \psi \end{pmatrix} \cdot \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos \chi & \sin \chi \\ 0 & -\sin \chi & \cos \chi \end{pmatrix} \cdot \begin{pmatrix} X \\ Y \\ Z \end{pmatrix} + \begin{pmatrix} x_0 \\ y_0 \\ z_0 \end{pmatrix}, \quad (3)$$

unde:  $\varphi$  este unghiul de rotație în jurul axei  $z$ ;

$\psi$  — unghiul de rotație în jurul axei  $y$ ;

$\chi$  — unghiul de rotație în jurul axei  $x$ ;

$x_0, y_0, z_0$  — coordonatele originii sistemului de referință  $X, Y, Z$  în sistemul de referință  $x, y, z$ , al mașinii.

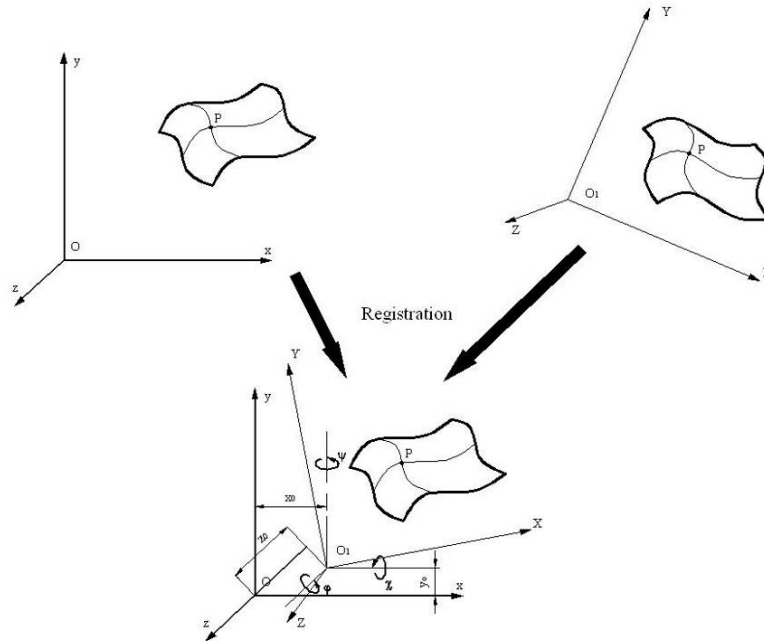
Această transformare de coordonate este prezentată în figura 2.

Ecuația transformării inverse (de la sistemul  $x, y, z$  la sistemul  $X, Y, Z$ ) este dată de:

$$X = \omega_1^T(\chi) \cdot \omega_2^T(\psi) \cdot \omega_3^T(\varphi) \cdot (x - x_0), \quad (4)$$

sau, dezvoltat:

$$\begin{pmatrix} Y \\ Y \\ Z \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos \chi & -\sin \chi \\ 0 & \sin \chi & \cos \chi \end{pmatrix} \cdot \begin{pmatrix} \cos \psi & 0 & \sin \psi \\ 0 & 1 & 0 \\ -\sin \psi & 0 & \cos \psi \end{pmatrix} \cdot \begin{pmatrix} \cos \varphi & -\sin \varphi & 0 \\ \sin \varphi & \cos \varphi & 0 \\ 0 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} x \\ y \\ z \end{pmatrix} - \begin{pmatrix} x_0 \\ y_0 \\ z_0 \end{pmatrix}. \quad (5)$$

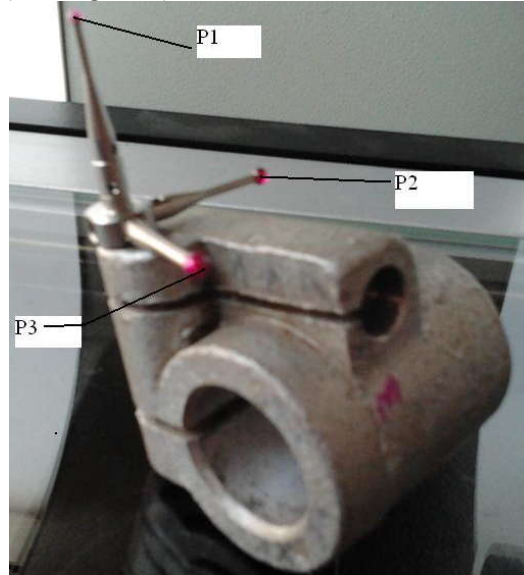


**Fig. 2.** Transformarea de coordonate

În practică, după măsurarea suprafețelor care pot fi inspectate cu palpatorul într-o anumită poziție a reperului, se vor măsura trei puncte de control poziționate astfel încât să poată fi atinse în ambele poziții de măsurare ale reperului.

Rezolvând ecuația transformării de coordonate (1) pentru coordonatele celor 3 puncte de control, obținute în două poziții distincte ale reperului, se poate determina transformarea de coordonate care realizează legătura între sistemele de referință asociate celor două măsurări.

În situația în care cele două poziții de măsurare nu permit alegerea a trei puncte care să poată fi măsurate în ambele poziții se poate utiliza un dispozitiv auxiliar care să materializeze punctele de control și care să fie fixat pe piesă printr-o metodă oarecare, într-o poziție accesibilă pentru ambele măsurători (vezi figura 3).



**Fig. 3.** Materializarea punctelor de control  $P_1, P_2, P_3$ , cu ajutorul unui dispozitiv auxiliar

### Rezolvarea numerică a problemei

Se consideră șase puncte în spațiu  $A, B, C$  respective  $A', B', C'$  astfel încât triunghiurile corespunzătoare sunt congruente:  $\Delta ABC \equiv \Delta A'B'C'$ .

Se pune problema găsirii unei serii de transformări geometrice astfel încât, aplicată asupra punctelor  $A', B', C'$ , punctele obținute prin transformare să se suprapună punctelor  $A, B, C$ .

Problema poate fi rezolvată în multiple moduri aplicând transformări asupra seturilor de puncte astfel încât vectorii determinați de punctele  $A, B, C$  ( $\overline{AB}, \overline{BC}, \overline{CA}, \overline{AB} \times \overline{BC}$ ), respectiv  $A', B', C'$  ( $\overline{A'B'}, \overline{B'C'}, \overline{C'A'}, \overline{A'B'} \times \overline{B'C'}$ ) să se suprapună.

Formal considerăm următoarele operații:

- Translația punctelor  $A', B', C'$  astfel încât  $A' = O$ ;

$$T1 = \begin{bmatrix} 1 & 0 & 0 & -X_{A'} \\ 0 & 1 & 0 & -Y_{A'} \\ 0 & 0 & 1 & -Z_{A'} \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (6)$$

- Fie  $\vec{N}_1 = \overline{A'B'} \times \overline{AB}$ , se aplică rotația punctelor  $A', B', C'$  cu unghiul  $\alpha = \arccos(\overline{A'B'} \cdot \overline{AB})$  în jurul originii și a axei  $\vec{N}_1$ , obținându-se punctele intermediare  $A'', B''$  și  $C''$ ;

$$T2 = \begin{bmatrix} (1 - \cos \alpha)N_{1x}N_{1x} + \cos \alpha & (1 - \cos \alpha)N_{1y}N_{1z} - N_{1z} \sin \alpha & (1 - \cos \alpha)N_{1z}N_{1x} + N_{1y} \sin \alpha & 0 \\ (1 - \cos \alpha)N_{1x}N_{1y} + N_{1z} \sin \alpha & (1 - \cos \alpha)N_{1y}N_{1y} + \cos \alpha & (1 - \cos \alpha)N_{1z}N_{1y} - N_{1x} \sin \alpha & 0 \\ (1 - \cos \alpha)N_{1x}N_{1z} - N_{1y} \sin \alpha & (1 - \cos \alpha)N_{1y}N_{1z} + N_{1x} \sin \alpha & (1 - \cos \alpha)N_{1z}N_{1z} + \cos \alpha & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Fie  $\vec{N}_2 = \overrightarrow{A''C''} \times \overrightarrow{AC}$ , se aplică rotația punctelor  $A''$ ,  $B''$ ,  $C''$  cu unghiul  $\beta = \arccos(\overrightarrow{A''C''} \cdot \overrightarrow{AC})$  în jurul originii și a axei  $\vec{N}_2$ , obținându-se  $A'''$ ,  $B'''$  și  $C'''$ ;

$$T3 = \begin{bmatrix} (1 - \cos \beta)N_{2x}N_{2x} + \cos \beta & (1 - \cos \beta)N_{2y}N_{2z} - N_{2z} \sin \beta & (1 - \cos \beta)N_{2z}N_{1x} + N_{2y} \sin \beta & 0 \\ (1 - \cos \beta)N_{2x}N_{2y} + N_{2z} \sin \beta & (1 - \cos \beta)N_{2y}N_{2y} + \cos \beta & (1 - \cos \beta)N_{2z}N_{2y} - N_{2x} \sin \beta & 0 \\ (1 - \cos \beta)N_{2x}N_{2z} - N_{2y} \sin \beta & (1 - \cos \beta)N_{2y}N_{2z} + N_{2x} \sin \beta & (1 - \cos \beta)N_{2z}N_{2z} + \cos \beta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

- Ultimul pas, se aplică translația punctelor  $A'''$ ,  $B'''$ ,  $C'''$  astfel încât punctul  $A'''$  se suprapune punctului  $A$ .

$$T4 = \begin{bmatrix} 1 & 0 & 0 & X_A - X_{A'''} \\ 0 & 1 & 0 & Y_A - Y_{A'''} \\ 0 & 0 & 1 & Z_A - Z_{A'''} \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (7)$$

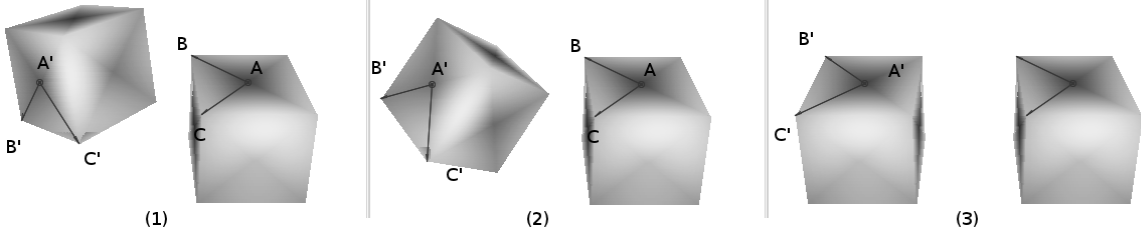


Fig. 4. Diferite faze ale transformării (translațiile au fost omise în scop ilustrativ)

#### 4.1. Controlul prin identificarea topologica utilizând algoritmi genetici

În situații reale, când punctele amintite mai sus sunt obținute în urma unor măsurători, triunghiurile  $\Delta ABC$  și  $\Delta A'B'C'$  sunt quasi-congruente și metoda amintită mai sus nu va conduce la o suprapunere exactă a celor șase puncte.

Se propune o abordare bazată pe algoritmi genetici. Algoritmii genetici sunt tehnici de optimizare ce mimează procesul de evoluție al organismelor vii. Implementând concepte precum: populație, moștenire, cromozomi, mutații, încrucișări, selecție – un set de soluții parțiale evoluează în mai multe etape în soluții cât mai aproape de optim.

Formalizat, fie  $P_j$  o populație compusă din indivizi:  $\{I_1^j, I_2^j, \dots, I_n^j\}$

Fiecare individ  $I_k^j$  codifică o soluție parțială a problemei considerate. Un individ este considerat ca fiind format dintr-o mulțime ordonată de elemente, numite sugestiv cromozomi:  $I_k^j = (C_k^j[1], \dots, C_k^j[m])$

Principalele operații ce se aplică asupra indivizilor sunt:

– **Încrucișarea**: considerând doi indivizi  $I = (C[1], \dots, C[n])$  și  $J = (D[1], \dots, D[n])$ , se alege arbitrar  $p, 1 < p < n$  – punctul de tăiere; rezultatul încrucișării este reprezentat de un urmaș având următoarea structură:

$$G = (C[1], \dots, C[p], D[p+1], \dots, D[n])$$

– **Mutația**: considerând un individ  $I = (C[1], \dots, C[n])$ , se alege arbitrar  $p, 1 \leq p \leq n$  – punctul de mutație; cromozomul  $C[p]$  va fi alterat în mod arbitrar și înlocuit cu un cromozom  $D$  – rezultatul mutației fiind un urmaș având următoarea structură:

$$G = (C[1], \dots, C[p-1], D, C[p+1], \dots, C[n])$$

–**Selecția**: consider o populație  $P = \{I_1, \dots, I_n\}$  și o funcție  $fitness : P \rightarrow \mathbb{R}$ , care evaluează calitatea indivizilor din componența populației. Se sortează indivizii după valoarea funcției  $fitness$  și se determină suma cumulată  $S = \sum_{j=1}^n fitness(I_j)$ . Apoi se calculează funcția  $fitness$  cumulativ normalizată pentru fiecare individ:

$$\overline{fitness}(I_i) = \sum_{j=1}^i fitness\left(\frac{I_j}{S}\right) \quad (8)$$

Alegând un număr aleatoriu  $R$ ,  $0 \leq R \leq 1$ , rezultatul selecției este acel individ al populației a cărui funcție  $fitness$  cumulativ normalizată este mai mare decât  $R$ .

Adaptând paradigma algoritmilor genetic la problema enunțată mai sus, reprezentarea unui individ ar putea fi considerată următoarea:

- $\partial x_1, \partial y_1, \partial z_1$  - parametrii translației T1
- $\alpha$  - unghiul rotației T2
- $\beta$  - unghiul rotației T3
- $\partial x_2, \partial y_2, \partial z_2$  - parametrii translației finale T4

Mutația se implementează prin alterarea parametrilor de mai sus cu o valoare aleatorie din intervalul  $(-\varepsilon, \varepsilon)$ , unde  $\varepsilon$  este apriori ales.

Funcția  $fitness$  se definește: 
$$\frac{1}{[dist(A, A'')]^2 + [dist(B, B'')]^2 + [dist(C, C'')]^2}$$

Acest mod de rezolvare a problemei permite găsirea transformării de coordonate, dar limitările legate de aplicarea acestui algoritm sunt cele caracteristice algoritmilor genetici.

După cum se știe, aceștia sunt algoritmi euristici, deci soluția furnizată de ei nu este întotdeauna optimă, dar se află într-o vecinătate a soluției optime.

Timpul necesar rulării acestor algoritmi este dependent de nivelul cerut al erorii. În cazul unor cerințe de precizie uzuale din punct de vedere tehnic acest timp de rulare poate fi relativ ridicat.

## 4.2. Controlul prin identificarea topologica pe baza rețelelor neuronale

În prezent, tendința în domeniul identificării structurilor mecanice se orientează spre abordarea tehnologică. În conformitate cu această abordare, structura mecanică este privită ca un întreg și este identificată prin metode specifice dintre care amintim: metoda circulației parametrilor, metoda algoritmilor genetici, metoda rețelelor neuronale.

Pentru ca aceste metode să poată fi aplicate cu succes este necesar ca elementele din care este compusă rețeaua neuronală să poată fi recunoscute și clasificate în mod automat.

Metoda de recunoaștere a formelor prezentată se bazează pe utilizarea rețelelor neuronale.

Rețeaua neuronală a fost antrenată să recunoască forme frecvent întâlnite în tehnică și anume: arce de elipsă; segmente de dreaptă și arce de evolventă.

Curbele de tip arc de cerc au fost asimilate cu cele de tip arc de elipsă având în vedere că reprezintă un caz particular al elipsei.

Pentru antrenarea rețelei neuronale a fost generată o matrice de antrenare formată din cele trei tipuri de curbe prezentate.

Deoarece existența unor coloane de valoare constantă în matricea de antrenare a rețelei duce la rezultate ambigue, pentru antrenare au fost folosite doar valorile coordonatei  $Y$  a punctelor generate.

În cadrul acestei activități a fost utilizată o variantă îmbunătățită a unei rețele neuronale create și antrenate anterior.

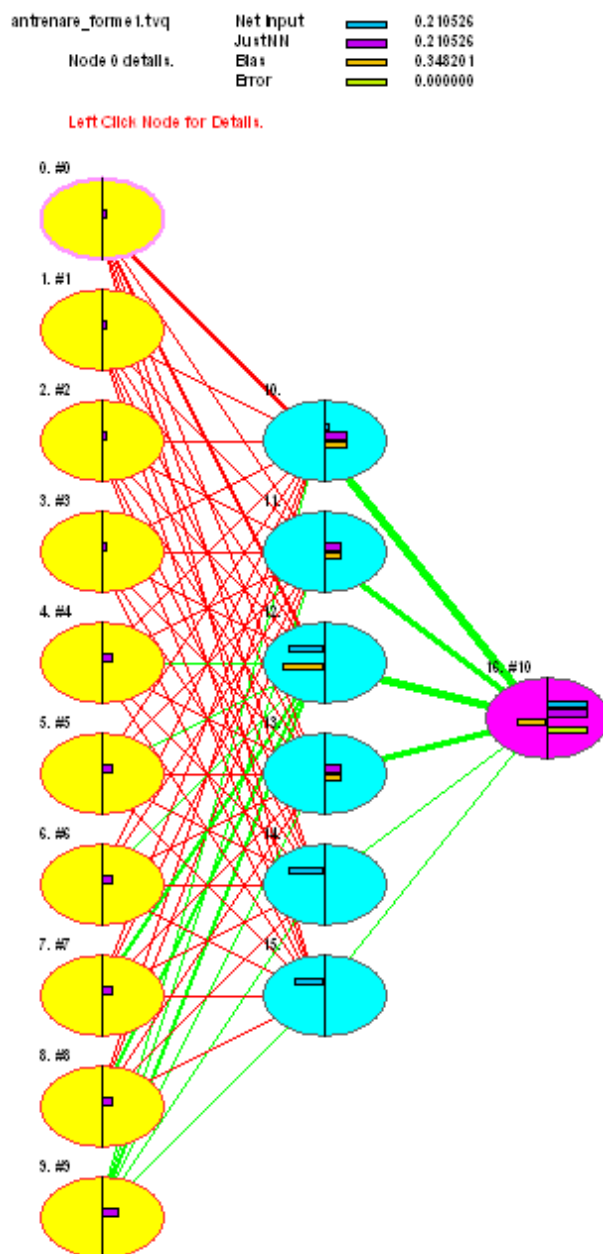
Modelul acestei rețele este prezentat în figura 5.

Echipamentul de măsurare 3D Sinowon achiziționat recent a permis verificarea practică a preciziei de recunoaștere a formelor geometrice prin măsurarea coordonatelor punctelor aparținând acestor forme și interogarea rețelei respective.

S-au măsurat 20 puncte uniforme distribuite în lungul axei  $Ox$ . Pentru a putea face în mod corect interogarea rețelei aceste puncte au fost deplasate astfel încât primul punct să aibă coordonatele  $[0;0]$ .

Pentru interogarea rețelei, ca și pentru antrenare, s-au folosit doar coordonatele  $Y$  ale acestor puncte.

Rezultatele interogării rețelei cu valorile măsurate pentru cele trei tipuri de forme sunt prezentate în figura 6.



**Fig. 5.** Modelul rețelei neuronale antrenate pentru recunoașterea formelor



	#0	#1	#2	#3	#4	#5	#6	#7	#8	#9	#10
Query	-0.0050	-0.0070	-0.0200	-0.0340	-0.0490	-0.0820	-0.1090	-0.1430	-0.1780	-0.2280	----cerc
Query	-0.0140	-0.0150	-0.0230	-0.0170	-0.0340	-0.0430	-0.0360	-0.0340	-0.0480	-0.0560	----dreapta
Query	-0.0280	-0.0760	-0.1510	-0.2290	-0.3000	-0.3920	-0.5210	-0.6490	-0.7920	-0.9420	----evolventa
#0	-0.0170	-0.0350	-0.0520	-0.0700	-0.0870	-0.1050	-0.1220	-0.1400	-0.1570	-0.1750	dreapta
#1	-0.0090	-0.0170	-0.0260	-0.0350	-0.0440	-0.0520	-0.0610	-0.0700	-0.0790	-0.0870	dreapta
#2	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	dreapta
#3	0.0090	0.0170	0.0260	0.0350	0.0440	0.0520	0.0610	0.0700	0.0790	0.0870	dreapta
#4	0.0170	0.0350	0.0520	0.0700	0.0870	0.1050	0.1220	0.1400	0.1570	0.1750	dreapta
#5	-0.0100	-0.0400	-0.0920	-0.1670	-0.2680	-0.4000	-0.5720	-0.8000	-1.1280	-2.0000	cerc
#6	-0.0080	-0.0320	-0.0730	-0.1310	-0.2090	-0.3070	-0.4290	-0.5790	-0.7650	-1.0000	cerc
#7	-0.0070	-0.0270	-0.0610	-0.1090	-0.1720	-0.2500	-0.3470	-0.4620	-0.6000	-0.7640	cerc
#8	-0.0060	-0.0230	-0.0520	-0.0930	-0.1460	-0.2120	-0.2920	-0.3870	-0.4980	-0.6280	cerc
#9	-0.0050	-0.0200	-0.0450	-0.0810	-0.1270	-0.1840	-0.2530	-0.3340	-0.4280	-0.5360	cerc
#10	-0.0040	-0.0180	-0.0400	-0.0720	-0.1130	-0.1630	-0.2230	-0.2940	-0.3760	-0.4690	cerc
#11	-0.0400	-0.1130	-0.2120	-0.3350	-0.4810	-0.6520	-0.8510	-1.0800	-1.3450	-1.6540	evolventa
#12	-0.0370	-0.1030	-0.1910	-0.3000	-0.4290	-0.5770	-0.7460	-0.9380	-1.1530	-1.3960	evolventa
#13	-0.0350	-0.0950	-0.1760	-0.2750	-0.3910	-0.5240	-0.6730	-0.8410	-1.0270	-1.2340	evolventa
#14	-0.0340	-0.0890	-0.1640	-0.2550	-0.3620	-0.4830	-0.6190	-0.7700	-0.9360	-1.1190	evolventa
#15	-0.0320	-0.0800	-0.1460	-0.2260	-0.3190	-0.4240	-0.5410	-0.6690	-0.8100	-0.9620	evolventa

Fig. 6. Rezultatul interogării rețelei

Rezultatele obținute pentru formele măsurate cu ajutorul sistemului Sinowon 3D confirmă faptul că această metodă poate fi utilizată în practică.

### 4.3. Controlul prin extinderea tehnicii de identificare bazată pe circulația parametrilor la identificarea structurilor topologice

Metoda circulației parametrilor constă în căutarea exhaustivă a valorilor parametrilor modelului matematic, într-un spațiu restrâns, în jurul unui set de valori stabilit inițial în mod arbitrar. Acest set de valori determină similitudinea optimă între model și elementul real.

Pentru cazul analizat modelul matematic îl reprezintă transformarea de coordonate, iar parametrii acestui model sunt  $\varphi$ ,  $\psi$ ,  $\chi$ , unghiurile de rotație în jurul axelor sistemului de referință al mașinii și respectiv  $x_0$ ,  $y_0$ ,  $z_0$ , distanțele între originile celor două sisteme de referință, măsurate în lungul axelor sistemului de referință al mașinii.

Se calculează derivatele parțiale ale funcției obiectiv în raport cu fiecare dintre parametrii modelului. Modificând câte puțin valorile fiecărui parametru se obține un set de valori pentru care fiecare dintre derivatele respective are o valoare minimă. În momentul în care acest set de valori corespunde unei valori acceptabile a funcției obiectiv se poate considera că procesul de optimizare este încheiat și respectivul set de valori reprezintă parametrii transformării de coordonate (parametrii modelului).

Deși precizia furnizată de această metodă este suficient de ridicată iar algoritmul este puternic convergent metoda are dezavantajul că precizia depinde în mare măsură de setul inițial de valori.

#### Aplicații numerice

##### a. Identificarea unor puncte obținute prin modelare

În scopul de a verifica algoritmul propus a fost simulată măsurarea a trei puncte având coordonatele:  $P_1 [0;0;0]$ ,  $P_2 [100;0;0]$ ,  $P_3 [0;100;0]$ .

Acestor puncte le-a fost aplicată transformarea de coordonate cu parametrii:  $\varphi=0.1$  rad,  $\psi=0.2$  rad,  $\chi=0.3$  rad,  $x_0=10$  mm,  $y_0=20$  mm și  $z_0=30$  mm.

În acest mod s-au obținut coordonatele punctelor transformate, având valorile:  $p_1 [10;20;30]$ ,  $p_2 [103.63;-7.5096;51.835]$ ,  $p_3 [38.963;115.64;26.304]$ .

Pe baza coordonatelor punctelor inițiale și a punctelor finale au fost aplicate două dintre metodele de determinare prezentate anterior și anume metoda algoritmilor genetici și metoda circulației parametrilor.

În ambele cazuri s-a considerat ca valoare admisibilă a funcției obiectiv  $\varepsilon=0.01$  mm.

În cazul metodei algoritmilor genetici s-a folosit programul *Genetic Algorithm Tool* din cadrul programului *Matlab*, versiunea 7, având opțiunile: Population Size (mărimea

populației)=20; Elite Count (elita)=2; Crossover Fraction (fracția de încrucișare)=0.8; Migration Interval (intervalul de migrare)=20; Migration Fraction (fracția de migrare)=0.2.

Rularea programului s-a făcut cu un număr de 200000 de generații, iar valoarea funcției obiectiv obținută la sfârșitul rulării a fost  $\varepsilon=0.009$  mm. Timpul de rulare a fost de aproximativ 75 minute.

În cazul metodei circulației parametrilor a fost utilizat un program de concepție proprie realizat de asemenea în programul *Matlab*.

Rularea programului a presupus parcurgerea a 23 de cicluri de modificare a parametrilor modelului, valoarea funcției obiectiv la sfârșitul rulării fiind de  $\varepsilon=0.0087$  mm. Timpul de rulare a fost de aproximativ 5 minute.

În tabelul 1 sunt prezentate erorile relative obținute în cele două cazuri menționate anterior.

**Tabelul 1.** Erorile relative în cazul metodei algoritmilor genetici și al metodei circulației parametrilor

Parametrul	Valoare teoretică	Valoare obținută prin AG	Valoare obținută prin CP	Eroare relativă AG [%]	Eroare relativă CP [%]
$\varphi$	0.1 rad	0.0997 rad	0.1 rad	0.3	0
$\psi$	0.2 rad	0.2006 rad	0.1999 rad	0.3	0.05
$\chi$	0.3 rad	0.2996 rad	0.3 rad	0.133333	0
$x_0$	10 mm	10.0184 mm	9.9989 mm	0.184	0.011
$y_0$	20 mm	19.982 mm	20.0022 mm	0.09	0.011
$z_0$	30 mm	29.9747 mm	30.0035 mm	0.084333	0.011667

Este evident că în ambele cazuri eroarea relativă a parametrilor obținuți este suficient de mică pentru a putea fi utilizată în practică.

#### **b. Identificarea unor puncte obținute prin măsurare**

**Tabelul 2.** Rezultatele măsurătorilor de control

Nr crt	Xc	Yc	R
1	60.086	72.424	30.189
2	60.192	72.806	32.044
3	60.188	72.975	32.070
4	60.312	72.346	31.683
5	60.169	72.696	32.063
6	60.184	72.983	32.061
7	60.142	73.068	32.057
8	60.227	72.971	31.851
9	60.231	72.749	32.242
10	60.351	72.963	32.004
11	60.156	72.850	32.072
12	60.218	72.928	32.410
13	60.135	73.001	31.876
14	60.236	72.937	32.460
15	60.270	73.070	31.673
16	60.246	72.979	32.110
17	60.192	72.961	32.125
18	60.129	72.939	31.775
19	60.150	72.941	31.872
20	60.197	73.021	32.069
<b>Valoare medie</b>	<b>60.201</b>	<b>72.880</b>	<b>31.935</b>
<b>Deviație standard</b>	0.062139	0.189785	0.44731

În scopul de a verifica aplicabilitatea practică a algoritmului propus a fost realizată măsurarea, în două poziții distincte, a unui alezaj cilindric.

Măsurarea a fost realizată pe mașina de măsurat în coordonate MicroHite 3D.

În prima poziție de măsurare s-au executat un număr de 20 de măsurări ale aceluiași cilindru, urmărindu-se obținerea parametrilor  $x_C$ ,  $y_C$  și  $R$ , reprezentând coordonatele punctului de intersecție a axei cilindrului cu planul  $xOy$  și respectiv raza centrului director al cilindrului. Aceste măsurători au scopul de a furniza datele de control necesare pentru a verifica rezultatele obținute în urma aplicării metodei. În aceeași poziție au fost determinate coordonatele a 3 puncte de control, materializate cu un dispozitiv similar cu cel prezentat în figura 3.

