



UNIVERZITET U NOVOM SADU
FAKULTET TEHNIČKIH NAUKA



PRIMENA SAVREMENIH METAHEURISTIČKIH ALGORITAMA U OPTIMIZACIJI TEHNOLOŠKIH PROCESA OBRADE PROIZVODA

DOKTORSKA DISERTACIJA

Mentor:
dr Mijodrag Milošević, vanr. prof.

Kandidat:
Mića Đurđev

Novi Sad, 2021. godine

КЉУЧНА ДОКУМЕНТАЦИЈСКА ИНФОРМАЦИЈА¹

Врста рада:	Докторска дисертација
Име и презиме аутора:	Мића Ђурђевић
Ментор (титула, име, презиме, звање, институција):	др Мијодраг Милошевић, ванредни професор, Факултет техничких наука, Универзитет у Новом Саду
Наслов рада:	Примена савремених метахеуристичких алгоритама у оптимизацији технолошких процеса обраде производа
Језик публикације (писмо):	Српски (латиница)
Физички опис рада:	Страница 168 Поглавља 9 Референци 177 Табела 57 Слика 62 Графикона 0 Прилога 5
Научна област:	Машинско инжењерство
Ужа научна област (научна дисциплина):	Пројектовање технолошких процеса
Кључне речи / предметна одредница:	Оптимизација технолошких процеса, метахеуристички алгоритми, ограничења претхођења
Резиме на језику рада:	Оптимизација технолошких процеса припада групи комплексних проблема код којих је акценат стављен на одређивање редоследа захвата обраде и оптималну селекцију варијанти технолошких ресурса међу којима се издвајају машине алатке, резни алати и смерови прилаза резних алата. Оптимизација се врши минимизирањем функције циља која је формулисана на бази трошкова, односно времена реализације захвата обраде делова призматичног и ротационог облика. Поред тога, правила и односи претхођења међу типским технолошким облицима и захватима обраде формирају тзв. ограничења претхођења која омогућавају проналажење изводљивих решења усклађених са технолошким захтевима разматраних машинских делова. Предложени метахеуристички алгоритми за решавање овог проблема су алгоритми врне, сивог вука и грбавог кита. Поред теоријске анализе ових метода извршена је верификација њихових перформанси на шест различитих експерименталних студија.

¹ Аутор докторске дисертације потписао је и приложио следеће Обрасце:

5б – Изјава о ауторству;

5в – Изјава о истоветности штампане и електронске верзије и о личним подацима;

5г – Изјава о коришћењу.

Ове Изјаве се чувају на факултету у штампаном и електронском облику и не кориче се са тезом.

Датум прихватања теме од стране надлежног већа:	11.07.2019.
Датум одбране: (Попуњава одговарајућа служба)	
Чланови комисије: (титула, име, презиме, звање, институција)	<p>Председник: др Миленко Секулић, редовни професор, Факултет техничких наука, Универзитет у Новом Саду</p> <p>Члан: др Зоран Миљковић, редовни професор, Машински факултет, Универзитет у Београду</p> <p>Члан: др Ацо Антић, редовни професор, Факултет техничких наука, Универзитет у Новом Саду</p> <p>Члан: др Дејан Лукић, ванредни професор, Факултет техничких наука, Универзитет у Новом Саду</p> <p>Члан, ментор: др Мијодраг Милошевић, ванредни професор, Факултет техничких наука, Универзитет у Новом Саду</p>
Напомена:	

KEY WORD DOCUMENTATION²

Document type:	Doctoral dissertation
Author:	Mića Đurđev
Supervisor (title, first name, last name, position, institution)	Mijodrag Milošević, PhD, associate professor, Faculty of Technical Sciences, University of Novi Sad
Thesis title:	Application of modern metaheuristic algorithms in optimization of process planning
Language of text (script):	Serbian language (latin)
Physical description:	Pages 168 Chapters 9 References 177 Tables 57 Illustrations 62 Graphs 0 Appendices 5
Scientific field:	Mechanical Engineering
Scientific subfield (scientific discipline):	Process planning
Subject, Key words:	Optimization of process planning, metaheuristic algorithms, precedence constraints
Abstract in English language:	Optimization of process planning belongs to the group of complex problems in which the emphasis is placed on determining the sequence of machining operations and the optimal selection of variants of technological resources such as machines, cutting tools and tool approach directions. Optimization is achieved by minimizing the objective function which is formulated on the basis of cost and time required for performing all the operations for prismatic or rotational parts. In addition, precedence rules and relationships among features and machining operations define so called precedence constraints which aid in finding feasible solutions that are complied with technological requirements of considered mechanical parts. The proposed metaheuristic algorithms for solving this problem are crow search optimization, grey wolf optimizer and whale optimization algorithm. Beside the theoretical analysis of these methods, verification of their performances was done on six different experimental studies.

² The author of doctoral dissertation has signed the following Statements:

56 – Statement on the authority,

5b – Statement that the printed and e-version of doctoral dissertation are identical and about personal data,

5r – Statement on copyright licenses.

The paper and e-versions of Statements are held at the faculty and are not included into the printed thesis.

Accepted on Scientific Board on:	11.07.2019.
Defended: (Filled by the faculty service)	
Thesis Defend Board: (title, first name, last name, position, institution)	<p>President: Milenko Sekulić, PhD, full professor, Faculty of Technical Sciences, University of Novi Sad</p> <p>Member: Zoran Miljković, PhD, full professor, Faculty of Mechanical Engineering, University of Belgrade</p> <p>Member: Aco Antić, PhD, full professor, Faculty of Technical Sciences, University of Novi Sad</p> <p>Member: Dejan Lukić, PhD, associate professor, Faculty of Technical Sciences, University of Novi Sad</p> <p>Member, mentor: Mijodrag Milošević, PhD, associate professor, Faculty of Technical Sciences, University of Novi Sad</p>
Note:	

SADRŽAJ

PREGLED SLIKA	I
PREGLED TABELA	IV
PREGLED OZNAKA.....	VII
1. UVODNA RAZMATRANJA.....	1
1.1 Predmet istraživanja.....	1
1.2 Cilj istraživanja i hipoteze	4
1.3 Opis strukture doktorske disertacije.....	5
2. OPTIMIZACIJA I VARIJANTNOST TEHNOLOŠKIH PROCESA OBRADE	8
2.1 Varijantnost tehnoloških procesa obrade i problem optimizacije.....	8
2.2 Uslovi i ograničenja prethođenja u optimizaciji tehnoloških procesa obrade	17
2.2.1 Dimenziono prethođenje	17
2.2.2 Geometrijsko prethođenje	17
2.2.3 Tehnološko prethođenje.....	18
2.2.4 Ekonomsko prethođenje.....	19
2.2.5 Ograničenja prethođenja	19
2.2.6 Predstavljanje varijantnih tehnoloških procesa obrade	21
2.3 Matematički model za optimizaciju tehnoloških procesa obrade	25
3. METAHEURISTIČKI ALGORITMI	31
3.1 Osnovni pojmovi	31
3.2 Istorijat i razvoj metaheurističkih algoritama	37
4. PREGLED STANJA U OBLASTI OPTIMIZACIJE TEHNOLOŠKIH PROCESA OBRADE PROIZVODA	41
4.1 Genetski algoritmi u optimizaciji tehnoloških procesa obrade	41
4.2 Optimizacija rojem čestica u optimizaciji tehnoloških procesa obrade	45
4.3 Algoritam kolonije mrava u optimizaciji tehnoloških procesa obrade.....	46
4.4 Pregled ostalih algoritama u optimizaciji tehnoloških procesa obrade.....	47
5. SAVREMENI BIOLOŠKI INSPIRISANI ALGORITMI U OPTIMIZACIJI TEHNOLOŠKIH PROCESA OBRADE PROIZVODA.....	51
5.1 Modifikovani algoritam vrane (GCSA) u optimizaciji tehnoloških procesa obrade.....	51
5.1.1 Algoritam vrane (CSA)	51

5.1.1.1	Matematički model CSA algoritma	52
5.1.2	Modifikovani algoritam vrane (GCSA)	55
5.1.2.1	Vektorsko predstavljanje tehnološkog procesa	56
5.1.2.2	Kriterijum evaluacije varijantnih tehnoloških procesa	57
5.1.2.3	Mehanizam najbližeg resursa	58
5.1.2.4	Genetske komponente GCSA algoritma	59
5.1.2.5	Mehanizam popravke jedinki	63
5.2	Modifikovani algoritam sivog vuka (mGWO) u optimizaciji tehnoloških procesa obrade.....	65
5.2.1	Algoritam sivog vuka (GWO).....	65
5.2.1.1	Matematički model GWO algoritma	67
5.2.2	Modifikovani algoritam sivog vuka (mGWO).....	70
5.2.2.1	Kodiranje i dekodiranje rešenja.....	71
5.2.2.2	Matematički model mGWO algoritma.....	73
5.2.2.3	Kriterijum evaluacije varijantnih tehnoloških procesa	74
5.2.2.4	Genetske operacije.....	74
5.3	Modifikovani algoritam grbavog kita u optimizaciji tehnoloških procesa obrade.....	77
5.3.1	Tradicionalni algoritam grbavog kita (WOA)	77
5.3.1.1	Matematički model tradicionalnog WOA algoritma	78
5.3.2	Modifikovani algoritam grbavog kita (mWOA).....	81
5.3.2.1	Matematički model mWOA algoritma.....	82
6.	ANALIZA EKSPERIMENTALNIH REZULTATA OPTIMIZACIJE TEHNOLOŠKIH PROCESA OBRADE PROIZVODA.....	85
6.1	Podešavanje parametara i analiza rezultata prve, druge i treće eksperimentalne studije.....	86
6.1.1	Prva eksperimentalna studija	89
6.1.1.1	Diskusija rezultata prve eksperimentalne studije.....	101
6.1.2	Druga eksperimentalna studija	102
6.1.2.1	Diskusija rezultata druge eksperimentalne studije	109
6.1.3	Treća eksperimentalna studija.....	110
6.1.3.1	Diskusija rezultata treće eksperimentalne studije.....	116
6.2	Podešavanje parametara i analiza rezultata četvrte eksperimentalne studije.....	116
6.2.1	Četvrta eksperimentalna studija	119
6.2.1.1	Diskusija rezultata četvrte eksperimentalne studije.....	130

6.3 Podešavanje parametara i analiza rezultata pete i šeste eksperimentalne studije.....	131
6.3.1 Peta eksperimentalna studija	132
6.3.1.1 Diskusija rezultata pete eksperimentalne studije.....	139
6.3.2 Šesta eksperimentalna studija.....	139
6.3.2.1 Diskusija rezultata šeste eksperimentalne studije	145
7. ZAKLJUČNA RAZMATRANJA	146
8. LITERATURA.....	150
9. PRILOZI	163
Prilog 1. Funkcija cilja – f_c (Izvorni kod u Matlab-u).....	163
Prilog 2. Mehanizam najbliže mašine (Izvorni kod u Matlab-u)	165
Prilog 3. Mehanizam popravke rešenja (Izvorni kod u Matlab-u).....	166
Prilog 4. 3SX operator ukrštanja (Izvorni kod u Matlab-u).....	167
Prilog 5. Mutacija resursa (vektora mašina) (Izvorni kod u Matlab-u).....	168

PREGLED SLIKA

Slika 2.1	Prostor odlučivanja pri projektovanju tehnoloških procesa	9
Slika 2.2.	Primeri tipskih tehnoloških oblika za mašinsku obradu.....	11
Slika 2.3	Struktura varijantnih tehnoloških procesa za jedan mašinski deo.....	12
Slika 2.4.	Trodimenzionalni modeli/Dvodimenzionalni crteži reprezentativnih proizvoda kao predmeta optimizacije tehnoloških procesa	13
Slika 2.5	Solid model prizmatičnog dela sa tipskim oblicima	14
Slika 2.6.	Obrada otvora omogućena u dva različita SPA	15
Slika 2.7.	I/II Mreža fleksibilnih tehnoloških procesa obrade	22
Slika 2.8.	Graf i matrica prethođenja zahvata obrade	23
Slika 2.9.	Vektorsko predstavljanje redosleda zahvata	24
Slika 3.1.	Klasifikacija metaheurističkih algoritama prema izvoru inspiracije	34
Slika 3.2.	Savremeni biološki inspirisani metaheuristički algoritmi	35
Slika 3.3.	Intenzifikacija i diverzifikacija metaheurističkih algoritama	36
Slika 3.4.	Intenzifikacija i diverzifikacija izraženi pomoću krive konvergencije	37
Slika 3.5.	Broj objavljenih radova u prethodnih deset godina koji sadrže ključnu reč „metaheuristic“	37
Slika 5.1.	Vrane u potrazi za hranom	52
Slika 5.2.	Varijacija parametra dužine leta vrane za dva scenarija.....	53
Slika 5.3.	Dijagram toka GCSA algoritma za optimizaciju tehnoloških procesa	56
Slika 5.4.	Turnirska selekcija sa vranama jedinkama	60
Slika 5.5.	3SX operator ukrštanja	61
Slika 5.6.	Operator mutacije zamene	61
Slika 5.7.	Operator mutacije vektora mašine (mutacija resursa)	62
Slika 5.8.	Logička procedura algoritma popravke rešenja	64
Slika 5.9.	Metod lova sivih vukova na bizona: od (A) prilaženja, gonjenja i praćenja, (B-D) maltretiranja i opkoljavanja plena do (E) stacionarne faze i finalnog napada	67
Slika 5.10.	Dvodimenzionalni vektori položaja i njihove potencijalne lokacije	68
Slika 5.11.	Ažuriranje položaja prema alfa, beta i delta vukovima.....	69
Slika 5.12.	Napad na plen - konvergencija (a) i potraga za plenom – divergencija (b) ..	69
Slika 5.13.	Dijagram toka mGWO algoritma za optimizaciju tehnoloških procesa.....	71
Slika 5.14.	Operator jednopozicionog ukrštanja	75
Slika 5.15.	Operator mutacije zamene	76
Slika 5.16.	Operator mutacije resursa	77

Slika 5.17. Tehnika lova stvaranjem mreže mehurića: (a) snimak iz vazduha i (b) šematski prikaz	78
Slika 5.18. Tehnika stvaranja mreže mehurića kroz mehanizme dva kretanja: (a) sužavanje kruga oko plena i (b) spiralno ažuriranje položaja	79
Slika 5.19. Mehanizam potrage za plenom [32]	80
Slika 5.20. Dijagram toka mWOA algoritma za optimizaciju tehnoloških procesa	82
Slika 6.1. Optimizacioni grafik uticaja ulaznih parametara na predviđeni odziv (MIN i SV troškova TWMC)	89
Slika 6.2. Trodimenzionalni solid model u konceptualnom (a) i transparentnom (b) prikazu i graf prethođenja zahvata (c) za prvi eksperiment	90
Slika 6.3. Matrica prethođenja za prvi eksperiment	91
Slika 6.4. Dijagrami uporednih rezultata ostvoreni različitim metaheurističkim algoritmima za prvi eksperiment i prvi tehnološki uslov	95
Slika 6.5 Dijagrami uporednih rezultata ostvoreni različitim metaheurističkim algoritmima za prvi eksperiment i drugi tehnološki uslov	95
Slika 6.6 Dijagrami uporednih rezultata ostvoreni različitim metaheurističkim algoritmima za prvi eksperiment i treći tehnološki uslov	96
Slika 6.7. Krive konvergencije za prvi eksperiment i tri tehnološka uslova	100
Slika 6.8. Trodimenzionalni solid model u konceptualnom (a) i transparentnom (b) prikazu, graf (c) i matrica prethođenja (d) za drugi eksperiment	103
Slika 6.9. Uporedni rezultati različitih algoritama za drugi reprezentativni deo i prvi tehnološki uslov	106
Slika 6.10. Uporedni rezultati različitih algoritama za drugi reprezentativni deo i drugi tehnološki uslov	106
Slika 6.11. Krive konvergencije za drugi eksperiment	109
Slika 6.12. Trodimenzionalni solid model u konceptualnom (a) i transparentnom (b) režimu, nepovezani graf prethođenja (c) i matrica prethođenja (d) za treći eksperiment	111
Slika 6.13. Uporedni rezultati ostvoreni različitim metaheurističkim algoritmima za treći eksperiment	113
Slika 6.14. Krive konvergencije za treći eksperiment	114
Slika 6.15. Optimizacioni grafik uticaja ulaznih parametara na predviđeni odziv (MIN, MAX i SV TWMC)	118
Slika 6.16. Trodimenzionalni solid model u konceptualnom prikazu za četvrti reprezentativni deo	119
Slika 6.17. Graf prethođenja (a) i matrica prethođenja (b) za četvrti reprezentativni deo	120
Slika 6.18. Uporedni rezultati ostvoreni različitim metaheurističkim metodama za četvrti eksperiment deo i prvi tehnološki uslov	124

Slika 6.19. Uporedni rezultati ostvareni različitim metaheurističkim metodama za četvrti eksperiment deo i drugi tehnološki uslov	125
Slika 6.20. Krive konvergencije za četvrti eksperiment i dva tehnološka uslova	130
Slika 6.21. Trodimenzionalni solid model rotacionog dela (a) i I/ILI mreža za peti eksperiment i reprezentativni deo rotacionog oblika.....	133
Slika 6.22. Uporedni rezultati različitih metaheurističkih algoritama za peti eksperiment (minimalni troškovi TWMT)	136
Slika 6.23. Uporedni rezultati različitih metaheurističkih algoritama za peti eksperiment (minimalni, maksimalni i srednji troškovi TWMT)	137
Slika 6.24. Krive konvergencije za peti reprezentativni deo	137
Slika 6.25. Trodimenzionalni model za šesti eksperiment	139
Slika 6.26. I/ILI mreža (b) za šesti reprezentativni deo rotacionog oblika.....	140
Slika 6.27. Uporedni rezultati ostvareni različitim metaheurističkim algoritmima za šesti eksperiment.....	143
Slika 6.28 Krive konvergencije za šesti eksperiment.....	145

PREGLED TABELA

Tabela 2.1. Pravila tehnološkog prethođenja obrade otvora i rupa (za mere 10 do 30 mm)	18
Tabela 2.2. Tvrd ograničenja	20
Tabela 2.3. Meka ograničenja	21
Tabela 2.4. Tipovi podataka	25
Tabela 2.5. Definicija promene alata	27
Tabela 2.6. Definicija promene stezanja dela	28
Tabela 5.1. Vektorsko predstavljanje varijantnog tehnološkog procesa	56
Tabela 5.2. Šema kodiranja varijantnog tehnološkog procesa za šesti eksperiment	72
Tabela 5.3. Dekodirani varijantni tehnološki proces iz tabele 5.2	72
Tabela 6.1. Delimični ortogonalni plan eksperimenta 2^5 sa izlaznim statističkim rezultatima	88
Tabela 6.2. Parametri odziva sa usvojenim limitima i ciljnim vrednostima	88
Tabela 6.3. Optimalni ulazni parametri GCSA-Par1	89
Tabela 6.4. Informacije o tipskim oblicima, zahvatima, resursima i ograničenjima za prvi eksperiment	91
Tabela 6.5. Odnosi prethođenja među tipskim oblicima i zahvatima za prvi eksperiment	92
Tabela 6.6. Informacije o troškovima za prvi eksperiment	92
Tabela 6.7. Uporedni rezultati GCSA i drugih metaheurističkih algoritama za prvi eksperiment	93
Tabela 6.8. Najbolji tehnološki procesi za prvi eksperiment i tri tehnološka uslova sa ulaznim parametrima u okviru „GCSA-Par1“	94
Tabela 6.9. Najbolji tehnološki procesi za prvi eksperiment i tri tehnološka uslova sa ulaznim parametrima u okviru „GCSA-Par2“	94
Tabela 6.10. Izlazni rezultati prvog eksperimenta za prvi tehnološki uslov u 20 ponavljanja	97
Tabela 6.11. Izlazni rezultati prvog eksperimenta za drugi tehnološki uslov u 20 ponavljanja	98
Tabela 6.12. Izlazni rezultati prvog eksperimenta za treći tehnološki uslov u 20 ponavljanja	99
Tabela 6.13. Informacije o tipskim oblicima, zahvatima i tehnološkim resursima za drugi eksperiment	104

Tabela 6.14. Ograničenja prethođenja među tipskim oblicima i zahvatima za drugi eksperiment	104
Tabela 6.15. Informacije o troškovima za drugi eksperiment	105
Tabela 6.16. Optimalni tehnološki procesi za drugi reprezentativni deo i dva tehnološka uslova	105
Tabela 6.17. Usporedni rezultati optimizacije za drugi reprezentativni deo.....	106
Tabela 6.18. Izlazni rezultati drugog eksperimenta za prvi tehnološki uslov u 20 ponavljanja	107
Tabela 6.19. Izlazni rezultati drugog eksperimenta za drugi tehnološki uslov u 20 ponavljanja	108
Tabela 6.20. Informacije o tipskim oblicima, zahvatima, resursima i ograničenjima prethođenja za treći eksperiment	112
Tabela 6.21. Odnosi prethođenja među zahvatima obrade za treći eksperiment.....	112
Tabela 6.22. Informacija o troškovima za treći prizmatični deo	112
Tabela 6.23. Optimalni tehnološki proces za treći eksperiment	113
Tabela 6.24. Usporedni rezultati GCSA-Par2 sa drugim algoritmima	113
Tabela 6.25. Izlazni rezultati trećeg eksperimenta dobijeni u 20 ponavljanja	115
Tabela 6.26. Delimični ortogonalni plan eksperimenta 2^5 sa izlaznim statističkim rezultatima	117
Tabela 6.27. Parametri odziva sa usvojenim limitima i ciljnim vrednostima za četvrti eksperiment	118
Tabela 6.28. Optimalni ulazni parametri za „GCSA-Par3“.....	118
Tabela 6.29. Tehnološke informacije koje se tiču tipskih tehnoloških oblika, zahvata, resursa i ograničenja prethođenja za četvrti reprezentativni deo	121
Tabela 6.30. Informacija o troškovima za četvrti reprezentativni deo	123
Tabela 6.31. Usporedni rezultati GCSA-Par3 sa drugim metaheurističkim algoritmima za četvrti eksperiment.....	124
Tabela 6.32. Optimalni tehnološki proces za četvrti eksperiment i prvi tehnološki uslov.....	126
Tabela 6.33. Najbolji tehnološki proces za četvrti eksperiment i drugi tehnološki uslov.....	127
Tabela 6.34. Izlazni rezultati četvrtog eksperimenta za prvi tehnološki uslov dobijeni u 20 ponavljanja	128
Tabela 6.35. Izlazni rezultati četvrtog eksperimenta za drugi tehnološki uslov dobijeni u 20 ponavljanja	129
Tabela 6.36. Informacije o tipskim oblicima, zahvatima i alternativnim mašinama za peti reprezentativni deo.....	133

Tabela 6.37. Detaljne informacije o zahvatima i podoperacijama za peti eksperiment	134
Tabela 6.38. Varijantne mašine i transportna vremena između mašina	135
Tabela 6.39. Šema kodiranja varijantnog tehnološkog procesa za peti	135
Tabela 6.40. Uporedni rezultati za peti eksperiment	136
Tabela 6.41. Najbolji tehnološki proces za peti reprezentativni deo - mGWO.....	136
Tabela 6.42. Izlazni rezultati petog eksperimenta dobijeni u 20 ponavljanja	138
Tabela 6.43. Informacije o tipskim oblicima, zahvatima i tehnološkim resursima za šesti reprezentativni deo	141
Tabela 6.44. Transportna vremena između mašina i vremenski indeksi za šesti reprezentativni deo	142
Tabela 6.45. Uporedni rezultati za šesti reprezentativni deo	142
Tabela 6.46. Optimalni tehnološki proces za šesti eksperiment – CSA.....	143
Tabela 6.47. Optimalni tehnološki proces za šesti eksperiment – mGWO i mWOA....	143
Tabela 6.48. Izlazni rezultati šestog eksperimenta dobijeni u 20 ponavljanja	144

PREGLED OZNAKA

Oznaka	Naziv na engleskom jeziku	Naziv na srpskom jeziku
3SX	Three-string Crossover	3SX ukrštanje
ACO	Ant Colony Optimization	Algoritam kolonije mrava
AP	Awareness Probability	Verovatnoća svesnosti vrane
APC	Additional Penalty Cost	Dodatni troškovi
BA	Bat Algorithm	Algoritam slepih miševa
CAD	Computer Aided Design	Računarom podržano projektovanje proizvoda
CAM	Computer Aided Manufacturing	Računarom podržana proizvodnja
CAPP	Computer Aided Process Planning	Računarom podržano projektovanje tehnoloških procesa
CSA	Crow Search Algorithm	Algoritam vrane
fl	Flight Length	Dužina leta vrane
GA	Genetic Algorithm	Genetski algoritam
GCSA	Genetic Crow Search Algorithm	Modifikovani algoritam vrane
GP	Genetic Programming	Genetsko programiranje
GP	Precedence graph	Graf prethođenja
GWO	Grey Wolf Optimizer	Algoritam sivog vuka
HBMO	Honey Bee Mating Optimization	Algoritam inspirisan parenjem pčela
MAX	Maximal Value	Maksimalna vrednost
MILP	Mixed Integer Linear Programming	Mešovito celobrojno linearno programiranje
MIN	Minimal Value	Minimalna vrednost
NAH	Nawaz Enscore Ham	NAH heuristika
NFL	„No Free Lunch“	„Nema besplatnog ručka“
NMC	Number of Machine Changes	Broj promena mašina
NS	Number of Setups	Broj stezanja
NSC	Number of Setup Changes	Broj promena stezanja
NTC	Number of Tool Changes	Broj promena alata

NVC	Number of Violated Constraints	Broj narušenih ograničenja
OTP	Process Planning Optimization	Optimizacija tehnoloških procesa
OX	Order Crossover	Redno ukrštanje
PMX	Partially Mapped Crossover	Delimično mapirano ukrštanje
PSO	Particle Swarm Optimization	Optimizacija rojem čestica
SA	Simulated Annealing	Simulirano žarenje
SCT	Setup Change Time	Ukupno vreme promene stezanja dela
SPA	Tool Approach Direction	Smer prilaza alata
STD	Standard Deviation	Standardna devijacija
SV	Mean Value	Srednja vrednost
TMC	Total Machine Cost	Ukupni troškovi rada mašina
TMCC	Total Machine Change Cost	Ukupni troškovi promena mašina
TMT	Total Manufacturing Time	Ukupno vreme obrade dela
TS	Tabu Search	Tabu pretraživanje
TSC	Total Setup Cost	Ukupni troškovi promene stezanja dela
TT	Transportation Time	Ukupno vreme transporta delova
TTC	Total Tool Cost	Ukupni troškovi korišćenja alata
TTCC	Total Tool Change Cost	Ukupni troškovi promena alata
TTCT	Total Tool Change Time	Ukupno vreme promene alata
TWMC	Total Weighted Manufacturing Cost	Ukupni težinski troškovi obrade dela
TWMT	Total Weighted Manufacturing Time	Ukupno težinsko vreme obrade dela
WOA	Whale Optimization Algorithm	Algoritam grbavog kita

1. UVODNA RAZMATRANJA

1.1 Predmet istraživanja

Intenzivan razvoj informacionih tehnologija i njihova primena u proizvodnji uveliko je doprinela ispunjenju visokih zahteva tržišta u pogledu kvaliteta proizvoda, kao i vremena i troškova njihove proizvodnje. Automatizacija i pojava fleksibilnih tehnoloških sistema omogućila je niz prednosti koje su vrhunac dostigle u poslednjih nekoliko godina uvođenjem koncepta „inteligentnog“ ponašanja u proizvodne procese. Rezultat toga doveo je do razvoja nove oblasti koja nosi naziv „inteligentni tehnološki sistemi“. Kao najviša klasa fleksibilnih tehnoloških sistema, inteligentni tehnološki sistemi, uz primenu savremenih informacionih tehnologija baziranih na implementaciji biološki inspirisanih tehnika veštačke inteligencije, poseduju sposobnost efektivnog adaptiranja na promene koje okruženje nameće [1,2].

Projektovanje tehnoloških procesa, odnosno tehnološka priprema proizvodnje predstavlja, između ostalih, neizostavnu funkciju inteligentnih tehnoloških sistema. Cilj projektovanja tehnoloških procesa je da zadovolji zahteve projektanata sa jedne strane i tehnološke zahteve proizvodnje sa druge strane, formirajući na taj način „most“ između projektovanja proizvoda i procesa njihove proizvodnje [3]. Razvoj sistema za projektovanje tehnoloških procesa izrade proizvoda uz podršku savremenih informacionih tehnologija (CAPP) predstavlja kompleksan zadatak sastavljen iz velikog broja složenih aktivnosti. Tu spadaju: prihvatanje i analiza ulaznih projektnih zadataka, izdvajanje i prepoznavanje tipskih tehnoloških oblika, izbor i definisanje polaznog materijala (priprema), definisanje sadržaja tehnološkog procesa, definisanje operacija i podoperacija izrade delova, izbor i definisanje zahvata obrade i određivanje njihovog redosleda izvođenja, izbor odgovarajućih mašina alatki, alata, pribora i merila, izbor i definisanje parametara i strategije obrade, određivanje troškova i vremena proizvodnje, generisanje upravljačkih programa za numerički upravljane obradne i tehnološke sisteme kao i generisanje odgovarajuće tehnološke dokumentacije [3-5]. Imajući u vidu navedeni skup aktivnosti, može se zaključiti da CAPP sistemi predstavljaju ključnu komponentu u integraciji računarom podržanog projektovanja proizvoda (CAD) i računarom podržane proizvodnje (CAM).

U okviru ovog istraživanja posebna pažnja će biti usmerena na tehnoeкономsku optimizaciju tehnoloških procesa izrade proizvoda. Uopšteno, optimizacija se može protumačiti kao kolekcija matematičkih principa i tehnika koja za cilj ima pronalaženje najpogodnijeg rešenja uz poštovanje postavljenih uslova. Optimizacija je pronašla veliku primenu u svim granama nauke i tehnike, pa samim tim i u proizvodnji. Kada se govori o tehnoeкономskoj optimizaciji, radi se o zadatku čiji je cilj ispunjavanje tehničkih zahteva kvaliteta proizvoda sa jedne strane, kao što su funkcionalnost, pouzdanost, tehnologičnost, vek upotrebe, oblik, tačnost obrade ili dizajn, i ekonomskih zahteva sa druge strane, među kojima preovladavaju proizvodnost, troškovi i vreme izrade, cena proizvoda kao i njegov tržišni vek [3,6,7]. Da bi se postupak tehnoeкономske optimizacije sproveo, potrebno je prethodno definisati funkciju cilja, odnosno kriterijum optimizacije.

Jedna od osnovnih osobina tehnoloških procesa izrade proizvoda jeste njihova višestruka varijantnost koja se javlja u svim fazama projektovanja. Na varijantnost tehnoloških procesa najveći uticaj imaju vrsta priprema, vrsta i redosled operacija i zahvata obrade, proizvodni resursi kao i tehnoeкономski efekti [3,5]. Analiza različitih varijanti tehnoloških procesa obrade zahteva dekomponovanje tehnoloških rešenja, pri čemu se vrši spoljašnja i unutrašnja optimizacija. Spoljašnja optimizacija je usmerena na izbor optimalnih elemenata tehnološkog procesa na makro nivou, a unutrašnja optimizacija na izbor optimalnih elemenata tehnološkog procesa na mikro nivou [3,6,7].

Težište istraživanja u ovoj disertaciji usmereno je na varijantnost redosleda izvođenja zahvata obrade i varijantnost tehnoloških resursa, koji predstavljaju bitne odlike varijantnosti tehnoloških procesa obrade proizvoda. Ove varijantnosti imaju veliki uticaj na proces spoljašnje optimizacije, kao dela projektovanja tehnoloških procesa na makro nivou, što predstavlja jednu od faza tehnološke pripreme proizvodnje. U okviru ove faze izvršena je dekompozicija reprezentativnih prizmatičnih i rotacionih delova do nivoa tipskih tehnoloških oblika za koje su definisane odgovarajuće varijante zahvata obrade. Odabrana varijanta ili varijante zahvata za svaki tipski tehnološki oblik formiraju niz celobrojnih vrednosti (permutaciju) koji predstavlja redosled izvođenja zahvata obrade za posmatrani mašinski deo. Na ovaj način izražava se fleksibilnost tehnološkog procesa obrade u pogledu određivanja optimalnog redosleda zahvata obrade. Rezultat ovih fleksibilnosti doveo je do pojave kompleksnog problema optimizacije koji je postao vrlo popularan u naučnoj literaturi [2,8-17].

Problem optimizacije fleksibilnih tehnoloških procesa obrade sveden je na problem određivanja optimalnog redosleda zahvata obrade uz optimalnu selekciju odgovarajuće mašine alatke, reznog alata i smera prilaza alata, odnosno neophodnih tehnoloških resursa za realizaciju selektovanih zahvata obrade [8,11]. Pored toga, ne sme se zanemariti postojanje prioriternih ograničenja koja se definišu na osnovu pravila prethođenja između zahvata i tipskih oblika. Ova ograničenja direktno obezbeđuju generisanje izvodljivog, a istovremeno tehnologičnog redosleda zahvata obrade [8,18,19]. Prioritetna ograničenja ili ograničenja prethođenja numerički se definišu pomoću matrica, dok se za grafički prikaz koriste grafovi ili I/II mreže [2,16,20,21].

Osnovni zadatak u predmetnom istraživanju pripada grupi problema kombinatorne optimizacije koji zajedno sa celobrojnim programiranjem čine dve osnovne grane diskretne optimizacije. Njihova osnovna karakteristika jeste pronalaženje rešenja u okviru

diskretnog, odnosno konačnog skupa mogućih rešenja [22]. Sa druge strane, uzimajući u obzir dimenzije kombinatornog problema optimizacije inteligentnih tehnoloških procesa, ovaj problem se svrstava u klasu teških nedeterminističko polinomialnih (NP) problema koje prvenstveno karakterišu visoki zahtevi u pogledu računarskih resursa (prostorna složenost) kao i nemogućnost pronalaska optimalnog rešenja u realnom vremenskom periodu (vremenska složenost) [23]. Porastom dimenzija problema, na primer, broja zahvata, broja mašina alatki, broja reznih alata i/ili broja smerova prilaza alata, vreme pretraživanja i zahtevi za računarskim resursima značajno rastu.

Na bazi iznetih informacija, rešavanje problema optimizacije fleksibilnih tehnoloških procesa može se ostvariti primenom metaheurističkog pristupa. Metaheuristike predstavljaju stohastičke metode na osnovu kojih se vrši pretraživanje prostora mogućih rešenja i na taj način dolazi do „kvalitetnih“ rešenja u zadovoljavajućem vremenskom periodu [24,25]. Pod „kvalitetnim“ rešenjima smatraju se ona rešenja koja su prihvatljiva, odnosno bliska optimalnom jer ne postoji garancija da je pronađeni rezultat zaista optimalan. Stoga, metaheuristike imaju ulogu alata pomoću kojih je moguće uspešno rešiti „težak“ optimizacioni problem ili makar pronaći rešenje koje će u određenoj meri zadovoljiti prethodno postavljeni kriterijum. Metaheuristike karakteriše univerzalnost, jednostavnost i stohastičnost, a glavni zadatak im je ostvarenje ravnoteže između lokalnog i globalnog pretraživanja, odnosno težnje da se što intenzivnije pretraže regioni u prostoru pretrage (intenzifikacija) i uvedu elementi slučajnosti pomoću kojih će se pretražiti regioni koji nisu istraženi (diverzifikacija) [25].

Metaheuristike se implementiraju u formi algoritama i veliki broj njih baziran je na oponašanju inteligentnih društvenih, a prvenstveno bioloških procesa koji su zastupljeni u našem okruženju, kako društvenom tako i prirodnom. Grana veštačke inteligencije koja proučava ove algoritme sa aspekta biološke inteligencije i koja je dostigla veliku popularnost i ekspanziju u poslednje dve decenije poznata je pod nazivom „inteligencija rojeva“ [26-29]. Priroda je bogata jedinkama koje žive u zajednicama kao što su ribe, pčele, ptice ili mravi. Njihova kompleksna međusobna interakcija pomoću koje jedinke stiču relevantne informacije iz okruženja omogućava opstanak same zajednice i rezultat je pojave kolektivnih inteligentnih mehanizama. Ovi mehanizmi ogledaju se prvenstveno kada je u pitanju potraga za hranom (kod mrava ili riba), lov na plen (kod vukova ili kitova), krađa hrane (kod vrana) i mnogi drugi mehanizmi u prirodi koji su inspirisali razvoj velikog broja algoritamskih rešenja [30-36].

Postupak pretraživanja biološki inspirisanih metaheuristika zasnovan je na slučajno generisanoj populaciji agenata pretraživanja koji reprezentuju mrave, ribe, ptice ili neke druge žive organizme. Dinamika algoritma određena je matematičkim izrazima koji karakterišu, odnosno oponašaju posmatrani biološki proces i na osnovu kojih sama populacija evoluirala. Drugim rečima, jedinke u populaciji se vrednuju na osnovu postavljenog kriterijuma optimizacije i postupak evoluiranja se ponavlja kroz definisan broj iteracija. Cilj svakog algoritma iz oblasti inteligencije rojeva, a ujedno i svake metaheuristike, je da ostvari ravnotežu između lokalnog i globalnog pretraživanja. To nužno iziskuje podešavanje ulaznih parametara i u čestim slučajevima unapređenje algoritma modifikacijom postojećih operatora ili razvojem hibridnih pristupa.

Osnovni predmet istraživanja u ovom doktorskom radu odnosi se na optimizaciju tehnoloških procesa obrade delova prizmatičnog i rotacionog oblika. Fokus je postavljen na optimizaciju redosleda izvođenja zahvata obrade i optimalnu selekciju odgovarajućih tehnoloških resursa – mašina alatki, reznih alata i smerova prilaza alata (SPA). Uz poštovanje određenih pravila prethođenja na osnovu kojih se formiraju neophodna ograničenja, izvršena je optimizacija tehnoloških procesa obrade konkretnih delova, na bazi dva kriterijuma optimizacije: minimalnog ukupnog vremena i troškova obrade. U cilju rešavanja ovog kompleksnog problema kombinatorne optimizacije, izvršena je analiza i primena savremenih biološki inspirisanih metaheurističkih algoritama među kojima je akcenat postavljen na algoritam vrane (CSA), algoritam sivog vuka (GWO) i algoritam grbavog kita (WOA). Zbog neophodnosti adaptiranja korišćenih algoritama, razmatranom problemu optimizacije tehnoloških procesa obrade, izvršena je njihova modifikacija uvođenjem genetskih operatora i parametara. Ova modifikacija je omogućila unapređenje pretraživačkih kapaciteta CSA, GWO i WOA algoritama, u cilju poboljšanja i kasnije verifikacije njihovih performansi. Njihova implementacija i testiranje izvršeno je u Matlab-u, programskom jeziku četvrte generacije koji se prema literaturnim izvorima smatra vrlo pogodnim i pouzdanim matematičkim alatom za rešavanje složenih problema optimizacije.

1.2 Cilj istraživanja i hipoteze

Osnovni cilj istraživanja o posmatranoj doktorskoj disertaciji odnosi se na optimizaciju tehnoloških procesa obrade proizvoda primenom naprednih, modifikovanih, biološki inspirisanih metaheurističkih algoritama.

U skladu sa definisanim ciljem istraživanja, pažnja je usmerena ka unapređenju funkcije tehnološke pripreme proizvodnje kroz optimizaciju redosleda zahvata obrade sa ciljem ostvarenja tehnološke obrade mašinskih delova prizmatičnog i rotacionog oblika. Pored toga, akcenat je takođe postavljen na optimalnu selekciju odgovarajućih tehnoloških resursa među kojima se izdvajaju mašine alatke, rezni alati i smerovi prilaza alata, pomoću kojih se obezbeđuje obrada posmatranih tipskih oblika.

Prilikom izrade ove doktorske disertacije primenjene su sledeće naučne metode:

- *Metode analize i sinteze,*
- *Metode indukcije i dedukcije,*
- *Metode i tehnike projektovanja tehnoloških procesa,*
- *Metode tehnoeekonomske optimizacije,*
- *Metode veštačke inteligencije.*

U skladu sa definisanom metodologijom i ciljevima istraživanja postavljene su sledeće hipoteze:

Polazna hipoteza:

- *Primenom savremenih biološki inspirisanih metaheurističkih algoritama moguće je, u skladu sa postavljenim kriterijumom optimizacije generisati optimalan ili približno*

optimalan tehnološki proces obrade dela koji se može svesti na redosled izvođenja zahvata obrade uz pronalaženje optimalnih varijanti mašina alatki, reznih alata i smerova prilaza alata pri obradi.

Pomoćne hipoteze:

- *Moguće je primenom savremenih biološki inspirisanih metaheurističkih algoritama izvršiti minimizaciju ukupnih troškova i vremena obrade mašinskih delova prizmatičnog i rotacionog oblika.*
- *Optimizacija tehnoloških procesa obrade može se ostvariti primenom različitih biološki inspirisanih optimizacionih algoritama.*

1.3 Opis strukture doktorske disertacije

Zadatak u ovoj disertaciji odnosi se na analizu i primenu odgovarajućih savremenih i modifikovanih biološki inspirisanih metaheurističkih algoritama za optimizaciju tehnoloških procesa obrade prizmatičnih i rotacionih delova. Shodno posmatranom zadatku, istraživanje sprovedeno u radu je podeljeno u devet delova:

- *Definisanje predmeta i cilja istraživanja;*
- *Analiza problema optimizacije tehnoloških procesa sa naglaskom na varijantnost u pogledu zahvata obrade i tehnoloških resursa, formulisanje ograničenja na osnovu uslova i odnosa prethođenja među zahvatima i tipskim oblicima, i formiranje matematičkog modela kriterijuma minimizacije ukupnih troškova i vremena obrade;*
- *Opis, istorijat i savremeni prilazi u razvoju metaheurističkih algoritama;*
- *Analiza dosadašnjih istraživanja u oblasti primene metaheurističkih algoritama u optimizaciji tehnoloških procesa;*
- *Metodološki koncept savremenih biološki inspirisanih metaheurističkih algoritama sa naglaskom na algoritme vrane, sivog vuka i grbavog kita i njihove modifikacije za primenu u optimizaciji tehnoloških procesa;*
- *Eksperimentalno istraživanje i verifikacija savremenih modifikovanih biološki inspirisanih metaheurističkih algoritama na primerima obrade mašinskih delova prizmatičnog i rotacionog oblika;*
- *Donošenje zaključnih razmatranja i davanje pravaca za buduća istraživanja;*
- *Pregled korišćene literature;*
- *Priloženi materijal u vidu izvornog programskog koda.*

Prvi deo disertacije fokusiran je na definisanje predmeta i cilja istraživanja, kao i hipoteza istraživanja. Takođe je sažeto predstavljena struktura doktorske disertacije.

Drugi deo disertacije naglašava problem optimizacije tehnoloških procesa u okviru tehnološke pripreme proizvodnje. Pored opisa osnovnih elemenata fleksibilnih tehnoloških procesa, fokus je dat i na varijantnost tehnoloških procesa. Na primeru konkretnih mašinskih delova je ukazano na mogućnosti različitih varijanti obrade tipskih

tehnoloških oblika, sa aspekta zahvata obrade i tehnoloških resursa neophodnih za realizaciju tih zahvata (mašina alatki, reznih alata i smerova prilaza alata obratku). Dodatna pažnja posvećena je ograničenjima definisanim na bazi pravila i odnosa prethođenja koji se javljaju između zahvata, odnosno tipskih tehnoloških oblika. Različite varijante tehnoloških postupaka su prikazane pomoću grafova, matrica i I/II mreža. Na kraju drugog dela doktorske disertacije formulisani su matematički modeli funkcija cilja, koje su o ovom slučaju bile minimalni ukupni troškovi obrade i minimalno vreme obrade mašinskih delova prizmatičnog i rotacionog oblika.

Treći deo disertacije opisuje razvoj savremenih metaheurističkih algoritama. Pored uvodnih informacija o njihovom razvoju i istorijatu, izdvojene su njihove glavne prednosti na osnovu kojih su pronašli široku primenu u raznim područjima inženjerske optimizacije. Takođe, navedene su popularne klasifikacije metaheurističkih algoritama prema različitim kriterijumima, pri čemu posebna pažnja usmerava na klasifikaciju prema izvorima inspiracije (oponašanja), tj. na biološki inspirisane algoritme.

Četvrti deo prikazuje analizu dosadašnjih istraživanja i stanja u oblasti primene metaheurističkih algoritama za optimizaciju tehnoloških procesa obrade delova. S obzirom na popularnost ove oblasti značajan broj metaheurističkih algoritama pronašao je svoju primenu kod rešavanja ovog problema. Pored genetskih algoritama, ukratko su opisane primene algoritma optimizacije rojevima čestica, algoritma mravljih kolonija i drugih metaheurističkih algoritama u optimizaciji tehnoloških procesa obrade i pronalaženju optimalnih, odnosno približno optimalnih rešenja.

Peti deo disertacije bavi se teorijskom analizom biološki inspirisanih metaheurističkih algoritama. Pažnja je posvećena savremenim optimizacionim tehnikama, algoritmu vrane, algoritmu sivog vuka i algoritmu grbavog kita. Pored tradicionalnih karakteristika navedenih algoritama izvršena je analiza koncepta njihovih modifikacija koje su bazirane na uvođenju genetskih operatora. Sva modifikovana algoritamska rešenja predstavljena su na dijagramima tokova i u formi pseudokoda.

Šesti deo istraživanja u disertaciji ima za cilj da predstavi verifikaciju prethodno modifikovanih biološki inspirisanih algoritama optimizacije na šest eksperimentalnih studija koje razmatraju šest različitih mašinskih delova usvojenih iz literaturnih izvora. Svaki eksperiment, odnosno svaki reprezentativni mašinski deo predstavljen je u vidu trodimenzionalnog solid modela za koji su date tehnološke specifikacije u pogledu tipskih tehnoloških oblika, zahvata obrade, tehnoloških resursa za izvođenje zahvata obrade kao i odnosa prethođenja između zahvata i tipskih oblika. Za predstavljanje tehnoloških procesa korišćeni su grafovi, matrice i I/II mreže. Takođe, informacije o troškovima i vremenima obrade date su za svaki mašinski deo. Za svaki eksperiment izvršena je uporedna analiza dobijenih rezultata optimizacije sa rezultatima drugih studija, čime je omogućena verifikacija performansi predloženih modifikovanih algoritama.

Sedmi deo disertacije pruža zaključna razmatranja i daje kratak osvrt na pravce budućih istraživanja koji se odnose na primenu metaheurističkih algoritama u integrisanoj optimizaciji tehnoloških procesa i planova njihovog terminiranja u statičkim i dinamičkim proizvodnim okruženjima. Takođe, istaknut je potencijal primene ovih metoda u

inteligentnim servisima pametne proizvodnje što predstavlja jedan od osnovnih ciljeva koncepta Industrija 4.0.

U osmom delu dat je pregled korišćene literature, dok je u poslednjem, devetom delu doktorske disertacije prikazan izvorni kod određenih funkcija modifikovanih algoritama napisanih u Matlab programskom jeziku.

Glavna ideja ove doktorske disertacije je da se pokaže, da je primenom naprednih biološki inspirisanih optimizacionih algoritama, moguće poboljšati funkciju tehnološke pripreme proizvodnje, koja predstavlja usko grlo u tehničkoj pripremi proizvodnje. Pored toga, optimizacija tehnoloških procesa doprinosi povećanju efektivnosti inteligentnih tehnoloških sistema u okviru koncepta inteligentne proizvodnje koja naglašava primenu tehnika veštačke inteligencije. Usvojene metaheurističke metode odlikuju se relativno jednostavnom implementacijom koja može dovesti do optimalnih redosleda zahvata obrade i optimalni odabir tehnoloških resursa za obradu delova prizmatičnog i rotacionog oblika.

2. OPTIMIZACIJA I VARIJANTNOST TEHNOLOŠKIH PROCESA OBRADE

2.1 Varijantost tehnoloških procesa obrade i problem optimizacije

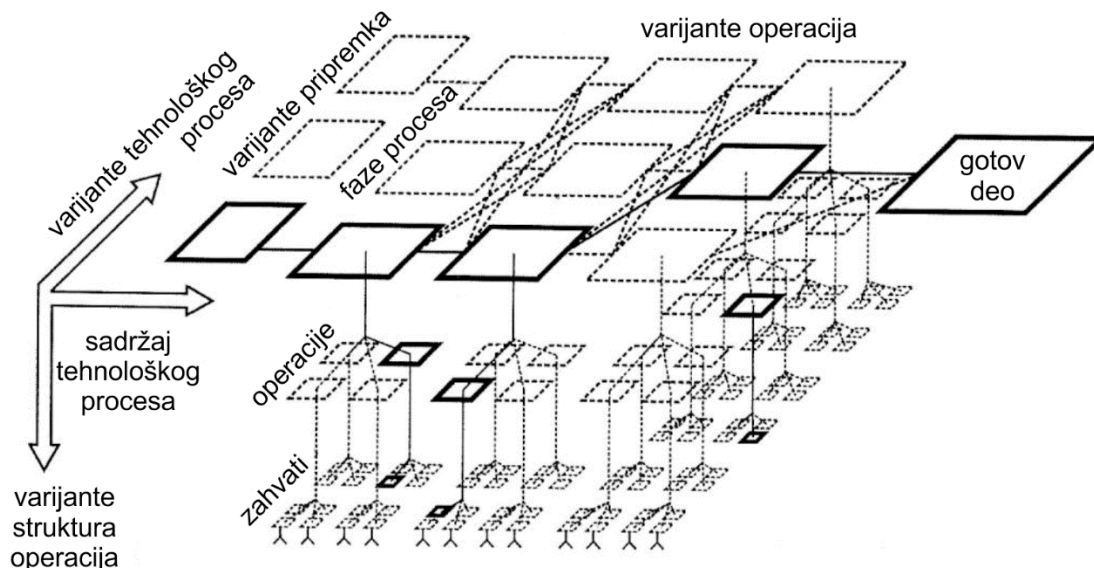
Priprema proizvodnje kao funkcija proizvodnih sistema ima veliki značaj za unapređenje procesa proizvodnje koji se, pre svega, oslikava kroz neophodnost detaljnog predviđanja, planiranja, pripreme i organizovanja proizvodnih aktivnosti. Sa jedne strane, tržište od proizvođača zahteva efikasniju proizvodnju, razvoj i poslovanje, dok sa druge strane, neophodno je brzo prilagođavanje zahtevima potrošača [5,37]. Delokrug rada pripreme proizvodnje odnosi se na razvoj i oblikovanje proizvoda, na planiranje i razradu dokumentacije i informacija za proizvodnju, kao i na sve mere i funkcije koje osiguravaju i kontrolišu realizaciju proizvodnje. Sistemski prilaz u radu pripreme proizvodnje je osnovni preduslov za ekonomičnu i produktivnu proizvodnju [38]. Priprema proizvodnje u industriji prerade metala deli se na tehničku i operativnu pripremu. Tehnička priprema proizvodnje obuhvata dve najvažnije tehničke funkcije proizvodnih sistema. Prva se odnosi na projektovanje proizvoda, poznata pod nazivom konstrukciona priprema, a druga na projektovanje tehnoloških procesa, poznata pod nazivom tehnološka priprema proizvodnje. Operativna priprema obuhvata druge dve funkcije koje se odnose na planiranje i upravljanje proizvodnjom. U okviru tehnološke pripreme proizvodnje ubrajaju se njeni osnovni zadaci [5]:

- *Analiza tehnološkičnosti konstrukcije proizvoda;*
- *Projektovanje tehnoloških procesa izrade proizvoda (u užem smislu);*
- *Optimizacija tehnoloških procesa;*
- *Konstruisanje specijalnih alata, pribora, merila i drugih uređaja;*
- *Simulacija i vizuelizacija tehnoloških procesa i drugih aktivnosti, itd.;*
- *Generisanje upravljačkih informacija za obradu, montažu, merenje, transport, rukovanje materijalom, itd.;*
- *Određivanje vremena i troškova proizvodnje;*
- *Definisanje vrste, karakteristika i količine pojedinih sredstava za rad;*
- *Definisanje uputstava za rad;*
- *Definisanje normativa materijala, vremena, itd.;*

- Definisanje vrsta i količina pomoćnog i potrošnog materijala;
- Definisanje složenosti procesa rada, stručnosti i broja radnika;
- Projektovanje tehnoloških osnova proizvodnog sistema;
- Projektovanje tehnoloških rešenja zaštite životne sredine od primenjenih tehnologija.

Razvoj informacionih tehnologija i sve veći tržišni zahtevi u pogledu smanjenja glavnog vremena proizvodnje doveli su do razvoja i napredovanja sistema za automatizaciju projektovanja tehnoloških procesa izrade proizvoda poznatijih po akronimu CAPP sistemi. Kao neizostavni sistem koji pruža podršku projektantima i tehnolozima i omogućuje ostvarenje integracije CAD i CAM sistema, CAPP predstavlja „most“ između projektovanja proizvoda i procesa njihove proizvodnje [3]. Osnovni cilj moderne tehnološke pripreme proizvodnje i razvoja CAPP sistema jeste projektovanje kvalitetnih tehnoloških procesa kojim se ostvaruje veza između projektovanja proizvoda i njegove izrade, odnosno proizvodnje.

Pre isticanja nekih od osnovnih osobina tehnoloških procesa, čije se projektovanje vrši u okvirima tehnološke pripreme proizvodnje, neophodno je definisati šta tehnološki proces generalno predstavlja. Tehnološki proces deo je proizvodnog procesa u kome se vrši izmena oblika, dimenzija, estetskog izgleda i unutrašnjih osobina materijala, odnosno proizvoda. Sastoji se od jedne ili više operacija obrade, koja se dalje može razložiti na jedan ili više zahvata obrade, pri čemu se zahvat obrade dalje dekomponuje na jedan ili više prolaza, a prolazi na veći broj pokreta. Varijantnost operacija, zahvata, prolaza, pokreta, pored varijantnosti pripremača sačinjava jedan okvirni prostor odlučivanja kod projektovanja tehnološkog procesa. Ovaj prostor može biti određen preko tri ose, kao što je prikazano na slici 2.1. Prva osa odnosi se na izbor faza, odnosno varijanti i redosleda operacija, druga na izbor varijanti strukture operacija, dok se treća odnosi na izbor varijanti tehnološkog procesa izrade proizvoda [3].



Slika 2.1 Prostor odlučivanja pri projektovanju tehnoloških procesa [3]

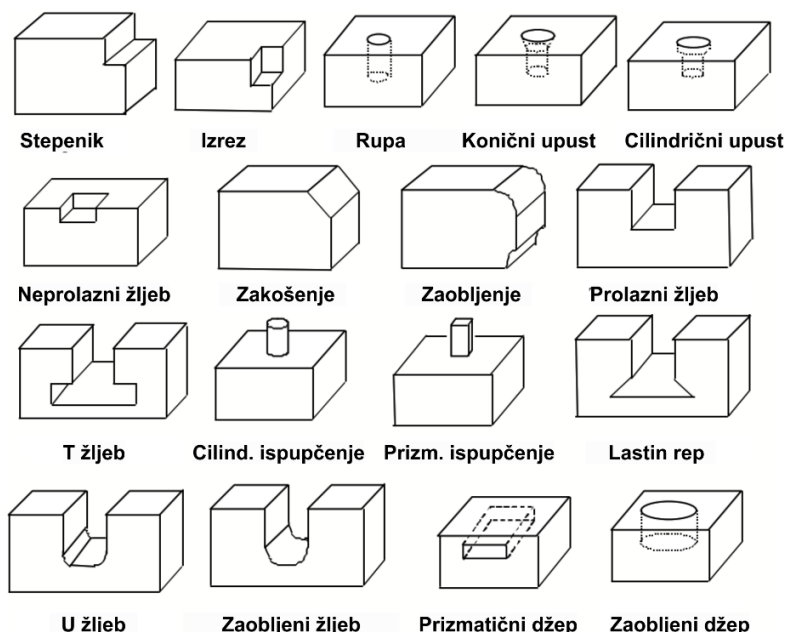
Upravo varijantnost tehnoloških procesa predstavlja jednu od osnovnih osobina tehnoloških procesa obrade proizvoda koja se javlja u gotovo svim fazama projektovanja. Na varijantnost tehnoloških procesa pre svega uticaj imaju vrsta priprema, vrsta i redosled operacija i zahvata obrade, proizvodni resursi, kao i tehnoeekonomski efekti [3,5]. Usvajanjem ovih varijanti vrši se dekompozicija tehnološkog procesa u cilju optimizacije, gde je spoljašnja optimizacija usmerena na izbor optimalnih elemenata tehnološkog procesa na makro nivou, dok je unutrašnja optimizacija usmerena na izbor optimalnih elemenata tehnološkog procesa na mikro nivou [3,6,7].

Imajući u vidu višestruku varijantnost tehnoloških procesa zadatak koji se nameće je na koji način optimizovati tehnološki proces, a pri tom pridržavati se kriterijuma smanjenja vremena i troškova obrade, odnosno sa druge strane, kako ostvariti neophodan nivo proizvodnosti i ekonomičnosti. Broj varijanti tehnoloških rešenja znatno zavisi od raspoloživosti mašina alatki, reznih alata i pribora, vrste zahvata za obradu tipskih oblika na mašinskim delovima, varijantnosti broja alata kao i smerova prilaza alata (SPA) tokom obrade. Takođe, varijantnosti se mogu definisati i u pogledu strategija obrade određenih zahvata, režima obrade, primene sredstava za hlađenje i podmazivanje, itd.

Uzimajući u obzir važnost varijantnosti tehnoloških procesa u nauci se javila potreba za uvođenjem tzv. problema optimizacije tehnoloških procesa (OTP). Ovaj problem je poslednjih nekoliko decenija privukao veliku pažnju naučnika i inženjera koji prvenstveno pripadaju oblastima mašinstva i informatike. Razlog leži u činjenici da varijantnost tehnoloških procesa utiče na dimenzije ovog problema što ga čini vrlo izazovnim optimizacionim zadatkom. Rešavanje ovakvog zadatka zahteva primenu efikasnih algoritama za generisanje optimalnih ili blisko optimalnih rezultata.

Problem optimizacije tehnološkog procesa podrazumeva dva osnovna zadatka koji se sprovode simultano sa ciljem dobijanja optimalnog rešenja. To su određivanje redosleda izvođenja zahvata obrade sa jedne strane i selekcija tehnoloških resursa sa druge [12,20,39,40].

Proizvodi, odnosno mašinski delovi čiji se tehnološki procesi obrade optimizuju generalno su sastavljeni od tzv. „*feature-a*“. Kako bi se definisali zadaci optimizacije u više detalja neophodno je definisati šta se podrazumeva pod pojmom „*feature*“. U domaćem rečniku ovaj pojam može se sresti pod različitim terminima, kao npr. karakteristika, osobina, svojstvo, obeležje, oblik, forma, značajka, itd. U nastavku istraživanja usvojen je termin tipski oblik, odnosno tipski tehnološki oblik. Prema jednoj definiciji „*feature*“ predstavlja semantičku grupu ili atom modeliranja, određen skupom parametara, koji se koristi za opis objekta koji se ne može dodatno razložiti, sa aspekta jedne ili više aktivnosti u vezi sa projektovanjem i primenom proizvoda [41]. U okviru problema optimizacije tehnoloških procesa razmatraju se tipski tehnološki oblici za mašinsku obradu (eng. *machining feature*). Sa tehnološke tačke gledišta, ovi tipski oblici predstavljaju zapreminu materijala koja se uklanja sa obratka u toku izvođenja tehnološkog procesa. Primeri ovih tipskih oblika mogu se uočiti na raznim trodimenzionalnim solid modelima u literaturi među kojima su neki grafički predstavljeni na slici 2.2.



