

Parametarsko modeliranje cilindričnog zupčanika s evolventnim ozubljenjem

Rastija, Marko

Master's thesis / Diplomski rad

2017

Degree Grantor / Ustanova koja je dodijelila akademski / stručni stupanj: **University of Zagreb, Faculty of Mechanical Engineering and Naval Architecture / Sveučilište u Zagrebu, Fakultet strojarstva i brodogradnje**

Permanent link / Trajna poveznica: <https://urn.nsk.hr/urn:nbn:hr:235:155392>

Rights / Prava: [In copyright](#)/[Zaštićeno autorskim pravom.](#)

Download date / Datum preuzimanja: **2024-07-14**

Repository / Repozitorij:

[Repository of Faculty of Mechanical Engineering and Naval Architecture University of Zagreb](#)



SVEUČILIŠTE U ZAGREBU
FAKULTET STROJARSTVA I BRODOGRADNJE

DIPLOMSKI RAD

Marko Rastija

Zagreb, 2017.

SVEUČILIŠTE U ZAGREBU
FAKULTET STROJARSTVA I BRODOGRADNJE

DIPLOMSKI RAD

Mentori:

Doc.dr.sc. Krešimir Vučković, dipl.ing.

Student:

Marko Rastija

Zagreb, 2017.

Izjavljujem da sam ovaj rad izradio samostalno koristeći znanja stečena tijekom studija i navedenu literaturu.

Zahvaljujem mentoru doc.dr.sc. Krešimiru Vučkoviću na strpljenju, susretljivosti, razumijevanju, a nadasve korisnim savjetima i vodstvu tijekom izrade ovog diplomskog rada.

Zahvaljujem i svojoj obitelji, roditeljima, bratu i sestrama, djedu i baki i svima ostalima bez čije ljubavi i podrške ne bih uspio završiti ovaj studij.

Veliko hvala i mojoj djevojci Ivni za potporu i ljubav kojom me pratila tijekom svih ovih godina.

Marko Rastija



SVEUČILIŠTE U ZAGREBU
FAKULTET STROJARSTVA I BRODOGRADNJE



Središnje povjerenstvo za završne i diplomske ispite
Povjerenstvo za diplomske ispite studija strojarstva za smjerove:
procesno-energetski, konstrukcijski, brodstrojarski i inženjersko modeliranje i računalne simulacije

Sveučilište u Zagrebu Fakultet strojarstva i brodogradnje	
Datum 13 - 07 - 2017	Prilog
Klasa: 602-04/17-6/4	
Ur. broj: 15-1703-17-279	

DIPLOMSKI ZADATAK

Student: **Marko Rastija** Mat. br.: 0035183776

Naslov rada na hrvatskom jeziku: **Parametarsko modeliranje cilindričnog zupčanika s evolventnim ozubljenjem**

Naslov rada na engleskom jeziku: **Parametric Modelling of Spur Gear with Involute Teeth**

Opis zadatka:

Da bi se odredilo najveće lokalno glavno naprezanje u korijenu zuba zupčanika primjenom metode konačnih elemenata (MKE) najprije je potrebno izraditi dvodimenzijski model zupčanika u čeonj ravnini na osnovu kojeg se, po potrebi, može izraditi i trodimenzijski model. Jednadžbe koje opisuju geometriju profila zuba zupčanika dobivenog odvaljivanjem alata u obliku ravne ozubnice navedene su u dodatku norme ISO 6336-2:2006.

U radu je potrebno izraditi programsku proceduru za parametarsko modeliranje dvodimenzijskog modela zupčanika u čeonj ravnini u sklopu programskog paketa AutoCAD. Pri tom je potrebno predvidjeti mogućnost modeliranja cilindričnog zupčanika s evolventnim ozubljenjem:

- ravnim simetričnim,
- ravnim asimetričnim,
- kosim (helikoidnim) simetričnim i
- kosim (helikoidnim) asimetričnim zubima.

Parametri koje je potrebno obuhvatiti programskom procedurom su:

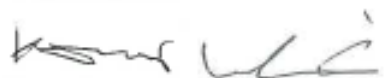
- normalni modul,
- broj zubi,
- faktor pomaka profila,
- kut nagiba boka zuba te
- profil ozubljenja.


Kod izbora profila ozubljenja omogućiti prethodni odabir profila prema normi ISO 53:1998 te zadavanje profila ozubljenja koji nisu obuhvaćeni navedenom normom.

U radu je potrebno navesti korištenu literaturu i eventualno dobivenu pomoć.

Zadatak zadan: Rok predaje rada: Predviđeni datum obrane:
11. svibnja 2017. 13. srpnja 2017. 19., 20. i 21. srpnja 2017.

Zadatak zadao: Predsjednica Povjerenstva:


Doc. dr. sc. Krešimir Vučković


Prof. dr. sc. Tanja Jurčević Lulić

SADRŽAJ

SADRŽAJ	I
POPIS SLIKA	II
POPIS TABLICA.....	III
POPIS OZNAKA	IV
SAŽETAK.....	VIII
SUMMARY	IX
1. UVOD.....	1
2. EVOLVENTNO OZUBLJENJE	3
2.1. Izrada evolventnog ozubljenja pomoću alata s ravnim bokom.....	6
2.2. Matematički opis evolvente i prijelazne krivulje.....	10
3. ZUPČANIK	17
3.1. Osnovni profil ozubljenja.....	17
3.2. Karakteristične mjere zupčanika	20
3.3. Zupčanik s kosim (helikoidnim) zubima	21
3.4. Zupčanik s asimetričnim zubima	24
3.5. Pomak profila.....	27
3.6. Zahvat zupčanika	28
4. PROGRAMSKA PROCEDURA	32
4.1. Učitavanje programske procedure	32
4.2. Modeliranje dvodimenzijskog modela zupčanika u čeonj ravni.....	33
4.3. Modeliranje zahvata zupčanika u čeonj ravni.....	40
4.3.1. Zahvat u točki C.....	42
4.3.2. Zahvat u točki A.....	43
4.3.3. Zahvat u točki E	45
4.3.4. Zahvat u točki B.....	46
4.3.5. Zahvat u točki D.....	47
5. TRODIMENZIJSKI MODEL ZUPČANIKA	49
6. ZAKLJUČAK.....	51
LITERATURA.....	52
PRILOZI.....	53

POPIS SLIKA

Slika 1.	Evolventa i evoluta.....	3
Slika 2.	Izvod evolventne funkcije	4
Slika 3.	Odvalno glodanje pužnim glodalom [1].....	6
Slika 4.	Izrada ozubljenja alatom u obliku ozubnice [1]	7
Slika 5.	Profil uzubine ozubnice [3]	7
Slika 6.	Oblikovanje zuba ozubnicom [3]	8
Slika 7.	Podrezivanje zuba [3].....	9
Slika 8.	Izvod jednadžbe evolvente [1]	10
Slika 9.	Produžena evolventa [1].....	12
Slika 10.	Evolventa i trohoida u polarnom koordinatnom sustavu [4].....	13
Slika 11.	Parametar trohoide [4].....	14
Slika 12.	Osnovni profil ozubljenja [5]	17
Slika 13.	Ozubnica [1].....	18
Slika 14.	Karakteristične mjere zupčanika	20
Slika 15.	Nastajanje zupčanika s kosim zubima [2]	22
Slika 16.	Normalna i čeona ravnina [1].....	23
Slika 17.	Asimetrični zub [9].....	25
Slika 18.	Asimetrični zubi u zahvatu [7]: a) prijenosnik generatora, b) pumpa.....	26
Slika 19.	Zupčanik prvog stupnja planetarnog reduktora motora TV7 – 117 [8]	26
Slika 20.	Debljina zuba kod pomaka profila [2].....	27
Slika 21.	Detalj zahvata zupčanika	28
Slika 22.	Korisničko sučelje pri modeliranju zupčanika	33
Slika 23.	Kut simetrale simetričnog zuba	35
Slika 24.	Položaj simetričnog zuba bez pomaka za kut γ	36
Slika 25.	Položaj simetričnog zuba nakon pomaka za kut γ	36
Slika 26.	Asimetrični zub	37
Slika 27.	Zupčanik s asimetričnim zubima.....	38
Slika 28.	Zupčanik sa simetričnim nepodrežanim zubima	39
Slika 29.	Zupčanik sa simetričnim podrežanim zubima.....	39
Slika 30.	Podrežani simetrični zub	40
Slika 31.	Korisničko sučelje za modeliranje zahvata zupčanika	41
Slika 32.	Točka zahvata A	43
Slika 33.	Točka zahvata E	45
Slika 34.	Zahvat simetričnih zubi u točki A	47
Slika 35.	Zahvat asimetričnih zubi u točki C.....	48
Slika 36.	Zupčanik sa simetričnim kosim zubima	49
Slika 37.	Zupčanik s asimetričnim ravnim zubima	50

POPIS TABLICA

Tablica 1. Standardne vrijednosti veličina osnovnog profila	19
Tablica 2. Tipovi osnovnog profila	19

POPIS OZNAKA

Oznaka	Jedinica	Opis
A	-	Oznaka točke
A	-	Početna točka zahvata zupčanika
A	-	Pomoćni faktor u jednadžbi trohoide
a	mm	Osni razmak zupčanika
B	-	Točka zahvata zupčanika
B	-	Pomoćni faktor u jednadžbi trohoide
B_{an}	-	Pomoćni faktor u jednadžbi trohoide
C	-	Oznaka točke
C	-	Kinematski pol
c_p	mm	Tjemena zračnost
D	-	Oznaka točke
d	mm	Diobeni promjer zupčanika
D	-	Točka zahvata zupčanika
d_a	mm	Promjer tjemene kružnice zupčanika
d_{a1}	mm	Promjer tjemene kružnice pogonskog zupčanika
d_{a2}	mm	Promjer tjemene kružnice gonjenog zupčanika
d_b	mm	Promjer temeljne kružnice zupčanika
d_{b1}	mm	Promjer temeljne kružnice pogonskog zupčanika
d_{b2}	mm	Promjer temeljne kružnice gonjenog zupčanika
d_f	mm	Promjer podnožne kružnice zupčanika
d_w	mm	Promjer kinematske kružnice zupčanika
d_{w1}	mm	Promjer kinematske kružnice pogonskog zupčanika
d_{w2}	mm	Promjer kinematske kružnice gonjenog zupčanika
E	-	Završna točka zahvata zupčanika
e_p	mm	Širina uzubine osnovnog profila na srednjoj liniji
h	mm	Udaljenost točke od generatrise
h_{aP}	mm	Tjemena visina zuba osnovnog profila
h_{aP}^*	-	Faktor tjemene visine zuba osnovnog profila
h_{fP}	mm	Podnožna visina zuba osnovnog profila
h_{fP}^*	-	Faktor podnožne visine zuba osnovnog profila
h_P	mm	Visina zuba osnovnog profila
h_{wP}	mm	Radna visina zuba osnovnog profila
M	-	Oznaka točke
m	mm	Modul

m_n	mm	Normalni modul
m_t	mm	Čeoni modul
N	-	Oznaka točke
O_1	-	Središte pogonskog zupčanika
O_2	-	Središte gonjenog zupčanika
p	mm	Korak osnovnog profila
p_n	mm	Korak u normalnoj ravnini
p_t	mm	Korak u čeonoj ravnini
r	mm	Polumjer pri izvodu jednadžbe evolvente
r_A	mm	Polumjer točke pogonskog zupčanika za zahvat u A
r_{a1}	mm	Polumjer tjemene kružnice pogonskog zupčanika
r_{a2}	mm	Polumjer tjemene kružnice gonjenog zupčanika
r_{Ax}	mm	Pomoćni polumjer točke pogonskog zupčanika za zahvat u A
r_b	mm	Polumjer temeljne kružnice
r_E	mm	Polumjer točke gonjenog zupčanika za zahvat u A
r_{Ex}	mm	Pomoćni polumjer točke gonjenog zupčanika za zahvat u A
r_{inv}	mm	Polarna koordinata/polumjer točke evolvente
r_{soi}	mm	Polumjer početka evolvente
r_{tro}	mm	Polarna koordinata/polumjer točke trohoide
s	mm	Širina zuba zupčanika na diobenom promjer
s_p	mm	Širina zuba osnovnog profila na srednjoj liniji
s_t	mm	Širina zuba zupčanika u čeonoj ravnini
T_1	-	Točka dodira temeljne kružnice i zahvatne crte pogonskog zupčanika
T_2	-	Točka dodira temeljne kružnice i zahvatne crte gonjenog zupčanika
x	-	Koordinata u pravokutnom koodrinatnom sustavu
x	-	Faktor pomaka profila
x_1	-	Faktor pomaka profila pogonskog zupčanika
x_2	-	Faktor pomaka profila gonjenog zupčanika
x_C	-	Koordinata točke <i>COAST</i> boka pogonskog zupčanika
x_{C1}	-	Koordinata točke <i>COAST</i> boka gonjenog zupčanika
x_D	-	Koordinata točke <i>DRIVE</i> boka pogonskog zupčanika
x_{D1}	-	Koordinata točke <i>DRIVE</i> boka gonjenog zupčanika
y	-	Koordinata u pravokutnom koodrinatnom sustavu
y_C	-	Koordinata točke <i>COAST</i> boka pogonskog zupčanika
y_{C1}	-	Koordinata točke <i>COAST</i> boka gonjenog zupčanika
y_D	-	Koordinata točke <i>DRIVE</i> boka pogonskog zupčanika
y_{D1}	-	Koordinata točke <i>DRIVE</i> boka gonjenog zupčanika
z	-	Broj zubi zupčanika
z_1	-	Broj zubi pogonskog zupčanika

z_2	-	Broj zubi gonjenog zupčanika
α	°	Kut zahvatne crte
α	°	Kut pri izvodu evolventne funkcije
α_n	°	Kut zahvatne crte u normalnoj ravnini
α_P	°	Kut boka zuba osnovnog profila
α_t	°	Kut zahvatne crte u čeonj ravnini
α_{tw}	°	Pogonski kut zahvatne crte u čeonj ravnini
α_w	°	Pogonski kut zahvatne crte
β	°	Kut nagiba boka zuba
β_b	°	Kut nagiba boka zuba na temeljnoj kružnici
γ	rad	Kut simetrale zuba
γ_1	rad	Kut simetrale pogonskog zuba
γ_2	rad	Kut simetrale gonjenog zuba
δ	°	Kut u izvodu evolventne funkcije
ε	-	Pomoćni faktor u jednadžbi trohoide
ζ_A	rad	Pomoćni kut točke gonjenog zupčanika za zahvat u A
ζ_E	rad	Pomoćni kut točke gonjenog zupčanika za zahvat u E
η_{inv}	rad	Polarna koordinata/kut točke evolvente
$\eta_{inv,d}$	rad	Kut točke evolvente na diobenoj kružnici
η_{tro}	rad	Polarna koordinata/kut točke trohoide
θ	°	Kut pri izvodu jednadžbe evolvente
θ	-	Pomoćni faktor u jednadžbi trohoide
ϑ_A	rad	Pomoćni kut točke pogonskog zupčanika za zahvat u A
ϑ_E	rad	Pomoćni kut točke pogonskog zupčanika za zahvat u E
ν	rad	Kut luka poluširine zuba na diobenoj kružnici
ζ	-	Parametar evolvente
ζ_{A1}	-	Parametar evolvente za točku A zahvata pogonskog zupčanika
ζ_{A2}	-	Parametar evolvente za točku A zahvata gonjenog zupčanika
ζ_{C1}	-	Parametar evolvente za točku C zahvata pogonskog zupčanika
ζ_{C2}	-	Parametar evolvente za točku C zahvata gonjenog zupčanika
ζ_d	-	Parametar evolvente za točku na diobenoj kružnici
ζ_{E1}	-	Parametar evolvente za točku E zahvata pogonskog zupčanika
ζ_{E2}	-	Parametar evolvente za točku E zahvata gonjenog zupčanika
ρ_{IP}	mm	Polumjer zakrivljenja u podnožju zuba osnovnog profila
ρ_{IP}^*	-	Faktor polumjera zakrivljenja u podnožju zuba
Φ	°	Kut rotacije generatriše
φ	-	Parametar trohoide

ω_{A1}	rad	Kut zakreta pogonskog zupčanika za točku A zahvata
ω_{A2}	rad	Kut zakreta gonjenog zupčanika za točku A zahvata
ω_{B1}	rad	Kut zakreta pogonskog zupčanika za točku B zahvata
ω_{B2}	rad	Kut zakreta gonjenog zupčanika za točku B zahvata
ω_{C1}	rad	Kut zakreta pogonskog zupčanika za točku C zahvata
ω_{C2}	rad	Kut zakreta gonjenog zupčanika za točku C zahvata
ω_{D1}	rad	Kut zakreta pogonskog zupčanika za točku D zahvata
ω_{D2}	rad	Kut zakreta gonjenog zupčanika za točku D zahvata
ω_{E1}	rad	Kut zakreta pogonskog zupčanika za točku E zahvata
ω_{E2}	rad	Kut zakreta gonjenog zupčanika za točku E zahvata

SAŽETAK

U ovom radu izrađena je programska procedura za parametarsko modeliranje dvodimenzijskog modela zupčanika s evolventnim ozubljenjem u čeonj ravni. Za izradu programske procedure korišten je VBA modul u sklopu programskog paketa *AutoCAD 2014*. Opisan je proces izrade programske procedure uz teorijski opis evolventnog ozubljenja i objašnjenje pojedinih pojmova vezanih uz teoriju i primjenu zupčanika. Programskom procedurom je omogućeno dvodimenzijsko parametarsko modeliranje cilindričnog zupčanika s ravnim i kosim, simetričnim i asimetričnim zubima. Parametri obuhvaćeni programskom procedurom su normalni modul, broj zubi, faktor pomaka profila, kut nagiba boka zuba i profil ozubljenja.

Ključne riječi: zupčanik, evolventno ozubljenje, parametarsko modeliranje, dvodimenzijski model, asimetričan zub, osnovni profil ozubljenja

SUMMARY

In this thesis a procedure for parametric modelling of two-dimensional model of gear with involute teeth in transverse plain was programmed. The procedure was programmed in VBA module of *AutoCAD* 2014 software. Description of programming process is given, as well as explanation of involute gearing and certain terms and concepts of gear theory and application. Procedure is programmed for two-dimensional parametric modeling of spur and helical gears with symmetric or asymmetric teeth. Parameters used by procedure are normal module, number of teeth, coefficient of profile shift, helix angle and basic rack parameters.

Key words: gear, involute gearing, parametric modelling, two-dimensional model, asymmetric tooth, basic rack

1. UVOD

Zupčanici su najkorišteniji strojni elementi za prijenos snage i gibanja. S obzirom da prijenos snage i gibanja ostvaruju zahvatom posebno oblikovanih zubi, dakle pomoću veze oblikom, odlikuje ih visoka točnost prijenosa i stalan, točno određen prijenosni omjer. Takva krutost pogona je i nedostatak jer uzrokuje vibracije i buku, a prenosi i udarna opterećenja bez ublažavanja. U nedostatke se mogu ubrojiti i visoka cijena zbog potrebne visoke točnosti izrade. Prednosti zupčanih prijenosnika su visoki stupanj iskoristivosti, široko radno područje (od najmanjih do najvećih brzina i snaga), male dimenzije u odnosu na snagu koja se prenosi i velika trajnost pogona.

Težnja za prijenosom što veće snage uz istovremeno smanjenje dimenzija i mase samih zupčanika i zupčanih prijenosnika uzrokuje znatno povećanje naprezanja boka i korijena zuba. Bok zuba opterećen je dodirnim pritiskom, a korijen zuba savijanjem. Budući da zubi tijekom rada periodično ulaze u zahvata i iz njega izlaze, naprezanja koja se javljaju ciklički osciliraju. Analitički se proračun zupčanika provodi prema određenoj normi (DIN, ISO...), a osim na principima mehanike i čvrstoće, temelji se i na nizu eksperimentalno određenih faktora koji uzimaju u obzir specifičan oblik, način i raspodjelu opterećenja zuba zupčanika.

Provedba eksperimenta je skupa, složena, a često i dugotrajna. Svaka značajnija promjena u konstrukciji zupčanika iziskuje novi eksperiment kojim se utvrđuje utjecaj takve promjene na naprezanje zuba zupčanika. Razvoj računalne tehnologije, kako fizičkih komponenti računala (procesor, memorija, grafičke komponente) tako i programskih paketa, omogućava provedbu sve složenijih numeričkih proračuna i analiza. Provedba numeričke analize na računalu jeftinija je od eksperimenta, omogućava jednostavnu promjenu parametara ili konstrukcije zupčanika i ponovnu provedbu analize bez potrebe za ponovnim postavljanjem ispitnog postava i opreme što rezultira dobivanjem većeg broja podataka i rezultata u kraćem vremenu. Za provedbu numeričke analize, najčešće pomoću metode konačnih elemenata (MKE), potrebno je izraditi trodimenzijski geometrijski model zupčanika. Temelj svakog takvog trodimenzijskog modela je dvodimenzijski model samog zupčanika u čeonj ravnini. Kako bi se dobili što točniji rezultate same analize, nužno je da geometrijski model zupčanika bude što

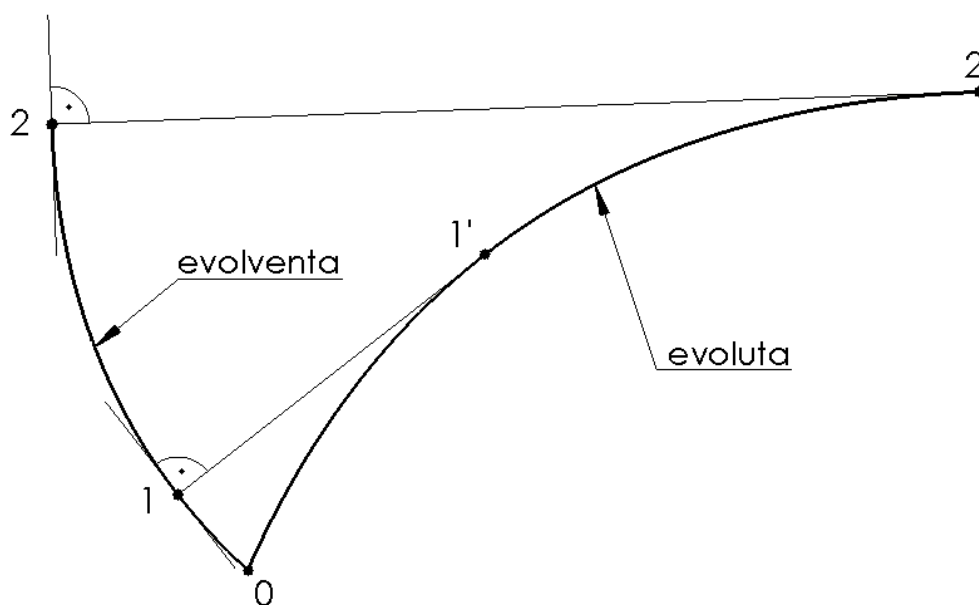
sličniji stvarnom zupčaniku. Budući da je profil zuba zupčanika definiran točno određenim krivuljama koje je moguće matematički opisati, za izradu što točnijeg dvodimenzijskog modela zupčanika u čeonj ravnini potrebno je te krivulje što točnije modelirati. Matematičke formule omogućuju analitički izračun koordinata točaka krivulja koje opisuju zub zupčanika, što onda omogućuje i njihovo precizno modeliranje, ali takav postupak je dugotrajan i zamoran, podložan greškama, a za svaku se promjenu parametara zupčanika postupak mora ponoviti. To značajno produžuje proces samog modeliranja geometrijskih modela zupčanika, a time i cijele numeričke analize. Postojanje programske procedure koja bi automatski crtala tražene krivulje, odnosno sam profil zuba i u konačnici cijeli zupčanik, na osnovi matematičkih formula i zadanih parametara zupčanika (modul, broj zubi, kut nagiba boka, pomak profila itd.) uvelike bi olakšala sam proces modeliranja i ubrzala cijeli proces numeričke analize, uz postizanje tražene preciznosti i točnosti. Izrada takve programske procedure za parametarsko modeliranje dvodimenzijskog modela zupčanika u čeonj ravnini tema je ovog diplomskog rada.

2. EVOLVENTNO OZUBLJENJE

U ovom diplomskom radu razmatrana je geometrija profila zuba zupčanika s evolventnim ozubljenjem jer je to najzastupljenija vrsta ozubljenja. Prema [1] razlog za to je što evolventno ozubljenje posjeduje određene prednosti u odnosu na druge vrste ozubljenja, a to su:

- alati za izradu evolventnog ozubljenja mogu se proizvesti s velikom preciznošću
- moguće je mijenjati debljinu zuba i ostvariti nestandardni osni razmak promjenom položaja alata prilikom izrade zupčanika
- nestandardni (s pomakom profila) zupčanici mogu se proizvesti standardiziranim alatima koji se koriste i za proizvodnju standardnih zupčanika
- mala promjena osnog razmaka ne uzrokuje greške u prijenosu

Općenito, evolventa je ravninska krivulja koja je nerazdvojivo povezana sa svojom evolutom, ravninskom krivuljom koja predstavlja geometrijsko mjesto svih središta polumjera zakrivljenjaevolvente [Slika 1]. Normala na tangentu u bilo kojoj točki evolvente tangira evolutu [1].

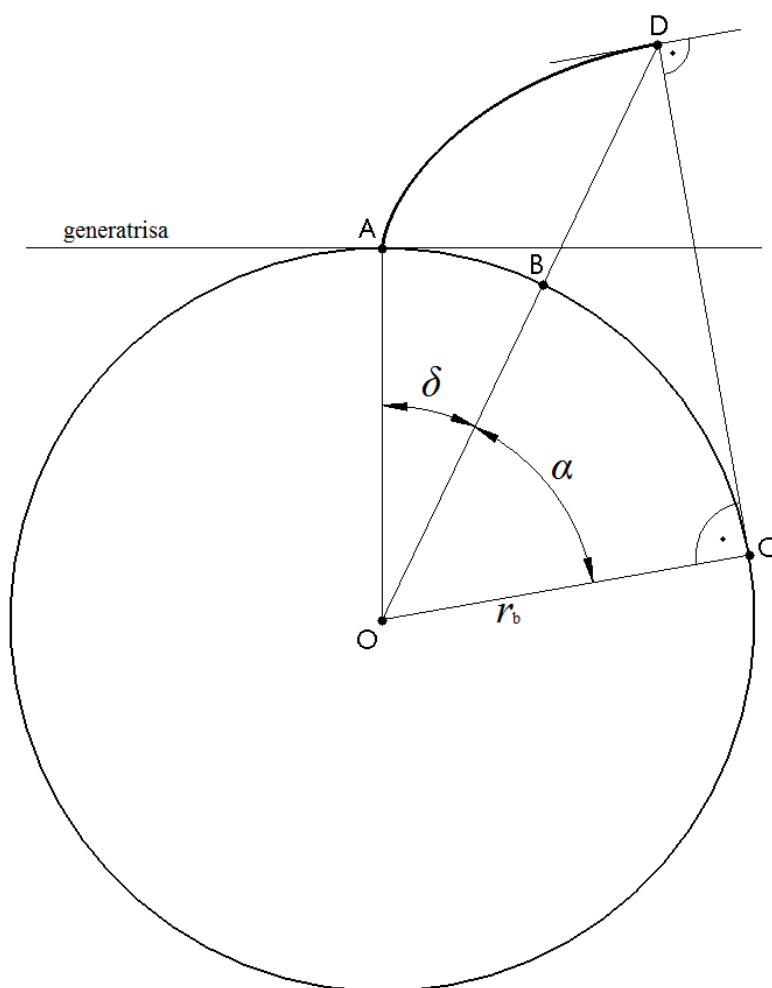


Slika 1. Evolventa i evoluta

Evolventnu krivulju opisuje svaka točka na pravcu koji se bez klizanja valja po evoluti. Nastajanje evolvente se prema [1] može slikovito prikazati na sljedeći način: zamislimo da je na evolutu čvrsto namotana nit koja na jednom svom kraju ima vezanu olovku. Kada se nit odmotava s evolute uz održavanje napetosti niti olovka će crtati evolventu.

Evolventa čija je evoluta kružnica opisuje profil zuba zupčanika s evolventnim ozubljenjem. Dakle, svaka točka na valjnom pravcu (generatrisi) koji se bez klizanja valja po obodu temeljne kružnice r_b opisuje evolventu koja predstavlja bok zuba zupčanika [2]. Kao što je ranije rečeno, temeljna kružnica kao evoluta predstavlja geometrijsko mjesto svih središta polumjera zakrivljenjaevolvente.

U svrhu izvoda evolventne funkcije promatrana je temeljna kružnica čiji polumjer iznosi $r_b = 1$ [Slika 2].



Slika 2. Izvod evolventne funkcije

Točka D odabrana je točka na evolventi. Točka C je točka na temeljnoj kružnici u kojoj normala na tangentu u točki D dodiruje temeljnu kružnicu. Točka B je točka sjecišta između temeljne kružnice i pravca koji spaja središte temeljne kružnice O i točku D. Točka A prva je točka evolvente i leži na temeljnoj kružnici. S obzirom da se generatrisa valja po temeljnoj kružnici bez klizanja lučna udaljenost točaka A i C jednaka je udaljenosti između točaka C i D:

$$\widehat{AC} = \overline{CD}. \quad (1)$$

Udaljenost između točaka C i D ujedno predstavlja i polumjer zakrivljenjaevolvente u točki D [2].