Rezultatele măsurătorilor sunt prezentate în tabelul 2.

Coordonatele punctelor de control în cele două poziții de măsurare sunt prezentate în tabelul 4.

**Tabelul 3.** *Coordonatele punctelor de control în poziția inițială și în poziția rotită*

Puncte inițiale			Puncte rotite		
$X_i$	$Y_i$	$Z_i$	$X_f$	$Y_f$	$Z_f$
57.257	-15.749	58.130	31.369	8.895	57.902
-26.682	10.004	40.416	-56.746	12.384	40.578
139.532	118.687	50.311	76.439	160.151	52.020

După această etapă, poziția piesei a fost modificată prin rotații în jurul axelor de coordonate și translații în lungul aceluiași axe. Pe suprafața cilindrică a alezajului au fost determinate coordonatele a 10 puncte și au fost determinate din nou coordonatele punctelor de control.

Aplicând algoritmul de identificare propus la capitolul IV s-au obținut valorile parametrilor de transformare de coordonate prezentați în tabelul 4.

**Tabelul 4.** *Parametrii transformării de coordonate*

Parametrul	Valoare teoretică
$\varphi$	-0.0135 rad
$\psi$	-0.0036 rad
$\chi$	-0.2593 rad
$x_0$	-28.315 mm
$y_0$	10.223 mm
$z_0$	0.029 mm

Pe baza acestor valori a fost realizată transformarea de coordonate inversă (4). Aplicând această transformare de coordonate celor 10 puncte de pe suprafața alezajului, măsurate în poziția modificată, se obțin coordonatele acestor puncte în poziția inițială.

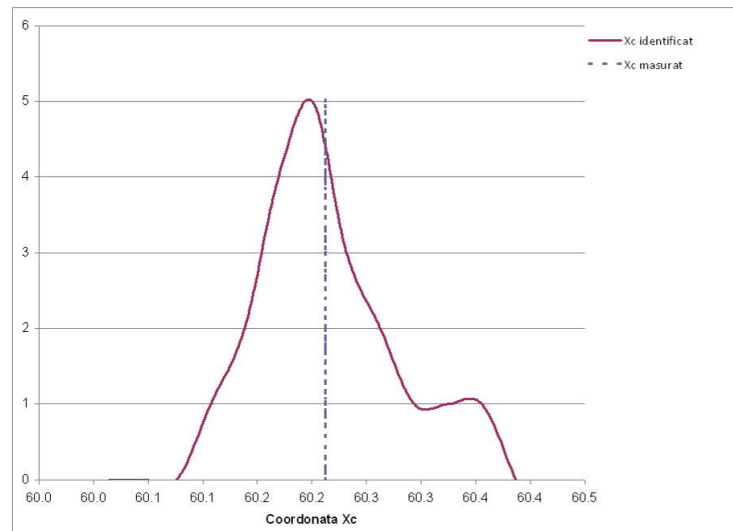
Aplicând metoda de identificare bazată pe circulația parametrilor, dezvoltată anterior și prezentată în Raportul pe anul 2009, Capitolul II, paragraful 2.1.3, au fost determinați parametrii structurii topologice de tip cilindru.

În sistemul inițial de referință, punctele determinate pe suprafața cilindrică au coordonatele prezentate în tabelul 5. În același tabel sunt indicate valorile parametrilor structurii topologice cărora le aparțin aceste puncte (respectiv valorile coordonatele punctului de intersecție a axei cilindrului cu planul  $xOy$  și raza centrului director al cilindrului).

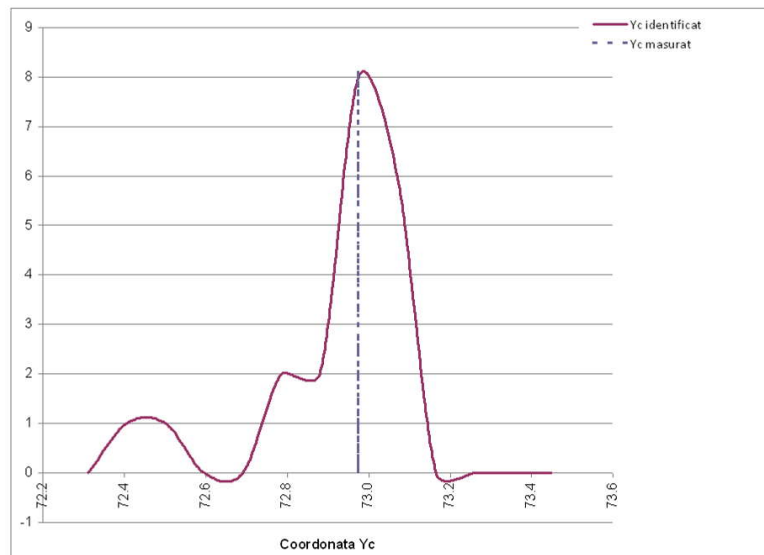
În figurile 7, 8 și 9 sunt prezentate deviațiile standard ale valorilor obținute prin cele 20 de măsurători (vezi și tabelul 2) precum și valorile obținute în urma identificării cilindrului, prin algoritmul propus (vezi tabelul 5).

**Tabelul 5.** Rezultatele măsurătorilor de control

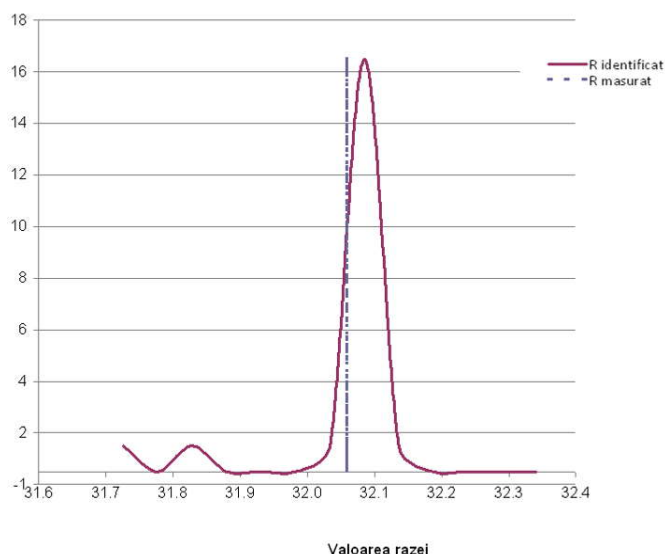
Punctul numărul	X	Y	Z	$X_c$	$Y_c$	R
1	28.023	73.084	-2.4021	60.2128	72.9735	32.0595
2	33.125	90.421	-4.5744			
3	48.584	102.67	-1.835			
4	91.579	67.226	-1.313			
5	31.863	58.995	-3.3523			
6	29.11	81.658	-4.6701			
7	41.494	98.73	-4.8984			
8	78.912	47.774	-5.2579			
9	55.064	41.486	-5.1814			
10	31.345	59.983	-5.4483			



**Fig. 7.** Coordonata  $X_c$  a centrului obținută prin măsurare și prin identificare



**Fig. 8.** Coordonata  $Y_c$  a centrului obținută prin măsurare și prin identificare



**Fig. 9.** Valoarea  $R$  a razei cercului director obținută prin măsurare și prin identificare

## **OBIECTIVUL V. APLICAREA INTEGRATA A METODOLOGIILOR DE MODELARE ȘI CONTROL AL GENERĂRII STRUCTURILOR TOPOLOGICE ȘI DEZVOLTAREA UNEI TEORII UNITARE, DENUMITA GEOMETRIA TOPOLOGICA A CONSTRUCȚIILOR MECANICE**

Problematika erorilor suprafețelor generate este complexă.

Erorile suprafețelor, mai ales ale suprafețelor complexe, sunt determinate de: erori teoretice (erori fundamentale ale procesului de generare); erori de așchiere (erori datorate fenomenelor termo-mecanice care însoțesc inerent procesele de așchiere); erori datorate formelor muchiilor de așchiere a sculelor așchietoare; erori datorate istoriei tehnologice a semifabricatelor pe care sunt generate suprafețele, uneori ansambluri de suprafețe complexe.

Ca urmare, este dificil de stabilit influența fiecărui grup de factori asupra erorii finale a piesei. Aceasta este cu atât mai complicat de realizat cu cât, în procesul generării, sunt incluse ansambluri de mișcări relative între scula profilată și semifabricat, așa cum este cazul suprafețelor reciproc înfășurătoare.

În acest fel, pornind de la reprezentarea topologică a suprafețelor, poate fi dezvoltată o teorie unitară, utilizând principiile analitice ale înfășurării suprafețelor sau dezvoltându-se o soluție grafică, în mediul de proiectare grafică CATIA. Geometria topologică de reprezentare a suprafețelor este generală și permite abordarea unor multiple probleme legate de realizarea suprafețelor, ca urmare a facilităților oferite de mediul de proiectare grafică, în care sunt dezvoltate aplicațiile. Sunt abordate probleme de generare privind profilarea suprafețelor generatoare, prin înfășurare, a suprafețelor asociate unor axoide în rulare, a suprafețelor elicoidale cilindrice și de pas constant și, de asemenea, a suprafețelor reciproc înfășurătoare cu contact punctiform, conform teoremei a II-a Olivier. Metodologia grafică, ca aplicație a geometriei topologice a suprafețelor, a fost verificată, privind acuratețea rezultatelor obținute, prin metode analitice dezvoltate numeric.

### **5.1. Aplicarea metodologiilor de modelare a structurilor topologice**

Ca o primă aplicație, se prezintă abordarea problematicii profilării sculei de tip cuțit roată, comparativ, printr-o metodă analitică și metoda grafică, a unei structuri ordonate de suprafețe reprezentată topologic.

Soluția grafică a problemei, în mediul de proiectare grafică CATIA, este bazată pe facilitățile oferite de *Part\_Environment*, care permite crearea elementelor unui mecanism virtual, asamblate în mediul *Ansembly*, ale cărui cuple cinematice vor fi definite în *DMU Kinematics*.

Mișcarea mecanismului virtual s-a realizat în baza comenzilor *Simulation și Replay*, pentru diferite poziții intermediare, *Shots*.

Utilizând comanda *Trace*, se poate determina traiectoria oricărui punct al mecanismului virtual, în raport față de oricare element al mecanismului și, de asemenea, față de un element fix considerat, *Base*, determinându-se, astfel, linia de angrenare.

Traietoriile determinate în CATIA sunt curbe de tip *Spline*, care pot fi explicitate sub formă de fișiere txt.

Sunt preconizate trei tipuri de profiluri ale piesei de generat și, în legătură cu acestea, s-au imaginat mecanisme virtuale de generare specifice (M.G.M.C.), vezi tabelul 6.

**Tabelul 6** .Rezultatele măsurătorilor de control (acțiuni prezentate în Sinteza 791/2011)

Tip M.G.M.C.	Structuri topologice	Tipul de structură topologică generată	
		Cremalieră	Cuțit Roată
M.G.M.C. pentru segment de dreaptă	Profiluri formate din segmente rectilinii	Cremalieră	Cuțit Roată
M.G.M.C. pentru arc de cerc	Profiluri formate din arce de cerc, tangente sau nu în punctele de contact	Cremalieră	Cuțit Roată
	Profiluri formate din curbe necunoscute, date prin puncte sau cunoscute prin ecuații (evolvente ale cercului, cicloide, etc.)	Cremalieră	Cuțit Roată

## 5.2. Aplicarea metodologiilor de control a generării structurilor topologice

S-au realizat aplicații de control al generării structurilor topologice complexe.

O aplicație imediată a acestor metodologii se referă la profilarea sculelor care generează prin înfășurare ansambluri de suprafețe complexe, reprezentate topologic.

### Verificarea metodei grafice în mediul grafic CATIA pentru suprafețe de revoluție-scula disc

Se prezintă, în cele ce urmează un exemplu de analiză comparativă, vis-a-vis de o metodă analitică, în scopul testării calității metodologiei grafice. Se analizează, prin exemple numerice profilarea sculei disc pentru generarea suprafețelor elicoidale cilindrice de pas constant.

Astfel de suprafețe, așa cum este cunoscut, sunt utilizate frecvent în construcția angrenajelor, a sculelor așchietoare, pompe elicoidale, compresoare de gaze. Pentru generarea unor astfel de suprafețe sunt utilizate diferite tipuri de scule așchietoare, între acestea, o mare frecvență de utilizare, o au sculele disc, care sunt scule mărginite de o suprafață periferică primară de revoluție.

Aceasta suprafață trebuie să fie reciproc înfășurătoare cu suprafața elicoidală de generat, în procesul de prelucrare.

Se urmărește a ilustra calitatea metodologiei dezvoltate, utilizând facilitățile mediului de proiectare grafică CATIA, în raport cu o metodă analitică cunoscută, metoda descompunerii mișcării elicoidale, Nicolaev.

### Profilarea sculei disc – metoda analitică

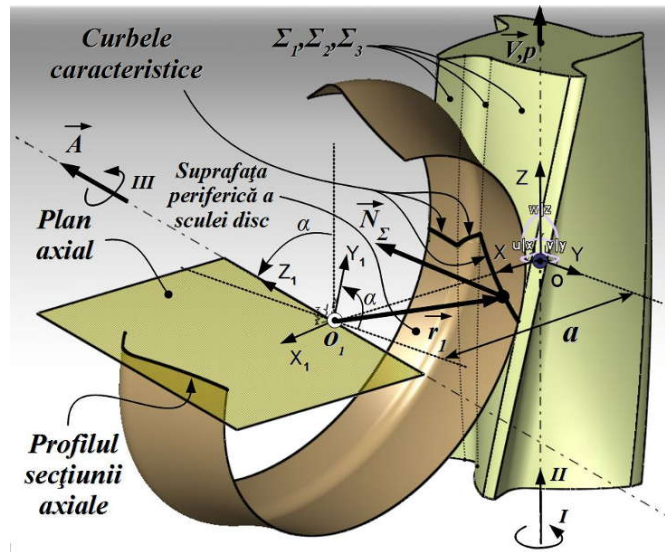
Procesul de generare a suprafeței elicoidale cu o suprafață de revoluție presupune o cinematică specifică, care include mișcările:

*I* este rotația semifabricatului în jurul axei viitoare suprafețe elicoidale;

*II* – translația semifabricatului în lungul axei suprafeței elicoidale de generat;

*III* – mișcarea de rotație a suprafeței de revoluție (suprafața periferică a sculei disc), constituind mișcarea de așchiere.

Ansamblul mișcărilor  $I$  și  $II$  generează mișcarea elicoidală de definiție a suprafeței de generat – suprafața elicoidală cilindrică de pas constant,  $\vec{V}, p$ .



**Fig. 10.** Suprafețe reciproc înfășurătoare, suprafața periferică de revoluție și suprafața elicoidală; mișcări de generare

Se definesc sistemele de referință:

$XYZ$  având axa  $Z$  suprapusă axei  $\vec{V}$ , axa suprafeței elicoidale;

$X_1Y_1Z_1$  – sistem atașat sculei disc, cu axa  $\vec{V}$  suprapusă axei  $Z_1$ .

Teorema Nikolaev pentru determinarea caracteristicii comune a suprafeței de generat,  $\Sigma$  și a suprafeței periferice primare a sculei disc,  $S$ , se exprimă în forma (vezi și figura 10)

$$(\vec{A}, \vec{N}_\Sigma, \vec{r}_1) = 0, \quad (9)$$

în care:  $\vec{A}$  este vectorul asociat axei sculei disc;

$\vec{N}_\Sigma$  – normala la suprafața de generat, în sistemul  $XYZ$ ;

$\vec{r}_1$  – vectorul punctului curent de pe  $\Sigma$ , în sistemul  $X_1Y_1Z_1$ .

Relația (9) semnifică o condiție de tangență între suprafețele  $\Sigma$  și  $S$ , ca fiind locul geometric al punctelor aparținând suprafeței elicoidale, pentru care normala la aceasta intersectează axa  $\vec{A}$  ( $Z_1$ ), altfel spus, caracteristica suprafeței  $\Sigma$ , în mișcarea de rotație în jurul axei  $\vec{A}$ , este proiecția acestei axe pe suprafața elicoidală de generat.

Se acceptă că, ecuațiile parametrice ale suprafeței  $\Sigma$  sunt de forma:

$$\Sigma \begin{cases} X = X(u, v); \\ Y = Y(u, v); \\ Z = Z(u, v), \end{cases} \quad (10)$$

cu  $u$  și  $v$  parametri variabili independenți.

Axa  $\vec{A}$  este definită de vectorul

$$\vec{A} = \cos \alpha \cdot \vec{j} - \sin \alpha \cdot \vec{k} \quad (11)$$

și, de asemenea, vectorul  $\vec{r}_1$ , în forma (12)

$$\vec{r}_1 = [X(u, v) - a] \cdot \vec{i} + Y(u, v) \cdot \vec{j} + Z(u, v) \cdot \vec{k} \quad (12)$$

Se definește normala la suprafața elicoidală în forma

$$\vec{N}_\Sigma = \begin{pmatrix} i & j & k \\ \dot{X}_u & \dot{Y}_u & \dot{Z}_u \\ \dot{X}_v & \dot{Y}_v & \dot{Z}_v \end{pmatrix}. \quad (13)$$

Condiția pentru determinarea curbei caracteristice este de tipul

$$q(u, v) = 0. \quad (14)$$

Ansamblul ecuațiilor (9) și (14) reprezintă curba caracteristică la contactul suprafeței  $\Sigma$  cu suprafața de revoluție a sculei disc:

$$C_\Sigma \begin{cases} X = X(v); \\ Y = Y(v); \\ Z = Z(v). \end{cases} \quad (15)$$

Curba  $C_\Sigma$  poate fi determinată utilizând și facilitățile CATIA, tratând suprafața de generat într-o formă topologică oferită de facilitățile mediului grafic, existând posibilitatea ca prin comanda *Projection* să se determine proiecția axei  $\bar{A}$  pe suprafața elicoidală  $\Sigma$ , în acest fel, coordonatele punctelor de pe caracteristică putând fi exprimate printr-o matrice, echivalentă formei (16),

$$C_\Sigma = \begin{pmatrix} X_1 & Y_1 & Z_1 \\ X_2 & Y_2 & Z_2 \\ \dots & \dots & \dots \\ X_n & Y_n & Z_n \end{pmatrix}. \quad (16)$$

Dacă numărul de puncte prin care se definește curba caracteristică este suficient de mare, precizia cu care este determinată aceasta poate fi satisfăcătoare cerințelor tehnice necesare proiectării sculelor de generare a acestor tipuri de suprafețe.

Folosind transformarea de coordonate

$$\begin{pmatrix} X_1 \\ Y_1 \\ Z_1 \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos \alpha & -\sin \alpha \\ 0 & \sin \alpha & \cos \alpha \end{pmatrix} \cdot \left[ \begin{pmatrix} X \\ Y \\ Z \end{pmatrix} - \begin{pmatrix} a \\ 0 \\ 0 \end{pmatrix} \right]. \quad (17)$$

curba caracteristică este transferată în sistemul  $X_1Y_1Z_1$ , principial, în forma:

$$(C_\Sigma)_{X_1Y_1Z_1} = \begin{pmatrix} X_{1_1} & Y_{1_1} & Z_{1_1} \\ X_{1_2} & Y_{1_2} & Z_{1_2} \\ \dots & \dots & \dots \\ X_{1_i} & Y_{1_i} & Z_{1_i} \\ X_{1_n} & Y_{1_n} & Z_{1_n} \end{pmatrix}. \quad (18)$$

În final, se determină secțiunea axială a sculei disc, din (18), ca fiind:

$$S_A \begin{cases} R = \sqrt{X_{1_i}^2 + Y_{1_i}^2}; \\ H = Z_{1_i}; \end{cases} \quad (19)$$

cu  $i=1 \dots n$ .

### **Profilarea sculei disc – metoda reprezentării topologice a suprafeței în mediul de proiectare grafică**

Metoda 3D de profilare a sculei disc, este dezvoltată în mediul – *Generative Shape Design*. Suprafața de generat (suprafața elicoidală) este modelată 3D, așa cum se vede în figura 11 (cazul profilului transversal circular) sau în figura 12 (cazul profilului transversal rectiliniu).



Se definesc sistemele de referință,  $XYZ$ , asociat suprafeței de generat, și  $X_1Y_1Z_1$ , asociat sculei disc (vezi figurile 11 și 12).

Prin comanda “Projection”, se proiectează axa sculei disc pe suprafața elicoidală, determinându-se curba caracteristică.

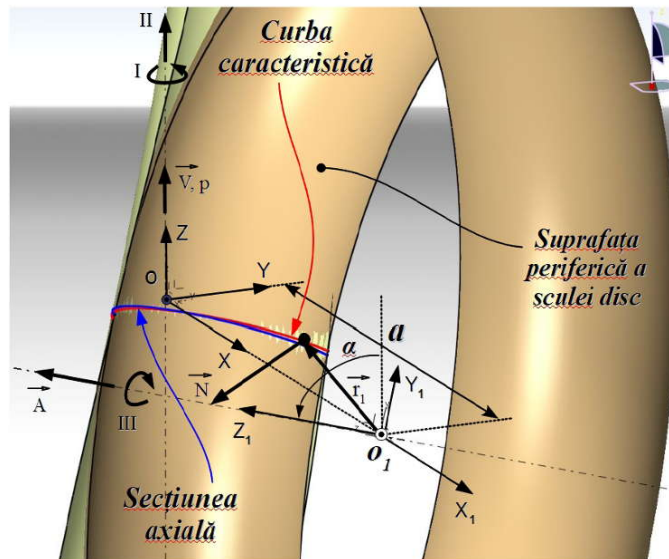


Fig. 11. Modelul 3D al suprafeței elicoidale (profil transversal circular)

Prin comanda “Revolve”, se rotește curba caracteristică în jurul axei sculei disc ( $Z_1$ ), obținându-se reprezentarea topologică virtuală (modelul virtual) a suprafeței sculei disc.

Secțiunea axială a sculei disc se obține prin intersecția acesteia cu un plan ce conține axa  $Z_1$ , prin intermediul comenzii “Intersection”.

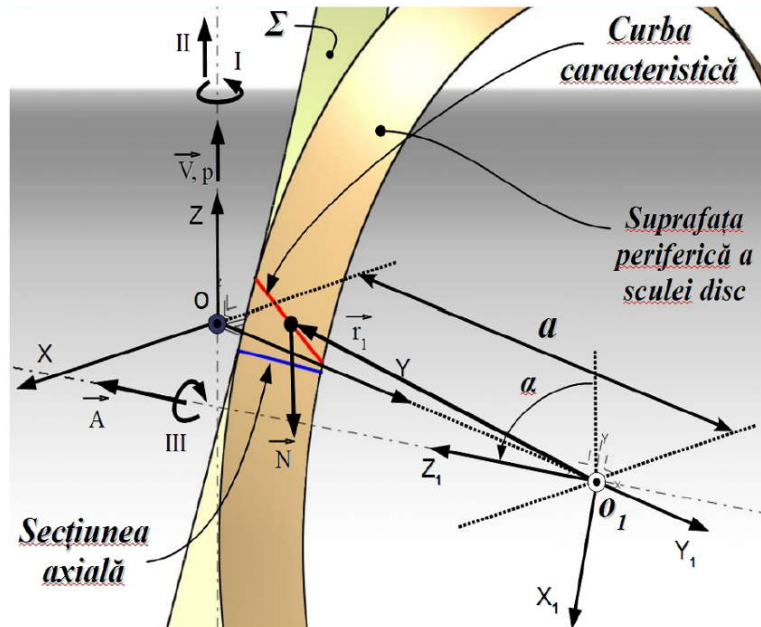


Fig. 12. Modelul 3D al suprafeței (profil transversal rectiliniu)

### Aplicație numerică

Pentru două tipuri de suprafețe elicoidale cilindrice și de pas constant, cu generatoare circulară și rectilinie, se urmărește a se determina secțiunea axială a sculei disc, prin cele două metode: analitică și metoda topologică generală de reprezentare grafică în mediul de proiectare CATIA a suprafețelor.

### Suprafața elicoidală cilindrică de pas constant, cu profil transversal circular

Forma generatoarei transversale este prezentată în figura 13. Arcul de cerc are centrul în  $O_c [X_{O_c}, Y_{O_c}]$  și trece prin punctele  $A [X_A, Y_A]$  și  $B [X_B, Y_B]$ , de ecuații:

$$\begin{cases} X = X_{O_c} - r \cdot \cos\theta; \\ Y = Y_{O_c} + r \cdot \sin\theta. \end{cases} \quad (20)$$

în care  $\theta$  este parametrul variabil,

$$r = \sqrt{((X_A - X_{O_c})^2 + (Y_A - Y_{O_c})^2)} = \sqrt{((X_B - X_{O_c})^2 + (Y_B - Y_{O_c})^2)} \quad (21)$$

Suprafața elicoidală de axa  $Z$  ( $\vec{V}$ ) și parametru elicoidal  $p$  are ecuațiile :

$$\begin{cases} X = [X_{O_c} - r \cdot \cos\theta] \cdot \cos\varphi - [Y_{O_c} + r \cdot \sin\theta] \cdot \sin\varphi; \\ Y = [X_{O_c} - r \cdot \cos\theta] \cdot \sin\varphi - [Y_{O_c} + r \cdot \sin\theta] \cdot \cos\varphi; \\ Z = p \cdot \varphi. \end{cases} \quad (22)$$

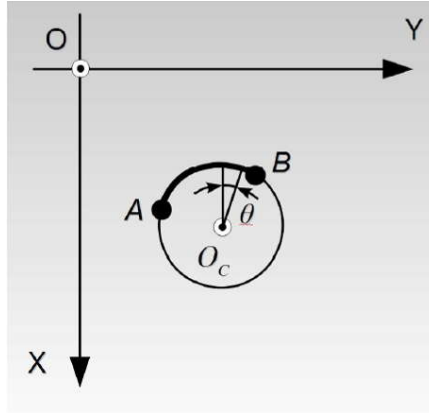


Fig. 13. Generatoarea circulară transversală

Normala la suprafața  $\Sigma$ , în sistemul XYZ, are parametrii directori:

$$\begin{cases} N_x = -p \cdot \cos(\theta - \varphi); \\ N_y = p \cdot \sin(\theta - \varphi); \\ N_z = -\cos(\theta - \varphi) \cdot \{ [X_{O_c} - r \cdot \cos\theta] \cdot \sin\varphi + [Y_{O_c} + r \cdot \sin\theta] \cdot \cos\varphi \} \\ \quad - \sin(\theta - \varphi) \cdot \{ [X_{O_c} - r \cdot \cos\theta] \cdot \cos\varphi + [Y_{O_c} + r \cdot \sin\theta] \cdot \sin\varphi \}. \end{cases} \quad (23)$$

Parametrul unghiular  $\alpha$  se definește ca fiind:

$$\alpha = \arctan\left(\frac{p}{R_e}\right) \quad (24)$$

în care,  $R_e$  este raza exterioră a suprafeței de generat.

Vectorul  $\vec{r}_1$ , din (21), devine

$$\begin{aligned} \vec{r}_1 = & \{ [X_{O_c} - r \cdot \cos\theta] \cdot \cos\varphi - [Y_{O_c} + r \cdot \sin\theta] \cdot \sin\varphi - a \} \cdot \vec{i} \\ & + \{ [X_{O_c} - r \cdot \cos\theta] \cdot \sin\varphi - [Y_{O_c} + r \cdot \sin\theta] \cdot \cos\varphi \} \cdot \vec{j} + p \cdot \varphi \cdot \vec{k}. \end{aligned} \quad (25)$$

Condiția de înfășurare are forma

$$q = \begin{vmatrix} N_x & N_y & N_z \\ 0 & -\sin\alpha & \cos\alpha \\ r_{1x} & r_{1y} & r_{1z} \end{vmatrix} \cdot \varepsilon. \quad (26)$$

cu  $\varepsilon$  suficient de mic, ( $10^{-3} \dots 10^{-5}$ ).

Ansamblul ecuațiilor (22) și condiția (26) determină, pentru o variație discretă a parametrului  $\varphi$ , coordonatele punctelor în lungul curbei caracteristice.

$$C_{\Sigma} = \begin{pmatrix} X_i \\ Y_i \\ Z_i \end{pmatrix}^T \quad (27)$$

cu  $i=1\dots n$ ,

Cu transformarea de coordonate,

$$\begin{aligned} X_{1i} &= X_i - a; \\ Y_{1i} &= Y_i \cdot \cos\alpha + Z_i \cdot \sin\alpha; \\ Z_{1i} &= -Y_i \cdot \sin\alpha + Z_i \cdot \cos\alpha, \end{aligned} \quad (28)$$

se transferă coordonatele punctelor curbei caracteristice în sistemul  $X_1Y_1Z_1$ , sistem solidar cu axa sculei disc.