Slika 2.2. Primeri tipskih tehnoloških oblika za mašinsku obradu [8]

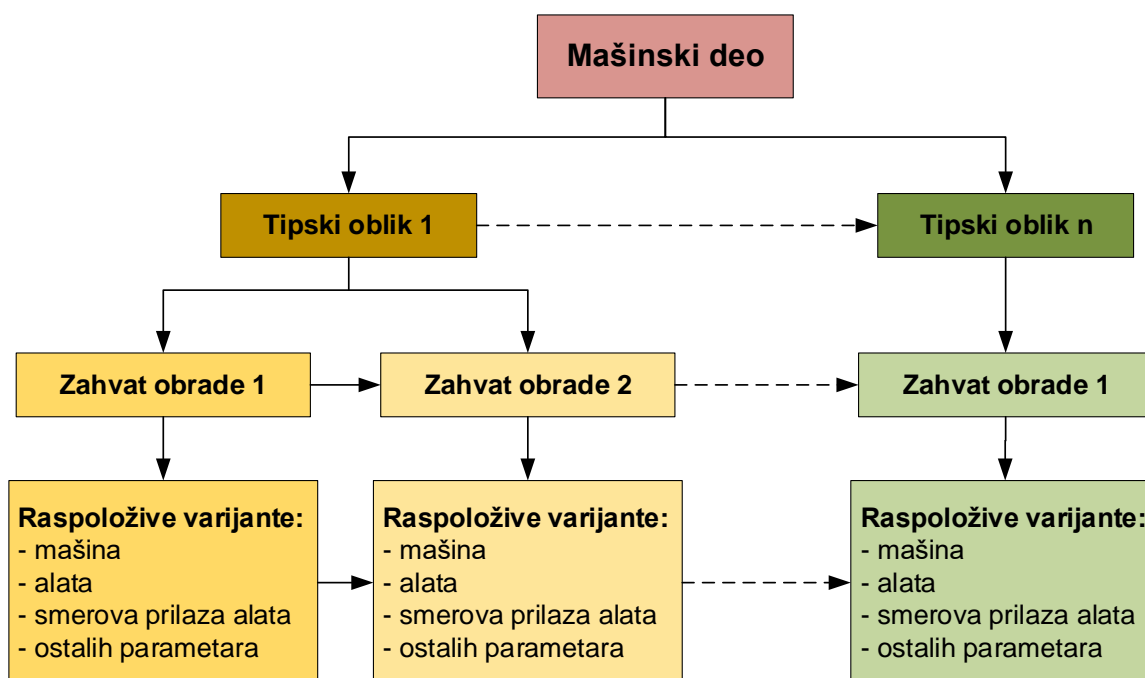
U skladu sa prethodnim informacijama, prvi zadatak optimizacije tehnoloških procesa može se podeliti u dve faze [8]:

- **Selekcija zahvata obrade** na bazi geometrije tipskih oblika i tehnoloških zahteva u pogledu njihove obrade. Ova faza podrazumeva selekciju jednog zahvata iz skupa potencijalnih varijanti, odnosno alternativa zahvata za svaki pojedinačni tipski oblik koji je definisan dimenzijama, tolerancijama i kvalitetom obrađene površine. Neki tipski oblici zahtevaju veći broj zahvata. Tako se npr. za obradu otvora određenog prečnika, koji ne poseduje zahteve u pogledu tolerancija i kvaliteta površine, bušenje otvora može usvojiti kao odgovarajuća varijanta zahvata obrade. Sa druge strane, ukoliko se za isti tipski oblik zahteva bolji kvalitet obrađene površine, tada se za njegovu obradu može usvojiti jedna od kombinacija zahvata (bušenje → proširivanje → razvrtanje) ili (bušenje → unutrašnje struganje).
- **Selekcija tehnoloških resursa** koja se odnosi na odabir odgovarajuće varijante mašine alatke, reznog alata i smera prilaza alata prilikom obrade. Na osnovu prethodno rečenog, može se zaključiti da je varijantnost tehnoloških procesa izrade određenog dela, posledica varijantnosti korišćenih tehnoloških resursa. Mašine alatke koje se najčešće razmatraju kod problema optimizacije tehnoloških procesa su pre svega konvencionalne mašine kao što su horizontalne i vertikalne glodalice, bušilice, brusilice, zatim CNC mašine alatke, obradni centri i druge. Pored mašina alatki i reznih alata neophodno je odrediti optimalnu varijantu smera prilaza, odnosno orijentacije alata (u nastavku SPA) koja direktno utiče na plan stezanja, a samim tim i na broj podoperacija obrade.

Drugi zadatak optimizacije tehnoloških procesa čini određivanje optimalnog redosleda zahvata obrade za koje se simultano selektuju optimalne varijante tehnoloških

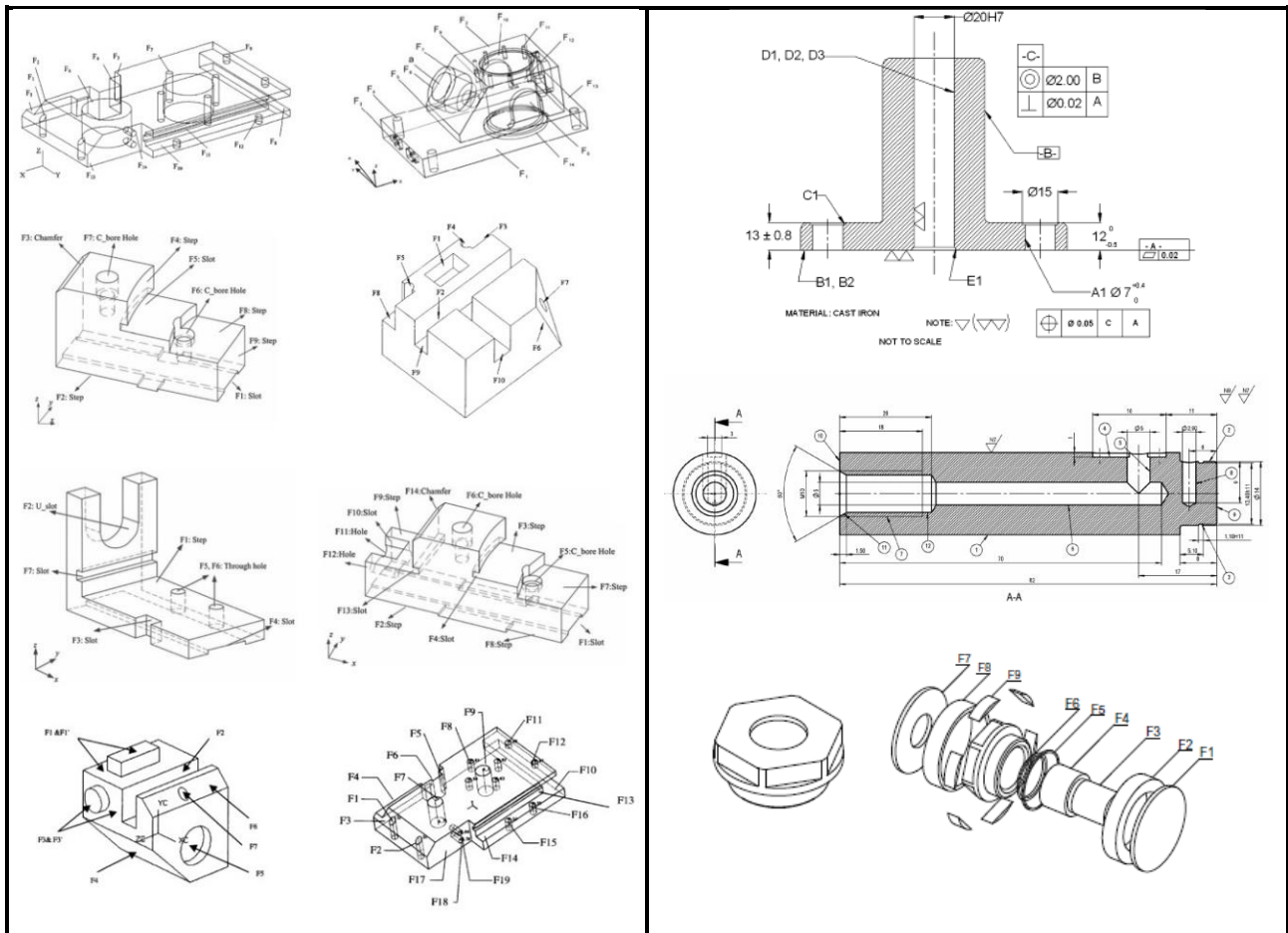
resursa u vidu mašine alatke, reznog alata i SPA. Cilj je ostvariti optimalan tehnološki proces u užem smislu koji poštuje uslove prethođenja zahvata, odnosno prioritete obrade o kojima će u nastavku poglavlja biti više reči. Pored ovih komponenata tehnološkog procesa takođe se mogu generisati odgovarajući planovi stezanja grupisanjem onih zahvata koji se realizuju na istoj mašini alatki koristeći isti SPA. Skup ovih aktivnosti, odnosno zadataka i njihovih faza u optimizaciji tehnološkog procesa prema nekim autorima može se definisati kao planiranje tehnološkog procesa [16].

Na bazi osnovnih zadataka optimizacije tehnoloških procesa, grafički se može predstaviti struktura jednog tehnološkog procesa obrade koja naglašava pomenute varijantnosti u pogledu zahvata odnosno tehnoloških resursa. Tehnološki proces strukturiran na ovakav način dat je na slici 2.3 [42,43]. U nastavku, umesto termina „mašine alatke“ i „rezni alati“, korišćeni su njihovi skraćeni oblici – „mašine“ i „alati“ dok je smer prilaza alata adresiran pomoću akronima SPA.



Slika 2.3 Struktura varijantnih tehnoloških procesa za jedan mašinski deo

Kada su u pitanju mašinski delovi koji se razmatraju u optimizaciji tehnoloških procesa, u literaturnim izvorima su češće analizirani delovi prizmatičnog nego rotacionog oblika [2]. Geometrija delova predstavlja se u vidu trodimenzionalnih solid modela, dok su za neke primere dati i tehnički crteži [12,19]. Na slici 2.4 prikazani su trodimenzionalni modeli i dvodimenzionalni crteži pojedinih reprezentativnih primera prizmatičnih i rotacionih delova koji se često susreću u eksperimentalnim studijama [2, 11, 12, 39, 44-46]. Ovi primeri predstavljaju osnovne modele za testiranje i verifikaciju performansi modifikovanih i hibridnih metoda koje se primenjuju za pronalaženje optimalnih tehnoloških procesa.

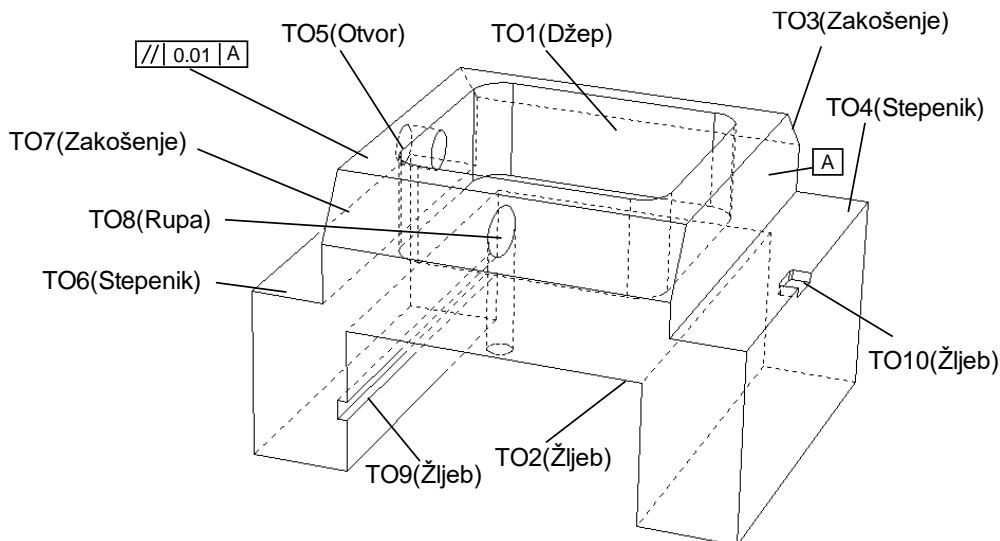


a) Prizmatični delovi

b) Rotacioni delovi

Slika 2.4. Trodimenzionalni modeli/Dvodimenzionalni crteži reprezentativnih proizvoda kao predmeta optimizacije tehnoloških procesa [19,46]

Na slici 2.5 prikazan je jedan primer trodimenzionalnog solid modela prizmatičnog dela koji sadrži deset tipskih oblika. Na osnovu ovog modela može se utvrditi koji zahvati i rezni alati se mogu upotrebiti za obradu prikazanih tipskih tehnoloških oblika. Recimo, za prvi tipski oblik TO1 koji predstavlja džep, optimalna varijanta alata predstavljala bi vretenasto glodalo odgovarajuće dužine i prečnika za obradu džepa na zahtevanoj dubini.

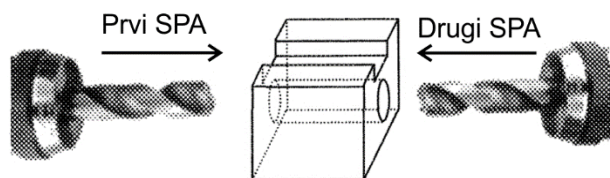


Slika 2.5 Solid model prizmatičnog dela sa tipskim oblicima [8]

Isti model dela sa slike 2.5 može se iskoristiti za određivanje izvodljivih SPA za svaki razmatrani alat. SPA se može definisati kao pravac, odnosno smer iz kojeg alat ostvaruje kontakt sa tipskim tehnološkim oblikom u cilju realizacije posmatranog zahvata obrade [16]. Takođe, vrlo je bitno napomenuti da svaka promena SPA može da utiče na promenu stezanja, odnosno pripreme steznog pribora (eng. *setup*). Stezanje dela podrazumeva postavljanje dela u položaj u kojem se omogućava obrada određene grupe zahvata u jednoj pripremi, odnosno jednoj konfiguraciji steznog pribora na mašini [11,16]. Uslovi koje svaki SPA treba da ispunjava su mogućnost nesmetanog pristupa delu, raspoloživost steznih pribora i različitih alata, kao i zahtevi u pogledu tolerancija i kvaliteta obrađene površine [11]. Većina ovih uslova pripada domenu tzv. uslova prethođenja zahvata o kojima je više rečeno u nastavku ovog poglavlja.

Kod prizmatičnih delova najčešće se koristi šest mogućih smerova alata prema tri ose, i to $\pm x$, $\pm y$ i $\pm z$ koji u literaturi nose naziv teorijski SPA [8]. Međutim, nedavna istraživanja razmatraju proizvode, odnosno mašinske delove koji sadrže veći broj varijanti SPA [12,47]. Pored njih, treba uzeti u obzir i ometanja do kojih može doći prilikom pristupanja alata obratku. Uzrok ovim ometanjima u ovom slučaju predstavlja geometrija alata i predmeta obrade. Autori su u radu [48] predstavili pristup kojim se vrši provera ovih konflikata putem simulacije kretanja alata u solid režimu. Tokom ove simulacije prati se kretanje alata duž svih teorijskih smerova prilaza za razmatrani zahvat obrade i svi SPA koji izazivaju ometanja prilikom obrade direktno se uklanjaju iz skupa varijanti.

Može se izvesti zaključak da se načini izvođenja zahvata odbacuju ukoliko ne postoji izvodljiva varijanta SPA za realizovanje razmatranog zahvata. Ako se uzme u obzir primer obrade otvora TO5 sa slike 2.5 raspoložive varijante teorijskih SPA su „+x“ i „-x“. Međutim, usled geometrije mašinskog dela, kao izvodljivo rešenje nameće se jedino prilazna putanja alata duž „+x“ ose. Jednostavan primer mašinskog dela sa dva izvodljiva SPA dat je na slici 2.6.



Slika 2.6. Obrada otvora omogućena u dva različita SPA [11]

U skladu sa prethodno iznetim, fokus istraživanja u okviru ove doktorske disertacije je usmeren na optimizaciju tehnoloških procesa obrade putem određivanja optimalnih varijanti zahvata obrade, kao i optimalnih varijanti tehnoloških resursa, neophodnih za realizaciju definisanih zahvata. Pored toga, simultano se vrši pronalaženje optimalnog redosleda izvođenja zahvata obrade što u celini predstavlja jedan optimalan tehnološki proces u užem smislu, odnosno plan tehnološkog procesa. Ovde je važno napomenuti da se u prvom zadatku optimizacije selekcija zahvata obrade vrši samo u određenim slučajevima kada se koriste I/ILI mreže, dok se kod drugih pristupa vrši isključivo određivanje redosleda zahvata i dodeljivanje optimalnih varijanti mašina, alata i SPA što direktno zavisi od tehnoloških uslova koji su prethodno definisani.

Sa ciljem odgovora na savremene zahteve i adaptiranje na česte promene u proizvodnim okruženjima, a sa akcentom na višestrukoj varijantnosti u okviru projektovanja tehnoloških procesa, uveden je pojam fleksibilnosti tehnoloških procesa [49]. Razlikuje se više tipova ovih fleksibilnosti koje se smatraju ključnim za efikasnost proizvodnje i samog proizvodnog sistema. To se prvenstveno ogleda u izborima odgovarajućih zahvata obrade, mašina, alata, SPA i konačno, redosleda zahvata obrade. Shodno tome, izdvajaju se sledeći tipovi fleksibilnosti tehnoloških procesa sa aspekta njihove optimizacije [40,50]:

- **Fleksibilnost zahvata obrade** – mogućnost obrade istog tipskog tehnološkog oblika primenom različitih zahvata obrade ili različitih redosleda zahvata obrade;
- **Fleksibilnost mašina** – mogućnost realizacije istog zahvata na različitim mašinama;
- **Fleksibilnost alata** – mogućnost realizacije istog zahvata korišćenjem različitih alata;
- **Fleksibilnost smerova prilaza alata** – mogućnost realizacije istog zahvata obrade pomoću različitih SPA, odnosno različitih orijentacija alata i tipova stezanja dela;
- **Fleksibilnost redosleda zahvata obrade** – mogućnost izmene redosleda zahvata u skladu sa postavljenim ograničenjima prethođenja.

Pri razvoju savremenih sistema za računom podržano projektovanje tehnoloških procesa, kao i njihove integracije sa drugim funkcijama i aktivnostima proizvodnog sistema i globalnog poslovnog okruženja, primenjuju se brojne metode i tehnike, koje se mogu koristiti zasebno ili integralno [5,16,51-56]. Među njima se mogu istaći one koje su pronašle najveću primenu:

- *Metode zasnovane na tipskim oblicima;*
- *Metode veštačke inteligencije;*
- *Agent-bazirane metode;*
- *Internet-bazirane metode;*
- *Metode bazirane na STEP standardu, i dr.*

Veštačka inteligencija, kao naučna disciplina, odnosi se na razvoj inteligentnih računarskih sistema, koji imaju karakteristike povezane sa inteligentnim ponašanjem čoveka, kao što su razumevanje, učenje, razmišljanje i rešavanje problema [53]. U oblasti projektovanja i optimizacije tehnoloških procesa, odnosno razvoju CAPP sistema, najveću primenu od metoda ili tehnika veštačke inteligencije u poslednje vreme pronašli su ekspertni sistemi, neuronske mreže [57], genetski algoritmi i fazi logika [52].

Od prethodno nabrojanih metoda veštačke inteligencije, u ovom istraživanju su posebno izdvojeni genetski algoritmi, zbog činjenice da su pronašli široku primenu u inženjerskoj optimizaciji i time otvorili vrata kasnijoj primeni brojnih biološki inspirisanih tehnika optimizacije. Radi se o širokoj oblasti veštačke inteligencije koja razmatra čitav spektar inteligentnih metaheurističkih algoritama, ili kraće, metaheuristika. Ove metode se u literaturi dodatno modifikuju sa ciljem poboljšanja njihovih performansi kako bi se generisala kvalitetna rešenja za razne kompleksne optimizacione probleme. Takva svojstva čine ih neizostavnim alatom pri rešavanju problema OTP kao važnim segmentom inteligentnih tehnoloških sistema.

S obzirom da koncept inteligentnog tehnološkog sistema integriše primenu veštačke inteligencije i računarom integrisane proizvodnje, njegova izgradnja je nemoguća bez uključivanja funkcije optimizacije tehnoloških procesa [2]. Kako je rečeno, optimalni redosledi zahvata obrade, sa selektovanim optimalnim varijantama tehnoloških resursa koji su u primarnom fokusu u optimizaciji tehnoloških procesa, mogu se usvojiti kao ulazni podaci za narednu fazu optimizacije planova terminiranja tehnoloških procesa [46,58-62]. Terminiranje tehnoloških procesa bavi se planiranjem mašinske obrade u odgovarajućem proizvodnom okruženju. U okviru ove funkcije vrši se vremensko raspoređivanje zahvata obrade i omogućava njihovo raspoređivanje na raspoložive mašine u odgovarajućem proizvodnom okruženju [2,63]. Veliku pažnju naučne zajednice privukla je integracija ove dve aktivnosti na bazi koje je nastao problem tzv. integracije projektovanja i terminiranja tehnoloških procesa. U cilju smanjenja troškova i vremena proizvodnje i povećanja iskorišćenosti tehnoloških resursa ovaj integrisani model karakteriše činjenica da se projektovanje (optimizacija) i terminiranje tehnoloških procesa obrade vrše simultano. Pri tome ne dolazi do konflikata između kriterijuma optimizacije i omogućeno je efektivno adaptiranje na dinamičke promene u proizvodnom okruženju. Cilj istraživanja u ovoj disertaciji usmerava se ka prvoj aktivnosti ove integracije, optimizaciji tehnoloških procesa.

2.2 Uslovi i ograničenja prethođenja u optimizaciji tehnoloških procesa obrade

Uslovi prethođenja među zahvatima predstavljaju važan segment optimizacije tehnološkog procesa iz razloga što od njih direktno zavisi da li tehnološki proces predstavlja dopustivo (izvodljivo) rešenje ili ne. Ovi uslovi, ili drugim rečima ograničenja, moraju biti identifikovana kako bi se pravovremeno izvršila provera izabranog ili slučajno generisanog redosleda zahvata obrade. U literaturi se ističu dve osnovne grupe uslova prethođenja [8]:

- 1) Uslovi prethođenja zahvata u okviru obrade tipskih tehnoloških oblika, gde spadaju:
 - *Uslovi stezanja,*
 - *Zavisnost izbora baznih površina,*
 - *Međusobni odnos tipskih oblika,*
 - *Uslovi tehnološkičnosti obrade,*
 - *Uslovi efikasnosti obrade.*
- 2) Uslovi prethođenja zahvata u okviru skupa različitih tipskih tehnoloških zahvata.

Za svaki dobijeni skup zahvata obrade razmatranog tipskog tehnološkog oblika postoje uslovi prethođenja, kao što npr. ograničenje da zahvati grube obrade prethode zahvatima fine obrade (npr. bušenje se realizuje pre razvrtanja).

Pravila prethođenja za izvođenje zahvata obrade mogu se grupisati i na osnovu tehničkih i ekonomskih ograničenja. Shodno tome, definisane su sledeće grupe tehničkih i ekonomskih ograničenja, odnosno prioriteta [64]: dimenziono prethođenje, geometrijsko prethođenje, tehnološko prethođenje i ekonomsko prethođenje.

2.2.1 Dimenziono prethođenje

Svaka površina je geometrijski definisana preko kota pomoću kojih su utvrđene dimenzije površine, ali i odnosi prema drugim površinama na delu. Utvrđivanje prvenstva između zahvata obrade zbog dimenzionog razloga je vezano za kotiranje površina i može se definisati preko sledećeg pravila: "Pre neke površine obrađuje se površina u odnosu na koju je posmatrana površina dimenzionisana, a prioritet imaju površine dimenzionisane u odnosu na bazu". Na ovaj način ostvaruje se poklapanje tehnološke baze sa konstrukcionom bazom, često i sa mernom i montažnom bazom. U slučaju da iz nekih razloga ne može da se primeni ovo pravilo neophodno je izvršiti transfer dimenzija, što često može voditi do smanjenja ostvarenih tolerancija [19].

2.2.2 Geometrijsko prethođenje

Utvrđivanje prvenstva zahvata zbog geometrijskog razloga vezano je za zahteve u pogledu međusobnog odnosa površina i osa, odnosno tolerancije položaja. Prvenstva se

u ovom slučaju mogu definisati u obliku sledećeg pravila: "Površine, u odnosu na koje su definisane tolerancije položaja neke površine, imaju prioritet pri obradi". Tačnost međusobnog odnosa površina-tolerancija položaja može se ostvariti primenom sledećih metoda, pri čemu ostvarena tačnost opada od prve ka poslednjoj metodi [19]:

- **Primena iste baze za obradu u istoj operaciji.** Visoka tačnost međusobnog položaja se ostvaruje ako se ove površine obrađuju u istom pozicioniranju i stezanju. Na tačnost obrade utiče samo tačnost mašine i njenog podešavanja.
- **Primena iste baze za obradu u različitoj operaciji.** Kod ove metode pored tačnosti mašine i njenog podešavanja utiču i greške pozicioniranja i stezanja.
- **Primena jedne površine kao baze za obradu druge površine, odnosno pozicioniranje i stezanje preko jedne površine a obrada druge površine.** Takođe i kod ove metode na tačnost utiče tačnost mašine i njenog podešavanja, kao i greške pozicioniranja i stezanja.
- **Korišćenje različitih baza za obradu ili pak uvođenje dopunske baze za obradu.** Dopunska baza za pozicioniranje je površina specijalno uspostavljena-dodata na delu radi pozicioniranja pri obradi. Ova površina je dodatno napravljena površina koja nije specificirana crtežom ili postojeća površina na delu čija je tačnost obrade povećana kako bi zadovoljila uslove da bude tehnološka baza. Ova metoda se koristi kada ni jedna od prethodnih metoda nije izvodljiva.

2.2.3 Tehnološko prethođenje

Redosled obrade treba da bude tehnološki. Ova grupa ograničenja se najčešće odnosi na pravila redosleda zahvata obrade tipskih oblika koji zahtevaju veći broj zahvata. U tabeli 2.1 dat je primer ovih pravila za obradu otvora i rupa prečnika od 10 do 30 mm različitih kvaliteta iz punog priprema i prethodno odlivenog ili dobijenog kovanjem.

Tabela 2.1. Pravila tehnološkog prethođenja obrade otvora i rupa (za mere 10 do 30 mm) [65]

Broj i redosled zahvata pri obradi otvora i rupa				
Nazivni prečnik D (mm)	Prethodno stanje priprema	Zahtevana tačnost otvora ili rupe		
		IT 7	IT 8, IT 9	IT10 – IT 13
10 do 30	Izrada otvora iz punog priprema	1. Zabušivanje 2. Bušenje 3. Proširivanje 4. Grubo razvrtanje 5. Fino razvrtanje	1. Zabušivanje 2. Bušenje 3. Proširivanje 4. Razvrtanje	1. Zabušivanje 2. Bušenje 3. Proširivanje
	Obrada prethodno odlivenog ili otkovanog otvora	1. Proširivanje 2. Grubo razvrtanje 3. Fino razvrtanje	1. Proširivanje 2. Razvrtanje	1. Proširivanje

2.2.4 Ekonomsko prethođenje

U cilju ostvarenja tehnoeekonomskih efekata, pre svega manjih troškova i vremena obrade, neophodno je uzeti u obzir i pravila ekonomskog prethođenja. Ovo se vrlo često odnosi na primenu produktivnije ili ekonomičnije metode/alata umesto nekog prethodnog lošijeg rešenja [19].

2.2.5 Ograničenja prethođenja

Pored varijantnosti, odnosno fleksibilnosti tehnološkog procesa koje su definisane u poglavlju 2.1, na složenost problema optimizacije tehnološkog procesa uticaj imaju i navedeni geometrijski i tehnološki zahtevi i interakcije među tipskim oblicima. Ovi uslovi, odnosno pravila prethođenja formiraju tzv. **ograničenja prethođenja** (eng. *precedence constraints*) među zahvatima, koja direktno uvećavaju dimenzije problema optimizacije tehnoloških procesa. Prema autorima [11,18] ova ograničenja mogu se podeliti na tvrda (eng. *hard*) i meka (eng. *soft*). Tehnološki proces mora biti dosledan tvrdim ograničenjima, jer ona direktno utiču na izvodljivost same obrade, dok sa druge strane meka ograničenja utiču na kvalitet, troškove ili efikasnost procesa i stoga mogu u nekim slučajevima biti narušena kako bi se ostvarili minimalni troškovi obrade. Ograničenja prethođenja često se formulišu u vidu veza, odnosno interakcija do kojih dolazi prilikom procesa obrade. Otuda, mogu se izdvojiti sledeća tvrda ograničenja [11]:

- **Interakcija pribora** (eng. *Fixture interaction*) – Prilikom obrade tipskih oblika pozicioniranje i stezanje ima vrlo bitnu ulogu jer u slučaju pogrešno postavljenih prioriteta zahvata može se onemogućiti stezanje samog obratka za obradu svih tipskih oblika;
- **Interakcija alata** (eng. *Tool interaction*) – Prioriteti među tipskim oblicima moraju se odrediti kako bi se ostvarilo pravilno pozicioniranje i kretanje alata pri obradi jer bi u suprotnom ono takođe bilo nemoguće ili bi došlo do kolizije sa mašinom ili priborom;
- **Interakcija tehnološke baze** (eng. *Datum interaction*) – Kako bi se obradak tokom obrade pravilno orijentisao neophodno je definisati tehnološku bazu koja ima ulogu referentne površine prilikom obrade i prioritet u odnosu na druge površine;
- **Prioriteti obrade tipskih oblika** (eng. *Feature priorities*) – Kod ovog ograničenja postoje dva slučaja. Prvi je kada primarni tipski oblik treba obraditi pre drugih vezanih tipskih oblika, dok je kod drugog slučaja neophodno prvobitno obraditi tipski oblik koji će obezbediti prilaz obradi drugih tipskih oblika;
- **Fiksni redosled zahvata obrade** (eng. *Fixed order of machining operations*) – Pod ovim se smatraju posebna ograničenja koja određuju redosled zahvata pri obradi tipskih oblika, npr. prioritet pri obradi grube u odnosu na finu obradu.

Kod mekih ograničenja razlikuju se dva tipa [11]:

- **Tankozidna interakcija** (eng. *Thin-wall interaction*) – Ova interakcija se javlja u slučajevima kad je veoma malo rastojanje između tipskih oblika i neophodno je obezbediti dobru proizvodnu praksu određivanjem njihovih prioriteta;

- **Interakcija skidanja materijala** (eng. *Material removal interaction*) – Ova interakcija se javlja ukoliko, na primer za dva tipska oblika sa geometrijskim interakcijama, redosled skidanja materijala utiče na troškove ili kvalitet obrade i uzrokuje prioritizovanje zahvata, npr. u slučaju kada se prvo realizuje obrada veće zapremine, a potom manje.

Meka i tvrda ograničenja su, u okviru ovog predmetnog istraživanja usvojena u drugom eksperimentu koji je opisan u poglavlju 6.1.2. Kratak opis navedenih ograničenja sa ilustrovanim primerima dat je u tabelama 2.2 i 2.3.

Tabela 2.2. Tvrda ograničenja [11,18]

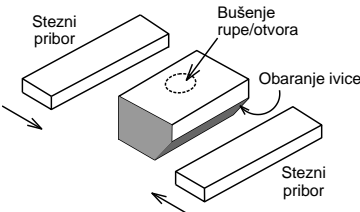
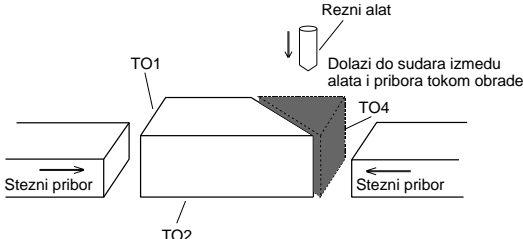
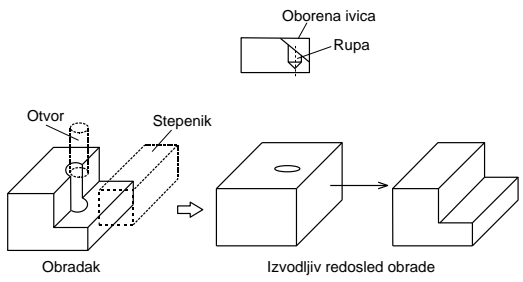
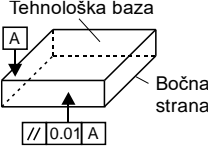
Naziv ograničenja	Ilustrovani primer	Kratko obrazloženje
Interakcija pribora		<p>Identična je uslovu stezanja iz prve grupe uslova prethođenja; realizacija jednog zahvata obrade utiče na potpuni ili delimični gubitak tehnoloških baza neophodnih za obradu drugih tipskih oblika. Na slici se obaranje ivice mora realizovati nakon bušenja rupe/otvora.</p>
Interakcija alata i pribora		<p>Do ove interakcije dolazi kada postoji mogućnost sudara između reznog alata i steznog pribora tokom obrade rezanjem.</p>
Interakcija alata i tipskog oblika		<p>Identična je uslovu tehnologičnosti iz prve grupe uslova prethođenja; Pogrešan izbor redosleda zahvata u ovim slučajevima može voditi deformaciji ili lomu reznog alata (npr. burgije). Na gornjem primeru vidi se da obaranje ivice mora uslediti nakon bušenja rupe. Isti je slučaj sa primerom na donjoj slici gde takođe bušenje rupe ima prioritet u odnosu na obradu stepenika.</p>
Interakcija tehnološke baze		<p>Identična je uslovu zavisnosti baznih površina iz prve grupe uslova prethođenja; Tehnološke baze uvek imaju prioritet obrade u odnosu na ostale površine.</p>

Tabela 2.2. (nastavak). Tvrda ograničenja [11,18]

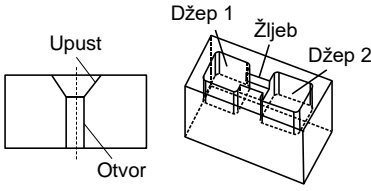
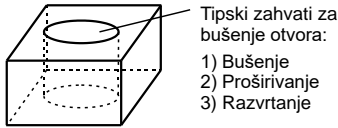
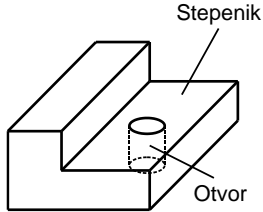
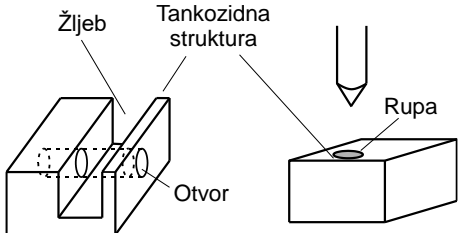
Naziv ograničenja	Ilustrovani primer	Kratko obrazloženje
Međusobni odnos tipskih oblika		Na prvoj slici jasno se vidi da bušenje rupe/otvora ima prioritet u odnosu na upuštanje; kod drugog primera džepovi se prvobitno moraju obraditi kako bi se obezbedile ulazne površine za obradu žljeba.
Fiksni redosled zahvata obrade		Identičan drugoj grupi uslova prethođenja. Fiksni redosled zahvata podrazumeva da se radi o prioritetima koji se tiču grube i fine obrade, a bitno utiču na poboljšanje kvaliteta obrađene površine.

Tabela 2.3. Meka ograničenja [11,18]

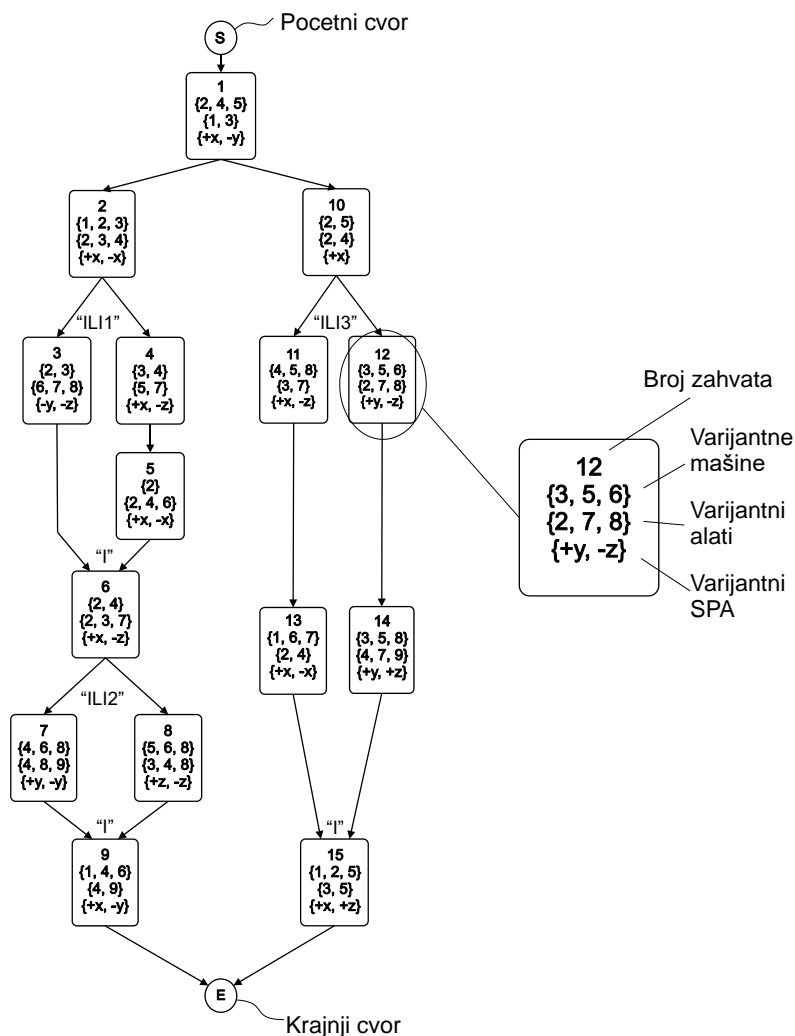
Naziv ograničenja	Primer	Kratko obrazloženje
Efikasnost i ekonomičnost mašinske obrade		Pripada uslovu efikasnosti obrade iz prve grupe uslova prethođenja; Efikasnost po pitanju troškova i vremena može imati vrlo značajan uticaj na redosled zahvata obrade. Prema primeru sa slike, obrada stepenika glodanjem se realizuje brže u poređenju sa bušenjem otvora pa je samim tim vreme obrade kraće, kao i bolji kvalitet obrađene površine.
Tankozidna interakcija		Tankozidne strukture mogu uzrokovati mnogo komplikacija tokom procesa obrade. Kod primera na slici levo, najbolje rešenje je da bušenje otvora prethodi izradi žljeba jer bi u suprotnom bušenje otvora nakon glodanja žljeba moglo izazvati lom tankog zida. Na desnoj slici vidi se slučaj bušenja rupe na poziciji koja je vrlo blizu ivice prizmatičnog dela što predstavlja slučaj koji se uglavnom izbegava.

2.2.6 Predstavljanje varijantnih tehnoloških procesa obrade

Nakon definisanja fleksibilnosti tehnoloških procesa sa jedne strane, i pravila prethođenja koja formiraju različite tipove ograničenja prethođenja sa druge strane, korak koji sledi je određivanje načina na koji se ovi tipovi fleksibilnosti, odnosno ograničenja prethođenja mogu predstaviti. U naučnoj literaturi mogu se pronaći različiti načini predstavljanja tehnoloških procesa, a među prvima se izdvajaju Petrijeve mreže [66].

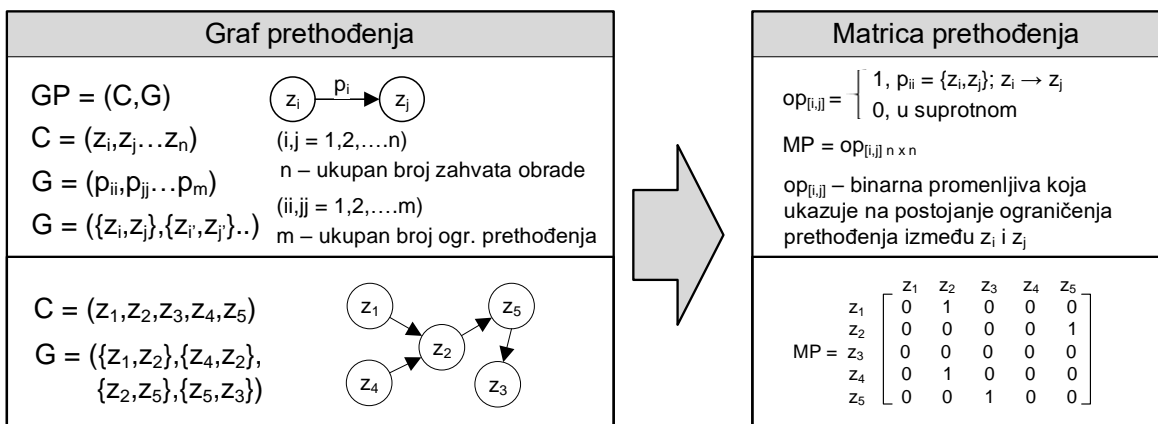
Pored njih veliku primenu imaju mrežni grafovi, odnosno I/IL mreže [2,20,67,68]. Jedan primer ovakve mreže nalazi se na slici 2.7 [20] koji prikazuje sve tipove fleksibilnosti tehnološkog procesa: fleksibilnost zahvata obrade, mašina, alata, SPA kao i fleksibilnost redosleda zahvata.

I/IL mreža se sastoji iz tri različita čvora, od kojih početni i krajnji čvor ukazuju na početak i završetak procesa obrade dela, dok se središnji čvorovi odnose na zahvate obrade, pri čemu su za svaki zahvat dati skupovi raspoloživih varijanti mašina, alata i SPA. Kako je prikazano na slici 2.7, strelice koje spajaju čvorove označavaju prioritete odnose koji su definisani u poglavlju 2.2.6. Fleksibilnost procesa je opisana pomoću „IL“ konektora kojim se mogu odabrati različiti redosledi zahvata za obradu datog dela. „IL“ veza je ona veza koja započinje „IL“ konektorom i završava se „I“ konektorom kojim se spaja sa drugim vezama u mreži. Na slici 2.7 se može videti da čvor koji označava zahvat pod brojem 2 ima označen „IL1“ konektor kojim se omogućava selekcija zahvata 3 ili redosleda zahvata 4→5 kao sledbenika koji se završavaju „I“ konektorom u čvoru, a koji označava zahvat broj 6. Procedura selektovanja različitih zahvata identična je za ostale čvorove nakon kojih sledi „IL“ konektor [69].



Slika 2.7. I/IL Mreža fleksibilnih tehnoloških procesa obrade [20]

Sa aspekta pravila i ograničenja prethođenja, najveću primenu za predstavljanje varijantnosti tehnoloških procesa imaju matrice i grafovi prethođenja (GP) [12,47,70]. GP je usmereni graf koji se sastoji iz čvorova i grana, $GP=(C,G)$, kod kojeg čvorovi C predstavljaju skup zahvata, dok grane G označavaju veze između čvorova, odnosno zahvata. Svaki element u okviru vektora redosleda zahvata preslikava se u odgovarajući čvor GP-a. S druge strane, svaki odnos prethođenja predstavlja se pomoću grana. Čvorovi koji nisu povezani granama nemaju međusobnih odnosa prethođenja. Radi lakšeg manipulisanja podacima koji se tiču odnosa prethođenja koristi se matrica susedstva, odnosno matrica prethođenja koja se nalazi na slici 2.8. Prikazani su glavni elementi GP-ova i njihovo mapiranje u matrični oblik. Koristeći jednostavan primer sa pet zahvata i četiri ograničenja prethođenja, definisana je matrica prethođenja MP. Vrednost $op[i, j]$ je binarna promenljiva koja definiše prethođenje između zahvata i i zahvata j . Kao što je prikazano, vrednost p koja reprezentuje prethođenje može biti broj 1 koji označava da odnos prethođenja između posmatranih zahvata postoji, dok, sa druge strane, broj 0 ukazuje na nedostatak prethođenja. Nedostatak odnosa prethođenja ilustruje se nepostojanjem grana između zahvata i i zahvata j . Prateći grane GP-a potrebno je pokriti sve čvorove kako bi se dobio izvodljiv redosled zahvata.



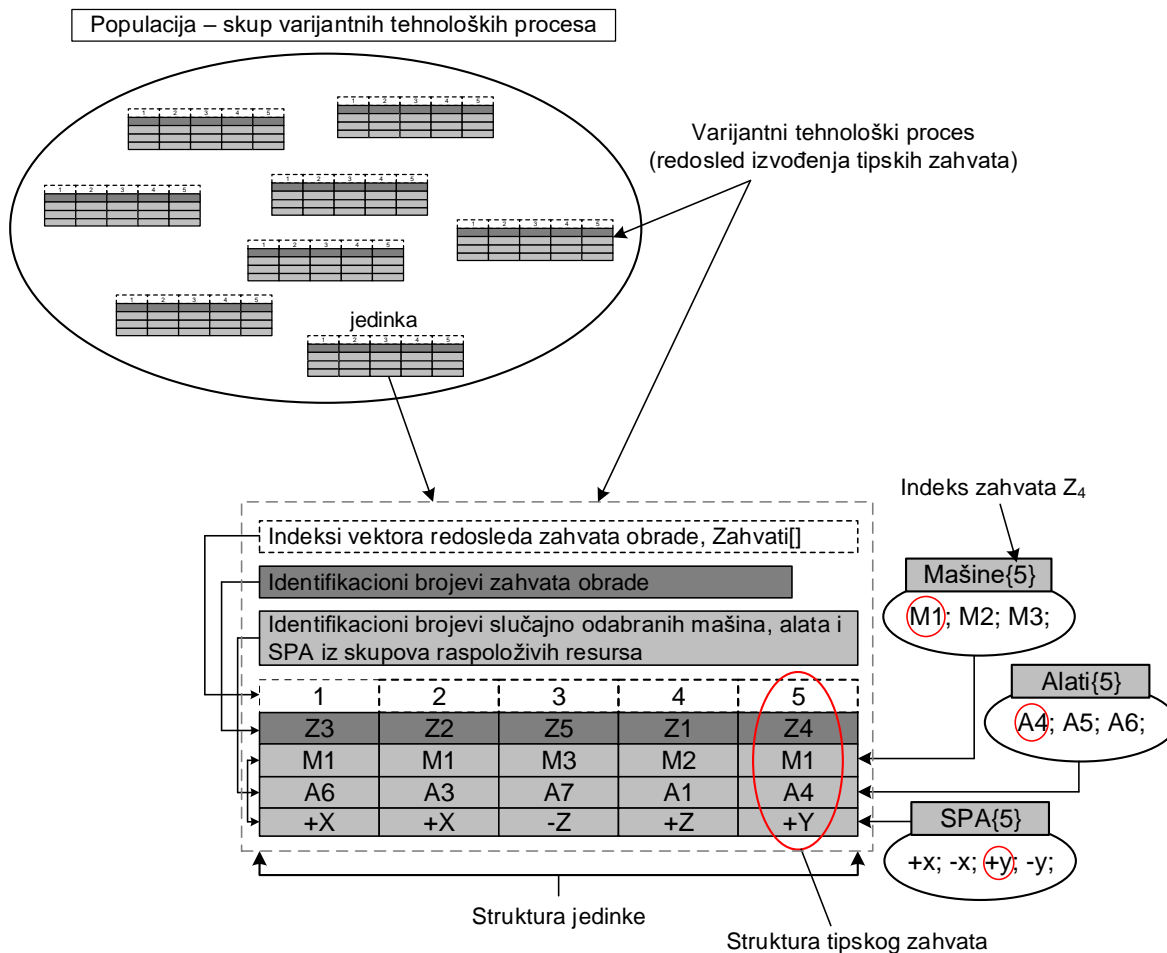
Slika 2.8. Graf i matrica prethođenja zahvata obrade

Tehnološki proces obrade se u užem smislu sastoji od određenog broja selektovanih zahvata obrade potrebnih za obradu tipskih tehnoloških oblika, zatim tehnoloških resursa kao što su mašine, alati i SPA kao i planovi stezanja. Budući da raspoloživost i mogućnosti tehnoloških resursa igraju važnu ulogu pri generisanju niza zahvata, u obzir se mora uzeti nekoliko pravila [71]:

- *Plan stezanja uključuje grupu zahvata koji se mogu obraditi primenom istih SPA.*
- *Prethodna grupa zahvata se mora obrađivati na istoj mašini.*
- *Jedan od ciljeva je smanjenje promena mašina, alata i SPA.*
- *Samo jedan zahvat može se obrađivati istovremeno na jednoj mašini.*
- *Samo jedan zahvat može se obrađivati primenom jednog alata.*
- *Samo jedan SPA se može koristiti u datom trenutku.*

Imajući u vidu da se optimizacija tehnoloških procesa vrši primenom odgovarajućih metaheurističkih algoritama, o kojima će više biti rečeno u poglavlju 3, implementacija

ovih metoda zahteva poznavanje i rad u određenim programskim okruženjima. U okviru ovog istraživanja primenjen je Matlab, programski jezik četvrte generacije. Informacije o zahvatima, tehnološkim resursima i ograničenjima koje se predstavljaju na grafovima i u matričnom obliku moraju formirati određeni domen promenljivih veličina koje će biti manipulirane u okviru programskog jezika. Jedan od načina, da se zahvati obrade opišu u skladu sa prethodnom rečenim, je predstavljanje redosleda zahvata kao vektora od n bitova. Ovi bitovi su celobrojne vrednosti gde svaka vrednost sadrži relevantne informacije o zahvatima i tehnološkim resursima. S obzirom da se zadatak optimizacije tehnoloških procesa odnosi na simultano određivanje redosleda zahvata i selekciju tehnoloških resursa, koriste se četiri različita vektora za predstavljanje tehnološkog procesa. Na slici 2.9 dat je jednostavan primer vektora tehnološkog procesa sa ukupno 5 zahvata obrade i indeksima koji se kreću od 1 do 5 (prvi red). Vektor redosleda zahvata od 5 bitova označen je tamnom bojom i nosi naziv $Zahvati[i]$ pri čemu i podrazumeva indeks odgovarajućeg zahvata obrade ($i = 1 \dots n$, gde je n je ukupan broj zahvata). Na primer, vektor $Zahvati[5]$ podrazumeva da se radi o zahvatu Z4. Pod istim indeksom nalaze se mašina M1 alat A4 i smer prilaza alata +y. Ovi kandidati su redom elementi vektora mašina, $Mašine[]$, vektora alata, $Alati[]$ i vektora SPA, $SPA[]$. Vektori su vizuelno predstavljeni svetlo sivom bojom na slici 2.9. Strukturu jednog tipskog zahvata sačinjavaju selektovani zahvat obrade i odabrane varijante mašine, alata i SPA.



Slika 2.9. Vektorsko predstavljanje redosleda zahvata

Tabela 2.4 prikazuje osnovne tipove podataka koji su upotrebljeni u Matlab programskom okruženju za razvoj izvornog koda i predstavljanje problema optimizacije tehnološkog procesa. Prve četiri promenljive, $Zahvati[i]$, $Mašine[i]$, $Alati[i]$ i $SPA[i]$ odnose se na redoslede koji se sastoje iz n elemenata koji redom reprezentuju zahvate obrade, zatim varijantne mašine, alate i SPA. Naredne tri promenljive $Mašine\{\}$, $Alati\{\}$ i $SPA\{\}$ predstavljaju baze sastavljene iz nizova i ćelija koje sadrže sve raspoložive varijante mašina, alata i SPA za svaki zahvat obrade. Poslednja promenljiva, $Vremena\{\}$, čini bazu vremena obrade koja odgovaraju odabranoj varijanti tipskog zahvata.

Tabela 2.4. Tipovi podataka

Tipovi podataka	Promenljive	Opis
double	$Zahvati[i]$	Indeksirani broj zahvata obrade i kao dela vektora redosleda zahvata obrade
double	$Mašine[i]$	Indeksirani broj mašine i kao elementa vektora mašina u okviru kojeg se na n mašina realizuje n zahvata
double	$Alati[i]$	Indeksirani broj reznog alata i kao elementa vektora alata u okviru kojeg se n alata koristi za realizaciju n zahvata
double	$SPA[i]$	Indeksirani broj SPA i kao elementa vektora SPA u okviru kojeg se n SPA koristi za realizovanje n zahvata
cell	$Mašine\{\}$	Niz od n kandidata mašina u okviru kojeg n ćelija sadrži alternativne mašine alatke za realizaciju n zahvata obrade
cell	$Alati\{\}$	Niz od n kandidata alata u okviru kojeg n ćelija sadrži alternativne alate za realizaciju n zahvata obrade
cell	$SPA\{\}$	Niz od n kandidata SPA u okviru kojeg n ćelija sadrži alternativne SPA za realizaciju n zahvata obrade
cell	$Vremena\{\}$	Niz od n vremena obrade u okviru kojeg n ćelija sadrži vremena obrade koja odgovaraju datoj varijanti tipskog zahvata

2.3 Matematički model za optimizaciju tehnoloških procesa obrade

Osnovni kriterijumi vrednovanja tehnoloških procesa obrade, odnosno redosleda tipskih zahvata obrade najčešće su minimizacija ukupnih troškova i minimizacija ukupnog vremena obrade razmatranog mašinskog dela. Kada je u pitanju prvi kriterijum optimizacije, ukupne troškove obrade određenog dela čine troškovi korišćenja i promene odgovarajućih resursa i to: troškovi rada mašine, troškovi korišćenja alata, troškovi promene mašine, troškovi promene alata i troškovi promene stezanja, odnosno pozicioniranja dela [10,12,17,20,21,39,40,43-45,49,71-75]. U nastavku je prikazan matematički oblik ovih troškova.

- 1) **Ukupni troškovi rada mašina** predstavljaju zbir ukupnih troškova rada svih mašina koje se koriste u okviru tehnološkog procesa. Izračunavaju se pomoću sledećeg izraza:

$$TMC = \sum_{i=1}^n MCI_{j,i} = Zahvat[i].id_masine \quad (1)$$

gde su:

n – ukupan broj zahvata obrade,

$MCI_{j,i}$ – troškovi rada mašine alatke j prilikom realizovanja zahvata i (konstantna vrednost za svaku mašinu alatku)

$Zahvat[i].id_masine$ - indeksirani broj mašine alatke za realizovanje zahvata i u okviru vektora $Zahvat[]$.

- 2) **Ukupni troškovi korišćenja alata** predstavljaju zbir ukupnih troškova alata upotrebljenih u okviru tehnološkog procesa. Izraz za izračunavanje je sledeći:

$$TTC = \sum_{i=1}^n TCI_{j,i} = Zahvat[i].id_alata \quad (2)$$

gde su:

$TCI_{j,i}$ – troškovi korišćenja alata j prilikom realizovanja zahvata i (konstantna vrednost za svaki alat)

$Zahvat[i].id_alata$ - indeksirani broj alata za realizovanje zahvata i u okviru vektora $Zahvat[]$.

- 3) **Ukupni troškovi promena mašina** i ukupan broj promena mašina su važni faktori koje razmatraju promene mašina koje se javljaju u slučajevima kada se dva uzastopna zahvata obrade u okviru vektora $Zahvat[]$ realizuju na različitim mašinama. Ukupan broj promena mašina u tehnološkom procesu se može izračunati pomoću uslova koji je formulisan na sledeći način:

$$NMC = \sum_{i=1}^{n-1} C_1(Zahvat[i].id_masine, Zahvat[i+1].id_masine) \quad (3)$$

$$C_1(X, Y) = \begin{cases} 1, & \text{ako je } X \neq Y \\ 0, & \text{ako je } X = Y \end{cases} \quad (4)$$

$$TMCC = \sum_{i=1}^{NMC} MCCI_i \quad (5)$$

gde su:

$Zahvat[i+1].id_masine$ - indeksirani broj mašine za realizovanje zahvata $i+1$ u okviru vektora $Zahvat[]$,

$MCCI_i$ – troškovi promene mašina koji predstavljaju konstantnu vrednost bez obzira na vrstu i tip mašine.

- 4) **Ukupni troškovi promena alata** i ukupan broj promena alata jednako su značajni faktori kao i prethodni, a odnose se na promene alata u slučajevima kada se dva uzastopna zahvata obrade u okviru vektora $Zahvat[]$ realizuju upotrebom različitih alata. U tabeli 2.5 prikazani su slučajevi na bazi kojih se definišu promene alata.

Ukupan broj promena alata u tehnološkom procesu se može izračunati pomoću dva uslova, koji su predstavljeni sledećim izrazima:

$$NTC = \sum_{i=1}^{n-1} C_2(C_1(\text{Zahvat}[i].id_masine, \text{Zahvat}[i+1].id_masine), C_1(\text{Zahvat}[i].id_alata, \text{Zahvat}[i+1].id_alata)) \quad (6)$$

$$C_2(X, Y) = \begin{cases} 0, & X = Y = 0 \\ 1, & u \text{ suprotnom} \end{cases} \quad (7)$$

$$TTCC = \sum_{i=1}^{NTC} TCCI_i \quad (8)$$

gde su:

$\text{Zahvat}[i+1].id_alata$ - identifikacioni broj alata za realizovanje zahvata $i+1$ u okviru vektora $\text{Zahvat} []$,

$TCCI_i$ – troškovi promene alata koji predstavljaju konstantnu vrednost bez obzira na vrstu i tip alata.

Tabela 2.5. Definicija promene alata

Uslovi realizovanja dva uzastopna zahvata obrade	Promena alata
Isti alat i ista mašina	Ne
Isti alat i različite mašine	Da
Različiti alati i ista mašina	Da
Različiti alati i različite mašine	Da

- 5) **Ukupni troškovi promene stezanja dela** su troškovi za koje su ključni faktori ukupan broj stezanja (NS) i broj promena stezanja (NSC) u okviru tehnološkog procesa. Do promene stezanja dolazi u slučajevima kada je dva susedna zahvata potrebno realizovati u različitoj pripremi mašine, odnosno primenom različitih SPA. Tabela 2.6 daje prikaz definisanja promene stezanja dela. Ukupan broj promena stezanja dela se određuje sličnim postupkom kao ukupan broj promena alata:

$$NSC = \sum_{i=1}^{n-1} C_2(C_1(\text{Zahvat}[i].id_masine, \text{Zahvat}[i+1].id_masine), C_1(\text{Zahvat}[i].id_SPA, \text{Zahvat}[i+1].id_SPA)) \quad (9)$$

$$C_2(X, Y) = \begin{cases} 0, & X = Y = 0 \\ 1, & u \text{ suprotnom} \end{cases} \quad (10)$$

$$NS = NSC + 1 \quad (11)$$

$$TSC = \sum_{i=1}^{NS} SCI_i \quad (12)$$

gde su:

$Zahvat[i].id_SPA$ - identifikacioni broj SPA za realizovanje zahvata i u okviru vektora $Zahvat []$,

$Zahvat[i + 1].id_SPA$ - identifikacioni broj SPA za realizovanje zahvata $i+1$ u okviru vektora $Zahvat []$.

SCI_i - troškovi promene stezanja dela koji se usvajaju kao konstantna vrednost bez obzira na vrstu i tip stezanja.

Prema jednačini (11), ukupan broj promena stezanja predstavlja zbir broja promena stezanja i broja jedan koji podrazumeva inicijalnu pripremu mašine, odnosno početno stezanje i pozicioniranje mašinskog dela.

Tabela 2.6. Definicija promene stezanja dela

Uslovi realizovanja dva uzastopna zahvata obrade	Promena stezanja
Isti SPA i ista mašina	Ne
Isti SPA i različite mašine	Da
Različiti SPA i ista mašina	Da
Različiti SPA i različite mašine	Da

- 6) **Dodatni troškovi** i broj narušenih ograničenja deo su matematičke formulacije ukupnih troškova obrade u studijama kod kojih su ograničenja prethodjenja grupisana u tvrda i meka ograničenja. Iako se smatra da se određena meka ograničenja mogu da narušiti, njihov broj treba da teži minimumu što je omogućeno uvođenjem dodatnih troškova. Ovaj broj se izračunava putem sledećeg izraza:

$$NVC = \sum_{i=1}^{n-1} C_3(Zahvat[i].id_zahvata, Zahvat[i + 1].id_zahvata) \quad (13)$$

$$C_3(X, Y) = \begin{cases} 0, & \text{ako prethodjenje } X \rightarrow Y \text{ narušava meko ograničenje} \\ 1, & \text{ako je prethodjenje } X \rightarrow Y \text{ u skladu sa mekim ograničenjem} \end{cases} \quad (14)$$

$$APC = \sum_1^{NVC} PC \quad (15)$$

gde su:

$Zahvat[i].id_zahvata$ – indeksirani broj zahvata obrade i u okviru vektora $Zahvat []$,
 $Zahvat[i + 1].id_zahvata$ - indeksirani broj zahvata obrade $i+1$ u okviru vektora $Zahvat []$.

PC – dodatni troškovi koji predstavljaju konstantnu vrednost za svako narušeno meko ograničenje.

- 7) **Ukupni troškovi obrade dela** predstavljaju zbir svih prethodno definisanih troškova koji definišu kriterijum za optimizaciju tehnološkog procesa obrade. Izračunavaju se pomoću sledećeg matematičkog izraza:

$$TWMC = w_1 \cdot TMC + w_2 \cdot TTC + w_3 \cdot TMCC + w_4 \cdot TTCC + w_5 \cdot TSC + w_6 \cdot APC \quad (16)$$

gde su:

$w_1 - w_6$ – težinski koeficijent koji omogućavaju primenu ovih troškova u različitim studijama slučajeva.

Kod ukupnih težinskih troškova obrade dela važno je napomenuti da su dodatni troškovi uključeni u jednačinu (16) za one studije koje razmatraju probleme optimizacije tehnoloških procesa na bazi tvrdih i mekih ograničenja (drugi eksperiment u poglavlju 6). U ostalim slučajevima usvaja se sledeći oblik:

$$TWMC = w_1 \cdot TMC + w_2 \cdot TTC + w_3 \cdot TMCC + w_4 \cdot TTCC + w_5 \cdot TSC \quad (17)$$

- 8) **Funkcija cilja** se formira u skladu sa definisanim troškovima, odnosno ukupnim težinskim troškovima obrade dela. Njen oblik je definisan sledećim izrazom:

$$f_c = \max\left(\frac{1}{TWMC}\right) \quad (18)$$

Izvorni kod matematičkog modela za kriterijum minimizacije troškova obrade nalazi se u prilogu 1.

Drugi kriterijum za optimizaciju tehnološkog procesa je minimiziranje ukupnog vremena obrade. Ovaj kriterijum baziran je na sličnom principu kao prethodni, a čine ga sledeća vremena: glavno vreme obrade, vreme transporta i pomoćna vremena koja podrazumevaju vreme promene alata i vreme promene pribora/stezanja dela [2,49,67,76-80]. U narednim izrazima ovi troškovi su detaljno opisani.

- 1) **Glavno vreme obrade** podrazumeva ukupno vreme neophodno za realizaciju svih zahvata obrade na odgovarajućem delu. Određuju se pomoću sledećeg izraza:

$$TMT = \sum_{i=1}^n MTI_{i,j,k} = \text{Zahvat}[i].id_masine \quad (19)$$

gde su:

n – ukupan broj zahvata obrade,

$MTI_{i,j,k}$ – vreme realizacije zahvata i na mašini j primenom alata k (usvojena je konstantna vrednost),

$\text{Zahvat}[i].id_masine$ - indeksirani broj mašine za realizovanje zahvata i u okviru vektora $\text{Zahvat}[]$.

- 2) **Ukupno vreme transporta** predstavlja vreme neophodno za transport delova u okviru tehnološkog sistema koji, drugim rečima, podrazumeva ukupno vreme promene mašina i određuje se prema izrazu:

$$TT = \sum_{i=1}^{n-1} TTI((i, j_1), (i + 1, j_2)) \quad (20)$$

gde su:

n – ukupan broj zahvata obrade,

$TTI((i, j_1), (i + 1, j_2))$ – vreme transporta dela između mašina j_1 i j_2 na kojima se realizuju dva uzastopna zahvata obrade i i $i+1$ (usvojena je konstantna vrednost).