Evolventna funkcija kuta α jednaka je duljini luka iznad kuta δ :

$$ev\alpha = \widehat{AB} = r_b \cdot \widehat{\delta}. \quad (2)$$

Tražena duljina luka može se izračunati iz sljedeće jednadžbe:

$$\widehat{AB} = \widehat{AC} - \widehat{BC}. \quad (3)$$

Duljinu luka između točaka B i C može se izračunati pomoću izraza:

$$\widehat{BC} = r_b \cdot \widehat{\alpha}. \quad (4)$$

Duljina luka između točaka A i C izračunava se iz jednadžbe (1) i pravokutnog trokuta ΔOCD :

$$\widehat{AC} = r_b \cdot \tan \alpha. \quad (5)$$

Kada se jednadžbe (2), (4) i (5) uvrste u jednadžbu (3) dobiva se sljedeća jednakost:

$$r_b \cdot \widehat{\delta} = r_b \cdot \tan \alpha - r_b \cdot \widehat{\alpha}, \quad (6)$$

a kako je temeljna kružnica jedinična, odnosno $r_b = 1$ evolventna funkcija kuta α u konačnici glasi:

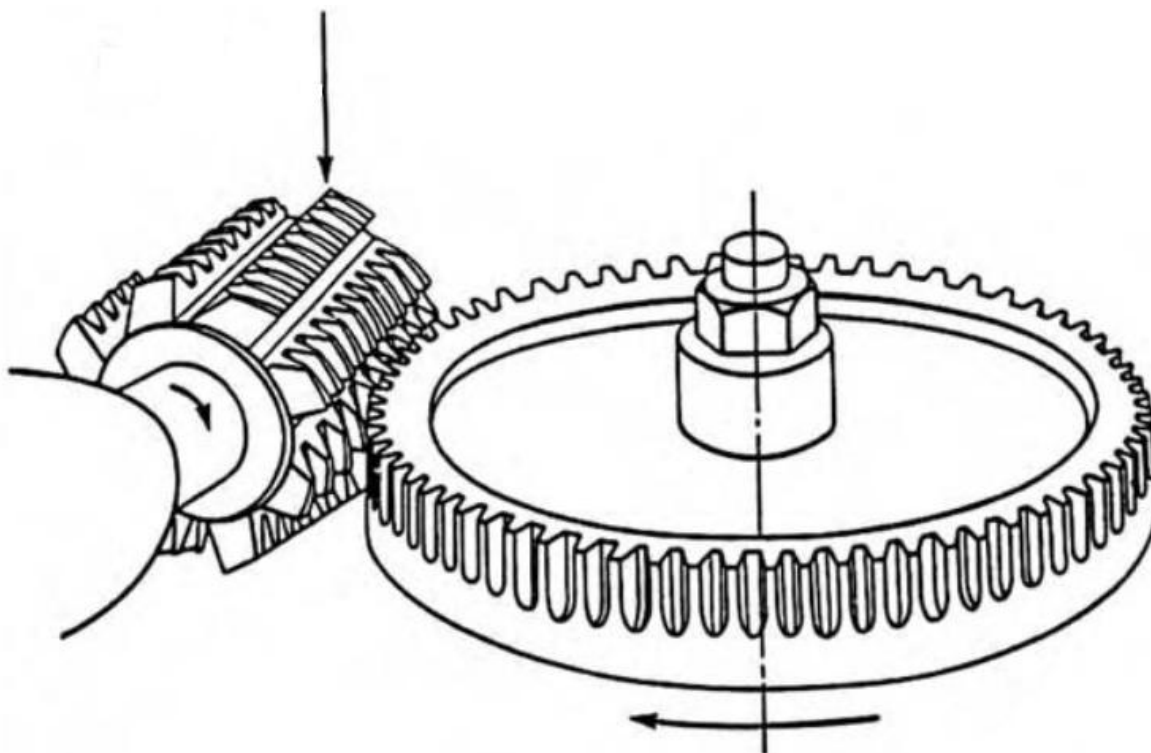
$$ev\alpha = \tan \alpha - \widehat{\alpha}. \quad (7)$$

2.1. Izrada evolventnog ozubljenja pomoću alata s ravnim bokom

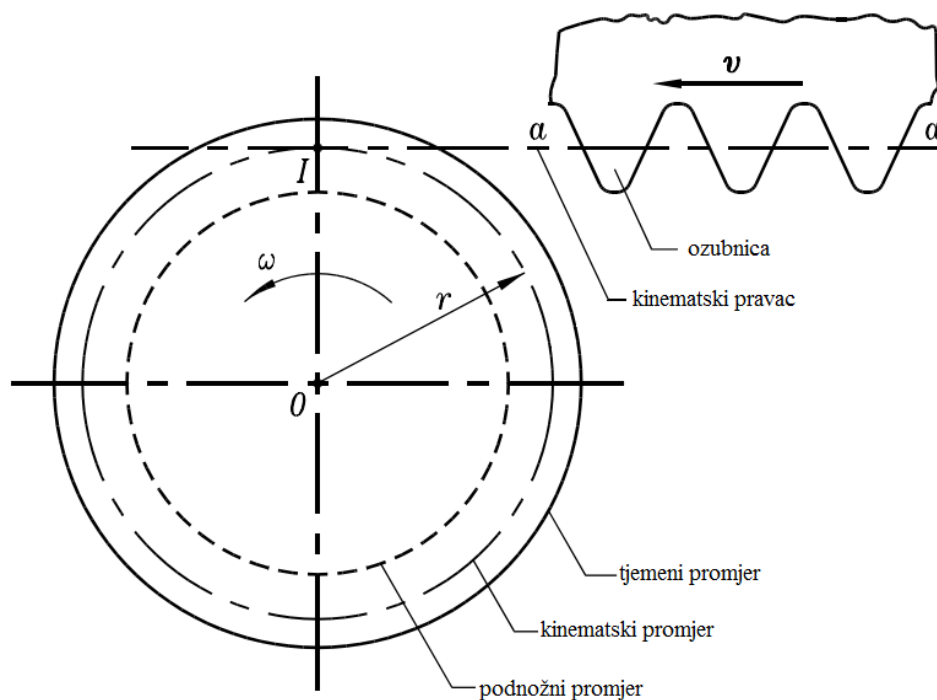
Zupčanike s evolventnim ozubljenjem moguće je izrađivati pomoću alata s ravnim bokovima, a to je najčešće alat u obliku ozubnice ili odvalnog pužnog glodala. Zupčanik se izrađuje tako da se alat u obliku ozubnice svojim kinematskim pravcem valja bez klizanja po kinematskoj kružnici zupčanika i pritom svojim ravnim bokom, oštricom, oblikuje bok zuba zupčanika [2]. Uzdužni poprečni presjek odvalnog pužnog glodala može se smatrati ozubnicom [1].

Tijekom procesa izrade zupčanika alatom u obliku ozubnice [Slika 4] izradak rotira oko svoje osi, a sam alat pomiče se pravocrtno u smjeru okomitom na os rotacije izratka, ali i naizmjenično pravocrtno u smjeru paralelnom s osi rotacije.

Prilikom izrade zupčanika pomoću odvalnog pužnog glodala [Slika 3] izradak rotira oko svoje osi, a glodalo oko svoje te zbog spiralnog rasporeda oštrica simulira pravocrtno gibanje ozubnice.

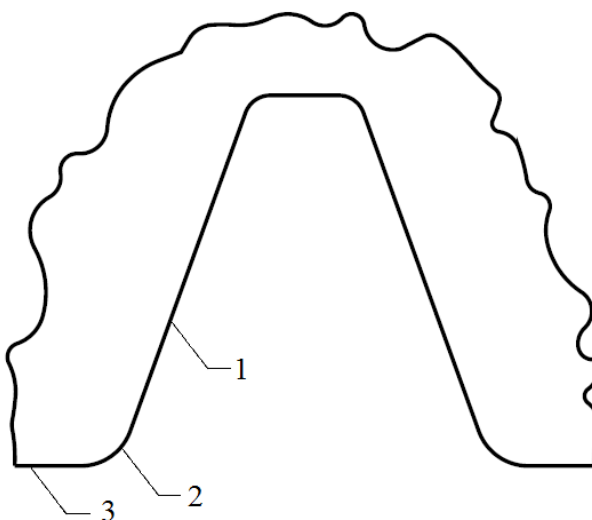


Slika 3. Odvalno glodanje pužnim glodalom [1]

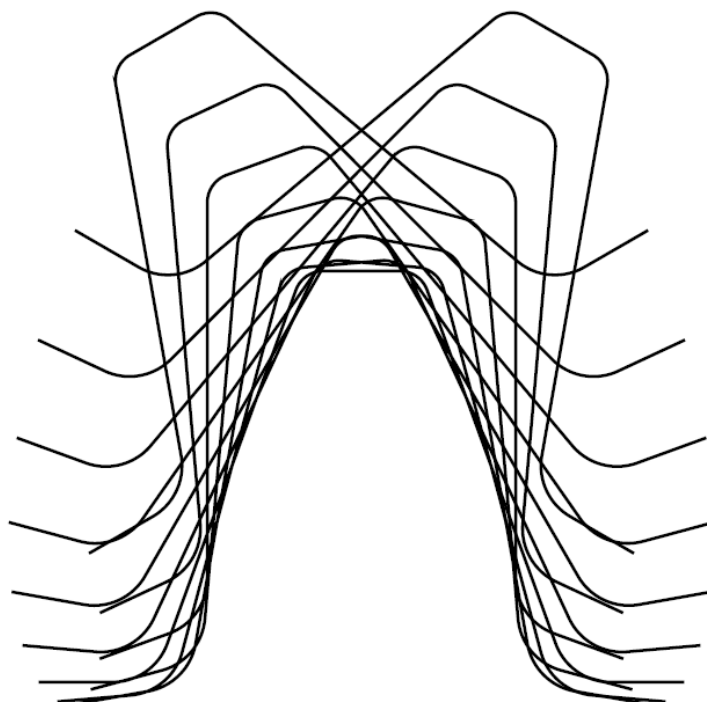


Slika 4. Izrada ozubljenja alatom u obliku ozubnice [1]

[Slika 5] prikazuje profil uzubine ozubnice koji se koristi za izradu evolventnog ozubljenja. Svaki dio ozubnice tijekom izrade zupčanika oblikuje određeni dio profila zuba [Slika 6] pa tako ravni bok (1) oblikuje glavni, evolventni dio profila zuba, a ravni tjemeni dio ozubnice (3) oblikuje podnožni dio zuba, odnosno luk podnožne kružnice na sredini uzubine. Zaobljeni dio ozubnice (2) tijekom izrade zupčanika oblikuje prijelaznu krivulju u korijenu zuba koja spaja evolventni dio zuba s podnožnom kružnicom.



Slika 5. Profil uzubine ozubnice [3]

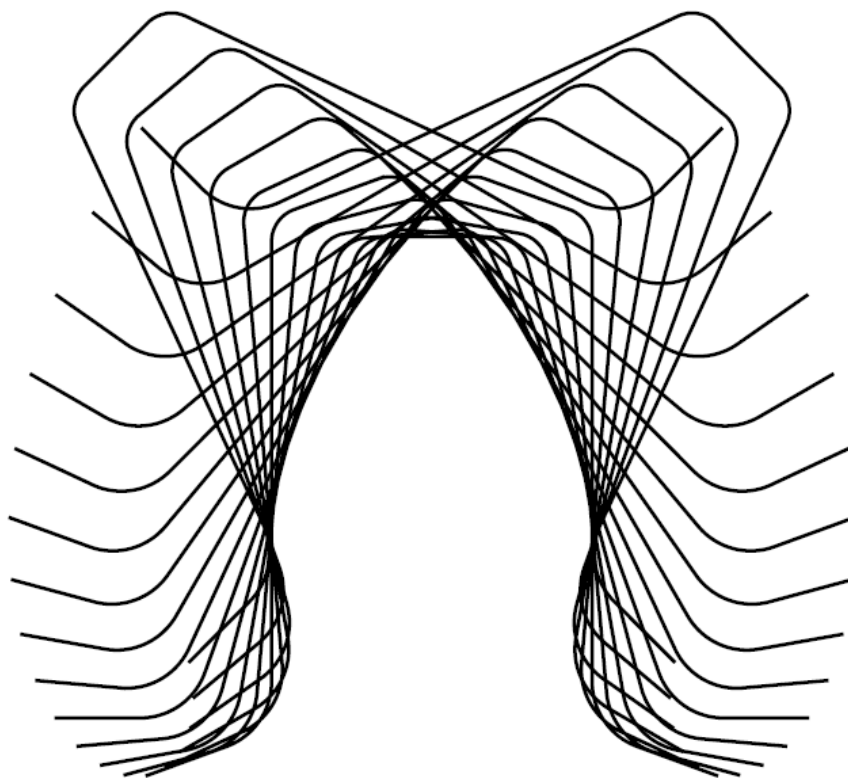


Slika 6. Oblikovanje zuba ozubnicom [3]

Ako je broj zubi zupčanika premali ili je ozubnica prilikom izrade zupčanika previše primaknuta središtu zupčanika dolazi do podrezivanja zuba. Naime, tada zaobljeni vrh ozubnice tijekom oblikovanja prijelazne krivulje između evolventnog i podnožnog dijela zuba ulazi preduboko u korijen zuba i podrezuje ga. Debljina se zuba u korijenu, koji je savijanjem najopterećeniji dio zuba, smanjuje [Slika 7], a prijelazna krivulja i evolventa boka zuba više nisu tangentne nego se sijeku pa se skraćuje korisni evolventni dio profila zuba čime se narušava ispravnost zahvata dvaju zupčanika. Iz navedenih razloga podrezivanje zuba nastoji se izbjeći.

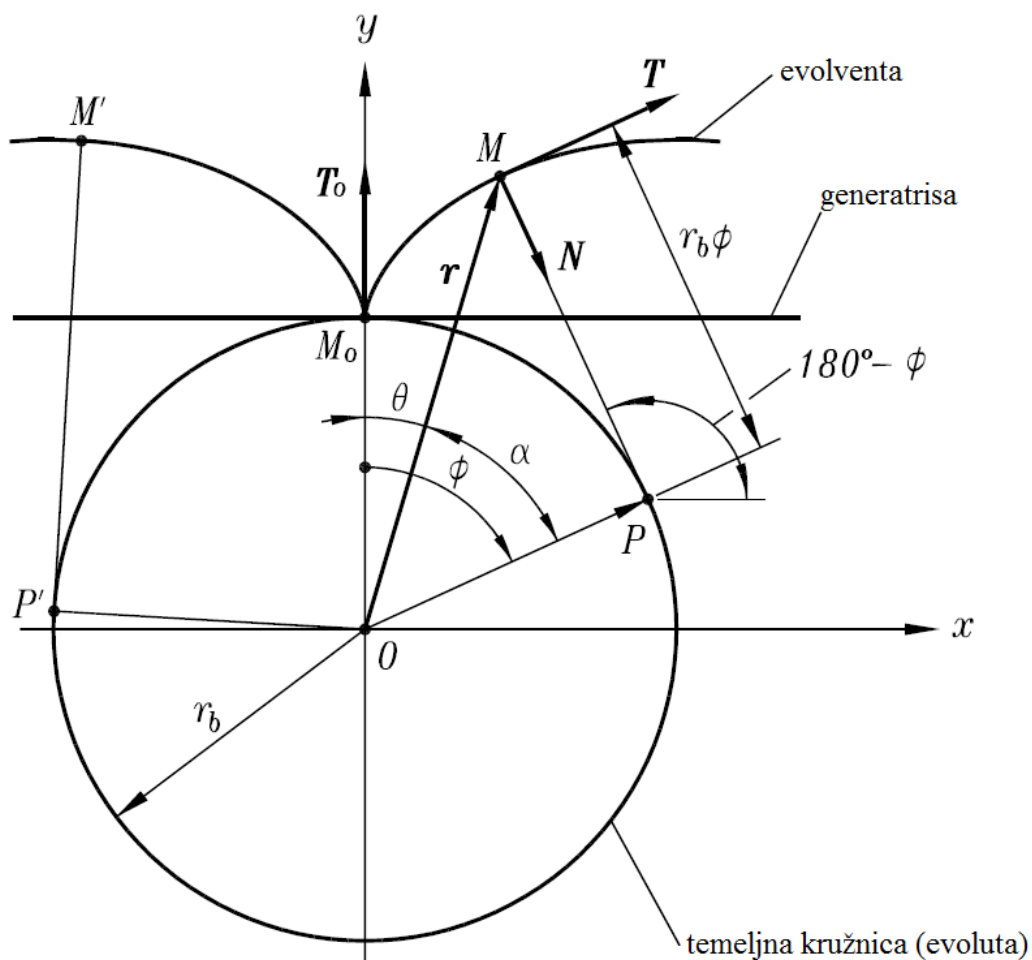
Podrežanost zuba moguće je izbjeći povećanjem broja zubi, povećanjem kuta zahvatne crte ili pomakom profila tj. odmicanjem ozubnice od središta zupčanika prilikom izrade pri čemu treba voditi računa da se debljina zuba na tjemenoj kružnici pretjerano ne smanji jer u tom slučaju zub postaje zašiljen.

Valja napomenuti da postoje i drugi postupci izrade zupčanika, ali u ovom je diplomskom radu razmatrana geometrija profila zuba zupčanika izrađenog pomoću alata s ravnim bokom.



Slika 7. Podrezivanje zuba [3]

2.2. Matematički opis evolvente i prijelazne krivulje



Slika 8. Izvod jednadžbe evolvente [1]

U [1] jednadžba evolvente [Slika 8] izvedena je na sljedeći način. Proizvoljna točka M evolvente određena je vektorskom jednadžbom:

$$\overline{OM} = \overline{OP} + \overline{PM}, \quad (8)$$

pri čemu su

$$\overline{OP} = r_b [\sin \phi \quad \cos \phi]^T, \quad (9)$$

$$\overline{PM} = PM [-\cos \phi \quad \sin \phi]^T. \quad (10)$$

Kako se generatrisa valja po obodu temeljne kružnice bez klizanja vrijedi:

$$\overline{PM} = \widehat{M_0P} = r_b \cdot \widehat{\phi}. \quad (11)$$

Kut Φ predstavlja kut rotacije generatriše. Uvrštavanjem jednadžbi (11), (10) i (9) u jednadžbu (8) dobivaju se jednadžbe za izračun koordinata točaka evolvente u Kartezijevom koordinatnom sustavu:

$$x = r_b \cdot (\sin \phi - \phi \cdot \cos \phi), \quad (12)$$

$$y = r_b \cdot (\cos \phi + \phi \cdot \sin \phi). \quad (13)$$

Prema [1] jednadžbe za izračun koordinata točaka evolvente u Kartezijevom koordinatnom sustavu moguće je izraziti u ovisnosti o kutu α [Slika 8] na sljedeći način:

$$x = r \cdot \sin \theta, \quad (14)$$

$$y = r \cdot \cos \theta, \quad (15)$$

pri čemu vrijede sljedeće jednakosti

$$r = \frac{r_b}{\cos \alpha}, \quad (16)$$

$$r \cdot (\widehat{\theta} + \widehat{\alpha}) = \widehat{M_0P}, \quad (17)$$

$$\widehat{M_0P} = \overline{MP}, \quad (18)$$

$$\overline{MP} = r_b \cdot \tan \alpha, \quad (19)$$

$$\theta = \tan \alpha - \widehat{\alpha}. \quad (20)$$

Prema ranijem izvodu u jednadžbama (1) do (7), jednadžba (20) predstavlja evolventnu funkciju kuta α pa se prema [1] uvrštavanjem jednadžbi (16) i (20) u jednadžbe (14) i (15) dobivaju jednadžbe za izračun koordinata točaka evolvente u ovisnosti o kutu α :

$$x = \frac{r_b}{\cos \alpha} \cdot \sin(\tan \alpha), \quad (21)$$

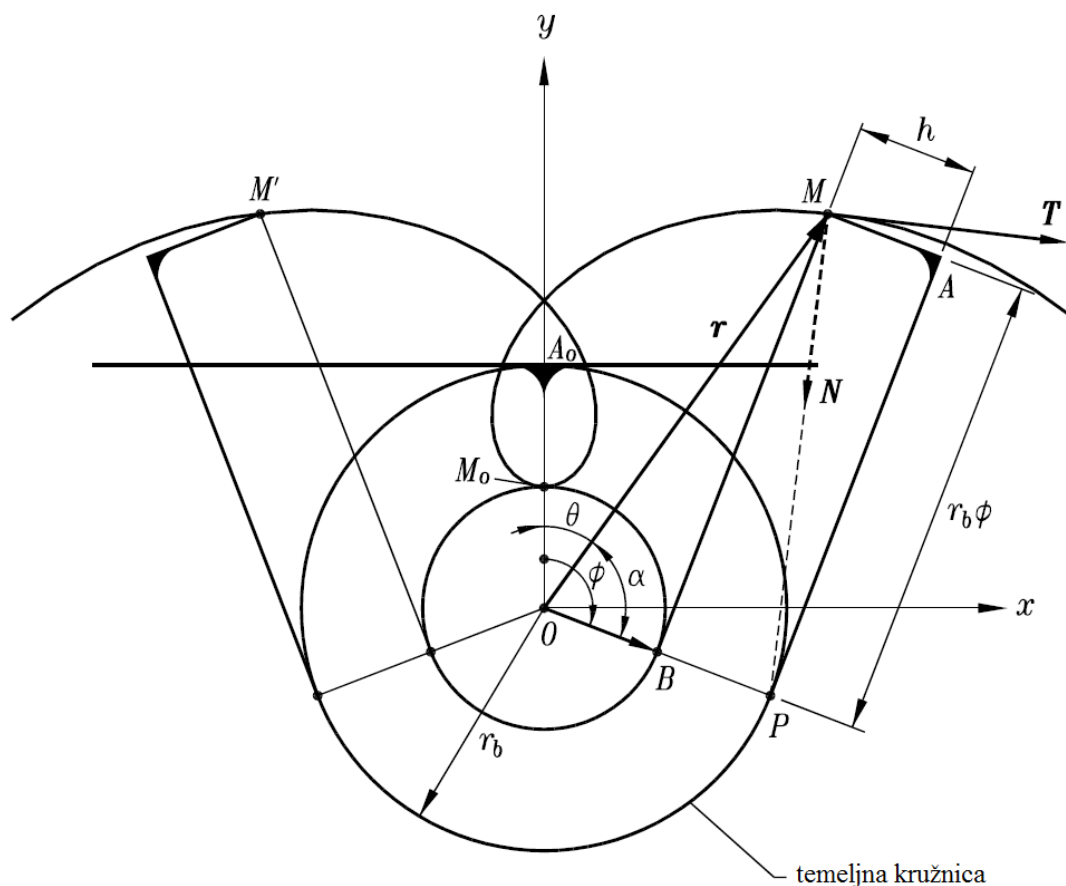
$$y = \frac{r_b}{\cos \alpha} \cdot \cos(\tan \alpha). \quad (22)$$

Prijelazna krivulja koja spaja evolventni dio boka zuba s podnožnom kružnicom opisuje se prema [1] kao produžena evolventa [Slika 9], a koordinate njenih točaka moguće je izračunati pomoću sljedećih jednadžbi:

$$x = (r_b - h) \cdot \sin \phi - r_b \cdot \phi \cdot \cos \phi, \quad (23)$$

$$y = (r_b - h) \cdot \cos \phi + r_b \cdot \phi \cdot \sin \phi \quad (24)$$

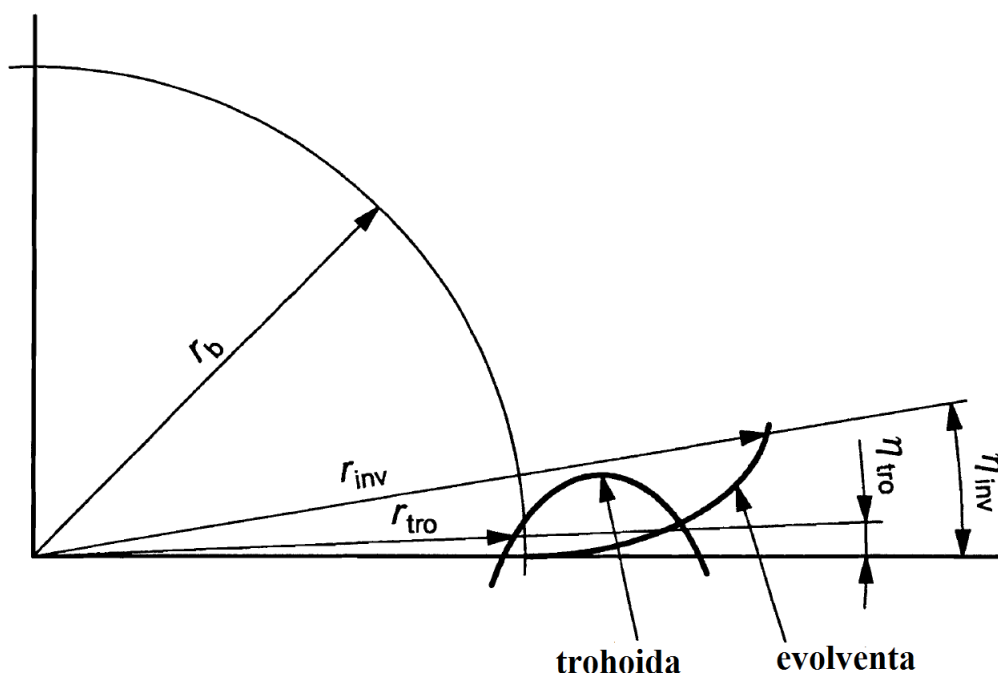
gdje je h udaljenost točke koja opisuje evolventu od pravca (generatriše) koji se valja po obodu temeljne kružnice.



Slika 9. Produžena evolventa [1]

Gore navedene jednadžbe omogućuju analitički izračun koordinata točaka evolvente i produžene evolvente kao prijelazne krivulje u Kartezijevom koordinatnom sustavu, ali ne uzimaju u obzir parametre zupčanika, osim polumjera temeljne kružnice. Također, ove jednadžbe ne omogućuju izravnomodeliranje cijelog profila zuba jer nisu izvedene imajući na umu putanju pojedinih dijelova alatai specifičan međusobni položaj evolvente i prijelazne

krivulje koji je posljedica takvog načina izrade. Zbog svega navedenog u ovom diplomskom radu za izradu programske procedure korištene su jednadžbe dane u [4].



Slika 10. Evolventa i trohoida u polarnom koordinatnom sustavu [4]

Jednadžbe su dane u polarnim koordinatama pa tako evolventu opisuju sljedeće jednadžbe u parametarskom obliku:

$$r_{\text{inv}} = \frac{d_b}{2} \cdot \sqrt{1 + \xi^2}, \quad (25)$$

$$\eta_{\text{inv}} = \xi - \arctan \xi \quad (26)$$

pri čemu je ξ parametar evolvente, a odgovara kutu rotacije generatriše (valjnog pravca).

Prema [4] prijelazna krivulja koja spaja evolventni dio zuba i podnožnikružnicu zuba jest trohoida koju opisuju sljedeće jednadžbe u parametarskom obliku

$$r_{\text{tro}} = \sqrt{\left(\frac{d}{2} - B\right)^2 + \left(\frac{B \cdot \cos \beta}{\tan \varphi}\right)^2}, \quad (27)$$

$$\eta_{\text{tro}} = \theta + \varepsilon - \alpha_t, \quad (28)$$

pri čemu su B , θ , ε i A pomoćni faktori određeni izrazima:

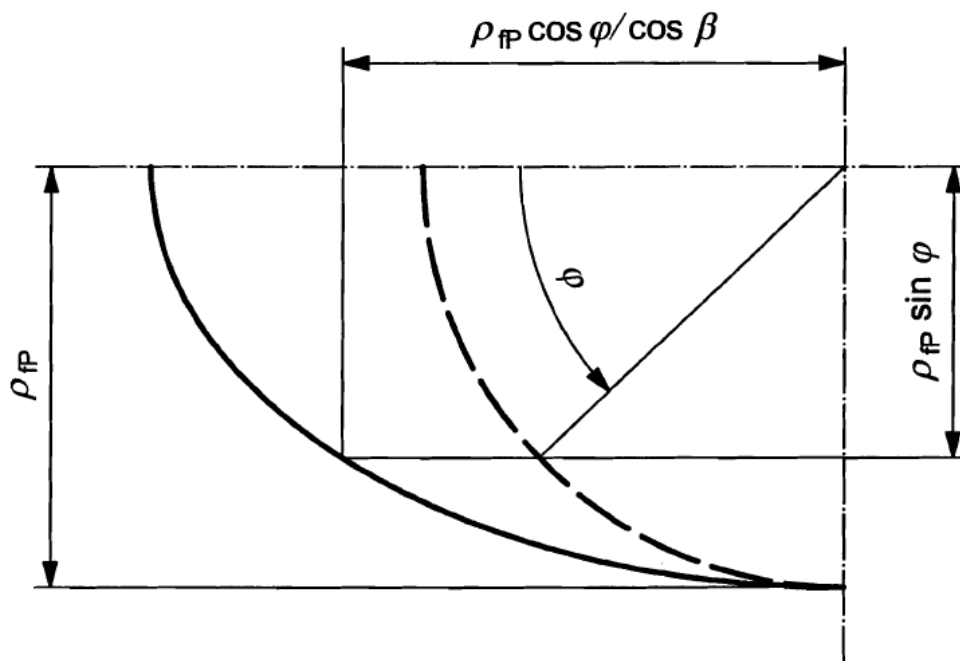
$$B = h_{\text{FP}} - x \cdot m_n - \rho_{\text{FP}} + \rho_{\text{FP}} \cdot \sin \varphi, \quad (29)$$

$$\theta = \tan \alpha_i + \frac{2}{d} \cdot \left(\rho_{\text{FP}} \cdot \frac{\cos \varphi}{\cos \beta} - A - B \cdot \frac{\cos \beta}{\tan \varphi} \right), \quad (30)$$

$$A = \frac{\rho_{\text{FP}}}{\cos \alpha_n \cdot \cos \beta} + (h_{\text{FP}} - x \cdot m_n - \rho_{\text{FP}}) \cdot \tan \alpha_i, \quad (31)$$

$$\varepsilon = \arctan \frac{B \cdot \cos \beta}{\left(\frac{d}{2} - B \right) \cdot \tan \varphi}, \quad (32)$$

dok je φ parametar trohoide definiran prema [Slika 11].