#### Aplicație numerică

Curba caracteristică, la generarea cu scula disc, va fi determinată în condițiile:

Coordonatele punctelor de definiție ale arcului – A[10; -8]; B[10; 8]; O<sub>C</sub>[12; 0], coordonate în mm;

distanța,  $a = 40$  mm;

pasul suprafeței elicoidale,  $pe = 310$  mm;

raza exterioară a piesei,  $Re = 12$  mm;

numărul de puncte considerate pe profil,  $n = 100$ ;

eroarea condiției de înfășurare,  $\varepsilon = 0,001$ .

În tabelul 7, sunt prezentate coordonatele punctelor de pe curba caracteristică, determinate prin cele două metode, metoda reprezentării topologice a suprafețelor, vis-a-vis de cea analitică.

În figura 14, sunt prezentate secțiunile axiale ale sculei disc calculate prin cele două metode.

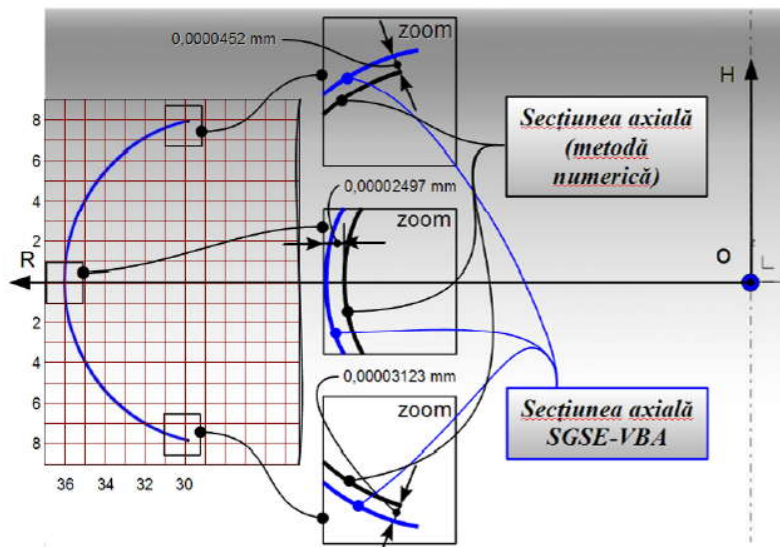


Fig. 14. Secțiunea axială a sculei disc calculată prin metodele CAD și analitic

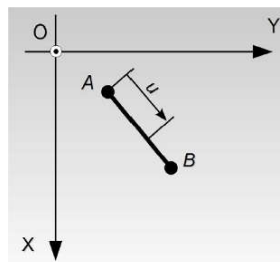
Este evident, din rezultatele numerice prezentate în tabelul 7 că, metoda reprezentării topologice a suprafețelor și metoda analitică clasică sunt identice din punctul de vedere al soluției oferite, pentru acest tip de problemă de generare analizată.

**Tabelul 7. Curba caracteristică, comparativ CAD - analitic**

<b>Nr. Crt.</b>	<b>X [mm] analitic</b>	<b>Y [mm] analitic</b>	<b>Z [mm] analitic</b>	<b>X [mm] CAD</b>	<b>Y [mm] CAD</b>	<b>Z [mm] CAD</b>
1	10.2558	-7.6692	1.6111	10.2558	-7.6693	1.6109
2	10.0381	-7.6215	1.6014	10.0391	-7.6218	1.6012
3	9.8227	-7.5685	1.5902	9.8235	-7.5688	1.5900
4	9.6083	-7.5099	1.5777	9.6093	-7.5103	1.5776
5	9.3956	-7.4458	1.5638	9.3967	-7.4462	1.5637
6	9.1847	-7.3762	1.5486	9.1858	-7.3767	1.5485
7	8.9757	-7.3012	1.5322	8.9767	-7.3017	1.5322
8	8.7689	-7.2208	1.5146	8.7697	-7.2212	1.5146
9	8.5643	-7.1351	1.4959	8.5649	-7.1354	1.4959
10	8.3620	-7.0440	1.4760	8.3624	-7.0442	1.4761
...	...	...	...	...	...	...
50	3.7545	-0.1090	0.0225	3.7546	-0.1087	0.0224
51	3.7545	0.1089	-0.0224	3.7546	0.1087	-0.0224
52	3.7609	0.3265	-0.0673	3.7609	0.3261	-0.0673
53	3.7736	0.5445	-0.1123	3.7735	0.5432	-0.1121
54	3.7926	0.7617	-0.1571	3.7924	0.7599	-0.1568
...	...	...	...	...	...	...
100	10.2558	7.6692	-1.6111	10.2559	7.6692	-1.6115

**Suprafață elicoidală cu generatoare transversală rectilinie**

În figura 15, este prezentată forma generatoarei transversale, rectilinie, a melcului de generat.



**Fig. 15. Generatoare frontală rectilinie**

Generatoarea rectilinie are ecuațiile:

$$\begin{cases} X = X_A + u \cdot \cos\theta; \\ Y = Y_A + u \cdot \sin\theta. \end{cases} \quad (29)$$

cu

$$\theta = \arctan \left( \frac{|Y_B - Y_A|}{|X_B - X_A|} \right) \quad (30)$$

și

$$u_{min}=0; u_{max}=\sqrt{((X_B - X_A)^2 + (Y_B - Y_A)^2)} \quad (31)$$

Ecuțiile suprafeței elicoidale sunt:

$$\Sigma \begin{cases} X = [X_A + u \cdot \cos\theta] \cdot \cos\varphi - [Y_A + u \cdot \sin\theta] \cdot \sin\varphi; \\ Y = [X_A + u \cdot \cos\theta] \cdot \sin\varphi + [Y_A + r \cdot \sin\theta] \cdot \cos\varphi; \\ Z = p \cdot \varphi. \end{cases} \quad (32)$$

cu  $u$  și  $\varphi$  – parametri variabili și  $p$  parametru elicoidal.

Pentru determinarea secțiunii axiale a sculei disc înfășurătoare a acestei suprafețe, se procedează conform algoritmului descris anterior.

### Aplicație numerică

Coordonatele punctelor – A[12;-8]; B[16; -2], în mm;

Distanța,  $a = 50$  mm;

Pasul elicoidal,  $p_e = 310$  mm;

Raza exterioară a piesei generate,  $R_e = 15$  mm;

Număr de puncte considerate pe profil,  $n = 100$ ;

Mărimea erorii admise,  $\varepsilon = 0,001$ .

În tabelul 8, sunt prezentate coordonatele punctelor aparținând curbei caracteristice determinate prin cele doua metode (metoda CAD vis-à-vis de metoda analitică).

**Tabelul 8.** *Coordonatele punctelor de pe curba caracteristică*

Nr. Crt.	X [mm] analitic	Y [mm] analitic	Z [mm] analitic	X [mm] CAD	Y [mm] CAD	Z [mm] CAD
1	13.0353	-6.1709	7.1961	13.0353	-6.1710	7.1963
2	13.0587	-6.1215	7.1347	13.0588	-6.1215	7.1347
3	13.0824	-6.0722	7.0732	13.0825	-6.0720	7.0730
4	13.1061	-6.0229	7.0117	13.1063	-6.0226	7.0114
5	13.1300	-5.9737	6.9502	13.1303	-5.9733	6.9498
6	13.1541	-5.9246	6.8887	13.1544	-5.9241	6.8882
7	13.1783	-5.8755	6.8272	13.1787	-5.8749	6.8266
8	13.2026	-5.8265	6.7657	13.2030	-5.8258	6.7650
9	13.2271	-5.7776	6.7043	13.2276	-5.7767	6.7034
10	13.2517	-5.7287	6.6428	13.2523	-5.7278	6.6418
...	...	...	...	...	...	...
50	14.3534	-3.8311	4.1897	14.3557	-3.8277	4.1853
51	14.3838	-3.7852	4.1286	14.3862	-3.7818	4.1242
52	14.4144	-3.7393	4.0675	14.4168	-3.7359	4.0630
53	14.4451	-3.6935	4.0064	14.4475	-3.6900	4.0019
54	14.4760	-3.6477	3.9454	14.4784	-3.6443	3.9409
...	...	...	...	...	...	...
100	16.0425	-1.6240	1.1572	16.0425	-1.6242	1.1572

În figura 16 și tabelul 8, sunt prezentate, comparativ, coordonatele și forma secțiunii axiale a sculei disc, pentru cele două metode abordate, analitică și metoda grafică.

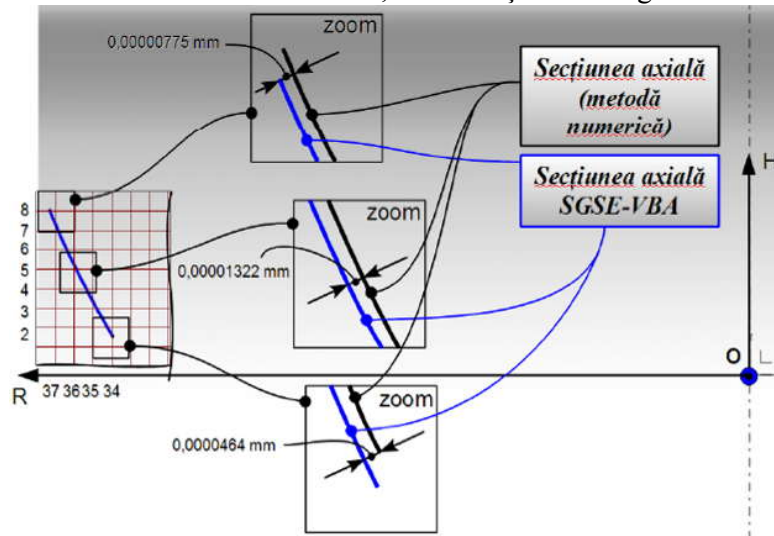


Fig. 16. Forma secțiunii axiale a sculei disc, metoda CAD și metoda analitică

### Concluzii

Rezultatele aplicării comparative a metodei generale de reprezentare topologică a suprafețelor, utilizând facilitățile mediului de proiectare grafică, și analitică, pentru determinarea secțiunii axiale a sculei disc, generatoare a unei suprafețe elicoidale cilindrice și de pas constant, în cadrul exemplelor prezentate, dovedește faptul că, cele două rezultate sunt, practic, identice, subliniind, pentru acest exemplu, calitatea metodei generale, propuse.

### Profilarea sculei-melc, cinematica generării

Se urmărește a se determina calitatea metodei grafice, de reprezentare topologică a suprafețelor, vis-a vis față de metodele analitice și în cazul profilării sculelor generatoare prin înfășurare cu contact punctiform – cazul sculei melc. Se prezintă, în cele ce urmează, o aplicație referitoare la profilarea suprafeței periferice primare a sculei melc reciproc înfășurătoare a unui arbore hexagonal, rezolvată prin cele două metode, metoda analitică și metoda grafică în mediul de programare CATIA

Se definesc sistemele:

$xyz$  este sistemul fix cu axa  $z$ , axa de rotație a axoidei asociată vârtejului de suprafețe de generat;

$x_0y_0z_0$  – sistemul fix, cu axa  $y_0$  suprapusă axei suprafeței periferice primare a sculei melc;

$XYZ$  – sistemul mobil solidar axoidei vârtejului de suprafețe de generat (cilindru de revoluție, de axă  $Z$ );

$\xi\eta\zeta$ – sistemul mobil solidar axoidei cremalierii (suprafață plană, suprapusă planului  $\xi\eta$ );

$X_1Y_1Z_1$  – sistemul mobil asociat suprafeței periferice primare a sculei-melc.

Este cunoscută cinematica de principiu a procesului de generare:

$$x = \omega_s^T(\varphi_1) \cdot X, \quad (33)$$

reprezentând rotația axoidei semifabricatului (cilindru de revoluție de rază  $R_{rp}$ ), solidară cu sistemul  $XYZ$ , de parametru  $\varphi_1$ – parametrul unghiular de mișcare;

$$x = \xi + a; a = \begin{pmatrix} -R_{rp} \\ -\lambda \\ 0 \end{pmatrix}, \quad (34)$$

reprezentând translația axoidei cremalierii (plan paralel cu planul  $\eta\zeta$ ), solidară cu sistemul  $\xi\eta\zeta$  și  $\lambda$  parametrul de mișcare;

$$x_0 = \omega_2^T(\varphi_2) \cdot X_1, \quad (35)$$

rotația sistemului  $X_1Y_1Z_1$  în jurul axei  $y_0$ , cu  $\varphi_2$  parametrul unghiular de mișcare.

De asemenea, sunt cunoscute condițiile:

$$\lambda = R_{r_p} \cdot \varphi_1 \quad (36)$$

condiția de rulare a axoidei semifabricatului și cea asociată cremalierii;

$$\lambda = p \cdot \varphi_2 \cdot \cos\omega; \quad (37)$$

reprezentând dependența dată de forma suprafeței periferice primare a sculei-melc (melc cilindric de pas constant, cu  $p$  – parametrul elicoidal).

Mișcarea relativă a sistemului solidar axei semifabricatului (a suprafeței de generat),  $XYZ$ , în raport cu sistemul de referință asociat spațiului cremalierii  $\xi\eta\zeta$  este dată de transformarea:

$$\xi = \omega_3^T(\varphi_1) \cdot X - a, \quad (38)$$

cu definiția punctului curent de pe suprafața de generat, ca o suprafață cilindrică cu generatoarele paralele cu direcția  $Z(\vec{k})$ :

$$\Sigma \begin{cases} X = X(u); \\ Y = Y(u); \\ Z = t. \end{cases} \quad (39)$$

pentru  $u$  și  $t$  variabile pe suprafața de generat.

#### **Determinarea formei suprafeței cremalierii**

Din (38) și (39), se determină familia de suprafețe  $\Sigma$ , în sistemul de referință al cremalierii,  $\xi\eta\zeta$ , cu  $\varphi$  parametru variabil,

$$\begin{pmatrix} \xi \\ \eta \\ \zeta \end{pmatrix} = \begin{pmatrix} \cos\varphi_1 & -\sin\varphi_1 & 0 \\ \sin\varphi_1 & \cos\varphi_1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} X(u) \\ Y(u) \\ t \end{pmatrix} - \begin{pmatrix} -R_{r_p} \\ -R_{r_p} \cdot \varphi \\ 0 \end{pmatrix}, \quad (40)$$

căreia i se asociază condiția de înfășurare, fie aceasta

$$[X - X(u)] \cdot X'_u + [Y - Y(u)] \cdot Y'_u = 0 \quad (41)$$

în care:

$$\begin{cases} X = R_{r_p} \cdot \cos\varphi_1 \\ Y = R_{r_p} \cdot \sin\varphi_1 \end{cases}, \quad (42)$$

reprezentând „condiția normalelor”, cu  $R_{r_p}$  raza cercului de rulare (raza axoidei cilindrice a semifabricatului, căreia îi este asociat vârtejul de suprafețe de generat).

Principial, înfășurătoarea familiei de suprafețe descrisă de ecuații de forma:

$$I \begin{cases} \xi = \xi(\varphi_1, t); \\ \eta = \eta(\varphi_1, t); \\ \zeta = \zeta(\varphi_1, t), \end{cases} \quad (43)$$

reprezentând forma flancului cremalierii reciproc înfășurătoare vârtejului de suprafețe de generat.

#### **Profilarea suprafeței periferice primare a sculei-melc**

Cunoscută fiind suprafața flancului cremalierii, ecuațiile (38), se propune determinarea caracteristicii (curba de contact) la contactul acesteia cu viitoarea suprafață periferică primară a sculei melc, prin utilizarea metodei descompunerii mișcării elicoidale.

Mișcarea elicoidală generatoare a suprafeței periferice primare a sculei-melc,  $(\vec{V}, p)$ , se descompune într-o sumă de mișcări echivalente: mișcarea de translație, după direcția  $\vec{t}$  a versorului generatoarelor suprafețelor cilindrice – flancul cremalierii – și o mișcare de rotație de axă  $\vec{A}$ , paralelă cu  $\vec{V}$  (axa elicoidului sculei melc) și aflată la distanța:

$$a = p \cdot \tan(\theta) \quad (44)$$

de axa suprafeței elicoidale  $\vec{V}$ . Se definește mărimea parametrului unghiular  $\theta$ , ca fiind unghiul între versorul  $\vec{t}$  (versorul generatoarelor suprafeței cremalierii) și axa elicoidului  $(\vec{V}, p)$ .

Astfel, caracteristica suprafeței  $I$ , în mișcarea compusă translație în lungul generatoarei  $\vec{t}$  și rotație în jurul axei  $\vec{A}$ , nu depinde de acea componentă a mișcării în decursul căreia suprafața se autogenerază, fiind îndeplinită identitatea

$$\vec{N}_I \cdot \vec{t} \equiv 0, \quad (45)$$

în care  $\vec{N}_I$  normala la suprafața  $I$  a flancului cremalierii (suprafața cilindrică) este întotdeauna perpendiculară pe generatoarea proprie și, deci, condiția pentru determinarea caracteristicii, în mișcarea elicoidală  $(\vec{V}, p)$ , va depinde numai de mișcarea de rotație în jurul axei  $\vec{A}$ .

Ca urmare, caracteristica suprafeței cilindrice  $I$  – flancul sculei cremalieră – în mișcarea elicoidală de axă  $\vec{V}$  și parametru elicoidal  $p$ , este definită ca fiind proiecția axei  $\vec{A}$  pe suprafața  $I$ . Aceasta se interpretează ca reprezentând locul geometric al punctelor de pe suprafața cilindrică  $I$ , pentru care normalele la aceasta intersectează axa  $\vec{A}$ .

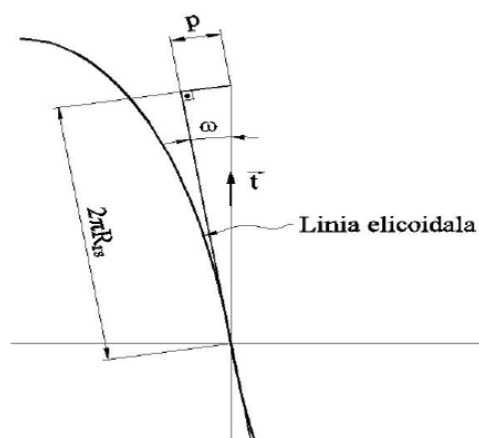
Se definesc, axa  $\vec{A}$ , în sistemul  $x_0y_0z_0$ ,

$$\vec{A} = -\cos\omega \cdot \vec{j} + \sin\omega \cdot \vec{k}, \quad (46)$$

și normala la suprafața  $I$ , principial, în forma,

$$\vec{N}_I = N_\xi \cdot \vec{i} + N_\eta \cdot \vec{j} + N_\zeta \cdot \vec{k}, \quad (47)$$

cu  $N_\xi, N_\eta, N_\zeta$  parametrii directori.



**Fig. 17.** Elicea cilindrului de rază  $R_{rs}$  a suprafeței periferice primare a sculei-melc

Mărimea parametrului  $\omega$  se determină din condiția ca elicea aparținând elicoidului  $(\vec{V}, p)$  aflată pe cilindrul de raza  $R_{rs}$  să fie paralelă cu versorul  $\vec{t}$  al generatoarei flancului cilindric al cremalierii,



$$\tan\omega = \frac{2\pi p}{2\pi R_{rs}} = \frac{p}{R_{rs}} \quad (48)$$

cu  $p$  – parametrul elicoidal al suprafeței periferice primare a sculei melc.

Astfel, condiția pentru determinarea curbei caracteristice devine:

$$\left(\vec{A}, \vec{N}_1, \vec{r}_1\right) = 0 \quad (49)$$

în care  $\vec{r}_1$  este de forma:

$$\vec{r}_1 = [x_0(\varphi_1) - a] \cdot \vec{i} + y_0(\varphi_1) \cdot \vec{j} + t \cdot \vec{k} \quad (50)$$

După înlocuiri, forma (45) devine:

$$\begin{vmatrix} N_z & N_\eta & N_\zeta \\ x_0(\varphi_1) - a & y_0(\varphi_1) & t \\ 0 & -\cos\omega & \sin\omega \end{vmatrix} = 0 \quad (51)$$

Principial, condiția (51) reprezintă o legătură între parametrii variabili  $\varphi_1$  și  $t$  de tipul:

$$q(\varphi_1, t) = 0 \quad (52)$$

Ansamblul ecuațiilor (43) și (52), reprezintă un loc geometric pe suprafața  $I$  semnificând caracteristica suprafeței  $I$  în mișcarea elicoidală de axă  $\vec{V}$  și parametru elicoidal  $p$  – axa și parametrul elicoidului căreia îi aparține suprafața periferică primară a sculei-melc,  $S$ , reciproc înfășurătoare suprafeței  $\Sigma$  – suprafața de generat.

Perechile de valori ale parametrilor  $\varphi_1$  și  $t$ , pentru care este satisfăcută condiția (52), prin înlocuirea în ecuațiile flancului cremalierii (43) determină matricea

$$C_1 = \|\xi_i \eta_i\|, (i=1..n), \quad (53)$$

reprezentând coordonatele curbei caracteristice  $C_{I,S}$ .

Curba  $C_{I,S}$ , cunoscută în formă numerică, reprezintă curba de tangență între suprafața  $I$  – flancul cremalierii și suprafața periferică primară a sculei elicoidale  $S$ , freza melc generatoare, prin înfășurare, a profilului  $\Sigma$ .

În mișcarea elicoidală de axă  $\vec{V}$  și parametru  $p$ , curba  $C_{I,S}$  generează suprafața periferică primară a viitoarei scule-melc  $S$ ,

$$\begin{vmatrix} X_1 \\ Y_1 \\ Z_1 \end{vmatrix} = \begin{vmatrix} \cos\varphi_2 & 0 & \sin\varphi_2 \\ 0 & 1 & 0 \\ -\sin\varphi_2 & 0 & \cos\varphi_2 \end{vmatrix} \cdot \begin{vmatrix} 1 & 0 & 0 \\ 0 & \cos\omega & -\sin\omega \\ 0 & \sin\omega & \cos\omega \end{vmatrix} \cdot \begin{vmatrix} X_{0i} \\ Y_{0i} \\ Z_{0i} \end{vmatrix} + \begin{vmatrix} 0 \\ p \cdot \varphi_2 \\ 0 \end{vmatrix} \quad (i=1 \dots n) \quad (54)$$

Se ajunge la forma de principiu:

$$S \begin{cases} X_1 = X_1[x_{0i}, y_{0i}, z_{0i}, \varphi_2]; \\ Y_1 = Y_1[x_{0i}, y_{0i}, z_{0i}, \varphi_2]; \\ Z_1 = Z_1[x_{0i}, y_{0i}, z_{0i}, \varphi_2], \end{cases} \quad (55)$$

reprezentând ecuațiile suprafeței periferice primare ale sculei-melc – suprafața  $S$ .

Asociind suprafeței  $S$  condiția,

$$Z_1 = 0 \quad (56)$$

se obține secțiunea axială a sculei-melc  $S_A$ , principial, în forma:

$$S \begin{cases} X_1 = X_1[x_{0i}, y_{0i}, z_{0i}, \varphi_2] \\ Y_1 = Y_1[x_{0i}, y_{0i}, z_{0i}, \varphi_2] \end{cases} \quad (57)$$

$i=1,2,\dots,n$ , cu variabila  $\varphi_{2A}$ , reprezentând mărimea parametrului  $\varphi_2$  corespunzător secțiunii axiale,  $Z_1 = 0$ .

### Aplicație numerică

Se consideră un arbore hexagonal cu geometria prezentată în figura 18 căruia i se dorește a se afla scula melc care îl generează, prin metoda analitică și, de asemenea, prin metoda grafică de reprezentare topologica suprafețelor, oferită de mediul de proiectare grafică CATIA.

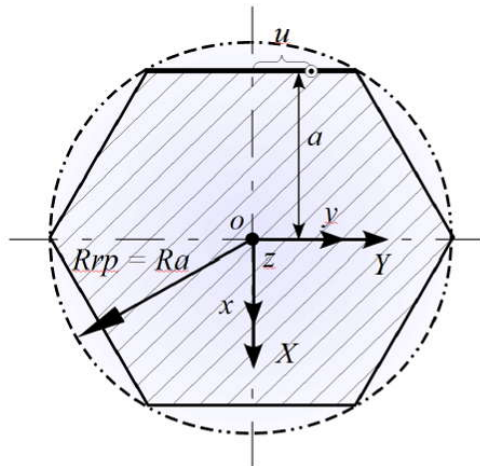


Fig. 18. Profilul arborelui hexagonal

Ecuțiile profilului rectiliniu reprezentând flancul arborelui hexagonal – profilul  $\Sigma$ , în acest caz concret, sunt:

$$\Sigma \begin{cases} X = -a; \\ Y = -u; \\ Z = t \end{cases} \quad (58)$$

cu  $u$  și  $t$  variabile și

$$u \in \left[ \frac{-R_{rp}}{2}, \frac{R_{rp}}{2} \right]. \quad (59)$$

Familia de profiluri  $\Sigma$  în sistemul sculei cremalieră este dată de transformarea:

$$\begin{pmatrix} \xi \\ \eta \\ \zeta \end{pmatrix} = \begin{pmatrix} \cos\varphi_1 & -\sin\varphi_1 & 0 \\ \sin\varphi_1 & \cos\varphi_1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} -a \\ -u \\ t \end{pmatrix} - \begin{pmatrix} -R_{rp} \\ -R_{rp} \cdot \varphi_1 \\ 0 \end{pmatrix}, \quad (60)$$

sau

$$\Sigma(\varphi_1) \begin{cases} \xi = -a \cdot \cos\varphi_1 + u \cdot \sin\varphi_1 + R_{rp}; \\ \eta = -a \cdot \sin\varphi_1 - u \cdot \cos\varphi_1 + R_{rp} \cdot \varphi_1; \\ \zeta = t. \end{cases} \quad (61)$$

Condiția de înfășurare a familiei de profiluri se reduce la:

$$u = -R_{rp} \cdot \sin\varphi_1, \quad (62)$$

Ansamblul ecuațiilor (61) și (62) determină profilul cremalierii:

$$I \begin{cases} \xi = -a \cdot \cos\varphi_1 + R_{rp} \cdot \cos^2\varphi_1; \\ \eta = -a \cdot \sin\varphi_1 + R_{rp} \cdot \sin\varphi_1 \cdot \cos\varphi_1 + R_{rp} \cdot \varphi_1; \\ \zeta = t. \end{cases} \quad (63)$$

Parametrii directori ai normalei la suprafața intermediară (63) sunt:

$$N_I \begin{cases} N_\xi = a \cdot \cos\varphi_1 + 2R_{rp} \cdot \cos^2\varphi_1; \\ N_\eta = -a \cdot \sin\varphi_1 + 2R_{rp} \cdot \sin\varphi_1 \cdot \cos\varphi_1; \\ N_\zeta = 0. \end{cases} \quad (64)$$

Se poate scrie în acest fel, condiția de înfășurare (51), în care au fost definiți toți vectorii. Este posibilă astfel determinarea curbei caracteristice, la contactul dintre cremaliera *I* și suprafața periferică primară a viitoarei scule – melc *S*.

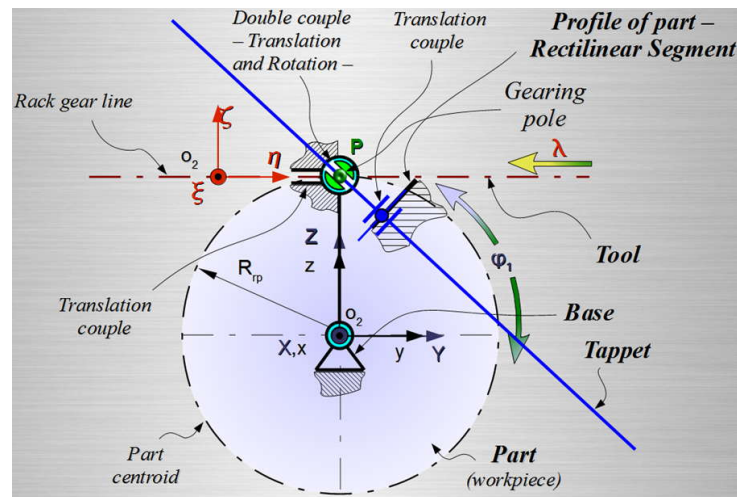
Pentru datele de intrare prezentate în tabelul 9 se determină profilul secțiunii axiale a sculei melc, în sistemul  $X_1Y_1Z_1$ , vezi tabelul 10.