- 3) **Ukupno vreme promene alata** uključuje ukupan broj promena alata kao važan faktor kao i u slučaju određivanja ukupnih troškova promene alata. Ovaj broj promena alata izračunava se preko identičnog izraza koji je dat u jednačini (6). Uslovi koji moraju biti ispunjeni prilikom optimizacije ovog broja dati su u jednačinama (4) i (7). Na kraju, ukupno vreme promene alata dobija se na sledeći način:

$$TTCT = \sum_{i=1}^{NTC} TCTI_i \quad (21)$$

gde je:

$TCTI_i$ - vreme promene alata koji predstavljaju konstantnu vrednost bez obzira na vrstu i tip reznog alata.

- 4) **Ukupno vreme promena stezanja** dela kao i troškovi promene stezanja dela uključuje važne komponente: broj stezanja obratka (NS) i broj promena stezanja (NSC) u okviru tehnološkog procesa. U skladu sa time, za određivanje ova dva broja koriste se izrazi koji su prethodno definisani u jednačinama (9-11). Ukupno vreme promene stezanja dobija se sledećim izrazom:

$$TSCT = \sum_{i=1}^{NS} SCTI_i \quad (22)$$

gde je:

$SCTI_i$ - vreme promene stezanja dela koje predstavlja konstantnu vrednost bez obzira na vrstu i tip stezanja.

- 5) **Ukupno težinsko vreme obrade** dela čini zbir prethodno definisanih vremena i predstavljaju drugi kriterijum za optimizaciju tehnoloških procesa obrade. Dobijaju se pomoću sledećeg izraza:

$$TWMT = w_1 \cdot TMT + w_2 \cdot TT + w_3 \cdot TTCT + w_4 \cdot TSCT \quad (23)$$

gde su:

$w_1 - w_4$ – težinski koeficijent koji omogućavaju korišćenje određenih troškova za različite tehnološke uslove.

- 6) **Funkcija cilja** se formira u skladu sa definisanim vremenima kao i u prethodnom slučaju. Funkcija se maksimizira, a njen matematički oblik dat je sledećim izrazom:

$$f_t = \max\left(\frac{1}{TWMT}\right) \quad (24)$$

3. METAHEURISTIČKI ALGORITMI

3.1 Osnovni pojmovi

Prema autorima [25], u pogledu kvaliteta rešenja optimizacione metode mogu se podeliti u dve velike grupe, egzaktne (precizne) i aproksimativne (približne) metode. Egzaktne metode omogućavaju precizno pronalaženje rešenja i pri tom garantuju da je to rešenje optimalno. Sa druge strane, aproksimativne metode pronalaze dovoljno dobra rešenja u zadovoljavajućem vremenskom periodu, ali pri tom ne mogu da garantuju njihovu optimalnost. Pažnja je u ovoj disertaciji posvećena aproksimativnim metodama kojima pored aproksimativnih algoritama pripadaju heuristički i metaheuristički algoritmi.

Sama reč „*heuristika*“ potiče iz grčkog jezika čije je značenje „saznati“, „pronaći“, „otkriti“, odnosno „vršiti ispitivanje“ [81]. *Russel i Norvig* [82] u svom obimnom istraživanju o veštačkoj inteligenciji heuristike definišu kao „tehnike koje traže dobra (približno optimalna) rešenja u zadovoljavajućem vremenu za rad i koje ne garantuju njihovu izvodljivost ili optimalnost, a u mnogim slučajevima čak ne mogu ni tvrditi koliko je određeno izvodljivo rešenje blizu optimalnom“. Heuristike se odlikuju „inteligentnim“ pristupom prilikom pretraživanja prostora mogućih rešenja pri čemu koriste stečena znanja o pronađenim rešenjima i pretragu usmeravaju ka obećavajućim regionima toga prostora. Često se u naučnim i praktičnim okvirima mogu pronaći problemi većih dimenzija kod kojih su prostori mogućih rešenja dovoljno veliki da se jedino heuristički algoritmi mogu primeniti kako bi se došlo do adekvatnih rešenja. Svoju efikasnost su pokazali time što „žrtvuju“ optimalnost rešenja i potpunost pretrage kako bi do takvih rešenja došli u zadovoljavajućem vremenskom periodu [81]. U skladu sa time, ističu se dva glavna pokazatelja [83]:

- **efektivnost** algoritma, koja se odnosi na rad uložen za dobijanje kvalitetnih rešenja,
- **efikasnost** algoritma, koja se odnosi na vreme neophodno za izvršenje svih operacija.

Međutim, ovde je vrlo važno naglasiti da prilikom rešavanja određenog tipa optimizacionog problema svaki heuristički algoritam karakteriše određeni stepen složenosti u pogledu vremena i memorijskog prostora [25,83,84]. Vremenska složenost algoritma povezuje vreme potrebno za izvršavanje svih programskih instrukcija sa kompleksnošću problema kako bi se došlo do rešenja u zadovoljavajućem, razumnom vremenskom periodu. Sa druge strane, prostorna složenost algoritma podrazumeva zauzeće memorijskog prostora prilikom rešavanja datog optimizacionog problema. Ova tzv. asimptotska složenost u računarstvu se u algebarskoj formi označava velikim O [85]. Ovom notacijom moguće je matematičkim putem definisati složenost algoritma, odnosno proceniti vreme pretraživanja koje uveliko raste sa porastom kompleksnosti, odnosno dimenzija optimizacionog problema koji se rešava [25,84].

Talbi [25] je izvršio podelu heurističkih algoritama u dve glavne familije prema tipovima problema na kojima se mogu primeniti. Prva familija su heuristički algoritmi koji se implementiraju za specifične tipove optimizacionih problema, a drugu familiju čine metaheuristički algoritmi ili skraćeno metaheuristike.

Termin „*metaheuristika*“ prvi je upotrebio *Glover* [86] koji je nazivu „heuristika“ dodao prefiks „meta“ koji znači „izvan“, „na višem nivou“. Metaheuristike su algoritmi opšte namene koji se mogu primeniti za rešavanje bilo kog problema optimizacije. One su primenu pronašle u svim oblastima optimizacije pokrivajući širok spektar problema manjih i većih razmera. *Luke* [87] u svojoj publikaciji tvrdi da se metaheuristike primenjuju na probleme tipa „*znam kada vidim*“. One se koriste da pronađu odgovore na probleme o kojima ne posedujemo dovoljno znanja. Ne možemo unapred i sa sigurnošću znati kakvo bi bilo optimalno rešenje pri čemu iscrpno pretraživanje ne uzimamo u obzir zbog obimnosti prostora rešenja koji treba pretražiti. Pored toga, dovoljno je da postoje kandidati rešenja koje možemo testirati u skladu sa prethodno definisanim kriterijumom i na taj način proveriti njihov kvalitet.

Pregledom literature stiče se zaključak da postoje određene prednosti koje metaheuristike čine toliko popularnim u naučnoj zajednici. *Mirjalili i sar.* [88] izdvajaju četiri glavna razloga:

- **Jednostavnost** – Odražava se prvenstveno u pogledu inspiracije koja potiče od raznih fizičkih pojava, ponašanja pojedinih ili kolektivnih organizama, ili odgovarajućih koncepata na bazi evolucije. Naučnici preko jednostavnih izvora inspiracije vrše simulacije, razvijaju nove metaheurističke algoritme, predlažu hibridna rešenja i poboljšanja već razvijenih metaheuristika. Takođe, sve to pruža potporu drugim naučnicima koji ovako jednostavne koncepte mogu implementirati u svojim inženjerskim izazovima.
- **Fleksibilnost** – Odnosi se na mogućnost primene metaheurističkih algoritama na različitim problemima optimizacije. Problemi se na taj način mogu posmatrati kao „crne kutije“ koje u računarstvu predstavljaju sisteme kod kojih su ulazi i izlazi najvažniji. Na taj način, jedan od najvažnijih zadataka prilikom implementacije metaheurističkog algoritma jeste kako predstaviti dati problem.
- U skladu sa svojom stohastičkom prirodom, većina metaheuristika **ne koristi informacije o izvodima funkcija** kako bi pronašla optimalno rešenje. Nasuprot

metodama opadajućeg gradijenta koje koriste izvode prilikom pretraživanja, metaheuristike polaze od slučajnih rešenja što ih čini vrlo pogodnim za rešavanje realnih problema kod kojih se putem izvoda ne može steći korisna informacija.

- **Izbegavanje lokalnih optimuma** – Smatra se karakteristikom koja omogućava metaheurističkim algoritmima da izbegnu stagniranje u lokalnim regionima i time uvećaju globalno pretraživanje. Često se prostor pretraživanja smatra nepoznatim ili veoma kompleksnim sa ogromnim brojem lokalnih optimuma pri čemu se metaheuristički algoritmi pokazuju veoma dobrim tehnikama za uspešno pretraživanje više regiona.

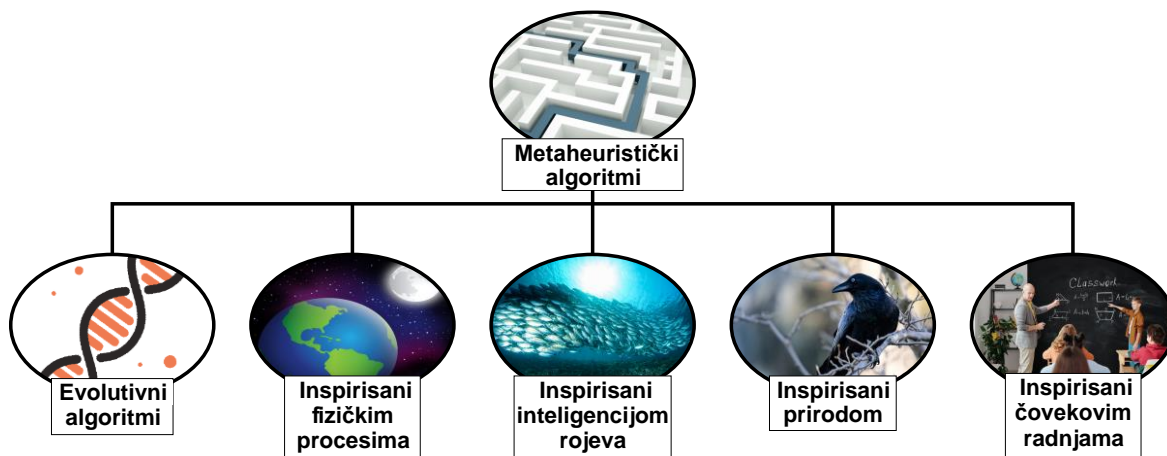
Imajući u vidu već naglašenu kompleksnost optimizacionih problema, metaheuristike se smatraju vrlo pouzdanim tehnikama koje na efikasan način mogu doći do kvalitetnih rešenja u prihvatljivom vremenskom periodu. Zadatak koji se stoga nameće naučnicima jeste odabir praktičnog algoritma koji će doneti uspeh prilikom rešavanja problema od interesa. Veliki broj razvijenih algoritama i onih čiji su razvoji i dalje u toku, čini ovaj zadatak izazovnim u naučnoj zajednici. Glavni razlog zbog kojeg se danas u literaturi može pronaći veliki izbor metaheurističkih algoritama leži u tzv. teoremi „NFL“. Ova teorema koju su predložili *Wolpert* i *Macready* nosi pun naziv „Nema besplatnog ručka“ [89]. Njome se logički potvrđuje da ne postoji algoritam koji može rešiti svaki optimizacioni problem. Pored svih svojih kvaliteta i mogućnosti, superiornost koju algoritam pokazuje za određenu klasu problema može biti dovedena u pitanje ukoliko se pronađe drugi algoritam koji za neku drugu klasu problema pruža znatno bolje performanse. Prema tome, obećavajući rezultati koje algoritam postiže za određeni tip problema optimizacije dolaze na naplatu ako se isti primeni za rešavanje drugih tipova problema [83]. Evidentno je da ova teorema održava oblast razvoja i primene metaheuristika i dalje veoma aktivnom što potvrđuju savremene modifikacije postojećih algoritama i predlozi novih metaheuristika [88].

Sudeći prema prethodno rečenom, ne iznenađuje činjenica što su metaheuristički algoritmi klasifikovani prema različitim kriterijumima i osnovama. Jednu od značajnih klasifikacija izvršio je *Talbi* [25], koji je metaheuristike podelio prema sledećim kriterijumima:

- 1) *algoritmi inspirisani prirodom i algoritmi koji nisu inspirisani prirodom,*
- 2) *algoritmi koji koriste memoriju (mogu da pamte informacije o prethodnim rešenjima, kao npr. algoritam tabu pretrage) i algoritmi bez memorije,*
- 3) *deterministički algoritmi koji donose determinističke odluke (ista početna populacija donosi uvek isto krajnje rešenje) i stohastički algoritmi koji su zasnovani na slučajnosti (ista početna populacija može rezultirati različitim konačnim rešenjima),*
- 4) *algoritmi bazirani na populaciji rešenja (npr. genetski algoritam ili algoritam kolonije mrava) i algoritmi koji rade sa jednim rešenjem (npr. simulirano žarenje),*
- 5) *iterativni algoritmi koji rade sa potpunim početnim rešenjem i „pohlepni“ algoritmi koji polaze od „praznog“ rešenja koje grade sve do kompletnog.*

Pored prethodne podele na bazi većeg broja kriterijuma, u literaturi se može pronaći značajan broj klasifikacija metaheuristika na osnovu izvora inspiracije. Autori *Dhiman* i

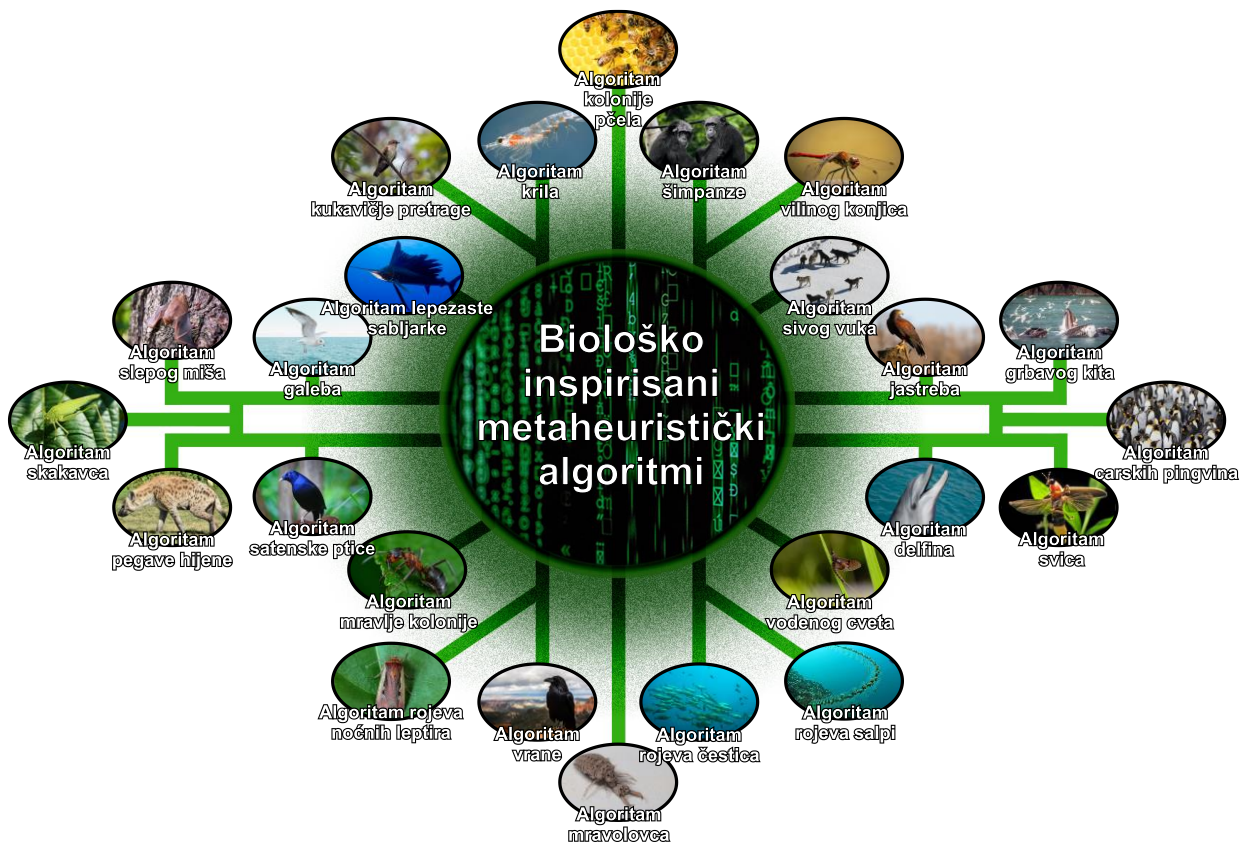
Kumar [90] su prema izvoru inspiracije klasifikovali metaheurističke algoritme u grupu evolutivnih algoritama, algoritama na bazi inteligencije rojeva, algoritama na bazi fizičkih procesa, biološki inspirisanih algoritama i prirodom inspirisanih algoritama. Iako postoje sličnosti između poslednje dve grupe algoritama jer su obe inspirisane inteligencijom živih organizama, algoritmi inspirisani prirodom pokrivaju određene metaheuristike kao što su algoritam hranjenja bakterija [91] ili algoritam veštačkog imunog sistema [92]. *Mirjalili i sar.* [93] naglasak su postavili na evolutivnom računarstvu koje čini podoblast tzv. računarske inteligencije (eng. *Computational intelligence*). Ova oblast pokriva skup inteligentnih sistema kojoj između ostalih pripadaju fazi logika i mašinsko učenje. Kako autori tvrde, kada je u pitanju polje evolutivnog računarstva, sam termin „evolutivni“, pored evolutivnih algoritama se odnosi takođe na ostale biološki, odnosno prirodom inspirisane algoritme. Pored evolutivnih algoritama, posebnu kategoriju čine algoritmi na bazi inteligencije rojeva kao i metaheuristike koje nisu inspirisane biološkom inteligencijom. *Mirjalili i Lewis* [32] su u svom istraživanju o algoritmu inspirisanom grbavim kitovima izvršili klasifikaciju metaheuristika na evolutivne algoritme, algoritme na bazi fizičkih procesa, algoritme na bazi inteligencije rojeva i algoritme inspirisane čovekovim radnjama. U nedavno sprovedenoj studiji autora *Harifi i sar.* [94] predložen je novi izvor drevno-inspirisanih metaheuristika kojoj pripada savremeni algoritam inspirisan postupkom izgradnje Velikih piramida u Gizi. Pored navedenih primera klasifikacije metaheurističkih algoritama, druge podele su u najvećoj meri bazirane na sličnim principima. U skladu sa time, na slici 3.1 dat je ilustrovani prikaz klasifikacije metaheuristika prema izvorima inspiracije koji se najčešće spominju u referentnim radovima i publikacijama.



Slika 3.1. Klasifikacija metaheurističkih algoritama prema izvoru inspiracije

Težište rada u ovoj disertaciji je usmereno na savremene algoritme koji se uglavnom svrstavaju u grupu metaheuristika inspirisanih inteligencijom rojeva i grupu metaheuristika koje su inspirisane prirodom. S obzirom na to da je biološka inteligencija različitih organizama ono što odlikuje algoritme iz obe grupe, u mnogim publikacijama autori se u sopstvenim prilazima većinom opredeljuju za sveobuhvatan termin „metaheuristike inspirisane prirodom“ [93,95,96] odnosno „biološki inspirisani optimizacioni (metaheuristički) algoritmi“ [2,97]. U nastavku istraživanja usvojen je drugi termin.

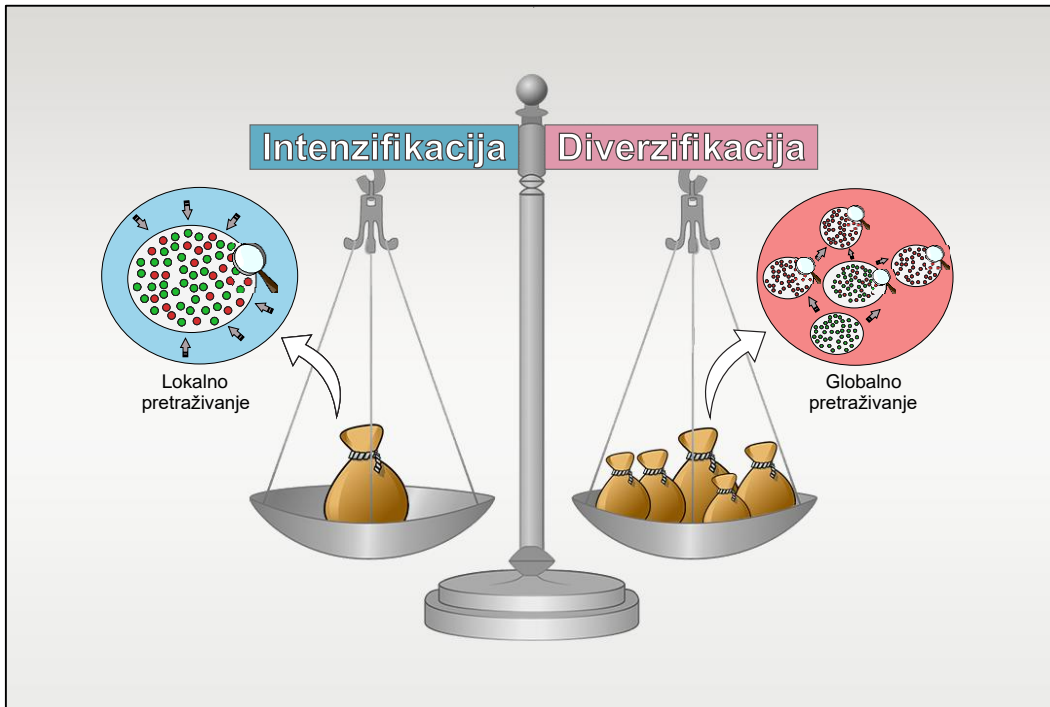
Na slici 3.2 ilustrovani su biološki inspirisani metaheuristički algoritmi zasnovani na inteligentnim mehanizmima koje jedinke koriste zasebno ili kao pripadnici neke veće zajednice. U pitanju su populacioni optimizacioni algoritmi među kojima je veći deo njih pronašao primenu u rešavanju različitih problema optimizacije. Prikazani algoritmi inspirisani su: skakavcima [98], slepim miševima [36], hijenama [90], galebovima [99], satenskim pticama [100], lepezastim sabljarkama [101], mravljim kolonijama [31], pticama kukavicama [102], noćnim leptirima [103], krilovima [104], vranama [105], mrvolovcima [34], pčelinjim kolonijama [106], šimpanzama [107], inteligencijom rojeva [108], vilinim konjicima [33], salpama [109], sivim vukovima [88], vodenim cvetovima [110], jastrebovima [111], ehlokacijom delfina [30], grbavim kitovima [32], svicima [35] i carskim pingvinima [112]. Pored navedenih u literaturi se mogu pronaći i mnoga druga predložena rešenja.



Slika 3.2. Savremeni biološki inspirisani metaheuristički algoritmi [113-118]

U skladu sa istaknutim prednostima metaheuristika kao i njihovom klasifikacijom, vrlo je važno napomenuti dve osnovne karakteristike svakog metaheurističkog algoritma, intenzifikaciju i diverzifikaciju. Sa jedne strane, intenzifikacija podrazumeva intenzivno pretraživanje lokalnih regiona kako bi se pronašla bolja rešenja. Drugim rečima, intenzifikacija ili eksploatacija predstavljaju iskorišćavanje pronađenih rešenja [83]. Sa druge strane, diverzifikacija ili istraživanje podrazumevaju uvođenje stohastičkih operatera koji će obezbediti pretraživanje različitih regiona u prostoru pretrage [119]. Na slici 3.3 može se uočiti ilustrovani prikaz intenzifikacije i diverzifikacije.

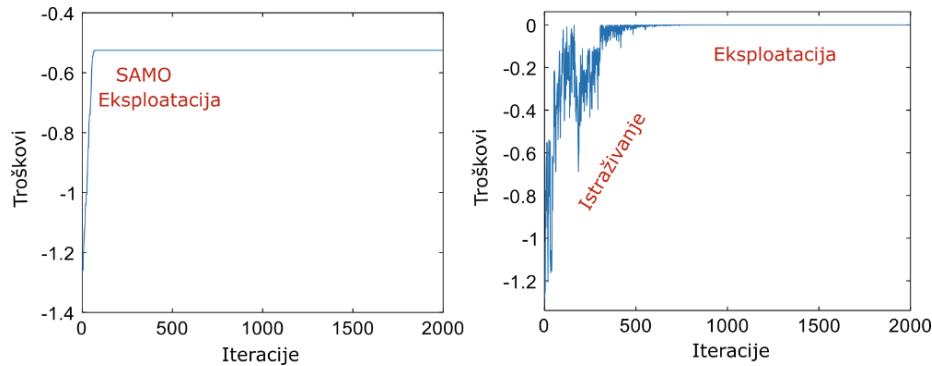
Dok je eksploatacija rešenja usmerena ka lokalnom pretraživanju, diverzifikacija naglasak stavlja na globalno pretraživanje i jedan od glavnih ciljeva naučnika koji implementiraju metaheurističke algoritme je da uspostave balans između ova dva pravca. Na taj način, algoritam bi posedovao kapacitet da, sa jedne strane, temeljno pretraži svaki region i , sa druge strane, usmeri pretraživanje ka što većem broju regiona u okviru prostora pretraživanja. Zbog stohastičke prirode metaheuristika i kompleksnosti optimizacionog problema ovaj zadatak balansiranja predstavlja svojevrsan izazov.



Slika 3.3. Intenzifikacija i diverzifikacija metaheurističkih algoritama

Jedan od načina kako se intenzifikacija i diverzifikacija mogu posmatrati je pomoću krivih konvergencije. Konvergencija predstavlja svojstvo koje nam pruža informaciju o načinu na koji algoritam istražuje prostor pretrage [120]. Drugim rečima, ona ukazuje na približavanje globalnom optimalnom rešenju [121]. *Mirjalili* i *Dong* [122] su u svojoj kratkoj studiji predstavili eksploataciju i istraživanje, odnosno intenzifikaciju i diverzifikaciju na krivama konvergencije za dva slučaja. Ukazali su na pojavu fluktuacija pri usponu rešenja u prostoru pretrage (maksimizacija funkcije cilja). Na slici 3.4 date su krive konvergencije sa fluktuacijama rešenja za algoritam penjanja uzbrdo (slika 3.4a) i algoritam simuliranog žarenja (slika 3.4b). Može se zapaziti da za prvi slučaj nema fluktuacija na krivoj što implicira da algoritam penjanja uzbrdo konvergira ka lokalnim rešenjima u inicijalnim fazama procesa pretraživanja. To je ujedno karakterističan slučaj za ovu metaheuristiku koja je usmerena ka lokalnom pretraživanju jer potiče od činjenice da se pretraživanje uvek odvija „penjanjem uzbrdo“. Ovakav prilaz pretraživanju naglašava eksploataciju, odnosno intenzifikaciju što se smatra poželjnom karakteristikom prilikom pristupanja linearnim problemima optimizacije. Kod drugog slučaja, algoritam simuliranog žarenja pokazuje različit obrazac prema prikazanoj krivi konvergencije. U prvoj fazi pretraživanja vrednost funkcije cilja se naglo menja pri čemu do izražaja dolazi istraživački element,

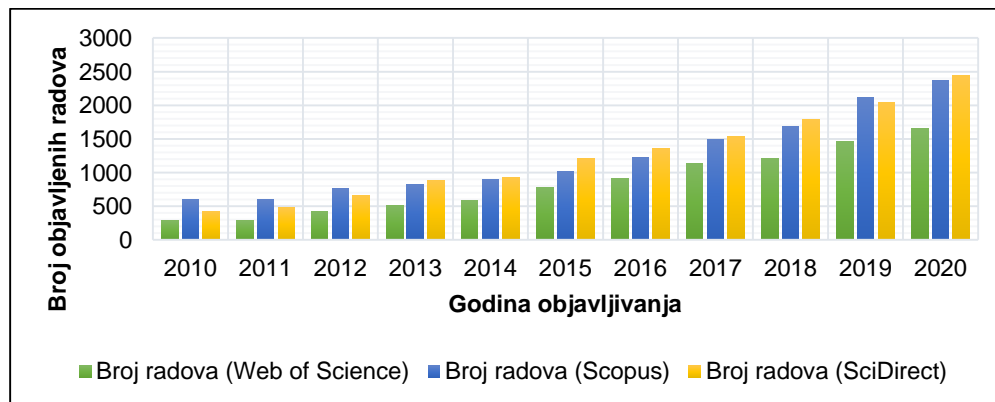
odnosno diverzifikacija. Rezultat toga je što simulirano žarenje poseduje operatore koji omogućavaju preduzimanje „lošijih“ poteza prilikom pretraživanja kako bi sprečio stagnaciju u lokalnom optimumu. Nakon približno šest stotina iteracija iz diverzifikacije se prelazi u fazu intenzivnije pretrage, odnosno eksploatacije. Ovakav slučaj pretraživanja sreće se često pri rešavanju nelinearnih problema optimizacije.



Slika 3.4. Intenzifikacija i diverzifikacija izraženi pomoću krive konvergencije [122]

3.2 Istorijat i razvoj metaheurističkih algoritama

U poređenju sa drugim optimizacionim metodama i tehnikama, metaheuristički algoritmi su ostvarili vrlo široku primenu u rešavanju različitih praktičnih optimizacionih problema. Pored obimnih izvora u kojima se mogu pronaći modifikacije postojećih algoritama, oblast razvoja i primene metaheurističkih algoritama svedoči pojavi novih algoritamskih rešenja, gotovo na mesečnom nivou. Jedan od pokazatelja koji potvrđuje njihovu popularnost je broj objavljenih radova iz relevantnih naučnih baza. Na slici 3.5 može se videti dijagram koji pokazuje broj naučnih radova i publikacija objavljenih u prethodnih deset godina. Naglasak je postavljen na izvorima koji u naslovu, rezimeu ili ključnim rečima sadrže reč „metaheuristic“ pri čemu su korišćene tri vodeće naučne baze *Web of Science*, *Scopus* i *Science Direct*. Imajući u vidu da se ne nazire stagnacija u razvoju ovog polja veštačke inteligencije, očekuje se da će broj objavljenih radova u ovoj oblasti nastaviti eksponencijalno da raste.



Slika 3.5. Broj objavljenih radova u prethodnih deset godina koji sadrže ključnu reč „metaheuristic“

Ovaj deo poglavlja ima za cilj da pruži kratak osvrt na istorijat razvoja metaheuristika sa akcentom na najpopularnije algoritme, njihove autore i autore koji su izvršili značajne pomake po pitanju njihovih modifikacija i primene. Počevši od Alana Tjuringa koji se smatra ocem današnje informatike, ne može se sa sigurnošću tvrditi kada su primenjene prve metaheurističke metode. Među prvim slučajevima najviše pažnje naučne javnosti privukla je familija optimizacionih tehnika koja pokriva četiri istorijske paradigme nastale na bazi Darvinove prirodne selekcije i genetike organizama [123]. Radi se o jednoj od najstarijih grupa optimizacionih algoritama kojoj pripadaju evolucione strategije, evoluciono programiranje, genetsko programiranje i genetski algoritam. Osnovne razlike između njih leže u proceduri predstavljanja jedinki (rešenja) kao i operacijama selekcije i reprodukcije novih jedinki. Najpopularniji među njima je genetski algoritam koji je 1975. godine razvio *Holland* [124], a kasnije utemeljio njegovu primenu *Goldberg* [125]. Genetski algoritam baziran je na populaciji jedinki koje se nazivaju hromozomima i podrazumevaju rešenja koja su predstavljena u vidu vektora ili stringa određene dužine. Pomoću odgovarajućih genetskih operatora kao što su kodiranje i dekodiranje jedinki, zatim selekcije, ukrštanja i mutacije, populacija jedinki kroz određeni broj generacija evoluira i na taj način oponaša postupak genetske selekcije i opstanak najsposobnijih jedinki.

Dalji napredak u razvoju metaheuristika ostvarili su autori *Kirkpatrick S. i sar.* [126]. Oni su 1983. godine razvili algoritam simuliranog žarenja koji je inspirisan postupkom žarenja metala u metalurgiji. Proces je baziran na zagrevanju metala na visoku temperaturu i njegovom laganom hlađenju kojim se uklanjaju defekti kristalne rešetke i postiže stanje minimalne energije.

Glover F. [110] je 1986. godine doprineo pojavljivanju još jednog metaheurističkog algoritma koji se i u današnje vreme uveliko primenjuje. Radi se o algoritmu tabu pretraživanja. Ono što ovu metaheuristiku prvenstveno odlikuje je pamćenje prethodno pronađenih rešenja kojim se utiče na odabir narednih rešenja u prostoru pretraživanja.

Dorigo M. [29,31] takođe pripada grupi pionira u oblasti inteligencije rojeva i autorom sa preko stotinu hiljada citata u naučnoj bazi što ga stavlja u sam vrh oblasti istraživanja. Tom uspehu najviše je doprineo razvoj algoritma inspirisanog kolonijama mrava koji se ubraja u najuspešnije razvijene metaheurističke algoritme sa primenom u mnogim naučnim oblastima u poslednje dve i po decenije. Algoritam kolonije mrava (ACO) inspirisan je ponašanjem mrava prilikom potrage za hranom. Mravi imaju običaj da ostavljanjem tragova u vidu supstance koja se zove feromon pruže informaciju ostalim mravima u koloniji o putanji kojom se kreću. Na taj način omogućena je njihova direktna komunikacija što odražava kolektivnu inteligenciju same kolonije.

Autori *Kennedy J., Eberhart R.C.* kao i *Yuhui Shi* [108,127] pioniri su u razvojnom svetu veštačke inteligencije koji su 1985. godine stekli reputaciju razvojem i daljim usavršavanjem algoritma inspirisanog rojevima čestica. Ovaj metaheuristički algoritam inspirisan je kolektivnim ponašanjem grupe jedinki (ptica, riba ili drugih organizama) u prirodi prilikom kretanja u potrazi za hranom. Prednost pre svega leži u sposobnosti grupe jedinki da kooperativno savlada prepreke koje inače pojedinci samostalno ne bi mogli [2]. Ovi pojedinci u okviru populacije okarakterisani su svojim položajem i brzinom. Naglasak je postavljen na njihovoj međusobnoj komunikaciji pri čemu jedinke ažuriraju informacije

o svojim položajima na bazi sopstvenog iskustva kao i na bazi iskustva celokupne populacije. Sa tako stečenim znanjima omogućeno je populaciji, odnosno jatu ili roju, da evoluira i pri tom pronađe najkvalitetnija rešenja.

Novi vek sa sobom je doveo pojavu novih metaheuristika. Početkom 2000-ih godina, najviše pažnje privlače algoritmi inspirisani pčelinjim kolonijama. Njihov postupak pretraživanja baziran je na inteligenciji kolonije pčela prilikom potrage za hranom. Ovde se može izdvojiti doprinos većeg broja autora među kojima se mogu istaći *Nakrani S. i Tovey C.* koji su 2004. godine razvili algoritam medonosnih pčela [128], zatim *Pham D.T. i sar.* su 2005. godine predložili algoritam pčela [129] i autor *Karaboga D.* koji je iste godine doprineo razvoju algoritma veštačke kolonije pčela [106].

Naredni period pa sve do danas svedoči velikoj ekspanziji razvoja algoritama čija inspiracija polazi iz raznih sfera prirode i društva. Iz tog razloga u ovom tekstu je izdvojen učinak pojedinih autora koji su svojim dostignućima pružili značajan doprinos razvoju ove oblasti veštačke inteligencije. Prvi među njima je *Xin-She Yang*, autor koji se svrstava među najpoznatije iz oblasti inteligencije rojeva i računске inteligencije. Zaslužan je za razvoj nekih vrlo popularnih metaheurističkih algoritma kao što su algoritam svica [35], algoritam kukavičje pretrage [102], algoritam slepog miša [36] i algoritam polinacije cveća [130]. Pored toga, ističe se po objavljenim publikacijama na temu inženjerske optimizacije i biološki inspirisanih metaheuristika [24,27,95].

Rashedi E. i sar. [131] bazirali su svoje algoritamsko rešenje na Njutnovim zakonima gravitacije i kretanja. U pitanju je gravitacioni pretraživački algoritam, popularna metaheuristika zasnovana na zakonima fizike. Kod ovog algoritma rešenja predstavljaju objekte određene mase koji stupaju u interakciju sa ostalim objektima iz populacije. Pri tome dolazi do privlačenja, odnosno objekti manje mase svoje kretanje usmeravaju ka objektima veće mase.

Kaveh A. se izdvaja po doprinosu razvoju novih metaheuristika i njihovoj primeni na problemima građevinskog inženjerstva i mehanike. Zaslužan je za pojavu algoritma inspirisanog eholakcijom delfina [30] i drugih fizički inspirisanih algoritama među kojima se može izdvojiti algoritam zasnovan na svetlosnim zracima [132].

Mirjalili S. pripada novijoj generaciji autora koji je afirmaciju u oblasti inteligencije rojeva stekao u poslednjih sedam godina. Uspešan prodor u ovom polju ostvario je razvojem nekoliko interesantnih metaheuristika kao što su algoritam sivog vuka [88], algoritam grbavog kita [32], algoritam rojeva salpi [109], algoritam mravolovaca [34], zatim algoritmi inspirisani noćnim leptirima [103], vilinim konjicima [33], skakavcima [98] kao i algoritam inspirisan kosmologijom [133]. Takođe, može se izdvojiti slučaj koji ne pripada oblasti inteligencije rojeva, a radi se o algoritmu inspirisanom sinusnom i kosinusnom funkcijom [134]. Mnogi od navedenih metaheurističkih algoritama uveliko pronalaze primenu kako u jednokriterijumskoj tako i višekriterijumskoj optimizaciji. Iako im je izvorna primena u kontinualnoj optimizaciji, mogu se pronaći primeri njihove primene za probleme uslovne, diskretne i druge optimizacione probleme.

Askarzadeh, A. je izvršio primenu različitih algoritama pretraživanja u oblasti elektroenergetskih sistema i obnovljivih izvora energije. Veliki uspeh u pogledu citiranosti

ostvario je razvojem prirodno inspirisanog algoritma vrana [105] koji oponaša inteligenciju vrana prilikom krađe i čuvanja hrane od drugih vrana. Takođe, izdvaja se i njegov algoritam inspirisan parenjem ptica [135] koji se svrstava u grupu savremenijih evolutivnih algoritama.

Dhiman G. jedan je od autora koji se u poslednjih par godina izdvaja u polju razvoja biološki inspirisanih metaheurističkih algoritama za rešavanje problema jednokriterijumske i višekriterijumske optimizacije kao i kompleksnih problema velikih dimenzija. Naučnu reputaciju zajedno sa svojim saradnicima stekao je razvojem algoritma pegave hijene [90], algoritma carskih pingvina [112], algoritma galeba [99], algoritma plaštaša [136], algoritma čađave čigre [137] i drugih biološki inspirisanih algoritama. Pored toga, i dalje radi na razvoju kako novih metaheurističkih rešenja tako i na predlaganju strategija za njihovo modifikovanje.

4. PREGLED STANJA U OBLASTI OPTIMIZACIJE TEHNOLOŠKIH PROCESA OBRADE PROIZVODA

4.1 Genetski algoritmi u optimizaciji tehnoloških procesa obrade

Polazeći od prvih primena genetskih algoritama za rešavanje kombinatornog problema optimizacije tehnoloških procesa [138], do danas mnoga hibridna i modifikovana rešenja koja uključuju genetske algoritme i dalje se uveliko razvijaju i implementiraju. Razlozi prvenstveno leže u činjenici da je potrebno određene limite genetskih algoritama, kao što su prerana konvergencija i slabiji kapaciteti lokalnog pretraživanja, prevazići dodatnim modifikacijama, odnosno adaptiranju datom problemu optimizacije. U skladu sa time, u nastavku će se pružiti kratak osvrt na savremenije pristupe koji uključuju genetske algoritme i mehanizme njihovog poboljšavanja u cilju rešavanja problema optimizacije tehnoloških procesa.

Li W.D. i sar. [139] predložili su hibridni algoritam koji kombinuje populacione i globalne prednosti genetskog algoritma sa jedne strane, i individualne i lokalne prednosti algoritma simuliranog žarenja sa druge strane. Na taj način, globalna optimizacija tehnoloških procesa se može ostvariti na efektivan način pri čemu se inicijalno zadovoljavajuća rešenja generišu u prvoj fazi pretraživanja primenom genetskog algoritma, nakon čega se primenjuje algoritam simuliranog žarenja sa ciljem pretraživanja optimalnih ili približno optimalnih tehnoloških procesa.

Salehi M. i Tavakkoli-Moghaddam R. [43] izvršili su optimizaciju tehnoloških procesa primenom genetskog algoritma u dve faze, preliminarnoj i detaljnoj fazi optimizacije. U okviru preliminarne faze, generiše se inicijalna populacija jedinki koja je usklađena sa prethodno postavljenim i analiziranim ograničenjima što kao rezultat daje izvodljive redoslede zahvata, odnosno izvodljive tehnološke procese. Faza detaljne optimizacije orijentisana je na dobijanje optimalnih jedinki, odnosno tehnoloških procesa što uključuje optimalnu selekciju varijantnih mašina, alata i SPA. Genetski algoritam kao efikasna metoda u radu odlikuje se po usvojenoj turnirskoj selekciji hromozoma, zatim delimično mapiranom (PMX), kao i rednom (OX) ukrštanju i dvopozicionoj mutaciji. Prizmatični deo iz [138] usvojen je u eksperimentalnoj studiji u kojoj su testirane performanse ovog prilaza putem minimizacije ukupnih troškova obrade datog dela.

Sličan pristup prethodnom može se pronaći u radu [76], u kojem se glavna razlika u odnosu na prethodni ogleda u primeni inteligentne pretrage, heuristike koja pored genetskog algoritma učestvuje u preliminarnoj i detaljnoj fazi optimizacije. Zadatak inteligentne pretrage je da prevede neizvodljive jedinice u izvodljiva rešenja koja pri tom ne narušavaju prethodno postavljena ograničenja. Ova heuristika vrši proveru izvodljivosti prema prethodno formiranom grafu prethođenja zahvata. Postupak je usklađen sa definisanim ograničenjima koja se tiču redosleda obrade kao i ograničenjima koja se odnose na grupisane zahvate čija obrada se može obaviti u istoj pripremi mašine.

Genetski algoritam za rešavanje problema određivanja plana stezanja i redosleda zahvata obrade u fleksibilnom tehnološkom okruženju primenjen je u radu autora *Kafashi S.* [45]. Sa naglaskom na tehnološke zahteve i ograničenja, odnose tolerancija i predstavljanje rešenja na bazi tipskih oblika, generisani su zadovoljavajući planovi stezanja i redosledi zahvata obrade. Pravila prethođenja formulisana su u matičnom obliku. Operatori rednog ukrštanja i dvopozicione mutacije usvojeni su kao osnovne komponente genetskog algoritma i rezultati optimizacije su minimizirani troškovi obrade prizmatičnog modela donjeg kućišta. Performanse algoritma su testirane i predstavljene na krivoj konvergencije.

Huang W. i sar. [12] razvili su hibridni pristup zasnovan na grafu prethođenja i genetskom algoritmu. Kombinujući teoriju grafova kao i teoriju matrica, odnosno, grafove prethođenja i matrice susedstva, formulisana su ograničenja prethođenja. Potom je razvijen je genetski algoritam baziran na modifikovanom operatoru ukrštanja i dva operatora mutacije za rešavanje problema optimizacije tehnoloških procesa. Dva mehanizma za popravku hromozoma su usvojena radi prevođenja neizvodljivih tehnoloških procesa u domen izvodljivih rešenja. Nakon generisanja inicijalne populacije primenjen je algoritam topološkog sortiranja, dok je heuristički algoritam popravke usvojen nakon prvog dvopozicionog operatora mutacije. Efikasnost i izvodljivost ovog hibridnog metaheurističkog algoritma testirani su minimiziranjem troškova obrade popularnog prizmatičnog dela iz literature.

Hibridni model sličan prethodnom primeru, razvijen je u delu autora *Borojević S.* [16] i koji predstavlja deo programskog sistema za simultano projektovanje proizvoda i tehnoloških procesa. Modifikovani genetski algoritam primenjen je u okviru planiranja tehnoloških procesa izrade hidrauličnog cilindra, a koje pripada fazi detaljnog projektovanja hidrauličkog modula prese za ugaono savijanje lima. Razlika u odnosu na prethodni hibridni algoritam odražava se u primeni samo jednog algoritma za popravku hromozoma koji je ekvivalentan prethodnom heurističkom algoritmu popravke čiji postupak prevođenja rešenja u domen izvodljivih stupa nakon dvopozicione mutacije. Pored te faze, isti algoritam primenjen je i nakon generisanja inicijalne populacije. Na taj način, uvećana je efikasnost predloženog hibridnog pristupa i optimalni rezultati za redosleda izvođenja zahvata obrade hidrauličkog cilindra dobijeni su minimizacijom ukupnih troškova obrade.

Li i sar. [140] zaslužni su za razvoj savremenog genetskog algoritma koji je implementiran za rešavanje problema određivanja procesa obrade rupa na prizmatičnom delu primenom paralelnog bušenja uz simultano korišćenje dva rezna alata. Genetski algoritam sa klasičnim strategijama razvijen je za dve faze optimizacije. U prvoj fazi,

izvršena je optimizacija parametara paralelne obrade – pomaka pri bušenju i brzine glavnog vretena, i dobijena su minimalna vremena obrade svake rupe pri čemu su se poštovala ograničenja u pogledu pomaka, brzine glavnog vretena, sile rezanja, obrtnog momenta, snage i postojanosti alata. U drugoj fazi, genetski algoritam je takođe primenjen u cilju dobijanja optimalnog redosleda obrade sa položajem rupa kao glavnim ograničenjem. Kriterijum optimizacije u ovoj fazi je minimizacija vremena završetka svih zahvata bušenja. Da bi se efektivnost ovakvog prilaza verifikovala, sprovedene su dve studije slučaja, jedna u kojoj je razmatrana paralelna obrada većeg broja rupa na jednom paru paralelnih površina i druga u kojoj je izvršena paralelna obrada većeg broja rupa raspoređenih na većem broj paralelnih površina.

Su i sar. [17] predložili su hibridni pristup na bazi genetskog algoritma i lokalne pretrage za rešavanje problema određivanja redosleda obrade na CNC strugu. Problem optimizacije redosleda obrade formulisan je u vidu modela mešovito celobrojnog programiranja u okviru kojeg su definisana ograničenja prethođenja kao i funkcija cilja koja podrazumeva minimizaciju ukupnog vremena obrade dela. Pored redosleda obrade, pristup razmatra dve moguće pripreme mašine i dve revolver glave na CNC strugu koji zajedno sa ostalim elementima definišu proces obrade. Pored operacije struganja, na ovoj mašini mogu se takođe realizovati operacije glodanja i bušenja što je prikazano na primeru optimizacije rotacionog dela sa 33 tipska oblika i 43 zahvata obrade.

Još jedan hibridni optimizacioni algoritam koji iskorišćava globalne kapacitete genetskog algoritma i lokalne kapacitete simuliranog žarenja može se pronaći u radu autora *Huang W. i sar.* [47]. Na bazi geometrijskih i tehnoloških relacija razvijena su ograničenja prethođenja koja su predstavljena na usmerenom grafu a potom mapirana u matricu prethođenja. Prilikom inicijalizacije početne populacije hromozoma, neizvodljiva rešenja su eliminisana primenom algoritma topološkog sortiranja. Prva faza pretraživanja upošljava genetski algoritam koji za cilj ima generisanje izvodljivih rešenja, a zatim se u drugoj fazi, određeni broj jedinki sa najboljim vrednostima funkcije cilja usvajaju kao inicijalna rešenja za algoritam simuliranog žarenja. Ovaj algoritam zatim ima zadatak da pronađe približno optimalna i/ili optimalna rešenja, odnosno redoslede zahvata obrade. Kompleksniji primer dela sa 28 tipskih tehnoloških oblika i 46 zahvata obrade usvojen je u eksperimentalnoj studiji u kojoj je izvršena minimizacija troškova obrade dela.

U studiji koju su predstavili *Su i sar.* [71], primenjen je genetski algoritam sa novom strategijom kodiranja jedinki za rešavanje problema optimizacije tehnološkog procesa. Ova strategija kodiranja zasnovana na selekciji grana na grafu obezbeđuje izvodljivost hromozoma i utiče na poboljšanje efikasnosti genetskog algoritma tako što ne iziskuje mehanizme popravke rešenja. Pored nove strategije kodiranja, genetski algoritam uključuje operator rednog ukrštanja koji ne narušava postavljena ograničenja prethođenja. Takođe, implementiran je i modifikovani operator mutacije koji slično prethodnom operatoru ne remeti izvodljivost rešenja prilikom izmena varijantne mašine, alata i SPA.

Dou i sar. [141] predložili su poboljšani genetski algoritam za rešavanje problema određivanja redosleda zahvata obrade. Glavna prednost ovog metaheurističkog pristupa ističe se u činjenici da se kao i u prethodnom pomenutom slučaju ne koriste kaznene

funkcije ili mehanizmi za popravke rešenja. Cilj je da kodirana rešenja prilikom inicijalizacije početne populacije kao rezultat daju prostor izvodljivih rešenja koji se potom neće poremetiti u narednim koracima algoritma. Tako se izvodljivi redosledi zahvata podvrgavaju izmenama koje nameću elitističko-bazirana selekcija hromozoma, zatim fragmentarno ukrštanje i fragmentarna mutacija koji osiguravaju usklađenost sa ograničenjima prethođenja. Istraživačke mogućnosti poboljšanog genetskog algoritma uvećane su usvajanjem adaptivnih verovatnoća ukrštanje i mutacije koje se prilagođavaju promenama maksimalne i prosečne vrednosti funkcije cilja u okviru generacije.

Falih i Shammari [142] su razvili hibridni pristup na bazi algoritma ograničene permutacije i genetskog algoritma. Algoritam ograničene permutacije koristi se za generisanje izvodljivih redosleda zahvata pomoću minimizacije broja promena stezanja obratka i broja promena reznih alata. Genetski algoritam je zatim uposlen sa ciljem pronalaženja optimalnog rešenja u izvodljivom prostoru pretraživanja. Izbegavanje prerane konvergencije genetskog algoritma omogućeno je uvođenjem operatora mešovitog ukrštanja koji kombinuje prednosti troslojnog (3SX) i rednog ukrštanja. Sledi operator mutacije resursa koji vrši promene nad vektorima mašina, alata i pravaca prilaza alata sa ciljem da obezbedi neophodan diverzitet populacije.

U studiji autora *Liu i sar.* [143] formulisan je novi matematički model mešovitog celobrojnog linearnog programiranja na bazi mrežnog grafa sa ILI čvorovima. Hibridni evolutivni algoritam razvijen na osnovu ovog MILP modela kombinuje genetski algoritam sa simuliranim žarenjem. Specijalan metod kodiranja i dekodiranja hromozoma zajedno sa modifikovanim genetskim operatorima omogućuju izbegavanje neizvodljivih rešenja koji narušavaju usklađenost sa ograničenjima prethođenja. Lokalno pretraživanje je poboljšano uvođenjem algoritma simuliranog žarenja, dok je turnirska selekcija sa promenljivom veličinom turnira uvedena kako bi se smanjio rizik od prerane konvergencije algoritma i slučajnog pretraživanja.

Luo i sar. [144] bavili su se problemom sekvenciranja redosleda zahvata obrade u okviru optimizacije tehnoloških procesa sa naglaskom na probleme velikih dimenzija. Razvijen je hibridni metaheuristički algoritam koji kombinuje prednosti genetskog algoritma sa prednostima metode promenljivih okolina. Ideja iza ovog prilaza zasnovana je na dekomponovanju ogromnog i kompleksnog prostora mogućih rešenja na veći broj manjih prostora. Svaki od tih prostora, tzv. „okolina“ formiran je na osnovu mehanizma generisanja okolina pri čemu genetski algoritam pronalazi optimalno rešenje u svakoj okolini dok se metod promenljivih okolina bazira na prebacivanju u naredne okoline. Na taj način izbegava se prerana konvergencija i pretražuju se različiti prostori pretraživanja što za posledicu ima pronalaženje optimalnog rešenja. Korišćenjem grafova i matrica za prikaz prioriteta odnosa među zahvatima, minimizacija troškova obrade je usvojena za kriterijum optimizacije. Performanse hibridnog algoritma testirane su na složenom primeru bloka motora sa 42 tipska tehnološka oblika koji iziskuju ukupno 76 zahvata obrade.

4.2 Optimizacija rojem čestica u optimizaciji tehnoloških procesa obrade

PSO algoritam karakteriše pre svega jednostavnost primene, a kao jedan od glavnih nedostataka izdvaja se sklonost „zarobljavanju“ u lokalnom optimumu. Stoga s pribegava mehanizmima i načinima za poboljšanje pretraživanja kada se pristupa rešavanju kompleksnijih problema i njihovih instanci.

Među vidovima poboljšanja performansi PSO algoritma mogu se izdvojiti haotične mape [2,21,145-147], genetski operatori, selekcija, ukrštanje i mutacija [43,77,148] i mnoge druge strategije koje unapređuju proces pretraživanja ove metaheurističke metode.

Napredni algoritam inspirisan rojevima čestica kod kojeg su ugrađene genetske komponente može se pronaći u radovima autora [77,148] koji su usvojili ukrštanje i mutaciju kao glavne strategije za unapređenje procesa pretraživanja. Na ovaj način dobijena su zadovoljavajuća rešenja za nekoliko studija slučajeva u kojima su razmatrani uglavnom prizmatični delovi.

Primena teorije haosa, sa naglaskom na haotične mape, za poboljšanje performansi algoritma inspirisanog rojevima čestica koja se pokazala vrlo uspešnom izvršena je od strane većeg broja autora [21,145,149]. Za deset različitih haotičnih mapa moguće je ostvariti različite pravce prilikom pretraživanja što uveliko doprinosi poboljšanju diverzifikacije, a takođe i intenzifikacije pretraživanja. U ovim radovima izvršena je optimizacija tehnoloških procesa obrade na primerima rotacionih delova. Haotične mape uvedene su za unapređenje drugih metaheurističkih algoritama koji se mogu pronaći u [150-152].

Guo i sar. [13] su predložili modifikovani algoritam inspirisan rojevima čestica za određivanje redosleda zahvata obrade u okviru optimizacije tehnoloških procesa. Poboljšanje istraživačkih mogućnosti PSO algoritma i izbegavanje lokalnih optimuma ostvareni su uvođenjem operatora ukrštanja, mutacije i operatora zamene. Pored toga metod „bežanja“ je takođe primenjen u slučajevima kada ne dolazi do promene najbolje vrednosti funkcije cilja u više od definisanog broja iteracija. Tada se operatori mutacije i zamene primenjuju sa ciljem da spreče zaglavljivanje u lokalnom optimumu.

Wang i sar. [153] razvili su hibridni algoritam zasnovan na algoritmu inspirisanog rojem čestica za rešavanje problema optimizacije tehnološkog procesa. Ovaj pristup karakteriše nova shema kodiranja rešenja koja koristi realne promenljive za reprezentovanje resursa obrade, a koje se kasnije dekodiraju u celobrojne vrednosti. U narednoj fazi sledi heuristički algoritam koji za cilj ima generisanje izvodljivih rešenja. Pored ove heuristike, u strukturu PSO algoritma inkorporirana su dva algoritma lokalnog pretraživanja koji imaju zadatak da unaprede kvalitet dobijenih rešenja. Dva operatora mutacije, kao i dve metode varijacije uposlene su radi modifikovanja matrice položaja jedinki i generisanja novih rešenja.

X. Li i sar. [77] takođe su predložili modifikovani algoritam inspirisan rojevima čestica za optimizaciju tehnoloških procesa. Pored tradicionalnih operatora ukrštanja i mutacije, ovaj modifikovani pristup uključuje algoritam popravke jedinki kako bi se obezbedila izvodljivost rešenja. Sedam studija slučajeva koje predstavljaju sedam različitih delova sprovedeno je u cilju testiranja performansi modifikovanog algoritma u okviru kojeg je minimizacija vremena obrade delova usvojena za kriterijum optimizacije.

Petrović i sar. [21] su u svom radu predstavili savremeni haotični algoritam inspirisan rojem čestica za fleksibilno planiranje tehnoloških procesa. Ovaj prilaz optimizaciji tehnoloških procesa pokriva pet tipova fleksibilnosti: fleksibilnost mašina, fleksibilnost alata, fleksibilnost pravaca prilaza alata, fleksibilnost procesa i fleksibilnost redosleda. Ovi tipovi fleksibilnosti ukazuju na raspoložive varijante u pogledu svakog od navedenih elemenata tehnološkog procesa. I/II mrežne usvojene su za reprezentovanje pet tipova fleksibilnosti tehnoloških procesa. Izvršena je kombinacija sa teorijom haosa pri čemu je deset različitih haotičnih mapa ugrađeno u osnovnu strukturu PSO algoritma sa ciljem povećanja diverziteta prostora pretraživanja i sprečavanja konvergencije algoritma rojeva čestica u ranim fazama optimizacije. Minimizacija ukupnih troškova kao i minimizacija ukupnog vremena obrade delova usvojeni su za kriterijume optimizacije pomoću kojih su dobijeni rezultati za četiri eksperimentalne studije.

Miljković i Petrović [68] razvili su modifikovani višekriterijumski algoritam inspirisan rojem čestica za fleksibilno planiranje tehnoloških procesa. Za unapređenje pretraživanja, algoritam inspirisan rojem čestica nadograđen je genetskim operatorima kao što su rulet selekcija, jednopoziciono ukrštanje, mutacija zamene i mutacija resursa. Kao i u prethodnom primeru, tipovi fleksibilnosti su naglašeni i predstavljeni putem I/II mreža. Optimizacija primenom ovog algoritma izvršena je na četiri eksperimenta u kojima su pokrivena dva kriterijuma, ukupni troškovi obrade kao i ukupno vreme obrade delova.

Dou i sar. [70] autori su rada u kojem je predložen izvodljivi diskretni algoritam inspirisan rojevima čestica orijentisan na određivanje optimalnog redosleda zahvata obrade u optimizaciji tehnoloških procesa. Ovaj prilaz uvodi nove operatore ukrštanja i mutacije sa adaptivnim verovatnoćama koji pored efekata koji imaju na evoluciju rešenja, utiču i na poboljšanje istraživačkih (globalnih) kapaciteta. Uz zadovoljenje ograničenja prethodjenja i minimizaciju troškova obrade dva reprezentativna dela, ovaj hibridni algoritam manipuliše izvodljivim rešenjima u okviru diskretnog izvodljivog prostora pretraživanja. Mehanizam ukrštanja kao i mutacije ne remete izvodljivost rešenja koja biva održana u svim fazama optimiza

cije. Takođe, Tagučijeva metodologija primenjena je sa ciljem pronalaženja optimalnih vrednosti ulaznih parametara algoritma.

4.3 Algoritam kolonije mrava u optimizaciji tehnoloških procesa obrade

Jednu od prvih primena ACO algoritma u optimizaciji tehnoloških procesa izvršili su autori *Krishna A. G. i Rao K. M.* [14] koji su svoje hibridno rešenje bazirali na grafu

prethođenja sa ciljem generisanja optimalnih redosleda, i sa druge strane matrici troškova pomoću koje je omogućeno usvajanje relativnih troškova u zavisnosti od posmatranih tipskih oblika.

Liu X.-J. i sar. [39] razvili su sličan pristup na bazi algoritma kolonije mrava. Metod je primenjen za optimizaciju tehnoloških procesa obrade dva prizmatična dela. Ovaj problem preveden je u problem trgovačkog putnika formiranjem težinskog grafa pomoću kojeg se definiše težina, odnosno troškovi za svaku relaciju između „gradova“ koji u ovom slučaju predstavljaju zahvate obrade. Matrice stanja i ograničenja upotrebljene su kako bi se utvrdilo trenutno stanje raspoloživosti zahvata za obradu datog tipskog oblika i time obezbedili izvodljivi redosledi zahvata.

Wang J.F. i sar. [44] razvili su model sličan prethodnom pri čemu je akcenat u ovom radu postavljen na težinski usmereni graf koji uključuje čvorove – zahvate, zatim usmerene grane – prethođenja i neusmerene grane koje predstavljaju sve moguće putanje kojima se „mravi“ mogu kretati. Na taj način izražena je fleksibilnost redosleda zahvata obrade.

Wang i sar. [72] razvili su algoritam mravlje kolonije za dve faze optimizacije u cilju minimiziranja troškova obrade. Kod ovog pristupa usmereni graf je usvojen za predstavljanje problema optimizacije tehnološkog procesa. Prva faza pretraživanja odnosi se na selekciju odgovarajućih čvorova koji predstavljaju zahvate obrade nakon čega se sam graf redukuje na jednostavniji graf. Kasnije, u drugoj fazi selektovani zahvati se koriste za formiranje redosleda i generisanje procesa obrade.

Hu i sar. [154] predložili su pristup zasnovan na algoritmu mravlje kolonije za rešavanje problema optimizacije redosleda izvođenja zahvata u okviru CAPP sistema. Ograničenja prethođenja i grupisanja zahvata uzimaju se u obzir kako bi se osigurala izvodljivost planova obrade. Kapaciteti globalnog pretraživanja uvećani su uvođenjem metode adaptivnog ažuriranja i mehanizma lokalnog pretraživanja.