Slika 11. Parametar trohoide [4]

Za razliku od ranije izvedenih jednadžbi prema [1], gore navedene jednadžbe iz [4] uzimaju u obzir parametre zupčanika kao što su normalni modul, podnožna i tjemena visina zuba, kut

nagiba bok, faktor pomaka profila kao i polumjer zakrivljenja u podnožju zuba osnovnog profila. Osim samih jednadžbi evolvente i trohoide u polarnim koordinatama, u [4] su navedene i jednadžbe koje omogućuju određivanje polumjera točkeu kojoj se evolventa i trohoida spajaju ili sijeku, ovisno o tome je li zub podrezan ili nije. To omogućuje točno modeliranje profila zuba. Prema [4] zub neće biti podrezan ako vrijedi sljedeća nejednadžba:

$$\frac{d}{2} \cdot (\sin \alpha_t)^2 - [h_{\text{TP}} - x \cdot m_n - \rho_{\text{TP}} \cdot (1 - \sin \alpha_n)] \geq 0. \quad (33)$$

U tom slučaju, polumjer točke u kojoj se spajaju trohoida i evolventa, odnosno polumjer početka evolvente r_{soi} može se izračunati iz sljedećih jednadžbi:

$$B_{\alpha_n} = h_{\text{TP}} - x \cdot m_n - \rho_{\text{TP}} + \rho_{\text{TP}} \cdot \sin \alpha_n, \quad (34)$$

$$r_{\text{soi}} = r_{\text{tro}(\varphi=\alpha_n)} = \sqrt{\left[\frac{d}{2} - B_{\alpha_n}\right]^2 + \left[\frac{B_{\alpha_n}}{\tan \alpha_t}\right]^2}. \quad (35)$$

Ako je izraz (33) manji od nule, zub je podrezan i tom se slučaju evolventa i trohoida sijeku, a točka sjecišta određuje se iz sustava jednadžbi

$$r_{\text{inv}} = r_{\text{tro}}, \quad (36)$$

$$\eta_{\text{inv}} = \eta_{\text{tro}}. \quad (37)$$

Prema [4] iz jednadžbe (36) moguće je izraziti parametar evolvente ξ kao funkciju parametra trohoide φ na sljedeći način:

$$\xi(\varphi) = \sqrt{\left(\frac{d/2 - B(\varphi)}{d_b/2}\right)^2 + \left(\frac{B(\varphi) \cdot \cos \beta}{(d_b/2) \cdot \tan \varphi}\right)^2} - 1. \quad (38)$$

Tada se uvrštavanjem jednadžbe (38) u jednadžbu (37) dobiva izraz

$$\theta(\varphi) + \varepsilon(\varphi) - \xi(\varphi) + \arctan \xi(\varphi) - \alpha = 0. \quad (39)$$

Rješenje jednadžbe (39) je vrijednost parametra trohoide φ u sjecištu trohoide i evolvente, pomoću kojeg je moguće izračunati polumjer i kut početka evolvente, r_{soi} i η_{soi} , za slučaj kada je korijen zuba podrezan.

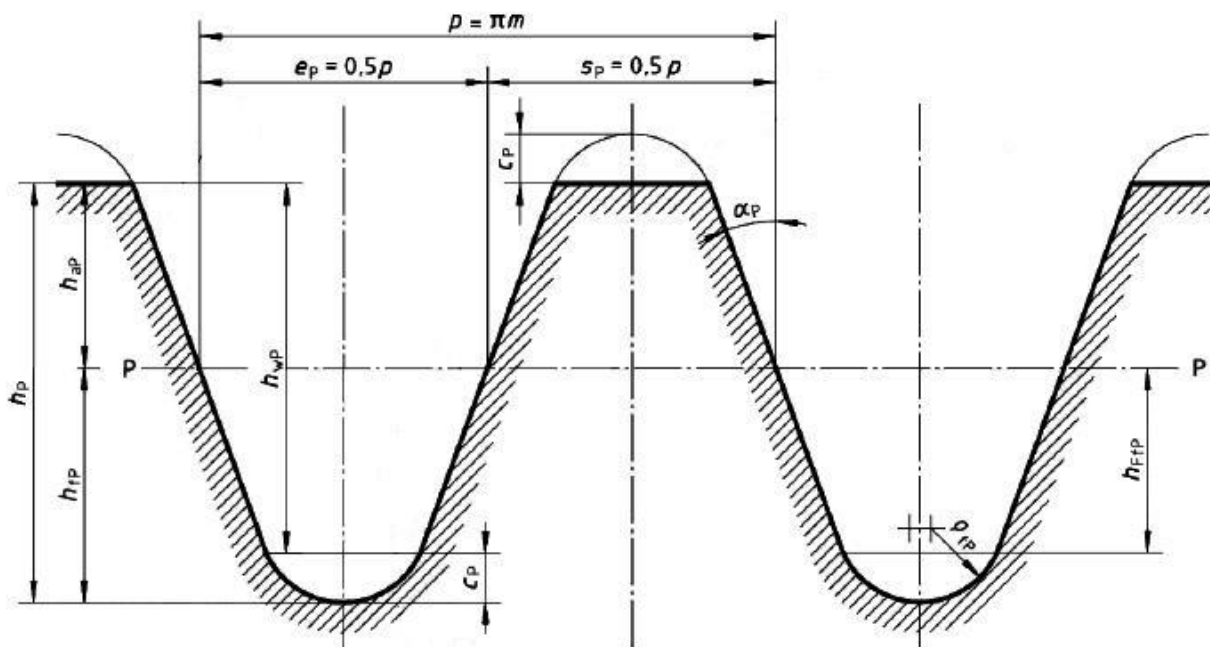
Prema [4], ako se jednadžba (39) promatra kao funkcija, ona može u intervalu $0 < \varphi < \pi/2$ imati dva rješenja. U tom slučaju, rješenje koje će dati točku početka evolvente je ono koje daje veću vrijednost r_{tro} , a to je manje od dva rješenja.

U [4] je predloženo da se jednadžba (39) rješava nekom iteracijskom metodom, kao što je Newton – Raphsonova. Ipak, prilikom izrade programske procedure unutar ovog diplomskog rada za rješavanje jednadžbe (39) korištena je metoda bisekcija, koja konvergira nešto sporije, ali i sigurnije nego Newton – Raphsonova metoda kojom u nekim slučajevima nije bilo moguće pronaći rješenje sa zadovoljavajućom točnošću.

3. ZUPČANIK

3.1. Osnovni profil ozubljenja

Prema [1] i [2] osnovni profil ozubljenja predstavlja ustvari zupčanik s beskonačno velikim brojem zubi zbog čega diobeni promjer postaje pravac tj. srednja linija standardnog profila, a bokovi su ravni jer polumjer zakrivljenja boka također postaje beskonačno velik. Upravo ova činjenica omogućuje izradu zupčanika s evolventnim ozubljenjem pomoću alata s ravnim bokovima. Prema [2] srednja linija standardnog profila (P-P) siječe standardni profil tako da je debljina zuba jednaka širini uzubine kako je prikazano na [Slika 12].



Slika 12. Osnovni profil ozubljenja [5]

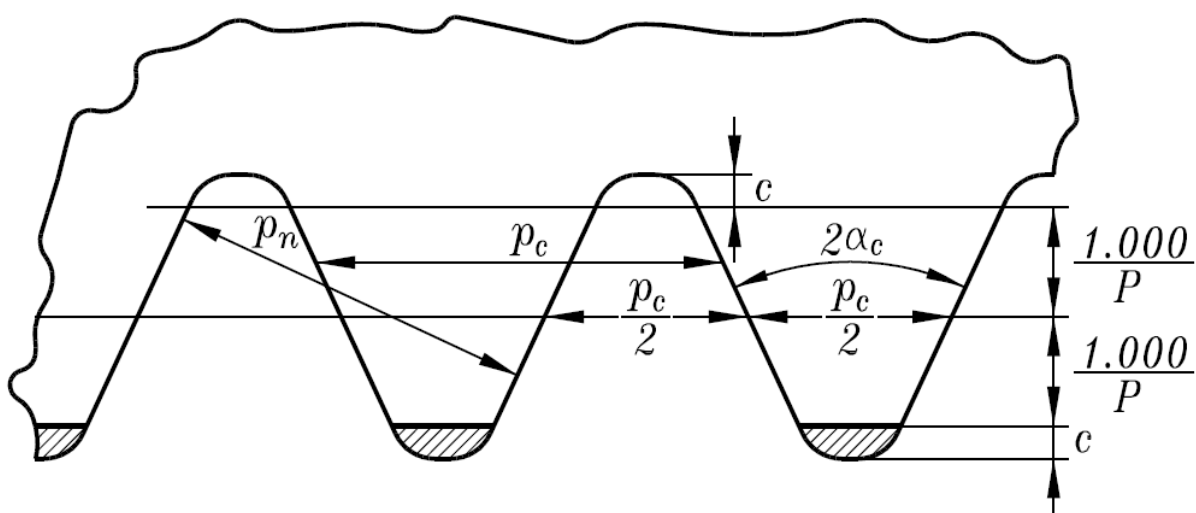
Veličine označene na slici su:

- h_p – visina zuba
- h_{rP} – visina podnožnog dijela zuba
- h_{aP} – visina tjemelog dijela zuba
- h_{wP} – visina radnog dijela zuba
- c_p – tjemena zračnost

- e_p – širina uzubine na srednjoj liniji
- s_p – širina zuba na srednjoj liniji
- p – korak osnovnog profila
- α_p – kut boka osnovnog profila
- ρ_{FP} – polumjer zakrivljenja podnožja zuba osnovnog profila

Kao što je vidljivo na [Slika 12], korak osnovnog profila jednak je umnošku konstante π i veličine m koja se naziva modul. Prema [2] u cilju pojednostavljenja proračuna i izrade (smanjenja broja alata) usvojeno je da je korak višekratnik broja π . To znači da je modul cijeli broj koji se prilikom proračuna i izrade zupčanika izabire između određenih standardnih vrijednosti i kao takav predstavlja karakterističnu veličinu samog zupčanika jer se preko njega izražavaju sve ostale veličine zuba i zupčanika. Odabir većeg modula za posljedicu ima i veći zub, kao i povećanje samog zupčanika. Jedinica je modula milimetar.

Treba razlikovati dimenzije osnovnog profila i alata za izradu zupčanika, odnosno ozubnice jer je tjemena visina ozubnice veća od tjemene visine osnovnog profila. Osjenčani dio na [Slika 13] ima samo ozubnica.



Slika 13. Ozubnica [1]

Prema [5] standardne vrijednosti veličina osnovnog profila zupčanika prikazane su u [Tablica 1].

Tablica 1. Standardne vrijednosti veličina osnovnog profila

Veličina	Vrijednost
α_P	20°
h_{aP}	$1 m$
c_P	$0,25 m$
h_{fP}	$1,25 m$
ρ_{fP}	$0,38 m$

Iz tablice je vidljivo, kao što je ranije navedeno, da se sve veličine, osim kuta α_P , zadaju kao višekratnici modula m .

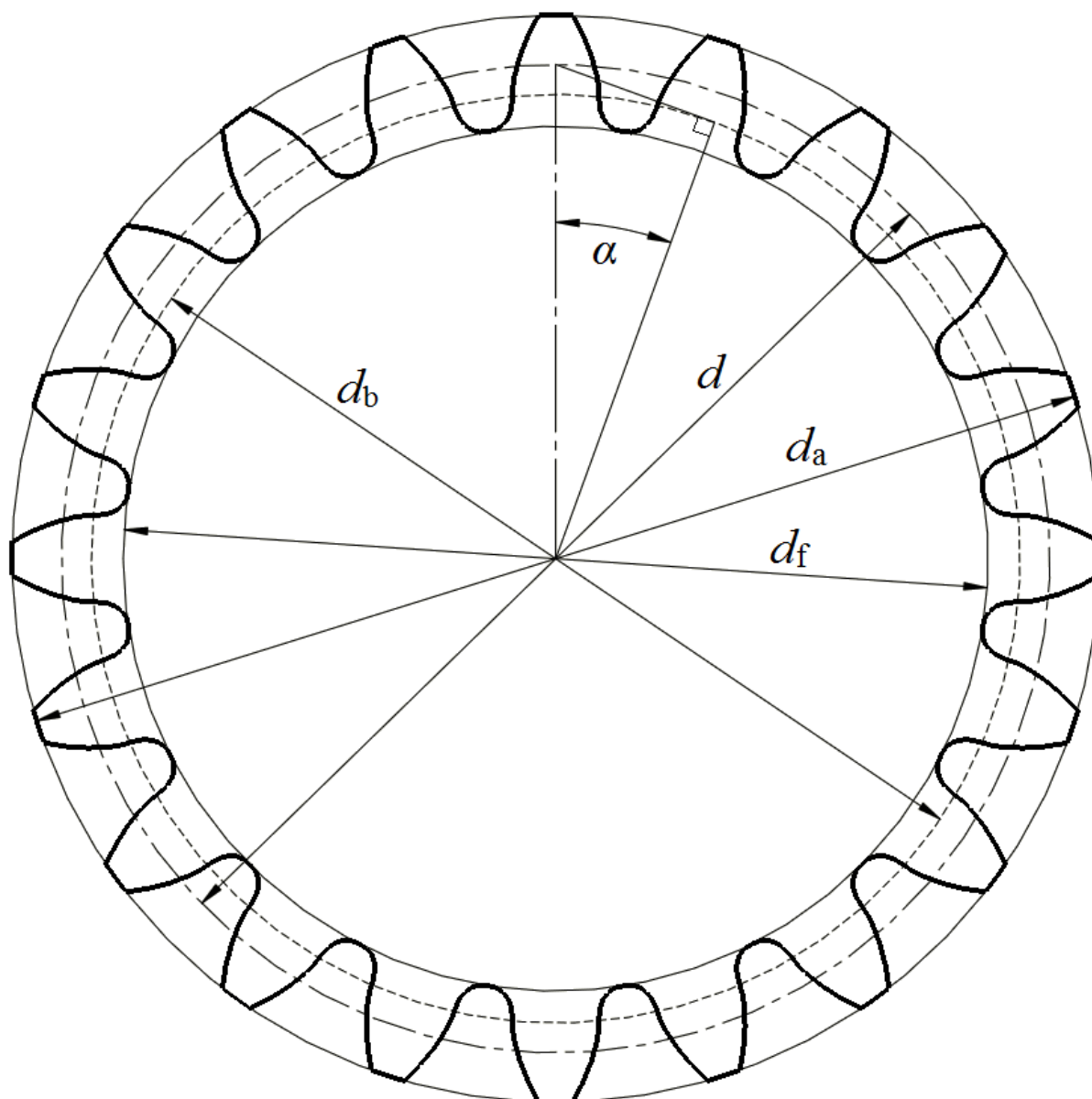
Prema [5] i [6], osim standardnog osnovnog profila, postoji više tipova osnovnog profila koji se mogu koristiti pri izradi zupčanika, a svaki od njih razlikuje se po vrijednostima karakterističnih veličina. U ovom diplomskom radu pri izradi programske procedure korišteni su tipovi osnovnog profila navedeni u [Tablica 2].

Tablica 2. Tipovi osnovnog profila

Tip osnovnog profila	Veličina				
	α_P	h_{aP}	c_P	h_{fP}	ρ_{fP}
ISO 53 Profil A	20°	$1 m$	$0,25 m$	$1,25 m$	$0,38 m$
ISO 53 Profil B	20°	$1 m$	$0,25 m$	$1,25 m$	$0,3 m$
ISO 53 Profil C	20°	$1 m$	$0,25 m$	$1,25 m$	$0,25 m$
ISO 53 Profil D	20°	$1 m$	$0,4 m$	$1,4 m$	$0,39 m$
DIN 3972 Profil I	20°	$1 m$	$0,167m$	$1,167 m$	$0,2m$
DIN 3972 Profil II	20°	$1 m$	$0,25 m$	$1,25 m$	$0,2m$

Prema [5] profil A je standardni osnovni profil i pogodan je za prijenos velikog okretnog momenta. Profili B i C namijenjeni su normalnoj, širokoj upotrebi. Profil D namijenjen je za precizne zupčanike koji prenose velike okretne momente.

3.2. Karakteristične mjere zupčanika



Slika 14. Karakteristične mjere zupčanika

[Slika 14] prikazuje karakteristične mjere zupčanika, a to su:

- d – diobeni promjer
- d_a – tjemeni promjer
- d_f – podnožni promjer
- d_b – podnožni promjer
- α – kut zahvatne crte

Kut zahvatne crte α nije računska veličina nego se odabire izborom tipa osnovnog profila budući da je jednak kutu boka zuba osnovnog profila α_p . U slučaju zupčanika s ravnim zubima, nakon odabira tipa osnovnog profila, modula (m), broja zubi (z) i faktora pomaka profila (x) moguće je izračunati karakteristične veličine zupčanika pomoću sljedećih izraza:

$$d = m \cdot z, \quad (40)$$

$$d_b = d \cdot \cos \alpha, \quad (41)$$

$$d_a = d + 2 \cdot (h_{ap} + x \cdot m), \quad (42)$$

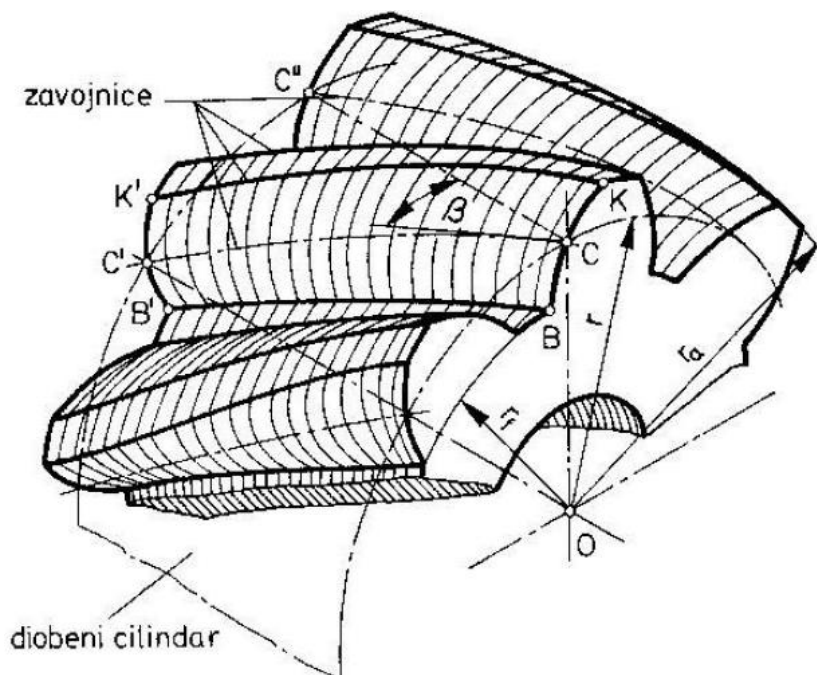
$$d_f = d - 2 \cdot (h_{fp} - x \cdot m). \quad (43)$$

3.3. Zupčanik s kosim (helikoidnim) zubima

Prema [2] bok zuba zupčanika s kosim zubima je helikoidna evolventna površina čija izvodnica zatvara s izvodnicom temeljnog cilindra kut β_b . Helikoidnu evolventnu površinu opisuje pravac ravnine koja se valja po temeljnom cilindru bez klizanja, a pravac pritom s izvodnicom temeljnog cilindra zatvara kut β_b .

Kut nagiba boka zuba β je kut nagiba zavojnice uzdužne linije boka zuba na cilindru diobenog promjera d .

Nastajanje zupčanika s kosim zubima može zamisliti i kao rezanje zupčanika s ravnim zubima na vrlo tanke pločice koje se zatim zakreću tako da kinematske točke C pojedinih pločica nalaze na zavojnici C – C' cilindra diobene kružnice [Slika 15].

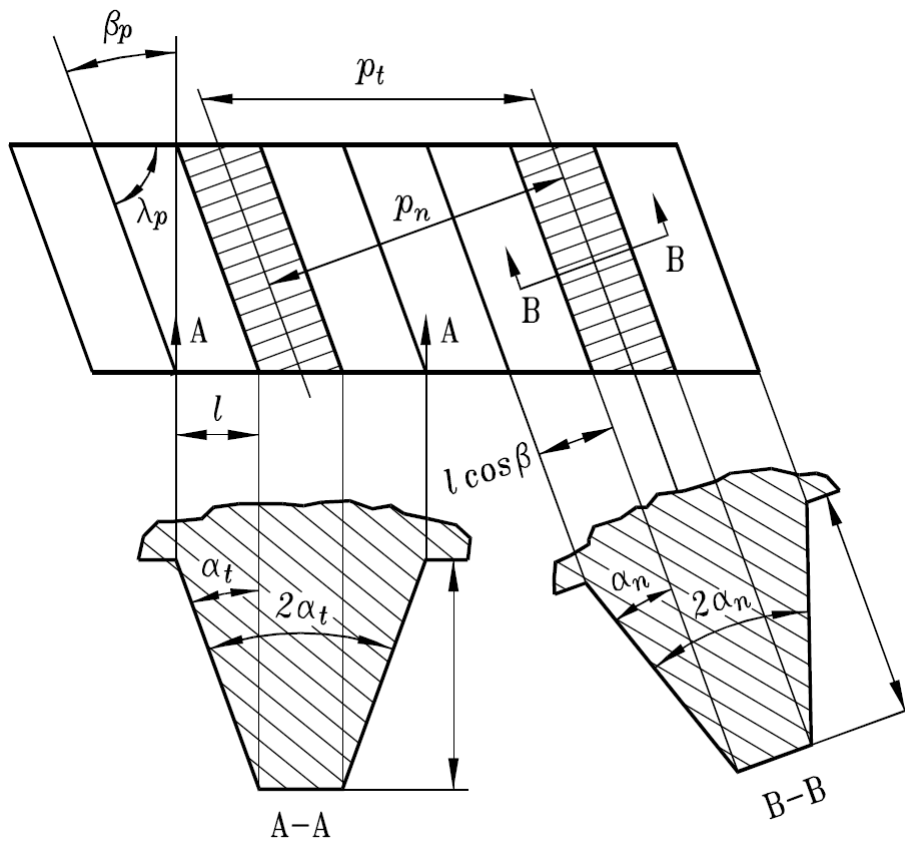


Slika 15. Nastajanje zupčanika s kosim zubima [2]

Prednost zupčanika s kosim zubima u odnosu na one s ravnim jest postepeni ulazak zubi u zahvat. Zahvat počinje na jednoj strani i postepeno se širi preko cijele širine zuba. Zbog postepenog ulaska zubi u zahvat, oni se postepeno opterećuju pa im je rad tih. S obzirom da opterećenje zuba nije istodobno preko cijele širine kao kod zupčanika s ravnim zubima, opteretivost zupčanika s kosim zubima je veća [2].

Nedostatak zupčanika s kosim zubima u odnosu na one s ravnim jest postojanje aksijalne komponente sile koja dodatno opterećuje ležajeve i vratilo. Upravo kako ta sila ne bi bila prevelika, prema [2] kut nagiba boka β trebao bi se kretati između 8° - 20° i imati cjelobrojnu vrijednost kako bi se olakšala izrada.

Zupčanici s kosim zubima izrađuju se pomoću istih alata kao i zupčanici s ravnim zubima, ali se alat zakreće u odnosu na izradakpa se tako standardne vrijednosti modula i ostalih veličina pojavljuju u normalnoj ravnini, okomitoj na zub zupčanika, dok se evolventna svojstva i oblik tj. geometrija očituju u čeonj ravnini. Zbog toga razlikujemo karakteristične veličine zupčanika u normalnoj i čeonj ravnini [Slika 16], poglavito normalni (m_n) i čeonj (m_t) modul, normalni (α_n) i čeonj (α_t) kut zahvatne crte te normalni (p_n) i čeonj (p_t) korak. Navedene su veličine međusobno povezane preko kuta nagiba boka zuba β .



Slika 16. Normalna i čeona ravnina [1]

Prema [2] normalni i čeoni korak povezani su na sljedeći način:

$$\cos \beta = \frac{p_n}{p_t}. \quad (44)$$

Kako vrijede izrazi $p_n = m_n \pi$ i $p_t = m_t \pi$, njihovim uvrštavanjem u jednadžbu (44) moguće je izraziti čeoni modul preko normalnog modula:

$$m_t = \frac{m_n}{\cos \beta}. \quad (45)$$

Također, kut zahvatne crte u čeonoj i normalnoj ravnini povezani su preko slične jednadžbe:

$$\tan \alpha_t = \frac{\tan \alpha_n}{\cos \beta}. \quad (46)$$

Izvod jednadžbe (47) prikazan je u [2].

Karakteristične veličine zupčanika s kosim (helikoidnim) zubima moguće je izračunati pomoću sljedećih izraza:

$$d = m_t \cdot z = \frac{m_n}{\cos \beta} \cdot z, \quad (47)$$

$$d_b = d \cdot \cos \alpha_t, \quad (48)$$

$$d_a = d + 2 \cdot m_n \cdot (h_{ap}^* + x), \quad (49)$$

$$d_f = d - 2 \cdot m_n \cdot (h_{fp}^* - x), \quad (50)$$

Veličine h_{ap}^* i h_{fp}^* predstavljaju faktore tjemene odnosno podnožne visine zuba te uz modul određuju tjemenu i podnožnu visinu zuba:

$$h_{ap} = h_{ap}^* \cdot m_n, \quad (51)$$

$$h_{fp} = h_{fp}^* \cdot m_n. \quad (52)$$

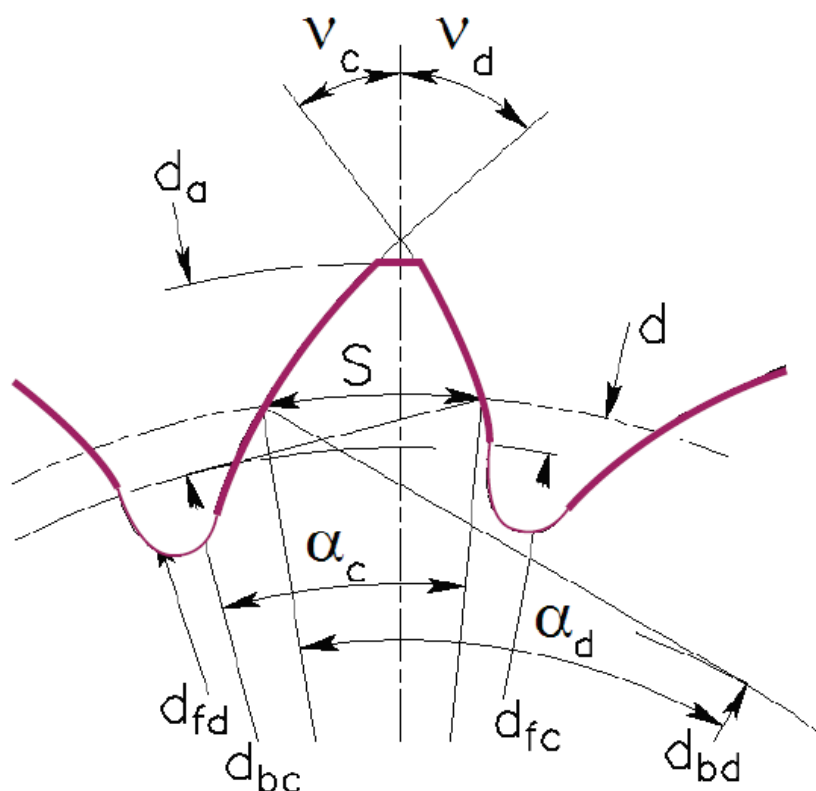
3.4. Zupčanik s asimetričnim zubima

Zubi zupčanika s asimetričnim zubima nemaju isti kut zahvatne crte na oba boka zuba. To znači da zub zupčanika s asimetričnim zubima možemo promatrati kao da je sastavljen od dva različita evolventna boka, od kojih svaki ima svoju temeljnu kružnicu. Budući da su im modul m i broj zubi z jednaki, jednaki su im i diobeni, tjemeni te podnožni promjer.

Prema [7] razlog za upotrebu zupčanika s asimetričnim zubima leži u činjenici da su prilikom upotrebe suprotni bokovi zuba većine zupčanika funkcionalno različiti tj. jedan bok je znatno opterećeniji tijekom dužeg vremenskog perioda od drugog boka, koji je često neopterećen ili znatno manje opterećen u vrlo kratkim vremenskim intervalima. Asimetričan zub prilagođen je takvoj upotrebi jer poboljšava svojstva opterećenijeg boka na račun svojstava neopterećenog boka.

Takvim se postupkom povećava nosivost i trajnost pogonskog, odnosno opterećenog boka što često rezultira i smanjenjem mase samog zupčanika, ali i mirnijim i tišim radom. Prema [8], budući da nosivost zuba uvelike ovisi o kutu zahvatne crte, u prvotnim se razmatranjima za pogonski bok odabire manji kut zahvatne crte, a za neopterećeni bok veći kut zahvatne crte.

U početku je za zahvatni kut pogonske crte biran uglavnom standardan kut od 20° čime je smanjeno naprezanje uzrokovano savijanjem dok je naprezanje uzrokovano kontaktnim pritiscima ostalo isto. Ipak, takvo oblikovanje asimetričnog zuba dovelo je i do povećanja krutosti zahvata te povećanja buke i vibracija. Noviji pristup, koji je u skladu s oblikovanjem zuba za upotrebu u zrakoplovnoj i svemirskoj industriji, za kut zahvatne crte pogonske strane odabire kut u iznosu $25^\circ - 28^\circ$ čime se povećava polumjer zakrivljenja evolventnog dijela zuba što rezultira smanjenjem naprezanja uzrokovano kontaktni pritiskom.



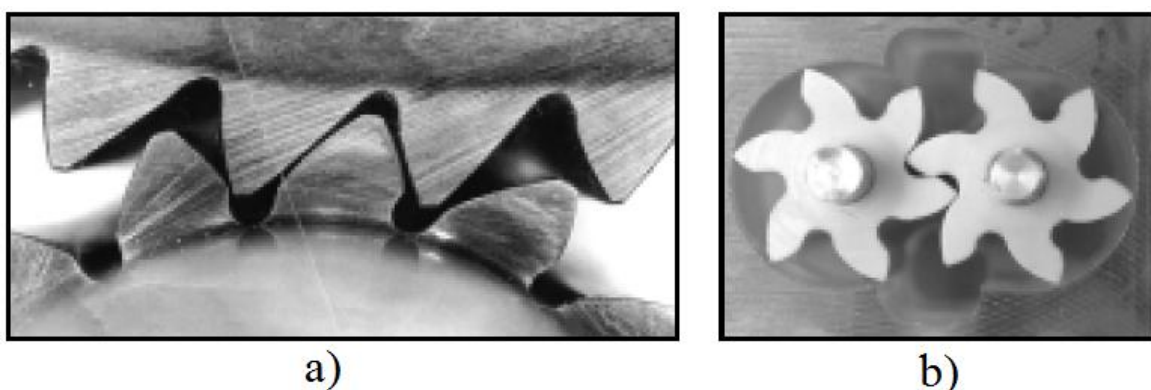
Slika 17. Asimetrični zub [9]

U [7] se navodi kako su zupčanici s asimetričnim zubima pogodni za upotrebu u uvjetima visoke opterećenosti zupčanika, kao što su zrakoplovna i svemirska industrija. Također, pogodni su za upotrebu u masovnoj proizvodnji zupčastih prijenosnik, gdje je trošak izrade alata u odnosu na broj proizvedenih zupčanika mali.