**Tabelul 9.** Parametrii de intrare ai profilului secțiunii transversale a arborelui hexagonal

<b>Simbol</b>	<b>Descriere</b>	<b>Valoare</b>	<b>UM</b>
$R_{rp}$	Raza de rulare a centroidei asociată arborelui	50	[mm]
$R_a$	Raza exterioară a hexagonului	50	[mm]
$Da$	Distanța dintre axa piesei și axa sculei melc	100	[mm]

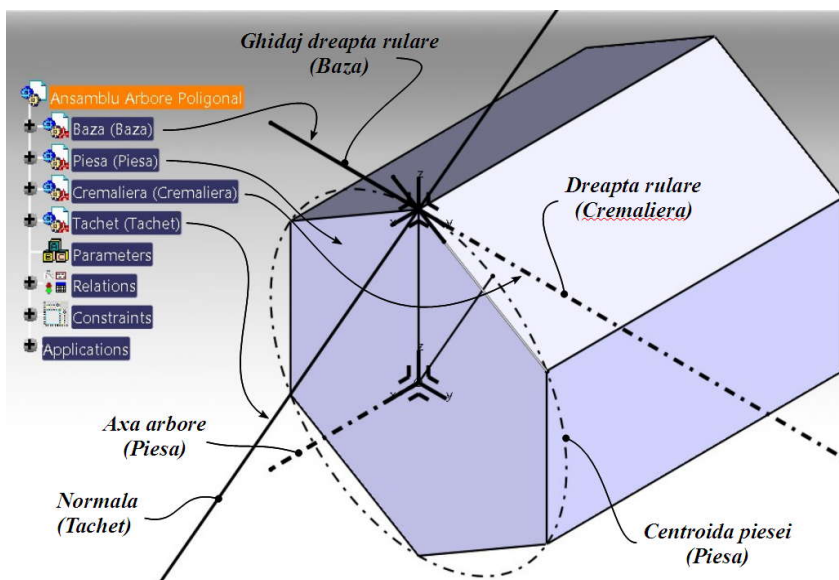
### Metoda cinematică în mediul grafic de proiectare CATIA

Într-o primă etapă, se determină profilul cremalierii, cu ajutorul unui mecanism MGMC, conceput pentru cazul unui segment de dreaptă, care este alcătuit din patru elemente de tip *Part*, create în mediul de proiectare grafică CATIA: *Bază*, *Piesă*, *Tachet* și *Sculă* (vezi figura 19).



**Fig. 19.** Mecanismul MGMC

Fișierul *Baza* conține partea fixă a mecanismului și reprezintă liniile de ghidare cu ajutorul cărora se pot cupla celelalte elemente. Arborele hexagonal este modelat în fișierul *Piesa* cu datele de intrare din tabelul 9. *Cremaliera* este fișierul în care este construită linia ce reprezintă dreapta de rulare a cremalierii de referință, *I*. Fișierul *Tachet* este alcătuit din două linii: *Tangenta*, care reprezintă tangenta ce culisează pe segmentul reprezentând latura hexagonului și *Normala*, care reprezintă normala dusă din punctul de start (capătul laturii), prin polul angrenării.

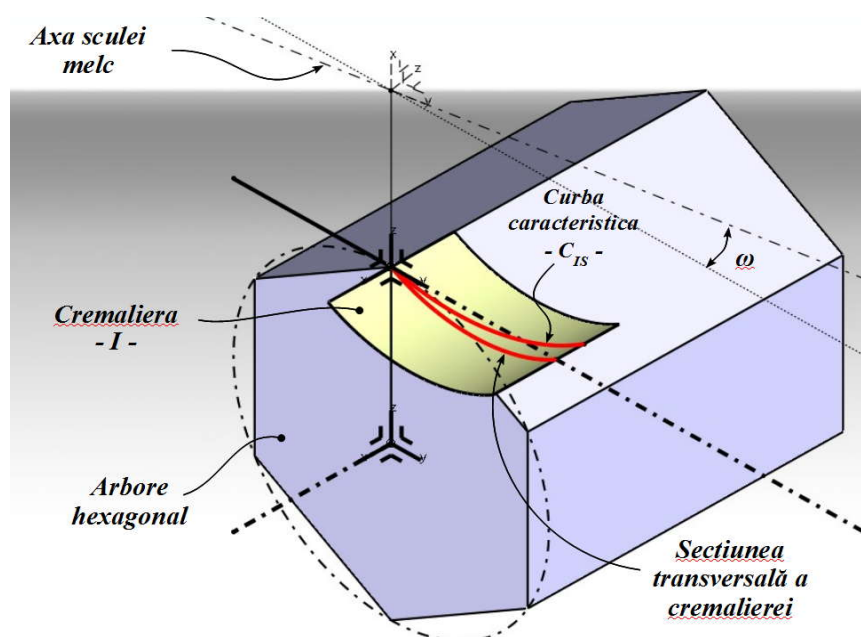


**Fig. 20.** MGMC în mediul de proiectare CATIA — elementele mecanismului

Aceste elemente, sunt asamblate și restricționate, realizând cuplele cinematice care să satisfacă condiția de rulare a mecanismului și anume, normala să rămână permanent în contact cu polul angrenării.

În mediul de proiectare *DMU Kinematics*, profilului cremalierii se determină ca traiectorie a punctului de contact dintre normală și latura hexagonului, în sistemul de referință al cremalierii.

În mediul de proiectare *Generative Shape Design*, se obține suprafața periferică primară a cremalierii de referință, cu ajutorul comenzii *Extrude*, reprezentând suprafața de contact, virtuală, simultan înfășurătoare cu suprafața arborelui hexagonal și a viitoarei suprafețe elicoidale a sculei melc, vezi figura 21. În această figură, este reprezentată cremaliera arborelui hexagonal curba caracteristică la contactul acesteia cu suprafața periferică primară a viitoarei scule melc precum și secțiunea transversală a cremalierii.



**Fig. 21.** Curba caracteristică: cremalieră – suprafața periferică primară a sculei melc

### Unghiul de înclinare al axei sculei melc

Se stabilește unghiul de înclinare al axei sculei melc în raport cu planul secțiunii transversale al arborelui hexagonal,  $\omega$ , vezi și figura 19, cunoscând relația

$$\operatorname{tg}\omega = \frac{P_e}{2 \cdot \pi \cdot R_{rs}}, \quad (65)$$

în care:  $P_e$  este pasul axial al suprafeței elicoidale – suprafața periferică primară a sculei melc,  $S$ ;

$R_{rs}$  – raza de rulare a suprafeței periferice primare a sculei melc cu cremaliera de referință  $I$ .

Este cunoscută, de asemenea, relația dintre pasul cremalierii de referință și pasul sculei melc, vezi figura 19,

$$\cos\omega = \frac{P_{cr}}{P_e}. \quad (66)$$

Din relațiile (66) și (65), rezultă formula de calcul a unghiului  $\omega$ , care este definit în funcție de elementele cremalierii  $P_{cr}$  și raza cilindrului de divizare a sculei melc,  $R_{rs}$ ,

$$\omega = \operatorname{asin}\left(\frac{P_{cr}}{2 \cdot \pi \cdot R_{rs}}\right), \quad (67)$$

În figura 22, este prezentată proiecția dreptei ce trece prin polul angrenării, paralelă cu axa melcului și înclinată cu unghiul  $\omega$ , proiecție ce determină pe suprafața cremalierii de referință, curba caracteristică, de contact, a suprafeței periferice primare a sculei melc cu cremaliera,

$C_{IS}$ .

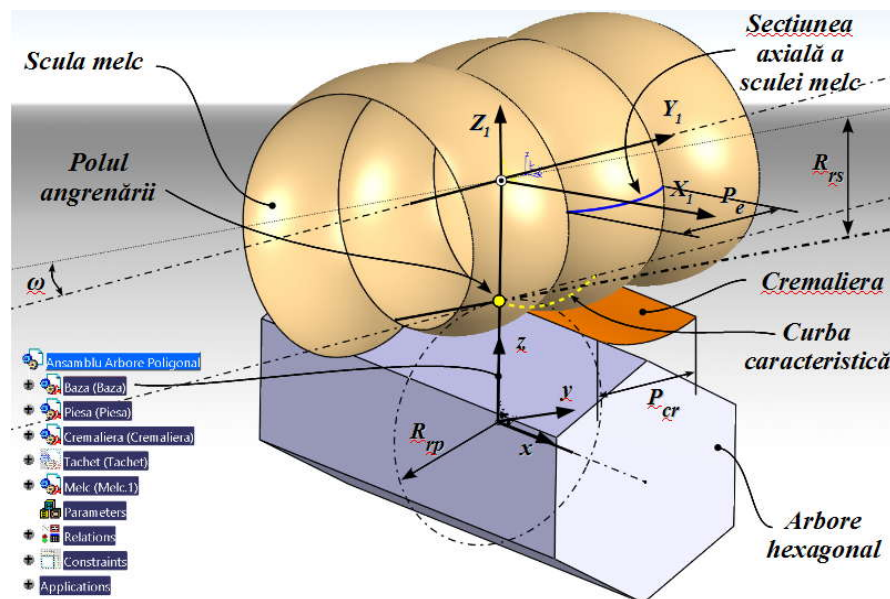
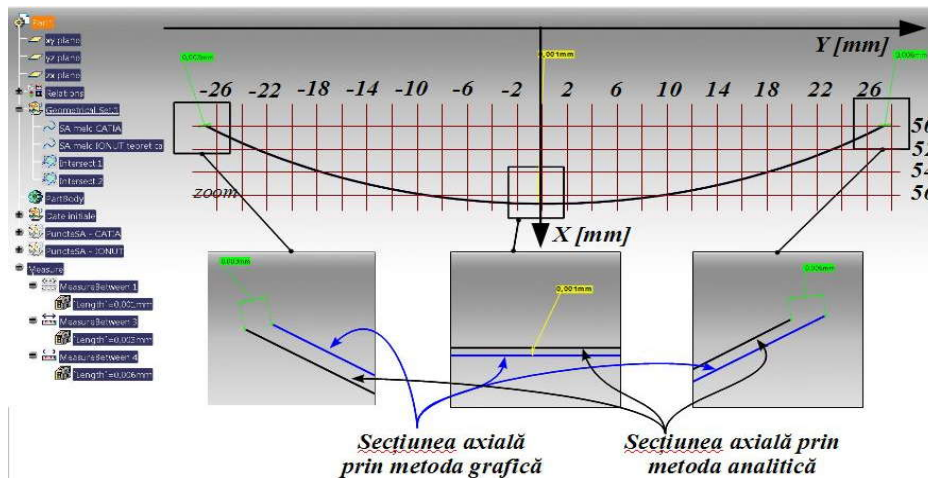


Fig. 22. Curba caracteristică

În tabelul 10 și figura 23, sunt prezentate coordonatele și forma profilului secțiunii axiale a sculei melc obținute prin cele două metode, analitică și cea în mediul de proiectare CATIA) (reprezentarea topologică, în mediul de proiectare grafică a suprafețelor)

**Tabelul 10.** *Coordonatele secțiunii axiale a sculei melc*

Nr. crt.	Metoda grafică			Metoda analitică		
	$X_1$ [mm]	$Y_1$ [mm]	$Z_1$ [mm]	$X_1$ [mm]	$Y_1$ [mm]	$Z_1$ [mm]
1	50.000	-26.551	0.000	50.001	-26.554	0.000
2	50.274	-26.064	0.000	50.274	-26.066	0.000
3	50.541	-25.572	0.000	50.541	-25.574	0.000
4	50.802	-25.078	0.000	50.802	-25.078	0.000
5	51.058	-24.581	0.000	51.056	-24.580	0.000
...	...	...	...	...	...	...
50	56.698	-0.280	0.000	56.697	-0.283	0.000
51	56.698	0.280	0.000	56.697	0.277	0.000
52	56.693	0.839	0.000	56.692	0.836	0.000
53	56.682	1.398	0.000	56.681	1.395	0.000
54	56.666	1.957	0.000	56.665	1.954	0.000
...	...	...	...	...	...	...
96	51.058	24.580	0.000	51.059	24.575	0.000
97	50.802	25.078	0.000	50.804	25.073	0.000
98	50.541	25.572	0.000	50.544	25.568	0.000
99	50.274	26.063	0.000	50.277	26.060	0.000
100	50.000	26.551	0.000	50.001	26.545	0.000



**Fig. 23.** *Forma profilului secțiunii axiale a frezei melc*

**Concluzii**

Determinarea profilului secțiunii axiale a sculei melc, generatoare a arborelui hexagonal, reliefează faptul că rezultatele numerice obținute prin cele două metode sunt practic identice. Erorile de profil determinate prin cele două metode sunt de ordinul  $10^{-3}$  mm, ceea ce dovedește calitatea metodei grafice de reprezentare topologică a suprafețelor de a rezolva riguros problematica profilării sculelor care generează prin înfășurare cu contact punctiform.

Metoda grafică dezvoltată în mediul de proiectare CATIA se bazează pe mecanisme de generare virtuală a profilului cremalierii de referință și, de asemenea, pe o metodă adecvată facilităților oferite de CATIA, pentru determinarea curbei caracteristice, la contactul unei suprafețe elicoidale cu o suprafață cilindrică.

### Metoda de modelare a generării prin rulare

O problema diferită care poate fi analizată prin metoda generală de reprezentare topologică, în mediul de proiectare grafică CATIA, se refera la modelarea generării suprafețelor, care este necesară pentru determinarea erorilor geometrice de generare, ca factor, adeseori, dominant în determinarea erorii de prelucrare.

Realizarea fizică a sculei este însoțită de erori. În acest fel, forma reală a sculei este diferită de forma teoretică a acesteia și, ca urmare, generarea cu o astfel de sculă este însoțită de o eroare inerentă – eroarea geometrică. În raport cu aceasta, ceilalți factori care conduc la apariția erorii de prelucrare o pot amplifica sau, în situații speciale, o pot diminua.

Descrierea suprafeței periferice primare a sculei se face numeric, deoarece aceasta poate fi cunoscută prin măsurare directă pe mașini de măsurare 3D. Reprezentarea acesteia se face sub formă de nor de puncte. O soluție simplificată este aceea în care este cunoscută prin măsurare o generatoare a suprafeței, de regulă o curbă plană, reprezentând o secțiune axială (sau transversală) a suprafeței periferice primare efective a sculei. Cu aceasta se generează o suprafață periferică virtuală a sculei, care servește la modelarea, ulterioară, a suprafeței virtuale a piesei efectiv generate.

Forma specifică în care este cunoscută suprafața periferică primară a sculei (nor de puncte) impune utilizarea pentru determinarea înfășurătoarei acesteia, în procesul de generare virtuală, a unei metode de înfășurare specifice – metoda distanței minime. Evident, soluția propusă pentru determinarea erorii geometrice de generare este limitativă dar poate, adesea, constitui o alternativă - controlul virtual, înaintea prelucrării propriu zise a suprafeței generate. De asemenea, soluția propusă poate fi utilizată la determinarea modificărilor constructive ale sculei, în vederea diminuării erorii geometrice și, implicit, a erorii de generare a suprafeței prelucrate.

### Modelarea generării suprafețelor la prelucrarea cu scula cremalieră.

Problematika modelării generării cu scula cremalieră trebuie tratată ca o problemă „inversă” celei referitoare la profilarea acestui tip de sculă.

Fiind cunoscut, în formă numerică, profilul sculei cremalieră, în mișcarea de rulare a centroidelor asociate sculei și vârtejului de suprafețe de generat, se determină forma profilului suprafeței de generat – modelul virtual al suprafeței.

Cinematica generării este cea cunoscută, figura 24.

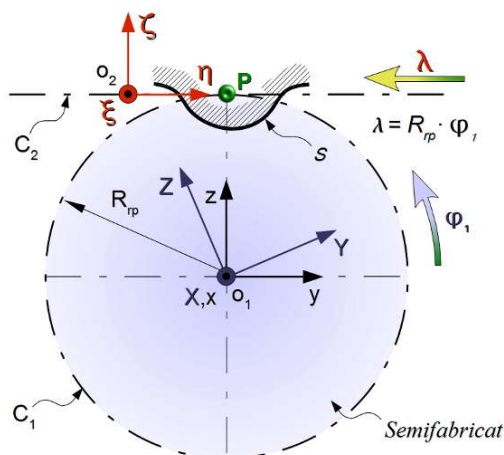


Fig. 24. Cinematica generării cu scula cremalieră

Se definesc, în legătură cu figura 24:

- $C_1$  este centronda, arc de cerc de raza  $R_{rp}$ , asociată vârtejului de generat;
- $C_2$  – centronda, rectilinie, asociată profilului  $S$  al sculei cremalieră;
- $\xi\eta\zeta$  – sistem de referință mobil, asociat sculei cremalieră;
- $XYZ$  – sistem de referință mobil, asociat semifabricatului;
- $xyz$  – sistem de referință fix.

### Cinematica procesului de generare

Sunt definite mișcările absolute ale cremalierii  $S$  și spațiul asociat semifabricatului, în raport cu sistemul de referință fix:

$$x = \omega_1^T(\varphi_1) \cdot X \quad (68)$$

și

$$x = \zeta + \begin{pmatrix} 0 \\ -R_{rp} \cdot \varphi_1 \\ R_{rp} \end{pmatrix} \quad (69)$$

în care  $\varphi_1$  este parametru unghiular al mișcării de rotație în jurul axei  $x$ .

Există, acum, în baza mișcărilor (68) și (69), posibilitatea determinării mișcării relative:

$$X = \omega_1(\varphi_1) \cdot \left[ \zeta + \begin{pmatrix} 0 \\ -R_{rp} \cdot \varphi_1 \\ R_{rp} \end{pmatrix} \right] \quad (70)$$

a cremalierii (spațiul  $\xi\eta\zeta$ ) în raport cu sistemul de referință al semifabricatului.

Dacă se presupune cunoscut profilul sculei cremalieră, în formă numerică, sub forma matricei de puncte

$$S = \begin{pmatrix} 0 \\ \eta_i \\ \zeta_i \end{pmatrix}_{i=1\dots n} \quad (71)$$

atunci, din (70), se poate determina un masiv de puncte aparținând familiei de profiluri  $S$ ,

$$\begin{pmatrix} X_{i,k} \\ Y_{i,k} \\ Z_{i,k} \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos(\varphi_1) & \sin(\varphi_1) \\ 0 & -\sin(\varphi_1) & \cos(\varphi_1) \end{pmatrix} \cdot \left[ \begin{pmatrix} 0 \\ \eta_i \\ \zeta_i \end{pmatrix} + \begin{pmatrix} 0 \\ -R_{rp} \cdot \varphi_1 \\ R_{rp} \end{pmatrix} \right]_{(i=1,\dots,n);(k=1,\dots,m)} \quad (72)$$

pentru o variație incrementală a parametrului  $\varphi_1$ , fie

$$\varphi_1 = k \cdot (\Delta\varphi_1) \quad (73)$$

în care:  $k=1,2,\dots,m$  și  $\Delta\varphi_1$  - increment.

Profilul  $\Sigma$  al suprafeței constituente se determină asociind masivului de puncte determinat de (72),

$$S_{i,k} = \begin{pmatrix} 0 \\ Y_{i,k} \\ Z_{i,k} \end{pmatrix}_{(i=1,\dots,n);(k=1,\dots,m)} \quad (74)$$

o formă a condiției de înfășurare, metoda „distanței minime”, vezi și figura 25, care are forma:

$$d_{min} = \left\{ \sqrt{[Y_{(i,k)} - R_{rp} \cdot \sin(k \cdot \Delta\varphi_1)]^2 + [Z_{(i,k)} - R_{rp} \cdot \sin(k \cdot \Delta\varphi_1)]^2} \right\}_{min} \quad (75)$$



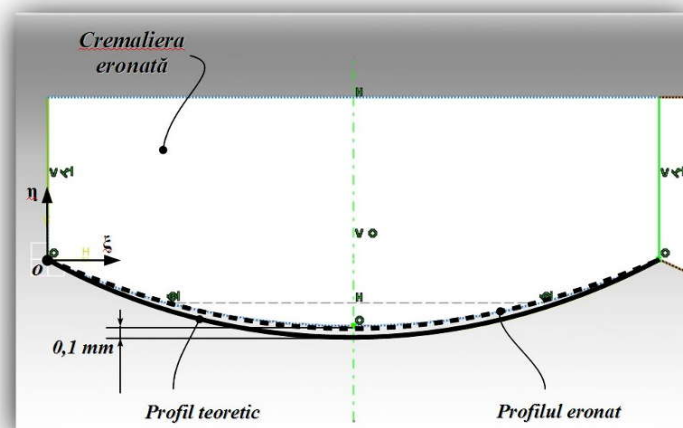


Pentru acest arbore, s-a determinat profilul secțiunii transversale a cremalierii, rulând un macro (*MGMC-VBA*), dedicat acestei aplicații, în care s-au introdus valorile parametrilor de intrare reprezentând diametrul exterior al arborelui (fiind considerat, în acest exemplu, egal cu cilindrul de rulare) și numărul de laturi ( $NbS = 6$ ). De asemenea, se acceptă  $Re = Rr = 50$  mm. În tabelul 11, sunt trecute coordonate de pe profilul secțiunii cremalierii teoretice și eronată.

**Tabelul 11.** Coordonatele punctelor de pe profilul cremalierii, teoretice și eronate,  $\xi\eta$

		$\xi$ [mm]	$\eta$ [mm]			$\xi$ [mm]	$\eta$ [mm]
<b>Puncte pe profilul eronat</b>	<b>1</b>	0.0000	0.0000	<b>Puncte pe profilul teoretic</b>	<b>1</b>	0.0000	0.0000
	<b>2</b>	5.4844	-2.5733		<b>2</b>	5.4465	-2.6760
	<b>3</b>	11.2165	-4.5336		<b>3</b>	11.1835	-4.6565
	<b>4</b>	17.1293	-5.8523		<b>4</b>	17.1102	-5.9658
	<b>5</b>	23.1509	-6.5156		<b>5</b>	23.1450	-6.6174
	<b>6</b>	29.2089	-6.5156		<b>6</b>	29.2148	-6.6174
	<b>7</b>	35.2305	-5.8523		<b>7</b>	35.2496	-5.9658
	<b>8</b>	41.1433	-4.5336		<b>8</b>	41.1763	-4.6565
	<b>9</b>	46.8756	-2.5733		<b>9</b>	46.9133	-2.6761
	<b>10</b>	52.3599	0.0000		<b>10</b>	52.3599	0.0000

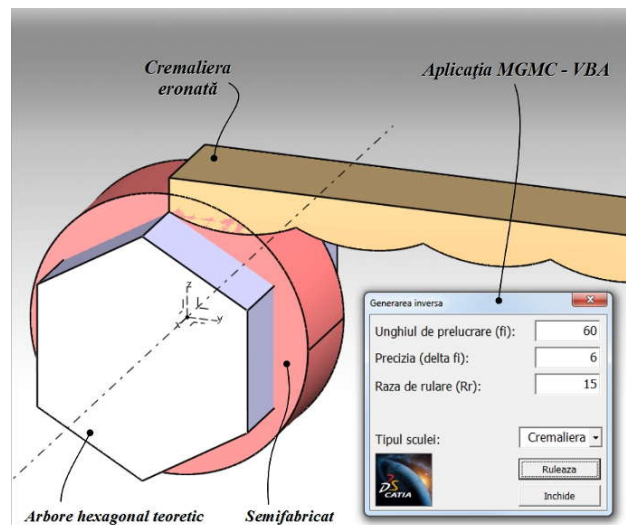
În figura 27, este prezentat modelul 2D al sculei cremalieră, modificată față de modelul teoretic al acesteia, ca urmare a erorii, rezultate în urma procesului de execuție a sculei sau a unui proces de exploatare a acesteia. Figura reprezintă forma profilului eronat al cremalierii efective, cu un factor de amplificare de 10 ori, pe direcția axei  $\eta$ .



**Fig. 27.** Cremaliera „eronată” - modelul 2D

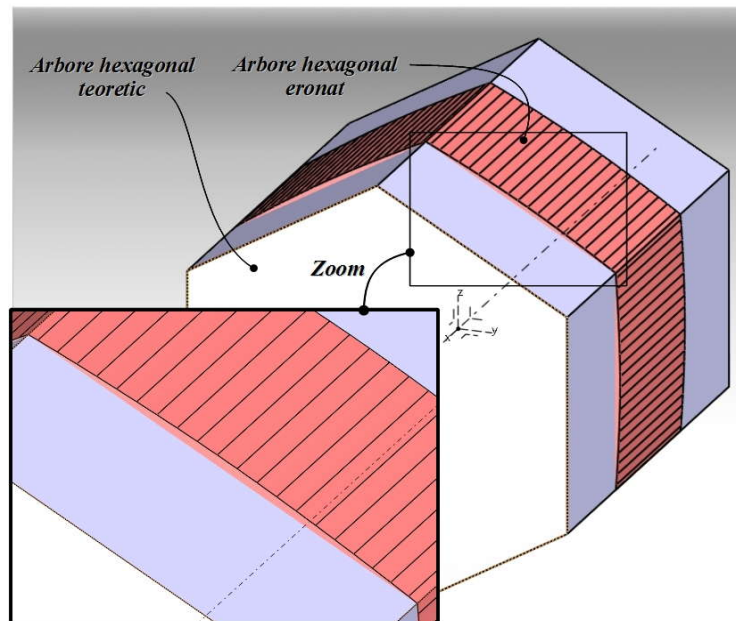
Se dorește a se determina profilul secțiunii transversale a modelului 3D al arborelui virtual, care va rezulta în urma simulării procesului de prelucrare cu această cremalieră „eronată”.

Pentru aceasta, se consideră modelul solid al semifabricatului reprezentat de un cilindru de rază mai mare decât raza periferică a arborelui poligonal de generat. Se lansează macro *MGMC-VBA* dedicat, în fereastra căruia se trec valorile limite ale unghiurilor de rulare și valoarea incrementului de unghi de rotație, pentru fiecare pas al simulării procesului de prelucrare virtuală (vezi figura 28)



**Fig. 28.** Generarea inversă. Aplicația MGMC - VBA

Aplicația *MGMC - VBA* specifică automatizează comenzile de copiere și inserare a solizilor în pozițiile corespunzătoare, rotite, și, respectiv, translate (*Copy, Paste as result, Rotate, Translate*), precum și automatizarea procesului de generare, prin extragerea solidului reprezentând scula (comanda *Remove*), din semifabricatul virtual.



**Fig. 29.** Modelele solide ale arborelui hexagonal, teoretic și eronat

În urma generării virtuale, se analizează profilul secțiunii transversale al arborelui rezultat în planul *YZ*, figurile 21 și 22, profilul obținut cu comanda *Intersection*, obținându-se astfel coordonate de pe acest profil.

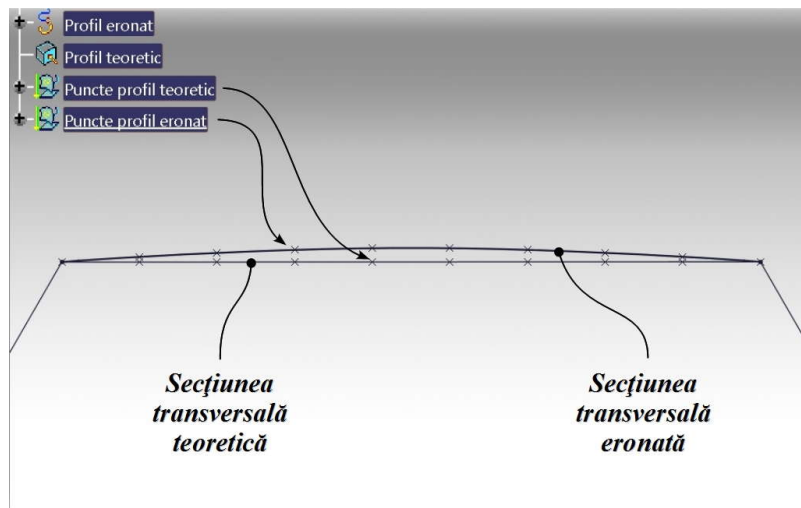


Fig. 30. Modelele secțiunii transversale a arborelui, teoretic și eronat

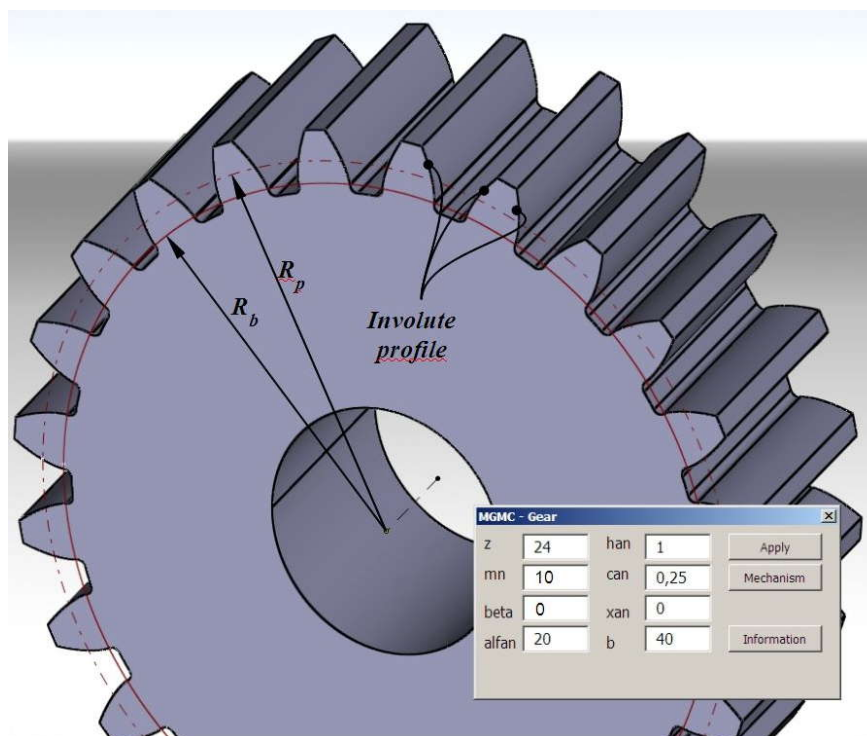
Coordonatele punctelor de pe acest profil pot fi exportate și analizate, tabelul 12.