Singh i Deb [155] su u svojoj studiji izvršili primenu tzv. algoritma mravljih sistema koji je zasnovan na sličnom principu kao algoritam mravljih kolonija. Autori su pokazali u dva eksperimenta da se algoritam pokazao značajno bržim u procesu pretraživanja u poređenju sa tradicionalnim ACO i GA. Jedinstvene karakteristike koje odlikuju ovaj modifikovani algoritam koncipirane su specifičnim načinom selekcije polaznog „čvora“ u početku mravljeg ciklusa pretrage kao i uvođenjem provere prethođenja zahvata kojom se rešenja održavaju u okvirima dopustivog skupa. Selekcija polaznog čvora sa grafa ograničava svakog „mrava“ da započne redosled direktno od zahvata koji nema nikakvih relacija u pogledu prethođenja sa drugim zahvatima (kao npr. tehnološka baza).

4.4 Pregled ostalih algoritama u optimizaciji tehnoloških procesa obrade

Lukić i sar. [19] izvršili su optimizaciju redosleda izvođenja zahvata u okviru softverskog paketa CATIA. Sa naglaskom na definisana pravila prethođenja formirane su

matrice za određivanje redosleda zahvata. Redosledi zahvata obrade su potom grupisani u odgovarajuće operacije obrade nakon čega su na izlazu dobijene racionalne varijante mogućih tehnoloških procesa. Slučaj koji se razmatrao je prizmatični deo sličan primeru donjeg kućišta koji je predložen u [45].

Lian i sar. [20] predložili su primenu imperijalističkog algoritma za rešavanje problema optimizacije tehnološkog procesa. Ovaj slučaj takođe razmatra tipove fleksibilnosti u pogledu redosleda zahvata, mašina, alata, SPA i samog tehnološkog procesa. Imperijalistički optimizacioni algoritam inspirisan je društveno-političkim odnosima i u okviru svog koncepta koristi korake asimilacije, takmičenja, revolucije i eliminacije prilikom traganja za optimalnim planovima tehnološkog procesa.

Lian K. i sar. [40] primenili su algoritam tabu pretraživanja koji je karakterističan po posedovanju memorije o pretraženim rešenjima i „zabrani“ njihovog ponavljanja u narednim koracima pretraživanja. Prilaz rešavanju problema u ovom radu zasnovan je na posmatranju problema u četiri dimenzije, koje analogno tipovima fleksibilnosti formulišu problem OTP u četiri dimenzije: redosled zahvata obrade, redosled mašina, redosled alata i redosled SPA. Pored standardnih koraka algoritma tabu pretrage, u višedimenzioni tabu pretraživački algoritam inkorporirane su globalne strategije koje vrše modifikacije u strukturi redosleda zahvata, kao i operator mutacije koji za cilj ima modifikovanje redosleda mašina, alata i SPA respektivno.

Li i sar. [156] izvršili su optimizaciju tehnoloških procesa primenom modifikovanog algoritma tabu pretraživanja. Koncept ovog pristupa sličan je njihovom prethodnom doprinosu ovoj problematici koji se odnosio na implementaciju hibridnog algoritma koji kombinuje prednosti genetskog algoritma i simuliranog žarenja [139]. Modifikovani tabu algoritam zasnovan je na tri osnovne strategije koje omogućavaju proveru optimalnosti tehnoloških procesa. Sa jedne strane strategija zabrane omogućava kontrolu rešenja koje ulaze u tabu listu, dok strategija oslobađanja upravlja rešenjima koji napuštaju tabu listu nakon izvesnog vremena. Poslednja, strategija aspiracije ima zadatak da uzajamno koristi prethodne dve strategije kako bi algoritam težio pronalaženju novih a ujedno i boljih rešenja, odnosno tehnoloških procesa. U ovu metodu uključena je takođe hibridna metoda upravljanja ograničenjima koja omogućava izvodljivost svih redosleda i samim tim rešenja održava u okviru dopustivog skupa. Sa naglaskom na dva prizmatična modela optimizacija primenom modifikovanog tabu pretraživanja smatra se jednim od tradicionalnih metoda pored genetskih algoritama, simuliranog žarenja i optimizacije rojem čestica koji su primenu pronašli u rešavanju problema OTP.

Wen i sar. [74] razvili su algoritam inspirisan parenjem pčela (HBMO) gde je naglasak postavljen na seksualnom ponašanju trutova i pčele kraljice. Osnovnu proceduru algoritma karakterišu let pčele kraljice i generisanje i poboljšanje položaja trutova u letu. Pored osnovnih koraka, u HBMO su uključene strategije kodiranja rešenja, operater ukrštanja i lokalna pretraga sa ciljem povećanja efikasnosti predloženog algoritma. Zadovoljavajući rezultati su postignuti testiranjem performansi na tri eksperimentalne studije.

Petrović i sar. [2,9] zaslužni su za primenu algoritma inspirisanog mravolovcima u optimizaciji tehnoloških procesa. Pridržavali su se standardnog istraživačkog pristupa

baziranog na I/ILI mrežama i tipovima fleksibilnosti tehnoloških procesa sa ciljem njihove optimizacije pomoću savremenog algoritma iz polja inteligencije rojeva koji je predložen u [34]. Izvršena je optimizacija ukupnog vremena obrade za jednostavan primer sa 9 zahvata obrade.

Nallakumarasamy G. i sar. [10] primenili su algoritam simuliranog žarenja u studiji u kojoj je izvršena optimizacija troškova obrade jednog rotacionog i jednog prizmatičnog proizvoda za koje su prethodno definisane odgovarajuće tolerancije oblika i položaja. Izvodljivost redosleda zahvata obezbeđena je pomoću matrice relativnih troškova u okviru koje su, na osnovu prethođenja među tipskim oblicima i informacija o promenama mašina, alata i SPA, generisani izvodljivi redosledi zahvata obrade.

Lu i Kiao [67] predložili su novi heuristički metod, tzv. metod unakrsne entropije, za optimizaciju fleksibilnih tehnoloških procesa. Autori su preko I/ILI mreža predstavili fleksibilne tehnološke procese i potom definisali matematički model za minimiziranje ukupnog vremena kao i ukupnih troškova fleksibilnih tehnoloških procesa. Uveden je nov postupak predstavljanja rešenja, a zatim i parametar verovatnoće distribucije koji uz metod ažuriranja omogućava generisanje boljih rešenja u narednim iteracijama algoritma.

Wang i sar. [73] usvojili su hibridni algoritam slepih miševa za problem OTP sa fokusom na oba ključna zadatka, izbor zahvata i određivanje redosleda zahvata. Predložene su strategije kodiranja, dekodiranja i inicijalizacije, a dve lokalne strategije pretraživanja ugrađene su u standardni algoritam slepih miševa (BA) kako bi se poboljšala njegova lokalna konvergencija. Izvršen je klasični simulacioni eksperiment kako bi se verifikovale performanse hibridnog BA.

Xu i sar. [75] predstavili su novi pristup optimizacije fleksibilnog planiranja procesa usmerenog ka ponovnom korišćenju procesa za prizmatične delove sa ciljem minimiziranja ukupnih proizvodnih troškova. Predstavljen je hibridni algoritam kolonije mrava (ACO) i simuliranog žarenja (SA) zasnovan na grafu prethođenja kako bi se generisao globalni optimalni NC plan obrade za posmatrani deo.

Gao i sar. [157] predložio je algoritam inteligentnih vodenih kapljica za rešavanje problema planiranja procesa. U prvoj fazi optimizacije definisani su zahvati i tehnološki resursi prema tehnološkim karakteristikama i ograničenjima prethođenja vezanim za razmatrane mašinske delove. Nakon toga algoritam vodenih kapi primenjen je za rešavanje problema. Koncept ove metode je takav da protok vodenih kapljica u rečnom koritu predstavlja jednu jedinicu definisanu tipskim zahvatom obrade, dok sa druge strane količina zemlje koju vodene kapi u rečnom koritu sadrže predstavlja vrednost troškova obrade čija je minimizacija izvršena u ovoj studiji. Pretraživanje prostora dopustivih rešenja omogućeno je pomoću matrice prioriternih odnosa.

U jednoj od novijih studija, *Milošević i sar.* [158] su predstavili inteligentno projektovanje tehnoloških procesa za koncepte pametne fabrike i pametne proizvodnje. Predloženi su prirodno-inspirisani metaheuristički algoritmi kao „pametni“ servisi veštačke inteligencije za inteligentnu optimizaciju tehnoloških procesa u okviru pametne proizvodnje. Tri savremena biološki inspirisana algoritma, GWO, WOA i CSA primenjeni su na tri klasične studije slučaja iz literature.

Durđev i sar ([159]) razvili su modifikaciju savremenog algoritma vrane u svrhu optimizacije tehnoloških procesa obrade prizmatičnih delova. Prilaz se zasniva na definisanju ograničenja prethođenja, njihovom predstavljanju na grafovima i matricama prethođenja i vektorskom predstavljanju varijanti tehnoloških procesa. Algoritam CSA je unapređen implementacijom genetskih operatora: turnirske selekcije, 3SX ukrštanja, mutacije zamene i mutacije resursa. Predložen je mehanizam popravke rešenja na osnovu kojeg se vrši prevođenje neizvodljivih tehnoloških procesa u izvodljiva rešenja. Testiranje i verifikacija GCSA algoritma izvršena je u dve eksperimentalne studije od kojih se prva odnosi na tradicionalan prizmatičan deo koji je predmet dugogodišnjih istraživanja u ovoj oblasti. Drugi eksperiment podrazumeva kompleksniji primer prizmatičnog proizvoda sa znatno većim brojem varijanti tehnološkog procesa obrade. Izvršena je uporedna analiza sa drugim rezultatima iz literature pri čemu se GCSA pokazao efikasnom metodom za pronalaženje optimalnih redosleda zahvata obrade u okviru ovog istraživanja.

5. SAVREMENI BIOLOŠKI INSPIRISANI ALGORITMI U OPTIMIZACIJI TEHNOLOŠKIH PROCESA OBRADE PROIZVODA

Kako se već može zaključiti na osnovu informacija iz prethodnih poglavlja, dimenzije problema OTP eksponencijalno se povećavaju sa povećanjem broja zahvata, kao i mašina, alata i SPA koji se upotrebljavaju za njihovu realizaciju. S obzirom na kompleksne prostore pretraživanja koji se mogu javiti kod određenih instanci ovog problema, vrši se implementacija metaheurističkih algoritama kako bi se došlo do optimalnih ili približno optimalnih rešenja.

U poslednjih nekoliko decenija primenjen je znatan broj metaheurističkih metoda sa ciljem rešavanja OTP problema. Počevši od genetskih algoritama (GA), simuliranog žarenja (SA), kolonije mrava (ACO), tabu pretraživanja (TS), optimizacije rojevima čestica (PSO) i mnogih drugih, dok su se u poslednje vreme pojavili različiti hibridni pristupi i metode kao što su algoritam kapljica vode (IWD) ili algoritam vrane (GCSA). U nastavku će biti predstavljena teorijska analiza savremenih biološki inspirisanih metaheurističkih algoritama koji su primenjeni u eksperimentalnim studijama, njihov osnovni koncept, pseudokodovi i predložene modifikacije.

5.1 Modifikovani algoritam vrane (GCSA) u optimizaciji tehnoloških procesa obrade

5.1.1 Algoritam vrane (CSA)

Tradicionalni algoritam vrane pripada grupi prirodom-inspirisanih metaheurističkih algoritama koji je 2016. godine razvio autor *Askarzadeh* [105]. Inspiracija polazi od inteligentnog ponašanja vrana prilikom krađe i sakrivanja hrane od drugih vrana.

Vrane (lat. *Corvus*) su široko rasprostranjena vrsta ptica i generalno se ubrajaju među najinteligentnije žive organizme, a pored čavki i svraka smatraju se najpametnijim pticama na planeti (Slika 5.1). U pogledu kognitivnih sposobnosti mogu parirati primatima

kao što su šimpanze ili gorile [160]. Vrane karakterišu različiti kvaliteti kao što su samosvest, prefinjeni vid komuniciranja, korišćenje i kreiranje raznih oruđa, zatim mogućnost pamćenja lica, skrivanja hrane, krađe hrane od drugih vrana, izbegavanje krađe i mnogi drugi. Prepoznatljive su po svojoj pohlepi i lukavosti koje se posebno ističu u situacijama kada prate kretanje drugih vrana s namerom da pronađu njihovo skrovište sa hranom koju će potom ukrasti od njih. Kada izvrše krađu, vrane kradljivci se pobrinu da ukradena hrana bude dobro sakrivena kako bi izbegle da one same postanu žrtve drugih lopova kao što su to bili prethodni vlasnici. Lukavost vrane je takođe izražena u slučajevima kada vrana poseduje svest o tome da druga vrana prati njeno kretanje. Ukoliko poseduje lopovsko iskustvo jer je sama bila kradljivac, sposobna je da predvidi ponašanje drugih vrana kradljivaca ne bi li ih prevarila i na taj način održala hranu ili plen dobro skrivenim.



Slika 5.1. Vrane u potrazi za hranom [161,162]

5.1.1.1 Matematički model CSA algoritma

Algoritam vrane, CSA, je optimizacioni algoritam zasnovan na populaciji rešenja koji karakteriše jednostavan izvorni kod kao i sam postupak implementacije. To se prvenstveno ogleda u činjenici da postoje samo dva parametra koje je potrebno podesiti za razliku od npr. GA ili PSO algoritma koji zahtevaju više parametara. Sa optimizacionog stanovišta, jato vrana podrazumeva populaciju agenata pretraživanja N gde je svaka vrana i okarakterisana svojim položajem X_i^{it} u iteraciji it . Svaka vrana u jatu sposobna je da zapamti (memoriše) svoje skrovište sa hranom pri čemu se ta memorija položaja označava sa M_i^{it} . Slično lokalnom najboljem rešenju, kao što je lokalni najbolji položaj čestice u PSO algoritmu, kod CSA se smatra da je memorija M_i^{it} najbolji izvor hrane (najbolje skrovište) koji je vrana i pronašla do tog trenutka, odnosno do iteracije it . Takozvani „let“ oko prostora pretraživanja i kretanje prema skrovištu druge vrane pomaže vranama da ažuriraju svoj položaj i memoriju i na taj način potencijalno pronađu bolji izvor hrane. Dva glavna scenarija mogu biti rezultat pokušaja vrane i da ukrade skrivenu hranu vrane j [105]:

- 1) *Vrana j nije svesna da je prati vrana i što rezultira situacijom kada vrana i uspešno vrši krađu hrane krećući se prema skrovištu vrane j . To se može izraziti pomoću sledećeg matematičkog izraza:*

$$X_i^{it+1} = X_i^{it} + rand_i \times fl_i^{it} \times (M_j^{it} - X_i^{it}) \quad (25)$$

gde su X_i^{it} i X_i^{it+1} položaji vrane i u iteracijama it i $it+1$, $rand_i$ označava slučajni broj između 0 i 1, fl_i^{it} je dužina leta vrane i u iteraciji it dok je M_j^{it} je memorisani položaj vrane j u iteraciji it .

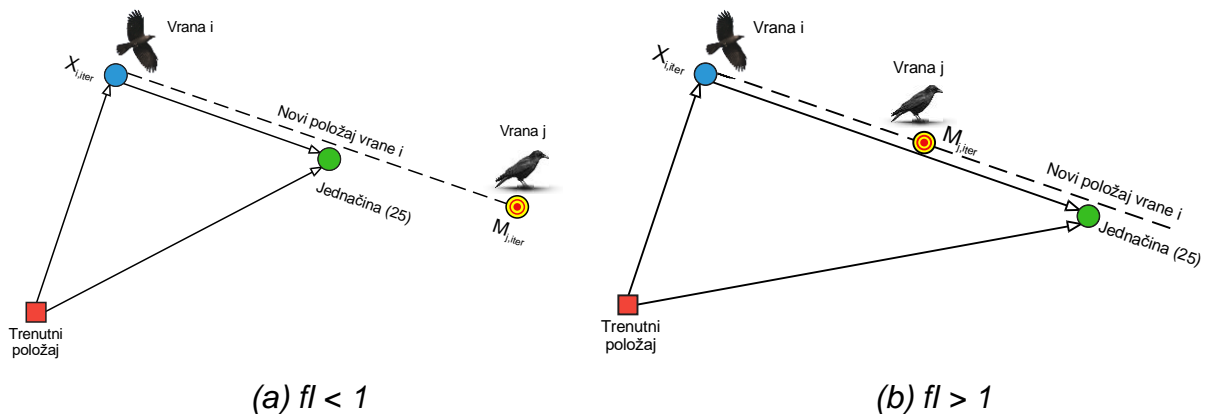
- 2) Vrana j uviđa da je vrana i prati, pri čemu je vara tako što nasumično bira novi položaj u prostoru pretraživanja sa ciljem da sačuva svoju hranu i spreči da bude pokradena. Verovatnoća da će doći do ovog slučaja izražava na sledeći način:

$$X_i^{it+1} = \begin{cases} \text{isto kao } i \text{ (25)} & \text{ako je } rand_j \geq AP_i^{it} \\ \text{slučajni položaj} & \text{ako je } rand_j < AP_i^{it} \end{cases} \quad (26)$$

gde je $rand_j$ slučajni broj između 0 i 1 dok je AP_i^{it} verovatnoća svesnosti vrane j u iteraciji it .

Dva glavna parametra CSA algoritma koja utiču na postupak pretraživanja su dužina leta vrane fl i verovatnoća svesnosti AP . Glavna uloga parametra dužine leta fl je da uravnoteži lokalne i globalne pretraživačke kapacitete CSA. Manje vrednosti parametra fl usmeravaju pretraživanje ka lokalnom optimumu, dok veće vrednosti ovog parametra vode pretraživanje ka globalnom optimumu. Kako je ilustrovano na slici 5.2a, vrednosti dužine leta koje su manje od 1 podrazumevaju da se novi položaj vrane i nalazi na isprekidanoj liniji između trenutnog položaja X_i^{it} i memorije položaja M_i^{it} , odnosno bliže položaju X_i^{it} . Sa druge strane, prema slici 5.2b, kada su vrednosti dužine leta veće od 1, novi položaj vrane i prelazi memoriju vrane j , odnosno udaljava se od položaja X_i^{it} .

Verovatnoća svesnosti vrane AP , kao i dužina leta fl ima značaj u balansiranju intenzifikacije i diverzifikacije. Takođe važi da manje vrednosti parametra AP povećavaju lokalne kapacitete pretraživanja, dok veće vrednosti AP naglašavaju globalno pretraživanje.



Slika 5.2. Varijacija parametra dužine leta vrane za dva scenarija

Primene tradicionalnog CSA algoritma mogu se prvenstveno pronaći kod rešavanja nelinearnih kontinualnih problema optimizacije, ali savremena istraživanja baziraju

primenu CSA i u kombinatornoj optimizaciji. Slično mnogim metaheuristikama, glavni problemi CSA koji se mogu u prvi plan izdvojiti su slaba konvergencija i „zarobljavanje“ u lokalnom optimumu [150,163]. Osim toga, razne varijacije u pogledu mehanizama za poboljšanje pretrage i hibridizacije za rukovanje sa različitim promenljivama otvorile su vrata ovoj metodi u mnogim poljima optimizacije. Pseudokod tradicionalnog algoritma vrana prikazan je u algoritmu 1.

Algoritam 1. Pseudokod tradicionalnog CSA algoritma

<p>1. Početak: Inicijalizacija Inicijalizovati parametre CSA: svesnost vrane AP i dužinu leta vrane fl; Inicijalizovati slučajne položaje jedinki (vrana) i formirati početnu populaciju (jato vrana); Odrediti vrednost funkcije cilja svake jedinke (vrane) u početnoj populaciji (jatu); Inicijalizovati memoriju svake jedinke (vrane) u početnoj populaciji (jatu);</p>
<p>2. Ponavljanje: Iterativni postupak Ponoviti iteracija = iteracija + 1; ponoviti za sve vrane u jatu Slučajno selektovati jedinku za praćenje (vrana j); ako je $rand_j \geq AP_i^{it}$ (prema jednačini (25)) ažurirati položaj vrane prema izrazu: $X_i^{it+1} = X_i^{it} + rand_i \times fl_i^{it} \times (M_j^{it} - X_i^{it})$ u suprotnom ažurirati položaj vrane: $X_i^{it+1} =$ slučajan položaj u prostoru pretraživanja završi uslov završi petlju proveriti izvodljivost novih položaja vrana u jatu (dopustivost rešenja) proveriti vrednost funkcije cilja svake vrane u jatu ažurirati memoriju vrana završi petlju</p>
<p>3. Izlazni rezultati</p>

U bazi naučne literature može se pronaći određeni broj radova u kojima je opisana primena modifikovanog algoritma CSA u rešavanju problema kombinatorne optimizacije. *Huang i sar.* [164] predložili su hibridni algoritam vrane za rešavanje NP-teškog problema permutacionog „flow shop“ terminiranja. Koristili su nekoliko tehnika za minimiziranje vremenskog raspona, među kojima se mogu izdvojiti pravila za najmanju vrednost položaja za pretvaranje kontinualnih u diskretne vrednosti, zatim NAH heuristika za generisanje populacije kvalitetnih jedinki, kao i algoritam simuliranog žarenja i metoda promenljivih okolina čiji zadatak je da obezbede diverzifikaciju rešenja tokom kompletnog postupka pretraživanja. *Laabadi i sar.* [165] su primenili binarni algoritam vrane za rešavanje problema pakovanja koji se takođe ubraja u klasu NP-teških kombinatornih problema. Ovaj pristup koristi sigmoidnu funkciju za mapiranje realno-promenljivih rešenja u binarne, i operator popravke baziran na znanju sa ciljem da izvrši poboljšanje diverziteta rešenja i održi ih u domenu skupa dopustivih rešenja. *Mohammadi i Abdi* [166] su usvojili posebnu metodu selekcije ciljnih vrana koje će biti praćene u toku leta (pretraživanja) kao i modifikovane metode za evaluaciju dužine leta vrane sa ciljem optimizacije problema minimiziranja ukupnih ekonomskih troškova za rad energetskih sistema. *Sivarami Reddy i sar.* [167] razvili su nelinearni model mešovito celobrojnog programiranja i primenili

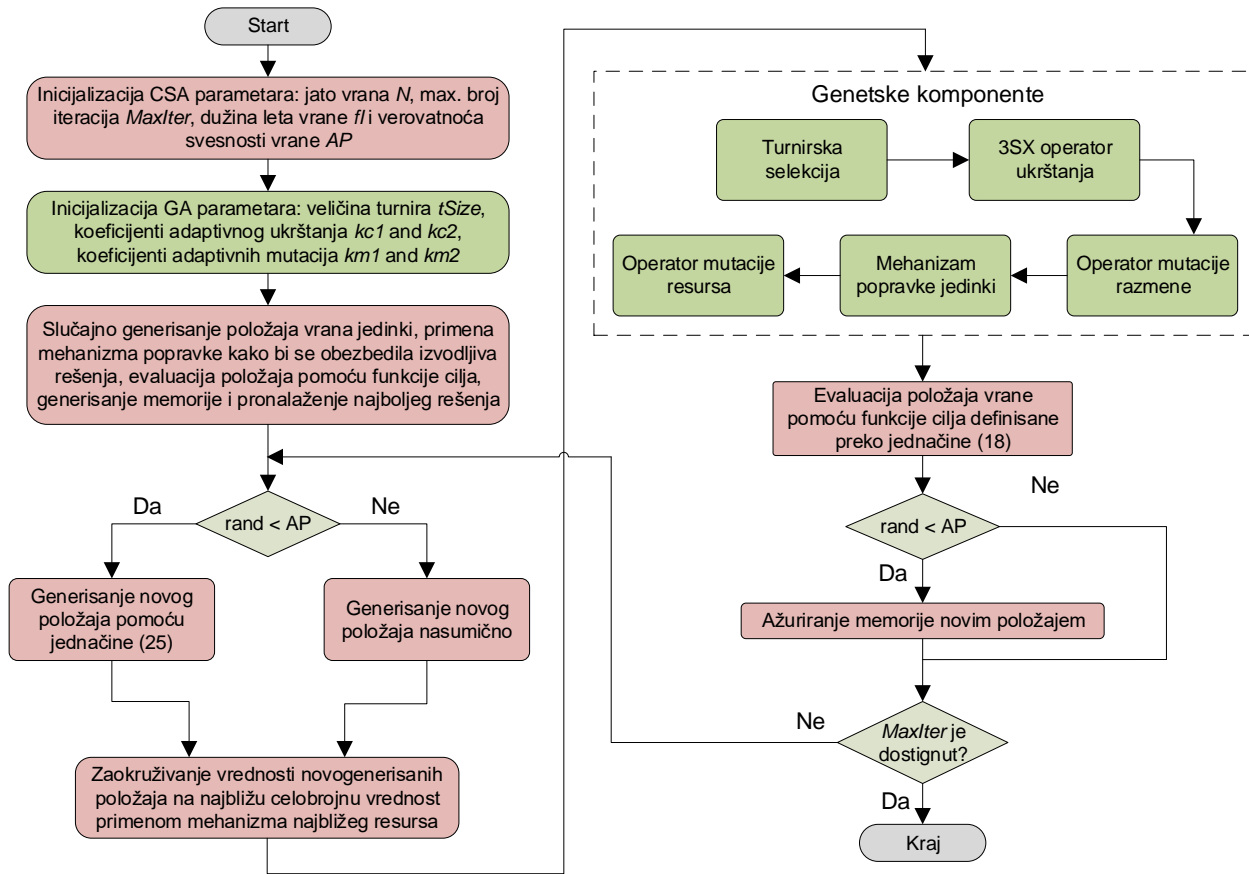
CSA za minimiziranje ukupnog vremena trajanja operacija kod terminiranja mašina, automatski vođenih vozila i alata u fleksibilnom tehnološkom sistemu. *Rizk-Allah* [168] i *Sayed* [150] inkorporirali su različite haotične mape u algoritam vrane sa ciljem rešavanja problema selekcije atributa i problema frakcionalne optimizacije.

Prema gore navedenim informacijama, može se ustanoviti da CSA algoritam, uz dodatak određenih strategija radi poboljšanja pretraživačkih sposobnosti, poseduje potencijal za generisanje optimalnih tehnoloških procesa. U narednom poglavlju opisan je koncept modifikovanog algoritma vrane koji je uspešno implementiran na problemu OTP.

5.1.2 Modifikovani algoritam vrane (GCSA)

U ovom poglavlju dat je opis unapređenja tradicionalnog CSA algoritma u cilju rešavanja NP-teškog problema OTP. Shodno tome, predlaže se hibridni pristup zasnovan na algoritmu vrane i glavnim komponentama genetskog algoritma. Pored toga, u okviru pristupa uključene su adaptivne verovatnoće za regulisanje operatora ukrštanja i operatora mutacije kako bi se ostvario kompromis između lokalnog i globalnog pretraživanja [70]. Koraci GCSA algoritma prikazani su na dijagramu toka na slici 5.3. U skladu sa kombinacijom dve različite metaheurističke metode, pojedinačna rešenja u populaciji biće adresirana kao „vrane jedinke“. U nastavku je opisan koncept ove novorazvijene metode.

Nakon inicijalizacije početnih parametara CSA i GA pojedinačno, GCSA vrši generisanje populacije izvodljivih vrana jedinki primenom vektorskog predstavljanja koji je detaljno opisan u 2.2.6. Vektorskim predstavljanjem formulišu se vektor redosleda zahvata, zatim vektori mašina, alata i SPA kao komponente varijantnog tehnološkog procesa. Vektori koji predstavljaju položaje vrana u prostoru pretraživanja se potom evaluiraju pomoću funkcije cilja, definisane u poglavlju 2.3. Položaji se memorišu kao najbolja lokalna rešenja koja je do tog trenutka pronašla svaka vrana, a najbolje memorisano rešenje skladišti se kao najbolje rešenje u celom jatuu (globalno rešenje populacije). Glavna petlja GCSA započinje standardnim koracima CSA koji su opisani jednačinama (25) i (26) Posle ovih koraka primenjuje se tzv. mehanizam najbližih resursa pomoću kojeg se vrši zaokruživanje rezultujućih vektorskih elemenata na najbliže cele brojeve u skladu sa raspoloživim mašinama, alatima i PPA kandidatima za razmatrani zahvat obrade. Mehanizam najbližih resursa opisan je detaljnije u odeljku 5.1.2.3. Proces pretraživanja dalje se nastavlja uvođenjem genetskih operatora koji za cilj imaju poboljšanje istraživačkih kapaciteta. Usvojene strategije su turnirska selekcija, 3SX operator ukrštanja i dva operatora mutacije, mutacija zamene i mutacija resursa. Ovi operatori opisani su detaljnije u odeljku 5.1.2.4. Mehanizam popravke potreban je nakon faze inicijalizacije kao i nakon primene operatora mutacije zamene a sve sa namerom da se vrane jedinke održe u skupu dopustivih rešenja. Po završetku popravke jedinki, GCSA vrši ponovnu evaluaciju funkcije cilja svih vrana u jatuu nakon koje sledi ažuriranje memorije vrana i ujedno najboljeg rešenja u jatuu (globalnog rešenja). Ovim korakom završava se jedna iteracija GCSA algoritma. Postupak pretraživanja okončava se nakon prethodno definisanog broja iteracija.



Slika 5.3. Dijagram toka GCSA algoritma za optimizaciju tehnoloških procesa

5.1.2.1 Vektorsko predstavljanje tehnološkog procesa

Modifikacija tradicionalnog CSA algoritma podrazumeva razvijanje odgovarajuće procedure za predstavljanje varijantnog tehnološkog procesa. U skladu sa time, usvojen je pristup na bazi vektorskog predstavljanja. Princip vektorskog predstavljanja je objašnjen na primeru prizmatičnog dela iz prve eksperimentalne studije u ovom istraživanju (poglavlje 6.1.1). U tabeli 5.1 nalazi se primer jedne slučajno generisane varijante tehnološkog procesa sa četiri predstavljena vektora. Deo sadrži 20 zahvata obrade pri čemu je formirano ukupno 20 tipskih zahvata.

Tabela 5.1. Vektorsko predstavljanje varijantnog tehnološkog procesa

Indeksi	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	
Zahvati	1	5	7	8	9	10	6	11	2	12	19	20	18	17	13	14	15	16	3	4	
Mašine	2	3	3	2	3	4	3	3	2	1	3	3	3	2	1	3	2	3	3	3	
Alati	8	7	8	3	9	10	8	8	6	4	9	10	6	8	9	10	1	5	8	2	
SPA	+z	+x	-a	-a	-a	-a	-z	-y	-z	-z	+z	+z	-z	-z	-z	-z	-z	-z	-z	+x	+z

Vektorsko predstavljanje tehnoloških procesa u programskom jeziku Matlab vrši se u formi matrica gde su četiri vektora predstavljeni kao četiri matrice tipa $1 \times n$ (n je ukupan broj zahvata). Shodno tome, po uzoru na haotični PSO algoritam iz [21] razvijeni su matematički izrazi za ažuriranje položaja vrana u prostoru pretraživanja koji su usklađeni sa izrazom (25). Vektori mašina, alata i SPA formulisani su na sledeći način:

$$X_{i,m}^{it+1} = X_{i,m}^{it} + rand() \times fl \times (M_{j,m}^{it} - X_{i,m}^{it}) \quad (27)$$

$$X_{i,a}^{it+1} = X_{i,a}^{it} + rand() \times fl \times (M_{j,a}^{it} - X_{i,a}^{it}) \quad (28)$$

$$X_{i,spa}^{it+1} = X_{i,spa}^{it} + rand() \times fl \times (M_{j,s}^{it} - X_{i,spa}^{it}) \quad (29)$$

gde je i broj vrane jedinke; $X_{i,m}^{it}$ i $X_{i,m}^{it+1}$ položaji vrana za vektor mašina u iteracijama it i $it + 1$, respektivno; $M_{j,m}^{it}$ je memorisan položaj vrane j za vektor mašina u iteraciji it ; $X_{i,a}^{it}$ i $X_{i,a}^{it+1}$ položaji vrana za vektor alata u iteracijama it i $it + 1$, respektivno; $M_{j,a}^{it}$ je memorisan položaj vrane j za vektor alata u iteraciji it ; $X_{i,spa}^{it}$ i $X_{i,spa}^{it+1}$ položaji vrana za vektor SPA u iteracijama it i $it + 1$, respektivno; $M_{j,spa}^{it}$ je memorisan položaj vrane j za vektor SPA u iteraciji it ; $rand()$ je slučajan broj u intervalu od 0 do 1 i fl je dužina leta vrane.

Na osnovu formulisanih izraza može se izraziti ukupan položaj vrane jedinke i koja uključuje sva tri vektora:

$$X_i^{it+1} = [X_{i,m}^{it+1}; X_{i,a}^{it+1}; X_{i,spa}^{it+1}] \quad (30)$$

Vektor redosleda zahvata obrade $Zahvat[i, j]$ nije manipulisan u okviru osnovnih koraka CSA algoritma. U tu svrhu primenjeni su genetski operatori koji su opisani u odeljku 5.1.2.4.

5.1.2.2 Kriterijum evaluacije varijantnih tehnoloških procesa

Brojna istraživanja u oblasti OTP-a i integracije projektovanja tehnoloških procesa i planova terminiranja najčešće razmatraju primenu dva kriterijuma za evaluaciju varijantnih tehnoloških procesa ili redosleda izvođenja tipskih zahvata. To su minimizacija ukupnih troškova obrade i minimizacija ukupnog vremena obrade razmatranih mašinskih delova. Prema [47, 157], detaljne informacije o putanjama alata i parametrima obrade nisu dostupne i stoga nije moguće tačno odrediti troškove kao i vreme obrade. Mogućnosti za uključivanje ovih informacija u okvire problema OTP predstavljaju predmet budućih istraživanja. U skladu sa matematičkim modelima formiranim u okviru poglavlja 2.3, GCSA algoritam evaluaciju varijantnih tehnoloških procesa, odnosno redosleda izvođenja tipskih zahvata obrade vrši primenom jednačine (18) pomoću koje se vrši minimiziranje ukupnih troškova obrade razmatranog dela.

5.1.2.3 Mehanizam najbližeg resursa

Nakon osnovnih koraka CSA u okviru GCSA metode, elementi u vektorima položaja vrane jedinki ne predstavljaju pozitivne celobrojne vrednosti i stoga je potrebno u narednom koraku izvršiti konverziju realnih vrednosti u najbliže cele brojeve koji odgovaraju kandidatima mašina, alata i SPA. Pri tome, da bi se usvojila adekvatna vrednost, proveravaju se raspoložive varijante iz skupova $Mašine\{ \}$, $Alati\{ \}$, odnosno $SPA\{ \}$, respektivno za sve kandidate. U ovoj modifikovanoj metodi predlaže se tzv. mehanizam najbližih resursa pomoću kojeg se vrši postupak zaokruživanja gore pomenutih vrednosti na najbliže cele brojeve i na taj način je novogenerisani položaj vrane jedinke spreman za sledeću fazu pretraživanja primenom GCSA. Pored vektora mašina i vektora alata, vektor SPA takođe je predstavljen pomoću celobrojnih promenljivih vrednosti od kojih svaki broj odgovara datom smeru prilaza koji se označavaju slovnim oznakama sa pozitivnim ili negativnim predznakom. Konkretno, za šest smerova prilaza $\{+x,-x,+y,-y,+z,-z\}$ koriste se sledeće celobrojne vrednosti: $\{1,2,3,4,5,6\}$. Analogno ovom postupku, više celobrojnih vrednosti usvaja se kod onih eksperimenata koji razmatraju primere sa većim brojem varijantnih SPA. U algoritmu 2 dat je prikaz pseudokoda predloženog mehanizma najbližeg resursa sa najvažnijim koracima. U pitanju je samo deo koda koji se odnosi na vektor mašina imajući u vidu da je postupak svođenja na celobrojne vrednosti identičan za vektore alata i SPA. U prilogu 2 može se videti izvorni kod mehanizma najbližeg resursa, odnosno mehanizma najbliže mašine napisanog u programskom jeziku Matlab.

Algoritam 2. Pseudokod mehanizma najbliže mašine

```

Mašine{zi} – skup svih kandidata mašina za zahvat zi
trenutnaMašina – trenutno usvojeni kandidat mašine
trenutniZahvat – trenutno usvojeni zahvat obrade
razlikaSort = |Mašine{trenutniZahvat} – trenutnaMašina|
ponoviti za i = 1: ukupan broj zahvata u vektoru redosleda zahvata
  proveri uslov trenutnaMašina > Mašine{ trenutniZahvat }
  ponoviti za j = 1: razlikaSort
    proveri uslov ako skup Mašine{trenutniZahvat} sadrži
      |trenutnaMašina – razlikaSort (j)|
      usvoji | trenutnaMašina – razlikaSort (j)| kao
      kandidat mašine;
    u suprotnom
      nastavi petlju;
    završi uslov
  završi petlju
u drugom slučaju trenutnaMašina < Mašine{ trenutniZahvat }
  ponoviti za j = 1: razlikaSort
    proveri uslov ako skup Mašine{trenutniZahvat} sadrži
      |trenutnaMašina + razlikaSort (j)|
      usvoji | trenutnaMašina + razlikaSort (j)| kao
      kandidat mašine;
    u suprotnom
      nastavi petlju;
    završi uslov
  završi petlju
u suprotnom
  ponoviti za j = 1: razlikaSort
    proveri uslov ako skup Mašine{trenutniZahvat} sadrži
      |trenutnaMašina – razlikaSort (j)|
      usvoji | trenutnaMašina – razlikaSort (j)| kao
      kandidat mašine;
    u suprotnom ako skup Mašine{trenutniZahvat} sadrži
      |trenutnaMašina + razlikaSort (j)|
      usvoji | trenutnaMašina + razlikaSort (j)| kao
      kandidat mašine;
    završi uslov
  završi petlju
završi uslov
završi petlju

```

5.1.2.4 Genetske komponente GCSA algoritma

Prilikom generisanja novih generacija jedinki, selekcija, ukrštanje i mutacija predstavljaju ključne operatore genetskih algoritama. Oponašanjem procesa prirodne selekcije omogućavaju evoluciju jedinki kroz određeni broj generacija. U nastavku su opisani glavni koraci odabranih genetskih operatora u okviru GCSA za rešavanje OTP problema.

Selekcija. Prvi korak u ovom procesu pripada selekciji. Za predloženi GCSA algoritam usvojen je operator turnirske selekcije. Ovaj operator je zasnovan na principu slučajnog selektovanja određenog broja vrana jedinki iz jata koje će potom postati učesnici u tzv. turnirima na kojima pobeđuje ona vrana jedinka koja ima najbolju vrednost funkcije cilja, odnosno najnižu vrednost troškova ili vremena. Odabrana vrana jedinka postaje deo nove

generacije jata koje se sastoji od najboljih vrana jedinki, pobednika u okviru turnira. Postupak turnirske selekcije sa vranama jedinkama ilustrovan je na slici 5.4.

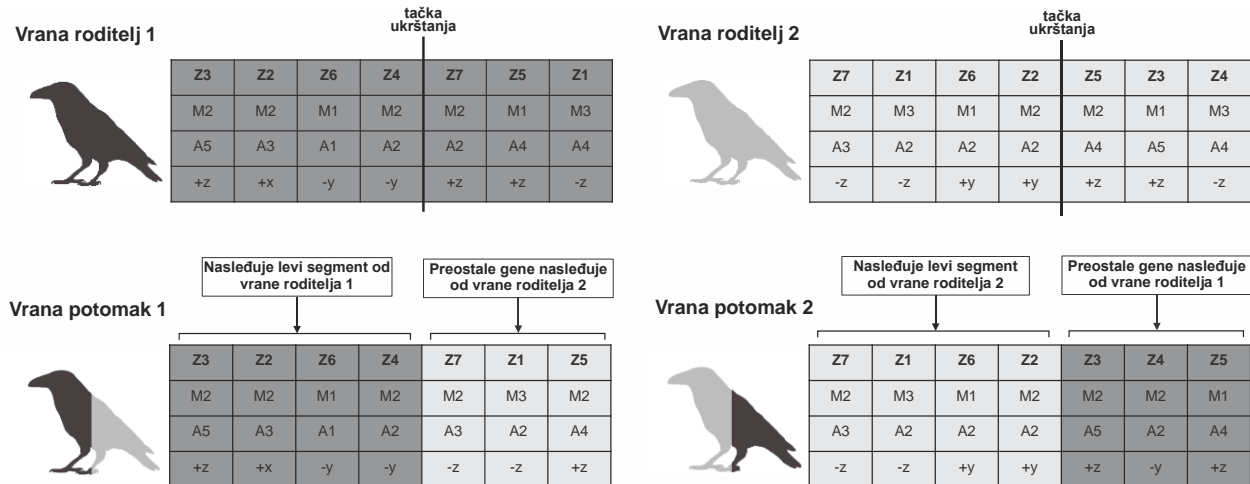


Slika 5.4. Turnirska selekcija sa vranama jedinkama

Ukrštanje. Po završetku selekcije kandidata, a na osnovu definisane verovatnoće p_c , sledi ukrštanje čija je uloga da ostvari razmenu genetske informacije između dve vrane jedinke iz jata. Za GCSA usvojen je 3SX operator ukrštanja koji je preuzet iz [142]. 3SX operator manipuliše isključivo sa vektorima mašina, alata i SPA, ostavljajući pri tom vektor redosleda zahvata neizmenjenim čime ne narušava izvodljivost tehnološkog procesa, odnosno redosleda izvođenja zahvata obrade. Osnovni koraci 3SX operater ukrštanja opisani su na sledeći način:

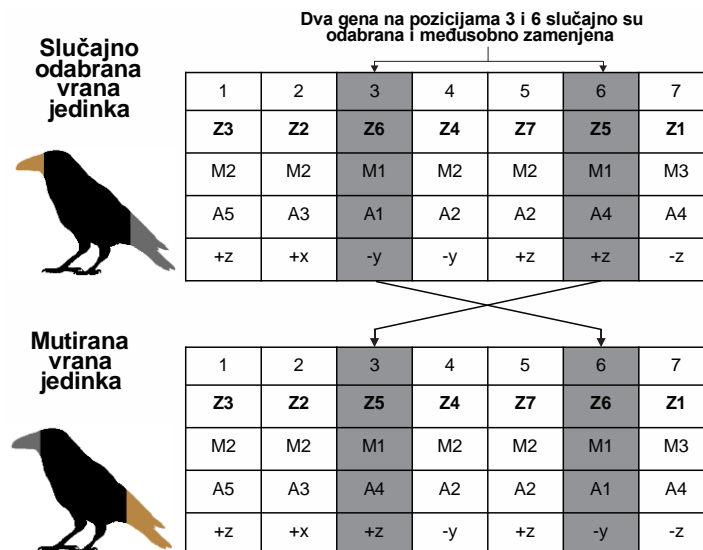
- Dve slučajno odabrane vrane jedinke iz novog jata usvajaju se kao vrane roditelji;
- Slučajnim odabirom tačke ukrštanja vrane roditelji su podeljeni u dva segmenta kako bi se proizvele dve vrane potomci;
- Vektori redosleda vrana potomaka identični su vektorima redosleda vrana roditelja;
- Resursi (mašine, alati i SPA) iz levog segmenta vrane roditelja 1 prenose se u iste pozicije vrane potomka 1; Nakon toga, preostale pozicije u desnom segmentu vrane potomka 1 bivaju popunjene resursima koji pripadaju istim zahvatima u vektorima vrane roditelja 2;
- Analogno prethodnom koraku, resursi iz levog segmenta vrane roditelja 2 prenose se u iste pozicije vrane potomka 2; Preostale pozicije u desnom segmentu vrane potomka 2 popunjavaju se resursima koji pripadaju istim zahvatima u vektorima vrane roditelja 1;

Na slici 5.5 ilustrovan je 3SX operator ukrštanja primenjen na vranu jedinku sa sedam elemenata u vektoru položaja i slučajno odabranom tačkom ukrštanja između zahvata Z4 i Z7 koji se nalaze na četvrtoj i petoj poziciji u redosledu. Kao što se može uočiti, vrane potomci 1 i 2 nasleđuju levi segment do tačke ukrštanja redom od vrana roditelja 1 i 2. Sa druge strane, desni segmenti od tačke ukrštanja bivaju ažurirani unakrsno pri čemu vrana potomak 1 dobija resurse koji odgovaraju identičnim zahvatima vrane roditelja 2, a vrana potomak 2 dobija resurse koji odgovaraju istim zahvatima vrane roditelja 1. Izvorni kod ovog operatora ukrštanja nalazi se u prilogu 4.



Slika 5.5. 3SX operator ukrštanja

Mutacija. Strategije mutiranja jedinki donose završne izmene u genetskom procesu formiranja novih i sposobnijih jedinki. Njihov cilj je da načine dodatna poboljšanja u diverzifikaciji i izbegnu prevremenu konvergenciju algoritma. U GCSA algoritam inkorporirana su dva operatora mutacije koji se upošljavaju prema prethodno definisanoj verovatnoći mutacije p_m . Prvi operator mutacije je mutacija zamene (eng. *shift mutation*) čiji je postupak sa dve vrane jedinke ilustrovan na slici 5.6. Jedna vrana jedinka slučajno se selektuje iz nove generacije jata vrana i dva gena koji reprezentuju dva tipska zahvata obrade slučajno su izabrana za zamenu. Nakon zamene gena, pojedina ograničenja prethođenja mogu biti narušena što može negativno uticati na izvodljivost redosleda zahvata, odnosno tehnološkog procesa. Da bi se takav slučaj izbegao primenjuje se mehanizam popravke kojim se vrane jedinke usaglašavaju sa ograničenjima prethođenja. Mehanizam popravke jedinki baziran je na sličnom principu kao heuristika za rukovanje ograničenjima koja se može videti u radu [12]. Isti mehanizam primenjuje se nakon faze inicijalizacije GCSA kada je jato vrana nasumično generisano. Mehanizam popravke vrana jedinki detaljno je opisan u narednom odeljku.

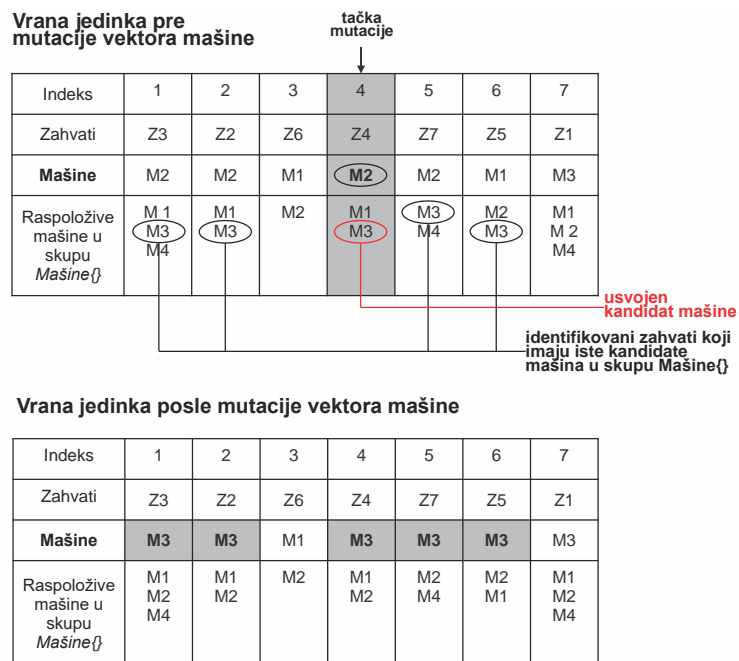


Slika 5.6. Operator mutacije zamene

Drugi operator mutacije predstavlja tzv. mutacija resursa koja, analogno 3SX operatoru ukrštanja, manipuliše vektorima mašina, alata i SPA. Izvodljivost redosleda zahvata se prema tome ne dovodi u pitanje. Koraci operatora mutacije mašina su sledeći:

- *Slučajna selekcija jedne vrane jedinke;*
- *Slučajna selekcija tačke mutacije, odnosno položaja gena vrane jedinke (tipskog zahvata);*
- *Provera raspoloživih kandidata mašina u skupu $Mašine\{\}$ za posmatrani zahvat;*
- *Slučajna selekcija mašine alatke iz skupa $Mašine\{\}$ za posmatrani zahvat kao trenutnu mašinu;*
- *Pronaći druge zahvate u redosledu kod kojih u skupovima $Mašine\{\}$ postoji trenutno odabrana mašina iz prethodnog koraka kao alternativni kandidat;*
- *Dodeliti identifikovane alternativne kandidate mašina koji odgovaraju trenutnoj mašini alatki svim zahvatima.*

Koraci mutacije resursa za kandidate alata i kandidate SPA vrše se na identičan način pri čemu se podaci o kandidatima preuzimaju iz odgovarajućih skupova $Alati\{\}$ i $SPA\{\}$. Postupak mutacije vektora mašina detaljno je ilustrovan na slici 5.7. Izvorni kod ovog operatora nalazi se u prilogu 5.



Slika 5.7. Operator mutacije vektora mašine (mutacija resursa)

Prema tvrdnji autora [70], fiksne vrednosti verovatnoća ukrštanja i mutacije pokazuju se manje efikasnim u poređenju sa prilagodljivim verovatnoćama čije vrednosti variraju u skladu sa vrednostima izračunatih funkcija cilja svih jedinki u populaciji. Kako bi se dodatno izvršilo poboljšanje predloženog GCSA pristupa, u sam koncept usvojene su adaptivne verovatnoće ukrštanja i mutacije koje se u svakoj iteraciji ažuriraju pomoću sledećih matematičkih izraza:

$$p_c = k_{c1} \cdot \frac{f_{max} - f_i}{(f_{max} - f_{avg})} + k_{c2} \quad (31)$$

$$p_m = k_{m1} \cdot \frac{f_{max} - f_i}{(f_{max} - f_{avg})} + k_{m2} \quad (32)$$

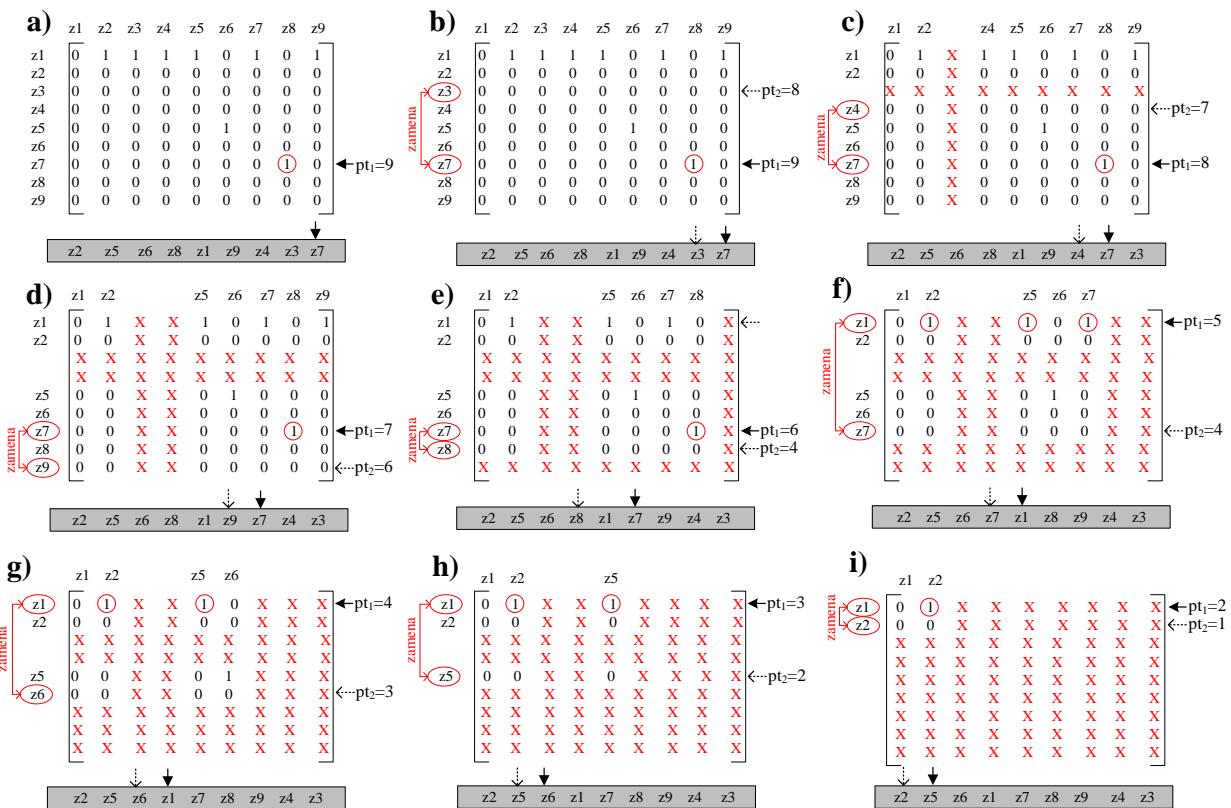
gde su f_{max} i f_{avg} maksimalna i prosečna vrednost funkcije cilja svih vrana jedinki u jatu, f_i je vrednost funkcije cilja trenutne vrane jedinke, dok su k_{c1} , k_{c2} , k_{m1} i k_{m2} specifični koeficijenti verovatnoće za operatore ukrštanja i mutacije respektivno.

5.1.2.5 Mehanizam popravke jedinki

Pomoću informacija o ograničenjima prethođenja i grafovima prethođenja, slučajno generisan redosled zahvata koji reprezentuje jedno potencijalno rešenje ima sledeću formu: [z₂, z₅, z₆, z₈, z₁, z₉, z₄, z₃, z₇]. Da bi se inicijalizovao tehnološki proces, vrši se slučajna selekcija varijantne mašine, alata i SPA na osnovu čega se formiraju redosledi tipskih zahvata obrade. U ovom odeljku više pažnje će biti posvećeno redosledu izvođenja zahvata, *Zahvat[i]*, koji ima ključnu ulogu prilikom eliminacije neizvodljivih rešenja. GCSA algoritam nakon inicijalizacije slučajnog jata vrana u početnoj fazi i nakon primene mutacije zamene u kasnijoj fazi implementacije, generiše određene tehnološke procese koji narušavaju ograničenja prethođenja i pri tom formiraju neizvodljive redoslede zahvata. Sa matricom prethođenja kao ulaznom informacijom (*Matrica* u programskom okruženju), razvijen je algoritam popravke rešenja. Koncept je zasnovan na primeru heuristike iz [12] sa minornim modifikacijama. Prema slici 5.8 može se zaključiti da matrica prethođenja igra važnu ulogu u transformisanju neizvodljivih u izvodljive redoslede zahvata. Logička procedura algoritma popravke rešenja sastoji se iz sledećih koraka:

- 1) Vektor *Zahvat[i]* se koristi za predstavljanje jednog redosleda zahvata obrade;
- 2) Generisanje promenljive *pt1* ($pt1 = n$) koja predstavlja indikator indeksa vektora *Zahvat[i]*;
- 3) Ponavljanje procedure unutar petlje sve dok se ne dostigne da je $pt1 = 0$;
- 4) Identifikovati zahvat obrade u tački $pt1 = n$;
- 5) Za identifikovani zahvat obrade izračunati sumu svih brojnih vrednosti u datom redu matrice prethođenja $\sum_{i=1}^n (PR_{ij})$;
 - I. Ako je $\sum_{i=1}^n (PR_{ij}) = 0$
 - a) Ostaviti zahvat obrade na poziciji *pt1*;
 - b) isključiti red i kolonu matrice „*Matrica*“ koji odgovaraju vrednosti zahvata obrade na poziciji *pt1*; umesto brisanja reda i kolone dodeliti neodređenu vrednost „NaN“ (nenumerička vrednost, eng. *Not a number*);
 - c) inicijalizovati $n = n-1$; pomeriti pointer *pt1* za jedno mesto ulevo;
 - II. If $\sum_{i=1}^n (PR_{ij}) \neq 0$
 - a) inicijalizovati $pt2 = pt1-1$;
 - b) identifikovati zahvat obrade na poziciji *pt2*;

- c) izračunati sumu $\sum_{i=1}^n (PR_{ij})$ svih brojnih vrednosti u odgovarajućem redu koji odgovara identifikovanom zahvatu obrade na poziciji $pt2$;
- d) Ponoviti naredne korake dok je $pt2 > 0$;
- e) Ako je $\sum_{i=1}^n (PR_{ij}) = 0$
- Zameniti zahvat obrade koji se nalazi na poziciji $pt1$ zahvatom obrade koji se nalazi na poziciji $pt2$;
 - Isključiti red i kolonu *Matrice* koji odgovaraju identifikacionom broju zahvata obrade na poziciji $pt2$;
 - Prekinuti petlju i nastaviti od koraka 3);
- f) Ako je $\sum_{i=1}^n (PR_{ij}) \neq 0$
- inicijalizovati $pt2 = pt1 - 1$;
 - preći na korak d);
- 6) vratiti se na korak 3) i ažurirati indikator indeksa $pt1 = pt1 - 1$;
- 7) Resetovati *Matricu* i primeniti iste korake za narednu jedinku.



Slika 5.8. Logička procedura algoritma popravke rešenja

Uzmimo npr. da je polazna pozicija indikatora pt_1 prema slici 5.8a poslednji indeksirani zahvat u vektoru *Zahvat[i]* koji sadrži ukupno 9 zahvata obrade. Indikator $pt_1=9$ pokazuje da je na toj poziciji u redosledu indeksiran zahvat obrade z7. Prema koraku 5 logičke procedure algoritma popravke rešenja, suma svih numeričkih vrednosti u redu

matrice prethođenja (red z_7) nije jednaka nuli. U skladu sa time, prema pravilu $\sum_{i=1}^n (PR_{ij}) \neq 0$ sledi da je naredni korak generisanje indikatora $pt_2 = pt_1 - 1$ (uslov II, korak a, slika 5.8b). Nakon provere istog uslova sa $\sum_{i=1}^n (PR_{ij})$, može se uočiti da indikator pt_2 ukazuje na zahvat z_3 koji je indeksiran na toj poziciji u redosledu. Uslov $\sum_{i=1}^n (PR_{ij}) = 0$ (uslov II, korak e) podrazumeva da zahvat obrade z_7 na poziciji pt_1 i zahvat z_3 na poziciji pt_2 međusobno zamenjuju mesta. Na taj način, kako je zahvat obrade promenio položaj, automatski se isključuje iz dalje logičke procedure dodeljivanjem nenumeričke vrednosti koja je na slici 5.8c označena sa „X“. Ono što sledi je ažuriranje indikatora pt_2 ($pt_2 = 9 - 1 = 8$) i postupak se ponavlja od koraka 3. Prateći logičku proceduru može se videti dalji tok popravke rešenja prema slici 5.8d do 5.8i. Na kraju ovog heurističkog postupka dobija se redosled zahvata koji ne narušava nijedno pravilo prethođenja. Izvorni kod ovog mehanizma napisanog u programskom jeziku Matlab nalazi se u prilogu 3.

5.2 Modifikovani algoritam sivog vuka (mGWO) u optimizaciji tehnoloških procesa obrade

5.2.1 Algoritam sivog vuka (GWO)

Algoritam sivog vuka pripada novijoj generaciji biološki inspirisanih algoritama optimizacije. Svrstava se u grupu metaheuristika inspirisanih inteligencijom rojeva s obzirom da je baziran na inteligenciji grupnog ponašanja vukova kao delova zajednice, odnosno čopora. *Mirjalili* [88] je zaslužan za razvoj ovog algoritamskog rešenja koji je do sada pronašao široku primenu u jednokriterijumskoj, a i višekriterijumskoj optimizaciji pri čemu je implementiran u velikom broju programskih jezika.

Sivi vukovi (lat. *Canis lupus*) pripadaju porodici pasa koji žive i love u čoporima. Istaknuti su predatori što ih stavlja u sam vrh lanaca ishrane u životinjskom svetu. Čopor može sačinjavati do 12 članova među kojima se izdvajaju mužjak i ženka i njihovo potomstvo. Prednosti socijalnog života sivih vukova prvenstveno se odlikuju zajedničkim lovom na plen, grupnoj nezi o mladuncima i teritorijalnoj odbrani u slučaju opasnosti od drugih sila [169]. Udruženim snagama razvili su sposobnost prilagođavanja lovu na krupniju divljač gde prvenstveno preovladavaju kopitari kao što su bizoni ili jeleni. Jedna od glavnih karakteristika zajednice sivih vukova, a ujedno i jedan od izvora inspiracije za razvoj algoritma, je strogo uspostavljen hijerarhijski sistem koji vlada među članovima čopora. Od najdominantnijih do najpodređenijih vukova redom se izdvajaju: alfe, bete, delte i omega [88].

Alfa mužjak i alfa ženka predstavljaju lidere čopora koji su najodgovorniji za donošenje glavnih odluka koje se tiču lova, spavanja, buđenja, itd. Pored toga što alfa vukovi predstavljaju najdominantnije pojedince, interesantno je da u zajednici vlada određena vrsta demokratije koja dozvoljava da se lideri mogu nalaziti u poziciji kada slušaju druge vukove. Sama činjenica da alfe nisu nužno i fizički najsnažnije jedinke u čoporu potvrđuje koliki se značaj pridaje disciplini i ogranizaciji grupe pre nego njenim fizičkim atributima.

Beta vukovi u hijerarhijskom sistemu nalaze se na nivou ispod lidera alfi. Takođe mogu biti mužjaci i ženke kao i njihovi nadređeni. Njihova uloga je da budu savetnici alfama i pomažu im oko donošenja važnih odluka za održanje discipline i opstanak čopora. Bete su podređeni alfama ali su u poziciji da komanduju ostalim vukovima koji se nalaze hijerarhijski ispod njih. One predstavljaju glavne kandidate da zamene alfa vukove u slučaju da neki od njih nastrada ili postane previše star.

Omega vukovi se nalaze na najnižem hijerarhijskom nivou u čoporu. S obzirom na podređeni položaj u odnosu na ostale vukove, omega su poslednje u redu za dobijanje hrane. Njihova uloga je budu žrtve, a u nekim slučajevima i čuvari mladunaca. Omega su važne za samu zajednicu jer gubitkom omega vukova izbijaju unutrašnji sukobi između ostalih članova. Zahvaljujući njima održava se dominantna struktura u celom čoporu.