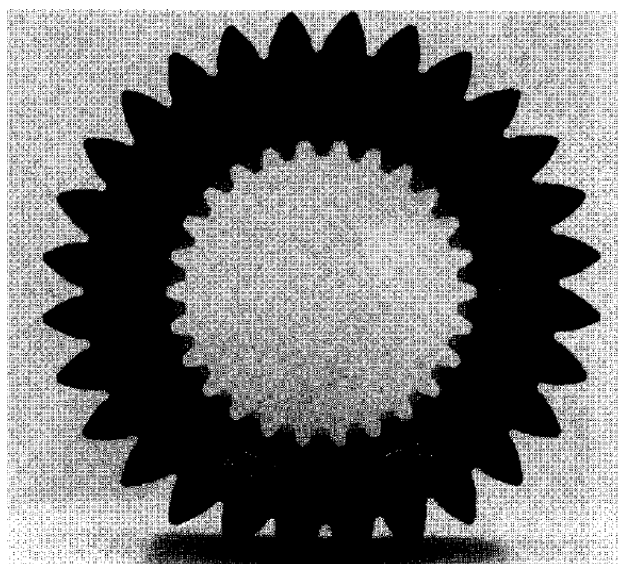
U [8] se navode primjeri konkretne primjene zupčanika s asimetričnim zubima. Jedan primjer primjene je dvostupanjski planetarni reduktor „turbo-prop“ motora TV7-117 koji se koristi za pogon zrakoplova Ilyushin-114. „Turbo-prop“ motor je zapravo propellerski motor čiji je

propeler pogonjen turbinskim mlaznim motorom, a između turbinskog i propelerskog vratila nalazi se spomenuti reduktor.

Drugi primjer primjene jest jednostupanjski zupčasti prijenosnik generatora za koji je izvršeno ispitivanje i usporedba s prijenosnikom koji koristi zupčanike sa simetričnim kosim zubima. U [8] su prikazani rezultati koji su pokazali da je naprezanje u korijenu zuba uzrokovano savijanjem, kao i naprezanje boka zuba uzrokovano kontaktnim pritiskom smanjeno upotrebom zupčanika s asimetričnim ravnim zubima u odnosu na upotrebu zupčanika sa simetričnim kosim zubima. Također, na prijenosniku koji koristi zupčanike s asimetričnim zubima zabilježena je znatno manja razina vibracija.



Slika 18. Asimetrični zubi u zahvatu [7]: a) prijenosnik generatora, b) pumpa



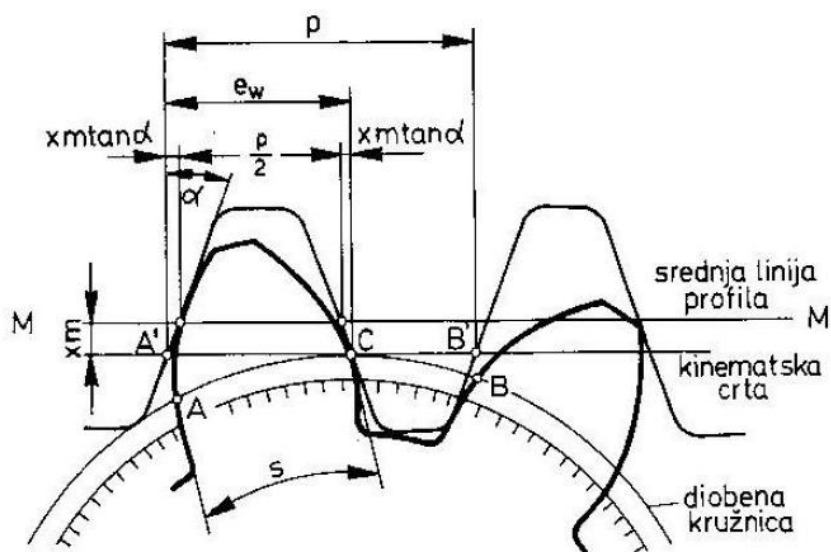
Slika 19. Zupčanik prvog stupnja planetarnog reduktora motora TV7 – 117 [8]

3.5. Pomak profila

Prema [2], s obzirom da su zupčanici s evolventnim ozubljenjem neosjetljivi na malu promjenu osnovnog razmaka, ta se činjenica koristi kako bi se mogli upotrebljavati i izrađivati i zupčanici čiji je broj zubi manji od minimalnog potrebnog za izbjegavanje podrezanosti korijena. To se ostvaruje tako da se prilikom izrade zupčanika standardni alat u obliku ozubnice odmiče od izratka za iznos pomaka profila što znači da se srednja linija alata prilikom izrade ne valja više po diobenoj kružnici zupčanika. Iznos pomaka profila izražava se pomoću modula kao $x \cdot m$ pri čemu x označava faktor pomaka profila.

Glavni razlog za upotrebu pomaka profila je izbjegavanje podrezanosti korijena zupčanika kod manjeg broja zubi, ali prema [2] on se koristi i za postizanje povoljnijeg oblika zuba, povećanja opteretivosti, poboljšanje uvjeta klizanja i trošenja boka, kao i za ostvarivanje propisanog razmaka osi.

Prema [2] prilikom izrade zupčanika s pomakom profila kinematska crta alata valja se po diobenoj kružnici zupčanika [Slika 20] pa debljina zuba s na diobenom promjeru zupčanika odgovara širini uzubine e_w alata na kinematskoj crti.



Slika 20. Debljina zuba kod pomaka profila [2]

Prema [2] izraz kojim se izračunava debljina zuba s na diobenom promjeru uz odnose sa [Slika 20] za zupčanik s ravnim zubima glasi:

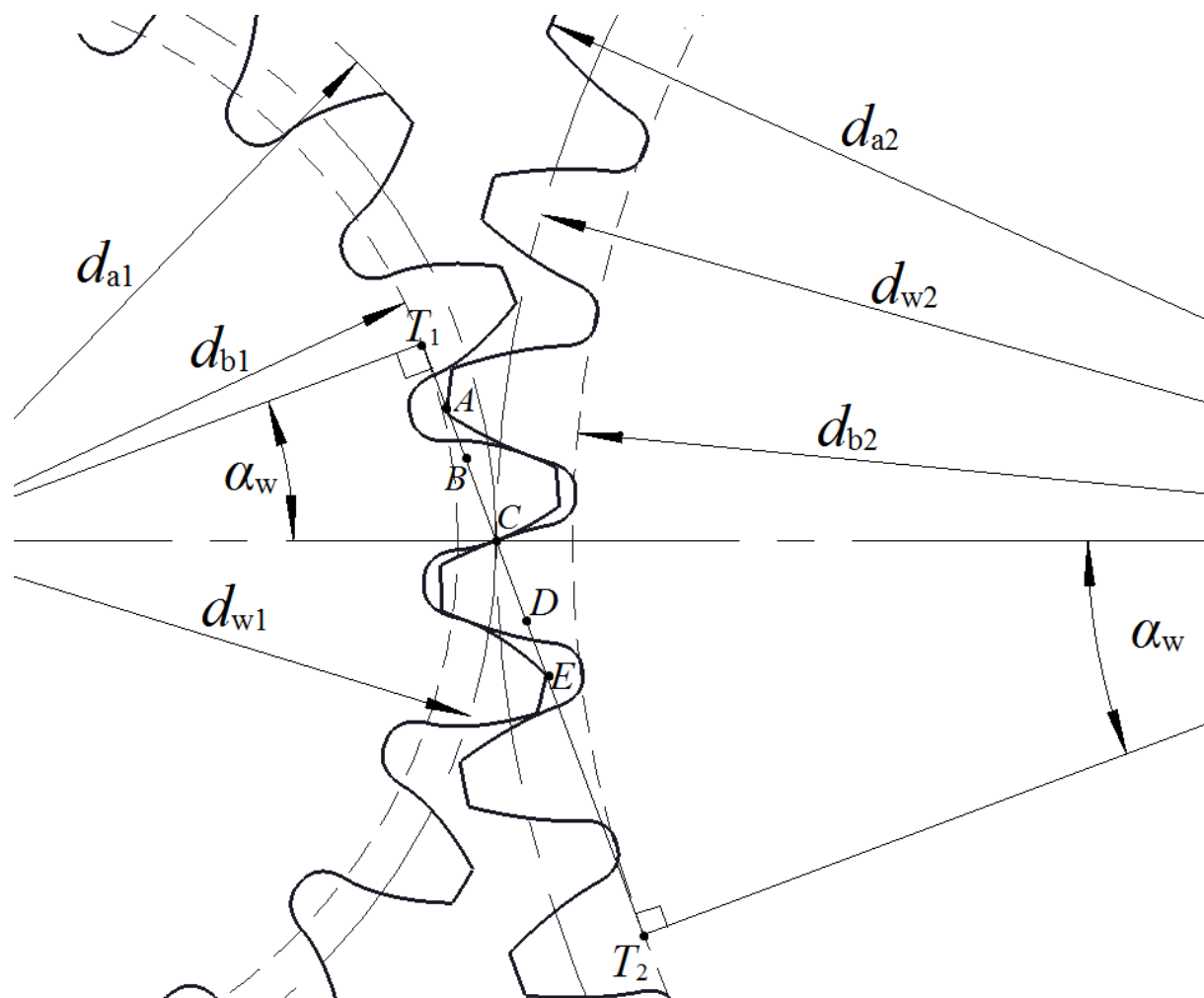
$$s = m \cdot \left(\frac{\pi}{2} + 2 \cdot x \cdot \tan \alpha \right) \quad (53)$$

Izraz za izračun debljine zuba s na diobenom promjeru u čeonj ravni za zupčanik s pomakom profila i kosim zubima glasi:

$$s_t = \frac{m_n}{\cos \beta} \cdot \left(\frac{\pi}{2} + 2 \cdot x \cdot \tan \alpha_n \right) \quad (54)$$

Izrazi za izračun karakterističnih mjera zupčanika s pomakom profila već su dani ranije u radu u jednadžbama (40) do (43) za zupčanike s ravnim zubima, odnosno u jednadžbama (47) do (50) za zupčanike s kosim zubima.

3.6. Zahvat zupčanika



Slika 21. Detalj zahvata zupčanika

Zupčanici u zahvat ulaze tako da podnožni dio zuba pogonskog zupčanika dolazi u dodir s točkom, odnosno linijom na sjecištu evolventnog dijela boka i tjemelog promjera gonjenog zupčanika. Situacija je obrnuta na izlasku iz zahvata. Kinematski pol C označava točku zahvata koja leži na spojnici osi dvaju zupčanika i prema zakonu ozubljenja iz [2] dijeli osni razmak u odnosu prijenosnog omjera [Slika 21]. Kinematski pol je ujedno točka sjecišta zahvatne crte i spojnice osi zupčanika. Prema [2] zahvatna crta predstavlja geometrijsko mjesto svih uzastopnih dodirnih točaka dvaju bokova. Također, zahvatna crta jest tangenta temeljnih kružnica pogonskog i gonjenog zupčanika, a predstavlja i okomicu na tangentu u svakoj točki dodira bokova dvaju zupčanika.

Osim kinematskog pola C karakteristične točke zahvata na [Slika 21] označene su slovima A, B, D i E. Točka A jest prva točka zahvata para zubi dvaju zupčanika, a određuje se kao sjecište zahvatne crte i tjemene kružnice gonjenog zupčanika. Nasuprot tome, točka E označava zadnju točku zahvata para zubi dvaju zupčanika, a određuje se kao sjecište zahvatne crte i tjemene kružnice pogonskog zupčanika. Točka B predstavlja točku zahvata u kojoj se nalazi par zubi dvaju zupčanika kada se par zubi koji je u zahvat ušao prije njih nalazi u točki E, odnosno upravo izlazi iz zahvata. Slično tome, točka D predstavlja točku zahvata u kojoj se nalazi par zubi u trenutku kada novi par zubi ulazi u zahvat, dakle nalazi se u točki A. To znači da su tijekom zahvata dvaju zupčanika od točke A do točke B, odnosno od točke D do točke E u zahvatu dva para zubi, ako se radi o zupčaniku s ravnim zubima. Također, to znači da je tijekom zahvata dvaju zupčanika od točke B do točke D u zahvatu samo jedan par zubi.

U općenitom slučaju kada su u zahvatu dva zupčanika koji su izrađeni svaki sa svojim pomakom profila, u kinematskom polu C dodiruju se njihove kinematske kružnice. Jednako tako, osni razmak tih zupčanika nije jednak zbroju polumjera diobenih kružnica. U tom se slučaju pojavljuju nove proračunske mjere i veličine zupčanika, a to su kinematski promjer (d_w) i pogonski kut zahvatne crte (α_w).

Kod računanja ovih veličina razlikuju se dva slučaja. U prvom slučaju poznati su faktori pomaka profila oba zupčanika (x_1 i x_2), a osni razmak a nije određen. U drugom je slučaju osni razmak a određen pa se faktori pomaka profila prilagođavaju kako bi se postigao traženi osni razmak.

U prvom se slučaju, prema [2] pogonski kut zahvatne crte određuje iz sljedeće jednadžbe:

$$ev\alpha_{tw} = 2 \cdot \frac{x_1 + x_2}{z_1 + z_2} \cdot \tan \alpha_n + ev\alpha_t. \quad (55)$$

Nakon što se izračuna vrijednost evolventne funkcije pogonskog kuta zahvatne crte u čeonj ravnini $ev\alpha_{tw}$, vrijednost samog kuta očita se iz tablice evolventne funkcije. Kada je poznata vrijednost pogonskog kuta zahvatne crte moguće je izračunati kinematske promjere zupčanika pomoću sljedećih izraza:

$$d_{w1} = d_1 \cdot \frac{\cos \alpha_t}{\cos \alpha_{tw}}, \quad (56)$$

$$d_{w2} = d_2 \cdot \frac{\cos \alpha_t}{\cos \alpha_{tw}}. \quad (57)$$

Gornji izrazi proizlaze iz činjenice da se promjeri temeljnih kružnica ne mijenjaju s pomakom profila pa vrijedi $d_b = d \cdot \cos \alpha_t$ i $d_b = d_w \cdot \cos \alpha_{tw}$.

Osni razmak a tada se računa iz jednadžbe

$$a = \frac{m_n}{\cos \beta} \cdot \frac{z_1 + z_2}{2} \cdot \frac{\cos \alpha_t}{\cos \alpha_{tw}} = \frac{d_{w1} + d_{w2}}{2}. \quad (58)$$

U drugom slučaju prema [2], osni razmak a je zadan pa se pogonski kuta zahvatne crte računa pomoću jednadžbe

$$\cos \alpha_{tw} = \frac{m_n}{\cos \beta} \cdot \frac{z_1 + z_2}{2 \cdot a} \cdot \cos \alpha_t. \quad (59)$$

Nakon što je poznata vrijednost pogonskog kuta zahvatne crte, zbroj pomaka profila određuje se prema [2] na sljedeći način:

$$x_1 + x_2 = (z_1 + z_2) \cdot \frac{ev\alpha_{tw} - ev\alpha_t}{2 \cdot \tan \alpha_n}. \quad (60)$$

Kinematski se promjeri tada ponovno određuju iz jednadžbi (56) i (57).

U ovom diplomskom radu pri izradi programske procedure obrađen je drugi slučaj, dakle slučaj kada je zadan osni razmak a . Razlog za to je činjenica da se u prvom slučaju kada osni

razmak nije zadan, ali su zadani faktori pomaka profila, računanje vrijednosti pogonskog kuta zahvatne crte, prema [1], svodi na rješavanje nelinearne jednačbe

$$\hat{\alpha}_{tw} - \tan \alpha_{tw} + e\nu \alpha_{tw} = 0. \quad (61)$$

Naime, iz jednačbe (55) moguće je izračunati vrijednost evolventne funkcije, ali se onda vrijednost kuta pri analitičkom rješavanju očitava iz tablice. U programskoj proceduri može se samo prema definiciji evolventne funkcije doći do jednačbe (61) i, s obzirom da se vrijednost kuta ne može očitati iz tablice, riješiti nelinearnu jednačbu nekom numeričkom metodom.

4. PROGRAMSKA PROCEDURA

S obzirom da je konačni cilj ovog diplomskog rada bila izrada programske procedure za parametarsko modeliranje dvodimenzijskog modela zupčanika u čeonj ravnini, za izradu programske procedure izabran je *Visual Basic for Application* modul unutar programskog paketa *AutoCAD* 2014 zbog jednostavnosti pristupanja naredbama unutar samog programa, ali i zbog želje da se aplikacija može koristiti na ranijim i kasnijim verzijama programskog paketa *AutoCAD*. Jednako tako, programski paket *AutoCAD* omogućuje pohranjivanje crteža u različitim formatima koje je moguće otvarati i koristiti u ostalim programskim paketima za 3D modeliranje i analizu metodom konačnih elemenata što onda omogućuje daljnje modeliranje trodimenzijskog modela zupčanika i provedbu numeričke analize. Programska procedura omogućuje modeliranje dvodimenzijskog modela zupčanika s evolventnim ozubljenjem u čeonj ravnini i to s ravnim simetričnim i asimetričnim zubima te kosim (helikoidnim) simetričnim i asimetričnim zubima, kako je i zahtijevano u diplomskom zadatku. Uz dvodimenzijski model zupčanika u čeonj ravnini, programska procedura omogućuje i modeliranje zahvata zupčanika u čeonj ravnini i to u pet karakterističnih točaka zahvata.

4.1. Učitavanje programske procedure

Nakon otvaranja novog crteža u programskom paketu *AutoCAD* 2014, korisnik na traci izbornika odabire modul *Manage* i pomoću naredbe *Load Application* učitava programsku proceduru "Zupčanik" iz direktorija u kojem je pohranjena. Nakon učitavanja, programska procedura se pokreće pomoću naredbe *Run VBA Macro* unutar istog modula *Manage*.

Za starije verzije programskog paketa *AutoCAD* (npr. *AutoCAD* 2007) potrebno je nakon otvaranja novog crteža u traci izbornika odabrati *Tools* pa odabirom naredbe *Load Application* iz direktorija u kojem je pohranjena učitati programsku proceduru "Zupčanik". Nakon učitavanja, programska procedura se pokreće ponovnim otvaranjem izbornika *Tools* na traci izbornika i odabirom naredbe *Macro* pa odabirom podnaredbe *Macros*. U prozoru koji se otvori odabire se macro "Zupčanik" (ako je jedini bit će automatski odabran) i pokreće pritiskom na gumb *Run*.

4.2. Modeliranje dvodimenzijskog modela zupčanika u čeonj ravnini

Pokretanjem programske procedure unutar *AutoCAD* crteža otvara se korisničko sučelje koje prikazuje [Slika 22].

Slika 22. Korisničko sučelje pri modeliranju zupčanika

Korisnik u za to predviđena polja unosi vrijednost normalnog modula (m_n), broj zubi (z), vrijednost kuta nagiba boka zuba (β) te vrijednost faktora pomaka profila (x), kao i broj točaka koje će se koristiti za crtanje trohoide i evolvente. Kuta nagiba zuba ozubnice (α), faktor podnožne (h_{fp}^*) i tjemene (h_{ap}^*) visine zuba i faktor polumjera zaobljenja podnožja zuba (ρ_{fp}^*) korisnik može odabrati kao standardne vrijednosti jednog od tipova osnovnog profila ozubljenja koji su ranije navedeni u [Tablica 2] ili može unijeti vlastite vrijednosti u za

to predviđena polja koja postaju vidljiva tek odabirom opcije "Vlastito". Također, nudi se izbor crtanja trohoide i evolvente kao dvije zasebne krivulje ako se radi o nepodrezanom zuba. Ako opcija nije odabrana u slučaju nepodrezanog zuba trohoida i evolventa crtaju se kao jedna krivulja, dok se u slučaju podrezanog zuba, bez obzira je li opcija odabrana ili ne, crtaju dvije zasebne krivulje. Razlog tome je činjenica da se u slučaju podrezanog zuba evolventa i trohoida koje opisuju profil boka sijeku što znači da se na profilu boka zuba pojavljuje oštri prijelaz u točki sjecišta koji je teško modelirati jednom krivuljom.

Kao što je vidljivo na [Slika 22], pokretanjem programske procedure automatski je odabrana opcija modeliranja zupčanika sa simetričnim zubima. Ako korisnik želi modelirati zupčanik s asimetričnim zubima, mora odabrati opciju "ASIMETRIČAN ZUB" nakon čega polja za unos parametara asimetričnog zuba postaju dostupna, a sve opcije vezane za simetričan zub postaju nedostupne. Ako se ponovno odabere opcija "SIMETRIČAN ZUB", sva polja za unos podataka asimetričnog zuba postaju nedostupna, a sve opcije vezane za simetričan zub ponovno dostupne.

Nakon što su odabrane željene opcije i unesene vrijednosti svih potrebnih parametara, pritiskom na tipku "NACRTAJ" izvršava se kod programske procedure koji koristi jednadžbe od (25) do (39) navedene u poglavlju 2 za izračunavanje koordinata točaka evolvente i trohoide koje čine profil boka zuba prema odabranim i zadanim parametrima. Koordinate točaka se izračunavaju u polarnom koordinatnom sustava, a zatim se transformiraju u koordinate pravokutnog koordinatnog sustava koji se koristi u programskom paketu *AutoCAD*.

S obzirom da je u najopćenitijem slučaju potrebno omogućiti modeliranje asimetričnog zuba kod kojeg kut zahvatne crte α nije jednak za oba boka (simetrični se zub može promatrati kao poseban slučaj asimetričnog zuba kojem je kut zahvatne crte oba boka jednak), u ovoj programskoj proceduri zasebno se računaju koordinate točaka jednog i drugog boka. U skladu s nomenklaturom asimetričnog zuba i radi lakšeg opisa jedan se bok naziva *DRIVE*, a drugi *COAST*.

Prilikom transformacije koordinata točaka *DRIVE* boka iz polarnog u pravokutni koordinatni sustav svaka se točka pomiče u smjeru kazaljke na satu za kut γ , tako da iz početnog položaja koji prikazuje [Slika 24] dobivenog jednadžbama evolvente i trohoide iz [4] bok zuba prelazi u položaj prikazan na [Slika 25]. U tom novom položaju simetrala simetričnog zuba s oba

DRIVE boka jest os x . Razlog tome je što je kut γ ustvari kut između simetrale simetričnog zuba i osi x [Slika 23]. Taj se kut izračunava pomoću sljedećih izraza:

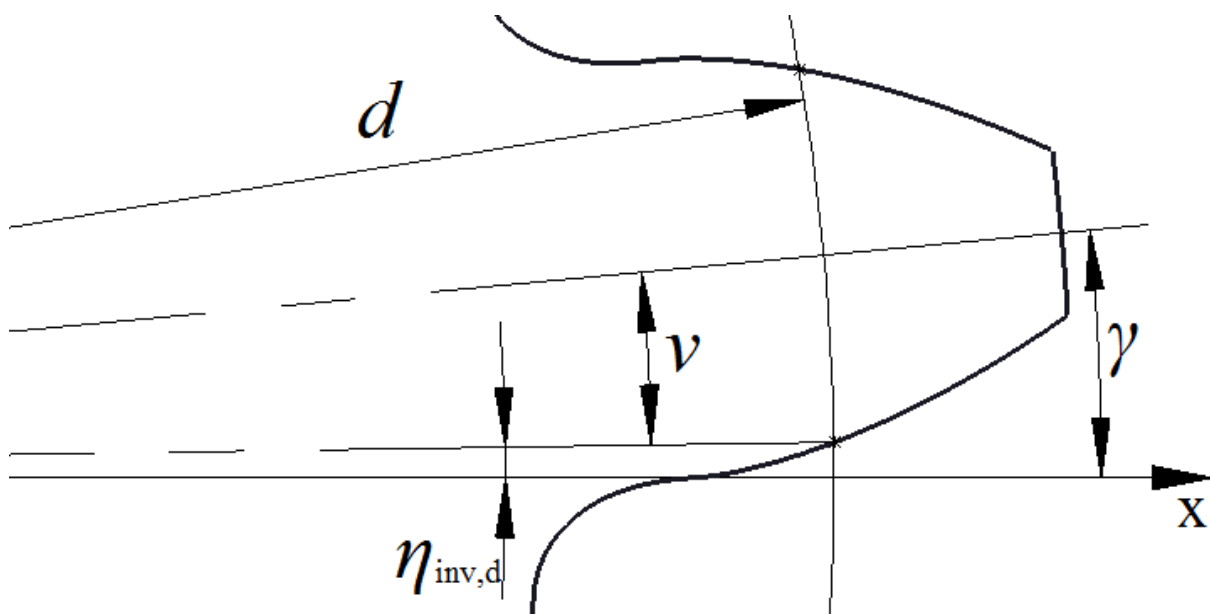
$$v = \frac{s/2}{d/2} = \frac{s}{d}, \quad (62)$$

$$\xi_d = \sqrt{\left(\frac{d}{d_b}\right)^2 - 1}, \quad (63)$$

$$\eta_{\text{inv},d} = \xi_d - \arctan \xi_d, \quad (64)$$

$$\gamma = v + \eta_{\text{inv},d}. \quad (65)$$

Jednadžbom (62) izračunava se kut koji odgovara luku na diobenoj kružnici čija je duljina jednaka polovici širine zuba s na diobenom promjeru. Jednadžbe (63) i (64) su jednadžbe evolvente prema [4], a izračunavaju kut točke evolvente čiji je polumjer jednak polumjeru diobene kružnice, dakle kut točke sjecišta evolvente i tjemene kružnice.



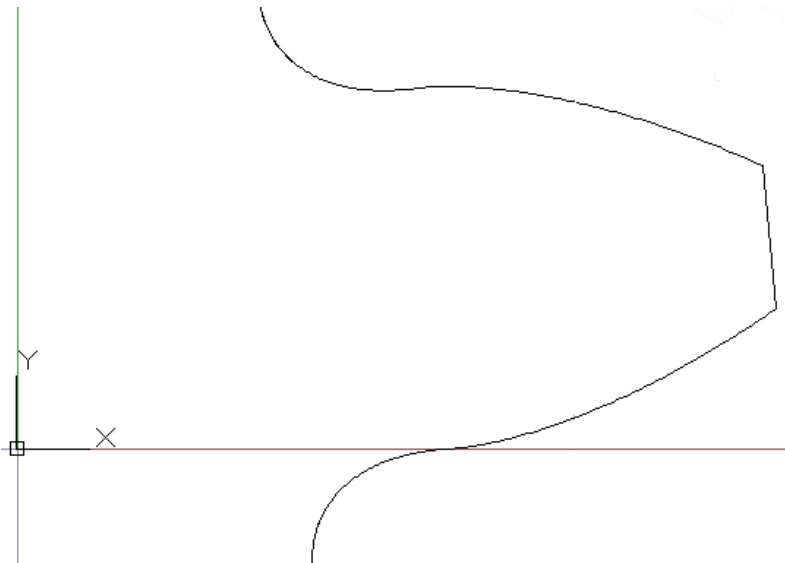
Slika 23. Kut simetrale simetričnog zuba

Jednadžbe transformacije koordinata točaka *DRIVE* boka tada su:

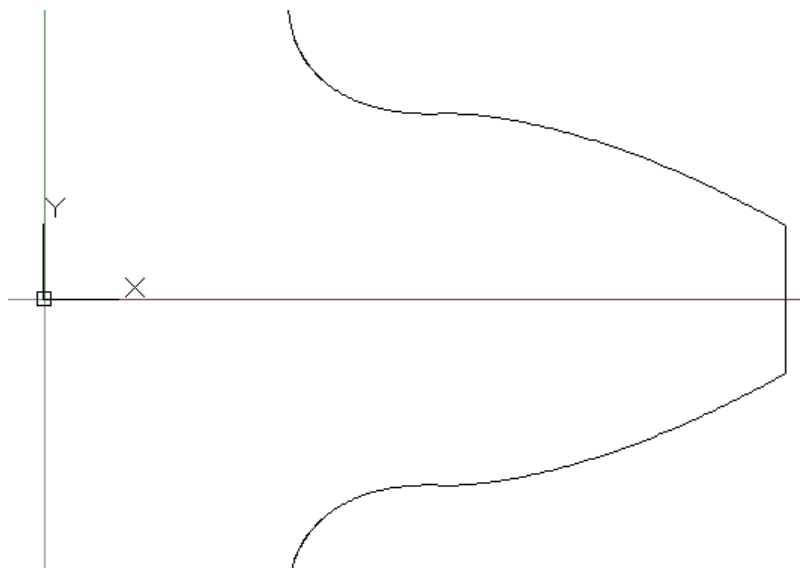
$$x_D = r \cdot \cos(\eta - \gamma), \quad (66)$$

$$y_D = r \cdot \sin(\eta - \gamma), \quad (67)$$

pri čemu r i η predstavljaju koordinate u polarnom koordinatnom sustavu i poprimaju vrijednosti r_{tro} i η_{tro} za dio boka koji opisuje trohoida, odnosno r_{inv} i η_{inv} za dio boka koji opisuje evolventa.