Tabelul 12. Coordonatele punctelor de pe profilul teoretic și modelat, YZ

	Y[mm]	Z[mm]		Y[mm]	Z[mm]		
<b>Puncte eronate</b>	<b>1</b>	43.301	25.000	<b>Puncte teoretice</b>	<b>1</b>	43.301	25.000
	<b>2</b>	43.408	19.445		<b>2</b>	43.301	19.444
	<b>3</b>	43.428	13.890		<b>3</b>	43.301	13.889
	<b>4</b>	43.417	8.334		<b>4</b>	43.301	8.333
	<b>5</b>	43.403	2.778		<b>5</b>	43.301	2.778
	<b>6</b>	43.403	-2.778		<b>6</b>	43.301	-2.778
	<b>7</b>	43.417	-8.334		<b>7</b>	43.301	-8.333
	<b>8</b>	43.428	-13.890		<b>8</b>	43.301	-13.889
	<b>9</b>	43.408	-19.446		<b>9</b>	43.301	-19.444
	<b>10</b>	43.301	-25.000		<b>10</b>	43.301	-25.000

### Metoda cinematică în mediul grafic de proiectare CATIA pentru modelarea topologică a generării unei roți dințate cilindrice cu dinți drepți

Se prezintă exemplul modelării generării unei roți dințate cilindrice cu dinți drepți realizată cu ajutorul aplicației *MGMC-VBA*. În figura 31 se prezintă modelul solid al roții dințate teoretice pentru parametrii din tabelul 14.



**Fig. 31.** Roata dințată teoretică și aplicația MGMC-VBA

**Tabelul 13.** Parametrii roții dințate teoretice

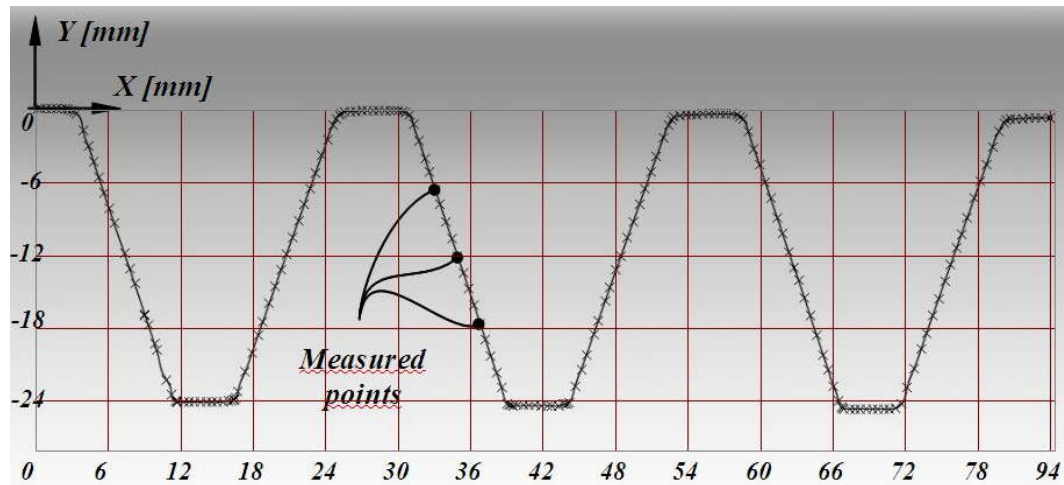
<i>Denumirea</i>	<i>Simbol</i>	<i>Valoarea</i>
Numărul de dinți	$z$	24
Modulul normal	$m_n$	10 mm
Unghiul de înclinare al dinților	$\beta$	$0^\circ$
Unghiul de angrenare normal	$\alpha_n$	$20^\circ$
Coefficientul capului	$h_n$	1
Coefficientul jocului	$c_n$	0,25
Coefficientul deplasării	$x_n$	0
Lățimea roții	$b$	40

**Tabelul 14.** Coordonatele punctelor de pe profilul cremalierii măsurate

<i>Nr. Crt.</i>	<i>X[mm]</i>	<i>Y[mm]</i>	<i>Nr. Crt.</i>	<i>X[mm]</i>	<i>Y[mm]</i>	<i>Nr. Crt.</i>	<i>X[mm]</i>	<i>Y[mm]</i>
<b>1</b>	0.041	0.078	<b>80</b>	35.508	-1.009	<b>160</b>	74.401	-20.949
<b>2</b>	0.325	0.004	<b>81</b>	36.061	-2.479	<b>161</b>	74.89	-22.271
<b>3</b>	0.829	0.010	<b>82</b>	36.61	-3.981	<b>162</b>	75.428	-23.722
<b>4</b>	1.326	0.008	<b>83</b>	37.074	-5.228	<b>163</b>	75.919	-25.034
<b>5</b>	1.825	0.039	<b>84</b>	37.622	-6.717	<b>164</b>	76.149	-25.519
<b>6</b>	2.325	0.035	<b>85</b>	38.087	-7.984	<b>165</b>	76.309	-25.645
<b>7</b>	2.622	0.032	<b>86</b>	38.68	-9.589	<b>166</b>	76.419	-25.68
<b>8</b>	2.901	-0.011	<b>87</b>	39.219	-11.036	<b>167</b>	76.927	-25.698
<b>9</b>	3.200	-0.086	<b>88</b>	39.811	-12.602	<b>168</b>	77.493	-25.691
...	...	...	...	...	...	...	...	...

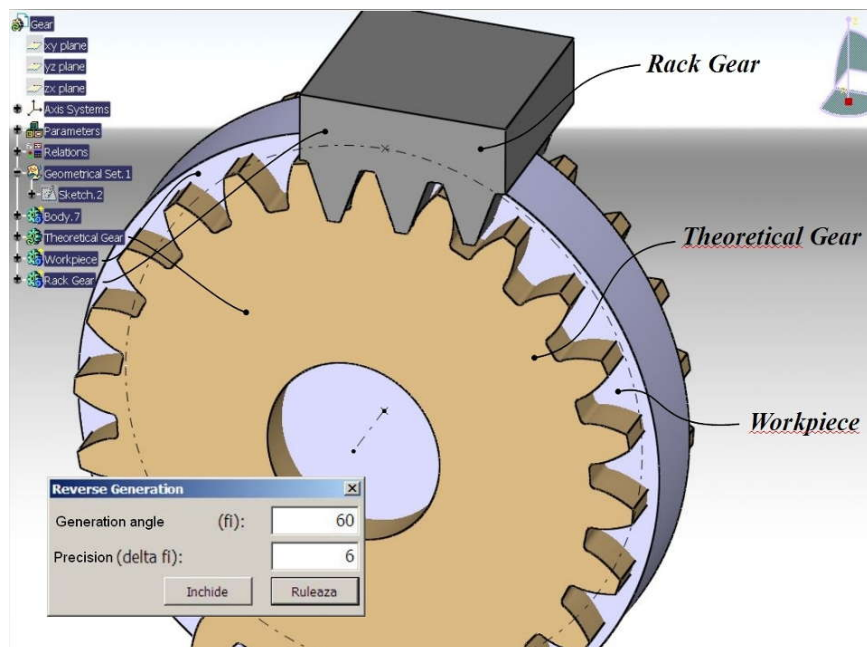
Pe aparatul de măsurat - profil proiector Starrett, s-au măsurat 200 de puncte de pe profilul unei cremalieri uzate care a fost utilizată la prelucrarea unor roți dințate asemenea celei modelate virtual, în mediul de proiectare CATIA.

Coordonatele acestor puncte, vezi tabelul 14, au fost introduse în CATIA cu ajutorul aplicației *MGMC-VBA*, obținându-se profilul efectiv al cremalierii măsurate.



**Fig. 32.** Puncte măsurate pe profilul cremalierii (cuțit pieptene)

Se urmărește modelarea roții generate cu cuțitul pieptene având profilul efectiv prezentat în figura 32. Pentru aceasta, s-au modelat și poziționat corespunzător solidele: roata dințată teoretică; cremaliera efectivă și un semifabricat cilindric, coaxial cu roata dințată teoretică, vezi figura 33.

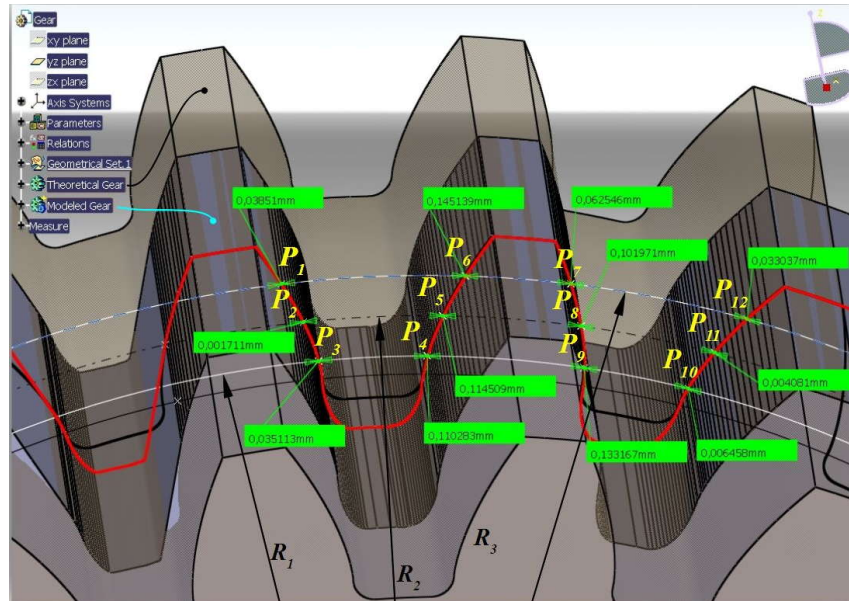


**Fig. 33.** Modelele solide ale roții dințate teoretice, a semifabricatului și a cremalierii măsurate

Rulând aplicația *MGMC-VBA* (*Generarea inversă*), în care s-au introdus parametrii aplicației: unghiul de generare ( $\phi_i=60^\circ$ ) și incrementul unghiului de rotație al semifabricatului ( $\Delta\phi_i=6^\circ$ ), s-a obținut modelul solid al roții dințate prelucrate virtual cu cremaliera măsurată.

Pentru a compara cele doua modele solide ale roților dințate, cea teoretică cu cea modelată, s-au considerat intersecțiile acestora într-un plan transversal, cu trei cercuri concentrice de diametre:  $R_1=230$  mm,  $R_2=240$  mm și  $R_3=250$  mm.

Punctele de intersecție dintre aceste cercuri și profilurile roților dințate sunt prezentate în figura 34.



**Fig. 34.** Roata dințată “prelucrată virtual” și roata dințată teoretică - erori

Distanțele minime dintre perechile de puncte corespunzătoare celor două profiluri ale roților dințate, reprezentând erorile de generare ale roții dințate modelate prelucrată virtual, față de cea teoretică, sunt trecute în tabelul 15.

**Tabelul 15.** Mărimile erorilor dintre cele două roți dințate, teoretică și modelată

Nr. crt.	Erori de modelare [mm]
$P_1 P_1'$	0.039
$P_2 P_2'$	0.002
$P_3 P_3'$	0.035
$P_4 P_4'$	0.110
$P_5 P_5'$	0.115
$P_6 P_6'$	0.145
$P_7 P_7'$	0.063
$P_8 P_8'$	0.102
$P_9 P_9'$	0.133
$P_{10} P_{10}'$	0.006
$P_{11} P_{11}'$	0.004
$P_{12} P_{12}'$	0.033

### Concluzii

Metoda modelării solide 3D a generării cu scula cremalieră, în mediul de proiectare CATIA s-a făcut utilizând aplicații dezvoltate pentru această problemă.

S-au prezentat exemple de aplicare care dovedesc capacitatea metodei de reprezentare topologica a suprafețelor de a soluționa riguros problematica modelării 3D a generării cu o sculă cremalieră.

**Modelarea topologică a generării suprafețelor la prelucrarea cu scula disc**

Problematica modelării generării suprafețelor elicoidale cilindrice și de pas constant, cu scule mărginite de suprafețe periferice primare de revoluție (scula disc, scula cilindro-frontală, sculele inelare) trebuie abordată ca o problemă „inversă” celei referitoare la profilarea unor asemenea scule.

Necesitatea abordării unor asemenea probleme apare atunci când, cunoscută fiind forma sculelor, se pune întrebarea dacă acestea pot genera, cu o anumită rigurozitate, suprafața elicoidală pentru care au fost realizate, în esență, generarea virtuală a suprafețelor elicoidale, modelare care reprezintă o primă modalitate de control a erorii geometrice (principiale) de generare. Această problemă poate, de asemenea, să fie rezolvată apelând la metoda reprezentării topologice, în mediul de proiectare grafică CATIA, a suprafețelor.

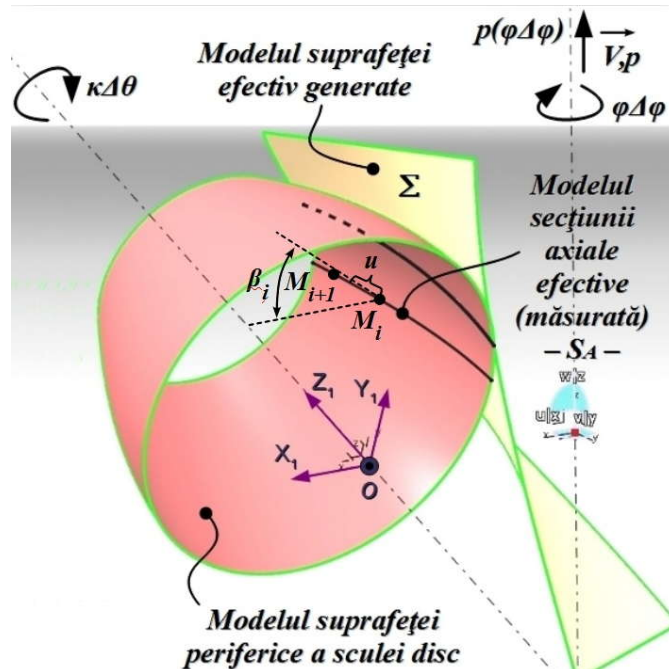
Se acceptă că, este definită, prin măsurare efectivă, o curbă generatoare pe suprafața efectivă a sculei disc, de exemplu, secțiunea axială a acesteia, în forma:

$$S_A = \begin{pmatrix} X_{1_1} & Z_{1_1} \\ X_{1_2} & Z_{1_2} \\ \dots & \dots \\ X_{1_n} & Z_{1_n} \end{pmatrix}, \tag{77}$$

vezi și figura 33, astfel încât segmentul  $M_i M_{i+1}$  să aibă o lungime mică,

$$d_s = \sqrt{(X_{1_i} - X_{1_{i+1}})^2 + (Z_{1_i} - Z_{1_{i+1}})^2} \leq \varepsilon \tag{78}$$

cu  $\varepsilon$  dat ( $\varepsilon = 1 \cdot 10^{-2} \dots 1 \cdot 10^{-1} \text{mm}$ ).



**Fig. 35.** Sisteme de referință și mișcări de generare

Evident, pentru o precizie mai mare, coordonatele punctelor  $M_i, M_{i+1}$  ar trebui să fie mai apropiate.



Prin rotirea curbei  $S_A$ , în jurul axei  $Z_I$ , se generează modelul suprafeței periferice primare efective a sculei disc:

$$\begin{pmatrix} X_1 \\ Y_1 \\ Z_1 \end{pmatrix} = \omega_3^T(k \cdot \Delta\theta) \begin{pmatrix} X_{1_1} & Z_{1_1} \\ X_{1_2} & Z_{1_2} \\ \dots & \dots \\ X_{1_n} & Z_{1_n} \end{pmatrix}, \quad (k=1 \dots m) \quad (79)$$

și  $\Delta\theta$ - increment unghiular de rotație, convenabil ales.

Pentru o suprafață de revoluție elementară, generată de segmentul  $M_i, M_{i+1}$ , vezi figura 35, transformarea (79), are forma:

$$\begin{pmatrix} X_1 \\ Y_1 \\ Z_1 \end{pmatrix} = \omega_3^T(k \cdot \Delta\theta) \begin{pmatrix} X_{1_i} - u \cdot \cos\beta_i \\ Z_{1_i} + u \cdot \sin\beta_i \end{pmatrix}, \quad 0 \leq u \leq ds \quad (k=1 \dots m). \quad (80)$$

Se definește mărimea unghiului  $\beta_i$  prin:

$$\beta_i = \arctg \left[ \frac{|Z_{1_{i+1}} - Z_{1_i}|}{|X_{1_{i+1}} - X_{1_i}|} \right]. \quad (81)$$

În formă dezvoltată, transformarea (80) devine:

$$\begin{pmatrix} X_1 \\ Y_1 \\ Z_1 \end{pmatrix} = \begin{pmatrix} \cos(k \cdot \Delta\theta) & -\sin(k \cdot \Delta\theta) & 0 \\ \sin(k \cdot \Delta\theta) & \cos(k \cdot \Delta\theta) & 0 \\ 0 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} X_{1_i} - u_i \cdot \cos\beta_i \\ 0 \\ Z_{1_i} + u_i \cdot \sin\beta_i \end{pmatrix}; \quad (82)$$

$$(i=1 \dots n); \quad (k=1 \dots m); \quad (0 \leq u_i \leq ds)$$

reprezentând modelul virtual al suprafeței periferice primare efective (măsurate) a sculei disc, în principiu în forma:

$$(S)_{u_i, \Delta\theta} \begin{cases} X_1 = X_{1_i}(u_i, k \cdot \Delta\theta); \\ Y_1 = Y_{1_i}(u_i, k \cdot \Delta\theta); \\ Z_1 = Z_{1_i}(u_i, k \cdot \Delta\theta). \end{cases} \quad (i=1 \dots n); \quad (k=1 \dots m) \quad (83)$$

Forma (83) reprezintă familia de suprafețe conice elementare care „substituie” suprafața de revoluție, periferică primară, a sculei disc.

**Notă:** pentru  $u \rightarrow 0$ , familia de suprafețe conice elementare (83) se reduce la o familie de cercuri având centrele pe axa  $Z_I$ .

### Cinematica generării

Modelul suprafeței efectiv generate se obține imprimând sculei virtuale, mărginită de familia de suprafețe elementare conice (83), ansamblul de mișcări de generare a suprafeței elicoidale cilindrice, de axă  $\vec{V}$  și parametru elicoidal  $p$ .

Poziția axei modelului suprafeței efective a sculei disc ( $A$ ) este definită, vezi și figura 36, de:

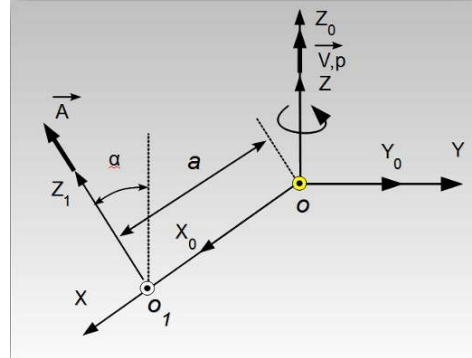
$a$  este distanța între axa modelului suprafeței periferice a sculei disc și axa suprafeței elicoidale, virtual generate (modelul suprafeței efectiv generate);

$\alpha$  – unghiul de înclinare al axei sculei disc în raport cu axa  $\vec{V}$  a suprafeței elicoidale

$$\alpha = \operatorname{arctg} \left[ \frac{p}{R_e} \right] \quad (84)$$

cu  $R_e$  – raza exterioară a suprafeței elicoidale (a semifabricatului).

Se definește un sistem mobil, solidar cu  $X_1Y_1Z_1$ , având axele paralele cu ale sistemului  $XYZ$  și, totodată, originea în punctul  $O$  – sistemul  $X_0Y_0Z_0$ , vezi figura 36.



**Fig. 36.** Sisteme de referință și mișcări de generare

Transformarea de coordonate între sistemele  $X_1Y_1Z_1$  și  $X_0Y_0Z_0$ , este:

$$\begin{pmatrix} X_0 \\ Y_0 \\ Z_0 \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos\alpha & -\sin\alpha \\ 0 & \sin\alpha & \cos\alpha \end{pmatrix} \cdot \left[ \begin{pmatrix} X_1 \\ Y_1 \\ Z_1 \end{pmatrix} - \begin{pmatrix} a \\ 0 \\ 0 \end{pmatrix} \right]. \quad (85)$$

Astfel, suprafața  $S$  (83) este exprimată în sistemul  $X_0Y_0Z_0$ , principal, în forma:

$$S_{X_0Y_0Z_0} \begin{cases} X_0 = X_0(u_i; k \cdot \Delta\theta); \\ Y_0 = Y_0(u_i; k \cdot \Delta\theta); \\ Z_0 = Z_0(u_i; k \cdot \Delta\theta). \end{cases} \quad (86)$$

Suprafața periferică primară a modelului virtual al sculei disc, în mișcarea elicoidală de axă  $\vec{V}$  și parametru elicoidal  $p$  (axa și parametrul suprafeței elicoidale țintă),

$$\begin{pmatrix} X \\ Y \\ Z \end{pmatrix} = \omega_3^T(q \cdot \Delta\varphi) \begin{pmatrix} X_0 \\ Y_0 \\ Z_0 \end{pmatrix} + \begin{pmatrix} 0 \\ 0 \\ p \cdot (q \cdot \Delta\varphi) \end{pmatrix}, \quad (q = 1 \dots t) \quad (87)$$

și  $\Delta\varphi$  increment unghiular de rotație în jurul axei  $Z$ , generează familia de suprafețe periferice primare ale sculei disc, în raport cu sistemul de referință al suprafeței elicoidale virtual generate:

$$\left( S_{u_i, \Delta\theta} \right)_{\Delta\varphi} \begin{cases} X = X(u_i, k \cdot \Delta\theta, q \cdot \Delta\varphi); \\ Y = Y(u_i, k \cdot \Delta\theta, q \cdot \Delta\varphi); \\ Z = Z(u_i, k \cdot \Delta\theta, q \cdot \Delta\varphi). \end{cases} \quad (88)$$

Înfășurătoarea familiei de suprafețe (88) reprezintă modelul suprafeței elicoidale efectiv generate.

#### Condiția de înfășurare

În forma GOHMAN, condiția de înfășurare poate fi scrisă ca fiind:

$$\vec{N}_s \cdot \vec{R}_{\Delta\varphi} = 0; \quad (89)$$

în care:  $\vec{N}_s$  este normala la suprafața conică elementară (83),

$$\vec{N}_S = \begin{vmatrix} \vec{i} & \vec{j} & \vec{k} \\ (\dot{X}_1)_{ui} & (\dot{Y}_1)_{ui} & (\dot{Z}_1)_{ui} \\ (\dot{X}_1)_{\Delta\varphi} & (\dot{Y}_1)_{\Delta\varphi} & (\dot{Z}_1)_{\Delta\varphi} \end{vmatrix}; \quad (90)$$

sau, principial,

$$\vec{N}_S = N_{X_1} \cdot \vec{i} + N_{Y_1} \cdot \vec{j} + N_{Z_1} \cdot \vec{k}; \quad (91)$$

$\vec{R}_{\Delta\varphi}$  – vectorul viteză, în mișcarea punctului curent de pe suprafața conică elementară, în raport cu sistemul de referință al suprafeței elicoidale generate, din (82),

$$R_{\Delta\varphi} = \frac{dX}{d(\Delta\varphi)} = \omega. \quad (92)$$

sau, în formă vectorială,

$$\vec{R}_{\Delta\varphi} = R_X \cdot \vec{i} + R_Y \cdot \vec{j} + R_Z \cdot \vec{k}; \quad (93)$$

Condiția de înfășurare (85), impune ca cei doi vectori, formele (86) și (87) să fie definiți în același sistem de referință.

Astfel, normala  $\vec{N}_S$  este referită la sistemul  $XYZ$  prin transformarea (78) (sistemele  $X_0Y_0Z_0$  și  $XYZ$  au axele paralele și orientate în același sens),

$$\begin{pmatrix} N_X \\ N_Y \\ N_Z \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos \alpha & \sin \alpha \\ 0 & -\sin \alpha & \cos \alpha \end{pmatrix} \cdot \begin{bmatrix} \begin{pmatrix} N_{X_1} \\ N_{Y_1} \\ N_{Z_1} \end{pmatrix} \\ - \begin{pmatrix} a \\ 0 \\ 0 \end{pmatrix} \end{bmatrix}. \quad (94)$$

sau, principial,

$$\vec{N}_{S_{XYZ}} = N_X \cdot \vec{i} + N_Y \cdot \vec{j} + N_Z \cdot \vec{k}; \quad (95)$$

Condiția (89) poate fi reformulată, acum, ținând seama de (93) și (95), încât, principial, poate fi examinată ca o funcție,

$$Q_{(u_i, \Delta\theta, \Delta\varphi)} = 0; \quad (96)$$

Ansamblul de ecuații (88) și (96) reprezintă modelul numeric virtual al suprafeței elicoidale generate:

$$C_{XYZ} \begin{cases} \left( S_{u_i, \Delta\theta} \right)_{\Delta\varphi} \cdot \begin{cases} X = X(u_i, k \cdot \Delta\theta, q \cdot \Delta\varphi); & (i = 1 \dots n) \\ Y = Y(u_i, k \cdot \Delta\theta, q \cdot \Delta\varphi); & (k = 1 \dots m) \\ Z = Z(u_i, k \cdot \Delta\theta, q \cdot \Delta\varphi). & (q = 1 \dots t) \end{cases} \\ Q_{(u_i, \Delta\theta, \Delta\varphi)} = 0. \end{cases} \quad (97)$$

Forma (97) se transpune numeric printr-un masiv de puncte din care poate fi extras un loc geometric reprezentând, de exemplu, o secțiune plană transversală a suprafeței elicoidale, ca formă de comparație cu suprafața țintă de prelucrat.

### **Metoda de reprezentare topologica în mediul grafic de proiectare CATIA pentru prelucrarea virtuală a canalului elicoidal al unui alezor**

În urma rulării aplicației *MGMCMC-VBA* specifice, se simulează procesul de prelucrare a semifabricatului, prin descompunerea mișcării elicoidale în două mișcări: rotația semifabricatului și translația sculei disc. Mișcarea de rotație a semifabricatului are loc cu un anumit increment unghiular *delta fi*. Mărimea acestui increment unghiular determină precizia suprafeței modelate. Mișcarea de translație a sculei disc, se face cu un increment de deplasare în direcția axei

semifabricatului, proporțional cu  $p \cdot \varphi$  ( $p$  – parametru elicoidal,  $p = \frac{R_e}{\text{tg } \beta}$ );  $R_e$  – raza exterioara a semifabricatului;  $\beta$  – unghiul de înclinare al elicei pe cilindrul de rază  $R_e$ , vezi figura 32.

Pentru exemplificare, se consideră canalul elicoidal al alezorului din figura 37, reprezentând modelul teoretic ce se dorește a fi prelucrat.