Delta vukovi pripadaju posebnom nivou u zajednici koji se nalazi između beta i omega vukova. Imaju zadatak da slušaju naređenja alfa i beta vukova, dok su u mogućnosti da komanduju omegama. U delta vukove ubrajaju se izviđači, čuvari, stariji vukovi, lovci i negovatelji. Izviđačima pripada uloga nadgledanja teritorije i upozoravanja čopora u slučajevima opasnosti. Čuvari su zaduženi da štite i održavaju bezbednost u okviru čopora. Stariji vukovi koji su nekada bili dominantni pomažu čoporu svojim stečenim iskustvom. Na kraju, lovci pomažu alfama i betama u lovu na plen i snabdevaju ostale članove ulovljenom hranom.

Pored opisanog hijerarhijskog sistema koji je zastupljen u zajednici sivih vukova, druga elementarna karakteristika koja je pomogla razvoju GWO algoritma odnosi se na ponašanje sivih vukova tokom lova na plen. Prema istraživanju [170] u kojem je preispitan uticaj veličine čopora na uspeh u lovu na bizona, ponašanje vukova tokom lova odlikuje prelaz od faze prilaženja i napadanja pa sve do hvatanja plena. Autori *Muro i sar.* [169] razvili su računarsku simulaciju multi-agent sistema u kojem je ponašanje agenata definisano jednostavnim pravilima na bazi grupnog lova sivih vukova. Na slici 5.9 može se videti metod kojim se služe sivi vukovi prilikom lova na bizona. Metod lova može se sumirati u tri osnovne faze:

- Faza prilaženja i jurnjave,
- Faza opkoljavanja i maltretiranja plena i
- Faza napada.

U skladu sa društvenom hijerarhijom i fazama grupnog lova sivih vukova razvijen je matematički model GWO algoritma koji je detaljnije opisan u narednom odeljku.



Slika 5.9. Metod lova sivih vukova na bizona: od (A) prilaženja, gonjenja i praćenja, (B-D) maltretiranja i opkoljavanja plena do (E) stacionarne faze i finalnog napada [169]

5.2.1.1 Matematički model GWO algoritma

U skladu sa društvenom hijerarhijom zajednice sivih vukova opisanom u početnom delu ovog poglavlja, alfa vuk (α) reprezentuje najbolje rešenje u populaciji. Analogno tome, drugo i treće najbolje rešenje u populaciji jedinki tokom iterativnog procesa pretraživanja predstavljaju beta (β) i delta (δ) vuk. Sve ostale jedinke se smatraju omegama (ω). Pretraživanje je zasnovano na položajima tri najbolja vuka koji, kao najbolja rešenja u celoj populaciji, procenjuju položaj plena (globalnog optimuma).

Prvi mehanizam koji je matematički modeliran je mehanizam opkoljavanja plena koji se definiše preko sledećih izraza:

$$\vec{D} = |\vec{C} \cdot \vec{X}_p(t) - \vec{X}(t)| \quad (33)$$

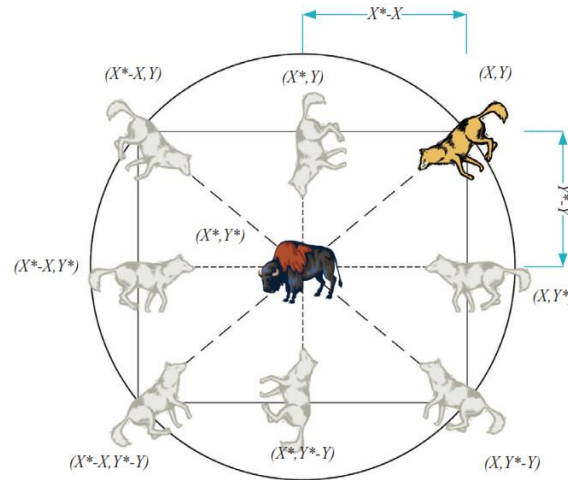
$$\vec{X}(t+1) = \vec{X}_p(t) - \vec{A} \cdot \vec{D} \quad (34)$$

$$\vec{A} = 2\vec{a}\vec{r}_1 - \vec{a} \quad (35)$$

$$\vec{C} = 2\vec{r}_2 \quad (36)$$

gde t predstavlja trenutnu iteraciju algoritma, dok su \vec{A} i \vec{C} koeficijent vektori koji definisani na osnovu slučajnih brojeva \vec{r}_1 i \vec{r}_2 koji se kreću između 0 i 1. \vec{X} je vektor položaja sivog vuka, a \vec{X}_p je vektor položaja plena. Vektor \vec{a} predstavlja veličinu koja linearno opada od 2 do 0.

Na slici 5.10 može se videti ilustrirani prikaz ažuriranja položaja sivog vuka (X, Y) i plena (X^*, Y^*) na osnovu dvodimenzionalnog vektora položaja. Podešavanjem koeficijent vektora \vec{A} i \vec{C} mogu se postići različiti položaji oko najboljeg rešenja.



Slika 5.10. Dvodimenzionalni vektori položaja i njihove potencijalne lokacije

Da bi se modelirao postupak lova, glavnu ulogu preuzimaju najbolji vukovi u populaciji: alfa (α), beta (β) i delta (δ) vuk. S obzirom da se radi o stohastičkoj prirodi GWO algoritma i kompleksnosti problema optimizacije koji se rešava, ne postoji saznanje o tome gde se optimalno rešenje, odnosno plen, nalazi. Polaznjem od pretpostavke da alfa (α), beta (β) i delta (δ) vukovi respektivno poseduju najviše znanja o potencijalnom položaju plena, ova tri rešenja smatraju se najboljim rešenjima koje populacija ostvaruje tokom iterativnog procesa. Ostali vukovi, omega (ω), ažuriraju svoje položaje u skladu sa položajima najboljih vukova u čoporu. Naredne tri jednačine formulišu ovu pretpostavku:

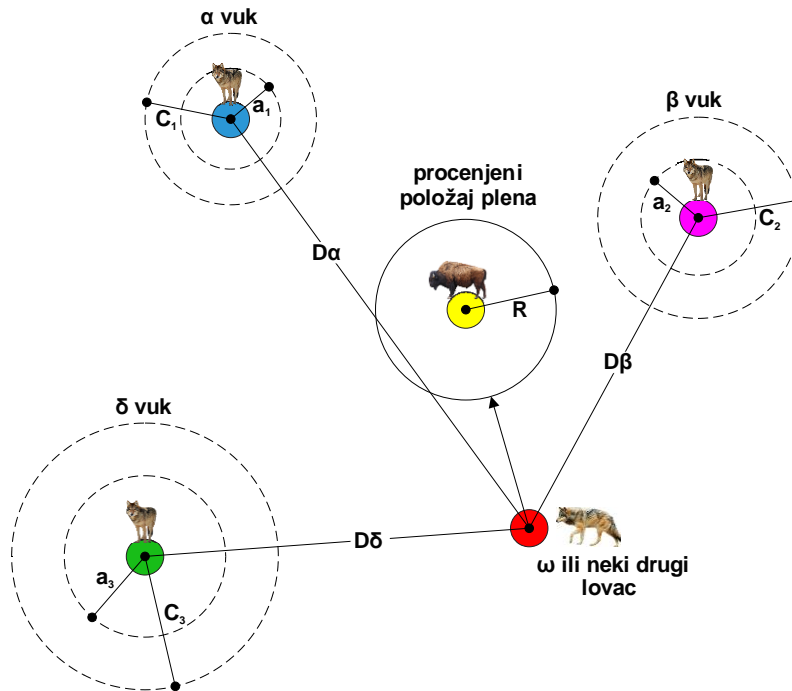
$$\vec{D}_\alpha = |\vec{C}_1 \cdot \vec{X}_\alpha - \vec{X}|, \quad \vec{D}_\beta = |\vec{C}_2 \cdot \vec{X}_\beta - \vec{X}|, \quad \vec{D}_\delta = |\vec{C}_3 \cdot \vec{X}_\delta - \vec{X}| \quad (37)$$

$$\vec{X}_1 = \vec{X}_\alpha - \vec{A}_1 \cdot (\vec{D}_\alpha), \quad \vec{X}_2 = \vec{X}_\beta - \vec{A}_2 \cdot (\vec{D}_\beta), \quad \vec{X}_3 = \vec{X}_\delta - \vec{A}_3 \cdot (\vec{D}_\delta) \quad (38)$$

$$\vec{X}(t+1) = \frac{\vec{X}_1 + \vec{X}_2 + \vec{X}_3}{3} \quad (39)$$

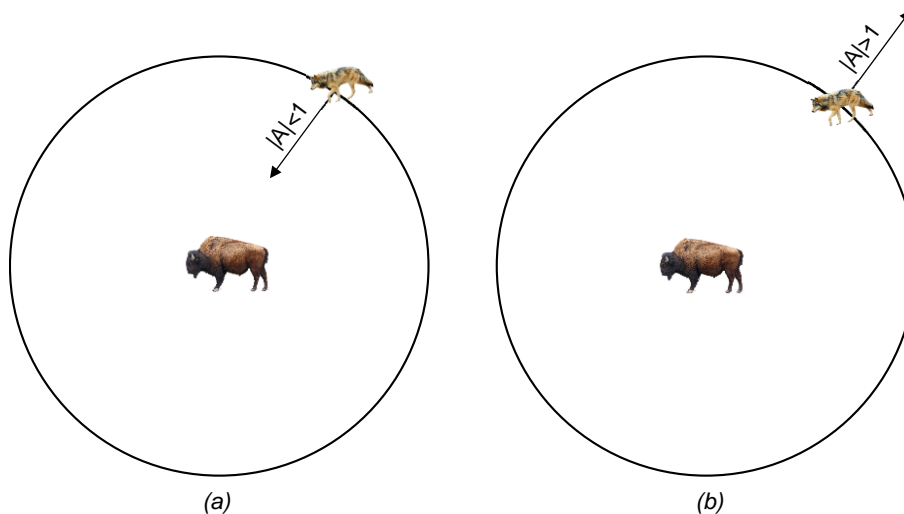
gde \vec{X} predstavlja položaj sivog vuka koji se kroz iteracije t ažurira prema rastojanjima od tri najbolja vuka, alfe, \vec{D}_α , bete, \vec{D}_β i delte \vec{D}_δ .

Postupak ažuriranja položaja u okviru GWO algoritma može se grafički ilustrovati u dvodimenzionalnom prostoru kao što je prikazano na slici 5.11. Omega ili vuk koji je u ulozi lovca kreće se u prostoru pretraživanja pri čemu svoj položaj bira na osnovu koeficijent vektora i rastojanja od tri najbolja vuka: alfa (α), beta (β) i delta (δ).



Slika 5.11. Ažuriranje položaja prema alfa, beta i delta vukovima

Intenzifikacija i diverzifikacija GWO ostvaruju se manipulisanjem koeficijent vektora \vec{A} čiji interval zavisi od linearnog smanjenja vektora \vec{a} . Tokom iteracija, vektor \vec{a} teži da umanji koeficijent vektor \vec{A} što direktno utiče na eksploatacione, odnosno istraživačke sposobnosti algoritma. Ukoliko je apsolutna vrednost koeficijent vektora \vec{A} manja od 1, tada se vukovi usmeravaju ka plenu i time se ističu lokalni kapaciteti, dok sa druge strane, kada je apsolutna vrednost \vec{A} veća od 1, tada u plan dolaze globalni kapaciteti. Drugim rečima, sivi vukovi ažuriraju svoje položaje u odnosu na tri najbolja vuka i na osnovu koeficijent vektora \vec{A} konvergiraju ka plenu, odnosno divergiraju u potrazi za drugim plenom, potencijalno boljim. Slika 5.12 ilustruje ova dva slučaja.



Slika 5.12. Napad na plen - konvergencija (a) i potraga za plenom – divergencija (b)

Divergencija od plena, odnosno istraživanje prostora pretraživanja takođe je naglašeno drugim koeficijent vektorom, \vec{C} , koji se kreće u rasponu od 0 do 2. Njegov zadatak je da nasumično dodeli odgovarajuću težinu plenu do kojeg vukovi žele da dopru. To podrazumeva da vukovima nasumično može biti dodeljen lakši, odnosno teži put do plena. Na ovaj način stohastički se želi pokazati uticaj plena u definisanju rastojanja (jednačina (33)). Kako bi se obezbedili globalni kapaciteti u svakoj iteraciji algoritma i potencijalno izbegli lokalni optimumi, vrednost \vec{C} nije linearno opadajuća, za razliku od vektora \vec{A} .

Pseudokod GWO metaheurističkog algoritma prikazan je u Algoritmu 3. Kao i drugi optimizacioni algoritmi bazirani na populaciji rešenja, GWO proces pretraživanja polazi od inicijalne populacije sivih vukova. Kroz iterativan postupak, tri najbolja vuka, alfa, beta i delta stiču znanje o lokaciji plena. U odnosu na njih, ostali vukovi ažuriraju svoje položaje. U međuvremenu na osnovu koeficijent vektora \vec{A} i \vec{C} omogućava se uravnoteženo pretraživanje prostora potencijalnih rešenja stavljanjem akcenta kako na lokalno tako i globalno pretraživanje. Najbolji vukovi ažuriraju svoje položaje u svakoj iteraciji algoritma i nakon predviđenog broja iteracija dolazi do zaustavljanja.

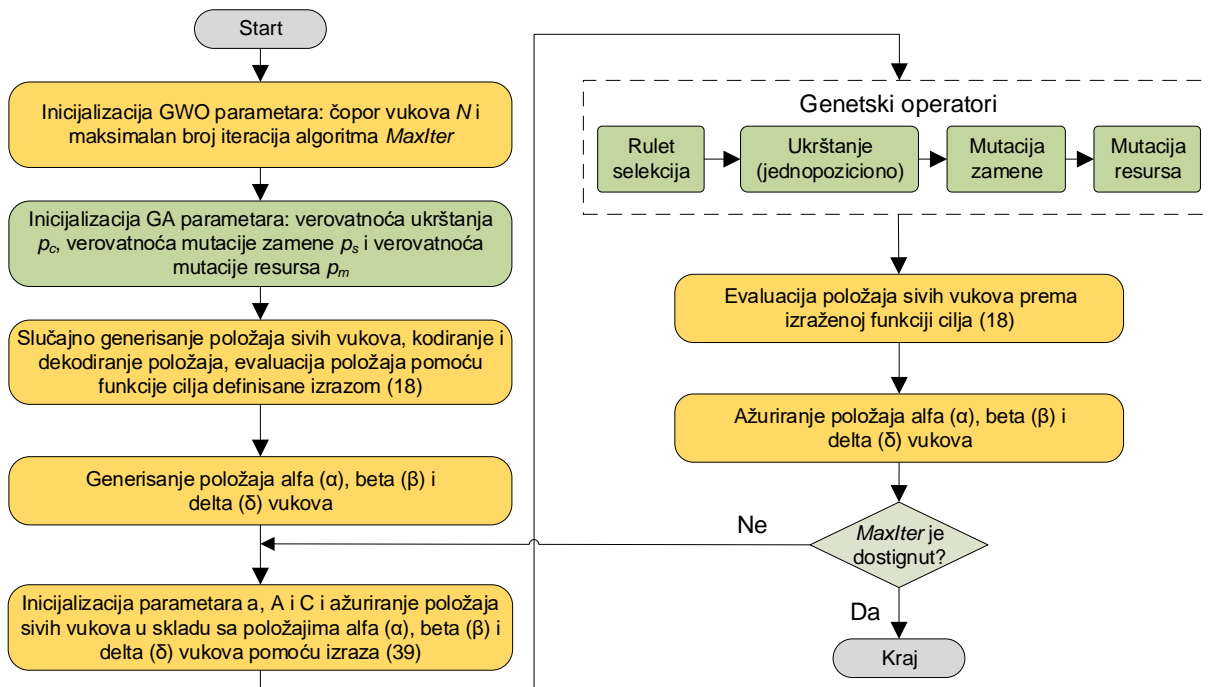
Algoritam 3. Pseudokod tradicionalnog GWO algoritma

<p>1. Početak: Inicijalizacija Inicijalizovati parametre GWO: koeficijente a, A i C; Inicijalizovati početnu populaciju (čopor) sivih vukova; Odrediti vrednost funkcije cilja svake jedinice (vuka) u početnoj populaciji (čoporu); Generisati najbolje početno rešenje/položaj vuka u čoporu - alfa; Generisati drugo najbolje početno rešenje/položaj vuka u čoporu - beta; Generisati treće najbolje početno rešenje/položaj vuka u čoporu - delta;</p>
<p>2. Ponavljanje: Iterativni postupak Ponoviti iteracija = iteracija + 1; ponoviti za sve vukove u čoporu Ažurirati položaj postojećeg vuka u čoporu pomoću izraza (39) završi petlju Ažurirati koeficijente a, A i C; proveriti vrednost funkcije cilja svakog vuka u čoporu; ažurirati tri najbolja položaja vukova – alfa, beta i delta; završi petlju</p>
<p>3. Izlazni rezultati</p>

5.2.2 Modifikovani algoritam sivog vuka (mGWO)

Drugi predloženi metaheuristički algoritam za optimizaciju tehnoloških procesa obrade proizvoda u ovom istraživanju je modifikacija tradicionalnog GWO algoritma. Za razliku od prethodnog GCSA, modifikovani GWO primenjen je u petom i šestom eksperimentu u okviru kojih je izvršena minimizacija ukupnog vremena obrade rotacionih delova. Radi izbegavanja lokalnih optimuma, tradicionalni GWO unapređen je ugradnjom genetskih operatora, selekcijom, ukrštanjem i mutacijom.

Dijagram toka mGWO algoritma dat je na slici 5.13. Prvi korak mGWO prema dijagramu toka predstavlja definisanje ulaznih parametara. U eksperimentalnim istraživanjima u poglavlju broj 6 više pažnje će biti posvećeno ulaznim parametrima, njihovoj optimizaciji i podešavanju. Zatim sledi inicijalizacija početne populacije, odnosno čopora sivih vukova pri čemu se predstavljanje tehnoloških procesa u formi sivih vukova i plena vrši putem vektora koji tehnološke informacije baziraju prema I/ILI mreži. Ovaj postupak zahteva fazu kodiranja i dekodiranja rešenja kojima je više rečeno u delu poglavlja 5.2.1.1. Radi manipulacije vektorima u Matlab programskom okruženju razvijeni su vektori položaja mašina, alata i SPA u skladu sa osnovnim jednačinama (36) i (38) predstavljenim u 5.2.1.1. Kriterijum za vrednovanje varijantnih tehnoloških procesa je minimalno vreme obrade razmatranog dela koji se optimizuje. Nakon evaluacije rešenja i osnovnih koraka GWO, sledi primena genetskih operatora. Prvi među njima je operator rulet selekcije koji na bazi prethodno definisane verovatnoće selektuje najbolje vukove jedinke iz čopora i priprema ih za naredni korak ukrštanja. Jednopolozicioni operator ukrštanja je usvojen za potrebe unapređenja algoritma koji je takođe primenjen u skladu sa odgovarajućom verovatnoćom. Nakon ukrštanja slede operatori mutacije gde su izdvojeni operator zamene kao i operator resursa. mGWO se obustavlja nakon definisanog broja iteracija pri čemu se na izlazu dobijaju optimalni i/ili blisko optimalni tehnološki procesi.



Slika 5.13. Dijagram toka mGWO algoritma za optimizaciju tehnoloških procesa

5.2.2.1 Kodiranje i dekodiranje rešenja

Evidentna je razlika između prilaza koji koristi GCSA i mGWO algoritam, a koji se prvenstveno ogleda u načinu predstavljanja tehnoloških procesa. U ovom slučaju radi se o I/ILI mrežama koje su opisane u odeljku 2.2.6. Na osnovu informacija prikazanih na mrežama neophodno je izvršiti kodiranje i dekodiranje predstavljenih elemenata varijantnih tehnoloških procesa. U skladu sa time usvojena je procedura kodiranja i

dekodiranja ekvivalentna onoj se može videti u [2]. Primer dela rotacionog oblika za koji je izvršena analiza postupka kodiranja i dekodiranja predstavljen je u okviru šestog eksperimenta u odeljku 6.3.2 na slici 6.25. Šema kodiranja jednog rešenja, odnosno u ovom slučaju jednog agenta pretraživanja koji reprezentuje sivog vuka ili potencijalni plen data je u tabeli 5.2. Kao što se može primetiti, četiri vektora (stringa) deo su i ovog predstavljanja varijantnog tehnološkog procesa. Redom su prikazani vektori zahvata, mašina, alata i SPA. Kada su u pitanju vektori mašina, alata i SPA, svaki element u vektorima podrazumeva varijantnu mašinu, alat i SPA koji su nasumično selektovani iz odgovarajućih baza, $Mašine\{\}$, $Alati\{\}$ i $SPA\{\}$. Poslednji niz vrednosti u tabeli predstavlja redosled vremena obrade za svaki tipski zahvat. Shodno optimizaciji ukupnog vremena obrade razmatranog rotacionog dela razvijena je baza raspoloživih vremena obrade u vremenskim jedinicama, $Vremena\{\}$. U okviru ove baze nalaze se vremena obrade koja odgovaraju svakom potencijalnom tipskom zahvatu. Na taj način formiran je i peti vektor, a to je vektor vremena obrade. Ovaj vektor ne učestvuje u postupku pretraživanja već je namenjen isključivo ažuriranju vremena u skladu sa generisanim tipskim zahvatima.

Dekodiranje agenta pretraživanja obavlja se direktno nakon kodiranja kako bi se generisao izvodljiv tehnološki proces. U tu svrhu koriste se I/ILI konektori pomoću kojih se definiše fleksibilnost tehnoloških procesa (poglavlje 2.2.6). U tabeli 5.2 u poslednjem redu mogu se videti slučajno odabrani I i ILI konektori za primer rotacionog dela u šestoj eksperimentalnoj studiji. U skladu sa tehnološkim informacijama sa mreže koja je data na slici 6.26 u odeljku 6.3.2 i odabrane I/ILI konektore dolazi se do izvodljivog varijantnog tehnološkog procesa koji je prikazan u tabeli 5.3. Kao što se može zaključiti, selektovana putanja $[Z_1, Z_2, Z_6, Z_7, Z_8, Z_9, Z_{10}, Z_{12}, Z_{13}, Z_{14}]$ sa I/ILI mreže predstavlja izvodljiv redosled zahvata obrade usklađen sa ograničenjima formulisanim na samoj mreži.

Tabela 5.2. Šema kodiranja varijantnog tehnološkog procesa za šesti eksperiment

Zahvati	1	2	3	4	5	6	7	8	9	10	11	12	13
Mašine	1	1	8	8	8	1	3	3	1	1	3	4	3
Alati	1	4	6	7	8	6	7	5	1	1	2	11	3
SPA	+z	+z	-z	-z	+z	-z	+z	+z	+z	+z	-z	-y	-z
Vremena	2,3	1	3,8	38,1	13,7	3,3	30,2	1,8	0,6	0,7	5,5	5,7	1,1
Zahvati	14	15	16	17	18	19	20	21	22	23	24	25	
Mašine	4	4	6	4	6	6	6	6	5	5	6	6	
Alati	11	9	9	9	6	7	9	11	10	8	12	12	
SPA	-x	-z	-z	-z	-z	+z	-x	+z	+z	+z	+z	+z	
Vremena	10,8	2,7	7,2	4,3	3,4	24,1	2,9	2,8	3,1	30,2	0,5	0,6	
I konektori: I1=0, I2=1, I3=0, I4=1													
ILI konektori: ILI1=0, ILI2=1, ILI3=1, ILI4=0													

Tabela 5.3. Dekodirani varijantni tehnološki proces iz table 5.2

Zahvati	1	2	6	7	8	9	10	12	13	14
Mašine	1	1	1	3	3	1	1	4	3	4
Alati	1	4	6	7	5	1	1	11	3	11
SPA	5	5	6	5	5	5	5	3	6	2
Vremena	2,3	1	3,3	30,2	1,8	0,6	0,7	5,7	1,1	10,8

5.2.2.2 Matematički model mGWO algoritma

Po analogiji sa prethodnom GCSA metodom, kodiranje varijantnih tehnoloških procesa u okviru mGWO vrši se takođe u matričnom obliku. Formira se ukupno pet matrica tipa $1 \times N$ pri čemu peta matrica podrazumeva vektor vremena obrade koji ne učestvuje u osnovnim operacijama GWO i GA već se ažurira u skladu sa ostalim vektorima. Imajući u vidu da sivi vukovi ažuriraju položaje prema najboljim položajima alfa, beta i delta putem jednačina (37-39), neophodna je njihova modifikacija po uzoru na prethodni modifikovani algoritam kako bi se uspešno realizovala optimizacija tehnoloških procesa. Prvo, rastojanja sivih vukova u odnosu alfa (α), beta (β) i delta (δ) vuka za redom vektore mašina, alata i SPA formulisana su na sledeći način:

$$\overrightarrow{D_{\alpha,m}} = |\vec{C}_1 \cdot \overrightarrow{X_{\alpha,m}} - \overrightarrow{X_m}|, \quad \overrightarrow{D_{\beta,m}} = |\vec{C}_2 \cdot \overrightarrow{X_{\beta,m}} - \overrightarrow{X_m}|, \quad \overrightarrow{D_{\delta,m}} = |\vec{C}_3 \cdot \overrightarrow{X_{\delta,m}} - \overrightarrow{X_m}| \quad (40)$$

$$\overrightarrow{D_{\alpha,a}} = |\vec{C}_1 \cdot \overrightarrow{X_{\alpha,a}} - \overrightarrow{X_a}|, \quad \overrightarrow{D_{\beta,a}} = |\vec{C}_2 \cdot \overrightarrow{X_{\beta,a}} - \overrightarrow{X_a}|, \quad \overrightarrow{D_{\delta,a}} = |\vec{C}_3 \cdot \overrightarrow{X_{\delta,a}} - \overrightarrow{X_a}| \quad (41)$$

$$\begin{aligned} \overrightarrow{D_{\alpha,spa}} &= |\vec{C}_1 \cdot \overrightarrow{X_{\alpha,spa}} - \overrightarrow{X_{spa}}|, \\ \overrightarrow{D_{\beta,spa}} &= |\vec{C}_2 \cdot \overrightarrow{X_{\beta,spa}} - \overrightarrow{X_{spa}}|, \\ \overrightarrow{D_{\delta,spa}} &= |\vec{C}_3 \cdot \overrightarrow{X_{\delta,spa}} - \overrightarrow{X_{spa}}| \end{aligned} \quad (42)$$

gde $\overrightarrow{X_m}$ predstavlja položaj sivog vuka za vektor mašina; $\overrightarrow{X_{\alpha,m}}$, $\overrightarrow{X_{\beta,m}}$ i $\overrightarrow{X_{\delta,m}}$ su položaji alfa (α), beta (β) i delta (δ) vuka za vektor mašina; $\overrightarrow{X_a}$ predstavlja položaj sivog vuka za vektor alata; $\overrightarrow{X_{\alpha,a}}$, $\overrightarrow{X_{\beta,a}}$ i $\overrightarrow{X_{\delta,a}}$ su položaji alfa (α), beta (β) i delta (δ) vuka za vektor alata; $\overrightarrow{X_{spa}}$ predstavlja položaj sivog vuka za vektor SPA; $\overrightarrow{X_{\alpha,spa}}$, $\overrightarrow{X_{\beta,spa}}$ i $\overrightarrow{X_{\delta,spa}}$ su položaji alfa (α), beta (β) i delta (δ) vuka za vektor SPA.

Dalje, položaj sivog vuka u iteraciji $it + 1$ formuliše se na osnovu vektora mašina, alata i SPA a u odnosu na vektore položaja alfa (α), beta (β) i delta (δ) vuka. Može prikazati preko sledećih izraza:

$$\overrightarrow{X_m}(t+1) = \frac{\overrightarrow{X_{1,m}} + \overrightarrow{X_{2,m}} + \overrightarrow{X_{3,m}}}{3} \quad (43)$$

$$\overrightarrow{X_a}(t+1) = \frac{\overrightarrow{X_{1,a}} + \overrightarrow{X_{2,a}} + \overrightarrow{X_{3,a}}}{3} \quad (44)$$

$$\overrightarrow{X_{spa}}(t+1) = \frac{\overrightarrow{X_{1,spa}} + \overrightarrow{X_{2,spa}} + \overrightarrow{X_{3,spa}}}{3} \quad (45)$$

$$\begin{aligned} \overrightarrow{X_{1,m}} &= \overrightarrow{X_{\alpha,m}} - \vec{A}_1 \cdot (\overrightarrow{D_{\alpha,m}}), \\ \overrightarrow{X_{2,m}} &= \overrightarrow{X_{\beta,m}} - \vec{A}_2 \cdot (\overrightarrow{D_{\beta,m}}), \end{aligned} \quad (46)$$

$$\begin{aligned}\overrightarrow{X_{3,m}} &= \overrightarrow{X_{\delta,m}} - \vec{A}_3 \cdot (\overrightarrow{D_{\delta,m}}) \\ \overrightarrow{X_{1,a}} &= \overrightarrow{X_{\alpha,a}} - \vec{A}_1 \cdot (\overrightarrow{D_{\alpha,a}}), \\ \overrightarrow{X_{2,a}} &= \overrightarrow{X_{\beta,a}} - \vec{A}_2 \cdot (\overrightarrow{D_{\beta,a}}), \\ \overrightarrow{X_{3,a}} &= \overrightarrow{X_{\delta,a}} - \vec{A}_3 \cdot (\overrightarrow{D_{\delta,a}})\end{aligned}\tag{47}$$

$$\begin{aligned}\overrightarrow{X_{1,spa}} &= \overrightarrow{X_{\alpha,spa}} - \vec{A}_1 \cdot (\overrightarrow{D_{\alpha,spa}}), \\ \overrightarrow{X_{2,spa}} &= \overrightarrow{X_{\beta,spa}} - \vec{A}_2 \cdot (\overrightarrow{D_{\beta,spa}}), \\ \overrightarrow{X_{3,spa}} &= \overrightarrow{X_{\delta,spa}} - \vec{A}_3 \cdot (\overrightarrow{D_{\delta,spa}})\end{aligned}\tag{48}$$

gde $\overrightarrow{X_m}(t+1)$, $\overrightarrow{X_a}(t+1)$ i $\overrightarrow{X_{spa}}(t+1)$ predstavljaju ažurirane vektore mašina, alata i SPA respektivno za položaj sivog vuka u iteraciji $it+1$; $\overrightarrow{X_{1,m}}$, $\overrightarrow{X_{2,m}}$ i $\overrightarrow{X_{3,m}}$ su ažurirani vektori mašina prema rastojanju u odnosu na vukove alfa (α), beta (β) i delta (δ); $\overrightarrow{X_{1,a}}$, $\overrightarrow{X_{2,a}}$ i $\overrightarrow{X_{3,a}}$ su ažurirani vektori alata prema rastojanju u odnosu na vukove alfa (α), beta (β) i delta (δ); $\overrightarrow{X_{1,spa}}$, $\overrightarrow{X_{2,spa}}$ i $\overrightarrow{X_{3,spa}}$ su ažurirani vektori alata prema rastojanju u odnosu na vukove alfa (α), beta (β) i delta (δ).

Vektor zahvata obrade formulisan je na osnovu generisanih I i ILI konektora. U skladu sa tehnološkim informacijama prezentovanim na I/ILI mreži dobija se optimalna putanja odnosno redosled zahvata obrade.

5.2.2.3 Kriterijum evaluacije varijantnih tehnoloških procesa

Prva četiri eksperimenta u okviru doktorske disertacije naglasak postavljaju na primeni GCSA metaheurističkog pristupa sa vektorskim predstavljanjem, ograničenjima i matricama prethođenja u minimiziranju ukupnih troškova obrade razmatranih proizvoda prizmatičnog oblika. Kada je modifikovani mGWO algoritam u pitanju, verifikacija performansi sprovedena je u petom i šestom eksperimentu gde su razmatrani delovi rotacionog oblika, a za koje je izvršena minimizacija ukupnog vremena obrade delova. U skladu sa tim, jednačina (24) se usvaja za funkciju cilja na osnovu koje se vrši vrednovanje varijantnih tehnoloških procesa.

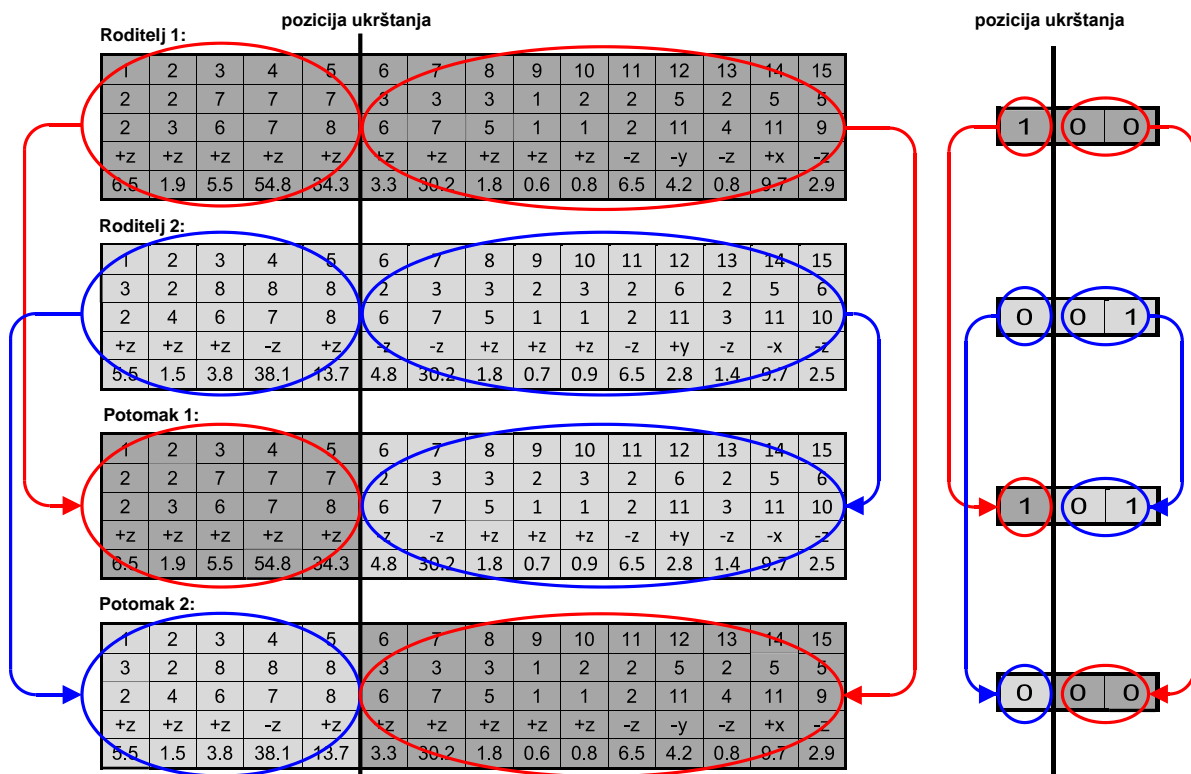
5.2.2.4 Genetske operacije

Zarobljavanje rešenja u prostorima lokalnih optimuma predstavlja jedan od glavnih nedostataka metaheurističkih algoritama [2,93]. I pored odgovarajućih parametara koji teže da povedu pretraživanje ka istraživanju prostora mogućih rešenja, bavljenje složenijim optimizacionim problemima kao što je problem OTP-a zahtevaju razvijanje hibridnih pristupa ili modifikovanje postojećih. Modifikacije se mogu izvršiti uvođenjem dodatnih operatora koji će doprineti poboljšanju istraživačkih i/ili eksploatacionih kapaciteta. Da bi se unapredio tradicionalni GWO algoritam predloženo je uvođenje tri

genetska operatora, slično GCSA algoritmu koji je baziran na sličnom principu. U pitanju su selekcija, ukrštanje i dva operatora mutacije koji su u nastavku detaljnije opisani.

1) **Selekcija**. Posle sprovođenja osnovnih koraka GWO koristeći izraze (40-48), sledi primena operatora genetske selekcije. Usvojena je rulet selekcija koja bira dve jedinke vuka proporcionalno njihovom fitnessu, odnosno funkciji cilja. Oni postaju vukovi roditelji koji prelaze u sledeću fazu, fazu ukrštanja.

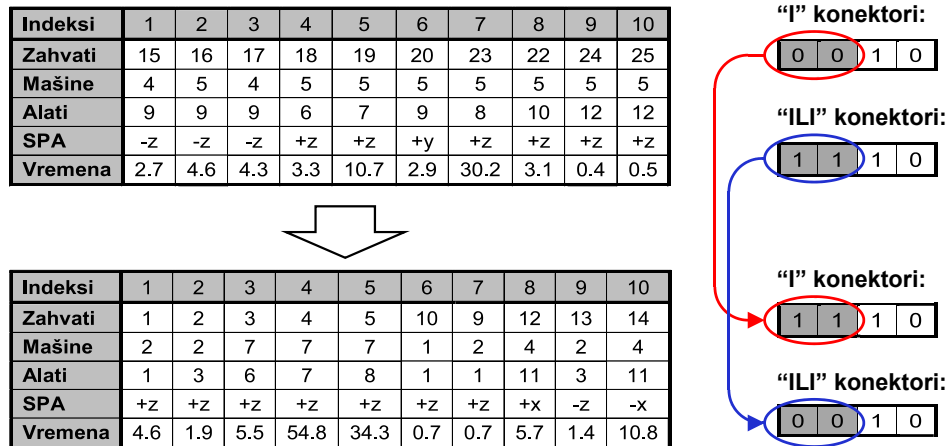
2) **Ukrštanje**. Nakon završetka faze selekcije, na bazi verovatnoće ukrštanja p_c primenjuje se operator jednopozicionog ukrštanja vukova. Cilj ukrštanja je da se izvrši kombinacija genetskog sadržaja dva roditelja na osnovu kojih će se dobiti potencijalno bolji potomci. U tu svrhu grafički je predstavljen operator jednopozicionog ukrštanja koji je dat na slici 5.14 [2]. U pitanju je delimičan prikaz kodiranog tehnološkog procesa iz tabele 5.2. Postupak ukrštanja započinje slučajnim odabirom pozicije ukrštanja između dve tačke što je u ovom primeru između zahvata obrade z_5 i z_6 . Time su prvi i drugi roditelj podeljeni u dva segmenta. Prvi potomak nastaje spajanjem levog segmenta prvog roditelja i desnog segmenta drugog, a drugi potomak inverznom procedurom. Takođe, ukrštanje se vrši za vektore koji sadrže odabrane ILI konektore. Ponovo se bira nasumična pozicija ukrštanja nakon čega se vrši unakrsno prenošenje konektora kao što je to slučaj kod vektora zahvata, mašina, alata, SPA i vremena obrade.



Slika 5.14. Operator jednopozicionog ukrštanja

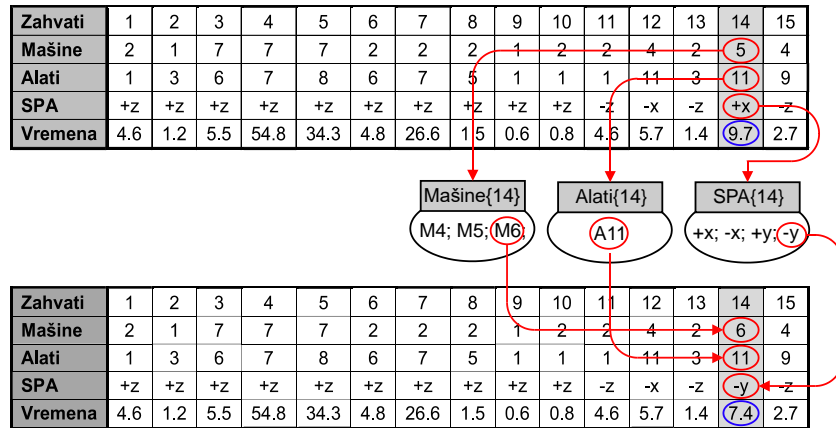
3) **Mutacija zamene**. Prvi operator mutacije koji je usvojen je operator zamene. Ovaj operator ima za cilj izmenu niza odabranih I i ILI konektora kojim se utiče na izmenu putanje na mreži a samim tim i selekciju drugog izvodljivog redosleda zahvata obrade.

Sa određenom verovatnoćom mutacije zamene, p_s , nasumično se biraju dve pozicije u nizovima „I“ i „|I|“ kao što je prikazano na slici 5.15 za pozicije 1 i 2. Razmatran je isti primer rotacionog dela sa slike 6.25 (odjeljak 6.3.2). U skladu sa selektovanim konektorima u oba niza vrši se njihova zamena gde nulti konektor koji bira levu stranu putanje biva zamenjen jediničnim konektorom koji bira desnu stranu i obratno. Jasno se može uočiti da se promenom datih konektora u potpunosti vrši izmena redosleda zahvata obrade i kao rezultat dobija se nova varijanta tehnološkog procesa. Na slici 5.15 su prikazana dekodirana rešenja. Vektori mašina, alata, SPA i vremena za novogenerisani redosled ažuriraju se nakon mutacije zamene.



Slika 5.15. Operator mutacije zamene

Mutacija resursa. Poslednji genetski operator inkorporiran u koncept mGWO metode je operator mutacije resursa. Zadatak ove mutacije je da za slučajno odabran zahvat obrade u okviru kodiranog tehnološkog procesa (primer u tabeli 5.2) izvrši izmenu postojećih varijanti mašina, alata i SPA. Procedura mutacije resursa ilustrovana je na slici 5.16 na kojoj je kodirani tehnološki proces delimično predstavljen. Za slučajno selektovani zahvat obrade vrši se zamena varijantne mašine, alata i SPA u skladu sa raspoloživim kandidatima za dati zahvat. Kao što se sa slike 5.16 može zaključiti, zahvat 14 je nasumično odabran kao kandidat za mutaciju. Crvenom bojom označene su varijantna mašina, alat i SPA. Proveravanjem raspoloživih varijanti u bazi $Mašine\{14\}$ za otkriva se da postoje ukupno tri varijantne mašine alatke od kojih je slučajno odabrana mašina 6. Izbačena je mogućnost ponavljanja iste varijante. Kada su u pitanju alati, alat 11 je jedini dostupan u bazi $Alati\{14\}$ što znači da nema izmene u tom vektoru. Baza $SPA\{14\}$ sadrži četiri varijantna prilaza alata gde je smer prilaza $-y$ uzet kao zamena za $+x$. Na kraju, nakon mutiranja gena vektora mašine alatke, alata i SPA respektivno pristupa se ažuriranju vremena obrade koji je plavom bojom označen na slici 5.16. Formiranjem novog tipskog zahvata dobija se drugačije vreme njegove realizacije što potencijalno može umanjiti ukupne troškove obrade razmatranog proizvoda.



Slika 5.16. Operator mutacije resursa

5.3 Modifikovani algoritam grbavog kita u optimizaciji tehnoloških procesa obrade

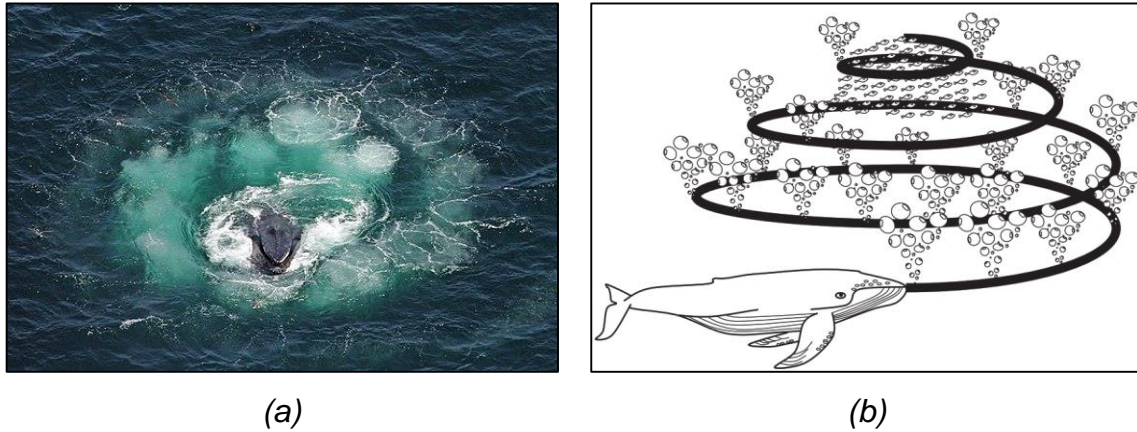
5.3.1 Tradicionalni algoritam grbavog kita (WOA)

Algoritam grbavog kita takođe se ubraja u savremenije metaheurističke algoritme koji pripadaju oblasti inteligencije rojeva. Isti autor *Mirjalili* [32] razvio je ovu metodu koja je inspirisana inteligencijom jata grbavih kitova i njihovim načinom hranjenja. Prema profilu autora sa Google Scholar naučnog servisa, može se primetiti da *GWO* i *WOA* predstavljaju njegove najpopularnije metaheurističke metode sa velikim brojem citata u poslednjih šest, odnosno sedam godina [171].

Poznato je da kitovi predstavljaju najveće sisare na našoj planeti. U mogućnosti su da dosegnu težine od nekoliko stotina tona i dužinu do čak 30 metara. Među kitovima možemo izdvojiti kitove ubice, ulješure, plave kitove, grbave kitove, perjare i mnoge druge. Većinom su predatori jer se hrane drugim morskim organizmima, od planktona, rakova, manjih riba pa sve do velikih sisara. Kitove karakteriše visok intelekt pre svega zahvaljujući ćelijama u mozgu koje su nalik čovečjim. To im daje sposobnost da rasuđuju, komuniciraju, uče, osećaju emocije pa čak i da razviju sopstveni jezik. Ono što najviše pažnje privlači jeste njihovo socijalno ponašanje. Kitovi mogu živeti sami ili u određenoj grupi kao što su kitovi ubice ili grbavi kitovi.

Grbavi kitovi (lat. *Megaptera novaeangliae*) predstavljaju jednu od najvećih jedinki kitova usana koji mogu doseći veličinu školskog autobusa. Ovi kitovi hrane se u jatima kako bi obezbedili najbolji ulov, a interesantni su zbog posebne tehnike koju koriste pri lovu na ribu i kril. Ona se zove tehnika mreže mehurića ili tehnika "zavese" i grafički je ilustrovana na slici 5.17. Kada pronađu plen, grbavi kitovi timskim naporom plivaju u krugu ispod jata krilova ili manjih riba, pri čemu krećući se spiralnom putanjom proizvode zvukove i ispuštaju mehuriće vazduha kroz disajne otvore koji na taj način formiraju "mrežu" ili "zavesu" mehurića oko plena. Dezorijentisan, plen biva zarobljen blizu površine što omogućava grbavim kitovima da izrone širom otvorenih usta i pojedu svoj ulov u

jednom zalogaju. Ova tehnika lova grbavih kitova je jedinstvena u svetu prirode i time je predstavila izvor inspiracije za razvoj WOA algoritma.



Slika 5.17. Tehnika lova stvaranjem mreže mehurića: (a) snimak iz vazduha [172] i (b) šematski prikaz [32]

5.3.1.1 Matematički model tradicionalnog WOA algoritma

Grbavi kitovi imaju potencijal da pronađu lokaciju plena koji potom opkoljavaju i love. Kao kod sivih vukova, ne postoji način da se unapred zna gde se plen nalazi, što podrazumeva da se najbolje pozicionirani agent pretraživanja smatra plenom, odnosno rešenjem koje je blisko optimalnom. Ostali agenti pretraživanja predstavljaju kitove koji ažuriraju svoje položaje u skladu sa položajem plena. Osnovne komponente matematičkog modela WOA algoritma čine opkoljavanje plena, tehnika stvaranja mreže mehurića sa spiralnim kretanjem i potraga za plenom.

Kada je u pitanju prva faza opkoljavanja plena, koriste se identični matematički izrazi (33-36) koji su definisani u kod GWO algoritma u odeljku 5.2.1.1. WOA algoritam takođe koristi koeficijent vektore su \vec{A} i \vec{C} kao i veličinu \vec{a} koja opada linearno. Pomoću ovih vektora moguće je ostvarivanje različitih položaja agenata pretraživanja u okviru prostora pretraživanja. U skladu sa time, za dvodimenzionalne probleme položaj kitova u odnosu na plen može se prikazati na isti način kao što je predstavljeno na slici 5.10. Jednačine kojima se modelira prva faza opkoljavanja plena su sledeće:

$$\vec{D} = |\vec{C} \cdot \vec{X}_p(t) - \vec{X}(t)| \quad (49)$$

$$\vec{X}(t+1) = \vec{X}_p(t) - \vec{A} \cdot \vec{D} \quad (50)$$

$$\vec{A} = 2\vec{a}\vec{r} - \vec{a} \quad (51)$$

$$\vec{C} = 2\vec{r} \quad (52)$$

gde \vec{r} predstavlja slučajnu vrednost između 0 i 1 dok su ostale veličine, odnosno vektori već opisani kod formulisanja jednačina (33-36).

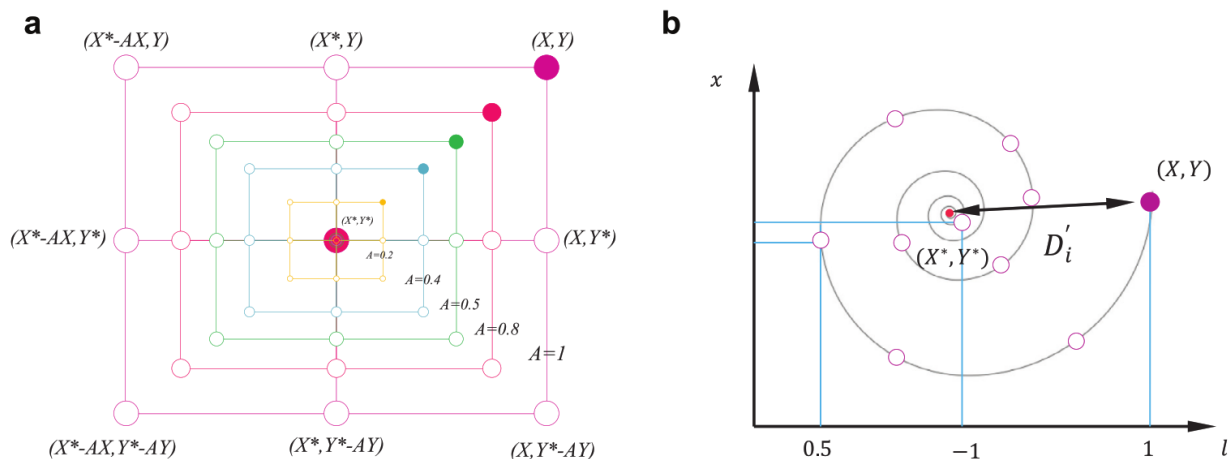
Matematičko modelovanje tehnike stvaranja mreže mehurića zasebno razmatra dva kretanja. Sa jedne strane radi se o mehanizmu sužavanja kruga kojim grbavi kitovi zarobljavaju plen u sredini svoje „mreže“. Drugi mehanizam odnosi se na spiralno ažuriranje položaja pri čemu izbacuju mehuriće vazduha i stvaraju „mrežu“ oko plena.

Prvi mehanizam koji se odnosi se na sužavanje kruga oko plena kontrolisan je pomoću vektora \vec{a} . Kao i kod GWO, linearno opadanje ove vrednosti od 2 do 0 omogućava smanjenje koeficijent vektora \vec{A} što ispunjava uslov da je apsolutna vrednost $\vec{A} < 1$. Na ovaj način omogućava se intenzivnije pretraživanje prostora, odnosno ističu se lokalni kapaciteti WOA algoritma. Kako se kit približava plenu sužavanjem kruga koji formira oko njega, na taj način WOA algoritam usmerava svoje pretraživačke kapacitete ka intenzifikaciji, odnosno eksploataciji potencijalnih rešenja. Na slici 5.18a nalazi se šematski prikaz ovog mehanizma.

Mehanizam spiralnog ažuriranja položaja kitova koji je šematski dat na slici 5.18b polazi od određivanja rastojanja između agenta pretraživanja ili grbavog kita (X, Y) i plena (X^*, Y^*) . Nakon toga definiše se jednačina spirale na osnovu koje se dobija novi položaj agenta pretraživanja za narednu iteraciju algoritma. Jednačina spirale ima sledeću formu:

$$\vec{X}(t+1) = \vec{D}' \cdot e^{bl} \cdot \cos(2\pi l) + \vec{X}_p(t) \quad (53)$$

gde $\vec{D}' = |\vec{X}_p(t) - \vec{X}(t)|$ predstavlja rastojanje grbavog kita od plena (najbolje rešenje do tog trenutka), b je konstanta za definisanje oblika logaritamske spirale i l je slučajan broj koji se kreće u intervalu od -1 do 1.



Slika 5.18. Tehnika stvaranja mreže mehurića kroz mehanizme dva kretanja: (a) sužavanje kruga oko plena i (b) spiralno ažuriranje položaja [32]

S obzirom na činjenicu da se grbavi kitovi kreću koristeći oba mehanizma simultano, pretpostavlja se da postoji 50% mogućnosti da će se pomoću jednog od ovih mehanizama izvršiti ažuriranje položaja kitova u prostoru pretraživanja. Shodno tome, formulisani su sledeći matematički izrazi za tehniku stvaranja mreže mehurića koja koristi dva opisana kretanja:

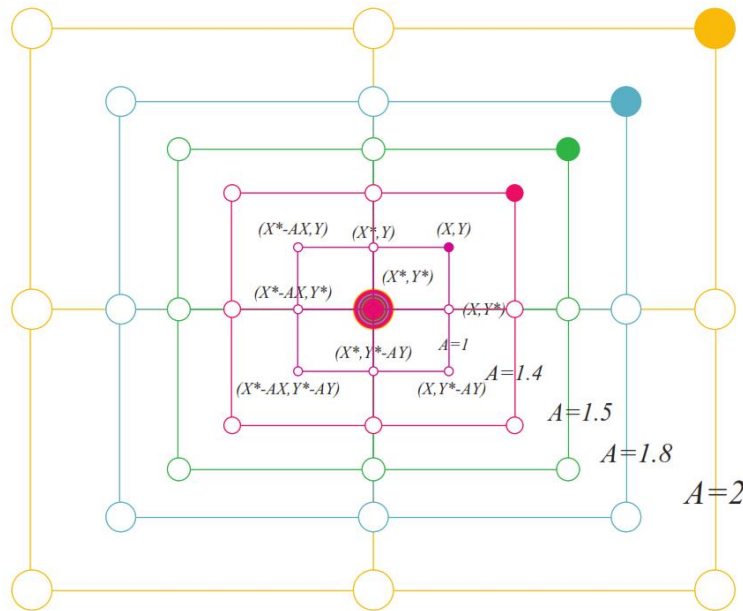
$$\vec{X}(t+1) = \begin{cases} \vec{X}_p(t) - \vec{A} \cdot \vec{D} & \text{ako je } p < 0,5 \\ \vec{D} \cdot e^{bl} \cdot \cos(2\pi l) + \vec{X}_p(t) & \text{ako je } p \geq 0,5 \end{cases} \quad (54)$$

Analogno konceptu GWO algoritma, istraživački kapaciteti, odnosno diverzifikacija pretraživanja u okviru WOA ostvaruje se kada je apsolutna vrednost koeficijent vektora \vec{A} veća od 1. Algoritam na bazi toga poseduje kapacitete za globalno pretraživanje što se matematički može izraziti preko sledećih izraza:

$$\vec{D} = |\vec{C} \cdot \overrightarrow{X_{rand}} - \vec{X}| \quad (55)$$

$$\vec{X}(t+1) = \overrightarrow{X_{rand}} - \vec{A} \cdot \vec{D} \quad (56)$$

gde $\overrightarrow{X_{rand}}$ predstavlja slučajno generisan vektor položaja grbavog kita. Na slici 5.19 prikazana je šema poslednjeg mehanizma potrage za plenom u kojem su istraživački kapaciteti WOA algoritma stavljeni u prvi plan. Može se uočiti da se u ovom slučaju radi o divergenciji grbavih kitova kao što je to slučaj sa njihovim ekvivalentom, sivim vukovima čija je divergencija prikazana na slici 5.12b.



Slika 5.19. Mehanizam potrage za plenom [32]

Na kraju, pseudokod modelovanog WOA algoritma dat je u algoritmu 4. WOA započinje proces pretraživanja generisanjem inicijalne populacije slučajnih agenata koji predstavljaju grbave kitove. Svi agenti ažuriraju svoje vektore položaja u skladu sa najboljim agentom pretraživanja koji reprezentuje plen. Na osnovu vektora \vec{a} linearnim opadanjem omogućava se prelaz iz istraživanja prostora u eksploataciju rešenja pri čemu su naglašene i globalne, a potom lokalne sposobnosti algoritma. Pomoću koeficijenta vektora \vec{A} ostvaruje se ažuriranje položaja jedinke sa jedne strane, odnosno nasumična pretraga sa druge. Verovatnoća p daje 50% šansi jednom od dva mehanizma kretanja,

spiralnom ažuriranju ili sužavanju kruga oko plena. Nakon prethodno definisanog konačnog broja iteracija postupak pretraživanja pomoću WOA algoritma se obustavlja.

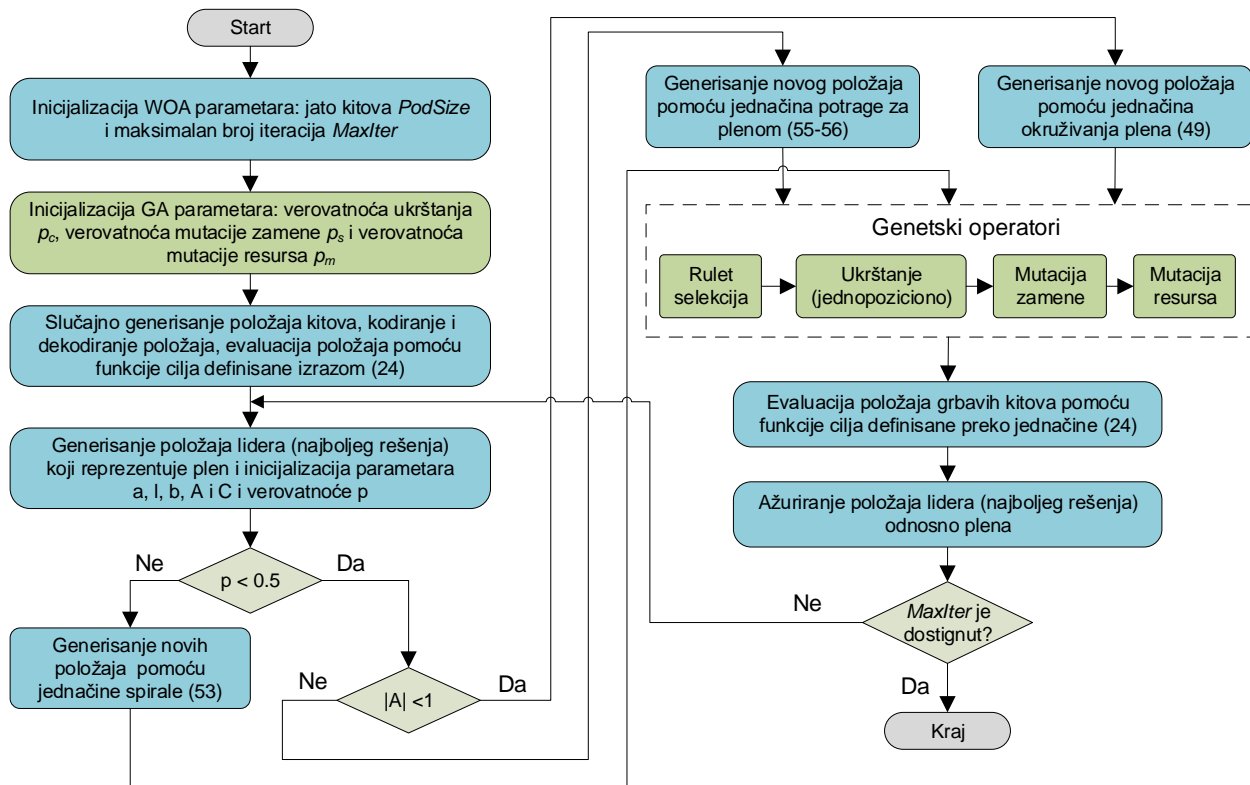
Algoritam 4. Pseudokod tradicionalnog WOA algoritma

<p>1. Početak: Inicijalizacija Inicijalizovati slučajne položaje agenata (grbavih kitova) i formirati populaciju (jato); Odrediti vrednost funkcije cilja svakog agenta (grbavog kita) u početnoj populaciji (jatu); Generisati najbolje početno rešenje – najbolji vektor položaja koji predstavlja plen;</p>
<p>2. Ponavljanje: Iterativni postupak Ponoviti iteracija = iteracija + 1; ponoviti za sve kitove u jatu Generisati/ažurirati parametre \vec{a}, \vec{A}, \vec{C}, l i p ako je verovatnoća $p < 0,5$ (prema jednačini (54) ako je $A < 1$ Ažurirati položaj postojećeg grbavog kita pomoću izraza (49) u suprotnom $A \geq 1$ Selektovati slučajnog kita \vec{X}_{rand} Ažurirati položaj postojećeg grbavog kita pomoću izraza (56) završi uslov u suprotnom ako je verovatnoća $p \geq 0,5$ (prema jednačini (54) Ažurirati položaj postojećeg grbavog kita pomoću jednačine spirale (53) završi uslov završi petlju proveriti izvodljivost novih položaja grbavih kitova u jatu proveriti vrednost funkcije cilja svakog grbavog kita u jatu ažurirati najbolji položaj, položaj plena završi petlju</p>
<p>3. Izlazni rezultati</p>

5.3.2 Modifikovani algoritam grbavog kita (mWOA)

Poslednji predloženi metaheuristički algoritam savremene generacije za optimizaciju tehnoloških procesa obrade u ovoj doktorskoj disertaciji je modifikovani algoritam grbavog kita (mWOA). Modifikacija tradicionalnog WOA algoritma bazirana je na identičnom principu kao i prethodna modifikacija tradicionalnog GWO. Ugrađeni su genetski operatori selekcije, ukrštanja, mutacije zamene i mutacije resursa sa ciljem poboljšanja istraživačkih kapaciteta i samim tim izbegavanje zarobljavanja u lokalnim okolinama. Kao i mGWO, mWOA primenu pronalazi u petom i šestom eksperimentu u kojima je funkcija cilja minimiziranje ukupnog vremena obrade proizvoda. Dijagram toka mWOA predložene metode dat je na slici 5.20. Procedura tipično započinje podešavanjem polaznih parametara WOA i GA. Potom nastupa kodiranje i dekodiranje jedinki kitova koje se odvija na način koji je već opisan u odeljku 5.2.2.1. Predstavljeni tehnološki procesi koji reprezentuju grbave kitove formiraju inicijalnu populaciju slučajno generisanih jedinki. Nakon vrednovanja tehnoloških procesa prema funkciji cilja definisanoj u jednačini (24), sledi selekcija najboljeg tehnološkog procesa koji će u nastavku reprezentovati lidera, odnosno plen. Naredni korak odnosi se na petlju tradicionalnih WOA koraka pri čemu su položaji grbavih kitova klasično formulisani matričnim putem. Razvijeno je pet vektora pri

čemu se vektor vremena ažurira u skladu izmenjenim vektorima mašina, alata i SPA. Modifikovane jednačine WOA algoritma za rešavanje OTP problema definisane su u odeljku 5.3.2.1. Nove jedinke grbavih kitova dobijaju se na tri načina prateći mehanizme WOA algoritma. One su dalje podvrgnute genetskim operacijama izmene vektora koje će doneti potencijalno bolja rešenja. Rulet selekcijom, zatim jednopozicionim ukrštanjem i na kraju mutacijama zamene i resursa dobijaju se izmenjeni tehnološki procesi koji se potom evaluiraju pomoću funkcije cilja. Lideri, odnosno najbolji tehnološki procesi se ažuriraju i iterativni postupak se nastavlja sve do maksimalnog broja iteracija.