Slika 24. Položaj simetričnog zuba bez pomaka za kut γ

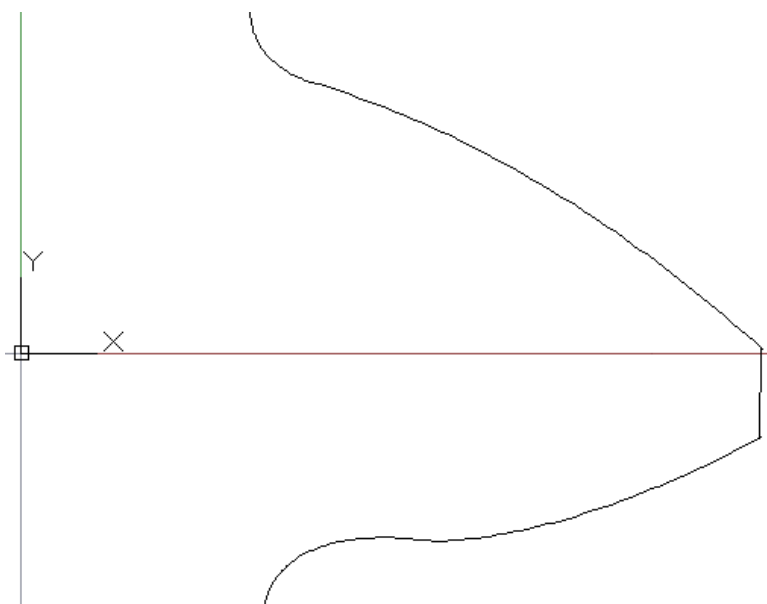


Slika 25. Položaj simetričnog zuba nakon pomaka za kut γ

Kao što je ranije rečeno, kako kod asimetričnog zuba *DRIVE* i *COAST* bok imaju različite kutove zahvatne crte α , tako imaju i različite kutove $\eta_{inv,d}$. Štoviše, ako se radi o zupčanicima s asimetričnim zubima s pomakom profila, razlikovat će se i kutovi ν jer prema jednadžbi (54) širina zuba na diobenom promjeru zupčanika s pomakom profila ovisi o kutu zahvatne crte. To znači da fiktivni zubi s oba *DRIVE*, odnosno oba *COAST* boka u općenitom slučaju kada se nalaze u početnom položaju određenom jednadžbama evolvente i trohoide iz [4] bez pomaka za kut γ nemaju zajedničku simetralu, odnosno položaji njihovih simetrala nisu isti. To onemogućava modeliranje asimetričnog zuba preslikavanjem *COAST* boka preko njegove simetrale. Zbog toga je potrebno pomicanje bokova za kut γ , čime os x postaje simetrala i fiktivnog zuba s oba *DRIVE* boka i fiktivnog zuba s oba *COAST* boka. Sada je moguće modelirati asimetrični zub preslikavanjem *COAST* boka iz položaja nakon pomaka za kut γ [Slika 25] preko zajedničke simetrale, odnosno osi x , dok *DRIVE* bok ostaje u svom položaju pomaknut za kut γ . Preslikavanje oko osi x izvodi se tako da se koordinati y promijeni predznak, dok koordinata x ostaje ista. Potpune jednadžbe transformacije koordinata točkica *COAST* boka iz polarnog u pravokutni koordinatni sustav, uz istovremeno preslikavanje preko osi x , tada glase:

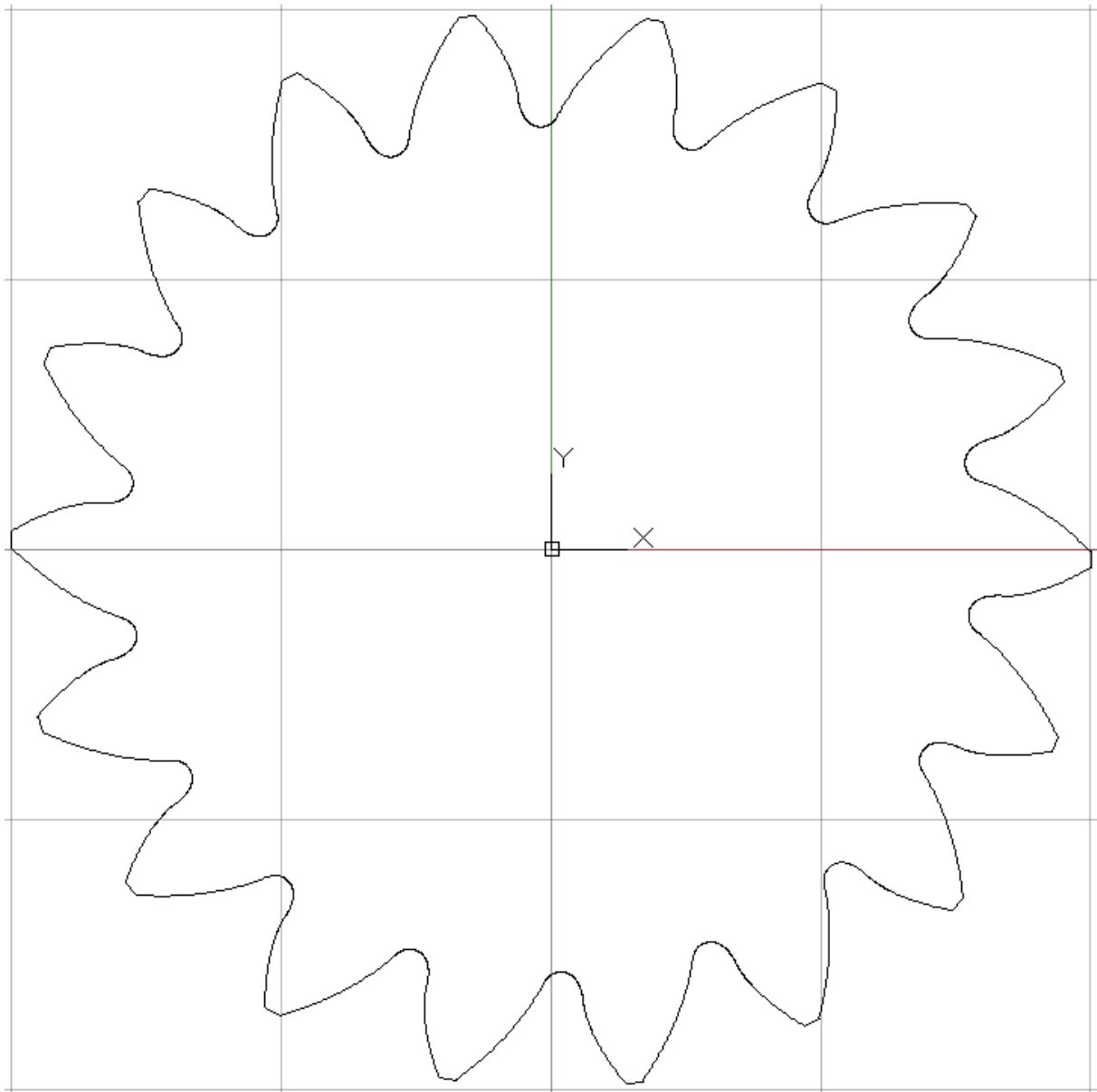
$$x_c = r \cdot \cos(\eta - \gamma), \quad (68)$$

$$y_c = -r \cdot \sin(\eta - \gamma). \quad (69)$$

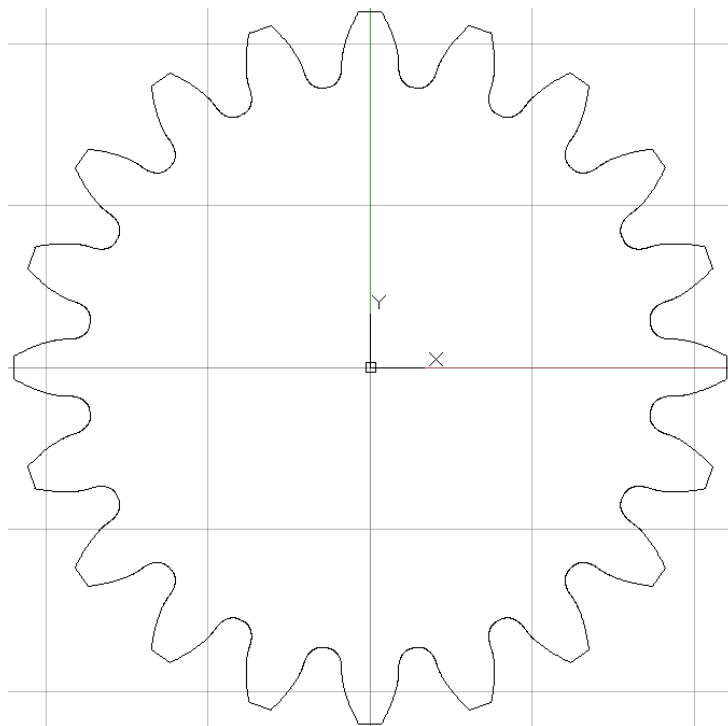


Slika 26. Asimetrični zub

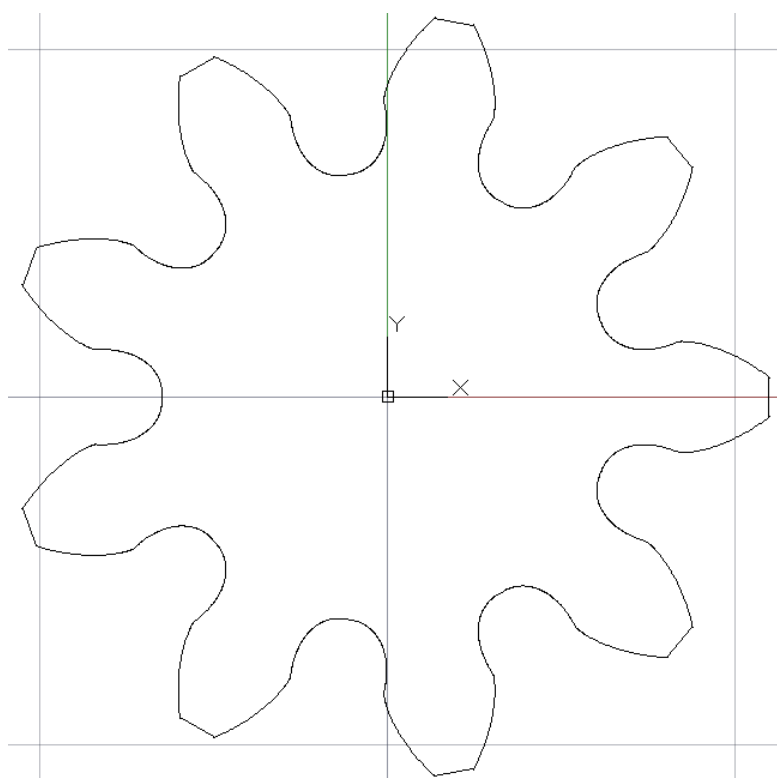
Koordinate točaka trohoide i evolvente koje opisuju bokove zuba dobivene jednadžbama (66) do (69) pohranjuju se u odgovarajuće nizove vrijednosti koji onda služe kao skup točaka kroz koje se crta *spline* krivulja. Nakon što su nacrtani, bokovi se povezuju lukom tjemene kružnice i tako tvore potpuni profil zuba. Zupčanik nastaje kopiranjem i rotiranjem tako modeliranog zuba oko središta tako da je svaki sljedeći zub od prethodnog udaljen za kut $2\pi/z$.



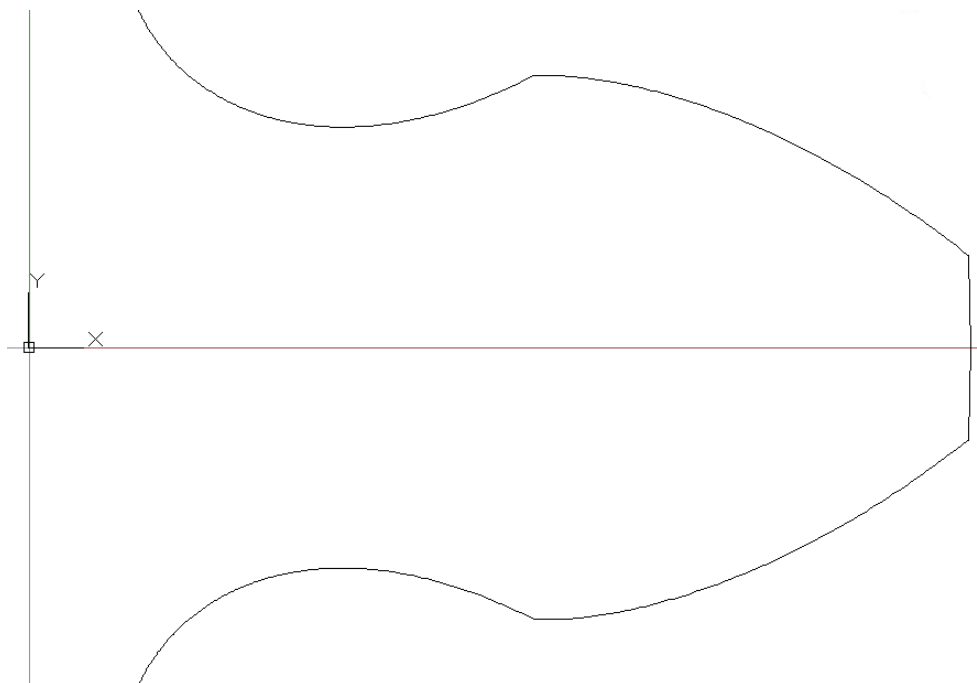
Slika 27. Zupčanik s asimetričnim zubima



Slika 28. Zupčanik sa simetričnim nepodrezanim zubima



Slika 29. Zupčanik sa simetričnim podrezanim zubima

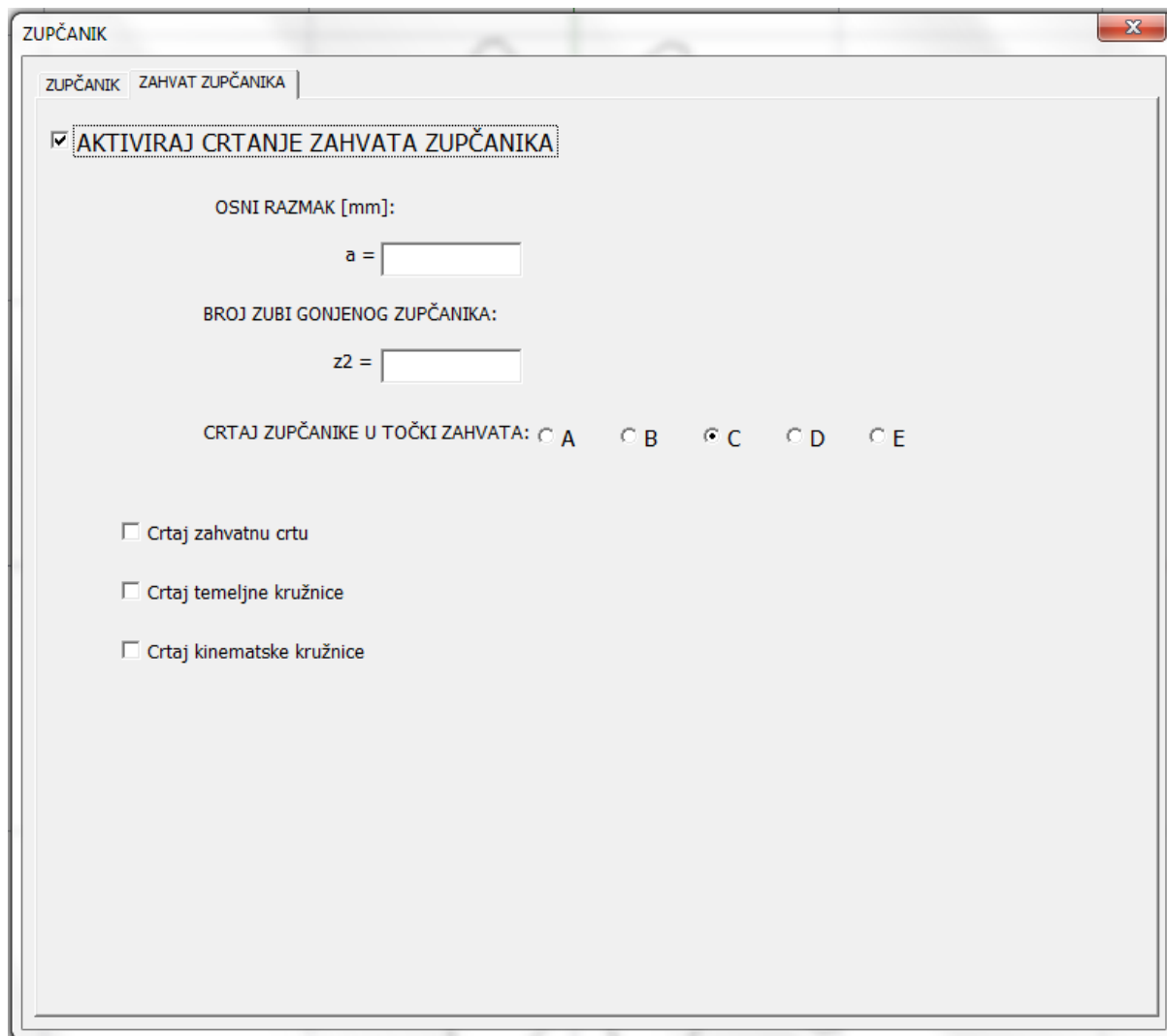


Slika 30. Podrezani simetrični zub

4.3. Modeliranje zahvata zupčanika u čeonj ravnini

Nakon učitavanja i pokretanja programske procedure na način opisan ranije u radu, modeliranje zahvata zupčanika započinje unošenjem svih potrebnih vrijednosti parametara kao u slučaju kada se modelira samo zupčanik. To sada postaju vrijednosti parametara pogonskog zupčanika. Za unos dodatnih parametara potrebnih za modeliranje zahvata zupčanika potrebno je otvoriti drugu stranicu korisničkog sučelja pritiskom na karticu "ZAHVAT ZUPČANIK" u lijevom gornjem kutu korisničkog sučelja [Slika 31].

Nakon otvaranja druge stranice, potrebno je označiti opciju "AKTIVIRAJ CRTANJE ZAHVATA ZUPČANIK" čime postaju dostupne sve opcije i polja za unos vrijednosti potrebnih parametara. U za to predviđena polja unosi se vrijednost osnovnog razmaka a i broj zubi drugog, odnosno gonjenog zupčanika. Vrijednosti svih ostalih parametara gonjenog zupčanika, dakle vrijednost normalnog modula (m_n), vrijednost kuta nagiba boka zuba (β) te vrijednost faktora pomaka profila (x), kao i broj točaka koje će se koristiti za crtanje trohoide i evolvente, kut nagiba zuba ozubnice (α), faktor podnožne (h_{fp}^*) i tjemene (h_{ap}^*) visine zuba i faktor polumjera zaobljenja podnožja zuba (ρ_{fp}^*), iste su kao kod pogonskog zupčanika.



Slika 31. Korisničko sučelje za modeliranje zahvata zupčanika

Programska procedura omogućuje crtanje zahvata zupčanika u pet karakterističnih točaka pa je potrebno označiti jednu od opcija, ovisno o tome koja se točka zahvata želi crtati. Također, moguće je odabrati opcije crtanja zahvatne crte, temeljnih i diobenih kružnica oba zupčanika. Nakon što su unesene sve potrebne vrijednosti i odabrane željene opcije, povratkom na prvu stranicu i pritiskom na tipku "NACRTAJ" pokreće se programska procedura i izvršavanje programskog koda. Najprije se izvršava kod koji je opisan na prethodnim stranicama i određuju se koordinate točaka pogonskog zupčanika. Potom se na isti način, korištenjem istih jednadžbi iz [4] određuju koordinate točaka evolvente i trohoide koje opisuju bok gonjenog zupčanika u polarnom koordinatnom sustavu. S obzirom da je potrebno nacrtati zupčanike u zahvatu, središte gonjenog zupčanika mora biti za osni razmak udaljeno od ishodišta koje

predstavlja središte pogonskog zupčanika. Zato se prilikom transformacije koordinata točaka evolvente i trohoide koje opisuju bokove gonjenog zupčanika iz polarnog u pravokutni koordinatni sustav vrši preslikavanje točaka preko pravca koji je okomit na os x i prolazi polovicom osnog razmaka. Također, kako bi nakon preslikavanja preko pravca koji prolazi polovicom osnog razmaka odgovarajući bokovi bili u pravom položaju, za razliku od pogonskog zupčanika kod kojeg se *COAST* bok preslikava preko osi x , kod gonjenog se zupčanika *DRIVE* bok preslikava preko osi x . Na taj način nakon preslikavanja bokova gonjenog zupčanika preko pravca koji prolazi polovicom osnog razmaka *DRIVE* bokovi pogonskog i gonjenog zupčanika nalaze se jedan nasuprot drugog te se zakretanjem zupčanika za odgovarajući kut mogu dovesti u zahvat. Zbog toga jednadžbe za transformaciju koordinata točaka *DRIVE* boka gonjenog zupčanika glase:

$$x_{D1} = a - r \cdot \cos(\eta - \gamma), \quad (70)$$

$$y_{D1} = -r \cdot \sin(\eta - \gamma). \quad (71)$$

Jednadžbe za transformaciju koordinata točaka *COAST* boka gonjenog zupčanika glase:

$$x_{C1} = a - r \cdot \cos(\eta - \gamma), \quad (72)$$

$$y_{C1} = r \cdot \sin(\eta - \gamma). \quad (73)$$

Kao i kod pogonskog zupčanika, koordinate točaka spremaju se u odgovarajuće nizove koji onda čine skup točaka pomoću kojih se crtaju bokovi gonjenog zupčanika. Nakon povezivanja bokova lukom tjemene kružnice nacrtan je cijeli profil zuba. Prije samog modeliranja zupčanika kopiranjem i rotiranjem zuba oko središta, potrebno je zub pomaknuti u položaj koji ovisi o tome koja je karakteristična točka zahvata zupčanika odabrana. Ovisno o odabranoj točki zahvata izračunavaju se kutovi za koje je potrebno zakrenuti pogonski i gonjeni zub, odnosno cijeli zupčanik.

4.3.1. Zahvat u točki C

Kinematski pol C predstavlja točku zahvata u kojoj su u dodiru točke boka nakinematskim kružnicama. U toj se točki sijeku zahvatna crta i spojnica osi zupčanika. Kut za koji je potrebno zakrenuti pogonski zub, odnosno zupčanik računa se pomoću sljedećih izraza:

$$\xi_{C1} = \sqrt{\left(\frac{d_{w1}}{d_{b1}}\right)^2 - 1}, \quad (74)$$

$$\omega_{C1} = \xi_{C1} - \arctan \xi_{C1}. \quad (75)$$

Jednadžbe (74) i (75) su jednadžbe evolvente u polarnom koordinatnom sustavu prema [4], a i izračunati kut ω_{C1} predstavlja kut točke evolvente *DRIVE* boka zuba pogonskog zupčanika koja se nalazi na kinematskoj kružnici.

Na isti se način izračunava i kut zakreta gonjenog zupčanika za zahvat u kinematskom polu C:

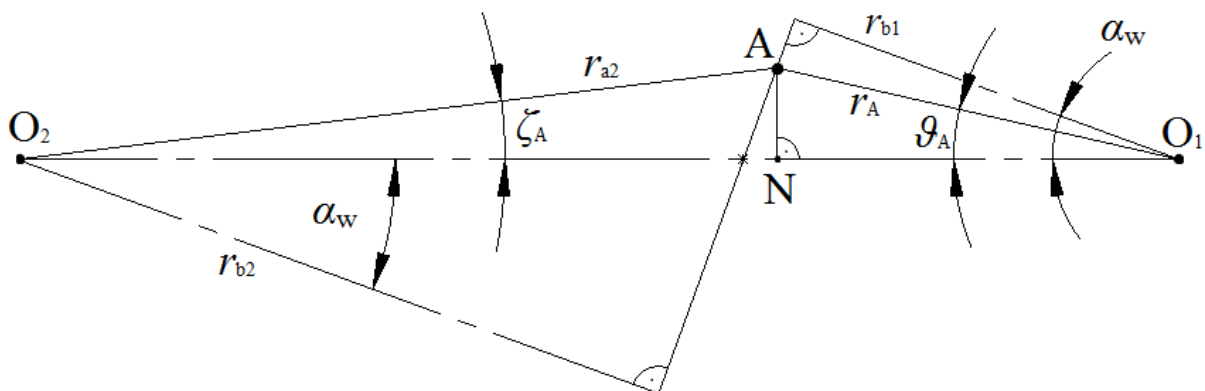
$$\xi_{C2} = \sqrt{\left(\frac{d_{w2}}{d_{b2}}\right)^2 - 1}, \quad (76)$$

$$\omega_{C2} = \xi_{C2} - \arctan \xi_{C2}. \quad (77)$$

Nakon zakreta pogonskog zupčanika za kut ω_{C1} , a gonjenog zupčanika za kut ω_{C2} zupčanici su dovedeni u položaj zahvata u kinematskom polu C.

4.3.2. Zahvat u točki A

Početna točka zahvata A određena je kao sjecište tjemene kružnice gonjenog zupčanika i zahvatne crte [Slika 32].



Slika 32. Točka zahvata A

Kut ζ_A izračunava se pomoću izraza:

$$\zeta_A = \cos^{-1}\left(\frac{r_{b2}}{r_{a2}}\right) - \alpha_w. \quad (78)$$

Udaljenost između točaka O_1 i N označena je kao r_{Ax} , a izračunava se pomoću izraza:

$$r_{Ax} = a - r_{a2} \cdot \cos \zeta_A. \quad (79)$$

Kut ϑ_A moguće je izračunati iz pravokutnog trokuta ΔO_1NA pomoću jednadžbe:

$$\vartheta_A = \arctan \left(\frac{r_{a2} \cdot \sin \zeta_A}{r_{Ax}} \right). \quad (80)$$

Iz istog pravokutnog trokuta ΔO_1NA sada je moguće izračunati polumjer r_A točke na evolventnom dijelu boka pogonskog zupčanika koji dolazi u dodir s gonjenim zupčanikom:

$$r_A = \frac{r_{Ax}}{\cos \vartheta_A}. \quad (81)$$

Parametar evolvente koji odgovara kutu na kojem se nalazi točka evolvente polumjera r_A izračunava se pomoću jednadžbe:

$$\xi_{A1} = \sqrt{\left(\frac{r_A}{r_{b1}} \right)^2 - 1}. \quad (82)$$

Kut za koji je potrebno zakrenuti pogonski zupčanik u slučaju zahvata u točki A tada iznosi:

$$\omega_{A1} = \gamma_1 - (\xi_{A1} - \arctan \xi_{A1}) + \vartheta_A. \quad (83)$$

Parametar evolvente koji odgovara kutu na kojem se nalazi točka evolvente na tjemenu polumjeru gonjenog zupčanika izračunava se pomoću jednadžbe:

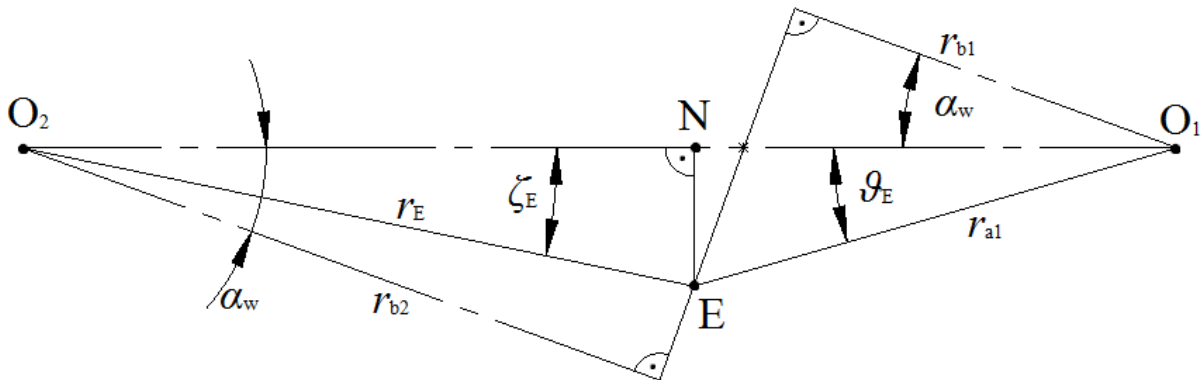
$$\xi_{A2} = \sqrt{\left(\frac{r_{a2}}{r_{b2}} \right)^2 - 1}. \quad (84)$$

Kut za koji je potrebno zakrenuti gonjeni zupčanik u slučaju zahvata u točki A tada iznosi:

$$\omega_{A2} = \gamma_2 - (\xi_{A2} - \arctan \xi_{A2}) - \zeta_A. \quad (85)$$

4.3.3. Zahvat u točki E

Završna točka zahvata E određena je kao sjecište tjemene kružnice pogonskog zupčanika i zahvatne crte [Slika 32].