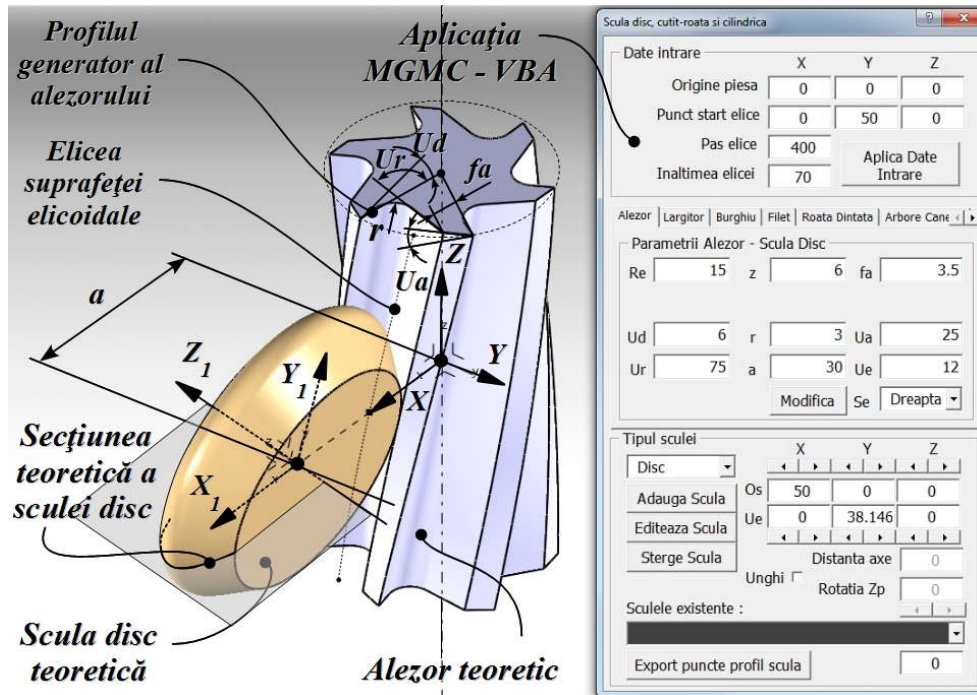


Fig. 37. Alezorul teoretic și scula disc teoretică rezultată cu aplicația MGMC - VBA

Profilul generator al canalului alezorului este prezentat în figura 38. S-a determinat profilul secțiunii transversale a sculei disc, rulând aplicația MGMC-VBA, în care s-au introdus valorile parametrilor de intrare reprezentând profilul generator al canalului alezorului și parametrii elicoidali ai suprafeței elicoidale, tabelul 16.

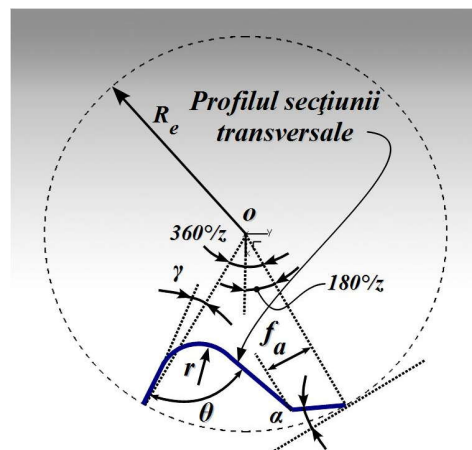
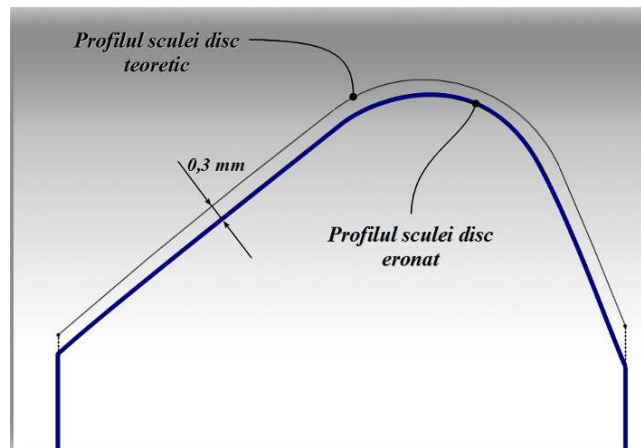


Fig. 38. Profilul secțiunii transversale a dintelui alezorului. (profil compus din arce de cerc și segmente de dreaptă)

**Tablul 16. Parametrii de intrare a profilului secțiunii transversale a canalului alezorului**

Simbol	Descriere	Valoare	UM
$R_e$	Raza exterioară al alezorului	15	[mm]
$z$	Numărul de dinți, $z$	6	-
$U_a$	Unghi de așezare $\alpha$	25	[°]
$U_d$	Unghi de degajare $\gamma$	6	[°]
$f_a$	Lățimea feței de așezare $f_a$	3,5	[mm]
$U_r$	Unghiul golului, $\theta$	75	[°]
$r$	Raza de racordare, $r$	3	[mm]
$U_e$	Unghi înclinare a elicei $\beta$	12	[°]
$S_e$	Sensul elicei canalului	dreapta	-
$a$	Distanța, pe direcția axei X, dintre axa sculei disc și axa alezorului de generat	30	[mm]

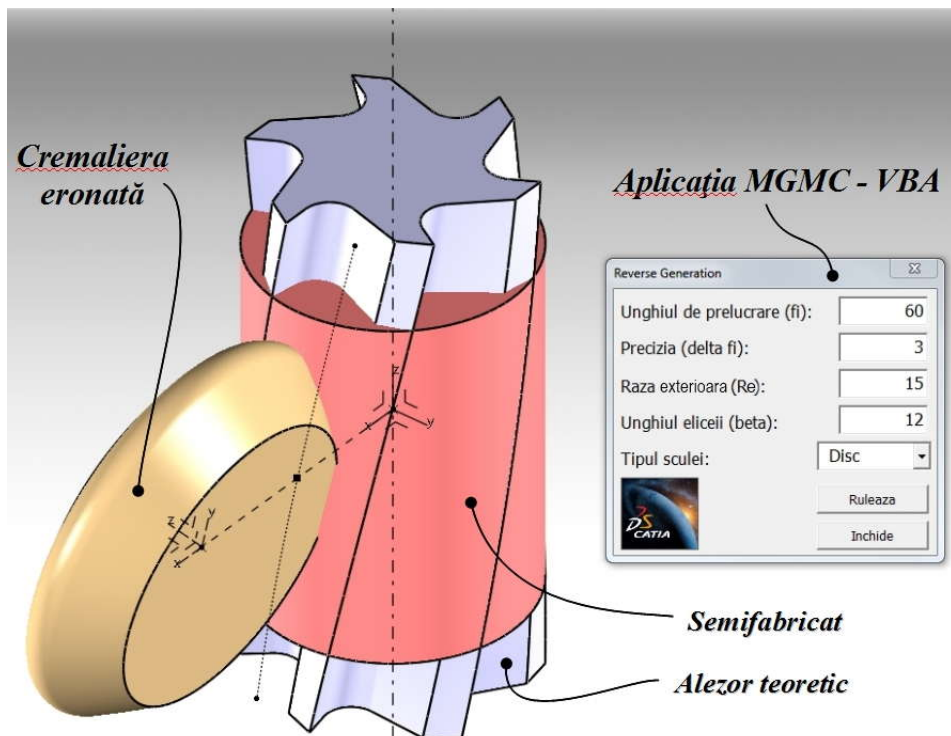
Pentru exemplificare se consideră modelul secțiunii axiale a sculei disc profil reprezentat în figura 39.



**Fig. 39. Secțiunea axială a sculei disc eronată (efectivă) pentru generarea canalului elicoidal**

Se dorește a se afla profilul secțiunii transversale virtuale a canalului ce va rezulta, în urma simulării procesului de prelucrare cu aceasta sculă disc eronată.

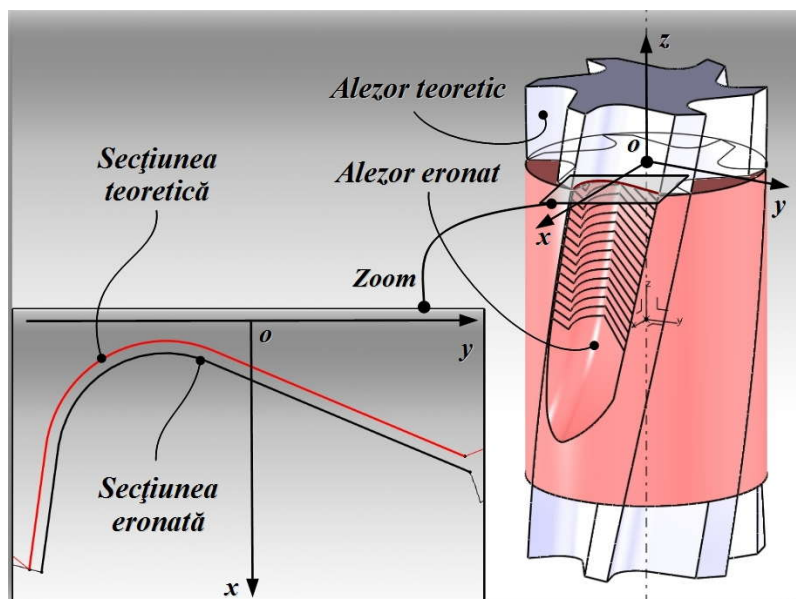
Semifabricatul, reprezentat de un solid de tip cilindru, modelul solid al alezorului teoretic și modelul solid al sculei disc eronată (efectivă) sunt elementele de intrare în algoritmul aplicației *MGMC – VBA*, în fereastra căreia se pot introduce parametrii de rulare (unghiul de prelucrare, incrementul unghiular care dă acuratețea suprafeței periferice rezultate, raza de rulare și unghiul elicei), figura 40.



**Fig. 40.** Generarea inversă a canalului alezorului, aplicația MGMC - VBA

Rulând aplicația MGMC - VBA, care automatizează comenzile de copiere și inserare a solidelor, rotite, și, respectiv, translate, prin extragerea solidului reprezentând scula, din semifabricatul virtual, rezultă, în final, modelul solid al alezorului modelat, putând fi comparat cu modelul solid teoretic al acestuia.

În figura 41, sunt prezentate cele două modele 3D, precum și secțiunile transversale ale acestora, rezultate în urma secționării cu un plan paralel cu XY, în sistemul de referință aferent.



**Fig. 41.** Arborele hexagonal teoretic și eronat

În tabelul 17, sunt prezentate coordonatele unor puncte de pe aceste profiluri.

**Tabelul 17.** *Coordonatele punctelor de pe profilul teoretic și cel modelat*

<i>Profil teoretic</i>	<i>Nr. Crt.</i>	<i>X[mm]</i>	<i>Y[mm]</i>	<i>Profil eronat</i>	<i>Nr. Crt.</i>	<i>X[mm]</i>	<i>Y[mm]</i>
	<i>1</i>	14.5694	-3.5683		<i>1</i>	14.6407	-3.2644
	<i>2</i>	12.8780	-3.3378		<i>2</i>	12.9778	-3.0403
	<i>3</i>	11.1894	-3.0895		<i>3</i>	11.3165	-2.8068
	<i>4</i>	9.7360	-2.2390		<i>4</i>	9.8949	-1.9646
	<i>5</i>	8.9698	-0.7394		<i>5</i>	9.2321	-0.4529
	<i>6</i>	9.1323	0.9370		<i>6</i>	9.5591	1.1707
	<i>7</i>	9.7924	2.5114		<i>7</i>	10.2122	2.7163
	<i>8</i>	10.4529	4.0855		<i>8</i>	10.8560	4.2657
	<i>9</i>	11.1134	5.6596		<i>9</i>	11.5082	5.8116
	<i>10</i>	11.7739	7.2338		<i>10</i>	12.1590	7.3582

### **Concluzii**

Prin problematica dezbătută anterior s-a dovedit faptul că, metodologia generală de reprezentare a structurilor topologice, prin facilitățile de reprezentare în mediul de proiectare grafică CATIA constituie un mijloc eficace de determinare a erorilor de generare a suprafețelor și, totodată de abordare multiplelor probleme, care apar în analiza proceselor de generare a suprafețelor prin înfășurare.

### **5.3. Teoria geometriei topologice a construcțiilor mecanice**

Obiectivul acestei cercetări este de a dezvolta un sistem de conducere asistată cu funcții de identificare a structurilor topologice și de avertizare a coliziunii.

A fost imaginat un sistem, care permite identificarea unor anumite tipuri de structuri topologice și a fost dezvoltată o aplicație care să permită pe baza acestei identificări avertizarea personalului uman privind iminența coliziunilor.

Sistemul este configurat cu o camera CCD pentru a obține imagini ale mediului, împreună cu un calculator de procesare a imaginii. Imaginea de intrare capturată de o camera este folosită pentru a recunoaște obstacolele posibile și anticipează traiectoria posibilă a coliziunii între aceste obstacole și elementul topologic ce este urmărit.

Sistemul dezvoltat este capabil de a emite în timp real avertismente auditive și vizuale atunci când elementul pe care instalat sistemul se află în situație de coliziune iminentă cu obstacole reprezentate de alte structuri topologice existente în mediul înconjurător, în scopul de a preveni un posibil pericol.

Una dintre situațiile în care poate fi utilizat acest sistem se referă la avertizarea operatorului privind iminența coliziunii între palpatorul unei mașini de măsurat în coordonate și elemente ale piesei de măsurat sau elemente ale structurilor de fixare a piesei.

Metoda descrisă este o extindere a cercetărilor anterioare, efectuate de membrii echipei de cercetare, în privința evitării coliziunilor. În cazul măsurătorilor cu echipamente 3D, există riscul coliziunii palpatorului cu diverse obstacole care conduc la ruperea sau decalibrarea componentelor echipamentului (figura 42). A fost propusă o metodă bazată pe vederea artificială, care să permită avertizarea utilizatorului în caz de coliziune iminentă.



**Fig. 42.** Posibilitatea de coliziune între elementele de palpate și piesa măsurată

Aplicația software dezvoltată servește ca sistem de supraveghere, care avertizează utilizatorul în cazul în care elementul palpator este în coliziune cu orice obstacol iminent, inclusiv partea care urmează să fie măsurată.

### **Sistem video de prevenire a coliziunilor bazat pe abordarea topologică**

O altă aplicație practică a acestei metode se referă la detectarea bazată pe previziune a coliziunii unui vehicul, pentru asistarea conducătorului. Există mai multe motive pentru punerea în aplicare a viziunii bazate pe calculator în asistența șoferului: în primul rând, pierderile enorme, atât în vieți omenești cât și financiare provocate de accidentele autovehiculelor iar, în al doilea rând, disponibilitatea algoritmilor de încredere în vederea artificială.

A fost imaginat și realizat un sistem care permite identificarea unor anumite tipuri de structuri topologice, reprezentate de mașinile și pietonii prezenți în trafic și a fost dezvoltată o aplicație, care să permită pe baza acestei identificări avertizarea șoferului privind iminența coliziunilor.

Imaginea de intrare capturată de o camera este folosită pentru a recunoaște obstacolele posibile, reprezentate de mașini și pietoni, anticipând traiectoria posibilă a coliziunii. Sistemul dezvoltat este capabil de a emite în timp real avertismentelor auditive și vizuale atunci când automobilul pe care este instalat sistemul este în situație de coliziune iminentă cu alte autovehicule sau pietoni, în scopul de a preveni un posibil pericol. Metoda propusă în această lucrare este reprezentată de detectarea obstacolelor cu ajutorul unei abordări topologice.

Sistemul de evitare/avertizare a coliziunii frontale (SECF) se referă la sistemul de alertă precoce /intervenție timpurie în cazul unui accident indiferent de tip, față-spate, cu un vehicul.

SECF monitorizează vehiculul gazdă și vehiculul din față și avertizează șoferul cu privire la iminența unei coliziuni. Această tehnologie se bazează pe utilizarea senzorilor iar sistemul de vedere artificială poate prezice viteza vehiculului monitorizat și distanța dintre două vehicule. Când distanța este prea mică, șoferul este avertizat, în scopul prevenției accidentului. Sistemul oferă conducătorului alertă sonoră, afișaj vizual sau alte alerte.

În SUA, procentul coliziunilor din spate, dintre toate coliziunile a fost de aproximativ 31,5% în 2009. Statisticile arată că, aproximativ 80 la sută dintre șoferi nu au efectuat nicio acțiune la impact.

În Uniunea Europeană, aproximativ 10.000 de pietoni și bicicliști sunt uciși și aproximativ 500.000 răniți. Jumătate din accidente au avut loc la un impact mai mic de 48 km/h. În Japonia, aproximativ 4000 de pietoni și bicicliști sunt uciși și aproximativ 30.000 răniți în fiecare an. Deși viteza la care se produc aceste accidente este foarte scăzută, atât pierderile umane și financiare sunt importante. Prin urmare, un sistem pentru a preveni sau diminua unui accident este o necesitate pentru industria auto.

În prezent, diferite companii au implementat cu succes sisteme de avertizare a coliziunii și includ sau sunt bazate exclusiv pe tehnologii de vedere artificiale (figura 43).

Sistemul Pre-Collision al firmei Toyota (SPC) este primul din producția în seria unui sistem de avertizare și este un sistem bazat pe radar.

În 2004, Toyota a implementat un sistem prin adăugarea la radar a unei singure camere digitale pentru a îmbunătăți precizia de detectare a coliziunilor, care a fost disponibilă pe Crown Majesta.



Front Assist de pe 2011 Volkswagen Touareg este capabil sa actioneze frânarea pentru a opri în cazul unui accident iminent și, de asemenea, sistemul tensionează centurile de siguranță. De asemenea, Audi are astfel de sistem pe modelele de top (A8).

Disponibil din 2010, în Volvo S60 și V60 Fig.2 b, c, sistemul de avertizare a coliziunii pietonilor, servește la avertizarea conducătorului auto cu privire la potențialele coliziuni cu vehiculele și pietonii.



**Fig. 43.** Sisteme de avertizare a coliziunii bazate pe vedere artificială

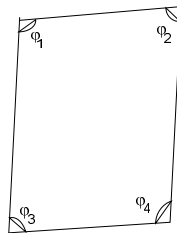
În prezent există mai multe abordări în ceea ce privește detectarea pietonilor, cum ar fi tehnica de suport vector-mașina (SVM), reconstrucția spațiului 3D, și alte abordări. Nu în ultimul rând, algoritmul de curgere (optical flow) este folosit pentru detectarea coliziunilor.

**Originalitatea sistemului imaginat de colectivul de cercetare al acestui proiect constă în identificarea obstacolelor pe baza reconstruirii mediului înconjurător sub formă de structuri topologice complexe.**

#### Aplicarea metodei

Se propune un cadru complet pentru identificarea obstacolelor pe baza structurii topologice, care o reprezintă obiectele. Metoda de identificare se bazează pe structura invariantă a fiecărei clase de caracteristici a formei, în evoluția ei topologică de mișcare.

Identificarea obiectelor în metoda imaginată se bazează pe ideea că obiectele sunt entități topologice compuse din forme geometrice simple în relații de locație între anumite forme geometrice de bază. Formele sunt definite ca și cvasi-forme geometrice (figura 44). De exemplu, dreptunghiurile sunt identificate ca forme alcătuite din 4 linii conectate, unghiul dintre fiecare doua este de aproximativ 90 de procente (cu o toleranță de 10%) și suma tuturor unghiurilor ar trebui să fie 360 de grade.

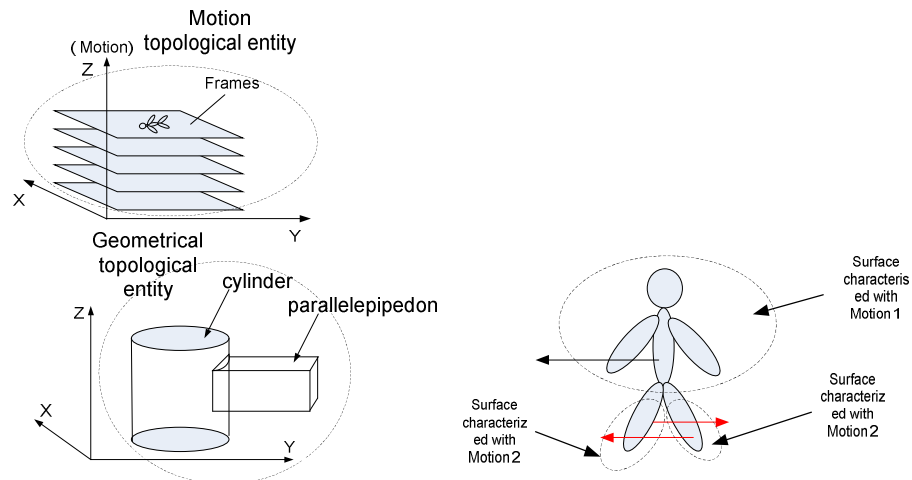


**Fig. 44.** Reprezentarea unui cvasi-dreptunghi

Fiecare caracteristică geometrică a formei este descrisă în tabelele 18, 19 și 20. Fiecare obstacol, mașină și pieton sunt descrise după cum se arată în fig. 45.

Mașinile, descrise ca fiind o reprezentare 3D generică, și pietonii, descriși ca structură geometrică topologică, sunt, în plus, descrise și ca structură de mișcare topologică figura 45. În

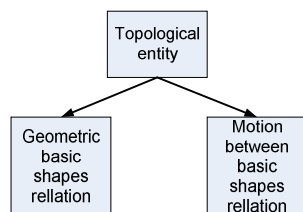
cea ce privește mișcarea, picioarele sunt într-o relație de mișcare, care definește structura entității fig. 45.



**Fig. 45.** Similitudini între geometria topologică și structura de mișcare topologică

În abordarea noastră, entitatea este definită prin două perspective: una cu privire la relațiile formelor geometrice de bază și a doua referitoare la mișcările dintre formele geometrice. Mișcarea este privită ca o modalitate de interacțiune și inter-relație între elementele entității.

În figura 46, este prezentată schematic similitudinea dintre structura topologică geometrică și topologia de mișcare.



**Fig. 46.** Entitate topologică conform abordării propuse: pe baza geometriei și a mișcării

Cadrele succesive sunt privite ca a treia dimensiune și mișcarea definește elementele structurii topologice. Dacă cilindrul și paralelipedul sunt elemente ale entității geometrice topologice, în cazul entității topologice în mișcare, pozițiile succesive ale pietonilor sunt elemente ale entității topologice. De exemplu, cilindrul și paralelipedul se află într-o relație de poziționare și sunt caracterizate prin parametrii lor geometrici, poziția succesivă a "pietonilor" se află în anumite relații pentru fiecare cadru succesiv.

**Tabelul 18.** Descrierea topologică a zonei frontale a mașinii

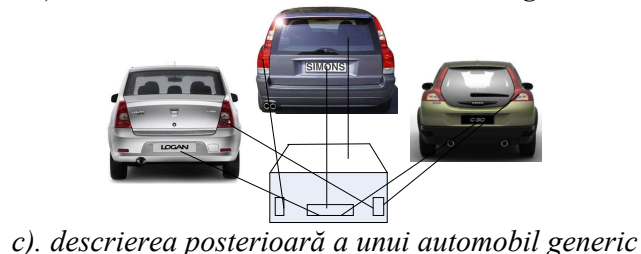
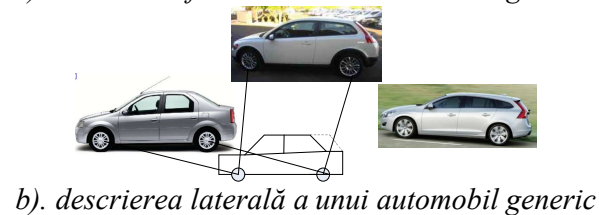
	Parbriz	Placuțe de înmatriculare	Faruri
Suprafață	30%-60% din întregul contur	5-15% din întregul contur	5-20% din întregul contur
Forma geometrică	cvasi-dreptunghi	dreptunghi	nedefinită
Poziție	deasupra plăcuțelor de înmatriculare, în jumătatea superioară a conturului, deasupra farurilor	în centrul conturului	la extremitățile stângă și dreaptă a conturului
culoare	nedefinită	alb și negru	alb și / sau galben

**Tabelul 19.** Descrierea topologică a zonei laterale a mașinii

	Geamuri	Roți
Suprafață	30%-60% din întregul contur	5-15% din întregul contur
Forma geometrică	cvasi-dreptunghi	cvasi-circular
Poziție	deasupra roților, în jumătatea superioară a conturului	la extremitatea inferioară a conturului
culoare	nedefinită	nedefinită

**Tabelul 20.** Descrierea topologică a zonei posterioare a mașinii

	Lunetă	Placuțe de înmatriculare	Semnalizatoare spate
Suprafață	30%-60% din întregul contur	5-15% din întregul contur	5-20% din întregul contur
Forma geometrică	cvasi-dreptunghi	dreptunghi	nedefinită
Poziție	deasupra plăcuțelor de înmatriculare, în jumătatea superioară a conturului,	în centrul conturului	la extremitățile stângă și dreaptă a conturului
culoare	nedefinită	alb și negru	roșu



**Fig. 47.** Descrierea unui obstacol (automobil) ca suma de trei entități topologice: zona frontală, laterală și posterioară. Fiecare entitate conține diferite forme geometrice

### Structura sistemului.

Dispozitivul de detectare este o camera web montată între parbriz și oglinda retrovizoare din interiorul mașinii 48. Camera video obține imagini ale drumului din față a vehiculului și se transferă imaginile achiziționate la un calculator pentru prelucrare, analiză a imaginilor și obiectelor și identificarea obiectelor din față vehiculului.



**Fig. 48.** Vehiculul de testare

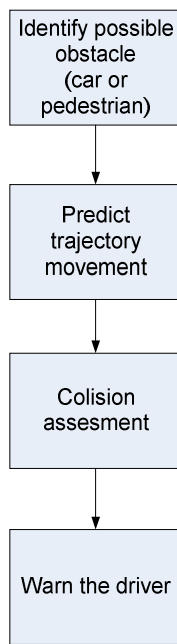
Softul a fost dezvoltat în Visual Studio 2008 IDE folosind limbajul C și biblioteca OpenCV. Camera folosită este Logitech HD Webcam C910 și rezoluția de lucru de 800×600 pixeli. Algoritmul procesează un cadru în aproximativ 200 ms pe un procesor de tip Celeron 1.6 Pentium IV. Un Laptop Toshiba Satellite L30-134 (memorie de 1 GB) a fost utilizat cu scopul de a testa metoda comparativ cu un calculator cu resurse relativ scăzute, arhitectura potrivită pentru un computer de bord pentru masină.

Figura 49 oferă o privire de ansamblu a algoritmului de extracție caracteristic și de identificare. În cazul aplicației noastre, o nouă imagine trebuie să fie identificată cu cea descrisă.

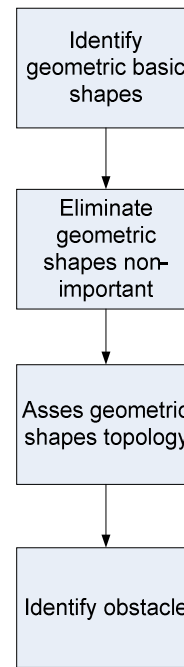
Algoritmul pentru detectarea obstacolelor din fața vehiculului este reprezentat de mai multe etape, așa cum se descrie în figura 50. Obstacolele posibile sunt identificate cu metoda descrisă în prezenta lucrare, bazată pe o abordare topologică iar traiectoria este precisă pentru fiecare obstacol posibil. Ulterior, în cazul în care distanța până la obstacol devine mai mică și există traiectorii de coliziune ale vehiculului și obstacol atunci este emis avertizarea.

Autorii au urmărit obținerea unei viteze de execuție crescute a algoritmului, în cazul în care localizarea are loc, în scopul de a permite nu numai procesarea imaginii în timp real pentru detectarea coliziunilor, ci și pentru a avea suficiente resurse de calcul rămase pentru a adăuga mai târziu noi caracteristici, cum ar fi recunoașterea semnelor de circulație sau de recunoaștere a ieșirii din banda de mers, sisteme care folosesc tehnici de vedere artificială.

Algoritmul bazat pe abordarea topologică este descris în fig. 50. În prima etapă sunt identificate formele geometrice de bază (cvasi-cerc și cvasi-dreptunghi). În a doua etapă, formele care sunt prea mici (fie sunt prea departe sau sunt obiecte statice neimportante) sunt eliminate. Topologia este estimată cu regulile descrise în tabelele 18, 19, 20. În cazul în care entitățile topologice sunt găsite, atunci, obiectele sunt urmărite cadru cu cadru pentru a le evalua traiectoria.