Slika 5.20. Dijagram toka mWOA algoritma za optimizaciju tehnoloških procesa

5.3.2.1 Matematički model mWOA algoritma

Kako je već istaknuto, kodiranje i dekodiranje jedinki kitova odvija se na isti način kao kod algoritma mGWO. Od ukupno pet vektora, vektor redosleda zahvata modifikovan je genetskim operacijama dok je vektor vremena ažuriran u skladu sa vektorima mašina, alata i SPA. Položaji grbavih kitova koji predstavljaju kodirane tehnološke procese mogu se ažurirati na nekoliko načina. Jednačine u okviru tradicionalnog WOA (49-56) formulisane su pojedinačno za vektore mašina, alata i SPA.

Glavna petlja WOA metode započinje generisanjem parametara \vec{A} , \vec{C} i l koji zavise ili sami predstavljaju slučajne promenljive. Konstanta b i vrednost \vec{a} se podešavaju u inicijalnoj fazi pretraživanja. Pored njih, neophodno je slučajno generisati i verovatnoću p koja omogućava 50% mogućnosti za selekciju jednog od dva mehanizma koji su deo tehnike stvaranja mreže mehurića. Ukoliko slučajno generisanja vrednost p u datoj

iteraciji ima vrednost manju od 0,5 javljaju se dva načina za ažuriranje položaja grbavih kitova koja su opisana prema jednačinama (49),(50) i (55),(56). Oba načina zavise direktno od koeficijent vektora \vec{A} a koji je, sa druge strane, podstaknut linearno opadajućom veličinom \vec{a} . Prvi način, kada je $|\vec{A}| < 1$, koristi metod opkoljavanja plena koji je u okviru mWOA formulisan preko sledećih matematičkih izraza:

$$\begin{aligned}\vec{D}_m &= |\vec{C} \cdot \vec{X}_{p,m}(it) - \vec{X}_m(it)| \\ \vec{D}_a &= |\vec{C} \cdot \vec{X}_{p,a}(it) - \vec{X}_a(it)| \\ \vec{D}_{spa} &= |\vec{C} \cdot \vec{X}_{p,spa}(it) - \vec{X}_{spa}(it)|\end{aligned}\quad (57)$$

$$\begin{aligned}\vec{X}_m(it+1) &= \vec{X}_{p,m}(it) - \vec{A} \cdot \vec{D}_m \\ \vec{X}_a(it+1) &= \vec{X}_{p,a}(it) - \vec{A} \cdot \vec{D}_a \\ \vec{X}_{spa}(it+1) &= \vec{X}_{p,spa}(it) - \vec{A} \cdot \vec{D}_{spa}\end{aligned}\quad (58)$$

gde \vec{D}_m , \vec{D}_a i \vec{D}_{spa} predstavljaju rastojanja između grbavih kitova i plena/lidera (najboljeg rešenja) za redom vektore mašina, alata i SPA; $\vec{X}_{p,m}(it)$, $\vec{X}_{p,a}(it)$ i $\vec{X}_{p,spa}(it)$ su vektori položaja najboljeg rešenja za redom vektore mašina, alata i SPA u tekućoj iteraciji it; $\vec{X}_m(it)$, $\vec{X}_a(it)$ i $\vec{X}_{spa}(it)$ su vektori položaja grbavih kitova (tekućeg rešenja) za vektore mašina, alata i SPA u tekućoj iteraciji it; $\vec{X}_m(it+1)$, $\vec{X}_a(it+1)$ i $\vec{X}_{spa}(it+1)$ su vektori položaja grbavih kitova (tekućeg rešenja) za vektore mašina, alata i SPA u narednoj iteraciji it + 1.

U drugom slučaju kada važi da je $|\vec{A}| > 1$, algoritam vrši divergenciju od plena i prostor pretražuje globalno pri čemu jednačine (55) i (56) za rastojanja i ažuriranje vektora položaja imaju sledeći matematički oblik:

$$\begin{aligned}\vec{D}_m &= |\vec{C} \cdot \vec{X}_{rand,m} - \vec{X}_m| \\ \vec{D}_a &= |\vec{C} \cdot \vec{X}_{rand,a} - \vec{X}_a| \\ \vec{D}_{spa} &= |\vec{C} \cdot \vec{X}_{rand,spa} - \vec{X}_{spa}|\end{aligned}\quad (59)$$

$$\begin{aligned}\vec{X}_m(it+1) &= \vec{X}_{rand,m} - \vec{A} \cdot \vec{D}_m \\ \vec{X}_a(it+1) &= \vec{X}_{rand,a} - \vec{A} \cdot \vec{D}_a \\ \vec{X}_{spa}(it+1) &= \vec{X}_{rand,spa} - \vec{A} \cdot \vec{D}_{spa}\end{aligned}\quad (60)$$

gde $\vec{X}_{rand,m}$, $\vec{X}_{rand,a}$ i $\vec{X}_{rand,spa}$ predstavljaju slučajno generisane vektore položaja grbavog kita koji redom važe za vektor mašina, alata i SPA.

Na kraju, akcentat se postavlja na verovatnoći odabira spiralne putanje kretanja koja se selektuje u slučajevima kada je vrednost p veća od 50%, odnosno 0,5. U tom slučaju, jednačina spirale (53) dobija ovakav oblik:

$$\begin{aligned}\overrightarrow{X}_m(it+1) &= \overrightarrow{D}'_m \cdot e^{bl} \cdot \cos(2\pi l) + \overrightarrow{X}_{p,m}(it) \\ \overrightarrow{X}_a(it+1) &= \overrightarrow{D}'_a \cdot e^{bl} \cdot \cos(2\pi l) + \overrightarrow{X}_{p,a}(it) \\ \overrightarrow{X}_{spa}(it+1) &= \overrightarrow{D}'_{spa} \cdot e^{bl} \cdot \cos(2\pi l) + \overrightarrow{X}_{p,spa}(it)\end{aligned}\quad (61)$$

gde su \overrightarrow{D}'_m , \overrightarrow{D}'_a i $\overrightarrow{D}'_{spa}$ rastojanja grbavog kita od plena/lidera (najboljeg rešenje do tog trenutka) za redom vektore mašina, alata i SPA. Ova rastojanja formulišu se analogno rastojanjima iz jednačine (59):

$$\begin{aligned}\overrightarrow{D}'_m &= |\overrightarrow{X}_{p,m}(it) - \overrightarrow{X}_m(it)| \\ \overrightarrow{D}'_a &= |\overrightarrow{X}_{p,a}(it) - \overrightarrow{X}_a(it)| \\ \overrightarrow{D}'_{spa} &= |\overrightarrow{X}_{p,spa}(it) - \overrightarrow{X}_{spa}(it)|\end{aligned}\quad (62)$$

Ovi izrazi kompletiraju matematički model predložene modifikacije WOA algoritma za optimizaciju tehnoloških procesa. Nakon gore prikazanih koraka WOA metode pristupa se genetskim izmenama elemenata tehnološkog procesa koji omogućavaju generisanje boljih varijanti tehnoloških procesa.

U narednom poglavlju rada sprovedeno je eksperimentalno istraživanje u okviru kojeg je izvršena verifikacija performansi triju savremenih biološki inspirisanih metoda prilikom rešavanja problema OTP. GCSA algoritam je primenjen u prva četiri eksperimenta za minimizaciju ukupnih troškova obrade prizmatičnih delova, dok su mGWO i mWOA implementirani u petom i šestom eksperimentu gde su pokazali svoj doprinos u minimizaciji ukupnog vremena obrade delova rotacionog oblika.

6. ANALIZA EKSPERIMENTALNIH REZULTATA OPTIMIZACIJE TEHNOLOŠKIH PROCESA OBRAD PROIZVODA

U ovom poglavlju su prikazani eksperimentalni rezultati optimizacije tehnoloških procesa obrade proizvoda sa ciljem verifikacije performansi predloženih biološki inspirisanih metaheurističkih algoritama. Radi se o modifikovanim algoritmima vrane, sivog vuka i grbavog kita koji su detaljno opisani u poglavlju 5. Realizovano je šest eksperimentalnih studija slučaja u okviru kojih je razmotrena optimizacija tehnoloških procesa obrade šest delova, od kojih su prva četiri delovi prizmatičnog oblika, a preostala dva rotacionog oblika. Razlog odabira ovih reprezentativnih delova leži u činjenici da se radi o primerima koji su deo višegodišnjeg istraživanja u oblasti OTP-a što je potvrđeno kroz veliki broj objavljenih radova i publikacija. Eksperimenti se zasnivaju na uporednim analizama rezultata ostvarenih primenom predloženih metoda sa rezultatima koji su ostvareni primenom drugih algoritama. Izlazni rezultati prikazani su u tabelama, zatim na graficima i krivama konvergencije koje na jednostavan način daju prikaz učinka metaheurističkih algoritama tokom iterativnog procesa.

Najviše pažnje u prva četiri eksperimenta biće posvećeno modifikovanom algoritmu vrane (GCSA) čiji je koncept detaljno opisan u poglavlju 5.1. Verifikacija performansi GCSA obavljena je na primerima prizmatičnih delova za koje se minimiziraju ukupni troškovi obrade. Podešavanje i optimizacija ulaznih parametara ovog unapređenog algoritma izvršeni su u poglavlju 6.1. U prvom i četvrtom eksperimentu ulazni parametri dobijaju se na dva načina. Prvi način je manuelno podešavanje nakon određenog broja probnih testova, dok drugi način podrazumeva optimizaciju parametara primenom metode višefaktornog eksperimenta. Kod ostalih eksperimenata preovladava manuelno podešavanje ulaznih parametara.

U petom i šestom eksperimentu optimizuju se tehnološki procesi obrade delova rotacionog oblika kod kojih je funkcija cilja minimalno vreme obrade. Akcenat je stavljen na druge dve biološki inspirisane metode koje su deo istraživanja u ovoj disertaciji. U pitanju su modifikovani algoritam sivog vuka (mGWO) i modifikovani algoritam grbavog kita (mWOA) koji su detaljno opisani u poglavljima 5.2 i 5.3. Ulazni parametri ovih algoritama manuelno su usvojeni nakon nekoliko probnih testova.

Delovi, odnosno proizvodi u okviru eksperimenata predstavljeni su u formi trodimenzionalnih solid modela među kojima su neki modeli dati u konceptualnom i transparentnom prikazu. Pored modela predstavljeni su grafovi i matrice prethođenja na bazi kojih se definišu odnosi prethođenja među zahvatima, odnosno tipskim tehnološkim oblicima. U slučaju rotacionih delova razmotreni su različiti tipovi fleksibilnosti tehnoloških procesa korišćenjem I/II mreža. Specifikacija tipskih tehnoloških oblika, tehnoloških resursa, konstantnih troškova i odnosa prethođenja dati su u tabelarnoj formi za svaki deo. Transportna vremena i jedinični troškovi promena su takođe definisani. Dalje, predstavljeni su rezultati optimizacije, odnosno optimalni tehnološki procesi obrade dela. Pored najboljih rešenja, takođe su dati tabelarni prikazi najboljih vrednosti troškova, odnosno vremena obrade i vrednosti funkcija cilja, f_c ili f_t , za određeni broj egzekucija algoritama. Statistički pokazatelji u vidu minimalne, maksimalne, srednje vrednosti i standardne devijacije rezultata pomažu u određivanju konzistentnosti predloženih metaheurističkih algoritama. Uporedne tabele sa dijagramima i krivama konvergencije omogućuju grafičko poređenje rezultata i performansi predloženih algoritama sa rezultatima i performansama tradicionalnih i drugih modifikovanih i hibridnih algoritama. Svi metaheuristički algoritmi implementirani su u Matlab programskom jeziku na računaru savremene konfiguracije sa 1,99 GHz Intel i7 procesorom i 8 GB RAM memorije.

6.1 Podešavanje parametara i analiza rezultata prve, druge i treće eksperimentalne studije

Podešavanje parametara metaheurističkog algoritma u inicijalnoj fazi je značajno jer ima veliki uticaj na dalji proces pretraživanja i potom na izlazne rezultate. U ovom poglavlju predstavljena su dva prilaza među kojima se prvi odnosi na optimizaciju ulaznih parametara GCSA, a sa druge strane je odrađena manuelna selekcija nakon probnih ispitivanja. Optimizacija parametara izvršena je primenom metode višefaktornog plana eksperimenta. Ova metoda koristi ortogonalne nizove na bazi kojih se dobija odgovarajući broj eksperimenata pri čemu se stiču informacije o svim faktorima koji utiču na performanse nekog procesa [70,173]. Faktornim eksperimentima može se izvršiti optimizacija parametara obrade kod konvencionalnih [174-176], kao i nekonvencionalnih metoda mašinske obrade [177]. Na osnovu ulaznih parametara algoritma i odgovarajućih nivoa varijacija, u ovim eksperimentalnim istraživanjima formiran je delimični (parcijalni) faktorni plan eksperimenta.

Razmatranjem devet različitih ulaznih parametara GCSA algoritma formulisanih u dva nivoa varijacije, razvijen je delimični faktorni plan 2^{9-4} koji predstavlja 1/16 replike potpunog faktornog plana 2^9 . U skladu sa time, delimični faktorni plan sadrži niz od $2^5 = 32$ eksperimenata. Nakon optimizacije izlaznih statističkih parametara usvojeni su ulazni parametri GCSA algoritma na osnovu kojih su testirane njegove performanse. U nastavku istraživanja, navedeni GCSA algoritam sa optimalnim ulaznim parametrima dobijenim na osnovu formiranog delimičnog plana eksperimenta adresiran je kao GCSA-Par1.

Sa druge strane, parametri GCSA algoritma takođe su usvojeni nakon nekoliko preliminarnih manualnih testova. Na bazi toga, u uporednoj analizi rezultata

eksperimenata primenjen je GCSA algoritam sa drugim ulaznim parametrima koji je adresiran kao GCSA-Par2.

Prvi predloženi algoritam GCSA-Par1 formiran je na osnovu sledećih ulaznih parametara i nivoa njihovih varijacija: maksimalan broj iteracija *MaxIt* [400; 1000], veličina jata *FlockSize* [60; 120], veličina turnira *TourSize* [2; 5] i adaptivni koeficijenti za ukrštanje i mutaciju k_{c1} [0,6; 0,9], k_{c2} [0,3; 0,6], k_{m1} [0,3; 0,6] i k_{m2} [0,1; 0,3], respektivno. Na osnovu delimičnog faktornog plana, dovoljna su 32 eksperimenta kako bi se izveli adekvatni zaključci o ulaznim parametrima. Za potpuni plan eksperimenta bilo bi neophodno sprovesti $2^9 = 512$ eksperimenata kako bi se utvrdila efikasnost ovih parametara.

Drugi predloženi algoritam GCSA-Par2 odnosi se na sledeće parametre GCSA algoritma dobijene manuelnom selekcijom: maksimalan broj iteracija *MaxIt* je 700, veličina jata *FlockSize* je 100, veličina turnira *TourSize* je 5, adaptivni koeficijenti za ukrštanje k_{c1} i k_{c2} su 0,9 i 0,6 respektivno i adaptivni koeficijenti za mutaciju k_{m1} i k_{m2} su 0,6 i 0,3 respektivno. Takođe, osnovni parametri algoritma vrane, dužina leta *fl* i verovatnoća svesnosti vrane *AP* usvojeni su na osnovu manualnih ispitivanja, a u skladu sa preporučenim vrednostima koje iznose 2 i 0,8 respektivno.

Sa ciljem povećanja tačnosti rezultata, a u okviru algoritma GCSA-Par1, izvršeno je 20 izvršavanja za svaki pojedinačni eksperiment. Na taj način, statističke vrednosti u vidu srednjeg, minimalnog, maksimalnog rezultata ukupnih troškova TWMC i standardne devijacije ostvarenih rezultata, zabeleženi su kao izlazni parametri optimizacije. Delimični ortogonalni plan eksperimenta za GCSA-Par1, zajedno sa statističkim rezultatima na izlazu prikazan je u tabeli 6.1. Optimizacija ulaznih parametara GCSA izvršena je pomoću optimizacije odziva u okviru statističkog alata Minitab. Optimizacijom odziva (eng. *Response optimizer*) omogućava se identifikovanje promenljive ili skupa promenljivih koje utiču na optimalnost jednog ili više odziva. Shodno planu eksperimenta iz tabele 6.1, manuelno se određuju donja, ciljna i gornja vrednost izlaznih statističkih parametara, njihovi težinski koeficijenti i koeficijenti važnosti. U tabeli 6.2 prikazane su ove vrednosti za minimalnu i srednju vrednost troškova TWMC. Dalje se vrši određivanje individualnih indeksa poželjnosti kojima se dodeljuju određene težinske vrednosti prema koeficijentima važnosti. Potom se ovi individualni indeksi kombinuju kako bi se odredila kompozitna željenost celokupnog odziva. Na optimizacionom grafiku prikazuju se optimalni parametri ulaznih promenljivih sa indeksima individualne željenosti (*d*) i kompozitne željenosti (*D*). Na osnovu ovih indeksa može se proceniti u kojoj meri postojeća kombinacija ulaznih parametara optimizuje pojedinačni, odnosno grupni odziv. Kreću se u rasponu od 0 do 1 pri čemu se broj 1 javlja za idealne slučajeve dok 0 ukazuje na to da su jedan ili više odziva izvan postavljenog limita. Indeks kompozitne željenosti dobija se određivanjem geometrijske sredine indeksa individualne željenosti. U tabeli 6.3 mogu se videti vrednosti optimalnih parametara za GCSA-Par1 sa ciljnim vrednostima izlaznih parametara. Pored toga, generisan je indeks kompozitne željenosti koji iznosi 1 što se smatra idealnim za ove uslove u kojima su kao pojedinačni odzivi usvojene minimalna i srednja vrednost TWMC. Optimizacioni grafik zavisnosti ulaznih parametara na izlazne statističke vrednosti nalazi se na slici 6.1. Na bazi optimalnih ulaznih parametara GCSA, izvršena je optimizacija tehnoloških procesa obrade prizmatičnog dela razmatranog u okviru prve eksperimentalne studije koja je predstavljena u nastavku.

Tabela 6.1. Delimični ortogonalni plan eksperimenta 2⁵ sa izlaznim statističkim rezultatima

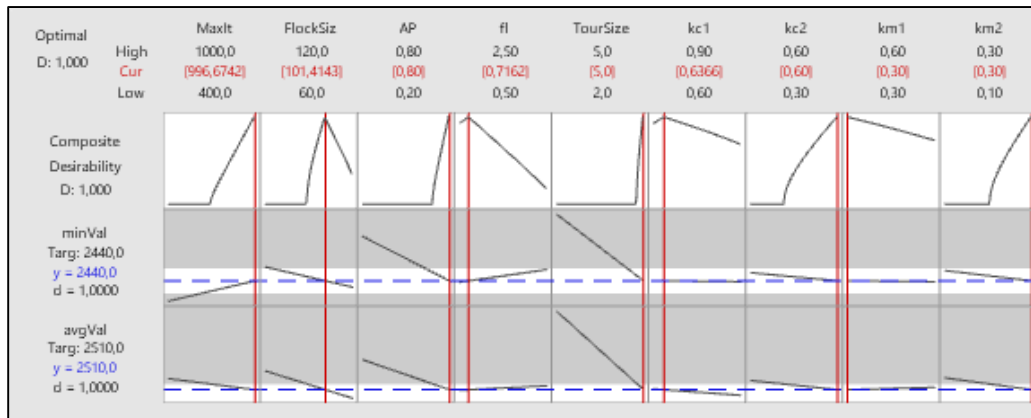
br. eksp.	Ulazni parametri modifikovanog algoritma vrane									Izlazni parametri			
	Maxlt	FlockSize	AP	fl	TourSize	kc ₁	kc ₂	km ₁	km ₂	SV	MIN	MAX	STD
1.	1000	60	0,8	0,5	2	0,6	0,6	0,3	0,3	2658,25	2535	2760	68,5042
2.	400	60	0,8	0,5	5	0,9	0,6	0,3	0,3	2576,5	2507	2720	83,8498
3.	400	60	0,2	2,5	2	0,6	0,3	0,3	0,3	3107,5	2990	3270	82,176
4.	400	60	0,2	0,5	2	0,9	0,6	0,6	0,3	2942,5	2795	3070	63,9453
5.	1000	120	0,8	0,5	5	0,6	0,3	0,3	0,3	2511,5	2442	2700	69,9455
6.	1000	120	0,8	2,5	5	0,9	0,6	0,6	0,3	2508,35	2444	2684	44,7676
7.	400	60	0,8	0,5	2	0,6	0,3	0,6	0,1	2748,25	2650	2840	43,5042
8.	1000	120	0,8	0,5	2	0,9	0,6	0,6	0,1	2730,3	2605	2790	44,3219
9.	400	120	0,8	2,5	2	0,6	0,6	0,6	0,3	2874,9	2700	2969	69,1458
10.	400	120	0,2	0,5	5	0,9	0,3	0,6	0,3	2584,7	2515	2745	80,49
11.	1000	120	0,2	2,5	2	0,9	0,6	0,3	0,3	2990,05	2855	3115	69,9424
12.	400	120	0,2	2,5	2	0,9	0,3	0,6	0,1	3060,55	2859	3255	98,7615
13.	1000	60	0,8	0,5	5	0,9	0,3	0,6	0,1	2546,45	2442	2715	83,1647
14.	400	120	0,8	0,5	2	0,9	0,3	0,3	0,3	2670,3	2545	2769	66,2508
15.	1000	60	0,8	2,5	5	0,6	0,6	0,3	0,1	2548,7	2470	2705	81,8928
16.	1000	60	0,2	2,5	5	0,9	0,3	0,3	0,3	2807,05	2695	2915	55,8951
17.	1000	60	0,2	2,5	2	0,6	0,6	0,6	0,1	2953,1	2697	3170	102,8028
18.	1000	120	0,2	0,5	5	0,9	0,6	0,3	0,1	2527,75	2469	2710	78,2613
19.	400	120	0,8	0,5	5	0,6	0,6	0,6	0,1	2536,7	2442	2705	79,9211
20.	1000	120	0,2	2,5	5	0,6	0,3	0,6	0,1	2807,65	2645	2912	80,016
21.	1000	60	0,8	2,5	2	0,9	0,3	0,6	0,3	2818	2729	2902	52,0395
22.	1000	60	0,2	0,5	5	0,6	0,6	0,6	0,3	2567	2500	2710	46,3794
23.	400	120	0,2	2,5	5	0,6	0,6	0,3	0,3	2831,85	2645	3005	84,5603
24.	1000	60	0,2	0,5	2	0,9	0,3	0,3	0,1	2725,75	2575	2875	75,538
25.	400	120	0,2	0,5	2	0,6	0,6	0,3	0,1	2770,6	2640	2880	55,2634
26.	400	60	0,2	0,5	5	0,6	0,3	0,3	0,1	2599,75	2490	2725	93,4637
27.	400	120	0,8	2,5	5	0,9	0,3	0,3	0,1	2532,75	2502	2705	54,6615
28.	1000	120	0,8	2,5	2	0,6	0,3	0,3	0,1	2592,2	2520	2705	74,759
29.	400	60	0,2	2,5	5	0,9	0,6	0,6	0,1	2864,65	2750	2977	71,9271
30.	400	60	0,8	2,5	2	0,9	0,6	0,3	0,1	2579,6	2507	2725	91,5725
31.	1000	120	0,2	0,5	2	0,6	0,3	0,6	0,3	2880	2790	2950	42,2711
32.	400	60	0,8	2,5	5	0,6	0,3	0,6	0,3	2529,85	2504	2705	60,296

Tabela 6.2. Parametri odziva sa usvojenim limitima i ciljnim vrednostima

Odziv	Donji limit	Ciljna vrednost	Gornji limit	Težinski koeficijent	Koeficijent važnosti
Minimalni TWMC	2420	2440	2460	1	1
Srednji TWMC	2490	2510	2520	1	1

Tabela 6.3. Optimalni ulazni parametri GCSA-Par1

MaxIt	FlockSize	AP	fl	TourSize	kc ₁	kc ₂	km ₁	km ₂	MIN TWMC	SV TWMC	Indeks kompozitne poželjnosti
996,674	101,414	0,8	0,716197	5	0,636569	0,6	0,3	0,3	2440	2510	1



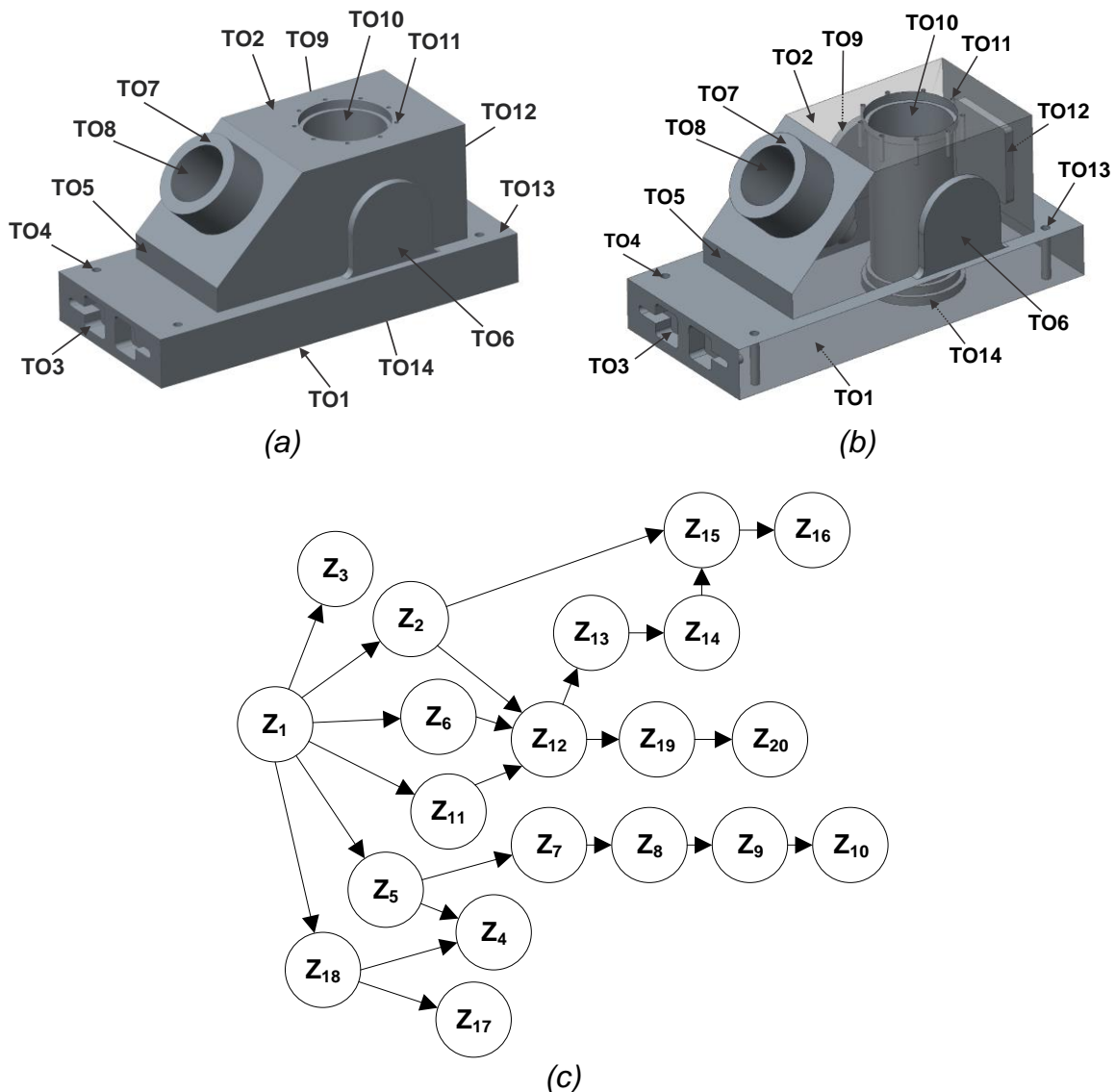
Slika 6.1. Optimizacioni grafik uticaja ulaznih parametara na predviđeni odziv (MIN i SV troškova TWMC)

6.1.1 Prva eksperimentalna studija

Prvu eksperimentalnu studiju za verifikaciju performansi predloženog GCSA algoritma čini jedan reprezentativni deo prizmatičnog oblika od ukupno četiri koliko će biti razmotreno u ovom eksperimentalnom istraživanju. U literaturi ovaj primer često je usvajan u svrhu verifikacije različitih optimizacionih metoda koje su primenu pronašle u oblasti OTP. Guo sa sar. [13] se izdvajaju među prvim istraživačima koji su predložili prizmatični deo koji poseduje 14 tipskih tehnoloških oblika za čiju obradu je neophodno realizovati nekoliko operacija obrade sa ukupno 20 zahvata obrade. Trodimenzionalni solid model dela sa konceptualnim i transparentnim efektom senčenja kao i odgovarajući graf prethođenja grupno su predstavljeni na slici 6.2. Dvodimenzionalni radionički crtež ovog dela može se videti u [44]. Matrica prethođenja data je na slici 6.3. Relevantne tehnološke informacije koje se tiču tipskih tehnoloških oblika, zahvata obrade i tehnoloških resursa vezanih za prvi prizmatični deo date su u tabeli 6.4. Informacije o karakterističnim odnosima prethođenja između tipskih oblika, odnosno zahvata obrade prikazane su u tabeli 6.5. Komponente troškova mašina i alata za prvi prizmatični deo date su u tabeli 6.6. Sve navedene informacije preuzete su iz [13].

Verifikacija metaheurističkih algoritama na određenom test modelu izvodi se u različitim tehnološkim uslovima. Kada je u pitanju prvi eksperiment, verifikacija algoritama izvršena je za tri različita uslova i to:

- 1) Optimizacija tehnoloških procesa se vrši u slučaju kad su dostupni svi tehnološki resursi, tačnije sve varijante mašina, alata i SPA. Težinski koeficijenti $w_1 - w_5$ u sklopu definisane funkcije cilja u jednačini (18) jednaki su 1.
- 2) Optimizacija tehnoloških procesa vrši se bez razmatranja alata, tačnije troškovi korišćenja alata i troškovi promena alata se ne uzimaju u obzir. U tom slučaju, u jednačini (18), težinski koeficijenti w_2 i w_4 jednaki su 0, dok su w_1 , w_3 i w_5 jednaki 1.
- 3) Slično uslovu 2, optimizacija tehnoloških procesa vrši se kada pored troškova korišćenja alata i promena alata, mašina M2 i alat A8 takođe nisu dostupni. Težinski koeficijenti su definisani na isti način, dok su navedena mašina i navedeni alat uklonjeni iz baze sa ulaznim podacima.



Slika 6.2. Trodimenzionalni solid model u konceptualnom (a) i transparentnom (b) prikazu i graf prethođenja zahvata (c) za prvi eksperiment

	Z ₁	Z ₂	Z ₃	Z ₄	Z ₅	Z ₆	Z ₇	Z ₈	Z ₉	Z ₁₀	Z ₁₁	Z ₁₂	Z ₁₃	Z ₁₄	Z ₁₅	Z ₁₆	Z ₁₇	Z ₁₈	Z ₁₉	Z ₂₀
Z ₁	0	1	1	0	1	1	0	0	0	0	1	0	0	0	0	0	0	1	0	0
Z ₂	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1	0	0	0	0	0
Z ₃	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Z ₄	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Z ₅	0	0	0	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
Z ₆	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
Z ₇	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
Z ₈	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0
Z ₉	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0
Z ₁₀	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Z ₁₁	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
Z ₁₂	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	1	0
Z ₁₃	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0
Z ₁₄	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0
Z ₁₅	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0
Z ₁₆	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Z ₁₇	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Z ₁₈	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
Z ₁₉	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
Z ₂₀	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Slika 6.3. Matrica prethođenja za prvi eksperiment

Tabela 6.4. Informacije o tipskim oblicima, zahvatima, resursima i ograničenjima za prvi eksperiment

Tipski tehnološki oblici	Zahvati	Mašine{ }	Alati{ }	SPA{ }
TO1 – Ravna površina (baza)	Glodanje (Z1)	M2,M3	A6,A7,A8	+z
TO2 – Ravna površina	Glodanje (Z2)	M2,M3	A6,A7,A8	-z
TO3 – 2 džepa kao umnoženi TO	Glodanje (Z3)	M2,M3	A6,A7,A8	+x
TO4 – 4 otvora kao umnoženi TO	Bušenje (Z4)	M1,M2,M3	A2	+z,-z
TO5 – Stepjenik	Glodanje (Z5)	M2,M3	A6,A7	+x,-z
TO6 - Rebro	Glodanje (Z6)	M2,M3	A7,A8	+y,-z
TO7 - Ispupčenje	Glodanje (Z7)	M2,M3	A7,A8	-a
TO8 - Rupa	Bušenje (Z8)	M1,M2,M3	A2,A3,A4	-a
	Proširivanje (Z9)	M1,M2,M3	A9	
	Razvrtanje (Z10)	M2,M3	A10	
TO9 - Rebro	Glodanje (Z11)	M2,M3	A7,A8	-y,-z
TO10 - Otvor	Bušenje (Z12)	M1,M2,M3	A2,A3,A4	-z
	Proširivanje (Z13)	M1,M2,M3	A9	
	Razvrtanje (Z14)	M3,M4	A10	
TO11 – 9 rupa kao umnoženi TO	Bušenje (Z15)	M1,M2,M3	A1	-z
	Urezivanje navoja (Z16)	M1,M2,M3	A5	
TO12 - Džep	Glodanje (Z17)	M2,M3	A7,A8	-x
TO13 - Stepjenik	Glodanje (Z18)	M2,M3	A6,A7	-x,-z
TO14 - Rupa	Proširivanje (Z19)	M1,M2,M3	A9	+z
	Razvrtanje (Z20)	M3,M4	A10	

Tabela 6.5. Odnosi prethođenja među tipskim oblicima i zahvatima za prvi eksperiment

Tipski oblik	Naziv zahvata	Opis odnosa prethođenja
TO1	Glodanje (Z1)	TO1 (Z1) je bazna površina i stoga se treba obraditi pre ostalih tipskih oblika
TO2	Glodanje (Z2)	TO2 (Z2) obrađuje se pre TO10 (Z12, Z13, Z14) i TO11 (Z15, Z16)
TO3	Glodanje (Z3)	
TO4	Bušenje (Z4)	
TO5	Glodanje (Z5)	TO5 (Z5) obrađuje se pre TO4 (Z4) i TO7 (Z7) zbog bazne površine i jednostavnije obrade
TO6	Glodanje (Z6)	TO6 (Z6) obrađuje se pre TO10 (Z12, Z13, Z14) zbog bazne površine
TO7	Glodanje (Z7)	TO7 (Z7) obrađuje se pre TO8 (Z8, Z9, Z10) zbog bazne površine i jednostavnije obrade
TO8	Bušenje (Z8)	Z8 realizuje se pre Z9 i Z10; Z9 realizuje se pre Z10 zbog fiksnog redosleda obrade
	Proširivanje (Z9)	
	Razvrtanje (Z10)	
TO9	Glodanje (Z11)	TO9 (Z11) obrađuje se pre TO10 (Z12, Z13, Z14) zbog bazne površine
TO10	Bušenje (Z12)	Z12 realizuje se pre Z13 i Z14; Z13 realizuje se pre Z14; TO10 (Z12, Z13, Z14) obrađuje se pre TO11 (Z15, Z16) zbog bazne površine; Z12 obrađuje se pre TO14 (Z19 i Z20)
	Proširivanje (Z13)	
	Razvrtanje (Z14)	
TO11	Bušenje (Z15)	Z15 realizuje se pre Z16 zbog fiksnog redosleda obrade
	Urezivanje navoja (Z16)	
TO12	Glodanje (Z17)	
TO13	Glodanje (Z18)	TO13 (Z18) obrađuje se pre Z4 i Z17 zbog jednostavnije obrade
TO14	Proširivanje (Z19)	Z19 realizuje se pre Z20 zbog fiksnog redosleda obrade
	Razvrtanje (Z20)	

Tabela 6.6. Informacije o troškovima za prvi eksperiment

Redni broj	Naziv troškova	Novčana vrednost
Troškovi rada (korišćenja) mašina:		
M1	Stubna bušilica	10
M2	Troosna vertikalna glodalica	40
M3	CNC troosna vertikalna glodalica	100
M4	Bušilica-glodalica	60
Troškovi rada (korišćenja) alata:		
A1	Burgija 1	7
A2	Burgija 2	5
A3	Burgija 3	3
A4	Burgija 4	8
A5	Ureznik	7

Tabela 6.6. Informacije o troškovima za prvi eksperiment (nastavak)

Redni broj	Naziv troškova	Novčana vrednost
Troškovi rada (korišćenja) alata:		
A6	Glodalo 1	10
A7	Glodalo 2	15
A8	Glodalo 3	30
A9	Proširivač	15
A10	Razvrtač	20
Troškovi promene:		
MCI	Mašina	160
TCI	Alata	20
SCI	Stezanja (smerova prilaza alata)	100

Sa ciljem sticanja relevantnih informacija o performansama modifikovanog CSA algoritma, izvršeno je poređenje dobijenih rezultata sa rezultatima koji su dobijeni primenom tradicionalnih algoritama i savremenih hibridnih i modifikovanih metoda koje su analizirane u literaturi. Uporedni rezultati za prvi eksperiment dati su u tabeli 6.7. Kao što se može uočiti, prikazani su minimalni, srednji i maksimalni troškovi TWMC ostvareni primenom GCSA algoritma sa ulaznim parametrima dobijenim iz dva plana eksperimenta i 13 drugih metaheurističkih pristupa iz literature, i to: HGASA [139], PSO [13], HGGA [12], HBMO [74], ACO [44], TSACO [72], cPSO [21], mACO [154], ESGA [71], IGA [141], FSDPSO [70], CPAGA [142] i IWD [157].

Tabela 6.7. Uporedni rezultati GCSA i drugih metaheurističkih algoritama za prvi eksperiment

Algoritam	Uslov 1			Uslov 2			Uslov 3		
	SV	MAX	MIN	SV	MAX	MIN	SV	MAX	MIN
*GCSA-Par1	2516,9	2700	2502	2047	2080	2020	2500	2500	2500
*GCSA-Par2	2499	2509	2444	2065	2080	2020	2500	2500	2500
HGASA	2546	2585	2527	2120	2120	2120	-	2600	2590
PSO	2680,5	-	2535	-	-	-	-	-	-
HGGA	-	-	2527	-	-	2120	-	-	2590
HBMO	2543,5	2557	2525	2098	2120	2090	2592,4	2600	2590
ACO	2456,1	2527	2435	2115,4	2380	2090	2600	2740	2580
TSACO	2552,4	2557	2525	2120,5	2380	2090	2600,8	2740	2590
cPSO	2629	2687	2520	2100	2220	2020	2515	2600	2500
mACO	2666	-	2530	2115	-	2090	-	-	-
ESGA	2539,1	2562	2530	-	-	2090	-	-	2590
IGA	2538,7	2547	2530	2120	2111	2090	-	-	-
FSDPSO	2532	-	2530	2090	-	2090	-	-	-
CPAGA	2530,5	2535	2530	2090	2090	2090	2500	2500	2500
IWD	2553,5	2554	2527	2123	2380	2090	2615,3	2740	2590

Eksperimentalni rezultati dobijeni primenom GCSA algoritma za sva tri tehnološka uslova kao i najbolji tehnološki procesi sa minimalnim troškovima obrade predstavljeni su u tabelama 6.8 i 6.9, za parametre predloženih algoritama „GCSA-Par1“ i „GCSA-Par2“ respektivno.

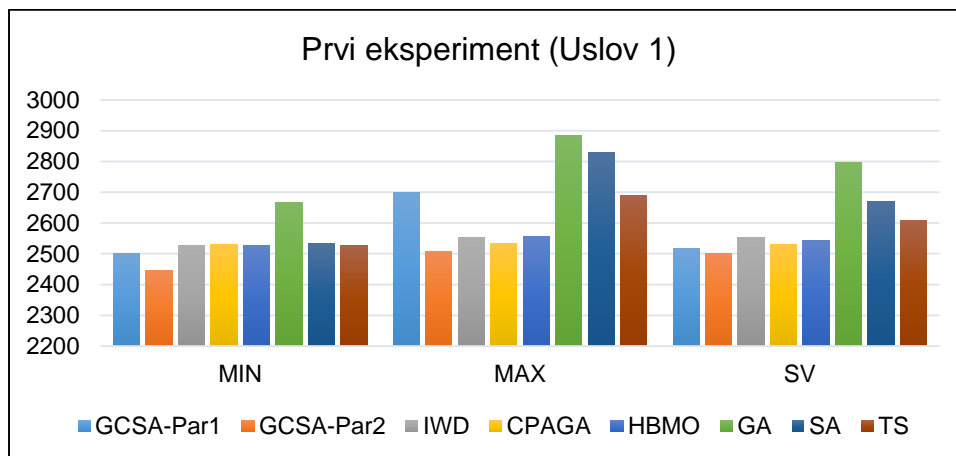
Tabela 6.8. Najbolji tehnološki procesi za prvi eksperiment i tri tehnološka uslova sa ulaznim parametrima u okviru „GCSA-Par1“

Uslov 1	Indeksi	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
	Zahvati	1	3	18	5	2	17	6	11	4	12	13	19	7	8	9	10	20	14	15	16
	Mašine	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	4	4	4	2	2
	Alati	6	6	6	6	6	7	7	7	2	2	9	9	7	3	9	10	10	10	1	5
	SPA	+z	+x	-z	-z	-z	-z	-z	-z	-z	-z	-z	+z	-a	-a	-a	-a	+z	-z	-z	-z
TMC=860, TMCC=320, NMC=2, TTC=242, TTCC=180, NTC=9, TSC=900, NSC=9, TWMC=2502																					
Uslov 2	Indeksi	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
	Zahvati	1	6	18	17	5	2	11	12	4	13	7	8	9	3	19	20	10	14	15	16
	Mašine	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	4	4	4	1	1
	Alati	7	7	7	7	6	8	8	4	2	9	8	4	9	6	9	10	10	10	1	5
	SPA	+z	-z	-z	-z	-z	-z	-z	-z	-z	-z	-a	-a	-a	+x	+z	+z	-a	-z	-z	-z
TMC=800, TMCC=320, NMC=2, TSC=900, NSC=9, TWMC=2020																					
Uslov 3	Indeksi	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
	Zahvati	1	3	5	7	8	9	10	2	6	11	18	12	17	13	14	15	16	4	19	20
	Mašine	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3
	Alati	6	6	6	7	4	9	10	6	7	7	6	4	7	9	10	1	5	2	9	10
	SPA	+z	+x	+x	-a	-a	-a	-a	-z	-z	-z	-z	-z	-z	-z	-z	-z	-z	-z	+z	+z
TMC=2000, TMCC=0, NMC=0, TTC=250, TTCC=3200, NTC=16, TSC=500, NSC=5, TWMC=2500																					

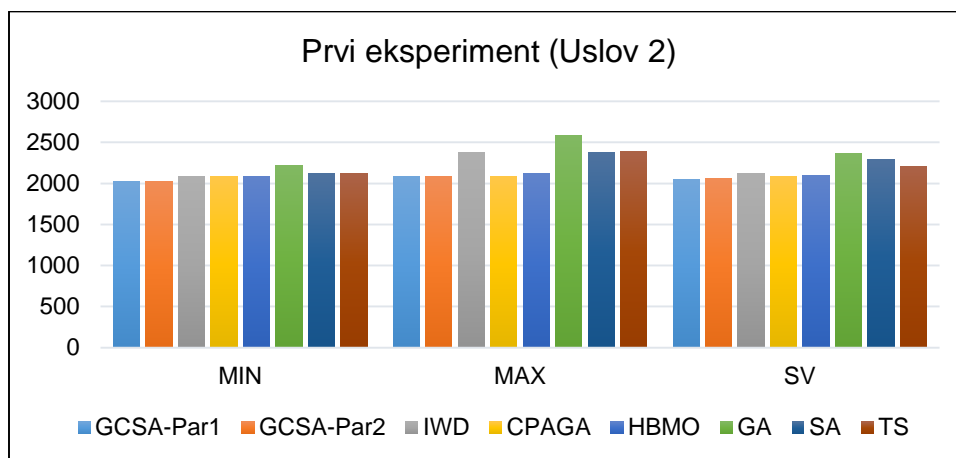
Tabela 6.9. Najbolji tehnološki procesi za prvi eksperiment i tri tehnološka uslova sa ulaznim parametrima u okviru „GCSA-Par2“

Uslov 1	Indeksi	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
	Zahvati	1	3	5	18	2	11	17	6	4	12	13	19	7	8	9	10	20	14	15	16
	Mašine	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	4	4	4	1	1
	Rezni alati	6	6	6	6	6	7	7	7	2	2	9	9	7	2	9	10	10	10	1	5
	PPA	-z	+x	-z	-z	-z	-z	-z	-z	-z	-z	-z	+z	-a	-a	-a	-a	+z	-z	-z	-z
TMC=800, TMCC=320, NMC=2, TTC=244, TTCC=180, NTC=9, TSC=900, NSC=9, TWMC=2444																					
Uslov 2	Indeksi	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
	Zahvati	1	11	2	6	12	13	5	18	4	17	3	19	7	8	9	10	20	14	15	16
	Mašine	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	4	4	4	1	1
	Rezni alati	7	7	7	7	4	9	6	6	2	7	6	9	7	4	9	10	10	10	1	5
	PPA	+z	-z	-z	-z	-z	-z	-z	-z	-z	-z	+x	+z	-a	-a	-a	-a	+z	-z	-z	-z
TMC=800, TMCC=320, NMC=2, TSC=900, NSC=9, TWMC=2020																					
Uslov 3	Indeksi	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
	Zahvati	1	3	5	7	8	9	10	2	6	11	18	12	17	13	14	15	16	4	19	20
	Mašine	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3
	Rezni alati	6	6	6	7	4	9	10	6	7	7	6	4	7	9	10	1	5	2	9	10
	PPA	+z	+x	+x	-a	-a	-a	-a	-z	-z	-z	-z	-z	-z	-z	-z	-z	-z	-z	+z	+z
TMC=2000, TMCC=0, NMC=0, TTC=250, TTCC=3200, NTC=16, TSC=500, NSC=5, TWMC=2500																					

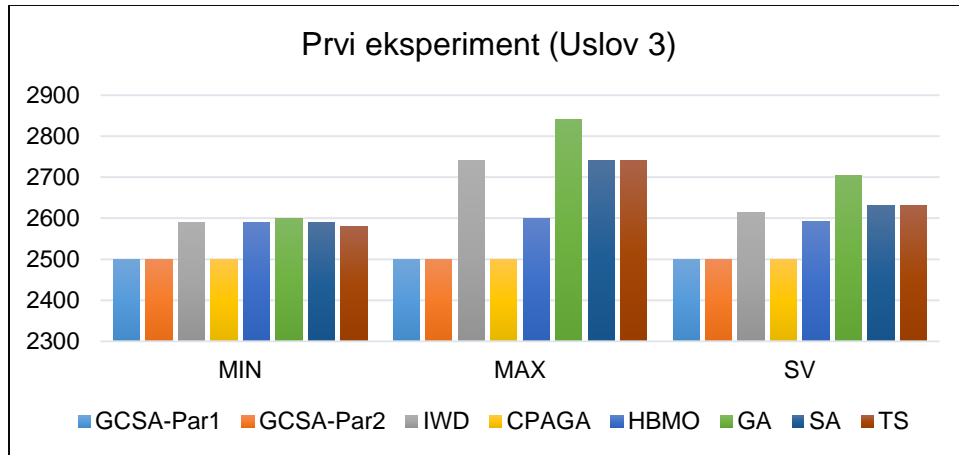
Svi rezultati predstavljeni u tabeli 6.7 grafički su prikazani na dijagramima koji se nalaze na slikama 6.4, 6.5 i 6.6. Horizontalna osa dijagrama prikazuje izlazne rezultate u vidu minimalne, maksimalne i srednje vrednosti troškova TWMC koji su posebno ostvarili svi metaheuristički algoritmi u studiji. Vertikalna osa dijagrama sa druge strane označava skalu ukupnih troškova obrade TWMC.



Slika 6.4. Dijagrami uporednih rezultata ostvareni različitim metaheurističkim algoritmima za prvi eksperiment i prvi tehnološki uslov



Slika 6.5 Dijagrami uporednih rezultata ostvareni različitim metaheurističkim algoritmima za prvi eksperiment i drugi tehnološki uslov



Slika 6.6 Dijagrami uporednih rezultata ostvareni različitim metaheurističkim algoritmima za prvi eksperiment i treći tehnološki uslov

Drugi deo uporedne analize vezane za prvi eksperiment odnosi se na prikaz krivih konvergencije. Na osnovu ovih krivih može se pratiti generisanje rezultata tokom iterativnog procesa i na taj način utvrditi i uporediti približavanje algoritama globalnom optimumu. U tu svrhu testirane su performanse „GCSA-Par2“. Performanse su upoređene sa performansama algoritama HGGA, PSO, GA, mPSO i CSA. Detaljan prikaz uporednih vrednosti funkcija cilja f_c i troškova TWMC sa statističkim pokazateljima u vidu standardnog odstupanja i srednje vrednosti troškova TWMC za svih 20 pokretanja algoritma mogu se videti u tabelama 6.10, 6.11 i 6.12. Krive konvergencije za prvi eksperiment i tri tehnološka uslova nalaze se na slici 6.7. Dve krive konvergencije su prikazane za svaki tehnološki uslov.

Tabela 6.10. Izlazni rezultati prvog eksperimenta za prvi tehnološki uslov u 20 ponavljanja

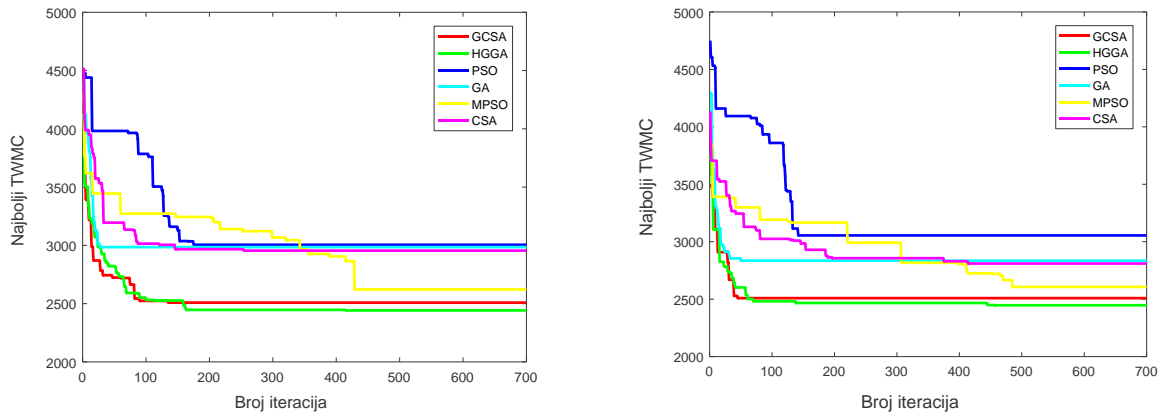
Br.	Maksimizacija funkcije cilja fc i minimizacija troškova TWMC															
	GCSA-Par2		HGGA		MPSO		PSO		GA		CSA					
	fc	TWMC	fc	TWMC	fc	TWMC	fc	TWMC	fc	TWMC	fc	TWMC				
1.	0,0004	2502	0,00039	2565	0,00038	2660	0,00038	2610	0,00036	2794	0,00034	2910				
2.	0,0004	2507	0,00041	2435	0,00038	2615	0,00038	2640	0,00034	2939	0,00033	2995				
3.	0,0004	2507	0,00041	2435	0,00036	2740	0,00038	2605	0,00035	2827	0,00032	3100				
4.	0,0004	2502	0,00041	2465	0,00036	2765	0,00039	2590	0,00032	3127	0,00032	3135				
5.	0,0004	2504	0,00038	2615	0,00036	2742	0,00032	3115	0,00032	3130	0,00034	2927				
6.	0,0004	2504	0,00041	2447	0,00036	2765	0,00036	2740	0,0003	3365	0,00034	2930				
7.	0,0004	2509	0,00039	2565	0,00036	2815	0,00038	2600	0,0003	3302	0,00033	3045				
8.	0,0004	2502	0,00041	2435	0,00036	2805	0,00038	2610	0,00028	3525	0,00033	3010				
9.	0,0004	2502	0,00041	2435	0,00037	2715	0,00038	2625	0,00036	2810	0,00033	3065				
10.	0,0004	2509	0,00041	2442	0,00038	2650	0,00038	2655	0,00036	2794	0,00033	3010				
11.	0,0004	2504	0,00039	2585	0,00038	2660	0,00039	2580	0,00036	2772	0,00032	3105				
12.	0,0004	2507	0,00038	2600	0,00037	2680	0,00038	2635	0,00034	2915	0,00034	2950				
13.	0,00041	2444	0,00039	2565	0,00037	2710	0,00038	2655	0,00031	3235	0,00032	3155				
14.	0,0004	2502	0,0004	2487	0,00038	2630	0,00038	2665	0,00037	2732	0,00031	3207				
15.	0,0004	2507	0,0004	2480	0,00037	2680	0,00039	2595	0,00035	2824	0,00032	3162				
16.	0,0004	2509	0,00041	2462	0,00037	2710	0,00038	2640	0,0003	3297	0,00035	2890				
17.	0,0004	2504	0,0004	2475	0,00037	2735	0,00038	2655	0,00029	3417	0,00033	3050				
18.	0,0004	2502	0,00041	2465	0,00037	2690	0,00038	2645	0,00036	2752	0,00034	2905				
19.	0,00041	2449	0,00041	2465	0,00036	2765	0,00037	2670	0,00035	2874	0,00033	3007				
20.	0,0004	2504	0,00041	2467	0,00038	2660	0,00038	2665	0,00039	2580	0,00035	2830				
Statistički parametri (Standardna devijacija rezultata i srednja vrednost troškova)																
	σ	SV	σ	SV	σ	SV	σ	SV	σ	SV	σ	SV				
	18,15	2499	60,41	2494,5	54,96	2709,6	110,38	2659,75	266,53	3000,55	101,17	3019,4				

Tabela 6.11. Izlazni rezultati prvog eksperimenta za drugi tehnološki uslov u 20 ponavljanja

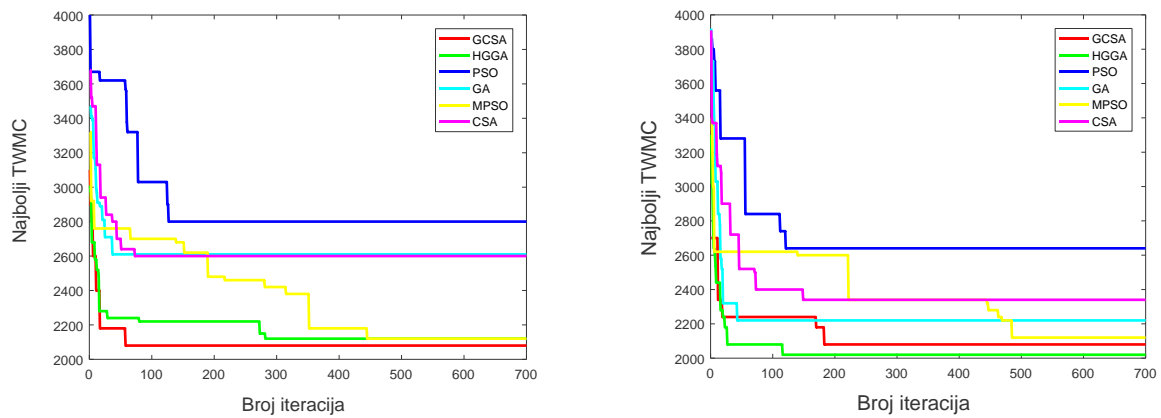
Br.	Maksimizacija funkcije cilja fc i minimizacija troškova TWMC															
	GCSA-Par2		HGGA		MPSO		PSO		GA		CSA					
	fc	TWMC	fc	TWMC	fc	TWMC	fc	TWMC	fc	TWMC	fc	TWMC				
1.	0,00048	2080	0,0005	2020	0,00047	2120	0,0004	2500	0,00039	2230	0,00042	2380				
2.	0,00048	2080	0,00049	2060	0,00048	2080	0,00048	2080	0,00048	2580	0,00044	2260				
3.	0,00048	2080	0,00046	2160	0,00047	2120	0,0004	2500	0,00037	2080	0,00042	2400				
4.	0,00048	2080	0,0005	2020	0,00047	2120	0,00047	2120	0,00037	2720	0,00042	2400				
5.	0,0005	2020	0,00047	2120	0,00045	2220	0,00047	2120	0,00047	2700	0,00039	2560				
6.	0,00048	2080	0,0005	2020	0,00047	2140	0,00049	2060	0,00044	2140	0,00042	2400				
7.	0,00048	2080	0,00047	2120	0,00047	2140	0,0005	2020	0,00047	2280	0,00042	2400				
8.	0,0005	2020	0,0005	1990	0,00045	2200	0,00049	2060	0,00043	2140	0,0004	2500				
9.	0,00048	2080	0,00047	2120	0,00045	2200	0,00048	2080	0,00048	2340	0,00038	2640				
10.	0,0005	2020	0,00047	2120	0,00047	2120	0,00047	2120	0,00045	2080	0,00042	2400				
11.	0,00048	2080	0,0005	2020	0,00048	2080	0,0005	2020	0,00044	2200	0,00041	2460				
12.	0,0005	2020	0,00047	2120	0,00048	2080	0,00047	2120	0,00045	2260	0,0004	2500				
13.	0,0005	2020	0,00049	2060	0,00047	2140	0,0004	2500	0,00041	2240	0,00042	2400				
14.	0,00048	2080	0,0005	2020	0,00047	2140	0,00048	2080	0,00038	2440	0,00039	2560				
15.	0,00048	2080	0,0005	1990	0,00049	2060	0,00049	2060	0,00045	2600	0,00042	2400				
16.	0,00048	2080	0,0005	1990	0,00047	2140	0,00047	2120	0,00042	2220	0,0004	2500				
17.	0,00048	2080	0,00048	2080	0,0005	2020	0,00049	2060	0,0004	2360	0,0004	2500				
18.	0,00048	2080	0,00047	2120	0,00049	2060	0,0005	2020	0,00044	2470	0,00042	2400				
19.	0,00048	2080	0,0005	2020	0,00046	2180	0,00047	2120	0,00036	2260	0,00042	2400				
20.	0,00048	2080	0,0005	2020	0,00047	2120	0,00049	2060	0,00039	2780	0,00042	2400				
Statistički parametri (Standardna devijacija rezultata i srednja vrednost troškova)																
σ	26,66	2065	53,8	2059,5	50,04	2124	154,46	2141	212,45	2356	82,29	2443				
SV																

Tabela 6.12. Izlazni rezultati prvog eksperimenta za treći tehnološki uslov u 20 ponavljanja

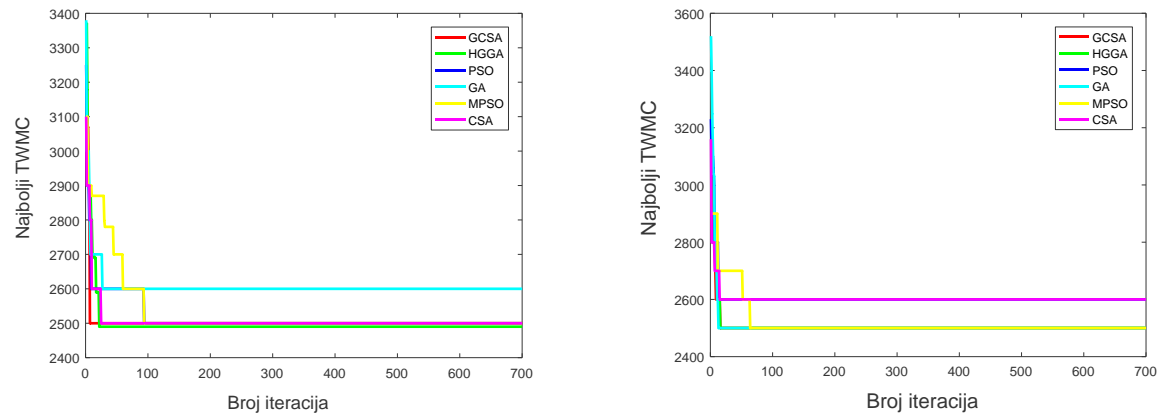
Br.	Maksimizacija funkcije cilja fc i minimizacija troškova TWMC																						
	GCSA-Par2		HGGA		MPSO		PSO		GA		CSA												
	fc	TWMC	fc	TWMC	fc	TWMC	fc	TWMC	fc	TWMC	fc	TWMC											
1.	0,0004	2500	0,0004	2500	0,0004	2500	0,0004	2500	0,0004	2500	0,0004	2500	0,0004	2500									
2.	0,0004	2500	0,0004	2500	0,0004	2500	0,0004	2500	0,0004	2500	0,0004	2500	0,0004	2500									
3.	0,0004	2500	0,0004	2500	0,0004	2500	0,0004	2500	0,0004	2500	0,0004	2500	0,0004	2500									
4.	0,0004	2500	0,0004	2500	0,0004	2500	0,0004	2500	0,0004	2500	0,0004	2500	0,0004	2500									
5.	0,0004	2500	0,0004	2500	0,0004	2500	0,00038	2600	0,0004	2500	0,00038	2600	0,00038	2600									
6.	0,0004	2500	0,0004	2500	0,0004	2500	0,00038	2600	0,0004	2500	0,00038	2600	0,0004	2500									
7.	0,0004	2500	0,0004	2500	0,0004	2500	0,0004	2500	0,0004	2500	0,0004	2500	0,0004	2500									
8.	0,0004	2500	0,00038	2600	0,0004	2500	0,0004	2500	0,0004	2500	0,0004	2500	0,0004	2500									
9.	0,0004	2500	0,0004	2500	0,0004	2500	0,0004	2500	0,0004	2500	0,0004	2500	0,00038	2600									
10.	0,0004	2500	0,0004	2490	0,0004	2500	0,00038	2600	0,0004	2500	0,0004	2500	0,0004	2500									
11.	0,0004	2500	0,00039	2590	0,0004	2500	0,0004	2500	0,0004	2500	0,0004	2490	0,0004	2500									
12.	0,0004	2500	0,00039	2580	0,0004	2500	0,00038	2600	0,0004	2500	0,00038	2600	0,00038	2600									
13.	0,0004	2500	0,0004	2500	0,0004	2500	0,0004	2500	0,0004	2500	0,0004	2500	0,0004	2500									
14.	0,0004	2500	0,0004	2500	0,0004	2500	0,00038	2600	0,0004	2500	0,00038	2600	0,00038	2600									
15.	0,0004	2500	0,0004	2500	0,0004	2500	0,0004	2500	0,0004	2500	0,0004	2500	0,00038	2600									
16.	0,0004	2500	0,00038	2600	0,0004	2500	0,00038	2600	0,0004	2500	0,00038	2600	0,00038	2600									
17.	0,0004	2500	0,0004	2490	0,0004	2500	0,0004	2500	0,0004	2500	0,0004	2490	0,0004	2500									
18.	0,0004	2500	0,00037	2700	0,0004	2500	0,0004	2500	0,0004	2500	0,0004	2500	0,00038	2600									
19.	0,0004	2500	0,0004	2500	0,0004	2500	0,0004	2500	0,0004	2500	0,0004	2500	0,00038	2600									
20.	0,0004	2500	0,0004	2500	0,0004	2500	0,0004	2500	0,0004	2500	0,0004	2500	0,00038	2600									
Statistički parametri (Standardna devijacija rezultata i srednja vrednost troškova)																							
σ	0	SV	2500	σ	54,58	SV	2527,5	σ	0	SV	2500	σ	45,83	SV	2530	σ	3,57	SV	2498,5	σ	50	SV	2550



(a) Prvi tehnološki uslov



(b) Drugi tehnološki uslov



(c) Treći tehnološki uslov

Slika 6.7. Krive konvergencije za prvi eksperiment i tri tehnološka uslova

6.1.1.1 Diskusija rezultata prve eksperimentalne studije

Rezultati prve studije, odnosno optimizacije prizmatičnog dela usvojenog iz [13] potvrđuju efikasnost predloženog GCSA metaheurističkog algoritma koji je pokazao veoma dobre rezultate nadmašivši pri tom rezultate koji su ostvarili mnogi drugi tradicionalni, kao i novorazvijeni metaheuristički algoritmi. U okviru eksperimenta, GCSA algoritam izvršen je u 20 ponavljanja kako bi se verifikovala konzistentnost dobijenih rezultata.