Slika 33. Točka zahvata E

Kut g_E izračunava se pomoću izraza:

$$g_E = \cos^{-1} \left(\frac{r_{b1}}{r_{a1}} \right) - \alpha_w. \quad (86)$$

Udaljenost između točaka O_2 i N označena je kao r_{Ex} , a izračunava se pomoću izraza:

$$r_{Ex} = a - r_{a1} \cdot \cos g_E. \quad (87)$$

Kut ζ_E moguće je izračunati iz pravokutnog trokuta ΔO_2NA pomoću jednadžbe:

$$\zeta_E = \arctan \left(\frac{r_{a1} \cdot \sin g_E}{r_{Ex}} \right). \quad (88)$$

Iz istog pravokutnog trokuta ΔO_2NA sada je moguće izračunati polumjer r_E točke na evolventnom dijelu boka gonjenog zupčanika koji je u dodiru s pogonskim zupčanikom:

$$r_E = \frac{r_{Ex}}{\cos \zeta_E}. \quad (89)$$

Parametar evolvente koji odgovara kutu na kojem se nalazi točka evolvente polumjera r_E izračunava se pomoću jednadžbe:

$$\xi_{E2} = \sqrt{\left(\frac{r_E}{r_{b2}}\right)^2} - 1. \quad (90)$$

Kut za koji je potrebno zakrenuti gonjeni zupčanik u slučaju zahvata u točki E tada iznosi:

$$\omega_{E2} = \gamma_2 - (\xi_{E2} - \arctan \xi_{E2}) + \zeta_E. \quad (91)$$

Parametar evolvente koji odgovara kutu na kojem se nalazi točka evolvente na tjemenu polumjeru pogonskog zupčanika izračunava se pomoću jednadžbe:

$$\xi_{E1} = \sqrt{\left(\frac{r_{a1}}{r_{b1}}\right)^2} - 1. \quad (92)$$

Kut za koji je potrebno zakrenuti pogonski zupčanik u slučaju zahvata u točki E tada iznosi:

$$\omega_{E1} = \gamma_1 - (\xi_{E1} - \arctan \xi_{E1}) - \zeta_E. \quad (93)$$

4.3.4. Zahvat u točki B

Točka B od točke E je udaljena za korak zahvatne crte koji je jednak koraku na temeljnoj kružnici. To znači da par zubi koji je u zahvatu u točki B nakon zakreta za kut $2\pi/z$ dolazi u zahvat u točki E. Prema tome, kut za koji je potrebno zakrenuti pogonski zupčanik u slučaju zahvata u točki B tada iznosi:

$$\omega_{B1} = \omega_{E1} + \frac{2\pi}{z_1}, \quad (94)$$

a kut za koji je potrebno zakrenuti gonjeni zupčanik u slučaju zahvata u točki B iznosi:

$$\omega_{B2} = \omega_{E2} - \frac{2\pi}{z_2}, \quad (95)$$

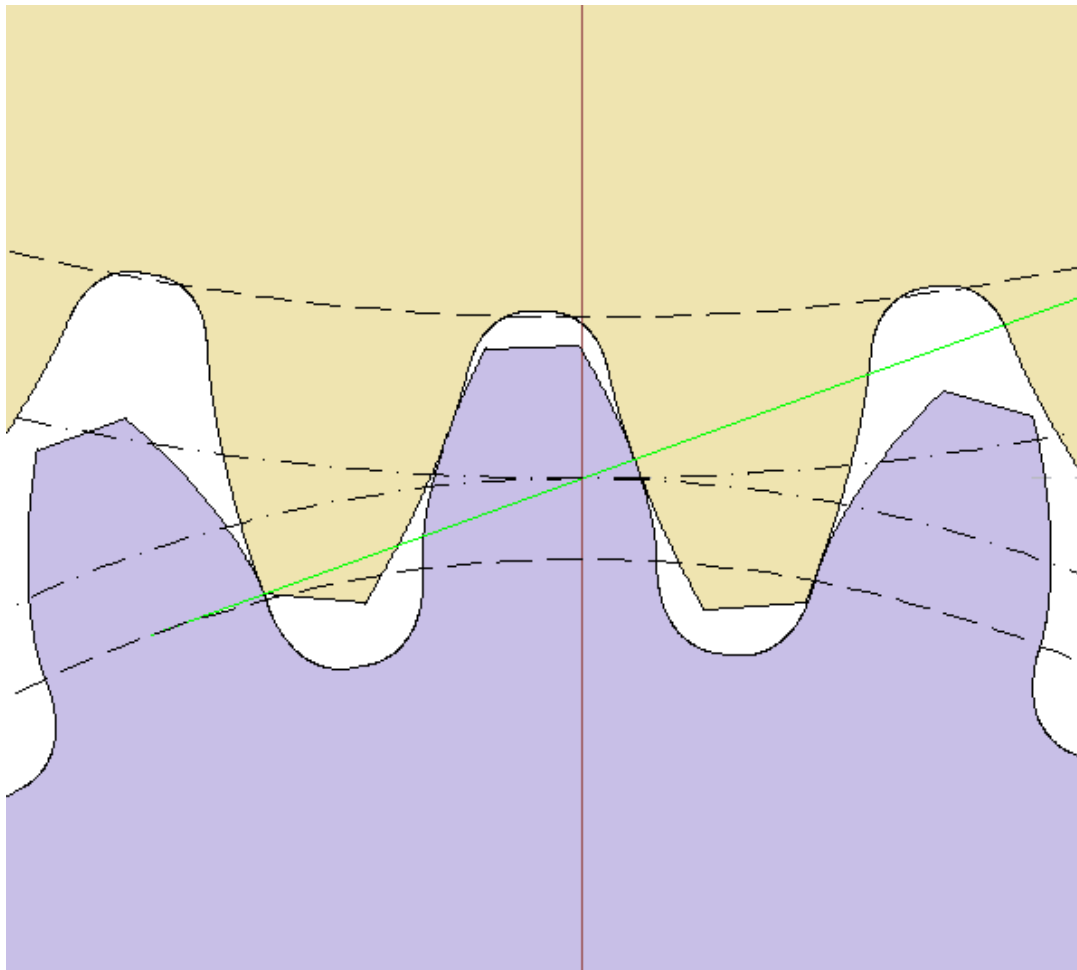
4.3.5. Zahvat u točki D

Slično kao kod zahvata u točki B, točka D od točke A udaljena je za korak zahvatne crte koji je jednak koraku na temeljnoj kružnici. To znači da par zubi koji je u zahvatu u točki A nakon zakreta za kut $2\pi/z$ dolazi u zahvat u točki D. Prema tome, kut za koji je potrebno zakrenuti pogonski zupčanik u slučaju zahvata u točki D tada iznosi:

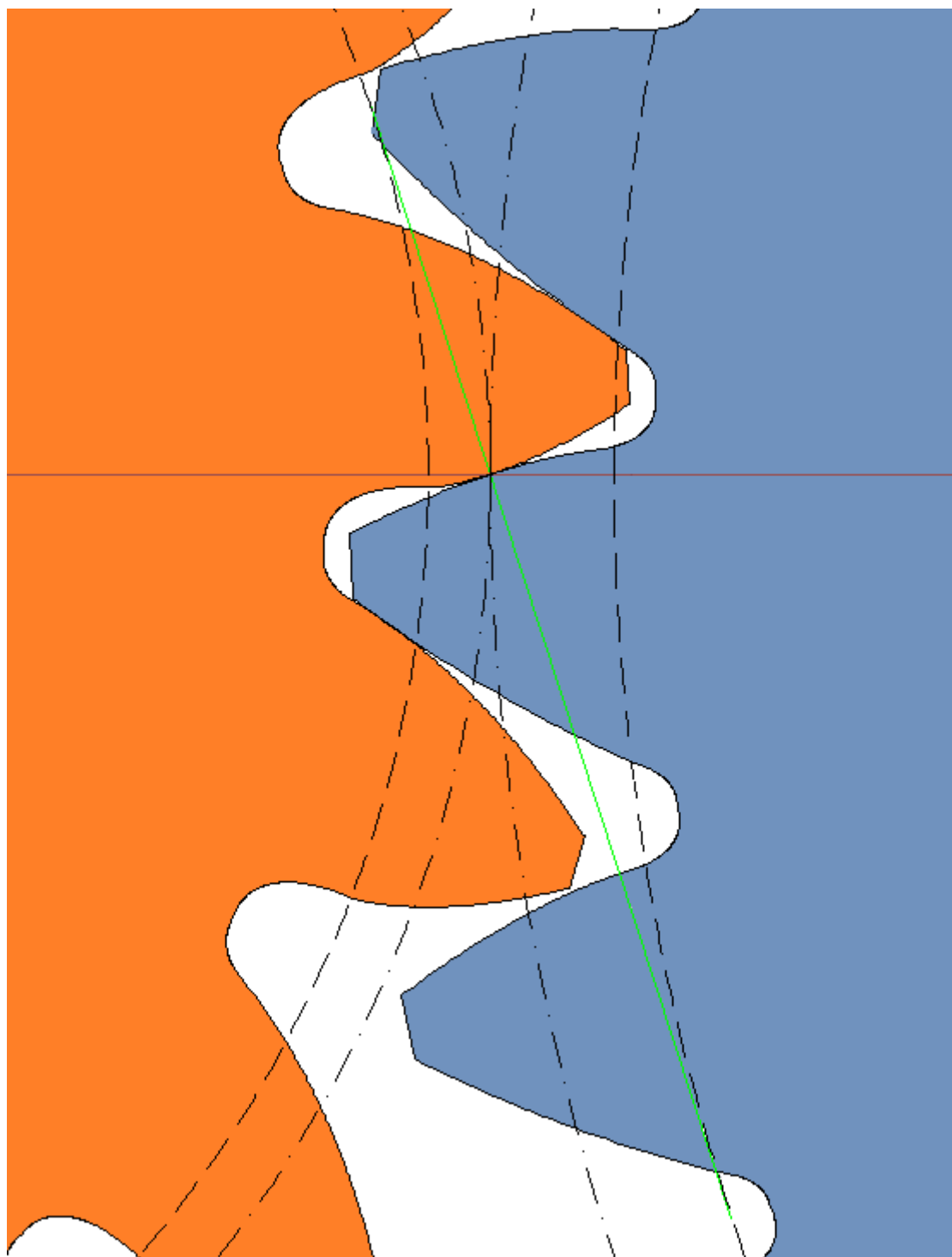
$$\omega_{D1} = \omega_{A1} - \frac{2\pi}{z_1}, \quad (96)$$

a kut za koji je potrebno zakrenuti gonjeni zupčanik u slučaju zahvata u točki D iznosi:

$$\omega_{D2} = \omega_{A2} + \frac{2\pi}{z_2}, \quad (97)$$



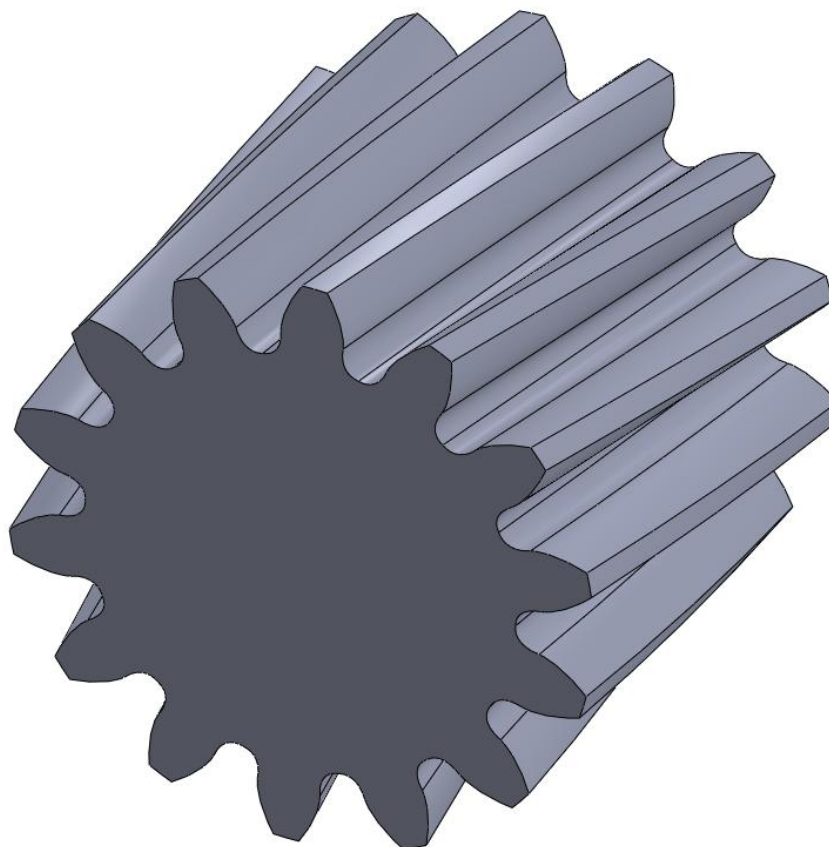
Slika 34. Zahvat simetričnih zubi u točki A



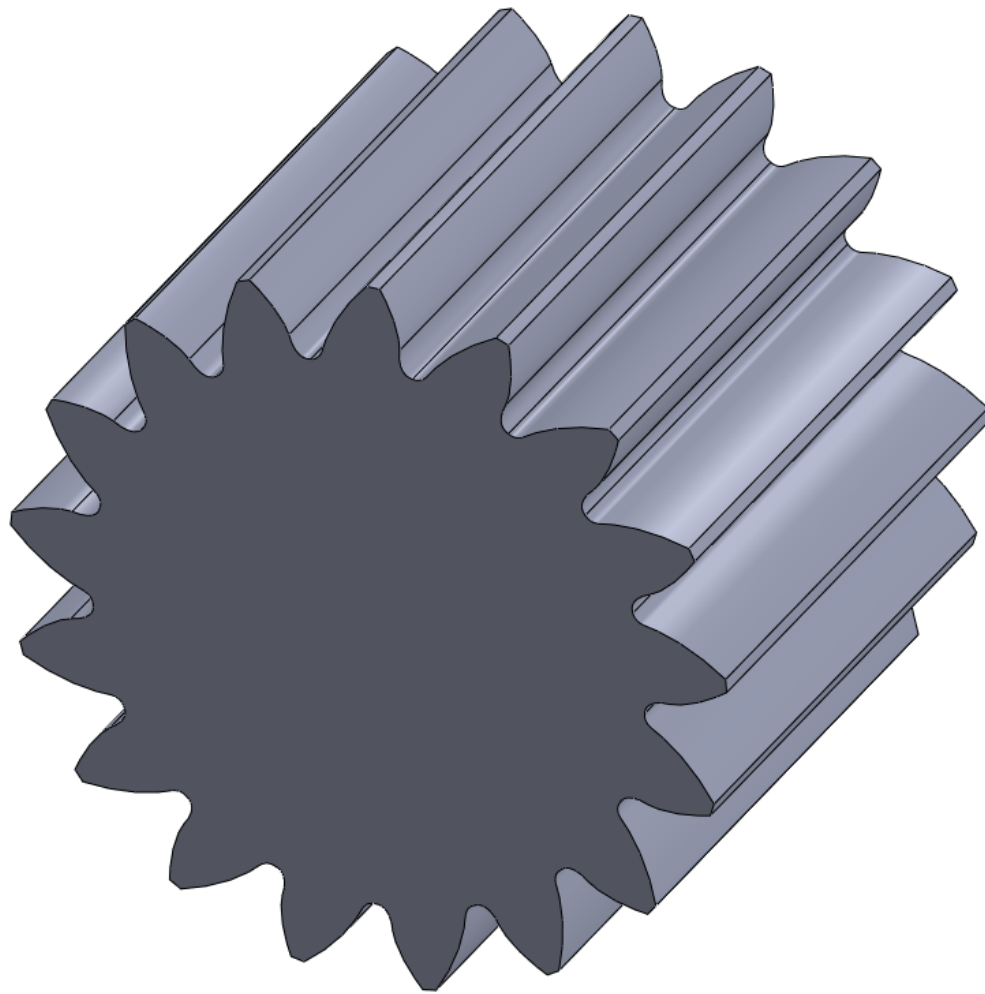
Slika 35. Zahvat asimetričnih zubi u točki C

5. TRODIMENZIJSKI MODEL ZUPČANIKA

Kao što je već ranije rečeno, dvodimenzijski model čini temelj trodimenzijskog modela. U tom je smislu, procedura izrađena u ovom radu početni korak u procesu izrade trodimenzijskog modela i provedbe numeričkih analiza i simulacija. Programski paket *AutoCAD* pruža mogućnost pohrane crteža u DXF formatu koji je moguće otvoriti u ostalim programskim paketima za trodimenzijsko modeliranje i provedbu numeričkih analiza. Modeli prikazani na [Slika 36] i [Slika 37] izrađeni su u programskom paketu *Solidworks*. Najprije su pomoću programske procedure izrađeni dvodimenzijski modeli zupčanika te su pohranjeni u DXF formatu. Zatim su ti modeli učitani u programski paket *Solidworks* i od njih su izrađeni trodimenzijski modeli. Važno je napomenuti da programski paket *Solidworks* prilikom učitavanja DXF datoteke nudi mogućnost popravka modela koji se učitava. Ta opcija nije korištena, već su dvodimenzijski modeli učitani bez promjena i kao takvi su se pokazali preciznima s obzirom da su se pomoću njih uspješno izradili trodimenzijski modeli.



Slika 36. Zupčanik sa simetričnim kosim zubima



Slika 37. Zupčanik s asimetričnim ravnim zubima

6. ZAKLJUČAK

U ovome radu uspješno je izrađena programska procedura za parametarsko modeliranje dvodimenzijskog modela zupčanika u čeonj ravnini u programskom paketu *AutoCAD*.

Napredak industrije zahtijeva proizvodnju zupčanika koji prenose sve veće snage uz što manju dimenziju i masu, što dovodi do znatnog porasta naprezanja zuba zupčanika. Eksperimentalne metode kojima bi se ispitivali takvi novi zupčanici skupe su, zahtijevaju puno rada, a obično su i dugotrajne. Napredak računalne tehnologije, kako fizičkih komponenti računala tako i programskih paketa, omogućuje provedbu sve složenijih numeričkih analiza i simulacija u sve kraćem vremenu, što znači da se sve češće koriste umjesto eksperimenata.

Pri proračunu i oblikovanju zupčanika sve se više koriste numeričke metode, poglavito metoda konačnih elemenata, za koju je potrebno izraditi trodimenzijski model zupčanika. Takav model treba biti što sličniji stvarnom zupčaniku kako bi i rezultati numeričke analize bili što kvalitetniji. Temelj svakog trodimenzijskog modela je dvodimenzijski model u ravnini. Kao temelj, dvodimenzijski model mora biti što precizniji kako bi se omogućila izrada kvalitetnog trodimenzijskog modela i provedba numeričke analize koja će dati točne rezultate. Ručna izrada dvodimenzijskih modela spora je i podložna greškama. Programska procedura izrađena u sklopu ovog rada rješava upravo taj problem jer omogućuje brzo i točno parametarsko modeliranje dvodimenzijskog modela zupčanika uzimajući u obzir relevantne parametre kao što su normalni modul, faktor pomaka profila, kut nagiba boka zuba itd.

Ova programska procedura otvara brojne mogućnosti, predstavlja temelj i početak za potencijalno daljnje razvijanje i izradu sličnih procedura unutar samih programskih paketa za numeričku analizu čime bi se dodatno ubrzao cjelokupni proces numeričke analize i proračuna.

LITERATURA

- [1] Litvin, F. L.; Fuentes, A.: Gear geometry and applied theory, Cambridge University Press, The Edinburgh Building, Cambridge, 2004.
- [2] Oberšmit, E.: Ozubljenja i zupčanici, Sveučilišna naklada Liber, Zagreb, 1982.
- [3] Litvin, F. L.: Development of gear technology and theory of gearing, NASA Reference publication 1406, 1997.
- [4] ISO 6336-2, Calculation of Load Capacity of Spur and Helical Gears, 2006.
- [5] ISO 53, Calculation Cylindrical gears for general and heavy engineering - Standard basic rack tooth profile, 1998.
- [6] DIN 3972, Bezugsprofile von Verzahnwerkzeugen für Evolventen-Verzahnungen nach DIN 867, 1952.
- [7] Kapelevich A. L., Kleiss R. E.: Direct Gear Design for Spur and Helical Involute Gears, Gear Technology, September/October, 29-35, 2002.
- [8] Kapelevich A. L.: Geometry and design of involute spur gears with asymmetric teeth, Mechanism and Machine Theory, Issue 35, 117–130, 2000.
- [9] Kapelevich A. L.: Measurement of Directly Designed Gears with Symmetric and Asymmetric Teeth, Gear Technology, January/February, 60-65, 2011.

PRILOZI

- I. CD-R disc
- II. Programski kod

PRILOG II – Programski kod:

```

PrivateSub NACRTAJ_Click()
Dim sve As AcadSelectionSet
Set sve = ThisDrawing.SelectionSets.Add("ToErase")
sve.Select acSelectionSetAll
sve.Erase
sve.Delete
Dim z, z2, k, i, br, brtr, brinv, bro, brto, brtotroh1, brtoinv1,
brtotroh2, brtoinv2 AsInteger
Dim m, d, pom, h, ro, ha, alfan1, alfan2, alfat1, alfat2, beta, db1, db2,
da, df, A1, A2, under1, under2, B1, rsoi, fi, rtroh, etatroh, ksii AsDouble
Dim xtroh, ytroh, xtroh2, ytroh2, rinv, etainv, xinv, yinv, xinv2, yinv2,
ksii1, ksii2, korak, x0, x1, fi1, fi0, korak1, korakf, kut, kut11, kut12,
kut21, kut22 AsDouble
Dim etal1, eta12, eta21, eta22, s1, s2, ksi1, ksi2, xkf, ykf, xf, yf, j,
preciznost, omikron, ost, cp, romax1, romax2, osraz, alfatw, dw1 AsDouble
Dim tockee1()AsDouble
Dim tockeeetroh1()AsDouble
Dim tockeeinv1()AsDouble
Dim tockee2()AsDouble
Dim tockeeetroh2()AsDouble
Dim tockeeinv2()AsDouble
Dim tockel AsNew Collection
Dim tocketroh1 AsNew Collection
Dim tockeeinv1 AsNew Collection
Dim tockee2 AsNew Collection
Dim tocketroh2 AsNew Collection
Dim tockeeinv2 AsNew Collection
Dim PI AsDouble

    PI = Atn(1)*4

OnErrorGoTo HandlerError
    m =CDBl(Replace(TextBox1.Text, ".", ","))
    z =CInt(TextBox2.Text)
    pom =CDBl(Replace(TextBox3.Text, ".", ","))
    brtr =CInt(TextBox4.Text)
    brinv =CInt(TextBox5.Text)
    beta =CDBl(Replace(TextBox10.Text, ".", ","))* PI /180

If m <=0Then
    MsgBox "Modul mora biti veći od 0!"
ExitSub
EndIf

If InStr(TextBox2.Text, ".")>0Or InStr(TextBox2.Text, ",")>0Then
    MsgBox "Broj zubi mora biti cijeli broj!"
ExitSub
EndIf

If z <=4Then
    MsgBox "Broj zubi mora biti veći od 4!"
ExitSub
EndIf

If brtr <3Or brinv <3Then
    MsgBox "Minimalni broj točaka trohoide i evolvente mora biti 3!"

```

```
ExitSub
EndIf

If beta <0Or beta >=(PI /2)Then
    MsgBox "Kut nagiba boka zuba mora biti 0 ili veći, ali manji od
90!"
ExitSub
EndIf

If InStr(TextBox4.Text, ".")>0Or InStr(TextBox4.Text, ",")>0Then
    MsgBox "Broj točaka trohoide mora biti cijeli broj!"
ExitSub
EndIf

If InStr(TextBox5.Text, ".")>0Or InStr(TextBox5.Text, ",")>0Then
    MsgBox "Broj točaka evolvente mora biti cijeli broj!"
ExitSub
EndIf

If CheckBox2.Value =TrueThen
If OptionButton1.Value =TrueThen
    alfan1 =20* PI /180
    alfan2 =20* PI /180
    ha =1
    h =1.25
    ro =0.38
ElseIf OptionButton2.Value =TrueThen
    alfan1 =20* PI /180
    alfan2 =20* PI /180
    ha =1
    h =1.25
    ro =0.3
ElseIf OptionButton3.Value =TrueThen
    alfan1 =20* PI /180
    alfan2 =20* PI /180
    ha =1
    h =1.25
    ro =0.25
ElseIf OptionButton4.Value =TrueThen
    alfan1 =20* PI /180
    alfan2 =20* PI /180
    ha =1
    h =1.4
    ro =0.39
ElseIf OptionButton6.Value =TrueThen
    alfan1 =20* PI /180
    alfan2 =20* PI /180
    ha =1
    h =1.167
    ro =0.2
ElseIf OptionButton7.Value =TrueThen
    alfan1 =20* PI /180
    alfan2 =20* PI /180
    ha =1
    h =1.25
    ro =0.2
ElseIf OptionButton5.Value =TrueThen
OnErrorGoTo HandlerError
```

```
        alfan1 =Cdbl (Replace (TextBox6.Text, ".", ", ")) * PI /180
        alfan2 =Cdbl (Replace (TextBox6.Text, ".", ", ")) * PI /180
        ha =Cdbl (Replace (TextBox7.Text, ".", ", "))
        h =Cdbl (Replace (TextBox8.Text, ".", ", "))
        ro =Cdbl (Replace (TextBox9.Text, ".", ", "))
EndIf

If ha <=0Or h <=0Then
    MsgBox "Faktor podnožne i tjemene visine zuba ozubnice mora biti
veći od 0!"
ExitSub
EndIf

If ro <=0Then
    MsgBox "Faktor polumjera zaobljenja podnožja ozubnice mora biti
veći od 0!"
ExitSub
EndIf

If alfan1 <=0Then
    MsgBox "Kut nagiba zuba ozubnice mora biti veći od 0!"
ExitSub
EndIf

ElseIf CheckBox3.Value =TrueThen
    alfan1 =Cdbl (Replace (TextBox11.Text, ".", ", ")) * PI /180
    alfan2 =Cdbl (Replace (TextBox12.Text, ".", ", ")) * PI /180
    ha =Cdbl (Replace (TextBox13.Text, ".", ", "))
    h =Cdbl (Replace (TextBox14.Text, ".", ", "))
    ro =Cdbl (Replace (TextBox15.Text, ".", ", "))

If ha <=0Or h <=0Then
    MsgBox "Faktor podnožne i tjemene visine zuba ozubnice mora
biti veći od 0!"
ExitSub
EndIf

If ro <=0Then
    MsgBox "Faktor polumjera zaobljenja podnožja ozubnice mora
biti veći od 0!"
ExitSub
EndIf

If alfan1 <=0Then
    MsgBox "Kut nagiba zuba ozubnice mora biti veći od 0!"
ExitSub
EndIf

If alfan2 <=0Then
    MsgBox "Kut nagiba zuba ozubnice mora biti veći od 0!"
ExitSub
EndIf
EndIf

d = (m / Cos (beta)) * z
alfat1 = Atn (Tan (alfan1) / Cos (beta))
alfat2 = Atn (Tan (alfan2) / Cos (beta))
dbl = d * Cos (alfat1)
```

```

db2 = d * Cos(alfat2)
da = d +2* ha * m +2* pom * m
df = d -2* h * m +2* pom * m

If CheckBox4.Value =TrueThen
    osraz =Cdbl(Replace(TextBox16.Text, ".", ", "))
    z2 =CInt(TextBox17.Text)

If osraz <=0Then
    MsgBox "Osni razmak mora biti veći od 0!"
ExitSub
EndIf

If z2 <=4Then
    MsgBox "Broj zubi mora biti veći od 4!"
ExitSub
EndIf

If InStr(TextBox2.Text, ".")>0Or InStr(TextBox2.Text, ",")>0Then
    MsgBox "Broj zubi mora biti cijeli broj!"
ExitSub
EndIf

    alfatw = Acos((m / Cos(beta))*((z + z2)/(2* osraz))*
Cos(alfat1))
    dw1 = d *(Cos(alfat1)/ Cos(alfatw))

    MsgBox "Pogonski kut zahvatne crte iznosi: "+CStr(alfatw *180/
PI)+" stupnjeva"

'dimenzije drugog zupčanika:
Dim d2, pom2, alfan3, alfan4, alfat3, alfat4, db3, db4, da2, df2, dw4
AsDouble

    alfan3 = alfan2
    alfan4 = alfan1

    d2 =(m / Cos(beta))* z2
    alfat3 = Atn(Tan(alfan3)/ Cos(beta))
    alfat4 = Atn(Tan(alfan4)/ Cos(beta))
    db3 = d2 * Cos(alfat3)
    db4 = d2 * Cos(alfat4)

    pom2 =((z + z2)*((Tan(alfatw)- alfatw)-(Tan(alfat4)-
alfat4))/(2* Tan(alfan4)))- pom
    MsgBox "Faktor pomaka gonjenog zupčanika iznosi:
"+CStr(pom2)

    da2 = d2 +2* ha * m +2* pom2 * m
    df2 = d2 -2* h * m +2* pom2 * m

    dw4 =2* osraz - dw1

EndIf

    A1 = ro * m /(Cos(alfan1)* Cos(beta))+(h * m - pom * m - ro * m)*
Tan(alfat1)

```

```

    A2 = ro * m / (Cos(alfan2) * Cos(beta)) + (h * m - pom * m - ro * m) *
Tan(alfat2)

'uvjet podrezanosti:
    under1 = (d / 2) * (Sin(alfat1))^2 - (h * m - pom * m - ro * m * (1 -
Sin(alfan1)))
    under2 = (d / 2) * (Sin(alfat2))^2 - (h * m - pom * m - ro * m * (1 -
Sin(alfan2)))

    fil = PI / 2

'kut između simetrala susjednih zubi:
    kut = 2 * PI / z

'debljina zuba na diobenom promjeru:
    s1 = (m / Cos(beta)) * (PI / 2 + 2 * pom * Tan(alfan1))
    s2 = (m / Cos(beta)) * (PI / 2 + 2 * pom * Tan(alfan2))

'kut između simetrale zuba i točke sjecišta boka zuba i diobene kružnice:
    eta11 = (s1 / (d / 2)) / 2
    eta12 = (s2 / (d / 2)) / 2

'parametar evolvente za točku na diobenoj kružnici:
    ksi1 = Sqr(((d / 2) / (db1 / 2))^2 - 1)
    ksi2 = Sqr(((d / 2) / (db2 / 2))^2 - 1)

'kut između točke na sjecištu boka zuba i diobene kružnice i osi x; formula
evolvente:
    eta21 = ksi1 - Atn(ksi1)
    eta22 = ksi2 - Atn(ksi2)

'kut prve točke podnožne kružnice/točke na sredini uzubine:
    kut11 = eta11 + eta21 - (kut / 2)
    kut12 = eta12 + eta22 - (kut / 2)

'kut zadnje točke podnožne kružnice/prve točke trohoide; formula trohoide:
    kut21 = theta(fil, m, ro, alfat1, d, A1, pom, h, beta) +
epsilon(fil, d, m, ro, pom, h, beta) - alfat1
    kut22 = theta(fil, m, ro, alfat2, d, A2, pom, h, beta) +
epsilon(fil, d, m, ro, pom, h, beta) - alfat2

'kut između simetrale zuba i osi x:
omega1 = eta11 + eta21
omega2 = eta12 + eta22

omikron = omega1 + omega2 - kut21 - kut22
'omikron = Abs(kut21) + Abs(kut22) + omega1 + omega2

If omikron < kut Then
    ost = kut - omikron
    korakf = (ost / 2) / 4
'korakf = (kut21 - kut11) / 4
For j = (kut21 - (ost / 2)) To kut21 Step korakf
'For j = kut11 To kut21 Step korakf
    xf = (df / 2) * Cos(j - omega1)
    yf = (df / 2) * Sin(j - omega1)

If (CheckBox1.Value = False) And (under1 >= 0) Then

```



```

        tockel.Add (xf)
        tockel.Add (yf)
        tockel.Add (0)
Else
        tocketroh1.Add (xf)
        tocketroh1.Add (yf)
        tocketroh1.Add (0)
EndIf
Next
'korakf = (kut22 - kut12) / 4
For j =(kut22 - (ost /2))To kut22 Step korakf
'For j = kut12 To kut22 Step korakf
        xkf =(df /2)* Cos(j - omega2)
        ykf =(df /2)* Sin(j - omega2)
'preslikavanje preko osi x:
        xf = xkf
        yf =-ykf
If (CheckBox1.Value =False)And (under2 >=0) Then
        tocke2.Add (xf)
        tocke2.Add (yf)
        tocke2.Add (0)
Else
        tocketroh2.Add (xf)
        tocketroh2.Add (yf)
        tocketroh2.Add (0)
EndIf
Next

ElseIf omikron = kut Then
ResumeNext

ElseIf omikron > kut Then
        MsgBox "Zupčanik je nemoguće izraditi odvalnim glodanjem! Na
sredini uzubine nalazi se sjecište podnožnih krivulja (trohoida) susjednih
zubi!"
ExitSub
EndIf

'nepodrezani bok1:
If under1 >=0Then
        B1 = B(alfan1, m, h, ro, pom)
        rsoi = Sqr(((d /2)- B1)^2+(B1 * Cos(beta)/ Tan(alfan1))^2)

If CheckBox1.Value =TrueThen
        korak1 =(fil - alfan1)/(brtr -1)
Else
        korak1 =(fil - (alfan1 +0.025))/(brtr -1)
EndIf

        fi = fil
For i =1To (brtr -1)
rtroh = Sqr((d /2- B(fi, m, h, ro, pom))^2+((B(fi, m, h, ro, pom)*
Cos(beta))/ Tan(fi))^2)
etatroh = theta(fi, m, ro, alfat1, d, A1, pom, h, beta)+ epsilon(fi, d, m,
ro, pom, h, beta)- alfat1
        xtrroh = rtroh * Cos(etatroh - omegal)
        ytrroh = rtroh * Sin(etatroh - omegal)