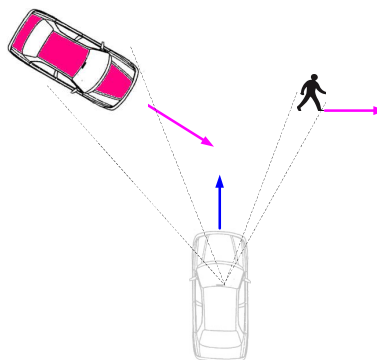


**Fig. 48.** Algoritm de extracție caracteristic

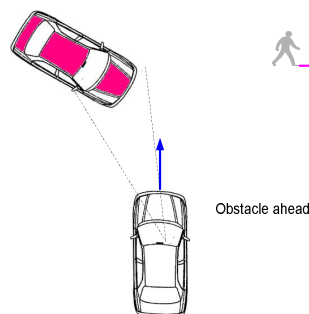


**Fig. 49.** Algoritm de detectare a obstacolelor

Autorii au prezentate mai multe scenarii, fiind descrise mai jos (figura 50).



a). identificarea posibilelor obstacole și prezicerea traiectoriei

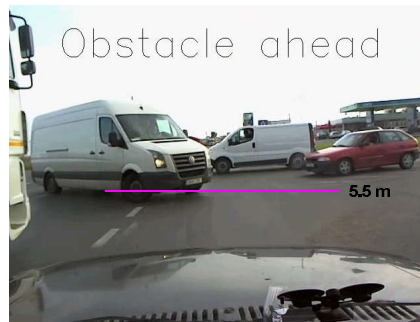


b). traiectoria de coliziune prezisă este analizată și, în cazul unei coliziuni iminente, este emisă avertizarea

**Fig. 50.** Scenariu pentru detecția coliziunii

S-a testat vehiculul cu acest sistem rulând la viteze de până la 40 km/h. S-a testat sistemul și în trafic, iar care conducătorul auto a fost avertizat dacă distanța față de obstacolul din față a fost sub 10 metri. De asemenea, s-a testat automobilul la coliziuni din spate cu 0 accelerație, coliziuni din spate și față cu o mașină. Coliziunea iminentă cu pietoni a fost testată la o viteză de

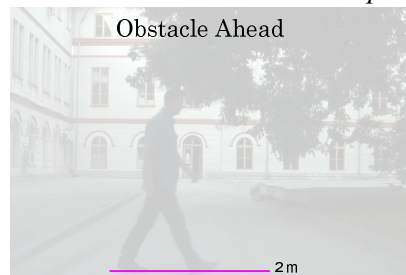
5 km/h. Testele au arătat rezultate bune, sistemul reușind să avertizeze cu succes conducătorul auto.



**Fig. 51.** Cazul în care vehiculul cu sistemul dezvoltat se apropie de o coliziune laterala cu un automobil



**Fig. 52.** Cazul în care vehiculul cu sistemul dezvoltat se apropie de un automobil staționat



**Fig. 53.** Cazul în care vehiculul cu sistemul dezvoltat se apropie de un pieton

Cu toate acestea, în cazul unui drum accidentat și a unei atmosfere cu fum sau ploaie, sistemul dezvoltat nu reușește să detecteze obstacolele.

Un alt dezavantaj al sistemului este reprezentat de faptul ca nu este capabil să recunoască obstacole, altele decât vehiculelor și pietonii. Diferite aspecte trebuie să fie luate în considerare pentru dezvoltari ulterioare. De exemplu modul în care un astfel de sistem va trata cazurile în care obstacolele aparute în trafic sunt obiecte cum fi copaci sau pietre căzute.

Aceasta cercetare a dezvoltat cu succes un sistem integrat de vedere artificiala bazat pe platforma unui uni PC. Bazat pe o singura camera CMOS montata pe parbriz, sistemul poate recunoaște vehiculului precedent, traiectoria vehiculului și oferă funcții de avertizare a coliziunii. Modelul de estimare a distanță permite evaluarea condițiilor necesare pentru a evita o coliziune și oferă o timp necesar pentru frânarea vehiculului de către conducătorul auto. Dezvoltări viitoare vor include reconstrucția 3D a scenei din fața vehiculului. Pentru a evita probleme cum ar fi vederea în noapte, în condiții de ploaie sau ninsoare, o soluție este folosirea unei camere de luat vederi cu termo-viziune.

## ANEXE

### Lucrări publicate în cadrul contractului ID-791/2008

- [1]. Berbinschi S., Teodor V., Oancea N., 3D Graphical Method for Profiling Tools that Generate Helical Surfaces, în *International Journal of Advanced Manufacturing Technology*, 2011, DOI 10.1007/s00170-011-3637-3, ISSN: 0268-3768 (format tipărit), ISSN: 1433-3015 (format electronic), publicată on-line 16.09.2011;
- [2]. Berbinschi S., Teodor V., Oancea N., A study on helical surface generated by the primary peripheral surfaces of ring tool, în *International Journal of Advanced Manufacturing Technology*, 2011, DOI 10.1007/s00170-011-3687-6, ISSN: 0268-3768 (format tipărit), ISSN: 1433-3015 (format electronic), publicată on-line 11.10.2011;
- [3]. Berbinschi, S., Teodor V., Baroiu, N., Nicolae Oancea, N., Ring Tangential Tool Topological Representation of Peripheral Surfaces, în *Analele Universității „Dunărea de Jos” din Galați*, vol. 1, fascicula V, ISSN 1221-4566, 2011, pag. 11-19;
- [4]. Berbinschi, S., Teodor V., Nicolae Oancea, A 3D Method for Profiling the Shaping Tool for Generation of the Helical Surfaces, în *Analele Universității „Dunărea de Jos” din Galați*, vol. 1, fascicula V, ISSN 1221-4566, 2011, pag. 21-27;
- [5]. Berbinschi, S., Teodor V., Nicolae Oancea, A Graphical Method Developed in the CATIA Design Environment for the Modeling of Generation by Enveloping, în *Analele Universității „Dunărea de Jos” din Galați*, vol. 1, fascicula V, ISSN 1221-4566, 2011, pag. 29-35;
- [6]. Marin, F. B., Teodor, V., Constantin, I., Epureanu Al., Video Collision Warning System Based on Topological Approach, în *Analele Universității „Dunărea de Jos” din Galați*, vol. 1, fascicula V, ISSN 1221-4566, 2011, pag. 37-43;
- [7]. Popa, C., Teodor V., Popa, I., Oancea, N., Surface Profiling Method of the Disk Cutter of the Female Rotor From the Screw Compressor Component, în *Analele Universității „Dunărea de Jos” din Galați*, vol. 1, fascicula V, ISSN 1221-4566, 2011, pag. 45-52;
- [8]. Popa, C., Teodor V., Popa, I., Oancea, N., Surface Profiling Method of the Disk Cutter of the Male Rotor From the Screw Compressor Component, în *Analele Universității „Dunărea de Jos” din Galați*, vol. 1, fascicula V, ISSN 1221-4566, 2011, pag. 53-61;
- [9]. Baroiu, N., Teodor V., Dumitrașcu, N., Oancea, N., Comparative Analytical Models for Sharpening of Multi-Flute Drills with Curved Cutting Edges, în *Analele Universității „Dunărea de Jos” din Galați*, vol. 1, fascicula V, ISSN 1221-4566, 2011, pag. 93-99;
- [10]. Berbinschi, S., Baroiu, N., Teodor, V., 3D Method for Determination of Discontinuities on Tools which Generated by Enwrapping, în *Buletinul Institutului Politehnic din Iași*, tomul LVII (LXI), fascicula 5, ISSN 1011-2855, 2011, pag. 31-38;
- [11]. Berbinschi, S., Teodor, V., Baroiu, N., Oancea, N., The Topological Representation of the Primary Peripheral Surfaces of Ring Tool which Generate a Helical Surface, în *Buletinul Institutului Politehnic din Iași*, tomul LVII (LXI), fascicula 4, ISSN 1011-2855, 2011, pag. 1-8;
- [12]. Berbinschi, S., Teodor, V., Oancea, N., A 3D Method for Profiling the Shaping Tool for Generation of the Helical Surfaces, în *Proceedings of the 5<sup>th</sup> International Conference on Manufacturing Science and Education MSE*, volumul I, ISSN 1843-2522, 2-5 Iunie 2011, Sibiu, pag. 281-284;
- [13]. Berbinschi, S., Teodor, V., Oancea, N., 3D Method, in CATIA Design Environment, for End Mill Tool's Profiling, în *Proceedings of The 15<sup>th</sup> International Conference Modern Technologies, Quality and Inovation, ModTech*, volumul I, ISSN 2069-6736, 25-27 Mai 2011, Vadul lui Vodă, Republica Moldova, pag. 81-84;
- [14]. Berbinschi, S., Baroiu, N., Teodor, V., Oancea, N., A Profiling Method for Secondary Order Tool for Drill Manufacturing, în *Proceedings of the International Conference NEWTECH on Advanced Manufacturing Engineering*, ISBN 978-80-214-4267-2, 14-15 Septembrie 2011, Brno, Cehia, pag. 57-62;
- [15]. Baroiu, N., Teodor, V., Berbinschi, S., Oancea, N., Multi-Flute Helical Drills — The Modeling of a Hyperboloid Sharpening Method of the Back Faces, în *Proceedings of the*

## Lucrări transmise spre publicare

[1]. Berbinschi S., Teodor V., Oancea N., 3D Graphical Method for Profiling Gear Hob Tools, transmis la *International Journal of Advanced Manufacturing Technology*, 2011, ISSN: 0268-3768 (format tipărit), ISSN: 1433-3015 (format electronic);

[2]. Dura, G., Teodor V., Popa, I., Oancea N., Gear shaped tool profiling using Bezier polynomials, transmisă la *International Journal of Advanced Manufacturing Technology*, 2011, ISSN: 0268-3768 (format tipărit), ISSN: 1433-3015 (format electronic).

## Codurile sursă ale programelor realizate în cadrul contractului de cercetare

### Identificarea topologică a suprafețelor utilizând algoritmi genetici

```
% inregistrare Algoritmi Genetici
function [distanta]=transformare(parametri);
fi=parametri(1,1); psi=parametri(1,2); kapa=parametri(1,3);
dx=parametri(1,4); dy=parametri(1,5); dz=parametri(1,6);
load('masurare2.mat');
set_initial(:,1)=X1(:,1); set_initial(:,2)=X2(:,1); set_initial(:,3)=X3(:,1);
set_transformat(:,1)=x1(:,1); set_transformat(:,2)=x2(:,1); set_transformat(:,3)=x3(:,1);
nr_puncte=size(set_initial);
omega1=[[1, 0, 0];
        [0, cos(fi), sin(fi)];
        [0, -sin(fi), cos(fi)]];
omega2=[[cos(psi), 0, -sin(psi)];
        [0, 1, 0];
        [sin(psi), 0, cos(psi)]];
omega3=[[cos(kapa), sin(kapa), 0];
        [-sin(kapa), cos(kapa), 0];
        [0, 0, 1]];
T=[dx; dy; dz];
distanta=0;
for contor=1:1:nr_puncte(2);
    coordonate_transformate=omega1*omega2*omega3*set_initial(:,contor)+T;
    distanta=distanta+((coordonate_transformate(1,1)-set_transformat(1,contor))^2+...
        (coordonate_transformate(2,1)-set_transformat(2,contor))^2+...
        (coordonate_transformate(3,1)-set_transformat(3,contor))^2)^(1/2);
end;
distanta;
```

### Identificarea topologică a suprafețelor utilizând metoda circulației parametrilor

```
% Procesare cu circulatia parametrilor
syms fis psis kapas a1s a2s a3s x1s y1s z1s real;
contor=[0 0 0 0 0 0]; epsilon=0.01;
f_ob=2;
continua=1;
load('masurare2.mat');
Pi1=X1; Pi2=X2; Pi3=X3;
Pf1=x1; Pf2=x2; Pf3=x3;
```



```

omega1s=[[1 0 0];
          [0 cos(fis) sin(fis)];
          [0 -sin(fis) cos(fis)]];
omega2s=[[cos(psis) 0 -sin(psis)];
          [ 0 1 0];
          [sin(psis) 0 cos(psis)]];
omega3s=[[cos(kapas) sin(kapas) 0];
          [-sin(kapas) cos(kapas) 0];
          [0 0 1]];
A=[a1s; a2s; a3s];
p1=omega1s*omega2s*omega3s*Pi1+A;
p2=omega1s*omega2s*omega3s*Pi2+A;
p3=omega1s*omega2s*omega3s*Pi3+A;

FunctiaObiectiv=(p1(1,1)-Pf1(1,1))^2+(p1(2,1)-Pf1(2,1))^2+(p1(3,1)-Pf1(3,1))^2+...
                (p2(1,1)-Pf2(1,1))^2+(p2(2,1)-Pf2(2,1))^2+(p2(3,1)-Pf2(3,1))^2+...
                (p3(1,1)-Pf3(1,1))^2+(p3(2,1)-Pf3(2,1))^2+(p3(3,1)-Pf3(3,1))^2;
% Valorile minime ale intervalelor de cautare
lim_interv_fi=10^(-6)*pi/180;lim_interv_psi=10^(-6)*pi/180;lim_interv_kapa=10^(-6)*pi/180;
lim_interv_a1=10^(-6);lim_interv_a2=10^(-6);lim_interv_a3=10^(-6);
% Initializare parametri
fiv=-0.0246; psiv=0.0191; kapav=0.5617;
a1v=-16.9716; a2v=20.2064; a3v=0.0079;
fi_c=fiv;psi_c=psiv;kapa_c=kapav;
a1_c=a1v;a2_c=a2v;a3_c=a3v;
inc_fi=fi_c/2; inc_psi=psi_c/2; inc_kapa=kapa_c/2;
inc_a1=a1_c/2; inc_a2=a2_c/2; inc_a3=a3_c/2;
fi_max=fi_c+inc_fi; fi_min=fi_c-inc_fi;
psi_max=psi_c+inc_psi; psi_min=psi_c-inc_psi;
kapa_max=kapa_c+inc_kapa; kapa_min=kapa_c-inc_kapa;
a1_max=a1_c+inc_a1; a1_min=a1_c-inc_a1;
a2_max=a2_c+inc_a2; a2_min=a2_c-inc_a2;
a3_max=a3_c+inc_a3; a3_min=a3_c-inc_a3;
f_ob_fi=10^6; f_ob_psi=10^6; f_ob_kapa=10^6; f_ob_a1=10^6; f_ob_a2=10^6; f_ob_a3=10^6;
dFunctiaObiectiv_fi=diff(FunctiaObiectiv,fis);
dFunctiaObiectiv_psi=diff(FunctiaObiectiv,psis);
dFunctiaObiectiv_kapa=diff(FunctiaObiectiv,kapas);
dFunctiaObiectiv_a1=diff(FunctiaObiectiv,a1s);
dFunctiaObiectiv_a2=diff(FunctiaObiectiv,a2s);
dFunctiaObiectiv_a3=diff(FunctiaObiectiv,a3s);

% Inceput ciclu cautare
while continua;
    continua_fi=1;continua_psi=1;continua_kapa=1;continua_a1=1;continua_a2=1;continua_a3=1;
    % Incrementi cautare

dFunctiaObiectiv_fi_c=eval(subs(dFunctiaObiectiv_fi,{fis,psis,kapas,a1s,a2s,a3s},{fi_c,psi_c,ka
pa_c,a1_c,a2_c,a3_c}));
    if abs(dFunctiaObiectiv_fi_c)<=epsilon;
        continua_fi=0;
    end;
    while continua_fi;

```

```

dFunctiaObiectiv_fi_max=eval(subs(dFunctiaObiectiv_fi,{fis,psis,kapas,a1s,a2s,a3s}, ...
    {fi_max,psi_c,kapa_c,a1_c,a2_c,a3_c}));
dFunctiaObiectiv_fi_min=eval(subs(dFunctiaObiectiv_fi,{fis,psis,kapas,a1s,a2s,a3s},...
    {fi_min,psi_c,kapa_c,a1_c,a2_c,a3_c}));
dFunctiaObiectiv_fi_c=eval(subs(dFunctiaObiectiv_fi,{fis,psis,kapas,a1s,a2s,a3s},...
    {fi_c,psi_c,kapa_c,a1_c,a2_c,a3_c}));
if dFunctiaObiectiv_fi_max*dFunctiaObiectiv_fi_min>0;
    inc_fi=inc_fi*2;
    fi_max=fi_c+inc_fi; fi_min=fi_c-inc_fi; fi_c=(fi_max+fi_min)/2;
elseif dFunctiaObiectiv_fi_max*dFunctiaObiectiv_fi_c<0;
    fi_min=fi_c; fi_c=(fi_max+fi_min)/2; inc_fi=abs(fi_max-fi_min);
else
    fi_max=fi_c; fi_c=(fi_max+fi_min)/2; inc_fi=abs(fi_max-fi_min);
end;
if (contor(1,1)>10 | inc_fi<lim_interv_fi);
    continua_fi=0;
end;
contor(1,1)=contor(1,1)+1;
end;
%sfarsit bucla fi;

dFunctiaObiectiv_psi_c=subs(dFunctiaObiectiv_psi,{fis,psis,kapas,a1s,a2s,a3s},...
    {fi_c,psi_c,kapa_c,a1_c,a2_c,a3_c});
if abs(dFunctiaObiectiv_psi_c)<=epsilon;
    continua_psi=0;
end;
while continua_psi
    dFunctiaObiectiv_psi_max=subs(dFunctiaObiectiv_psi,{fis,psis,kapas,a1s,a2s,a3s},...
        {fi_c,psi_max,kapa_c,a1_c,a2_c,a3_c});
    dFunctiaObiectiv_psi_min=subs(dFunctiaObiectiv_psi,{fis,psis,kapas,a1s,a2s,a3s},...
        {fi_c,psi_min,kapa_c,a1_c,a2_c,a3_c});
    dFunctiaObiectiv_psi_c=subs(dFunctiaObiectiv_psi,{fis,psis,kapas,a1s,a2s,a3s},...
        {fi_c,psi_c,kapa_c,a1_c,a2_c,a3_c});
    if dFunctiaObiectiv_psi_max*dFunctiaObiectiv_psi_min>0;
        inc_psi=inc_psi*2;
        psi_max=psi_c+inc_psi; psi_min=psi_c-inc_psi; psi_c=(psi_max+psi_min)/2;
    elseif dFunctiaObiectiv_psi_max*dFunctiaObiectiv_psi_c<0;
        psi_min=psi_c; psi_c=(psi_max+psi_min)/2; inc_psi=abs(psi_max-psi_min);
    else
        psi_max=psi_c; psi_c=(psi_max+psi_min)/2; inc_psi=abs(psi_max-psi_min);
    end;
    if (contor(1,2)>10 | inc_psi<lim_interv_psi);
        continua_psi=0;
    end;
    contor(1,2)=contor(1,2)+1;
end;
%sfarsit bucla psi;

dFunctiaObiectiv_kapa_c=subs(dFunctiaObiectiv_kapa,{fis,psis,kapas,a1s,a2s,a3s},...
    {fi_c,psi_c,kapa_c,a1_c,a2_c,a3_c});
if abs(dFunctiaObiectiv_kapa_c)<=epsilon;
    continua_kapa=0;
end;

```

```

end;
while continua_kapa
    dFuncțiaObiectiv_kapa_max=subs(dFuncțiaObiectiv_kapa, {fis,psis,kapas,a1s,a2s,a3s},...
        {fi_c,psi_c,kapa_max,a1_c,a2_c,a3_c});
    dFuncțiaObiectiv_kapa_min=subs(dFuncțiaObiectiv_kapa, {fis,psis,kapas,a1s,a2s,a3s},...
        {fi_c,psi_c,kapa_min,a1_c,a2_c,a3_c});
    dFuncțiaObiectiv_kapa_c=subs(dFuncțiaObiectiv_kapa, {fis,psis,kapas,a1s,a2s,a3s},...
        {fi_c,psi_c,kapa_c,a1_c,a2_c,a3_c});
    if dFuncțiaObiectiv_kapa_max*dFuncțiaObiectiv_kapa_min>0;
        inc_kapa=inc_kapa*2;
        kapa_max=kapa_c+inc_kapa;
        kapa_min=kapa_c-inc_kapa;
        kapa_c=(kapa_max+kapa_min)/2;
    elseif dFuncțiaObiectiv_kapa_max*dFuncțiaObiectiv_kapa_c<0;
        kapa_min=kapa_c; kapa_c=(kapa_max+kapa_min)/2;
        inc_kapa=abs(kapa_max-kapa_min);
    else
        kapa_max=kapa_c; kapa_c=(kapa_max+kapa_min)/2;
        inc_kapa=abs(kapa_max-kapa_min);
    end;
    if (contor(1,3)>10 | inc_kapa<lim_interv_kapa);
        continua_kapa=0;
    end;
    contor(1,3)=contor(1,3)+1;
end;
%sfarsit bucla kapa;

dFuncțiaObiectiv_a1_c=eval(subs(dFuncțiaObiectiv_a1, {fis,psis,kapas,a1s,a2s,a3s},...
    {fi_c,psi_c,kapa_c,a1_c,a2_c,a3_c}));
if abs(dFuncțiaObiectiv_a1_c)<=epsilon;
    continua_a1=0;
end;
while continua_a1;
    dFuncțiaObiectiv_a1_max=eval(subs(dFuncțiaObiectiv_a1, {fis,psis,kapas,a1s,a2s,a3s},...
        {fi_c,psi_c,kapa_c,a1_max,a2_c,a3_c}));
    dFuncțiaObiectiv_a1_min=eval(subs(dFuncțiaObiectiv_a1, {fis,psis,kapas,a1s,a2s,a3s},...
        {fi_c,psi_c,kapa_c,a1_min,a2_c,a3_c}));
    dFuncțiaObiectiv_a1_c=eval(subs(dFuncțiaObiectiv_a1, {fis,psis,kapas,a1s,a2s,a3s},...
        {fi_c,psi_c,kapa_c,a1_c,a2_c,a3_c}));
    if dFuncțiaObiectiv_a1_max*dFuncțiaObiectiv_a1_min>0;
        inc_a1=inc_a1*2;
        a1_max=a1_c+inc_a1; a1_min=a1_c-inc_a1; a1_c=(a1_max+a1_min)/2;
    elseif dFuncțiaObiectiv_a1_max*dFuncțiaObiectiv_a1_c<0;
        a1_min=a1_c; a1_c=(a1_max+a1_min)/2; inc_a1=abs(a1_max-a1_min);
    else
        a1_max=a1_c; a1_c=(a1_max+a1_min)/2; inc_a1=abs(a1_max-a1_min);
    end;
    if (contor(1,4)>10 | inc_a1<lim_interv_a1);
        continua_a1=0;
    end;
    contor(1,4)=contor(1,4)+1;
end;
end;

```

```

%sfarsit bucla a1;

dFunctiaObiectiv_a2_c=eval(subs(dFunctiaObiectiv_a2,{fis,psis,kapas,a1s,a2s,a3s},...
    {fi_c,psi_c,kapa_c,a1_c,a2_c,a3_c}));
if abs(dFunctiaObiectiv_a2_c)<=epsilon;
    continua_a2=0;
end;
while continua_a2
    dFunctiaObiectiv_a2_max=eval(subs(dFunctiaObiectiv_a2,{fis,psis,kapas,a1s,a2s,a3s},...
        {fi_c,psi_c,kapa_c,a1_c,a2_max,a3_c}));
    dFunctiaObiectiv_a2_min=eval(subs(dFunctiaObiectiv_a2,{fis,psis,kapas,a1s,a2s,a3s},...
        {fi_c,psi_c,kapa_c,a1_c,a2_min,a3_c}));
    dFunctiaObiectiv_a2_c=eval(subs(dFunctiaObiectiv_a2,{fis,psis,kapas,a1s,a2s,a3s},...
        {fi_c,psi_c,kapa_c,a1_c,a2_c,a3_c}));
    if dFunctiaObiectiv_a2_max*dFunctiaObiectiv_a2_min>0;
        inc_a2=inc_a2*2;
        a2_max=a2_c+inc_a2; a2_min=a2_c-inc_a2; a2_c=(a2_max+a2_min)/2;
    elseif dFunctiaObiectiv_a2_max*dFunctiaObiectiv_a2_c<0;
        a2_min=a2_c; a2_c=(a2_max+a2_min)/2; inc_a2=abs(a2_max-a2_min);
    else
        a2_max=a2_c; a2_c=(a2_max+a2_min)/2; inc_a2=abs(a2_max-a2_min);
    end;
    if (contor(1,5)>10 | inc_a2<lim_interv_a2);
        continua_a2=0;
    end;
    contor(1,5)=contor(1,5)+1;
end;
%sfarsit bucla a2;

dFunctiaObiectiv_a3_c=eval(subs(dFunctiaObiectiv_a3,{fis,psis,kapas,a1s,a2s,a3s},...
    {fi_c,psi_c,kapa_c,a1_c,a2_c,a3_c}));
if abs(dFunctiaObiectiv_a3_c)<=epsilon;
    continua_a3=0;
end;
while continua_a3
    dFunctiaObiectiv_a3_max=eval(subs(dFunctiaObiectiv_a3,{fis,psis,kapas,a1s,a2s,a3s},...
        {fi_c,psi_c,kapa_c,a1_c,a2_c,a3_max}));
    dFunctiaObiectiv_a3_min=eval(subs(dFunctiaObiectiv_a3,{fis,psis,kapas,a1s,a2s,a3s},...
        {fi_c,psi_c,kapa_c,a1_c,a2_c,a3_min}));
    dFunctiaObiectiv_a3_c=eval(subs(dFunctiaObiectiv_a3,{fis,psis,kapas,a1s,a2s,a3s},...
        {fi_c,psi_c,kapa_c,a1_c,a2_c,a3_c}));
    if dFunctiaObiectiv_a3_max*dFunctiaObiectiv_a3_min>0;
        inc_a3=inc_a3*2;
        a3_max=a3_c+inc_a3; a3_min=a3_c-inc_a3; a3_c=(a3_max+a3_min)/2;
    elseif dFunctiaObiectiv_a3_max*dFunctiaObiectiv_a3_c<0;
        a3_min=a3_c; a3_c=(a3_max+a3_min)/2; inc_a3=abs(a3_max-a3_min);
    else
        a3_max=a3_c; a3_c=(a3_max+a3_min)/2; inc_a3=abs(a3_max-a3_min);
    end;
    if (contor(1,6)>10 | inc_a3<lim_interv_a3);
        continua_a3=0;
    end;
end;

```

```

    contor(1,6)=contor(1,6)+1;
end;
%sfarsit bucla a3;

omega1=[[1 0 0];[0 cos(fi_c) sin(fi_c)];[0 -sin(fi_c) cos(fi_c)]];
omega2=[[cos(psi_c) 0 -sin(psi_c)];[0 1 0];[sin(psi_c) 0 cos(psi_c)]];
omega3=[[cos(kapa_c) sin(kapa_c) 0];[-sin(kapa_c) cos(kapa_c) 0];[0 0 1]];
Av=[a1_c; a2_c; a3_c];
p1v=omega1*omega2*omega3*Pi1+Av;
p2v=omega1*omega2*omega3*Pi2+Av;
p3v=omega1*omega2*omega3*Pi3+Av;
FunctiaObiectiv=((p1v(1,1)-Pf1(1,1))^2+(p1v(2,1)-Pf1(2,1))^2+(p1v(3,1)-
Pf1(3,1))^2)^(1/2)+...
((p2v(1,1)-Pf2(1,1))^2+(p2v(2,1)-Pf2(2,1))^2+(p2v(3,1)-Pf2(3,1))^2)^(1/2)+...
((p3v(1,1)-Pf3(1,1))^2+(p3v(2,1)-Pf3(2,1))^2+(p3v(3,1)-Pf3(3,1))^2)^(1/2);
fval_cp=FunctiaObiectiv
continua=(fval_cp>epsilon & contor(1,7)<10);
cont=contor(1,7)+1;
contor=[0 0 0 0 0 cont]
end;
solutie=[fi_c psi_c kapa_c a1_c a2_c a3_c]

```

### **Identificarea topologică a suprafețelor utilizând metoda mixtă (pre-procesare prin algoritmi genetici și identificare finală prin metoda circulației parametrilor)**

```

% Pre-procesare cu algoritmi genetici
generatii=5000;
obiectiv=6;
options=gaoptimset('Generations',generatii,'FitnessLimit',obiectiv,'StallGenLimit',Inf,'StallTime
Limit',Inf);
[solutie_initala fval] = ga(@inregistrare_3D, 6, options)

```

```

% Procesare cu circulația parametrilor
syms fis psis kapas a1s a2s a3s x1s y1s z1s real;
contor=[0 0 0 0 0 0]; epsilon=0.01;
f_ob=fval;
continua=1;
load('masurare2.mat');
Pi1=X1; Pi2=X2; Pi3=X3;
Pf1=x1; Pf2=x2; Pf3=x3;

```

```

omega1s=[[1 0 0];
[0 cos(fis) sin(fis)];
[0 -sin(fis) cos(fis)]];
omega2s=[[cos(psis) 0 -sin(psis)];
[0 1 0];
[sin(psis) 0 cos(psis)]];
omega3s=[[cos(kapas) sin(kapas) 0];
[-sin(kapas) cos(kapas) 0];
[0 0 1]];
A=[a1s; a2s; a3s];
p1=omega1s*omega2s*omega3s*Pi1+A;