Kada je u pitanju prvi deo eksperimentalne analize, data je kratka diskusija rezultata GCSA algoritma sa ulaznim parametrima koji su dobijeni na osnovu delimičnog faktornog plana eksperimenta, a kao rezultat formiran je algoritam „GCSA-Par1“. Prema informacijama za prvi tehnološki uslov iz tabele 6.7, vrednost minimalnih troškova obrade iznosi 2502 novčane jedinice koja se u 20 ponavljanja ostvarila 9 puta. Pored nje, druga minimalna vrednost troškova obrade je 2507 koja se pojavila 7 puta u 20 ponavljanja. U pogledu minimalnih i srednjih TWMC koji iznose 2507 i 2516,9 novčanih jedinica respektivno, može se zaključiti da „GCSA-Par1“ nadmašuje performanse preostalih algoritama u uporednoj studiji, kako tradicionalnih tako i savremenih algoritama. Najlošiji rezultat ostvaren od strane GCSA blago odstupa u odnosu na nekoliko najlošijih rezultata drugih algoritama. Kada su u pitanju drugi i treći tehnološki uslov, „GCSA-Par1“ je pokazao slične performanse. Kod drugog uslova, minimalna, maksimalna i srednja vrednost troškova TWMC iznose 2020, 2080 i 2047, respektivno. U poređenju sa drugim pristupima, „GCSA-Par1“ je pokazao superiornost u pogledu sva tri statistička parametra. Kod trećeg uslova, troškovi TWMC iznose 2500 novčanih jedinica u svih 20 ponavljanja čime je „GCSA-Par1“ ostvario najbolju konzistentnost rezultata u prvom eksperimentu.

Sa druge strane, ostvareni su rezultati pomoću drugog predloženog algoritma „GCSA-Par2“ koji je formiran na osnovu manuelno selektovanih ulaznih parametara. Predloženi „GCSA-Par2“ algoritam je pokazao za nijansu bolje performanse od svog prethodnika „GCSA-Par1“. Ukupni izlazni rezultati prvog eksperimenta i prvi tehnološki uslov detaljno su prikazani u tabeli 6.10. Minimalna vrednost TWMC iznosi 2444 novčane jedinice koja se pojavila samo jednom u 20 ponavljanja algoritma. Takođe, sledeća minimalna vrednost od 2449 novčanih jedinica se pojavila samo u jednoj egzekuciji algoritma. Najkonzistentniji rezultati su TWMC od 2502 novčane jedinice koja je dobijena šest puta, zatim 2504 jedinice su se pojavile pet puta dok su isti troškovi od 2507 novčanih jedinica dobijeni četiri puta. Kao što se može videti u tabeli 6.10, predloženi „GCSA-Par2“ savremeni algoritam se, pored hibridnog HGGA pristupa pokazao najefikasnijim u generisanju optimalnih tehnoloških procesa. U pogledu srednje vrednosti TWMC, hibridni HGGA pristup pokazuje neznatno bolji rezultat, dok se „GCSA-Par2“ sa druge strane izdvaja po pitanju standardne devijacije rezultata koja potvrđuje mnogo manja odstupanja u ostvarenim troškovima TWMC za 20 egzekucija algoritma.

Kada je u pitanju drugi tehnološki uslov, u tabeli 6.7 se može uočiti da rezultati „GCSA-Par2“ nadmašuju ostale rezultate iz literaturnih izvora u pogledu tri relevantna statistička parametra – minimalnu, maksimalnu i srednju vrednost TWMC. Izlazni rezultati prvog eksperimenta za drugi tehnološki uslov detaljno su predstavljeni u tabeli 6.11. Može se uočiti da se ukupna vrednost troškova od 2020 novčanih jedinica pojavila četiri puta u

20 egzekucija algoritma. Ponovo se pogledu ostvarene standardne devijacije i srednje vrednosti troškova „GCSA-Par2“ i HGGA pristupi izdvajaju u odnosu na ostale algoritme.

Na kraju, za treći tehnološki uslov u prvom eksperimentu, GCSA je pokazao odlične performanse i zajedno sa savremenim metaheurističkim pristupom koji je opisan u [142] ostvaruje najbolju konzistentnost rezultata što se može videti u tabeli 6.7. Detaljan prikaz izlaznih rezultata za treći uslov optimizacije nalazi se u tabeli 6.12. S obzirom na činjenicu da treći uslov umanjuje težinu kriterijuma koji se optimizuje, to omogućava i drugim algoritmima da izraze svoje kapacitete za rešavanje problema OTP. Može se uočiti da GCSA algoritam ostvaruje maksimalan učinak pri čemu je dobijena najbolja srednja vrednost troškova sa minimalnom devijacijom rezultata. U ovoj instanci izdvaja se i mPSO kao i tradicionalni GA ali sa malom standardnom devijacijom rezultata.

Grafički rezultati prikazani na krivama konvergencije sa slike 6.7 pokazuju da GCSA, algoritam pored hibridnog HGGA pristupa, konvergira ka optimumu brže od ostalih algoritama u uporednom prikazu. Neminovno je da će modifikovani i hibridni pristupi pokazivati veoma dobar učinak u ovim studijama i stoga je cilj bio verifikovati performanse GCSA poređenjem sa tradicionalnim, a i hibridnim pristupima kako bi se potvrdila verodostojnost predloženog algoritma. Modifikovani PSO algoritam dostiže solidne rezultate, ali je u poređenju sa GCSA i HGGA njegova konvergencija znatno sporija. Razlog zbog kojeg HGGA i GCSA prevladavaju u odnosu na ostale algoritme po pitanju konvergencije prvenstveno leži u genetskim operatorima i adaptivnim koeficijentima za ukrštanje i mutaciju. Na osnovu njih omogućeno je balansiranje lokalnih i globalnih kapaciteta algoritma, a samim tim ostvareni su i bolji rezultati u odnosu na druge metaheurističke algoritme.

6.1.2 Druga eksperimentalna studija

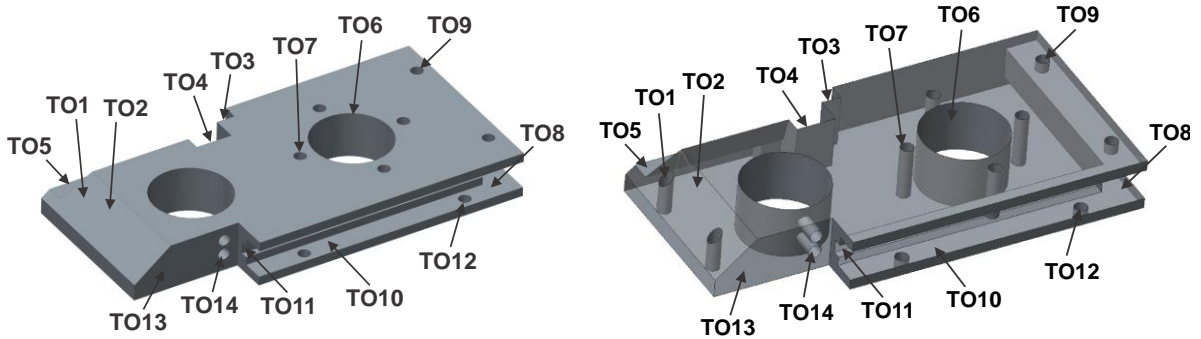
Drugi primer prizmatičnog dela za koji je izvršena OTP primenom predložene modifikacije savremenog algoritma CSA preuzet je iz [138]. Sa tehnološkog aspekta odnosi među zahvatima obrade kod ovog modela formiraju tvrda i meka ograničenja prethođenja. Za grafičko i numeričko predstavljanje ovakvih ograničenja upotrebljen je nepovezani graf prethođenja kao i matrice prethođenja. S obzirom na dva različita tipa prethođenja u matrici se koriste dve celobrojne vrednosti. Prizmatični deo kao u prethodnoj studiji poseduje 14 tipskih tehnoloških oblika, ali je za obradu ovog dela neophodno izvesti 14 zahvata obrade. Trodimenzionalni solid model dela sa konceptualnim i transparentnim efektom senčenja, nepovezani graf prethođenja i matrica prethođenja za tvrda i meka ograničenja grupno su prikazani na slici 6.8. Relevantne tehnološke informacije koje se tiču tipskih tehnoloških oblika, zahvata obrade i tehnoloških resursa date su u tabeli 6.13. Informacije o odnosima prethođenja između tipskih oblika, odnosno zahvata obrade date su u tabeli 6.14. Komponente troškova mašina i alata za drugi eksperimentalni primer predstavljene su u tabeli 6.15.

U ovom eksperimentu verifikacija metaheurističkih algoritama izvršena je razmatranjem dva tehnološka uslova optimizacije:

- 1) Optimizacija tehnoloških procesa kada su raspoloživi svi tehnološki resursi, odnosno sve varijante mašina, alata i SPA. Svi težinski koeficijenti u definisanoj

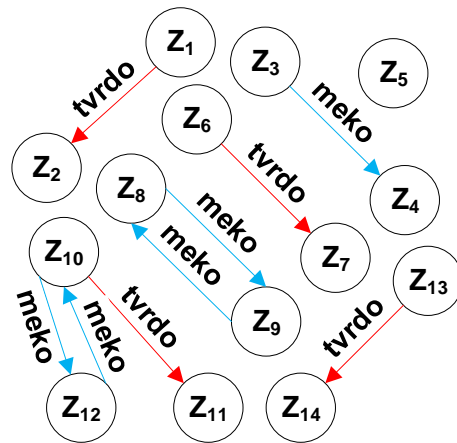
funkciji cilja u jednačini (18) sa troškovima TWMC iz jednačine (16) jednaki su 1.

- 2) Optimizacija tehnoloških procesa bez učešća troškova korišćenja alata i troškova promena alata. Težinski koeficijenti u istoj jednačini (18), w_2 i w_4 jednaki su 0 dok su w_1 , w_3 i w_5 jednaki 1.



(a)

(b)



(c)

	Z ₁	Z ₂	Z ₃	Z ₄	Z ₅	Z ₆	Z ₇	Z ₈	Z ₉	Z ₁₀	Z ₁₁	Z ₁₂	Z ₁₃	Z ₁₄
Z ₁	0	1	0	0	0	0	0	0	0	0	0	0	0	0
Z ₂	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Z ₃	0	0	0	2	0	0	0	0	0	0	0	0	0	0
Z ₄	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Z ₅	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Z ₆	0	0	0	0	0	0	1	0	0	0	0	0	0	0
Z ₇	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Z ₈	0	0	0	0	0	0	0	2	0	0	0	0	0	0
Z ₉	0	0	0	0	0	0	0	2	0	0	0	0	0	0
Z ₁₀	0	0	0	0	0	0	0	0	0	1	2	0	0	0
Z ₁₁	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Z ₁₂	0	0	0	0	0	0	0	0	2	0	0	0	0	0
Z ₁₃	0	0	0	0	0	0	0	0	0	0	0	0	0	1
Z ₁₄	0	0	0	0	0	0	0	0	0	0	0	0	0	0

(d)

Slika 6.8. Trodimenzionalni solid model u konceptualnom (a) i transparentnom (b) prikazu, graf (c) i matrica prethođenja (d) za drugi eksperiment

Tabela 6.13. Informacije o tipskim oblicima, zahvatima i tehnološkim resursima za drugi eksperiment

Tipski tehnološki oblici	Zahvati	Mašine{ }	Alati{ }	SPA{ }
TO1 – 2 otvora kao umnoženi TO	Bušenje (Z1)	M1,M2,M3	A1	+z,-z
TO2 – Zakošenje	Glodanje (Z2)	M2,M3	A8	-x,+y,-y,-z
TO3 – Žljeb	Glodanje (Z3)	M2,M3	A5,A6	+y
TO4 – Žljeb	Glodanje (Z4)	M2	A5,A6	+y
TO5 – Stepenik	Glodanje (Z5)	M2,M3	A5,A6	+y,-z
TO6 – 2 otvora kao umnoženi TO	Bušenje (Z6)	M1,M2,M3	A2	+z,-z
TO7 – 4 otvora kao umnoženi TO	Bušenje (Z7)	M1,M2,M3	A1	+z,-z
TO8 - Žljeb	Glodanje (Z8)	M2,M3	A5,A6	+x
TO9 – 2 otvora kao umnoženi TO	Bušenje (Z9)	M1,M2,M3	A1	-z
TO10 – Žljeb	Glodanje (Z10)	M2,M3	A5,A6	-y
TO11 – Žljeb	Glodanje (Z11)	M2,M3	A5,A7	-y
TO12 – 2 otvora kao umnoženi TO	Bušenje (Z12)	M1,M2,M3	A1	+z,-z
TO13 – Stepenik	Glodanje (Z13)	M2,M3	A5,A6	-x,-y
TO14 – 2 otvora kao umnoženi TO	Bušenje (Z14)	M1,M2,M3	A1	-y

Tabela 6.14. Ograničenja prethođenja među tipskim oblicima i zahvatima za drugi eksperiment

Tehnološka interakcija	Ograničenja prethođenja	Tip ograničenja	Tehnološka interakcija	Ograničenja prethođenja	Tip ograničenja
Interakcija alata i tipskog oblika	Z1 se mora realizovati pre Z2	Tvrd ograničenja	Interakcija tankozidnih struktura	Z9 se mora realizovati pre Z8	Meka ograničenja
Interakcija tipskog oblika i tehnološke baze	Z6 se mora realizovati pre Z7			Z12 se mora realizovati pre Z10	
	Z10 se mora realizovati pre Z11		Z8 se mora realizovati pre Z9		
	Z13 se mora realizovati pre Z14		Z10 se mora realizovati pre Z12		
			Interakcija jednostavnije obrade	Z13 se mora realizovati pre Z14	
				Z3 se mora realizovati pre Z4	

Tabela 6.15. Informacije o troškovima za drugi eksperiment

Redni broj	Naziv troškova	Novčana vrednost
Troškovi rada (korišćenja) mašina:		
M1	Stubna bušilica	10
M2	Vertikalna glodalica	35
M3	Troosna vertikalna glodalica	60
Troškovi rada (korišćenja) alata:		
A1	Burgija 1	3
A2	Burgija 2	3
A3	Razvrtač	8
A4	Nož za finu unutrašnju obradu	15
A5	Glodalo 1	10
A6	Glodalo 2	15
A7	Glodalo za žljebove	10
A8	Konturno glodalo	10
Troškovi promene:		
MCI	Mašina	300
TCI	Alata	15
SCI	Stezanja (smerova prilaza alata)	120

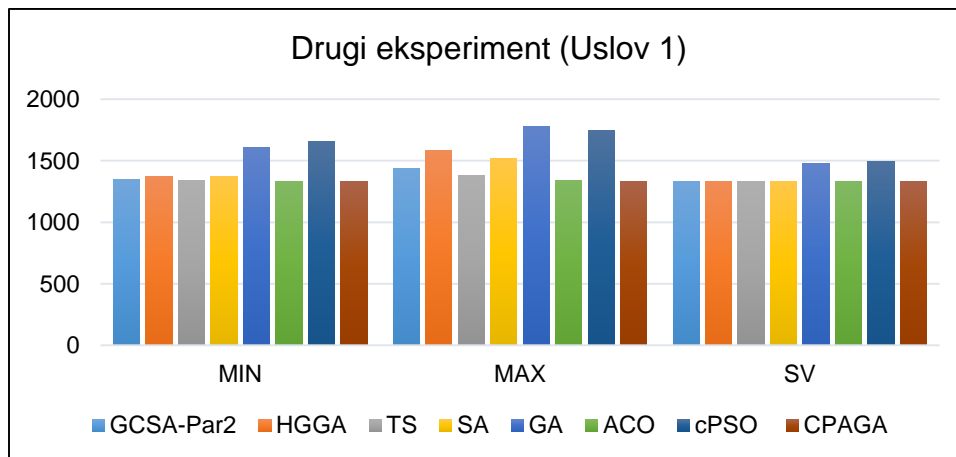
Za potrebe ovog eksperimenta testiran je algoritam GCSA-Par2 sa ulaznim parametrima dobijenim nakon manualnih testiranja. Postignuti su optimalni tehnološki procesi za drugi reprezentativni deo prizmatičnog oblika i dva tehnološka uslova optimizacije. Oni se mogu videti u tabeli 6.16. Poređenje ostvarenih rezultata izvršeno je u odnosu na nekoliko drugih rezultata iz literaturnih izvora: HGGA[12], TS, SA, GA i ACO [39], cPSO [21] i CPAGA [142]. Uporedna analiza troškova TWMC može se videti u tabeli 6.17. Na slikama 6.9 i 6.10 mogu se videti dijagrami koji odgovaraju uporednim rezultatima iz tabele 6.17.

Tabela 6.16. Optimalni tehnološki procesi za drugi reprezentativni deo i dva tehnološka uslova

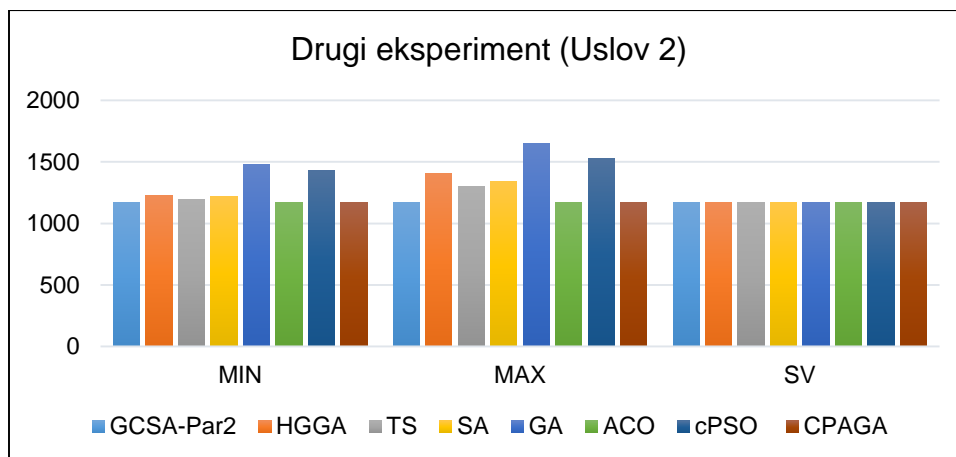
Uslov 1	Indeksi	1	2	3	4	5	6	7	8	9	10	11	12	13	14
	Zahvati	6	1	9	7	12	2	5	4	3	8	10	11	13	14
	Mašine	2	2	2	2	2	2	2	2	2	2	2	2	2	2
	Alati	2	1	1	1	1	8	5	5	5	5	5	5	5	1
	SPA	-z	-z	-z	-z	-z	+y	+y	+y	+y	+x	-y	-y	-y	-y
TMC=490, TMCC=0, NMC=0, TTC=98, TTCC=60, NTC=4, TSC=480, NSC=4, TWMC=1328															
Uslov 2	Indeksi	1	2	3	4	5	6	7	8	9	10	11	12	13	14
	Zahvati	8	10	13	14	11	5	3	4	6	1	7	9	12	2
	Mašine	2	2	2	2	2	2	2	2	2	2	2	2	2	2
	Alati	5	5	5	1	5	5	5	5	2	1	1	1	1	8
	SPA	+x	-y	-y	-y	-y	+y	+y	+y	-z	-z	-z	-z	-z	-z
TMC=490, TMCC=0, NMC=0, TSC=480, NSC=4, TWMC=1170															

Tabela 6.17. Uporedni rezultati optimizacije za drugi reprezentativni deo

Algoritam	Uslov 1			Uslov 2		
	SV	MAX	MIN	SV	MAX	MIN
GCSA-Par2	1344,5	1433	1328	1170	1170	1170
HGGA	1370	1583	1328	1224	1410	1170
TS	1342	1378	1328	1194	1298	1170
SA	1373,5	1518	1328	1217	1345	1170
GA	1611	1778	1478	1482	1650	1410
ACO	1329,5	1343	1328	1170	1170	1170
cPSO	1654	1748	1493	1428	1530	1270
CPAGA	1328	1328	1328	1170	1170	1170



Slika 6.9. Uporedni rezultati različitih algoritama za drugi reprezentativni deo i prvi tehnološki uslov



Slika 6.10. Uporedni rezultati različitih algoritama za drugi reprezentativni deo i drugi tehnološki uslov

U skladu sa ostvarenim troškovima obrade prizmatičnog dela u drugom eksperimentu, izvršena je uporedna analiza GCSA-Par2 sa algoritmima HGGA, PSO, GA, mPSO i CSA. Uporedni rezultati generisani u 20 ponavljanja dati su u tabelama 6.18 i

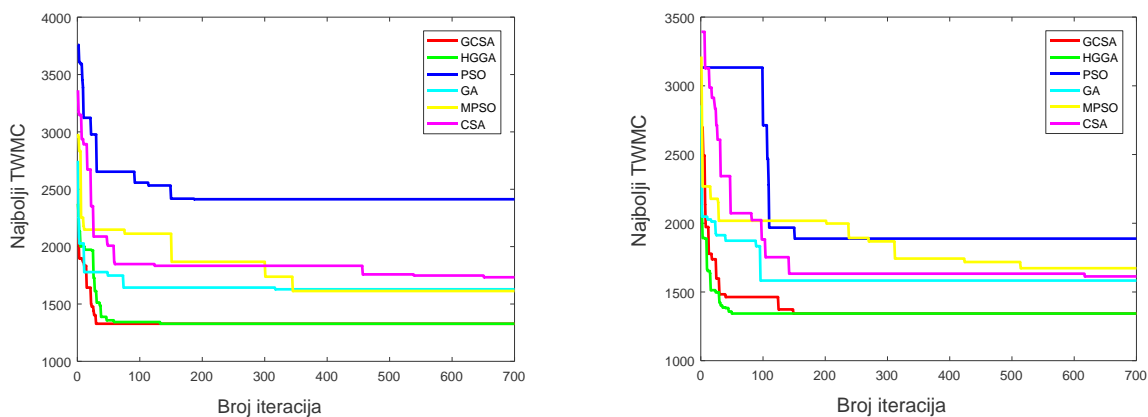
6.19 za prvi i drugi tehnološki uslov respektivno. Prikazane su najbolje vrednosti funkcija f_c i minimalne vrednosti troškova TWMC za 20 izvršavanja pri čemu su izražene standardna devijacija rezultata i srednja vrednost troškova TWMC. Na slici 6.11 dat je uporedni prikaz krivih konvergencije za drugi reprezentativni deo i dva tehnološka uslova optimizacije. Kao i kod prethodne studije, dve krive konvergencije prikazane su za oba tehnološka uslova.

Tabela 6.18. Izlazni rezultati drugog eksperimenta za prvi tehnološki uslov u 20 ponavljanja

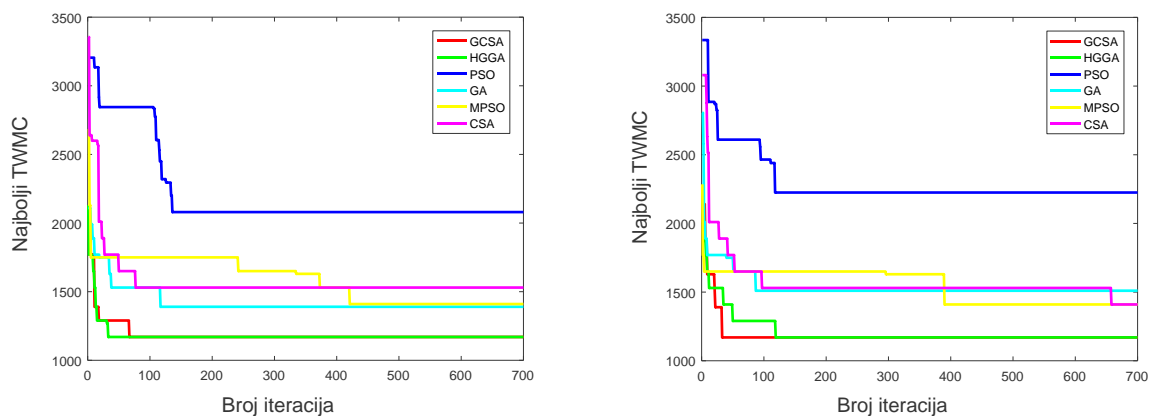
Br.	Maksimizacija funkcije cilja f_c i minimizacija troškova TWMC											
	GCSA-Par2		HGGA		MPSO		PSO		GA		CSA	
	f_c	TWMC	f_c	TWMC	f_c	TWMC	f_c	TWMC	f_c	TWMC	f_c	TWMC
1.	0,00074	1343	0,00075	1328	0,0006	1653	0,00063	1583	0,00061	1628	0,00059	1688
2.	0,00074	1343	0,00069	1448	0,00058	1733	0,0006	1673	0,00061	1643	0,00058	1733
3.	0,00074	1343	0,00068	1463	0,00062	1613	0,00063	1598	0,00058	1718	0,00057	1743
4.	0,00074	1343	0,00058	1718	0,00058	1733	0,00059	1688	0,00061	1643	0,00061	1648
5.	0,00074	1343	0,00068	1463	0,00057	1743	0,00067	1493	0,00058	1733	0,00057	1748
6.	0,00074	1343	0,00075	1328	0,00058	1718	0,00063	1598	0,00066	1523	0,00057	1763
7.	0,00075	1328	0,00075	1328	0,00058	1738	0,00045	2233	0,00058	1713	0,00058	1718
8.	0,00074	1343	0,0007	1433	0,00058	1718	0,00065	1528	0,00057	1763	0,00058	1718
9.	0,00074	1343	0,00074	1343	0,00063	1588	0,00062	1613	0,00063	1583	0,00055	1833
10.	0,00075	1328	0,00074	1358	0,00061	1643	0,00062	1603	0,00059	1698	0,00055	1803
11.	0,00074	1343	0,00074	1358	0,00058	1718	0,00063	1598	0,00059	1703	0,00063	1598
12.	0,00074	1358	0,00075	1328	0,00057	1763	0,0006	1688	0,00058	1723	0,00062	1613
13.	0,00074	1343	0,00075	1328	0,00057	1763	0,00062	1613	0,00058	1733	0,00066	1513
14.	0,00075	1328	0,00076	1313	0,00061	1633	0,00061	1638	0,00062	1613	0,00061	1628
15.	0,00074	1343	0,00075	1328	0,00058	1718	0,00059	1703	0,0006	1658	0,00062	1613
16.	0,00069	1448	0,00075	1328	0,00059	1703	0,00062	1623	0,00062	1613	0,00059	1698
17.	0,00075	1328	0,00068	1478	0,00058	1718	0,00062	1618	0,00058	1718	0,00058	1718
18.	0,00075	1328	0,00074	1343	0,00058	1733	0,00061	1633	0,00061	1628	0,00061	1643
19.	0,00074	1343	0,00074	1343	0,00059	1698	0,00042	2383	0,00058	1733	0,00061	1628
20.	0,00074	1343	0,00067	1493	0,00057	1743	0,00063	1598	0,00059	1688	0,00057	1748
Statistički parametri (Standardna devijacija rezultata i srednja vrednost troškova)												
	σ	SV	σ	SV	σ	SV	σ	SV	σ	SV	σ	SV
	24,77	1345,25	95,71	1392,5	48,89	1703,5	214,57	1684,25	60,28	1672,75	76,05	1689,75

Tabela 6.19. Izlazni rezultati drugog eksperimenta za drugi tehnološki uslov u 20 ponavljanja

Br.	Maksimizacija funkcije cilja fc i minimizacija troškova TWMC															
	GCSA-Par2		HGGA		MPSO		PSO		GA		CSA					
	fc	TWMC	fc	TWMC	fc	TWMC	fc	TWMC	fc	TWMC	fc	TWMC				
1.	0,00085	1170	0,00085	1170	0,00066	1510	0,00071	1410	0,00065	1530	0,00061	1630				
2.	0,00078	1290	0,00078	1290	0,00071	1410	0,00071	1410	0,00065	1530	0,00085	1170				
3.	0,00085	1170	0,00078	1290	0,00065	1530	0,00071	1410	0,00071	1410	0,00066	1510				
4.	0,00085	1170	0,00078	1290	0,00065	1530	0,00071	1410	0,00066	1510	0,00066	1510				
5.	0,00085	1170	0,00071	1410	0,00061	1630	0,00071	1410	0,00066	1510	0,00071	1410				
6.	0,00085	1170	0,00078	1290	0,00065	1530	0,00071	1410	0,00072	1390	0,00061	1630				
7.	0,00085	1170	0,00085	1170	0,00065	1530	0,00071	1410	0,00071	1410	0,00065	1530				
8.	0,00085	1170	0,00078	1290	0,00065	1530	0,00071	1410	0,00071	1410	0,00061	1630				
9.	0,00085	1170	0,00085	1170	0,00065	1530	0,00071	1410	0,00071	1410	0,00072	1390				
10.	0,00085	1170	0,00085	1170	0,00065	1530	0,00048	2080	0,00065	1530	0,00061	1630				
11.	0,00085	1170	0,00085	1170	0,00066	1510	0,00066	1510	0,00071	1410	0,00066	1510				
12.	0,00085	1170	0,00078	1290	0,00071	1410	0,00072	1390	0,00065	1530	0,00078	1290				
13.	0,00085	1170	0,00085	1170	0,00065	1530	0,00048	2080	0,00071	1410	0,00061	1630				
14.	0,00085	1170	0,00085	1170	0,00071	1410	0,00071	1410	0,00065	1530	0,00071	1410				
15.	0,00085	1170	0,00085	1170	0,00071	1410	0,00071	1410	0,00071	1410	0,00066	1510				
16.	0,00085	1170	0,00078	1290	0,00061	1630	0,00078	1290	0,00071	1410	0,00066	1510				
17.	0,00085	1170	0,00071	1410	0,00066	1510	0,00071	1410	0,00072	1390	0,00071	1410				
18.	0,00085	1170	0,00085	1170	0,00065	1530	0,00071	1410	0,00071	1410	0,00071	1410				
19.	0,00085	1170	0,00085	1170	0,00066	1510	0,00071	1410	0,00072	1390	0,00065	1530				
20.	0,00085	1170	0,00085	1170	0,00071	1410	0,00071	1410	0,00071	1410	0,00071	1410				
Statistički parametri (Standardna devijacija rezultata i srednja vrednost troškova)																
	σ	SV	σ	SV	σ	SV	σ	SV	σ	SV	σ	SV				
	26,15	1176	80,27	1236	64,06	1506	204,71	1475	57,37	1447	119,71	1483				



(a) Prvi tehnološki uslov



(b) Drugi tehnološki uslov

Slika 6.11. Krive konvergencije za drugi eksperiment

6.1.2.1 Diskusija rezultata druge eksperimentalne studije

Modifikovani CSA algoritam predložen u ovoj studiji pokazao je veoma dobre performanse pri rešavanju problema optimizacije tehnološkog procesa obrade prizmatičnog dela sa tvrdim i mekim ograničenjima prethođenja. Prema informacijama iz tabele 6.18 i 20 egzekucija algoritma, GCSA je optimalnu vrednost troškova TWMC od 1328 novčanih jedinica pronašao ukupno pet puta imajući u vidu prvi tehnološki uslov. Međutim, posmatranjem uporednih rezultata iz literaturnih izvora koji su dati u tabeli 6.17, tradicionalni pristupi iz literature, kao što su ACO i TS pokazuju neznatno bolju srednju vrednost rezultata troškova TWMC.

Kod drugog tehnološkog uslova, rezultati iz tabele 6.19 potvrđuju pronalaženje optimalnih troškova TWMC iznose 1170 novčanih jedinica pri čemu je njihova učestalost procentna pojavivši se 20 puta u 20 ponavljanja GCSA algoritma. Ovaj učinak se pokazao odličnim imajući u vidu da jedino ACO i CPAGA algoritmi (tabela 6.17) pokazuju identične rezultate.

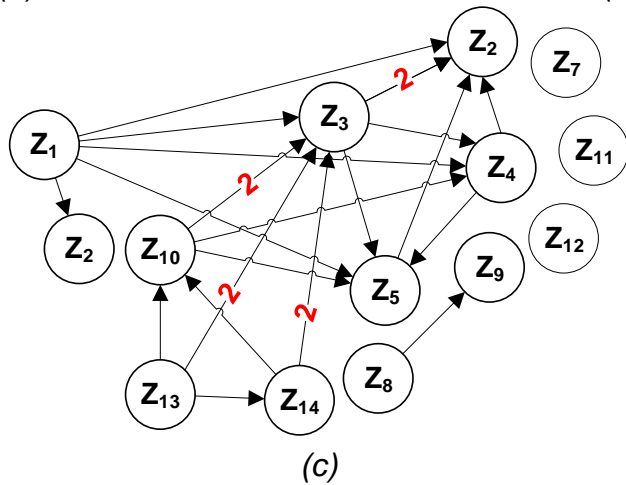
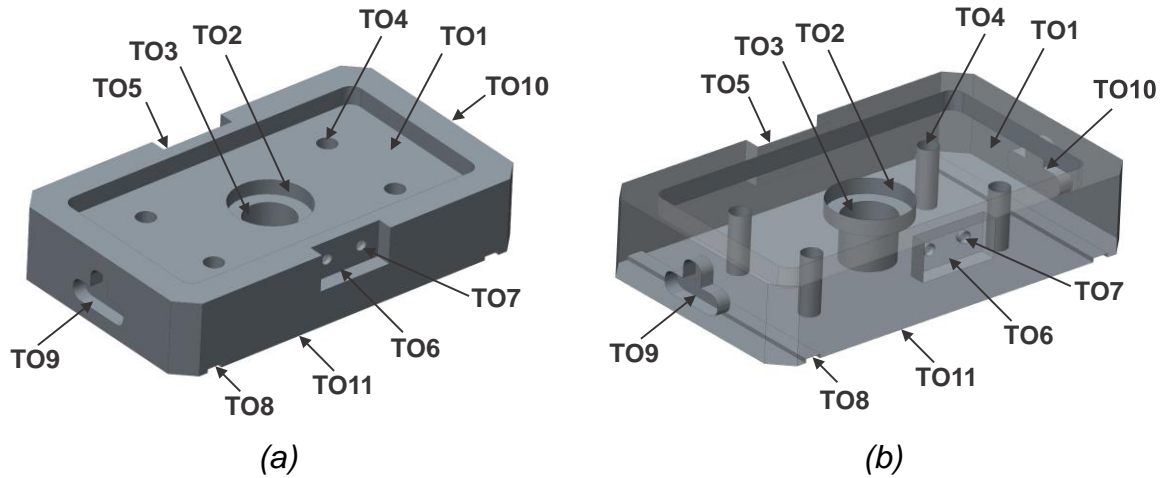
Brzu konvergenciju možemo pripisati modifikovanom algoritmu GCSA koji je značajno bolji u odnosu na tradicionalne algoritme PSO, GA i CSA, kao i modifikovani PSO. Kad je u pitanju prvi tehnološki uslov (slika 6.11), razlika u konvergenciji i kvalitetu rešenja HGGA i GCSA relativno malo se razlikuju što potvrđuje i srednja vrednost dobijenih rezultata u 20 ponavljanja. Detaljni rezultati dobijeni na izlazu prikazani su u tabeli 6.18. Analogno prethodnom slučaju, brza konvergencija uveliko zavisi od genetskih operatora i adaptivnih koeficijenata za ukrštanje i mutaciju k_c i k_m (odjeljak 5.1.2.4).

Prema slici 6.11 možemo primetiti da se za drugi tehnološki uslov javlja sličan scenario u kojem su se u pogledu konvergencije ka optimumu najpodobnijim pokazali GCSA i HGGA. Međutim, najbolju konzistentnost rezultata za 20 ponavljanja ostvario je predloženi GCSA koji se pokazao nadmoćnijim u poređenju sa ostalim algoritmima u uporednoj studiji. Objedinjeni izlazni rezultati drugog eksperimenta za drugi tehnološki uslov mogu se videti u tabeli 6.19.

6.1.3 Treća eksperimentalna studija

Treća eksperimentalna studija razmatra model prizmatičnog kućišta koji je preuzet iz [45]. Trodimenzionalni solid model sa konceptualnim i transparentnim efektom prikazan je na slici 6.12a i 6.12b. Dvodimenzionalni radionički crtež ovog prizmatičnog dela prikazan je u [19]. Nakon dve studije u kojima su predstavljeni odnosi prethođenja sa klasičnim, kao i sa tvrdim i mekim ograničenjima, u okviru ovog eksperimenta naglašen je model dela kod kog određeni odnosi među zahvatima zahtevaju obradu u istoj podoperaciji, odnosno stezanju dela. Stoga, formiran je nepovezani graf prethođenja na kojem su numeričkim vrednostima označeni zahtevi za istim stezanjem (broj 2). Prema tome, matrica prethođenja je formulisana na sličan način kao u prethodne dve studije, ali sa dve numeričke vrednosti, 1 i 2. Nepovezani graf prethođenja i matrica prethođenja za treći reprezentativni deo prizmatičnog oblika prikazani su na slici 6.12c i 6.12d. Relevantne tehnološke informacije koje se tiču tipskih tehnoloških oblika, zahvata obrade i tehnoloških resursa za treći deo date su u tabeli 6.20. Informacije o odnosima prethođenja između zahvata obrade date su u tabeli 6.21. U ovoj studiji prioriteta među zahvatima određeni su na bazi dimenzionih, geometrijskih, tehnoloških i ekonomskih prethođenja koji su detaljnije definisani u poglavlju 2.2. Komponente troškova mašina i alata za treći deo date su u tabeli 6.22. Optimizacija tehnološkog procesa obrade posmatranog dela vrši se isključivo za jedan tehnološki uslov optimizacije kada su na raspolaganju svi tehnološki resursi. Sve navedene informacije preuzete su iz izvora [45].

Optimalni tehnološki procesi za treći reprezentativni deo dati su u tabeli 6.23. Prilikom verifikacije GCSA na ovom primeru isti ulazni parametri generisani u okviru „Plana B“ faktornog eksperimenta usvojeni su kao kod prethodnih studija. U tabeli 6.24 predstavljeni su uporedni rezultati za prizmatično kućište koji uključuju rezultate ostvarene sledećim algoritmima: HGASA, PSO, GA, HGGA, mPSO i CSA. Dijagram prikaza uporednih rezultata nalazi se na slici 6.13.



	Z ₁	Z ₂	Z ₃	Z ₄	Z ₅	Z ₆	Z ₇	Z ₈	Z ₉	Z ₁₀	Z ₁₁	Z ₁₂	Z ₁₃	Z ₁₄
Z ₁	0	1	1	1	1	1	0	0	0	0	0	0	0	0
Z ₂	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Z ₃	0	2	0	1	1	0	0	0	0	0	0	0	0	0
Z ₄	0	1	0	0	0	0	0	0	0	0	0	0	0	0
Z ₅	0	1	0	1	0	0	0	0	0	0	0	0	0	0
Z ₆	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Z ₇	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Z ₈	0	0	0	0	0	0	0	0	1	0	0	0	0	0
Z ₉	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Z ₁₀	0	0	2	1	1	0	0	0	0	0	0	0	0	0
Z ₁₁	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Z ₁₂	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Z ₁₃	0	0	2	0	0	0	0	0	0	1	0	0	0	1
Z ₁₄	0	0	2	0	0	0	0	0	0	1	0	0	0	0

Slika 6.12. Trodimenzionalni solid model u konceptualnom (a) i transparentnom (b) režimu, nepovezani graf prethođenja (c) i matrica prethođenja (d) za treći eksperiment

Tabela 6.20. Informacije o tipskim oblicima, zahvatima, resursima i ograničenjima prethođenja za treći eksperiment

Tipski tehnološki oblici	Zahvati	Mašine{ }	Alati{ }	SPA{ }
TO1 – Džep	Glodanje (Z1)	M2,M3	A5,A6,A7	-z
TO2 – Upust	Bušenje (Z2)	M1,M2,M3	A2,A3,A4	-z
TO3 – Otvor	Bušenje (Z3)	M1,M2,M3	A2,A3,A4	+z,-z
	Proširivanje (Z4)	M3,M4	A8	+z,-z
	Razvrtanje (Z5)	M1,M2,M3	A9	+z,-z
TO4 – 4 otvora	Bušenje (Z6)	M1,M2,M3	A2	+z,-z
TO5 – Žljeb	Glodanje (Z7)	M2,M3	A5,A6	-y,-z
TO6 - Žljeb	Glodanje (Z8)	M2,M3	A5,A6	+y,-z
TO7 – 2 upusta	Bušenje (Z9)	M1,M2,M3	A1	+y
TO8 – 2 žljeba	Glodanje (Z10)	M2,M3	A6,A7,A10	+y,-y,+z
TO9 - Džep	Glodanje (Z11)	M2,M3	A5,A6,A7	-x
TO10 - Džep	Glodanje (Z12)	M2,M3	A5,A6,A7	+x
TO11 – Ravna površina	Grubo glodanje (Z13)	M2,M3	A5,A6,A7	+x,-x,+y,
	Fino glodanje (Z14)	M2,M3	A5,A6,A7	-y,+z

Tabela 6.21. Odnosi prethođenja među zahvatima obrade za treći eksperiment

Zahvat	Vrsta prethođenja			
	Dimenziono	Geometrijsko	Tehnološko	Ekonomsko
Z1				
Z2	Z1	Z3, Z4	Z3, Z4	OP1
Z3		Z13,Z14,Z10		Z1/Z13, Z14
Z4		Z13,Z14,Z10	Z3	Z1/Z13, Z14
Z5		Z13,Z14,Z10	Z3, Z4	Z1/Z13, Z14
Z6				Z1/Z13, Z14
Z7				
Z8				
Z9	Z8		Z8	
Z10	Z13,Z14		Z13,Z14	
Z11				
Z12				
Z13				
Z14			Z13	

Tabela 6.22. Informacija o troškovima za treći prizmatični deo

Redni broj	Naziv troškova	Novčana vrednost
Troškovi rada (korišćenja) mašina:		
M1	Stubna bušilica	10
M2	Troosna vertikalna glodalica	40
M3	CNC troosna vertikalna glodalica	100
M4	Bušilica-glodalica	60
Troškovi rada (korišćenja) alata:		
A1	Burgija 1	7
A2	Burgija 2	5
A3	Burgija 3	3
A4	Burgija 4	8

Tabela 6.22. Informacija o troškovima za treći prizmatični deo (nastavak)

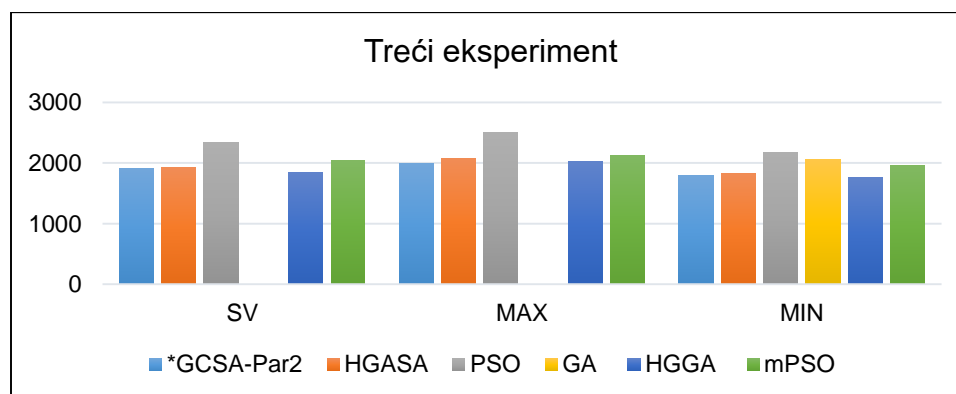
Redni broj	Naziv troškova	Novčana vrednost
Troškovi rada (korišćenja) alata:		
A5	Glodalo 1	7
A6	Glodalo 2	10
A7	Glodalo 3	15
A8	Proširivač	20
A9	Razvrtač	30
A10	Glodalo za žljebove	15
Troškovi promene:		
MCI	Mašina	160
TCI	Alata	20
SCI	Stezanja (SPA)	120

Tabela 6.23. Optimalni tehnološki proces za treći eksperiment

Indeksi	1	2	3	4	5	6	7	8	9	10	11	12	13	14
Zahvati	11	1	8	7	12	13	14	10	6	3	9	4	5	2
Mašine	2	2	2	2	2	2	2	2	2	2	2	3	3	3
Alati	5	5	5	5	5	5	5	6	2	2	1	8	9	3
PPA	-X	-Z	-Z	-Z	+X	+Z	+Z	+Z	+Z	+Z	+y	-Z	-Z	-Z
TMC=740, TMCC=160, NMC=1, TTC=129, TTCC=120, NTC=6, TSC=600, NSC=5, TWMC=1749														

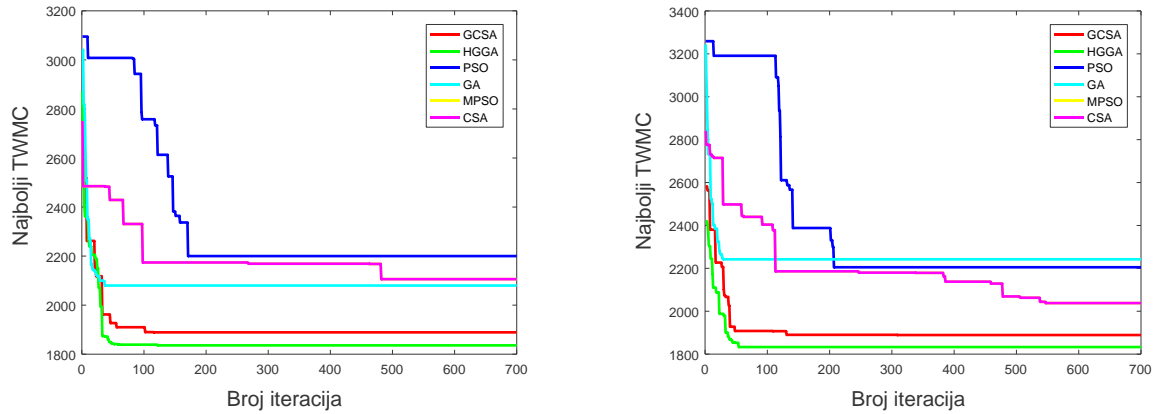
Tabela 6.24. Uporedni rezultati GCSA-Par2 sa drugim algoritmima za treći eksperiment

Algoritam	Uslov 1		
	SV	MAX	MIN
*GCSA-Par2	1908,3	1992	1792
HGASA	1927,1	2079	1823
PSO	2342,3	2497	2180
GA	/	/	2057
HGGA	1841	2020	1755
mPSO	2045	2120	1960
CSA	2257,85	2560	2100



Slika 6.13. Uporedni rezultati ostvareni različitim metaheurističkim algoritmima za treći eksperiment

Kao i u prethodne dve studije, izvršena je uporedna analiza koja uključuje rezultate GCSA-Par2, zatim HGGA, GA, PSO, mPSO i CSA. Krive konvergencije navedenih metaheuristika uporedno su prikazane na slici 6.14. Radi se o dva primera različityh krivyh za isti uslov optimizacije. Tabela 6.25 prikazuje rezultate funkcije cilja f_c i troškova TWMC sa statističkim parametrima za treći eksperiment.



Slika 6.14. Krive konvergencije za treći eksperiment

Tabela 6.25. Izlazni rezultati trećeg eksperimenta dobijeni u 20 ponavljanja

Br.	Maksimizacija funkcije cilja fc i minimizacija troškova TWMC																								
	GCSA-Par2		HGGA		MPSO		PSO		GA		CSA														
	fc	TWMC	fc	TWMC	fc	TWMC	fc	TWMC	fc	TWMC	fc	TWMC													
1.	0,0005	1992	0,00053	1870	0,00048	2079	0,00041	2460	0,00051	1955	0,00044	2260													
2.	0,00056	1792	0,00054	1869	0,00049	2058	0,00045	2220	0,0005	1982	0,00045	2223													
3.	0,00544	1839	0,00055	1833	0,0005	1994	0,00043	2334	0,00051	1948	0,00039	2540													
4.	0,0005	1992	0,00057	1755	0,00049	2035	0,00042	2397	0,00048	2088	0,00043	2343													
5.	0,0005	1992	0,00056	1778	0,00048	2079	0,00043	2337	0,00043	2324	0,00047	2126													
6.	0,00053	1894	0,00054	1835	0,0005	2000	0,00044	2297	0,00041	2415	0,00044	2277													
7.	0,00053	1891	0,00053	1870	0,00048	2071	0,0004	2480	0,00049	2052	0,00043	2321													
8.	0,00053	1889	0,00056	1772	0,00049	2022	0,00043	2337	0,00051	1972	0,00046	2160													
9.	0,00053	1889	0,00057	1758	0,00049	2050	0,0004	2473	0,00047	2126	0,00047	2137													
10.	0,00053	1891	0,00056	1785	0,00048	2077	0,00043	2340	0,00044	2263	0,00044	2249													
11.	0,00053	1889	0,00054	1869	0,00049	2044	0,00042	2390	0,00048	2102	0,00046	2189													
12.	0,00053	1889	0,00053	1870	0,00049	2037	0,00042	2393	0,00046	2161	0,00046	2181													
13.	0,00053	1889	0,00057	1764	0,00049	2035	0,00044	2262	0,00045	2203	0,00045	2203													
14.	0,0005	1992	0,00051	1965	0,00051	1960	0,00042	2380	0,00051	1949	0,00046	2195													
15.	0,00053	1891	0,00056	1791	0,00049	2043	0,00044	2290	0,00046	2166	0,00045	2223													
16.	0,00053	1892	0,00054	1869	0,00049	2052	0,00045	2217	0,00044	2254	0,00044	2263													
17.	0,00053	1891	0,0005	2020	0,00048	2095	0,00045	2217	0,00047	2134	0,00039	2560													
18.	0,00053	1891	0,00056	1778	0,00051	1979	0,0004	2497	0,00045	2235	0,00046	2181													
19.	0,00053	1891	0,00053	1898	0,00047	2120	0,00046	2180	0,00044	2264	0,00041	2426													
20.	0,0005	1990	0,00053	1870	0,00048	2070	0,00043	2345	0,00048	2076	0,00048	2100													
Statistički parametri (Standardna devijacija rezultata i srednja vrednost troškova)																									
	σ	SV	σ	SV	σ	SV	σ	SV	σ	SV	σ	SV	σ	SV	σ	SV	σ	SV	σ	SV	σ	SV	σ	SV	
	53,46	1908,3	68,93	1840,95	38,74	2045	91,42	2342,3	130,99	2133,45	123,15	2257,85													

6.1.3.1 Diskusija rezultata treće eksperimentalne studije

U trećoj eksperimentalnoj studiji izvršena je optimizacija tehnološkog procesa obrade kućišta prizmatičnog oblika. Modifikovani algoritam GCSA sa ulaznim parametrima dobijenim na osnovu „Plana B“ faktornog eksperimenta pokazao je veoma dobre performanse u poređenju sa ostalim algoritmima. Analiza je izvršena za algoritme čiji su postupci pretraživanja ponovljeni u 20 izvršavanja algoritma.

Prema uporednim rezultatima iz tabele 6.24 može se videti da minimalna vrednost troškova TWMC od 1792 novčane jedinice koju je ostvario GCSA predstavlja mnogo bolju vrednost u poređenju sa tradicionalnim GA i PSO kao i u odnosu na hibridne pristupe HGASA i mPSO. Pored GCSA, veoma dobar rezultat u pogledu minimalnih troškova postigao je hibridni HGGA koji je postigao najbolju vrednost od 1755 novčanih jedinica u celoj studiji. U pogledu na najlošiji rezultat kada su u pitanju troškovi TWMC, GCSA je pokazao najbolje performanse ostvarivši najbolje maksimalne troškove od 1992 novčane jedinice. Na kraju, najbolju srednju vrednost rezultata troškova ostvario je hibridni HGGA koja iznosi 1841 novčanu jedinicu, dok je predloženi GCSA pokazao nešto slabiji učinak od 1908,3. Pored toga, jasno se može zaključiti da GCSA nadmašuje performanse tradicionalnih i nekih savremenih metaheurističkih rešenja. Pogled u objedinjene rezultate optimizacije u tabeli 6.25 potvrđuje konzistentnost rezultata koje su ostvarili GCSA i HGGA pristupi. Najbolje standardno odstupanje rezultata pokazuje modifikovani PSO algoritam, ali sa znatno lošijim učinkom po pitanju srednjih troškova TWMC u odnosu na GCSA i HGGA.

Uporedne krive konvergencije rezultata sa slike 6.14 jasno ukazuju na brzu konvergenciju GCSA kao i HGGA koji su u pogledu brzine ispred ostalih algoritama iz ove studije. Može se zapaziti i poboljšanje konvergentnosti koje GCSA ostvaruje u odnosu na tradicionalne algoritme PSO, GA a naročito algoritam CSA.

6.2 Podešavanje parametara i analiza rezultata četvrte eksperimentalne studije

U ovom eksperimentu izvršena je optimizacija tehnološkog procesa obrade četvrtog prizmatičnog dela. S obzirom na znatno veće dimenzije ovog problema koje se ogledaju prvenstveno u broju zahvata i varijantnih tehnoloških resursa, izvršena je optimizacija ulaznih parametara pomoću faktornog plana eksperimenta. Formiran je algoritam GCSA-Par3, odnosno delimični ortogonalni plan sa 32 eksperimenta koji je identičan planu za algoritam GCSA-Par1 iz prvog eksperimenta. Ulazni parametri GCSA sa nivoima njihove varijacije su sledeći: maksimalan broj iteracija $MaxIt$ [1000, 2000], veličina jata $FlockSize$ [80; 100], verovatnoća svesnosti vrane AP [0,2; 0,8], dužina leta fl [0,5; 2,5], veličina turnira $TourSize$ [3; 5] i adaptivni koeficijenti za ukrštanje i mutaciju, k_{c1} [0,6; 0,9], k_{c2} [0,2; 0,6], k_{m1} [0,3; 0,7] i k_{m2} [0,1; 0,3]. Nakon 20 izvršavanja algoritma, statističke vrednosti u vidu srednjeg, minimalnog i maksimalnog rezultata TWPC zabeleženi su na izlazu. Delimični ortogonalni plan eksperimenta za razmatrani algoritam GCSA-Par3 i izlaznim statističkim rezultatima prikazana je u tabeli 6.26.

Tabela 6.26. Delimični ortogonalni plan eksperimenta 2^5 sa izlaznim statističkim rezultatima

br. eksp.	Ulazni parametri modifikovanog algoritma vrane									Izlazni parametri		
	MaxIt	FlockSize	AP	fl	TourSize	kc ₁	kc ₂	km ₁	km ₂	SV	MIN	MAX
1.	2000	100	0,8	0,5	5	0,6	0,6	0,7	0,1	4331,7	4104	4659
2.	2000	100	0,2	0,5	3	0,6	0,6	0,3	0,1	5425,7	4929	5919
3.	1000	100	0,2	0,5	5	0,9	0,6	0,3	0,1	4915,45	4480	5399
4.	1000	80	0,8	0,5	5	0,9	0,2	0,7	0,1	4433,75	4135	4721
5.	2000	80	0,8	0,5	3	0,6	0,2	0,7	0,1	4376,95	4166	4650
6.	2000	100	0,8	2,5	3	0,6	0,6	0,7	0,3	5107,25	4767	5305
7.	2000	80	0,8	2,5	3	0,9	0,6	0,3	0,1	4366,15	4136	4679
8.	1000	100	0,2	2,5	5	0,6	0,2	0,7	0,1	6533,55	6147	7040
9.	2000	100	0,2	2,5	5	0,6	0,6	0,3	0,3	6166,1	5512	6557
10.	2000	80	0,2	2,5	5	0,9	0,6	0,7	0,1	6364,8	5709	6888
11.	1000	100	0,8	0,5	5	0,6	0,2	0,3	0,3	4424,8	4141	4761
12.	1000	80	0,2	2,5	5	0,9	0,2	0,3	0,3	6394,65	6024	6801
13.	1000	80	0,2	2,5	3	0,6	0,6	0,7	0,1	6879,15	6443	7275
14.	2000	100	0,2	0,5	5	0,9	0,2	0,7	0,3	5730,1	5513	5916
15.	1000	80	0,2	0,5	3	0,9	0,2	0,3	0,1	5765,65	5156	6253
16.	1000	80	0,8	2,5	5	0,6	0,6	0,3	0,1	4528,65	4226	5041
17.	2000	100	0,8	2,5	5	0,9	0,2	0,3	0,1	4473,3	4099	4899
18.	1000	100	0,2	0,5	3	0,6	0,2	0,7	0,3	6357,05	6046	6670
19.	1000	80	0,8	2,5	3	0,9	0,2	0,7	0,3	5291,85	5043	5505
20.	2000	80	0,2	2,5	3	0,6	0,2	0,3	0,3	6335,2	5988	6629
21.	2000	80	0,8	2,5	5	0,6	0,2	0,7	0,3	4271,8	4136	4581
22.	2000	100	0,2	2,5	3	0,9	0,2	0,7	0,1	6587,85	6216	6865
23.	2000	80	0,2	0,5	5	0,6	0,2	0,3	0,1	4948,75	4612	5240
24.	2000	80	0,2	0,5	3	0,9	0,6	0,7	0,3	6252,75	5736	6613
25.	1000	80	0,8	0,5	3	0,6	0,6	0,3	0,3	4391,75	4136	4721
26.	1000	100	0,2	2,5	3	0,9	0,6	0,3	0,3	6470,3	6077	6907
27.	2000	80	0,8	0,5	5	0,9	0,6	0,3	0,3	4381	4136	4598
28.	1000	100	0,8	2,5	3	0,6	0,2	0,3	0,1	4492,4	4189	4764
29.	1000	100	0,8	0,5	3	0,9	0,6	0,7	0,1	4475,6	4237	4911
30.	1000	80	0,2	0,5	5	0,6	0,6	0,7	0,3	5859,25	5551	6154
31.	1000	100	0,8	2,5	5	0,9	0,6	0,7	0,3	4369,8	4136	4598
32.	2000	100	0,8	0,5	3	0,9	0,2	0,3	0,3	4300	4098	4569

Analogno optimizaciji parametara koja je izvršena za GCSA algoritam u prvom eksperimentu, izvršena je optimizacija odziva pri čemu su za plan eksperimenta iz tabele 6.26 određene donja, ciljna i gornja vrednost izlaznih parametara (tabela 6.27). Optimalne vrednosti ulaznih parametara GCSA za četvrti reprezentativni deo date su u tabeli 6.28.