```

```

If CheckBox1.Value =TrueThen
    tocketroh1.Add (xtroh)
    tocketroh1.Add (ytroh)
    tocketroh1.Add (0)
Else
    tockel1.Add (xtroh)
    tockel1.Add (ytroh)
    tockel1.Add (0)
EndIf
fi = fi - korak1
Next

If CheckBox1.Value =TrueThen
    fi = alfan1
Else
    fi = alfan1 +0.025
EndIf
rtroh = Sqr((d /2- B(fi, m, h, ro, pom))^2+((B(fi, m, h, ro, pom)*
Cos(beta))/ Tan(fi))^2)
etatroh = theta(fi, m, ro, alfat1, d, A1, pom, h, beta)+ epsilon(fi, d, m,
ro, pom, h, beta)- alfat1
    xtroh = rtroh * Cos(etatroh - omegal)
    ytroh = rtroh * Sin(etatroh - omegal)
If CheckBox1.Value =TrueThen
    tocketroh1.Add (xtroh)
    tocketroh1.Add (ytroh)
    tocketroh1.Add (0)
Else
    tockel1.Add (xtroh)
    tockel1.Add (ytroh)
    tockel1.Add (0)
EndIf

ksii1 = Sqr((rsoi / (db1 /2))^2-1)'parametar prve točke
evolvente
ksii2 = Sqr(((da /2)/(db1 /2))^2-1)'parametar zadnje točke
evolvente

korak =(ksii2 - ksii1)/(brinv -1)

ksii = ksii1
For i =1To (brinv -1)
    rinov =(db1 /2)* Sqr(1+(ksii)^2)
    etainv = ksii - Atn(ksii)
    xinov = rinov * Cos(etainv - omegal)
    yinov = rinov * Sin(etainv - omegal)

If CheckBox1.Value =TrueThen
    tockeinv1.Add (xinov)
    tockeinv1.Add (yinov)
    tockeinv1.Add (0)
Else
    tockel1.Add (xinov)
    tockel1.Add (yinov)
    tockel1.Add (0)
EndIf

ksii = ksii + korak
Next

```

```

    rinv =(dbl /2)* Sqr(1+(ksii2)^2)
    etainv = ksii2 - Atn(ksii2)
    xinv = rinv * Cos(etainv - omegal)
    yinv = rinv * Sin(etainv - omegal)
If CheckBox1.Value =TrueThen
    tockeinv1.Add (xinv)
    tockeinv1.Add (yinv)
    tockeinv1.Add (0)

Else
    tockel.Add (xinv)
    tockel.Add (yinv)
    tockel.Add (0)

EndIf

'podrezani bok1:
Else
    fi0 = bisekcija(m, ro, alfat1, d, A1, pom, dbl, h, beta)

If CheckBox3.Value =TrueThen
    MsgBox ("Sjecište evolvente i trohoide DRIVE boka određeno je
pomoću vrijednosti nultočke: "+CStr(f(fi0, m, ro, alfat1, d, A1, pom, dbl,
h, beta)))
Else
    MsgBox ("Sjecište evolvente i trohoide određeno je pomoću
vrijednosti nultočke: "+CStr(f(fi0, m, ro, alfat1, d, A1, pom, dbl, h,
beta)))
EndIf

    B1 = B(fi0, m, h, ro, pom)
    rsoi = Sqr(((d /2)- B1)^2+(B1 * Cos(beta)/ Tan(fi0))^2)

    korak1 =(fi1 - fi0)/(brtr -1)

    fi = fi1
For i =1To(brtr -1)
    rtroh = Sqr((d /2- B(fi, m, h, ro, pom))^2+((B(fi, m, h, ro, pom)*
Cos(beta))/ Tan(fi))^2)
    etatroh = theta(fi, m, ro, alfat1, d, A1, pom, h, beta)+ epsilon(fi, d, m,
ro, pom, h, beta)- alfat1
        xtroh = rtroh * Cos(etatroh - omegal)
        ytroh = rtroh * Sin(etatroh - omegal)
        tocketroh1.Add (xtroh)
        tocketroh1.Add (ytroh)
        tocketroh1.Add (0)
        fi = fi - korak1

Next

    fi = fi0
    rtroh = Sqr((d /2- B(fi, m, h, ro, pom))^2+((B(fi, m, h, ro, pom)*
Cos(beta))/ Tan(fi))^2)
    etatroh = theta(fi, m, ro, alfat1, d, A1, pom, h, beta)+ epsilon(fi, d, m,
ro, pom, h, beta)- alfat1
        xtroh = rtroh * Cos(etatroh - omegal)
        ytroh = rtroh * Sin(etatroh - omegal)
        tocketroh1.Add (xtroh)
        tocketroh1.Add (ytroh)
        tocketroh1.Add (0)

```

```

ksii1 = Sqr((rsoi / (dbl / 2))^2 - 1) 'parametar prve točke
evolvente
ksii2 = Sqr(((da / 2) / (dbl / 2))^2 - 1) 'parametar zadnje točke
evolvente

korak = (ksii2 - ksii1) / (brinv - 1)

ksii = ksii1
For i = 1 To (brinv - 1)
    rinv = (dbl / 2) * Sqr(1 + (ksii)^2)
    etainv = ksii - Atn(ksii)
    xinv = rinv * Cos(etainv - omegal)
    yinv = rinv * Sin(etainv - omegal)
    tockeinv1.Add (xinv)
    tockeinv1.Add (yinv)
    tockeinv1.Add (0)
    ksii = ksii + korak
Next

rinv = (dbl / 2) * Sqr(1 + (ksii2)^2)
etainv = ksii2 - Atn(ksii2)
xinv = rinv * Cos(etainv - omegal)
yinv = rinv * Sin(etainv - omegal)
tockeinv1.Add (xinv)
tockeinv1.Add (yinv)
tockeinv1.Add (0)

EndIf

'nepodrezani bok2:
If under2 >= 0 Then
    B1 = B(alfan2, m, h, ro, pom)
    rsoi = Sqr(((d / 2) - B1)^2 + (B1 * Cos(beta) / Tan(alfan2))^2)

If CheckBox1.Value = True Then
    korak1 = (fi1 - alfan2) / (brtr - 1)
Else
    korak1 = (fi1 - (alfan2 + 0.025)) / (brtr - 1)
EndIf

fi = fi1
For i = 1 To (brtr - 1)
    rtroh = Sqr((d / 2 - B(fi, m, h, ro, pom))^2 + ((B(fi, m, h, ro, pom) *
    Cos(beta)) / Tan(fi))^2)
    etatroh = theta(fi, m, ro, alfat2, d, A2, pom, h, beta) + epsilon(fi, d, m,
    ro, pom, h, beta) - alfat2
    xtroh2 = rtroh * Cos(etatroh - omega2)
    ytroh2 = rtroh * Sin(etatroh - omega2)
    xtroh = xtroh2
    ytroh = -ytroh2

If CheckBox1.Value = True Then
    tocketroh2.Add (xtroh)
    tocketroh2.Add (ytroh)
    tocketroh2.Add (0)
Else
    tocke2.Add (xtroh)
    tocke2.Add (ytroh)

```

```

                tocke2.Add (0)
EndIf
                fi = fi - korak1
Next

If CheckBox1.Value =TrueThen
    fi = alfan2
Else
    fi = alfan2 +0.025
EndIf
rtroh = Sqr((d /2- B(fi, m, h, ro, pom))^2+((B(fi, m, h, ro, pom)*
Cos(beta))/ Tan(fi))^2)
etatroh = theta(fi, m, ro, alfat2, d, A2, pom, h, beta)+ epsilon(fi, d, m,
ro, pom, h, beta)- alfat2
    xtroh2 = rtroh * Cos(etatroh - omega2)
    ytroh2 = rtroh * Sin(etatroh - omega2)
    xtroh = xtroh2
    ytroh =-ytroh2

If CheckBox1.Value =TrueThen
    tocketroh2.Add (xtroh)
    tocketroh2.Add (ytroh)
    tocketroh2.Add (0)
Else
    tocke2.Add (xtroh)
    tocke2.Add (ytroh)
    tocke2.Add (0)
EndIf

ksii1 = Sqr((rsoi / (db2 /2))^2-1)'parametar prve točke
evolvente
ksii2 = Sqr(((da /2)/(db2 /2))^2-1)'parametar zadnje točke
evolvente

korak =(ksii2 - ksii1)/(brinv -1)

ksii = ksii1
For i =1To (brinv -1)
    rinvs = (db2 /2) * Sqr(1+(ksii)^2)
    etainvs = ksii - Atn(ksii)
    xinvs2 = rinvs * Cos(etainvs - omega2)
    yinvs2 = rinvs * Sin(etainvs - omega2)
    xinvs = xinvs2
    yinvs =-yinvs2

If CheckBox1.Value =TrueThen
    tockeinvs2.Add (xinvs)
    tockeinvs2.Add (yinvs)
    tockeinvs2.Add (0)
Else
    tocke2.Add (xinvs)
    tocke2.Add (yinvs)
    tocke2.Add (0)
EndIf

    ksii = ksii + korak
Next

rinvs = (db2 /2) * Sqr(1+(ksii2)^2)

```

```

    etainv = ksii2 - Atn(ksii2)
    xinv2 = rinv * Cos(etainv - omega2)
    yinv2 = rinv * Sin(etainv - omega2)
    xinv = xinv2
    yinv = -yinv2

If CheckBox1.Value =TrueThen
    tockeinv2.Add (xinv)
    tockeinv2.Add (yinv)
    tockeinv2.Add (0)

Else
    tocke2.Add (xinv)
    tocke2.Add (yinv)
    tocke2.Add (0)

EndIf

'podrezani bok2:
Else
    fi0 = bisekcija(m, ro, alfat2, d, A2, pom, db2, h, beta)

If CheckBox3.Value =TrueThen
    MsgBox ("Sjecište evolvente i trohoide COAST boka određeno je
pomoću vrijednosti nultočke: "+CStr(f(fi0, m, ro, alfat2, d, A2, pom, db2,
h, beta)))
EndIf

    B1 = B(fi0, m, h, ro, pom)
    rsoi = Sqr(((d /2)- B1)^2+(B1 * Cos(beta)/ Tan(fi0))^2)

    korak1 =(fi1 - fi0)/(brtr -1)

    fi = fi1
For i =1To(brtr -1)
    rtroh = Sqr((d /2- B(fi, m, h, ro, pom))^2+((B(fi, m, h, ro, pom)*
Cos(beta))/ Tan(fi))^2)
    etatroh = theta(fi, m, ro, alfat2, d, A2, pom, h, beta)+ epsilon(fi, d, m,
ro, pom, h, beta)- alfat2
        xtroh2 = rtroh * Cos(etatroh - omega2)
        ytroh2 = rtroh * Sin(etatroh - omega2)
        xtroh = xtroh2
        ytroh = -ytroh2

        tocketroh2.Add (xtroh)
        tocketroh2.Add (ytroh)
        tocketroh2.Add (0)
        fi = fi - korak1

Next

    fi = fi0
    rtroh = Sqr((d /2- B(fi, m, h, ro, pom))^2+((B(fi, m, h, ro, pom)*
Cos(beta))/ Tan(fi))^2)
    etatroh = theta(fi, m, ro, alfat2, d, A2, pom, h, beta)+ epsilon(fi, d, m,
ro, pom, h, beta)- alfat2
        xtroh2 = rtroh * Cos(etatroh - omega2)
        ytroh2 = rtroh * Sin(etatroh - omega2)
        xtroh = xtroh2
        ytroh = -ytroh2

```

```

        tocketroh2.Add (xtroh)
        tocketroh2.Add (ytroh)
        tocketroh2.Add (0)

        ksii1 = Sqr((rsoi / (db2 / 2))^2 - 1) 'parametar prve točke
evolvente
        ksii2 = Sqr(((da / 2) / (db2 / 2))^2 - 1) 'parametar zadnje točke
evolvente

        korak = (ksii2 - ksii1) / (brinv - 1)

        ksii = ksii1
For i = 1 To (brinv - 1)
        rinv = (db2 / 2) * Sqr(1 + (ksii)^2)
        etainv = ksii - Atn(ksii)
        xinv2 = rinv * Cos(etainv - omega2)
        yinv2 = rinv * Sin(etainv - omega2)
        xinv = xinv2
        yinv = -yinv2

        tockeinv2.Add (xinv)
        tockeinv2.Add (yinv)
        tockeinv2.Add (0)
        ksii = ksii + korak
Next

        rinv = (db2 / 2) * Sqr(1 + (ksii2)^2)
        etainv = ksii2 - Atn(ksii2)
        xinv2 = rinv * Cos(etainv - omega2)
        yinv2 = rinv * Sin(etainv - omega2)
        xinv = xinv2
        yinv = -yinv2

        tockeinv2.Add (xinv)
        tockeinv2.Add (yinv)
        tockeinv2.Add (0)
EndIf

'crtanje krivulja:
If CheckBox1.Value = True Or under1 < 0 Then
        brtotroh1 = tocketroh1.Count
        brtoinv1 = tockeinv1.Count
ReDim tockeetroh1(0 To brtotroh1 - 1)
ReDim tockeeinv1(0 To brtoinv1 - 1)

For bro = 0 To (brtotroh1 - 1)
        tockeetroh1(bro) = tocketroh1.Item(bro + 1)
Next

For bro = 0 To (brtoinv1 - 1)
        tockeeinv1(bro) = tockeinv1.Item(bro + 1)
Next
Else
        brto = tockel.Count
ReDim tockeel(0 To brto - 1)
For bro = 0 To (brto - 1)
        tockeel(bro) = tockel.Item(bro + 1)

```

```

Next
EndIf

If CheckBox1.Value =TrueOr under2 <0Then
    brtotroh2 = tocketroh2.Count
    brtoinv2 = tockeinv2.Count
ReDim tockeetroh2(0To brtotroh2 -1)
ReDim tockeeinv2(0To brtoinv2 -1)

For bro =0To(brtotroh2 -1)
    tockeetroh2(bro)= tocketroh2.Item(bro +1)
Next

For bro =0To(brtoinv2 -1)
    tockeeinv2(bro)= tockeinv2.Item(bro +1)
Next
Else
    brto = tocke2.Count
ReDim tockee2(0To brto -1)
For bro =0To(brto -1)
    tockee2(bro)= tocke2.Item(bro +1)
Next
EndIf

Dim omega, xp, yp, sa, gama1, gama2, delta1, delta2, ksi, kutrot, rx, rax,
zeta AsDouble
Dim startan(0To2)AsDouble
Dim endtan(0To2)AsDouble
Dim center(0To2)AsDouble
Dim luk As AcadArc
Dim rotluk AsVariant

    startan(0)=0: startan(1)=0: startan(2)=0
    endtan(0)=0: endtan(1)=0: endtan(2)=0

    center(0)=0: center(1)=0: center(2)=0

'kut točke na sjecištu evolvente i tjemene kružnice:
    delta1 = Sqr(((da /2)/(db1 /2))^2-1)
    gama1 =(delta1 - Atn(delta1))- omega1
    delta2 = Sqr(((da /2)/(db2 /2))^2-1)
    gama2 =-((delta2 - Atn(delta2))- omega2)

If gama1 > gama2 Then
    MsgBox ("Zupčanik je nemoguće izraditi jer se evolvente sijeku
prije tjemene kružnice!")
ExitSub
EndIf

'debljina zuba na tjemenoj kružnici (duljina luka):
    sa =(da /2)*(Abs(gama1)+ Abs(gama2))

If sa <=(0.2* m)Then
    MsgBox "Tjemena debljina zuba premala; zub je ZAŠILJEN! Moguća
rješenja: smanjenje pomaka profila, smanjenje faktora tjemene visine zuba,
povećanje broja zubi."
ExitSub
EndIf

```



```

If sa <=(0.4* m) Then
    MsgBox "UPOZORENJE: Tjemena debljina zuba je premala za kaljeni
zupčanik! Moguća rješenja: odabir materijala zupčanika koji se ne kali,
smanjenje pomaka profila, smanjenje faktora tjemene visine zuba, povećanje
broja zubi."
EndIf

'određivanje kutova rotacije za crtanje određenih točki zahvata zupčanika:
If CheckBox4.Value =TrueThen
If OptionButton10.Value =TrueThen
    ksi = Sqr(((dw1 /2)/(db1 /2))^2-1)
    kutrot = omegal - (ksi - Atn(ksi))

ElseIf OptionButton8.Value =TrueThen
    omega = Acos((db4 /2)/(da2 /2))- alfatw
    rx = osraz - ((da2 /2)* Cos(omega))
    zeta = Atn(((da2 /2)* Sin(omega))/ rx)
    rax = rx / Cos(zeta)
    ksi = Sqr((rax / (db1 /2))^2-1)
    kutrot = omegal - (ksi - Atn(ksi))+ zeta

ElseIf OptionButton12.Value =TrueThen
    omega = Acos((db1 /2)/(da /2))- alfatw
    ksi = Sqr(((da /2)/(db1 /2))^2-1)
    kutrot = omegal - (ksi - Atn(ksi))- omega

ElseIf OptionButton11.Value =TrueThen
    omega = Acos((db4 /2)/(da2 /2))- alfatw
    rx = osraz - ((da2 /2)* Cos(omega))
    zeta = Atn(((da2 /2)* Sin(omega))/ rx)
    rax = rx / Cos(zeta)
    ksi = Sqr((rax / (db1 /2))^2-1)
    kutrot = omegal - (ksi - Atn(ksi))+ zeta - kut

ElseIf OptionButton9.Value =TrueThen
    omega = Acos((db1 /2)/(da /2))- alfatw
    ksi = Sqr(((da /2)/(db1 /2))^2-1)
    kutrot = omegal - (ksi - Atn(ksi))- omega + kut

EndIf
EndIf

If CheckBox1.Value =TrueOr under1 <0Then
Dim trohoidal As AcadSpline
Dim evolvental As AcadSpline
Dim rottrohoidal AsVariant
Dim rotevolvental AsVariant

Set trohoidal = ThisDrawing.ModelSpace.AddSpline(tockeetroh1, startan,
endtan)
    trohoidal.FitTolerance =0

Set evolvental = ThisDrawing.ModelSpace.AddSpline(tockeeinv1, startan,
endtan)
    evolvental.FitTolerance =0

```

```
If CheckBox4.Value =TrueThen
    trohoidal.Rotate center, kutrot
    evolvental.Rotate center, kutrot
EndIf

    rottrohoidal = trohoidal.ArrayPolar(z,2* PI, center)

    rotevolvental = evolvental.ArrayPolar(z,2* PI, center)

Else
Dim splinel As AcadSpline
Dim rotsplinel AsVariant

Set splinel = ThisDrawing.ModelSpace.AddSpline(tockeel, startan, endtan)
    splinel.FitTolerance =0

If CheckBox4.Value =TrueThen
    splinel.Rotate center, kutrot
EndIf

    rotsplinel = splinel.ArrayPolar(z,2* PI, center)
EndIf

If CheckBox1.Value =TrueOr under2 <0Then
Dim trohoida2 As AcadSpline
Dim evolventa2 As AcadSpline
Dim rottrohoida2 AsVariant
Dim rotevolventa2 AsVariant

Set trohoida2 = ThisDrawing.ModelSpace.AddSpline(tockeetroh2, startan,
endtan)
    trohoida2.FitTolerance =0

Set evolventa2 = ThisDrawing.ModelSpace.AddSpline(tockeeeinv2, startan,
endtan)
    evolventa2.FitTolerance =0

If CheckBox4.Value =TrueThen
    trohoida2.Rotate center, kutrot
    evolventa2.Rotate center, kutrot
EndIf

    rottrohoida2 = trohoida2.ArrayPolar(z,2* PI, center)

    rotevolventa2 = evolventa2.ArrayPolar(z,2* PI, center)

Else
Dim spline2 As AcadSpline
Dim rotspline2 AsVariant

Set spline2 = ThisDrawing.ModelSpace.AddSpline(tockee2, startan, endtan)
    spline2.FitTolerance =0

If CheckBox4.Value =TrueThen
    spline2.Rotate center, kutrot
EndIf

    rotspline2 = spline2.ArrayPolar(z,2* PI, center)
```

```

EndIf
Set luk = ThisDrawing.ModelSpace.AddArc(center, da /2, gama1, gama2)

If CheckBox4.Value =TrueThen
    luk.Rotate center, kutrot
EndIf

    rotluk = luk.ArrayPolar(z,2* PI, center)

'ZoomExtents
'Exit Sub

'ZUPČAN
IK2'

If CheckBox4.Value =TrueThen

Dim A3, A4, under3, under4 AsDouble
Dim xtroh3, ytroh3, xtroh4, ytroh4, xinv3, yinv3, xinv4, yinv4, korakf2,
kut2, kut13, kut14, kut23, kut24 AsDouble
Dim eta13, eta14, eta23, eta24, s3, s4, ksi3, ksi4, xkf3, xkf4, ykf4, xf3,
yf3, xf4, yf4, omikron2, ost2 AsDouble
Dim tockee3()AsDouble
Dim tockeeetroh3()AsDouble
Dim tockeeinv3()AsDouble
Dim tockee4()AsDouble
Dim tockeeetroh4()AsDouble
Dim tockeeinv4()AsDouble
Dim tocke3 AsNew Collection
Dim tocketroh3 AsNew Collection
Dim tockeinv3 AsNew Collection
Dim tocke4 AsNew Collection
Dim tocketroh4 AsNew Collection
Dim tockeinv4 AsNew Collection

    A3 = ro * m / (Cos(alfan3) * Cos(beta)) + (h * m - pom2 * m - ro * m) *
Tan(alfat3)
    A4 = ro * m / (Cos(alfan4) * Cos(beta)) + (h * m - pom2 * m - ro * m) *
Tan(alfat4)

'uvjet podrezanosti:
    under3 = (d2 /2) * (Sin(alfat3))^2 - (h * m - pom2 * m - ro * m * (1-
Sin(alfan3)))
    under4 = (d2 /2) * (Sin(alfat4))^2 - (h * m - pom2 * m - ro * m * (1-
Sin(alfan4)))

    fil = PI /2

'kut između simetrala susjednih zubi:
    kut2 =2* PI / z2

```

```

'debljina zuba na diobenom promjeru:
  s3 =(m / Cos(beta))* (PI /2+2* pom2 * Tan(alfan3))
  s4 =(m / Cos(beta))* (PI /2+2* pom2 * Tan(alfan4))

'kut između simetrale zuba i točke sjecišta boka zuba i diobene kružnice:
  eta13 =(s3 /(d2 /2))/2
  eta14 =(s4 /(d2 /2))/2

'parametar evolvente za točku na diobenoj kružnici:
  ksi3 = Sqr(((d2 /2)/(db3 /2))^2-1)
  ksi4 = Sqr(((d2 /2)/(db4 /2))^2-1)

'kut između točke na sjecištu boka zuba i diobene kružnice i osi x; formula
evolvente:
  eta23 = ksi3 - Atn(ksi3)
  eta24 = ksi4 - Atn(ksi4)

'kut prve točke podnožne kružnice/točke na sredini uzubine:
  kut13 = eta13 + eta23 -(kut2 /2)
  kut14 = eta14 + eta24 -(kut2 /2)

'kut zadnje točke podnožne kružnice/prve točke trohoide; formula trohoide:
  kut23 = theta(fil, m, ro, alfat3, d2, A3, pom2, h, beta)+
epsilon(fil, d2, m, ro, pom2, h, beta)- alfat3
  kut24 = theta(fil, m, ro, alfat4, d2, A4, pom2, h, beta)+
epsilon(fil, d2, m, ro, pom2, h, beta)- alfat4

'kut između simetrale zuba i osi x:
  omega3 = eta13 + eta23
  omega4 = eta14 + eta24

  omikron2 = omega3 + omega4 - kut23 - kut24

If omikron2 < kut2 Then
  ost2 = kut2 - omikron2
  korakf2 =(ost2 /2)/4
'korakf2 = (kut23 - kut13) / 4
For j =(kut23 -(ost2 /2))To kut23 Step korakf2
'For j = kut13 To kut23 Step korakf2
  xkf3 =(df2 /2)* Cos(j - omega3)
  yf3 =(df2 /2)* Sin(j - omega3)
'preslikavanje preko osi koja prolazi polovicom osnog razmaka:
  xf3 = xkf3 -(2*(xkf3 -(osraz /2)))

If(CheckBox1.Value =False)And(under3 >=0)Then
  tocke3.Add (xf3)
  tocke3.Add (yf3)
  tocke3.Add (0)

Else
  tocketroh3.Add (xf3)
  tocketroh3.Add (yf3)
  tocketroh3.Add (0)

EndIf
Next

'korakf2 = (kut24 - kut14) / 4
For j =(kut24 -(ost2 /2))To kut24 Step korakf2
'For j = kut14 To kut24 Step korakf2

```

```

        xkf4 = (df2 / 2) * Cos(j - omega4)
        ykf4 = (df2 / 2) * Sin(j - omega4)

        xf4 = xkf4 - (2 * (xkf4 - (osraz / 2)))
'preslikavanje preko osi x:
        yf4 = -ykf4
If (CheckBox1.Value = False) And (under4 >= 0) Then
        tocke4.Add (xf4)
        tocke4.Add (yf4)
        tocke4.Add (0)
Else
        tocketroh4.Add (xf4)
        tocketroh4.Add (yf4)
        tocketroh4.Add (0)
EndIf
Next

ElseIf omikron2 = kut2 Then
ResumeNext

ElseIf omikron2 > kut2 Then
        MsgBox "Gonjeni zupčanik je nemoguće izraditi odvalnim
glodanjem! Na sredini uzubine nalazi se sjecište podnožnih krivulja
(trohoida) susjednih zubi!"
ExitSub
EndIf

'nepodrezani bok3:
If under3 >= 0 Then
        B1 = B(alfan3, m, h, ro, pom2)
        rsoi = Sqr(((d2 / 2) - B1)^2 + (B1 * Cos(beta) / Tan(alfan3))^2)

If CheckBox1.Value = True Then
        korak1 = (fi1 - alfan3) / (brtr - 1)
Else
        korak1 = (fi1 - (alfan3 + 0.025)) / (brtr - 1)
EndIf

        fi = fi1
For i = 1 To (brtr - 1)
        rtroh = Sqr((d2 / 2 - B(fi, m, h, ro, pom2))^2 + ((B(fi, m, h, ro, pom2) *
Cos(beta)) / Tan(fi))^2)
        etatroh = theta(fi, m, ro, alfat3, d2, A3, pom2, h, beta) + epsilon(fi,
d2, m, ro, pom2, h, beta) - alfat3
        xtroh = rtroh * Cos(etatroh - omega3)
        ytroh3 = rtroh * Sin(etatroh - omega3)
        xtroh3 = xtroh - (2 * (xtroh - (osraz / 2)))

If CheckBox1.Value = True Then
        tocketroh3.Add (xtroh3)
        tocketroh3.Add (ytroh3)
        tocketroh3.Add (0)
Else
        tocke3.Add (xtroh3)
        tocke3.Add (ytroh3)
        tocke3.Add (0)
EndIf

        fi = fi - korak1

```

```

Next

If CheckBox1.Value =TrueThen
    fi = alfan3
Else
    fi = alfan3 +0.025
EndIf
    rtroh = Sqr((d2 /2- B(fi, m, h, ro, pom2))^2+((B(fi, m, h, ro, pom2)*
Cos(beta))/ Tan(fi))^2)
    etatroh = theta(fi, m, ro, alfat3, d2, A3, pom2, h, beta)+ epsilon(fi,
d2, m, ro, pom2, h, beta)- alfat3
    xtroh = rtroh * Cos(etatroh - omega3)
    ytroh3 = rtroh * Sin(etatroh - omega3)
    xtroh3 = xtroh - (2*(xtroh - (osraz /2)))

If CheckBox1.Value =TrueThen
    tocketroh3.Add (xtroh3)
    tocketroh3.Add (ytroh3)
    tocketroh3.Add (0)
Else
    tocke3.Add (xtroh3)
    tocke3.Add (ytroh3)
    tocke3.Add (0)
EndIf

ksii1 = Sqr((rsoi / (db3 /2))^2-1)'parametar prve točke
evolvente
ksii2 = Sqr(((da2 /2)/(db3 /2))^2-1)'parametar zadnje točke
evolvente

korak =(ksii2 - ksii1)/(brinv -1)

ksii = ksii1
For i =1To (brinv -1)
    rinov = (db3 /2) * Sqr(1+(ksii)^2)
    etainv = ksii - Atn(ksii)
    xinov = rinov * Cos(etainv - omega3)
    yinov3 = rinov * Sin(etainv - omega3)
    xinov3 = xinov - (2*(xinov - (osraz /2)))

If CheckBox1.Value =TrueThen
    tockeinv3.Add (xinov3)
    tockeinv3.Add (yinov3)
    tockeinv3.Add (0)
Else
    tocke3.Add (xinov3)
    tocke3.Add (yinov3)
    tocke3.Add (0)
EndIf

ksii = ksii + korak

Next

rinov = (db3 /2) * Sqr(1+(ksii2)^2)
etainv = ksii2 - Atn(ksii2)
xinov = rinov * Cos(etainv - omega3)
yinov3 = rinov * Sin(etainv - omega3)
xinov3 = xinov - (2*(xinov - (osraz /2)))