```

```

p2=omega1s*omega2s*omega3s*Pi2+A;
p3=omega1s*omega2s*omega3s*Pi3+A;

FunctiaObiectiv=((p1(1,1)-Pf1(1,1))^2+(p1(2,1)-Pf1(2,1))^2+(p1(3,1)-Pf1(3,1))^2+...
    (p2(1,1)-Pf2(1,1))^2+(p2(2,1)-Pf2(2,1))^2+(p2(3,1)-Pf2(3,1))^2+...
    (p3(1,1)-Pf3(1,1))^2+(p3(2,1)-Pf3(2,1))^2+(p3(3,1)-Pf3(3,1))^2)^(1/2);
% Valorile minime ale intervalelor de cautare
lim_interv_fi=10^(-6)*pi/180;lim_interv_psi=10^(-6)*pi/180;lim_interv_kapa=10^(-6)*pi/180;
lim_interv_a1=10^(-6);lim_interv_a2=10^(-6);lim_interv_a3=10^(-6);
% Initializare parametri

fiv=solutie_initiala(1); psiv=solutie_initiala(2); kapav=solutie_initiala(3);
a1v=solutie_initiala(4); a2v=solutie_initiala(5); a3v=solutie_initiala(6);
fi_c=fiv;psi_c=psiv;kapa_c=kapav;
a1_c=a1v;a2_c=a2v;a3_c=a3v;
inc_fi=fi_c/2; inc_psi=psi_c/2; inc_kapa=kapa_c/2;
inc_a1=a1_c/2; inc_a2=a2_c/2; inc_a3=a3_c/2;
fi_max=fi_c+inc_fi; fi_min=fi_c-inc_fi;
psi_max=psi_c+inc_psi; psi_min=psi_c-inc_psi;
kapa_max=kapa_c+inc_kapa; kapa_min=kapa_c-inc_kapa;
a1_max=a1_c+inc_a1; a1_min=a1_c-inc_a1;
a2_max=a2_c+inc_a2; a2_min=a2_c-inc_a2;
a3_max=a3_c+inc_a3; a3_min=a3_c-inc_a3;
f_ob_fi=10^6; f_ob_psi=10^6; f_ob_kapa=10^6; f_ob_a1=10^6; f_ob_a2=10^6; f_ob_a3=10^6;
dFunctiaObiectiv_fi=diff(FunctiaObiectiv,fi);
dFunctiaObiectiv_psi=diff(FunctiaObiectiv,psi);
dFunctiaObiectiv_kapa=diff(FunctiaObiectiv,kapas);
dFunctiaObiectiv_a1=diff(FunctiaObiectiv,a1s);
dFunctiaObiectiv_a2=diff(FunctiaObiectiv,a2s);
dFunctiaObiectiv_a3=diff(FunctiaObiectiv,a3s);

% Inceput ciclu cautare
while continua;

continua_fi=1;continua_psi=1;continua_kapa=1;continua_a1=1;continua_a2=1;continua_a3=1;
% Incrementi cautare
dFunctiaObiectiv_fi_c=subs(dFunctiaObiectiv_fi,{fis,psis,kapas,a1s,a2s,a3s},...
    {fi_c,psi_c,kapa_c,a1_c,a2_c,a3_c});
if abs(dFunctiaObiectiv_fi_c)<=epsilon;
    continua_fi=0;
end;
while continua_fi;
    dFunctiaObiectiv_fi_max=subs(dFunctiaObiectiv_fi,{fis,psis,kapas,a1s,a2s,a3s},...
        {fi_max,psi_c,kapa_c,a1_c,a2_c,a3_c});
    dFunctiaObiectiv_fi_min=subs(dFunctiaObiectiv_fi,{fis,psis,kapas,a1s,a2s,a3s},...
        {fi_min,psi_c,kapa_c,a1_c,a2_c,a3_c});
    dFunctiaObiectiv_fi_c=subs(dFunctiaObiectiv_fi,{fis,psis,kapas,a1s,a2s,a3s},...
        {fi_c,psi_c,kapa_c,a1_c,a2_c,a3_c});
    if dFunctiaObiectiv_fi_max*dFunctiaObiectiv_fi_min>0;
        inc_fi=inc_fi*2;
        fi_max=fi_c+inc_fi; fi_min=fi_c-inc_fi; fi_c=(fi_max+fi_min)/2;
    elseif dFunctiaObiectiv_fi_max*dFunctiaObiectiv_fi_c<0;

```

```

    fi_min=fi_c; fi_c=(fi_max+fi_min)/2; inc_fi=abs(fi_max-fi_min);
else
    fi_max=fi_c; fi_c=(fi_max+fi_min)/2; inc_fi=abs(fi_max-fi_min);
end;
if (contor(1,1)>10 | inc_fi<lim_interv_fi);
    continua_fi=0;
end;
contor(1,1)=contor(1,1)+1;
end;
%sfarsit bucla fi;

dFunctiaObiectiv_psi_c=subs(dFunctiaObiectiv_psi,{fis,psis,kapas,a1s,a2s,a3s},...
    {fi_c,psi_c,kapa_c,a1_c,a2_c,a3_c});
if abs(dFunctiaObiectiv_psi_c)<=epsilon;
    continua_psi=0;
end;
while continua_psi
    dFunctiaObiectiv_psi_max=subs(dFunctiaObiectiv_psi,{fis,psis,kapas,a1s,a2s,a3s},...
        {fi_c,psi_max,kapa_c,a1_c,a2_c,a3_c});
    dFunctiaObiectiv_psi_min=subs(dFunctiaObiectiv_psi,{fis,psis,kapas,a1s,a2s,a3s},...
        {fi_c,psi_min,kapa_c,a1_c,a2_c,a3_c});
    dFunctiaObiectiv_psi_c=subs(dFunctiaObiectiv_psi,{fis,psis,kapas,a1s,a2s,a3s},...
        {fi_c,psi_c,kapa_c,a1_c,a2_c,a3_c});
    if dFunctiaObiectiv_psi_max*dFunctiaObiectiv_psi_min>0;
        inc_psi=inc_psi*2;
        psi_max=psi_c+inc_psi; psi_min=psi_c-inc_psi; psi_c=(psi_max+psi_min)/2;
    elseif dFunctiaObiectiv_psi_max*dFunctiaObiectiv_psi_c<0;
        psi_min=psi_c; psi_c=(psi_max+psi_min)/2; inc_psi=abs(psi_max-psi_min);
    else
        psi_max=psi_c; psi_c=(psi_max+psi_min)/2; inc_psi=abs(psi_max-psi_min);
    end;
    if (contor(1,2)>10 | inc_psi<lim_interv_psi);
        continua_psi=0;
    end;
    contor(1,2)=contor(1,2)+1;
end;
%sfarsit bucla psi;

dFunctiaObiectiv_kapa_c=subs(dFunctiaObiectiv_kapa,{fis,psis,kapas,a1s,a2s,a3s},...
    {fi_c,psi_c,kapa_c,a1_c,a2_c,a3_c});
if abs(dFunctiaObiectiv_kapa_c)<=epsilon;
    continua_kapa=0;
end;
while continua_kapa
    dFunctiaObiectiv_kapa_max=subs(dFunctiaObiectiv_kapa,{fis,psis,kapas,a1s,a2s,a3s},...
        {fi_c,psi_c,kapa_max,a1_c,a2_c,a3_c});
    dFunctiaObiectiv_kapa_min=subs(dFunctiaObiectiv_kapa,{fis,psis,kapas,a1s,a2s,a3s},...
        {fi_c,psi_c,kapa_min,a1_c,a2_c,a3_c});
    dFunctiaObiectiv_kapa_c=subs(dFunctiaObiectiv_kapa,{fis,psis,kapas,a1s,a2s,a3s},...
        {fi_c,psi_c,kapa_c,a1_c,a2_c,a3_c});
    if dFunctiaObiectiv_kapa_max*dFunctiaObiectiv_kapa_min>0;
        inc_kapa=inc_kapa*2;

```

```

    kapa_max=kapa_c+inc_kapa;
    kapa_min=kapa_c-inc_kapa;
    kapa_c=(kapa_max+kapa_min)/2;
elseif dFuncțiaObiectiv_kapa_max*dFuncțiaObiectiv_kapa_c<0;
    kapa_min=kapa_c; kapa_c=(kapa_max+kapa_min)/2;
    inc_kapa=abs(kapa_max-kapa_min);
else
    kapa_max=kapa_c; kapa_c=(kapa_max+kapa_min)/2;
    inc_kapa=abs(kapa_max-kapa_min);
end;
if (contor(1,3)>10 | inc_kapa<lim_interv_kapa);
    continua_kapa=0;
end;
contor(1,3)=contor(1,3)+1;
end;
%sfarsit bucla kapa;

dFuncțiaObiectiv_a1_c=subs(dFuncțiaObiectiv_a1, {fis,psis,kapas,a1s,a2s,a3s},...
    {fi_c,psi_c,kapa_c,a1_c,a2_c,a3_c});
if abs(dFuncțiaObiectiv_a1_c)<=epsilon;
    continua_a1=0;
end;
while continua_a1;
    dFuncțiaObiectiv_a1_max=subs(dFuncțiaObiectiv_a1, {fis,psis,kapas,a1s,a2s,a3s},...
        {fi_c,psi_c,kapa_c,a1_max,a2_c,a3_c});
    dFuncțiaObiectiv_a1_min=subs(dFuncțiaObiectiv_a1, {fis,psis,kapas,a1s,a2s,a3s},...
        {fi_c,psi_c,kapa_c,a1_min,a2_c,a3_c});
    dFuncțiaObiectiv_a1_c=subs(dFuncțiaObiectiv_a1, {fis,psis,kapas,a1s,a2s,a3s},...
        {fi_c,psi_c,kapa_c,a1_c,a2_c,a3_c});
    if dFuncțiaObiectiv_a1_max*dFuncțiaObiectiv_a1_min>0;
        inc_a1=inc_a1*2;
        a1_max=a1_c+inc_a1; a1_min=a1_c-inc_a1; a1_c=(a1_max+a1_min)/2;
    elseif dFuncțiaObiectiv_a1_max*dFuncțiaObiectiv_a1_c<0;
        a1_min=a1_c; a1_c=(a1_max+a1_min)/2; inc_a1=abs(a1_max-a1_min);
    else
        a1_max=a1_c; a1_c=(a1_max+a1_min)/2; inc_a1=abs(a1_max-a1_min);
    end;
end;
if (contor(1,4)>10 | inc_a1<lim_interv_a1);
    continua_a1=0;
end;
contor(1,4)=contor(1,4)+1;
end;
%sfarsit bucla a1;

dFuncțiaObiectiv_a2_c=subs(dFuncțiaObiectiv_a2, {fis,psis,kapas,a1s,a2s,a3s},...
    {fi_c,psi_c,kapa_c,a1_c,a2_c,a3_c});
if abs(dFuncțiaObiectiv_a2_c)<=epsilon;
    continua_a2=0;
end;
while continua_a2
    dFuncțiaObiectiv_a2_max=subs(dFuncțiaObiectiv_a2, {fis,psis,kapas,a1s,a2s,a3s},...
        {fi_c,psi_c,kapa_c,a1_c,a2_max,a3_c});

```



```

dFunctiaObiectiv_a2_min=subs(dFunctiaObiectiv_a2,{fis,psis,kapas,a1s,a2s,a3s},...
    {fi_c,psi_c,kapa_c,a1_c,a2_min,a3_c});
dFunctiaObiectiv_a2_c=subs(dFunctiaObiectiv_a2,{fis,psis,kapas,a1s,a2s,a3s},...
    {fi_c,psi_c,kapa_c,a1_c,a2_c,a3_c});
if dFunctiaObiectiv_a2_max*dFunctiaObiectiv_a2_min>0;
    inc_a2=inc_a2*2;
    a2_max=a2_c+inc_a2; a2_min=a2_c-inc_a2; a2_c=(a2_max+a2_min)/2;
elseif dFunctiaObiectiv_a2_max*dFunctiaObiectiv_a2_c<0;
    a2_min=a2_c; a2_c=(a2_max+a2_min)/2; inc_a2=abs(a2_max-a2_min);
else
    a2_max=a2_c; a2_c=(a2_max+a2_min)/2; inc_a2=abs(a2_max-a2_min);
end;
if (contor(1,5)>10 | inc_a2<lim_interv_a2);
    continua_a2=0;
end;
contor(1,5)=contor(1,5)+1;
end;
%sfarsit bucla a2;

dFunctiaObiectiv_a3_c=subs(dFunctiaObiectiv_a3,{fis,psis,kapas,a1s,a2s,a3s},...
    {fi_c,psi_c,kapa_c,a1_c,a2_c,a3_c});
if abs(dFunctiaObiectiv_a3_c)<=epsilon;
    continua_a3=0;
end;
while continua_a3
    dFunctiaObiectiv_a3_max=subs(dFunctiaObiectiv_a3,{fis,psis,kapas,a1s,a2s,a3s},...
        {fi_c,psi_c,kapa_c,a1_c,a2_c,a3_max});
    dFunctiaObiectiv_a3_min=subs(dFunctiaObiectiv_a3,{fis,psis,kapas,a1s,a2s,a3s},...
        {fi_c,psi_c,kapa_c,a1_c,a2_c,a3_min});
    dFunctiaObiectiv_a3_c=subs(dFunctiaObiectiv_a3,{fis,psis,kapas,a1s,a2s,a3s},...
        {fi_c,psi_c,kapa_c,a1_c,a2_c,a3_c});
    if dFunctiaObiectiv_a3_max*dFunctiaObiectiv_a3_min>0;
        inc_a3=inc_a3*2;
        a3_max=a3_c+inc_a3; a3_min=a3_c-inc_a3; a3_c=(a3_max+a3_min)/2;
    elseif dFunctiaObiectiv_a3_max*dFunctiaObiectiv_a3_c<0;
        a3_min=a3_c; a3_c=(a3_max+a3_min)/2; inc_a3=abs(a3_max-a3_min);
    else
        a3_max=a3_c; a3_c=(a3_max+a3_min)/2; inc_a3=abs(a3_max-a3_min);
    end;
    if (contor(1,6)>10 | inc_a3<lim_interv_a3);
        continua_a3=0;
    end;
    contor(1,6)=contor(1,6)+1;
end;
%sfarsit bucla a3;

omega1=[[1 0 0];[0 cos(fi_c) sin(fi_c)];[0 -sin(fi_c) cos(fi_c)]];
omega2=[[cos(psi_c) 0 -sin(psi_c)];[0 1 0];[sin(psi_c) 0 cos(psi_c)]];
omega3=[[cos(kapa_c) sin(kapa_c) 0];[-sin(kapa_c) cos(kapa_c) 0];[0 0 1]];
Av=[a1_c; a2_c; a3_c];
p1v=omega1*omega2*omega3*Pi1+Av;
p2v=omega1*omega2*omega3*Pi2+Av;

```

```

p3v=omega1*omega2*omega3*Pi3+Av;
FunctiaObiectiv=((p1v(1,1)-Pf1(1,1))^2+(p1v(2,1)-Pf1(2,1))^2+(p1v(3,1)-
Pf1(3,1))^2)^(1/2)+...
((p2v(1,1)-Pf2(1,1))^2+(p2v(2,1)-Pf2(2,1))^2+(p2v(3,1)-Pf2(3,1))^2)^(1/2)+...
((p3v(1,1)-Pf3(1,1))^2+(p3v(2,1)-Pf3(2,1))^2+(p3v(3,1)-Pf3(3,1))^2)^(1/2);
fval_cp=FunctiaObiectiv
continua=(fval_cp>epsilon & contor(1,7)<30);
cont=contor(1,7)+1;
contor=[0 0 0 0 0 0 cont]
end;
solutie=[fi_c psi_c kapa_c a1_c a2_c a3_c]

```

### Soft de schimbare a coordonatelor punctelor măsurate

```

% transformare de coordonate dintr-un SR in altul
% se utilizeaza impreuna cu unul dintre softurile de identificare prezentate anterior
fi=-0.0135; psi=-0.0036; kapa=-0.2593;
dx=-28.3152; dy=10.2231; dz=0.0293;
% parametrii fi, psi, kapa, dx, dy, dz sunt identificati prin una dintre metodele prezentate anterior
load('alezaj_transformat.mat');
omega1=[[1 0 0];
[0 cos(fi) sin(fi)];
[0 -sin(fi) cos(fi)]];
omega2=[[cos(psi) 0 -sin(psi)];
[0 1 0];
[sin(psi) 0 cos(psi)]];
omega3=[[cos(kapa) sin(kapa) 0];
[-sin(kapa) cos(kapa) 0];
[0 0 1]];
A=[dx; dy; dz];
nr_puncte=size(Pf);
for contor=1:1:nr_puncte(2);
Pm(:,contor)=omega3'*omega2'*omega1'*(Pf(:,contor)-A);
end;

```

### Soft de identificare a unui cilindru pe baza coordonatelor punctelor aparținând suprafeței acestuia

```

% Identificare cilindru cu circulatia parametrilor
syms fis psis a1s a2s Rs real;
contor=[0 0 0 0 0 0]; epsilon=0.01;
f_ob=100000;
continua=1;
FunctiaObiectiv=0;
load('alezaj.mat');

omega1s=[[1 0 0];
[0 cos(fis) sin(fis)];
[0 -sin(fis) cos(fis)]];
omega2s=[[cos(psis) 0 -sin(psis)];
[0 1 0];
[sin(psis) 0 cos(psis)]];

```

```

A=[a1s; a2s; 0];
for c=1:1:length(Pf);
    p(:,c)=omega1s*omega2s*Pf(:,c)+A;
    FunctiaObiectiv=FunctiaObiectiv+ABS((p(1,c)^2+p(2,c)^2-Rs^2));
end;

% Valorile minime ale intervalelor de cautare
lim_interv_fi=10^(-6)*pi/180;lim_interv_psi=10^(-6)*pi/180;
lim_interv_a1=10^(-6);lim_interv_a2=10^(-6);lim_interv_R=10^(-6);
% Initializare parametri

fiv=0; psiv=0;
a1v=60.2128; a2v=72.9735; Rv=31.88;
fi_c=fiv;psi_c=psiv;
a1_c=a1v;a2_c=a2v;R_c=Rv;
inc_fi=fi_c/2; inc_psi=psi_c/2;
inc_a1=a1_c/2; inc_a2=a2_c/2; inc_R=R_c/2;
fi_max=fi_c+inc_fi; fi_min=fi_c-inc_fi;
psi_max=psi_c+inc_psi; psi_min=psi_c-inc_psi;
a1_max=a1_c+inc_a1; a1_min=a1_c-inc_a1;
a2_max=a2_c+inc_a2; a2_min=a2_c-inc_a2;
R_max=R_c+inc_R; R_min=R_c-inc_R;
f_ob_fi=10^6; f_ob_psi=10^6; f_ob_a1=10^6; f_ob_a2=10^6; f_ob_R=10^6;
FunctiaObiectiv_v=0;
dFunctiaObiectiv_fi=diff(FunctiaObiectiv,fi);
dFunctiaObiectiv_psi=diff(FunctiaObiectiv,psi);
dFunctiaObiectiv_a1=diff(FunctiaObiectiv,a1s);
dFunctiaObiectiv_a2=diff(FunctiaObiectiv,a2s);
dFunctiaObiectiv_R=diff(FunctiaObiectiv,Rs);

% Inceput ciclului de cautare
while continua;
    continua_fi=1;continua_psi=1;continua_a1=1;continua_a2=1;continua_R=1;
    FunctiaObiectiv_v=0;
    % Incrementi de cautare
    dFunctiaObiectiv_fi_c=eval((subs(dFunctiaObiectiv_fi,{fis,psis,a1s,a2s,Rs},...
        {fi_c,psi_c,a1_c,a2_c,R_c})));
    if abs(dFunctiaObiectiv_fi_c)<=epsilon;
        continua_fi=0;
    end;
    contor(1,1)=0;
    while continua_fi;
        dFunctiaObiectiv_fi_max=eval((subs(dFunctiaObiectiv_fi,{fis,psis,a1s,a2s,Rs},...
            {fi_max,psi_c,a1_c,a2_c,R_c})));
        dFunctiaObiectiv_fi_min=eval((subs(dFunctiaObiectiv_fi,{fis,psis,a1s,a2s,Rs},...
            {fi_min,psi_c,a1_c,a2_c,R_c})));
        dFunctiaObiectiv_fi_c=eval((subs(dFunctiaObiectiv_fi,{fis,psis,a1s,a2s,Rs},...
            {fi_c,psi_c,a1_c,a2_c,R_c})));
        if dFunctiaObiectiv_fi_max*dFunctiaObiectiv_fi_min>0;
            inc_fi=inc_fi*2;
            fi_max=fi_c+inc_fi; fi_min=fi_c-inc_fi; fi_c=(fi_max+fi_min)/2;
        elseif dFunctiaObiectiv_fi_max*dFunctiaObiectiv_fi_c<0;

```

```

    fi_min=fi_c; fi_c=(fi_max+fi_min)/2; inc_fi=abs(fi_max-fi_min);
else
    fi_max=fi_c; fi_c=(fi_max+fi_min)/2; inc_fi=abs(fi_max-fi_min);
end;
if (contor(1,1)>10 | inc_fi<lim_interv_fi);
    continua_fi=0;
end;
contor(1,1)=contor(1,1)+1;
end;
%sfarsit bucla fi;

dFunctiaObiectiv_psi_c=eval((subs(dFunctiaObiectiv_psi, {fis,psis,a1s,a2s,Rs},...
    {fi_c,psi_c,a1_c,a2_c,R_c})));
if abs(dFunctiaObiectiv_psi_c)<=epsilon;
    continua_psi=0;
end;
contor(1,2)=0;
while continua_psi
    dFunctiaObiectiv_psi_max=eval((subs(dFunctiaObiectiv_psi, {fis,psis,a1s,a2s,Rs},...
        {fi_c,psi_max,a1_c,a2_c,R_c})));
    dFunctiaObiectiv_psi_min=eval((subs(dFunctiaObiectiv_psi, {fis,psis,a1s,a2s,Rs},...
        {fi_c,psi_min,a1_c,a2_c,R_c})));
    dFunctiaObiectiv_psi_c=eval((subs(dFunctiaObiectiv_psi, {fis,psis,a1s,a2s,Rs},...
        {fi_c,psi_c,a1_c,a2_c,R_c})));
    if dFunctiaObiectiv_psi_max*dFunctiaObiectiv_psi_min>0;
        inc_psi=inc_psi*2;
        psi_max=psi_c+inc_psi; psi_min=psi_c-inc_psi; psi_c=(psi_max+psi_min)/2;
    elseif dFunctiaObiectiv_psi_max*dFunctiaObiectiv_psi_c<0;
        psi_min=psi_c; psi_c=(psi_max+psi_min)/2; inc_psi=abs(psi_max-psi_min);
    else
        psi_max=psi_c; psi_c=(psi_max+psi_min)/2; inc_psi=abs(psi_max-psi_min);
    end;
    if (contor(1,2)>10 | inc_psi<lim_interv_psi);
        continua_psi=0;
    end;
    contor(1,2)=contor(1,2)+1;
end;
%sfarsit bucla psi;

dFunctiaObiectiv_a1_c=eval((subs(dFunctiaObiectiv_a1, {fis,psis,a1s,a2s,Rs},...
    {fi_c,psi_c,a1_c,a2_c,R_c})));
if abs(dFunctiaObiectiv_a1_c)<=epsilon;
    continua_a1=0;
end;
contor(1,4)=0;
while continua_a1;
    dFunctiaObiectiv_a1_max=eval((subs(dFunctiaObiectiv_a1, {fis,psis,a1s,a2s,Rs},...
        {fi_c,psi_c,a1_max,a2_c,R_c})));
    dFunctiaObiectiv_a1_min=eval((subs(dFunctiaObiectiv_a1, {fis,psis,a1s,a2s,Rs},...
        {fi_c,psi_c,a1_min,a2_c,R_c})));
    dFunctiaObiectiv_a1_c=eval((subs(dFunctiaObiectiv_a1, {fis,psis,a1s,a2s,Rs},...
        {fi_c,psi_c,a1_c,a2_c,R_c})));

```

```

if dFunctiaObiectiv_a1_max*dFunctiaObiectiv_a1_min>0;
    inc_a1=inc_a1*2;
    a1_max=a1_c+inc_a1; a1_min=a1_c-inc_a1; a1_c=(a1_max+a1_min)/2;
elseif dFunctiaObiectiv_a1_max*dFunctiaObiectiv_a1_c<0;
    a1_min=a1_c; a1_c=(a1_max+a1_min)/2; inc_a1=abs(a1_max-a1_min);
else
    a1_max=a1_c; a1_c=(a1_max+a1_min)/2; inc_a1=abs(a1_max-a1_min);
end;
if (contor(1,4)>10 | inc_a1<lim_interv_a1);
    continua_a1=0;
end;
contor(1,4)=contor(1,4)+1;
end;
%sfarsit bucla a1;

dFunctiaObiectiv_a2_c=eval((subs(dFunctiaObiectiv_a2,{fis,psis,a1s,a2s,Rs},...
    {fi_c,psi_c,a1_c,a2_c,R_c})));
if abs(dFunctiaObiectiv_a2_c)<=epsilon;
    continua_a2=0;
end;
contor(1,5)=0;
while continua_a2
    dFunctiaObiectiv_a2_max=eval((subs(dFunctiaObiectiv_a2,{fis,psis,a1s,a2s,Rs},...
        {fi_c,psi_c,a1_c,a2_max,R_c})));
    dFunctiaObiectiv_a2_min=eval((subs(dFunctiaObiectiv_a2,{fis,psis,a1s,a2s,Rs},...
        {fi_c,psi_c,a1_c,a2_min,R_c})));
    dFunctiaObiectiv_a2_c=eval((subs(dFunctiaObiectiv_a2,{fis,psis,a1s,a2s,Rs},...
        {fi_c,psi_c,a1_c,a2_c,R_c})));
    if dFunctiaObiectiv_a2_max*dFunctiaObiectiv_a2_min>0;
        inc_a2=inc_a2*2;
        a2_max=a2_c+inc_a2; a2_min=a2_c-inc_a2; a2_c=(a2_max+a2_min)/2;
    elseif dFunctiaObiectiv_a2_max*dFunctiaObiectiv_a2_c<0;
        a2_min=a2_c; a2_c=(a2_max+a2_min)/2; inc_a2=abs(a2_max-a2_min);
    else
        a2_max=a2_c; a2_c=(a2_max+a2_min)/2; inc_a2=abs(a2_max-a2_min);
    end;
    if (contor(1,5)>10 | inc_a2<lim_interv_a2);
        continua_a2=0;
    end;
    contor(1,5)=contor(1,5)+1;
end;
%sfarsit bucla a2;

dFunctiaObiectiv_R_c=eval((subs(dFunctiaObiectiv_R,{fis,psis,a1s,a2s,Rs},...
    {fi_c,psi_c,a1_c,a2_c,R_c})));
if abs(dFunctiaObiectiv_R_c)<=epsilon;
    continua_R=0;
end;
contor(1,6)=0;
while continua_R
    dFunctiaObiectiv_R_max=eval((subs(dFunctiaObiectiv_R,{fis,psis,a1s,a2s,Rs},...
        {fi_c,psi_c,a1_c,a2_c,R_max})));

```

```

dFunctiaObiectiv_R_min=eval((subs(dFunctiaObiectiv_R,{fis,psis,a1s,a2s,Rs},...
{fi_c,psi_c,a1_c,a2_c,R_min})));
dFunctiaObiectiv_R_c=eval((subs(dFunctiaObiectiv_R,{fis,psis,a1s,a2s,Rs},...
{fi_c,psi_c,a1_c,a2_c,R_c})));
if dFunctiaObiectiv_R_max*dFunctiaObiectiv_R_min>0;
inc_R=inc_R*2;
R_max=R_c+inc_R; R_min=R_c-inc_R; R_c=(R_max+R_min)/2;
elseif dFunctiaObiectiv_R_max*dFunctiaObiectiv_R_c<0;
R_min=R_c; R_c=(R_max+R_min)/2; inc_R=abs(R_max-R_min);
else
R_max=R_c; R_c=(R_max+R_min)/2; inc_R=abs(R_max-R_min);
end;
if (contor(1,6)>10 | inc_R<lim_interv_R);
continua_R=0;
end;
contor(1,6)=contor(1,6)+1;
end;
%sfarsit bucla R;

omega1=[[1 0 0];[0 cos(fi_c) sin(fi_c)];[0 -sin(fi_c) cos(fi_c)]];
omega2=[[cos(psi_c) 0 -sin(psi_c)];[0 1 0];[sin(psi_c) 0 cos(psi_c)]];
Av=[a1_c; a2_c; 0];
for c=1:1:length(Pf);
pv(:,c)=omega1*omega2*Pf(:,c)+Av;
FunctiaObiectiv_v=FunctiaObiectiv_v+abs((pv(1,c)^2+pv(2,c)^2-R_c^2));
end;
fval_cp=FunctiaObiectiv_v
continua=(fval_cp>epsilon & contor(1,7)<30);
contor(1,7)=contor(1,7)+1
end;
solutie=[fi_c psi_c a1_c a2_c R_c]

```