```

```

If CheckBox1.Value =TrueThen
    tockeinv3.Add (xinv3)
    tockeinv3.Add (yinv3)
    tockeinv3.Add (0)
Else
    tocke3.Add (xinv3)
    tocke3.Add (yinv3)
    tocke3.Add (0)
EndIf

'podrezani bok3:
Else
    fi0 = bisekcija(m, ro, alfat3, d2, A3, pom2, db3, h, beta)

If CheckBox3.Value =TrueThen
    MsgBox ("Sjecište evolvente i trohoide COAST boka gonjenog
zupčanika određeno je pomoću vrijednosti nultočke: "+CStr(f(fi0, m, ro,
alfat3, d2, A3, pom2, db3, h, beta)))
Else
    MsgBox ("Sjecište evolvente i trohoide boka gonjenog
zupčanika određeno je pomoću vrijednosti nultočke: "+CStr(f(fi0, m, ro,
alfat3, d2, A3, pom2, db3, h, beta)))
EndIf

B1 = B(fi0, m, h, ro, pom2)
rsoi = Sqr(((d2 /2)- B1)^2+(B1 * Cos(beta)/ Tan(fi0))^2)

    korak1 =(fi1 - fi0)/(brtr -1)

    fi = fi1
For i =1To(brtr -1)
    rtroh = Sqr((d2 /2- B(fi, m, h, ro, pom2))^2+((B(fi, m, h, ro, pom2)*
Cos(beta))/ Tan(fi))^2)
    etatroh = theta(fi, m, ro, alfat3, d2, A3, pom2, h, beta)+ epsilon(fi,
d2, m, ro, pom2, h, beta)- alfat3
    xtroh = rtroh * Cos(etatroh - omega3)
    ytroh3 = rtroh * Sin(etatroh - omega3)
    xtroh3 = xtroh - (2*(xtroh - (osraz /2)))

    tocketroh3.Add (xtroh3)
    tocketroh3.Add (ytroh3)
    tocketroh3.Add (0)
    fi = fi - korak1
Next

    fi = fi0
    rtroh = Sqr((d2 /2- B(fi, m, h, ro, pom2))^2+((B(fi, m, h, ro, pom2)*
Cos(beta))/ Tan(fi))^2)
    etatroh = theta(fi, m, ro, alfat3, d2, A3, pom2, h, beta)+ epsilon(fi,
d2, m, ro, pom2, h, beta)- alfat3
    xtroh = rtroh * Cos(etatroh - omega3)
    ytroh3 = rtroh * Sin(etatroh - omega3)
    xtroh3 = xtroh - (2*(xtroh - (osraz /2)))

    tocketroh3.Add (xtroh3)
    tocketroh3.Add (ytroh3)
    tocketroh3.Add (0)

```

```

ksii1 = Sqr((rsoi / (db3 / 2))^2 - 1) 'parametar prve točke
evolvente
ksii2 = Sqr(((da2 / 2) / (db3 / 2))^2 - 1) 'parametar zadnje točke
evolvente

korak = (ksii2 - ksii1) / (brinv - 1)

ksii = ksii1
For i = 1 To (brinv - 1)
    rinv = (db3 / 2) * Sqr(1 + (ksii)^2)
    etainv = ksii - Atn(ksii)
    xinv = rinv * Cos(etainv - omega3)
    yinv3 = rinv * Sin(etainv - omega3)
    xinv3 = xinv - (2 * (xinv - (osraz / 2)))

    tockeinv3.Add (xinv3)
    tockeinv3.Add (yinv3)
    tockeinv3.Add (0)
    ksii = ksii + korak
Next

rinv = (db3 / 2) * Sqr(1 + (ksii2)^2)
etainv = ksii2 - Atn(ksii2)
xinv = rinv * Cos(etainv - omega3)
yinv3 = rinv * Sin(etainv - omega3)
xinv3 = xinv - (2 * (xinv - (osraz / 2)))

tockeinv3.Add (xinv3)
tockeinv3.Add (yinv3)
tockeinv3.Add (0)
EndIf

'nepodrezani bok4:
If under4 >= 0 Then
    B1 = B(alfan4, m, h, ro, pom2)
    rsoi = Sqr(((d2 / 2) - B1)^2 + (B1 * Cos(beta) / Tan(alfan4))^2)

If CheckBox1.Value = True Then
    korak1 = (fi1 - alfan4) / (brtr - 1)
Else
    korak1 = (fi1 - (alfan4 + 0.025)) / (brtr - 1)
EndIf

fi = fi1
For i = 1 To (brtr - 1)
    rtroh = Sqr((d2 / 2 - B(fi, m, h, ro, pom2))^2 + ((B(fi, m, h, ro, pom2) *
Cos(beta)) / Tan(fi))^2)
    etatroh = theta(fi, m, ro, alfat4, d2, A4, pom2, h, beta) + epsilon(fi,
d2, m, ro, pom2, h, beta) - alfat4
    xtroh = rtroh * Cos(etatroh - omega4)
    ytroh = rtroh * Sin(etatroh - omega4)

    xtroh4 = xtroh - (2 * (xtroh - (osraz / 2)))
    ytroh4 = -ytroh

If CheckBox1.Value = True Then
    tocketroh4.Add (xtroh4)
    tocketroh4.Add (ytroh4)

```



```

        tocketroh4.Add (0)
Else
        tocke4.Add (xtroh4)
        tocke4.Add (ytroh4)
        tocke4.Add (0)
EndIf
        fi = fi - korak1
Next

If CheckBox1.Value =TrueThen
        fi = alfan4
Else
        fi = alfan4 +0.025
EndIf
        rtroh = Sqr((d2 /2- B(fi, m, h, ro, pom2))^2+((B(fi, m, h, ro, pom2)*
Cos(beta))/ Tan(fi))^2)
        etatroh = theta(fi, m, ro, alfat4, d2, A4, pom2, h, beta)+ epsilon(fi,
d2, m, ro, pom2, h, beta)- alfat4
        xtroh = rtroh * Cos(etatroh - omega4)
        ytroh = rtroh * Sin(etatroh - omega4)

        xtroh4 = xtroh -(2*(xtroh -(osraz /2)))
        ytroh4 =-ytroh

If CheckBox1.Value =TrueThen
        tocketroh4.Add (xtroh4)
        tocketroh4.Add (ytroh4)
        tocketroh4.Add (0)
Else
        tocke4.Add (xtroh4)
        tocke4.Add (ytroh4)
        tocke4.Add (0)
EndIf

        ksii1 = Sqr((rsoi /(db4 /2))^2-1)'parametar prve točke
evolvente
        ksii2 = Sqr(((da2 /2)/(db4 /2))^2-1)'parametar zadnje točke
evolvente

        korak =(ksii2 - ksii1)/(brinv -1)

        ksii = ksii1
For i =1To(brinv -1)
        rinv =(db4 /2)* Sqr(1+(ksii)^2)
        etainv = ksii - Atn(ksii)
        xinv = rinv * Cos(etainv - omega4)
        yinv = rinv * Sin(etainv - omega4)

        xinv4 = xinv -(2*(xinv -(osraz /2)))
        yinv4 =-yinv

If CheckBox1.Value =TrueThen
        tockeinv4.Add (xinv4)
        tockeinv4.Add (yinv4)
        tockeinv4.Add (0)
Else
        tocke4.Add (xinv4)
        tocke4.Add (yinv4)

```

```

        tocke4.Add (0)
EndIf
        ksii = ksii + korak
Next

        rinv =(db4 /2)* Sqr(1+(ksii2)^2)
        etainv = ksii2 - Atn(ksii2)
        xinv = rinv * Cos(etainv - omega4)
        yinv = rinv * Sin(etainv - omega4)

        xinv4 = xinv -(2*(xinv -(osraz /2)))
        yinv4 =-yinv

If CheckBox1.Value =TrueThen
        tockeinv4.Add (xinv4)
        tockeinv4.Add (yinv4)
        tockeinv4.Add (0)
Else
        tocke4.Add (xinv4)
        tocke4.Add (yinv4)
        tocke4.Add (0)
EndIf

'podrezani bok2:
Else
        fi0 = bisekcija(m, ro, alfat4, d2, A4, pom2, db4, h, beta)

If CheckBox3.Value =TrueThen
        MsgBox ("Sjecište evolvente i trohoide DRIVE boka gonjenog
zupčanika određeno je pomoću vrijednosti nultočke: "+CStr(f(fi0, m, ro,
alfat4, d2, A4, pom2, db4, h, beta)))
EndIf

        B1 = B(fi0, m, h, ro, pom2)
        rsoi = Sqr(((d2 /2)- B1)^2+(B1 * Cos(beta)/ Tan(fi0))^2)

        korak1 =(fi1 - fi0)/(brtr -1)

        fi = fi1
For i =1To(brtr -1)
        rtroh = Sqr((d2 /2- B(fi, m, h, ro, pom2))^2+((B(fi, m, h, ro, pom2)*
Cos(beta))/ Tan(fi))^2)
        etatroh = theta(fi, m, ro, alfat4, d2, A4, pom2, h, beta)+ epsilon(fi,
d2, m, ro, pom2, h, beta)- alfat4
        xtroh = rtroh * Cos(etatroh - omega4)
        ytroh = rtroh * Sin(etatroh - omega4)

        xtroh4 = xtroh -(2*(xtroh -(osraz /2)))
        ytroh4 =-ytroh

        tocketroh4.Add (xtroh4)
        tocketroh4.Add (ytroh4)
        tocketroh4.Add (0)
        fi = fi - korak1
Next

        fi = fi0

```

```

    rtroh = Sqr((d2 /2- B(fi, m, h, ro, pom2))^2+((B(fi, m, h, ro, pom2)*
Cos(beta))/ Tan(fi))^2)
    etatroh = theta(fi, m, ro, alfat4, d2, A4, pom2, h, beta)+ epsilon(fi,
d2, m, ro, pom2, h, beta)- alfat4
    xtroh = rtroh * Cos(etatroh - omega4)
    ytroh = rtroh * Sin(etatroh - omega4)

    xtroh4 = xtroh -(2*(xtroh -(osraz /2)))
    ytroh4 =-ytroh

    tocketroh4.Add (xtroh4)
    tocketroh4.Add (ytroh4)
    tocketroh4.Add (0)

    ksii1 = Sqr((rsoi /(db4 /2))^2-1)'parametar prve točke
evolvente
    ksii2 = Sqr(((da2 /2)/(db4 /2))^2-1)'parametar zadnje točke
evolvente

    korak =(ksii2 - ksii1)/(brinv -1)

    ksii = ksii1
For i =1To(brinv -1)
    rinv =(db4 /2)* Sqr(1+(ksii)^2)
    etainv = ksii - Atn(ksii)
    xinv = rinv * Cos(etainv - omega4)
    yinv = rinv * Sin(etainv - omega4)

    xinv4 = xinv -(2*(xinv -(osraz /2)))
    yinv4 =-yinv

    tockeinv4.Add (xinv4)
    tockeinv4.Add (yinv4)
    tockeinv4.Add (0)

    ksii = ksii + korak

Next

    rinv =(db4 /2)* Sqr(1+(ksii2)^2)
    etainv = ksii2 - Atn(ksii2)
    xinv = rinv * Cos(etainv - omega4)
    yinv = rinv * Sin(etainv - omega4)

    xinv4 = xinv -(2*(xinv -(osraz /2)))
    yinv4 =-yinv

    tockeinv4.Add (xinv4)
    tockeinv4.Add (yinv4)
    tockeinv4.Add (0)

EndIf

'crtanje krivulja:
If CheckBox1.Value =TrueOr under3 <0Then
    brtotroh1 = tocketroh3.Count
    brtoinv1 = tockeinv3.Count
ReDim tockeetroh3(0To brtotroh1 -1)
ReDim tockeinv3(0To brtoinv1 -1)

```

```

For bro =0To(brtotroh1 -1)
    tockeetroh3(bro)= tocketroh3.Item(bro +1)
Next

For bro =0To(brtoinv1 -1)
    tockeeinv3(bro)= tockeinv3.Item(bro +1)
Next
Else
    brto = tocke3.Count
ReDim tockee3(0To brto -1)
For bro =0To(brto -1)
    tockee3(bro)= tocke3.Item(bro +1)
Next
EndIf

If CheckBox1.Value =TrueOr under4 <0Then
    brtotroh2 = tocketroh4.Count
    brtoinv2 = tockeinv4.Count
ReDim tockeetroh4(0To brtotroh2 -1)
ReDim tockeeinv4(0To brtoinv2 -1)

For bro =0To(brtotroh2 -1)
    tockeetroh4(bro)= tocketroh4.Item(bro +1)
Next

For bro =0To(brtoinv2 -1)
    tockeeinv4(bro)= tockeinv4.Item(bro +1)
Next
Else
    brto = tocke4.Count
ReDim tockee4(0To brto -1)
For bro =0To(brto -1)
    tockee4(bro)= tocke4.Item(bro +1)
Next
EndIf

Dim sa2, gama3, gama4, delta3, delta4 AsDouble
Dim startan2(0To2)AsDouble
Dim endtan2(0To2)AsDouble
Dim center2(0To2)AsDouble
Dim luk2 As AcadArc
Dim rotluk2 AsVariant

    startan2(0)=0: startan2(1)=0: startan2(2)=0
    endtan2(0)=0: endtan2(1)=0: endtan2(2)=0

    center2(0)= osraz: center(1)=0: center(2)=0

'kut točke na sjecištu evolvente i tjemene kružnice:
delta3 = Sqr(((da2 /2)/(db3 /2))^2-1)
gama3 =-((delta3 - Atn(delta3))- omega3)
delta4 = Sqr(((da2 /2)/(db4 /2))^2-1)
gama4 =(delta4 - Atn(delta4))- omega4

If gama3 < gama4 Then
    MsgBox ("Gonjeni zupčanik je nemoguće izraditi jer se evolvente
sijeku prije tjemene kružnice!")

```

```

ExitSub
EndIf

'debljina zuba na tjemenoj kružnici (duljina luka):
    sa2 =(da2 /2)* (Abs (gama3)+ Abs (gama4))

If sa2 <=(0.2* m) Then
    MsgBox "Tjemena debljina zuba gonjenog zupčanika premala; zub je
    ZAŠILJEN! Moguća rješenja: smanjenje pomaka profila, smanjenje faktora
    tjemene visine zuba, povećanje broja zubi."
ExitSub
EndIf

If sa2 <=(0.4* m) Then
    MsgBox "UPOZORENJE: Tjemena debljina zuba gonjenog zupčanika je
    premala za kaljeni zupčanik! Moguća rješenja: odabir materijala zupčanika
    koji se ne kali, smanjenje pomaka profila, smanjenje faktora tjemene visine
    zuba, povećanje broja zubi."
EndIf

'određivanje kutova rotacije za crtanje određenih točki zahvata zupčanika:

If OptionButton10.Value =TrueThen
    ksi = Sqr(((dw4 /2)/(db4 /2))^2-1)
    kutrot = omega4 -(ksi - Atn(ksi))

ElseIf OptionButton8.Value =TrueThen
    omega = Acos((db4 /2)/(da2 /2))- alfatw
    ksi = Sqr(((da2 /2)/(db4 /2))^2-1)
    kutrot = omega4 -(ksi - Atn(ksi))- omega

ElseIf OptionButton12.Value =TrueThen
    omega = Acos((db1 /2)/(da /2))- alfatw
    rx = osraz -((da /2)* Cos(omega))
    zeta = Atn(((da /2)* Sin(omega))/ rx)
    rax = rx / Cos(zeta)
    ksi = Sqr((rax / (db4 /2))^2-1)
    kutrot = omega4 -(ksi - Atn(ksi))+ zeta

ElseIf OptionButton11.Value =TrueThen
    omega = Acos((db4 /2)/(da2 /2))- alfatw
    ksi = Sqr(((da2 /2)/(db4 /2))^2-1)
    kutrot = omega4 -(ksi - Atn(ksi))- omega + kut2

ElseIf OptionButton9.Value =TrueThen
    omega = Acos((db1 /2)/(da /2))- alfatw
    rx = osraz -((da /2)* Cos(omega))
    zeta = Atn(((da /2)* Sin(omega))/ rx)
    rax = rx / Cos(zeta)
    ksi = Sqr((rax / (db4 /2))^2-1)
    kutrot = omega4 -(ksi - Atn(ksi))+ zeta - kut2

EndIf

If CheckBox1.Value =TrueOr under3 <0Then
Dim trohoida3 As AcadSpline

```

```
Dim evolventa3 As AcadSpline
Dim rotrhoida3 AsVariant
Dim rotevolventa3 AsVariant

Set trohoida3 = ThisDrawing.ModelSpace.AddSpline(tockeetroh3, startan2,
endtan2)
    trohoida3.FitTolerance =0

Set evolventa3 = ThisDrawing.ModelSpace.AddSpline(tockeeeinv3, startan2,
endtan2)
    evolventa3.FitTolerance =0

    trohoida3.Rotate center2, kutrot
    evolventa3.Rotate center2, kutrot

    rotrhoida3 = trohoida3.ArrayPolar(z2,2* PI, center2)
    rotevolventa3 = evolventa3.ArrayPolar(z2,2* PI, center2)

Else
Dim spline3 As AcadSpline
Dim rotspline3 AsVariant

Set spline3 = ThisDrawing.ModelSpace.AddSpline(tockee3, startan2, endtan2)
    spline3.FitTolerance =0

    spline3.Rotate center2, kutrot

    rotspline3 = spline3.ArrayPolar(z2,2* PI, center2)
EndIf

If CheckBox1.Value =TrueOr under4 <0Then
Dim trohoida4 As AcadSpline
Dim evolventa4 As AcadSpline
Dim rotrhoida4 AsVariant
Dim rotevolventa4 AsVariant

Set trohoida4 = ThisDrawing.ModelSpace.AddSpline(tockeetroh4, startan2,
endtan2)
    trohoida4.FitTolerance =0

Set evolventa4 = ThisDrawing.ModelSpace.AddSpline(tockeeeinv4, startan2,
endtan2)
    evolventa4.FitTolerance =0

    trohoida4.Rotate center2, kutrot
    evolventa4.Rotate center2, kutrot

    rotrhoida4 = trohoida4.ArrayPolar(z2,2* PI, center2)
    rotevolventa4 = evolventa4.ArrayPolar(z2,2* PI, center2)

Else
Dim spline4 As AcadSpline
Dim rotspline4 AsVariant

Set spline4 = ThisDrawing.ModelSpace.AddSpline(tockee4, startan2, endtan2)
    spline4.FitTolerance =0
```

```
spline4.Rotate center2, kutrot

rotspline4 = spline4.ArrayPolar(z2,2* PI, center2)
EndIf

Set luk2 = ThisDrawing.ModelSpace.AddArc(center2, da2 /2, (gama4 +
PI), (gama3 + PI))

luk2.Rotate center2, kutrot

rotluk2 = luk2.ArrayPolar(z2,2* PI, center2)

Dim tipcrte As AcadLineType
Dim found As Boolean

If CheckBox5.Value =TrueThen
Dim xzc(0To2)AsDouble
Dim yzc(0To2)AsDouble
Dim zahvatcrta As AcadLine

xzc(0)=((dbl /2)* Cos(alfatw)): xzc(1)=((dbl /2)*
Sin(alfatw)): xzc(2)=0
yzc(0)= osraz -((db4 /2)* Cos(alfatw)): yzc(1)=-((db4 /2)*
Sin(alfatw)): yzc(2)=0

Set zahvatcrta = ThisDrawing.ModelSpace.AddLine(xzc, yzc)
zahvatcrta.color = acGreen
EndIf

If CheckBox6.Value =TrueThen

Dim basecirc1 As AcadCircle
Dim basecirc2 As AcadCircle

found =False

ForEach tipcrte In ThisDrawing.Linetypes
If StrComp(tipcrte.Name,"DASHED",1)=0Then
found =True
ExitFor
EndIf
Next
IfNot(found)Then ThisDrawing.Linetypes.Load "DASHED","acad.lin"

Set basecirc1 = ThisDrawing.ModelSpace.AddCircle(center, dbl /2)
basecirc1.Linetype ="DASHED"

Set basecirc2 = ThisDrawing.ModelSpace.AddCircle(center2, db4 /2)
basecirc2.Linetype ="DASHED"

EndIf

If CheckBox7.Value =TrueThen

Dim kinemcirc1 As AcadCircle
Dim kinemcirc2 As AcadCircle
```

```

        found =False

ForEach tipcrte In ThisDrawing.Linetypes
If StrComp(tipcrte.Name,"DASHDOT",1)=0Then
        found =True
ExitFor
EndIf
Next
IfNot(found)Then ThisDrawing.Linetypes.Load "DASHDOT","acad.lin"

Set kinemcirc1 = ThisDrawing.ModelSpace.AddCircle(center, dw1 /2)
        kinemcirc1.Linetype ="DASHDOT"

Set kinemcirc2 = ThisDrawing.ModelSpace.AddCircle(center2, dw4 /2)
        kinemcirc2.Linetype ="DASHDOT"

EndIf

        ZoomExtents
ExitSub

Else
        ZoomExtents
ExitSub
EndIf

HandlerError:
        MsgBox "U polja moraju biti upisane brojčane vrijednosti!"
ExitSub

EndSub
PrivateFunction B(ByVal x AsDouble,ByVal m AsDouble,ByVal h AsDouble,ByVal
ro AsDouble,ByVal pom AsDouble)AsDouble
        B = h * m - pom * m - ro * m + ro * m * Sin(x)
EndFunction

PrivateFunction theta(ByVal x AsDouble,ByVal m AsDouble,ByVal ro
AsDouble,ByVal alfat AsDouble,ByVal d AsDouble,ByVal A AsDouble,ByVal pom
AsDouble,ByVal h AsDouble,ByVal beta AsDouble)AsDouble
        theta = Tan(alfat)+(2/ d)*(ro * m *(Cos(x)/ Cos(beta))- A -(B(x, m,
h, ro, pom)* Cos(beta))/ Tan(x))
EndFunction

PrivateFunction epsilon(ByVal x AsDouble,ByVal d AsDouble,ByVal m
AsDouble,ByVal ro AsDouble,ByVal pom AsDouble,ByVal h AsDouble,ByVal beta
AsDouble)AsDouble
        epsilon = Atn((B(x, m, h, ro, pom)* Cos(beta))/((d /2- B(x, m, h,
ro, pom))* Tan(x)))
EndFunction

PrivateFunction ksi(ByVal x AsDouble,ByVal d AsDouble,ByVal db
AsDouble,ByVal m AsDouble,ByVal ro AsDouble,ByVal pom AsDouble,ByVal h
AsDouble,ByVal beta AsDouble)AsDouble
        ksi = Sqr(((d /2- B(x, m, h, ro, pom))/(db /2))^2+((B(x, m, h, ro,
pom)* Cos(beta))/((db /2)* Tan(x)))^2-1)
EndFunction

```



```

PrivateFunction f(ByVal x AsDouble,ByVal m AsDouble,ByVal ro AsDouble,ByVal
alfat AsDouble,ByVal d AsDouble,ByVal A AsDouble,ByVal pom AsDouble,ByVal
db AsDouble,ByVal h AsDouble,ByVal beta AsDouble)AsDouble
    f = theta(x, m, ro, alfat, d, A, pom, h, beta)+ epsilon(x, d, m,
ro, pom, h, beta)- ksi(x, d, db, m, ro, pom, h, beta)+ Atn(ksi(x, d, db, m,
ro, pom, h, beta))- alfat
EndFunction

PrivateFunction Acos(ByVal x AsDouble)AsDouble
    Acos = Atn(-x / Sqr(-x * x +1))+2* Atn(1)
EndFunction

PrivateFunction bisekcija(ByVal m AsDouble,ByVal ro AsDouble,ByVal alfat
AsDouble,ByVal d AsDouble,ByVal A AsDouble,ByVal pom AsDouble,ByVal db
AsDouble,ByVal h AsDouble,ByVal beta AsDouble)AsDouble
Dim x0, x1, x2, x3 AsDouble
Dim k As Boolean
Dim br As Double
    x1 =0.1
    x2 =0.55
OnError ResumeNext
If IsError(f(x2, m, ro, alfat, d, A, pom, db, h, beta))=True Then
OnError ResumeNext
DoUntil IsError(f(x2, m, ro, alfat, d, A, pom, db, h, beta))=False
    x2 = x2 -0.000001

Loop
EndIf

    br =0
    k =True
DoWhile k
    br = br +1
    x0 =(x1 + x2)/2
If Abs(x2 - x1)=0 Or Abs(f(x0, m, ro, alfat, d, A, pom, db, h, beta))=0
Then
ExitDo
EndIf
If f(x1, m, ro, alfat, d, A, pom, db, h, beta)* f(x0, m, ro, alfat, d, A,
pom, db, h, beta)<0 Then
    x2 = x0

Else
    x1 = x0

EndIf
If br >1000 Then
    x3 = x0

ExitDo
EndIf
Loop

    bisekcija = x0
EndFunction

PrivateSub NOVI_UNOS_Click()
    TextBox1.Value =""
    TextBox2.Value =""
    TextBox3.Value =""
    TextBox4.Value =""
    TextBox5.Value =""
    TextBox6.Value =""

```

```
    TextBox7.Value = ""
    TextBox8.Value = ""
    TextBox9.Value = ""
    TextBox10.Value = ""
    TextBox11.Value = ""
    TextBox12.Value = ""
    TextBox13.Value = ""
    TextBox14.Value = ""
    TextBox15.Value = ""
    TextBox6.Enabled = False
    TextBox7.Enabled = False
    TextBox8.Enabled = False
    TextBox9.Enabled = False
    TextBox11.Enabled = False
    TextBox12.Enabled = False
    TextBox13.Enabled = False
    TextBox14.Enabled = False
    TextBox15.Enabled = False
    TextBox6.Visible = False
    TextBox7.Visible = False
    TextBox8.Visible = False
    TextBox9.Visible = False
    TextBox11.Visible = False
    TextBox12.Visible = False
    TextBox13.Visible = False
    TextBox14.Visible = False
    TextBox15.Visible = False
    CheckBox1.Value = False
    CheckBox2.Value = True
    OptionButton1.Value = True
EndSub

PrivateSub OptionButton1_Click()
    TextBox6.Enabled = False
    TextBox7.Enabled = False
    TextBox8.Enabled = False
    TextBox9.Enabled = False
    TextBox6.Visible = False
    TextBox7.Visible = False
    TextBox8.Visible = False
    TextBox9.Visible = False
EndSub

PrivateSub OptionButton2_Click()
    TextBox6.Enabled = False
    TextBox7.Enabled = False
    TextBox8.Enabled = False
    TextBox9.Enabled = False
    TextBox6.Visible = False
    TextBox7.Visible = False
    TextBox8.Visible = False
    TextBox9.Visible = False
EndSub

PrivateSub OptionButton3_Click()
    TextBox6.Enabled = False
    TextBox7.Enabled = False
    TextBox8.Enabled = False
```

```
        TextBox9.Enabled =False
        TextBox6.Visible =False
        TextBox7.Visible =False
        TextBox8.Visible =False
        TextBox9.Visible =False
EndSub

PrivateSub OptionButton4_Click()
    TextBox6.Enabled =False
    TextBox7.Enabled =False
    TextBox8.Enabled =False
    TextBox9.Enabled =False
    TextBox6.Visible =False
    TextBox7.Visible =False
    TextBox8.Visible =False
    TextBox9.Visible =False
EndSub

PrivateSub OptionButton6_Click()
    TextBox6.Enabled =False
    TextBox7.Enabled =False
    TextBox8.Enabled =False
    TextBox9.Enabled =False
    TextBox6.Visible =False
    TextBox7.Visible =False
    TextBox8.Visible =False
    TextBox9.Visible =False
EndSub

PrivateSub OptionButton7_Click()
    TextBox6.Enabled =False
    TextBox7.Enabled =False
    TextBox8.Enabled =False
    TextBox9.Enabled =False
    TextBox6.Visible =False
    TextBox7.Visible =False
    TextBox8.Visible =False
    TextBox9.Visible =False
EndSub

PrivateSub OptionButton5_Click()
    TextBox6.Visible =True
    TextBox7.Visible =True
    TextBox8.Visible =True
    TextBox9.Visible =True
    TextBox6.Enabled =True
    TextBox7.Enabled =True
    TextBox8.Enabled =True
    TextBox9.Enabled =True
EndSub

PrivateSub CheckBox2_Click()
    CheckBox2.Value =True
    CheckBox3.Value =False
    OptionButton1.Enabled =True
    OptionButton1.Value =True
    OptionButton2.Enabled =True
    OptionButton3.Enabled =True
```

```
OptionButton4.Enabled =True
OptionButton5.Enabled =True
OptionButton6.Enabled =True
OptionButton7.Enabled =True
TextBox11.Enabled =False
TextBox12.Enabled =False
TextBox13.Enabled =False
TextBox14.Enabled =False
TextBox15.Enabled =False
TextBox11.Visible =False
TextBox12.Visible =False
TextBox13.Visible =False
TextBox14.Visible =False
TextBox15.Visible =False
EndSub

PrivateSub CheckBox3_Click()
    CheckBox3.Value =True
    CheckBox2.Value =False
    OptionButton1.Enabled =False
    OptionButton2.Enabled =False
    OptionButton3.Enabled =False
    OptionButton4.Enabled =False
    OptionButton5.Enabled =False
    OptionButton6.Enabled =False
    OptionButton7.Enabled =False
    TextBox11.Visible =True
    TextBox12.Visible =True
    TextBox13.Visible =True
    TextBox14.Visible =True
    TextBox15.Visible =True
    TextBox11.Enabled =True
    TextBox12.Enabled =True
    TextBox13.Enabled =True
    TextBox14.Enabled =True
    TextBox15.Enabled =True
EndSub

PrivateSub CheckBox4_Click()
    If CheckBox4.Value =FalseThen

        TextBox16.Visible =False
        TextBox17.Visible =False
        TextBox16.Enabled =False
        TextBox17.Enabled =False
        OptionButton8.Enabled =False
        OptionButton9.Enabled =False
        OptionButton10.Enabled =False
        OptionButton11.Enabled =False
        OptionButton12.Enabled =False
        CheckBox5.Enabled =False
        CheckBox6.Enabled =False
        CheckBox7.Enabled =False

    Else
        TextBox16.Visible =True
        TextBox17.Visible =True
        TextBox16.Enabled =True
```

```
TextBox17.Enabled =True  
OptionButton8.Enabled =True  
OptionButton9.Enabled =True  
OptionButton10.Enabled =True  
OptionButton11.Enabled =True  
OptionButton12.Enabled =True  
OptionButton10.Value =True  
CheckBox5.Enabled =True  
CheckBox6.Enabled =True  
CheckBox7.Enabled =True  
EndIf  
EndSub
```