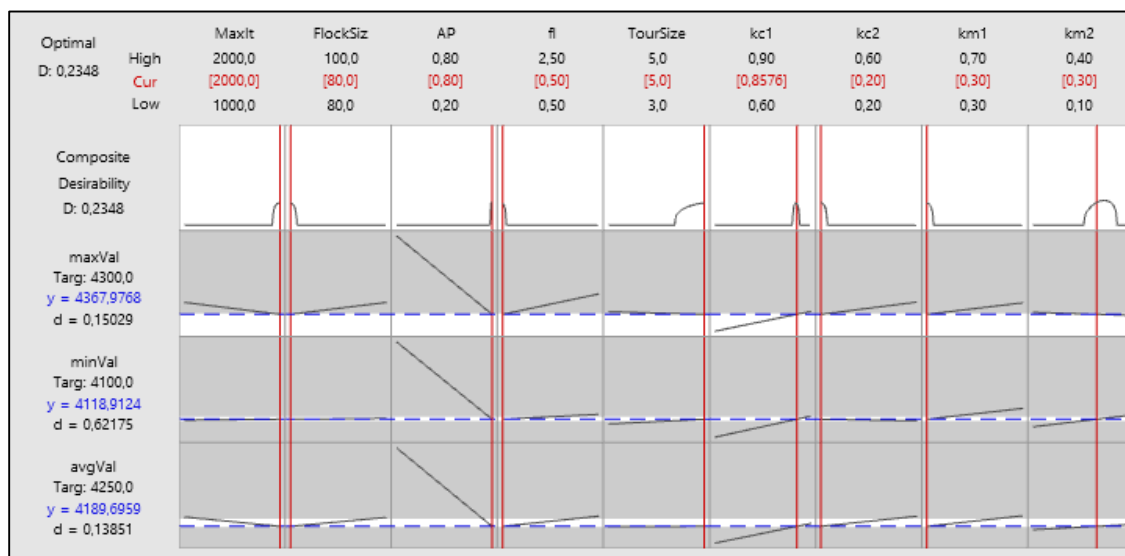
Uočljivo je da indeks kompozitne poželjnosti iznosi znatno manje nego u optimizaciji parametara u prvom eksperimentu. Razlog leži u činjenici da su postavljeni relativno strogi limiti za izlazne parametre troškova TWMC pri čemu dolazi do odstupanja ovih odzivnih veličina. Kao što se može na optimizacionom grafiku sa slike 6.15 primetiti, sve tri izlazne veličine za optimalne ulazne parametre su izvan postavljenih limita što je rezultiralo maloj vrednosti indeksa kompozitne poželjnosti koji iznosi 0,2348. Bez obzira na stroge limite, izlazne vrednosti troškova TWMC smatraju se odgovarajućim imajući u vidu rezultate koji su postignuti pomoću optimalnih parametara iz tabele 6.28. Analiza i diskusija rezultata četvrtog eksperimenta dati su u narednom odeljku.

Tabela 6.27. Parametri odziva sa usvojenim limitima i ciljnim vrednostima za četvrti eksperiment

Odziv	Donji limit	Ciljna vrednost	Gornji limit	Težinski koeficijent	Koeficijent važnosti
Maksimalni TWPC	4000	4300	4380	1	1
Minimalni TWPC	4098	4100	4150	1	1
Srednji TWPC	4180	4250	4300	1	1

Tabela 6.28. Optimalni ulazni parametri za „GCSA-Par3“

MaxIt	FlockSize	AP	fl	TourSize	kc ₁	kc ₂	km ₁	km ₂	SV TWMC	MIN TWMC	MAX TWMC	Indeks kompozitne poželjnosti
2000	80	0,8	0,5	5	0,8576	0,2	0,3	0,3	4189,70	4118,91	4367,98	0,234790



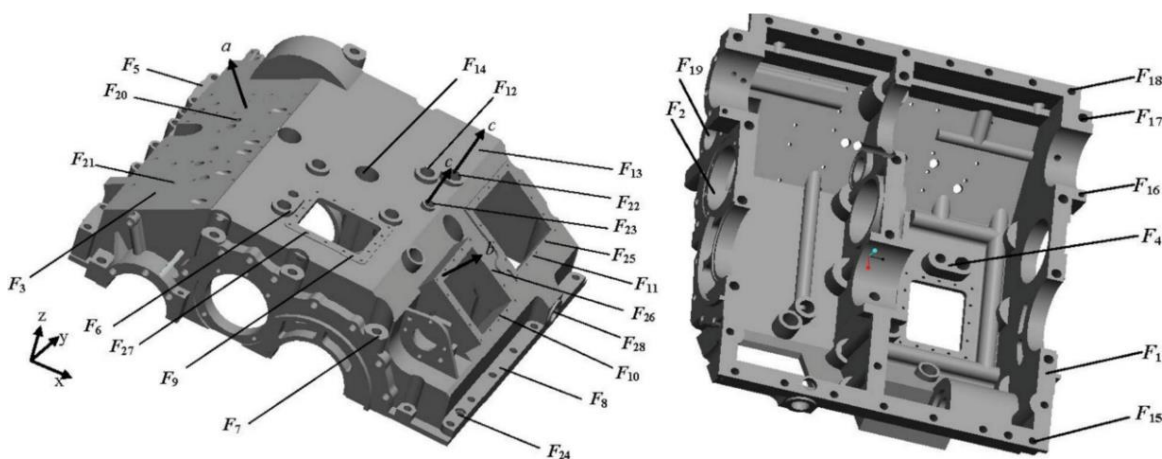
Slika 6.15. Optimizacioni grafik uticaja ulaznih parametara na predviđeni odziv (MIN, MAX i SV TWMC)

6.2.1 Četvrta eksperimentalna studija

Primer prizmatičnog dela usvojenog u četvrtom eksperimentu predložen je u [47]. Kao i u drugom eksperimentu, ova studija sprovedena je razmatranjem dva različita tehnološka uslova optimizacije:

- 1) Optimizacija tehnoloških procesa pri čemu su na raspolaganju svi tehnološki resursi. Ujedno, svi težinski koeficijenti u jednačini (16) jednaki su 1.
- 2) Optimizacija tehnoloških procesa pri čemu nisu raspoloživi svi tehnološki resursi. Mašine M3 i M7, kao i alat A8 uklonjeni su iz baze varijantnih mašina, odnosno varijantnih alata.

Reprezentativni proizvod predstavlja model gornjeg kućišta koji je prema dimenzijama problema dvostruko veći izazov u odnosu na prve tri studije kod kojih je broj tipskih tehnoloških oblika znatno manji. Na ovom primeru definisano je 28 tipskih tehnoloških oblika koje se mogu obraditi realizacijom ukupno 46 zahvata obrade. Trodimenzionalni solid model ovog dela sa konceptualnim efektom senčenja u dva pogleda dat je na slici 6.16. Odgovarajući graf i matrica prethođenja prikazani su na slici 6.17. Radi lakšeg uočavanja, odnosi prethođenja su u matrici prethođenja na slici 6.17b označeni žutom bojom. Relevantne tehnološke informacije koje se tiču tipskih tehnoloških oblika, zahvata obrade, tehnoloških resursa i prethođenja za ovaj reprezentativni deo date su u tabeli 6.29. Komponente troškova mašina i alata predstavljene su u tabeli 6.30. Sve navedene informacije iz tabela mogu se pronaći u izvoru [47].



Slika 6.16. Trodimenzionalni solid model u konceptualnom prikazu za četvrti reprezentativni deo

Tabela 6.29. Tehnološke informacije koje se tiču tipskih tehnoloških oblika, zahvata, resursa i ograničenja prethođenja za četvrti reprezentativni deo

Tipski tehn. oblici	Zahvati	Mašine{ }	Alati{ }	SPA{ }	Zahvati koji prethode
TO1 – Ravna površina	Grubo glodanje (Z1)	M1,M2	A1,A2,A3	+z	
	Poluzavršno glodanje (Z2)	M2	A1,A2,A3	+z	Z1,Z22
	Završno glodanje (Z3)	M2	A1,A2,A3	+z	Z1,Z22,Z2
TO2 – Otvor za ležaj	Gruba unutrašnja obrada (Z4)	M7, M8	A4	-y,+y	Z1,Z22,Z2,Z3,Z27
	Poluzavršna unutrašnja obrada (Z5)	M6,M7,M8	A5	-y,+y	Z1,Z22,Z2,Z3,Z27, Z4
	Završna unutrašnja obrada (Z6)	M6,M7,M8	A5	-y,+y	Z1,Z22,Z2,Z3,Z27, Z4,Z5
TO3 – Ugaona površina	Grubo glodanje (Z7)	M5,M7,M8	A7,A8,A9	-a	Z1,Z22,Z2,Z3,Z27, Z4
	Završno glodanje (Z8)	M3,M4,M6	A7,A8,A9	-a	Z1,Z22,Z2,Z3,Z28, Z31,Z32,Z33
TO4 – 4x ispučenje	Grubo glodanje (Z9)	M3,M4,M5	A8,A9	+z	Z1,Z22
TO5 – Ravna površina 1	Grubo glodanje (Z10)	M3,M4,M5	A8,A9	-z	Z1
TO6 – Gornje ispučenje	Grubo glodanje (Z11)	M3,M4,M5	A8,A9	-z	Z1
	Završno glodanje (Z12)	M4,M5	A8,A9	-z	Z1,Z22,Z2,Z3,Z11
TO7 – 6x ispučenje	Grubo glodanje (Z13)	M3,M4,M5	A8,A9	-z	Z1
TO8 – Ravna površina 2	Grubo glodanje (Z14)	M3,M4,M5	A8,A9	-z	Z1
TO9 – Gornja površina prozora	Grubo glodanje (Z15)	M3,M4,M5	A7,A8	-z	Z1,Z22,Z2,Z3
	Završno glodanje (Z16)	M3,M4,M5	A7,A8	-z	Z1,Z22,Z2,Z3,Z15
TO10 – Zakošena površina 1	Grubo glodanje (Z17)	M3,M4,M6,M7	A7,A8	-b	Z1,Z22,Z2,Z3
	Završno glodanje (Z18)	M3,M4,M6,M7	A7,A8	-b	Z1,Z22,Z2,Z3,Z17
TO11 - Zakošena površina 2	Grubo glodanje (Z19)	M3,M4,M6,M7	A7,A8	-b	Z1,Z22,Z2,Z3
	Završno glodanje (Z20)	M3,M4,M6,M7	A7,A8	-b	Z1,Z22,Z2,Z3,Z19
TO12 – Gornji otvori	Bušenje (Z21)	M9,M10	A10	-z,+z	Z1,Z22,Z2,Z3,Z12, Z9
TO13 – Gornja ravna površina	Grubo glodanje (Z22)	M4,M5	A7,A8,A9	-z	Z1
TO14 - Upust	Upuštanje (Z23)	M9,M10	A20	-z	Z1,Z22,Z2,Z3,Z27

Tabela 6.29. Tehnološke informacije koje se tiču tipskih tehnoloških oblika, zahvata, resursa i ograničenja prethođenja za četvrti reprezentativni deo (nastavak)

Tipski tehn. oblici	Zahvati	Mašine{ }	Alati{}	SPA{}	Zahvati koji prethode
TO15 – Otvor Ø18H7	Bušenje (Z24)	M9,M10	A11	+z	Z1,Z22,Z2,Z3
	Razvrtanje (Z25)	M9,M10	A22	+z	Z1,Z22,Z2,Z3,Z34
TO16 – 2x otvor Ø12,5	Bušenje (Z26)	M9,M10	A12	+z	Z1,Z22,Z2,Z3
TO17 – 12x otvor Ø21	Bušenje (Z27)	M9,M10	A13	+z	Z1,Z22,Z2,Z3
TO18 – 18x otvor Ø17	Bušenje (Z28)	M9,M10	A14	+z	Z1,Z22,Z2,Z3
TO19 – Poprečni otvor	Gruba unutrašnja obrada (Z29)	M7,M8	A6	-y	Z1,Z22,Z2,Z3
	Završna unutrašnja obrada (Z30)	M6,M7,M8	A6	-y	Z1,Z22,Z2,Z3,Z29
TO20 – 5x otvor Ø20	Bušenje (Z31)	M7,M8, M9,M10	A15	-a	Z1,Z22,Z2,Z3,Z7, Z28
TO21 – 24x otvor Ø8	Bušenje (Z32)	M7,M8,M9,M10	A16	-a	Z1,Z22,Z2,Z3,Z7,Z28
	Urezivanje navoja (Z33)	M7,M8,M9,M10	A23	-a	Z1,Z22,Z2,Z3,Z7, Z28,Z32
TO22 – Otvor za ulje 1	Razvrtanje (Z34)	M6,M7,M8	A28	-c	Z1,Z22
	Urezivanje navoja (Z35)	M6,M7,M8	A24	-c	Z1,Z22,Z34
TO23 – Otvor za ulje 2	Bušenje (Z36)	M6,M7,M8	A17	-c	Z1,Z22
	Urezivanje navoja (Z37)	M6,M7,M8	A25	-c	Z1,Z22,Z36
TO24 - Ø23 upust	Valjkasto upuštanje (Z38)	M3,M4,M9,M10	A21	-z	Z1,Z22,Z2,Z3,Z24,Z25
TO25 – Otvori na zakošenoj površini 1	Bušenje (Z39)	M9,M10	A18	-b	Z1,Z22,Z2,Z3,Z17, Z18,Z27,Z4
	Urezivanje navoja (Z40)	M9,M10	A26	-b	Z1,Z22,Z2,Z3,Z17, Z18,Z27,Z4,Z39
TO26 - Otvori na zakošenoj površini 2	Bušenje (Z41)	M9,M10	A18	-b	Z1,Z22,Z2,Z3,Z29, Z19,Z20
	Urezivanje navoja (Z42)	M9,M10	A26	-b	Z1,Z22,Z2,Z3,Z29, Z19,Z20,Z41
TO27 - Gornji otvori	Bušenje (Z43)	M3,M4,M9,M10	A18	-z	Z1,Z22,Z2,Z3,Z15, Z16
	Urezivanje navoja (Z44)	M3,M4,M9,M10	A26	-z	Z1,Z22,Z2,Z3,Z15,Z16,Z43
TO28 – Otvor za ulje 3	Bušenje (Z45)	M6,M7,M8	A19	-x	Z1,Z22,Z2,Z3,Z28
	Urezivanje navoja (Z46)	M6,M7,M8	A27	-x	Z1,Z22,Z2,Z3,Z28, Z45

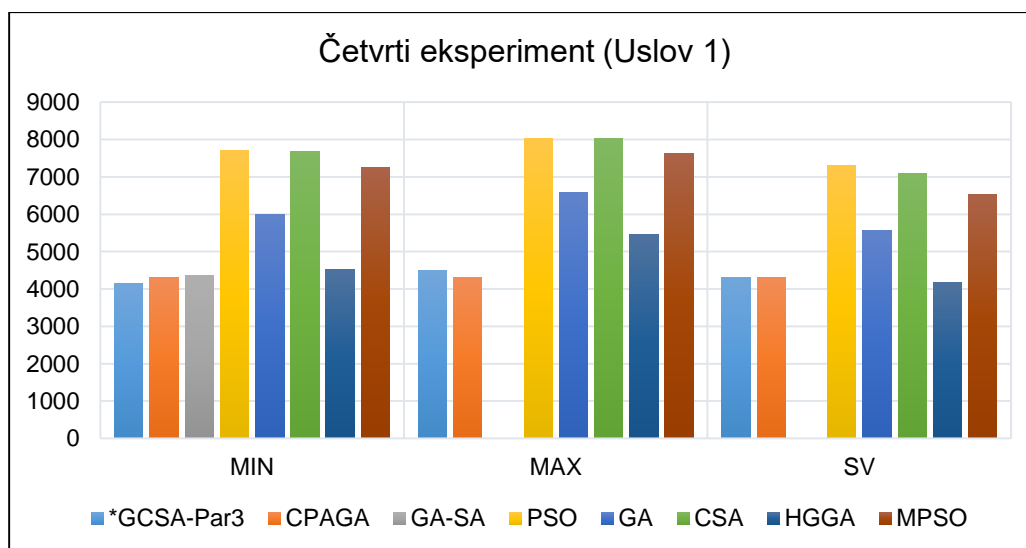
Tabela 6.30. Informacija o troškovima za četvrti reprezentativni deo

Redni broj	Naziv troškova	Novčana vrednost
Troškovi rada (korišćenja) mašina:		
	Broj modela	Tip mašine
M1	C5112A	Vertikalna glodalica
M2	CK5116D	CNC glodalica
M3	XK5763A/1	CNC glodalica
M4	XK5763A/2	CNC glodalica
M5	X53T	Glodalica
M6	TK6111	Bušilica-glodalica
M7	TSPX619	Bušilica-glodalica
M8	TX6111T	Bušilica-glodalica
M9	Z3050	Radijalna bušilica
M10	Z3060	Radijalna bušilica
Troškovi korišćenja alata:		
A1	Glodala	5
A2		6
A3		7
A4	Alati za unutrašnju obradu	12
A5		13
A6		9
A7	Glodala	8
A8		9
A9		10
A10	Burgije	4
A11		4
A12		3
A13		4
A14		3
A15		4
A16		3
A17		4
A18		2
A19	2	
A20	Konusni ili valjkasti upuštači	4
A21		3
A22	Razvrtač	5
A23	Ureznici	3
A24		4
A25		4
A26		3
A27		4
A28		Razvrtač
Troškovi promene:		
MCI	Mašina	120
TCI	Alata	15
SCI	Stezanja (SPA)	90

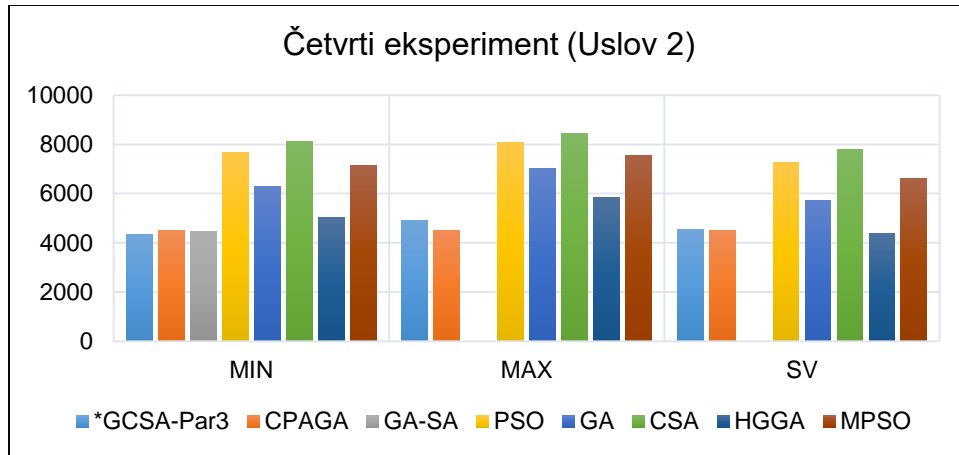
Imajući u vidu da predloženi model kućišta pripada eksperimentalnim primerima koji su nedavno predstavljeni u naučnoj literaturi, mali broj verifikacija performansi metaheurističkih metoda je do sada na ovom primeru izvršen. Rezultati dva relevantna algoritamska rešenja iz literature preuzeta su i priključena ovoj uporednoj analizi. Radi se o hibridnim algoritmima CPAGA [142] i GA-SA [47]. Rezultati analize predstavljeni su u tabeli 6.31 gde su pored GCSA, CPAGA i GA-SA uključeni sledeći pristupi: PSO, GA, CSA, HGGA i MPSO. Dijagrami sa prikazanim statističkim rezultatima iz tabele 6.31 dati su na slikama 6.18 i 6.19.

Tabela 6.31. Uporedni rezultati GCSA-Par3 sa drugim metaheurističkim algoritmima za četvrti eksperiment

Algoritam	Uslov 1			Uslov 2		
	SV	MAX	MIN	SV	MAX	MIN
*GCSA-Par3	4306,5	4496	4135	4544,7	4924	4338
CPAGA	4302	4315	4299	4503	4503	4503
GA-SA	/	/	4368	/	/	4450
PSO	7713,7	8033	7304	7665,5	8058	7271
GA	5984,05	6579	5568	6305,7	7027	5717
CSA	7673,8	8029	7088	8135,6	8421	7785
HGGA	4516,7	5451	4166	5022,7	5835	4371
MPSO	7252,1	7620	6539	7158,4	7551	6610



Slika 6.18. Uporedni rezultati ostvareni različitim metaheurističkim metodama za četvrti eksperiment deo i prvi tehnološki uslov



Slika 6.19. Uporedni rezultati ostvareni različitim metaheurističkim metodama za četvrti eksperiment deo i drugi tehnološki uslov

Analogno prethodnim studijama, GCSA-Par3 algoritam je zajedno sa drugim uporednim metodama izvršen u 20 ponavljanja. Optimalni tehnološki proces za prvi tehnološki uslov optimizacije dobijen pomoću GCSA algoritma prikazan je u tabeli 6.32. Optimalni tehnološki proces za drugi tehnološki uslov dat je u tabeli 6.33.

Postignuti rezultati funkcije cilja f_c i troškova TWMC za šest različitih metaheurističkih pristupa, GCSA-Par3, PSO, GA, CSA, HGGA i MPSO predstavljeni su za oba tehnološka uslova u tabelama 6.34 i 6.35. U podnožju tabela mogu se videti standardna devijacija rezultata i srednja vrednost troškova TWMC za svaki algoritam pojedinačno. Uporedni prikaz kriva konvergencije za šest uporednih metoda i oba tehnološka uslova optimizacije predstavljen je na slici 6.20.

Tabela 6.32. Optimalni tehnološki proces za četvrti eksperiment i prvi tehnološki uslov

Indeksi	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23
Zahvati	1	22	2	3	10	11	15	16	12	14	13	17	18	19	20	9	27	24	28	26	25	45	46
Mašine	1	5	2	2	5	5	5	5	5	5	5	5	5	5	5	5	9	9	9	9	9	7	7
Alati	1	7	1	1	8	8	8	8	8	8	8	8	8	8	8	8	13	11	14	12	22	19	27
SPA	+z	-z	+z	+z	-z	-z	-z	-z	-z	-z	-z	-b	-b	-b	-b	+z	+z	+z	+z	+z	+z	-x	-x
Indeksi	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46
Zahvati	36	37	34	35	7	32	33	31	4	5	6	29	30	8	41	39	40	42	23	38	43	44	21
Mašine	7	7	7	7	7	7	7	7	7	7	7	7	7	3	9	9	9	9	9	9	9	9	9
Alati	17	25	28	24	8	16	23	15	4	5	5	6	6	8	18	18	26	26	20	21	18	26	10
SPA	-c	-c	-c	-c	-a	-a	-a	-a	-y	-y	-y	-y	-y	-a	-b	-b	-b	-b	-z	-z	-z	-z	-z

TMC=1319, TMCC=840, NMC=7, TTC=281, TTCC=435, NTC=29, TSC=1260, NSC=14, TWMC=4135

Tabela 6.33. Najbolji tehnološki proces za četvrti eksperiment i drugi tehnološki uslov

Indeksi	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23
Zahvati	1	22	2	3	15	16	11	14	13	10	12	9	24	28	27	25	26	7	31	32	33	4	5
Mašine	1	5	2	2	5	5	5	5	5	5	5	5	9	9	9	9	9	8	8	8	8	8	8
Alati	1	7	1	1	7	7	9	9	9	9	9	9	11	14	13	22	12	7	15	16	23	4	5
SPA	+z	-z	+z	+z	-z	-z	-z	-z	-z	-z	-z	+z	+z	+z	+z	+z	+z	-a	-a	-a	-a	-y	-y
Indeksi	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46
Zahvati	6	29	30	45	46	34	35	36	37	17	18	19	20	8	41	39	42	40	43	23	21	44	38
Mašine	8	8	8	8	8	8	8	8	8	4	4	4	4	4	9	9	9	9	9	9	9	9	9
Alati	5	6	6	19	27	28	24	17	25	7	7	7	7	7	18	18	26	26	18	20	10	26	21
SPA	-y	-y	-y	-x	-x	-c	-c	-c	-c	-b	-b	-b	-b	-a	-b	-b	-b	-b	-z	-z	-z	-z	-z

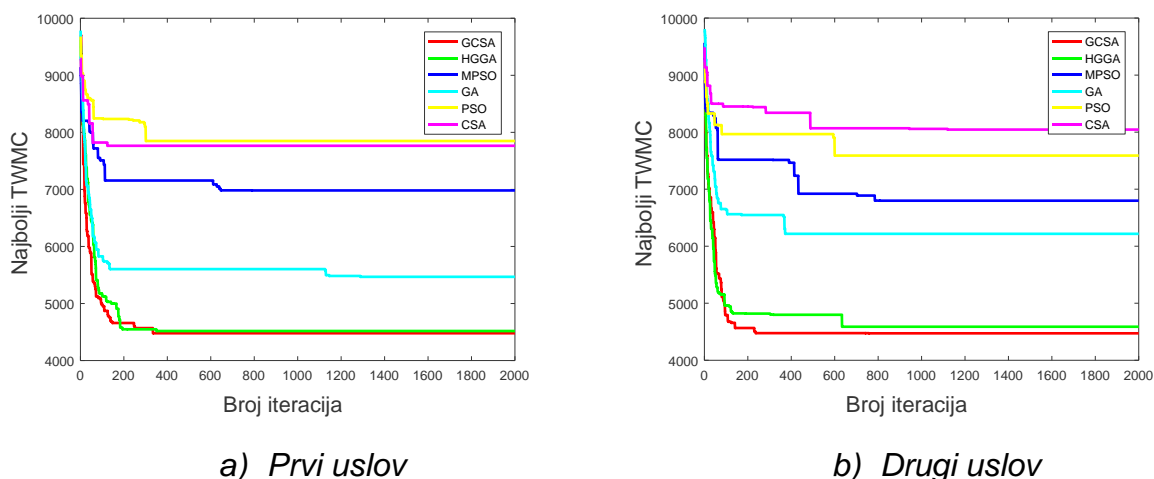
TMC=1509, TMCC=840, NMC=7, TTC=279, TTCC=450, NTC=30, TSC=1260, NSC=14, TWMC=4338

Tabela 6.34. Izlazni rezultati četvrtog eksperimenta za prvi tehnološki uslov dobijeni u 20 ponavljanja

Br.	Maksimizacija funkcije cilja fc i minimizacija troškova TWMC													
	GCSA-Par3		HGGA		MPSO		PSO		GA		CSA			
	fc	TWMC	fc	TWMC	fc	TWMC	fc	TWMC	fc	TWMC	fc	TWMC		
1.	0,00023	4390	0,00022	4639	0,00014	7316	0,00013	7681	0,00016	6267	0,00013	7636		
2.	0,00024	4189	0,00023	4362	0,00013	7481	0,00013	7888	0,00018	5588	0,00013	7912		
3.	0,00024	4137	0,00024	4166	0,00014	7082	0,00012	8033	0,00017	5798	0,00013	7799		
4.	0,00023	4429	0,00022	4602	0,00013	7577	0,00013	7664	0,00016	6089	0,00013	7792		
5.	0,00023	4339	0,00022	4605	0,00014	7209	0,00013	7882	0,00017	5866	0,00013	7494		
6.	0,00023	4371	0,00021	4704	0,00013	7499	0,00013	7802	0,00018	5568	0,00013	7969		
7.	0,00023	4356	0,00022	4635	0,00014	7088	0,00013	7607	0,00016	6189	0,00013	7887		
8.	0,00023	4356	0,00023	4338	0,00015	6539	0,00013	7734	0,00017	5795	0,00013	7572		
9.	0,00023	4374	0,00023	4256	0,00014	7065	0,00014	7330	0,00016	6113	0,00014	7399		
10.	0,00022	4496	0,00022	4575	0,00014	7143	0,00013	7930	0,00015	6579	0,00013	7652		
11.	0,00023	4391	0,00023	4315	0,00014	7339	0,00013	7959	0,00017	6040	0,00013	7713		
12.	0,00024	4211	0,00023	4396	0,00013	7530	0,00013	7851	0,00016	6264	0,00013	7757		
13.	0,00023	4354	0,00021	4662	0,00013	7420	0,00013	7519	0,00017	6009	0,00013	7816		
14.	0,00023	4326	0,00023	4344	0,00014	7374	0,00013	7836	0,00017	6026	0,00013	7799		
15.	0,00023	4271	0,00018	5451	0,00014	6901	0,00013	7645	0,00017	5893	0,00013	7854		
16.	0,00023	4271	0,00022	4464	0,00014	7158	0,00013	7493	0,00016	6075	0,00013	7672		
17.	0,00024	4249	0,00024	4254	0,00014	7056	0,00013	7443	0,00017	5731	0,00013	7690		
18.	0,00023	4349	0,00022	4566	0,00013	7620	0,00013	7836	0,00017	5801	0,00013	7488		
19.	0,00024	4136	0,00022	4575	0,00014	7111	0,00014	7304	0,00017	6047	0,00013	7794		
20.	0,00024	4135	0,00023	4424	0,00013	7534	0,00013	7836	0,00017	5943	0,00013	7764		
Statistički parametri (Standardna devijacija rezultata i srednja vrednost troškova)														
	σ	SV	σ	SV	σ	SV	σ	SV	σ	SV	σ	SV	SV	
	100,03	4306,5	264,07	4516,65	261,66	7252,1	204,23	7713,65	236,81	5984,05	145,77	7722,95		

Tabela 6.35. Izlazni rezultati četvrtog eksperimenta za drugi tehnološki uslov dobijeni u 20 ponavljanja

Br.	Maksimizacija funkcije cilja fc i minimizacija troškova TWMC																	
	GCSA-Par3		HGGA		MPSO		PSO		GA		CSA							
	fc	TWMC	fc	TWMC	fc	TWMC	fc	TWMC	fc	TWMC	fc	TWMC						
1.	0,00022	4580	0,0002	5055	0,00014	7241	0,00016	7600	0,00013	6394	0,00012	8141						
2.	0,00022	4550	0,00022	4507	0,00014	7153	0,00016	7656	0,00013	6290	0,00012	8241						
3.	0,00022	4484	0,0002	5100	0,00014	7350	0,00016	8010	0,00012	6375	0,00012	8263						
4.	0,00021	4818	0,0002	5114	0,00014	7141	0,00016	7878	0,00013	6321	0,00012	8159						
5.	0,00022	4567	0,0002	5021	0,00014	7081	0,00016	7611	0,00013	6269	0,00012	8195						
6.	0,00022	4477	0,0002	4924	0,00014	7217	0,00017	7664	0,00013	5717	0,00013	7785						
7.	0,00022	4460	0,00019	5315	0,00014	7182	0,00015	7666	0,00013	6699	0,00012	8222						
8.	0,00023	4340	0,00023	4371	0,00014	7232	0,00016	7754	0,00013	6430	0,00012	8188						
9.	0,00021	4700	0,00017	5835	0,00015	6742	0,00015	7380	0,00014	6502	0,00013	7956						
10.	0,00023	4398	0,00018	5430	0,00013	7551	0,00016	7298	0,00014	6181	0,00013	7861						
11.	0,00022	4580	0,00019	5304	0,00014	6962	0,00016	7436	0,00013	6221	0,00012	8002						
12.	0,00022	4471	0,00019	5174	0,00015	6610	0,00014	7505	0,00013	7027	0,00012	8241						
13.	0,00022	4533	0,00021	4727	0,00014	7288	0,00017	7490	0,00013	5956	0,00012	8381						
14.	0,00023	4338	0,00019	5352	0,00014	6967	0,00015	8058	0,00012	6620	0,00012	8128						
15.	0,00022	4458	0,00021	4831	0,00014	7383	0,00016	8041	0,00012	6140	0,00012	8126						
16.	0,00021	4797	0,00021	4668	0,00014	7244	0,00017	7695	0,00013	5991	0,00012	8118						
17.	0,00021	4728	0,00021	4749	0,00014	7351	0,00016	7271	0,00014	6217	0,00013	7888						
18.	0,00022	4462	0,00019	5362	0,00014	7298	0,00015	7652	0,00013	6616	0,00012	8061						
19.	0,00022	4454	0,00021	4740	0,00014	7323	0,00016	7749	0,00013	6440	0,00012	8421						
20.	0,00022	4580	0,00021	4875	0,00015	6852	0,00017	7896	0,00013	5808	0,00012	8334						
	Statistički parametri (Standardna devijacija rezultata i srednja vrednost troškova)																	
	σ	SV	σ	SV	σ	SV	σ	SV	σ	SV	σ	SV	σ	SV	σ	SV	σ	SV
	135,6	4538,75	353,86	5022,7	231,11	7158,4	231,32	7665,5	305,84	6305,7	170,07	8135,55						



Slika 6.20. Krive konvergencije za četvrti eksperiment i dva tehnološka uslova

6.2.1.1 Diskusija rezultata četvrte eksperimentalne studije

Optimizacijom tehnološkog procesa obrade kućišta u četvrtoj studiji doktorske disertacije cilj je bio testirati i verifikovati modifikovani algoritam vrane na primeru većih dimenzija u pogledu broja tipskih tehnoloških oblika i varijanti tehnoloških resursa.

Posmatrajući uporedne rezultate iz tabele 6.31 za prvi tehnološki uslov optimizacije, može se izvesti zaključak da GCSA-Par3 nadmašuje učinak svih ostalih metoda uključenih u uporednu studiju. Prema tabeli 6.34, može se zapaziti da je ostvarena minimalna vrednost troškova TWMC od 4135 jedinica pored koje najbliže vrednosti TWMC iznose 4136 i 4137 jedinica. Srednji rezultat optimizacije primenom GCSA-Par3 iznosi 4306,5, dok najlošiji rezultat, odnosno maksimalni troškovi TWMC u 20 ponavljanja iznose 4496 jedinica. Ostvareni troškovi TWMC u samo dve egzekucije algoritma prelaze vrednost od 4400, dok su ostali rezultati u okvirima srednje vrednosti TWMC. Ukupno četiri tehnološka procesa prelaze vrednost ispod 4200 u 20 ponavljanja algoritma. GCSA-Par3 je pokazao superiornost u pogledu minimalnih rezultata TWMC u poređenju sa prvenstveno dva savremena hibridna rešenja, CPAGA i GA-SA što potvrđuju uporedni rezultata iz tabele 6.31. Tradicionalni algoritmi CSA, PSO i GA, kao ni modifikovani mPSO nisu pokazali zavidan učinak. Sa druge strane, hibridni HGGA je pokazao veoma dobre kapacitete za pronalaženje minimalne vrednosti koja je neznatno veća od minimuma ostvarenog pomoću GCSA-Par3. Kada su u pitanju najlošiji ostvareni rezultat, kao i srednja vrednost rezultata optimizacije, hibridni CPAGA pristup je pokazao najbolju konzistentnost sa najmanjim standardnim odstupanjem u odnosu na ostale metode. Predloženi GCSA-Par3 nalazi se na sledećem mestu sa malo većom srednjom vrednošću rezultata koja iznosi 4306,5 jedinica nasuprot 4302 koliko je ostvareno primenom CPAGA. HGGA je pokazao malo slabije performanse kada su u pitanju srednja i maksimalna vrednost rezultata, dok se ostali algoritmi, kao u pogledu minimalne vrednosti, nisu pokazali dovoljno efikasnim.

Posmatranjem rezultata optimizacije za drugi tehnološki uslov, mogu se doneti slični zaključci kao kod prethodnog. Prema podacima iz tabele 6.31, u pogledu minimalnih troškova TWMC, algoritam GCSA-Par3 ponovo je ponovo pokazao odlične performanse pri čemu je ostvario minimalnu vrednost od 4338 jedinica koja je znatno bolja u odnosu na minimalni rezultat savremenih CPAGA i GA-SA. Posmatrajući detaljne rezultate u tabeli 6.35, minimalni rezultat pojavio se tri puta u 20 egzekucija algoritma, dok je više od polovine ostvarenih vrednosti TWMC iznosilo manje od 4600 novčanih jedinica. HGGA se takođe izdvaja po pitanju pronalaženja minimalnih TWMC sa drugom najboljom vrednošću. Konzistentost rezultata najbolje pokazuje CPAGA algoritam sa najboljom srednjom i maksimalnom vrednošću troškova (tabela 6.31). Predloženi GCSA-Par3 pristup pokazuje manja, odnosno veća odstupanja u pogledu srednje i maksimalne vrednosti u odnosu na CPAGA pristup, respektivno. Sa druge strane, GCSA-Par3 se pokazao znatno naprednijom optimizacionom metodom u poređenju sa HGGA i MPSO, kao i tradicionalnim GA, PSO i CSA algoritmima što potvrđuju rezultati funkcije cilja f_c i troškova TWMC za 20 izvršenja algoritama (Tabela 6.35).

Konvergencija ka optimalnom rešenju testirana je za pet razvijenih algoritama, GCSA-Par3, HGGA, MPSO, PSO, CSA i GA. Prema slici 6.20, jasno se vidi da najbržu konvergenciju možemo pripisati hibridnom HGGA i predloženom GCSA-Par3 algoritmu koji u većoj meri prevazilaze sporu konvergenciju ostalih metoda. Na osnovu prikazanog može se uočiti i razlika u konvergiranju klasičnog CSA i predloženog GCSA-Par3 pristupa koja je drastično poboljšana uvođenjem genetskih operatora. Kao i CSA, ostale tradicionalne metode zahtevaju dodatne mehanizme i operatore za poboljšanje lokalne pretrage i izbegavanje lokalnih optimuma. Na taj način uvećale bi se njihove performanse i ubrzala konvergencija prilikom rešavanja složenijih problema kao što je to slučaj sa optimizacijom tehnoloških procesa.

6.3 Podešavanje parametara i analiza rezultata pete i šeste eksperimentalne studije

U ovom poglavlju predstavljeni su eksperimentalni rezultati dveju studija kod kojih je izvršena optimizacija ukupnog vremena obrade delova. U pitanju su delovi rotacionog oblika za koje je primenjen drugačiji prilaz u poređenju sa prva četiri eksperimenta. Fleksibilnost tehnoloških procesa u pogledu varijantnih tehnoloških resursa i zahvata obrade prikazani su pomoću I/ILI mrežnih grafova koji u obzir uzimaju različite tipove i nivoe fleksibilnosti koji su detaljno opisani u odeljku 2.2.6. Za optimizaciju ovih primera predlažu se dva novorazvijena algoritma, i to modifikovani algoritam sivog vuka, mGWO, i modifikovani algoritam grbavog kita, mWOA. Ostvarene performanse predloženih metoda upoređene su sa odabranim tradicionalnim metaheuristikama i uporedna analiza je realizovana na identičan način kao kod prethodnih eksperimenata.

Podešavanje parametara za peti i šesti eksperiment izvršeno je nakon nekoliko manuelnih testova sa ciljem dobijanja pogodnih rezultata. U skladu sa time, sledeći ulazni parametri usvojeni su za peti eksperiment:

- *mWOA*: maksimalni broj iteracija *MaxIt* je 100, veličina jata kitova *PodSize* je 30, promenljiva \vec{a} linearno opada od 0 do 1;
- *mGWO*: maksimalni broj iteracija *MaxIt* je 100, veličina čopora *SearchAgents_no* je 30;

S obzirom da su obe metode modifikovane genetskim operatorima, identične vrednosti ovih parametara usvojene su za *mWOA* i *mGWO*:

- verovatnoća ukrštanja p_c je 0,6; verovatnoća mutacije resursa p_m je 0,6 i verovatnoća operatora zamene p_s je 0,2.

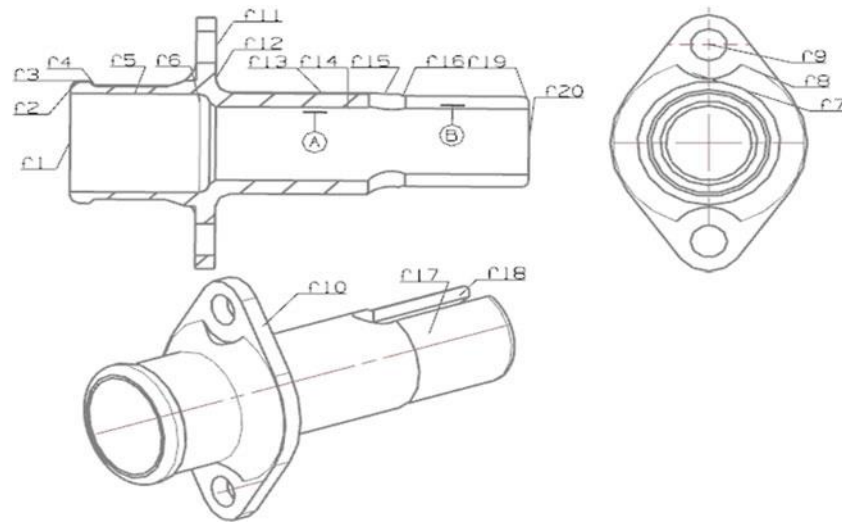
Na sličan način usvojeni su ulazni parametri za šesti eksperiment:

- *mWOA*: maksimalni broj iteracija *MaxIt* je 200, veličina jata kitova *PodSize* je 80, veličina \vec{a} linearno opada od 0 do 1;
- *mGWO*: maksimalni broj iteracija *MaxIt* je 200, veličina čopora *SearchAgents_no* je 80;
- verovatnoća ukrštanja p_c je 0,9; verovatnoća mutacije resursa p_m je 0,6 i verovatnoća operatora zamene p_s je 0,3.

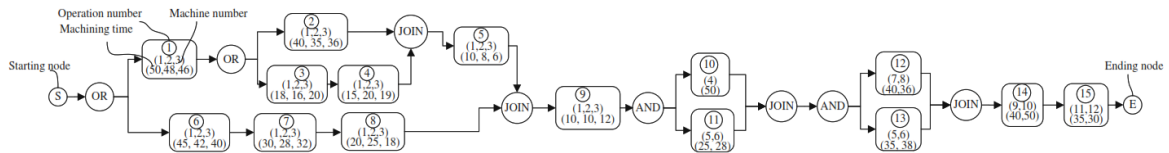
6.3.1 Peta eksperimentalna studija

Peti reprezentativni primer u ovom istraživanju preuzet je iz [67] i predstavlja prvu studiju u ovom istraživanju u kojem je izvršena minimizacija ukupnog vremena obrade. Radi se o delu rotacionog oblika koji sadrži 19 tipskih tehnoloških oblika koji se mogu obraditi pomoću maksimalno 15 podoperacija obrade. Na slici 6.21 pored trodimenzionalnog modela dela koji je izrađen od legiranog čelika 12CrNi3 prikazana je I/ILI mreža na kojoj su predstavljene moguće varijante tehnološkog procesa. Informacije o tipskim tehnološkim oblicima, kvalitetu tolerancija i hrapavosti površina, zahvatima obrade i varijantnim mašinama predstavljeni su u tabeli 6.36. S obzirom da se u ovom slučaju ne razmatraju fleksibilnosti, odnosno varijante alata i SPA, zahvati su definisani tako da formiraju odgovarajuće podoperacije. U tabeli 6.37 definisan je sadržaj alternativnih podoperacija. Tabela 6.38 prikazuje informaciju o mašinama za realizovanje podoperacija obrade ovog rotacionog dela, vremena transporta između mašina i vremena trajanja podoperacija na svakoj mašini. Šema kodiranja jedinke koja reprezentuje jedan tehnološki proces sa svim potencijalnim podoperacijama obrade koji formiraju fleksibilne redoslede prikazana je u tabeli 6.39. Pored vektora podoperacija, mašina i vremena obrade u podnožju tabele nalaze se odabrani ILI konektori na osnovu kojih se definiše odabrana putanja na mrežnom grafu.

Minimizacijom ukupnog vremena obrade pomoću funkcije cilja ft definisane pomoću izraza (24) u poglavlju 2.3, izvršena je uporedna analiza nekoliko metaheurističkih algoritama. Imajući u vidu manje dimenzije ove instance problema OTP koje se prvenstveno ogledaju u nedostatku varijantnih alata i SPA, relativno zadovoljavajući rezultati mogu se dobiti primenom tradicionalnih algoritama kao što je PSO. Sa ciljem da se ukaže na mogućnost unapređenja pretraživačkih kapaciteta tradicionalnih metoda, predloženi su *mGWO* i *mWOA* čije su performanse upoređene sa performansama tradicionalnih metaheuristika.



(a)



(b)

Slika 6.21. Trodimenzionalni solid model rotacionog dela (a) i I/ILI mreža za peti eksperiment i reprezentativni deo rotacionog oblika [67]

Tabela 6.36. Informacije o tipskim oblicima, zahvatima i alternativnim mašinama za peti reprezentativni deo

Tipski oblik	Kvalitet tolerancije (Ra, μm)	Naziv zahvata	Mašine{ }
TO1	IT14 (6,3)	Struganje	M1, M2, M3
TO2	IT14 (6,3)	Struganje	M1, M2, M3
TO3	IT14 (6,3)	Struganje	M1, M2, M3
TO4	IT14 (6,3)	Struganje	M1, M2, M3
TO5	IT14 (6,3)	Bušenje – Proširivanje	M1, M2, M3

Tabela 6.36. Informacije o tipskim oblicima, zahvatima i alternativnim mašinama za peti reprezentativni deo (nastavak)

Tipski oblik	Kvalitet tolerancije (Ra, μm)	Naziv zahvata	Mašine{ }
TO6	IT14 (6,3)	Bušenje – Proširivanje	M1, M2, M3
TO7	IT14 (6,3)	Upuštanje	M1, M2, M3
TO8	IT14 (6,3)	Struganje	M1, M2, M3
TO9	IT14 (6,3)	Bušenje	M5, M6
TO10	IT14 (12,5)	Profilno glodanje	M4
TO11	IT14 (6,3)	Struganje	M1, M2, M3
TO12	IT14 (6,3)	Grubo struganje	M1, M2, M3
TO13	IT6 (0,63)	Grubo struganje – Poluzavršno struganje – Završno struganje - Brušenje	M1, M2, M3, M11, M12
TO14	IT9 (0,63)	Bušenje – Proširivanje – Unutrašnje struganje – Grubo brušenje – Fino brušenje	M1, M2, M3, M9, M10
TO15	IT14 (6,3)	Bušenje	M5, M6
TO16	IT14 (6,3)	Struganje	M1, M2, M3
TO17	IT10 (1,25)	Grubo glodanje – Fino glodanje	M7, M8
TO18	IT14 (6,3)	Struganje	M1, M2, M3
TO19	IT14 (6,3)	Struganje	M1, M2, M3

Tabela 6.37. Detaljne informacije o zahvatima i podoperacijama za peti eksperiment

Broj podoperacije	Skup zahvata
P1	Struganje TO19, TO18, TO16, TO13 i TO12 Bušenje TO14 i TO5
P2	Struganje TO1, TO2, TO3, TO4 i TO8 Proširivanje TO5, TO6 i TO14
P3	Struganje TO1, TO2, TO3, TO4 i TO8 Bušenje TO5 i TO14
P4	Proširivanje TO14
P5	Unutrašnje struganje TO14
P6	Struganje TO19, TO18, TO16, TO13 i TO12
P7	Struganje TO1, TO2, TO3, TO4 i TO8
P8	Bušenje TO5, TO14, Proširivanje TO5, TO6 i TO14 i Unutrašnje struganje TO14
P9	Poluzavršno i završno struganje TO13
P10	Profilno glodanje TO10
P11	Upuštanje TO7 i Bušenje TO9
P12	Grubo glodanje i fino glodanje TO17
P13	Bušenje TO15
P14	Brušenje TO13
P15	Grubo i fino brušenje TO14

Tabela 6.38. Varijantne mašine i transportna vremena između mašina

Naziv mašine		Broj Mašine	1	2	3	4	5	6	7	8	9	10	11	12
M1	Univerzalni strug 1	1	0	2	3	1	3	5	3	2	4	3	7	6
M2	Univerzalni strug 2	2	2	0	2	6	2	2	3	5	2	5	4	5
M3	CNC strug	3	3	2	0	1	2	2	3	1	2	1	4	3
M4	Vertikalna glodalica 1	4	1	6	1	0	3	2	3	1	2	4	4	2
M5	Stubna bušilica 1	5	3	3	2	3	0	1	6	5	4	4	3	2
M6	Stubna bušilica 2	6	5	4	2	2	1	0	5	4	3	6	7	2
M7	Vertikalna glodalica 2	7	3	3	3	3	6	5	0	6	5	4	1	6
M8	Vertikalna glodalica 3	8	2	5	1	1	5	4	6	0	3	5	7	5
M9	Brusilica za unutrašnje brušenje 1	9	4	2	2	2	4	3	5	3	0	2	4	6
M10	Brusilica za unutrašnje brušenje 2	10	3	5	1	4	4	6	4	5	2	0	5	2
M11	Brusilica za okruglo brušenje 2	11	7	4	4	4	3	7	1	7	4	5	0	3
M12	Brusilica za okruglo brušenje 3	12	6	5	3	2	2	2	6	5	6	2	3	0
Jedinična vremena korišćenja mašine			5	5	5	10	5	4	6	5	9	8	7	7

Tabela 6.39. Šema kodiranja varijantnog tehnološkog procesa za peti reprezentativni deo

Podoperacije	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Mašine	3	3	3	3	3	3	3	3	3	4	5	8	5	12	9
Vremena	46	36	20	19	6	40	32	18	12	50	25	36	35	30	40
ILI konektori or1=0, or2=0															

Rezultati optimizacije rotacionog modela u petom eksperimentu uporedno su prikazani u tabeli 6.40. Srednja, maksimalna i minimalna vrednost vremena realizacije zahvata obrade za razmatrani deo, kao i standardna devijacija rezultata dobijeni su za četiri klasična algoritma, GWO, WOA, CSA, PSO pored modifikovanih algoritama mGWO i mWOA. Sa druge strane, radi temeljnije analize u tabelu 6.40 su uneta minimalna vremena ostvarena pomoću GA, GP, CEAND i CENAND čiji se rezultati mogu pronaći detaljnije u [67]. Optimalni tehnološki proces sa aspekta redosleda tipskih zahvata obrade koji su grupisani u odgovarajuće podoperacije predstavljeno je u tabeli 6.41. Generisano rešenje ostvareno je pomoću mGWO algoritma. Dijagrami ostvarenih rezultata iz tabele 6.40 prikazani su na slikama 6.22 i 6.23.

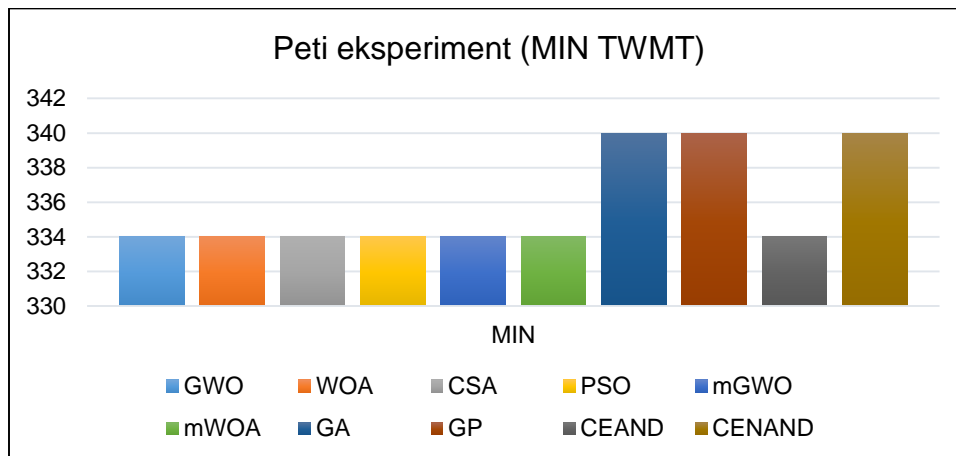
Tabela 6.40. Uporedni rezultati za peti eksperiment

Algoritam	MIN	MAX	SV	STD
GWO	334	340	336,25	2,1734
WOA	334	341	337,3	1,9222
CSA	334	340	336,05	1,9595
PSO	334	340	336,15	1,6631
mGWO	334	334	334	0
mWOA	334	336	334,1	0,4472
GA	340	-	-	-
GP	340	-	-	-
CEAND	334	-	-	-
CENAND	340	-	-	-

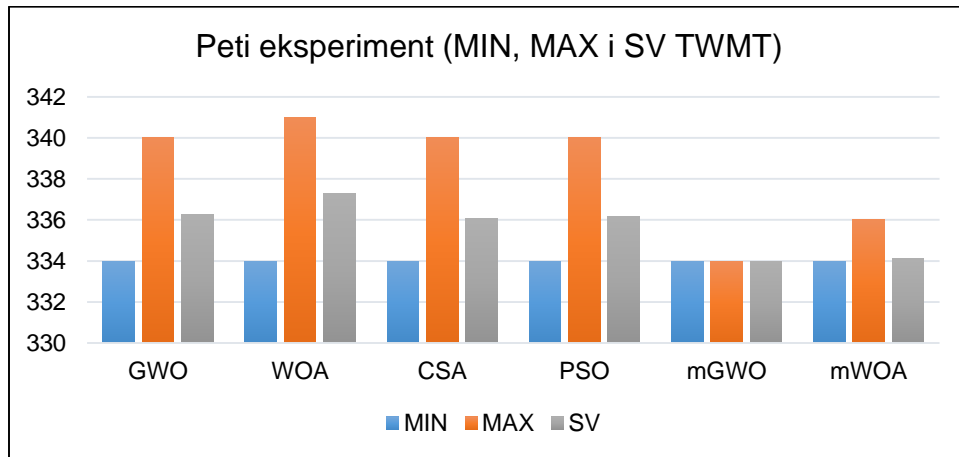
Tabela 6.41. Najbolji tehnološki proces za peti reprezentativni deo - mGWO

Podoperacije	1	2	5	9	10	11	13	12	14	15
Mašine	3	3	3	3	4	5	5	8	12	9
Vremena	46	36	6	12	50	25	35	36	30	40

Ukupno vreme obrade: 334
 TW=316; TT=18; TWMT=334; $f_t=0,0029941$

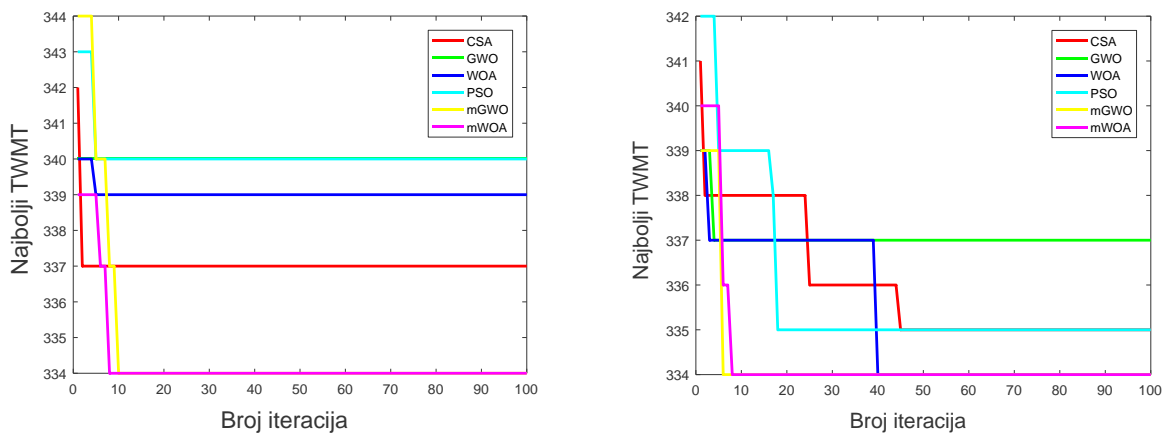


Slika 6.22. Uporedni rezultati različitih metaheurističkih algoritama za peti eksperiment (minimalni troškovi TWMT)



Slika 6.23. Uporedni rezultati različitih metaheurističkih algoritama za peti eksperiment (minimalni, maksimalni i srednji troškovi TWMT)

Krive konvergencije koje ukazuju na približavanje optimalnom vremenu TWMT grafički su predstavljene na slici 6.24. Prikazane su dve različite krive za iste optimizacione uslove. Detaljan pregled rezultata optimizacije postignutih u 20 ponavljanja razmatranih metaheurističkih algoritama dat je u tabeli 6.42.



Slika 6.24. Krive konvergencije za peti reprezentativni deo

Tabela 6.42. Izlazni rezultati petog eksperimenta dobijeni u 20 ponavljanja

Br.	Maksimizacija funkcije cilja f_t i minimizacija vremena TWMT																	
	CSA		GWO		WOA		PSO		mGWO		mWOA							
	f_t	TWMT	f_t	TWMT	f_t	TWMT	f_t	TWMT	f_t	TWMT	f_t	TWMT						
1.	0,00299	334	0,00295	339	0,00297	337	0,00298	336	0,00299	334	0,00299	334						
2.	0,00296	338	0,00299	334	0,00296	338	0,00299	334	0,00299	334	0,00299	334						
3.	0,00299	335	0,00295	339	0,00299	334	0,00297	337	0,00299	334	0,00299	334						
4.	0,00295	339	0,00295	339	0,00298	336	0,00296	338	0,00299	334	0,00299	334						
5.	0,00297	337	0,00297	337	0,00298	336	0,00297	337	0,00299	334	0,00299	334						
6.	0,00297	337	0,00298	336	0,00295	339	0,00298	336	0,00299	334	0,00299	334						
7.	0,00297	337	0,00294	340	0,00295	339	0,00294	340	0,00299	334	0,00299	334						
8.	0,00294	340	0,00298	336	0,00297	337	0,00297	337	0,00299	334	0,00299	334						
9.	0,00299	334	0,00299	334	0,00296	338	0,00297	337	0,00299	334	0,00299	334						
10.	0,00297	337	0,00299	334	0,00298	336	0,00298	336	0,00299	334	0,00298	336						
11.	0,00297	337	0,00297	337	0,00298	336	0,00299	335	0,00299	334	0,00299	334						
12.	0,00297	337	0,00297	337	0,00298	336	0,00295	339	0,00299	334	0,00299	334						
13.	0,00296	338	0,00294	340	0,00293	341	0,00299	335	0,00299	334	0,00299	334						
14.	0,00299	334	0,00299	335	0,00297	337	0,00299	334	0,00299	334	0,00299	334						
15.	0,00299	334	0,00298	336	0,00297	337	0,00299	334	0,00299	334	0,00299	334						
16.	0,00299	334	0,00298	336	0,00294	340	0,00299	335	0,00299	334	0,00299	334						
17.	0,00297	337	0,00299	334	0,00298	336	0,00299	334	0,00299	334	0,00299	334						
18.	0,00299	334	0,00299	334	0,00294	340	0,00298	336	0,00299	334	0,00299	334						
19.	0,00299	334	0,00299	334	0,00295	339	0,00297	337	0,00299	334	0,00299	334						
20.	0,00299	334	0,00299	334	0,00299	334	0,00298	336	0,00299	334	0,00299	334						
Statistički parametri (Standardna devijacija rezultata i srednja vrednost troškova)																		
σ	1,96	336,05	2,17	336,25	1,92	337,3	1,66	336,15	0	334	0,45	334,1						

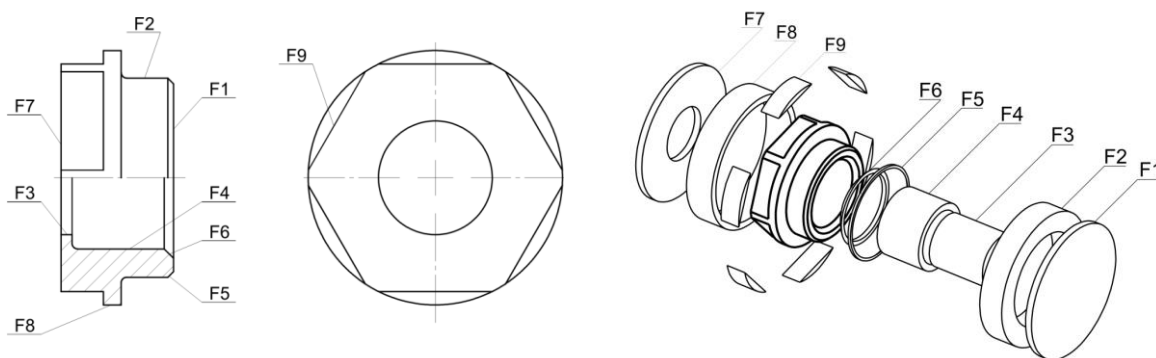
6.3.1.1 Diskusija rezultata pete eksperimentalne studije

Na osnovu sprovedenog petog eksperimenta, izvedeni su određeni zaključci radi verifikacije performansi predloženih mGWO i mWOA algoritama. Prema prikazanim rezultatima iz tabela 6.40 i 6.42, minimalnu vrednost ukupnog vremena obrade od 334 vremenske jedinice pronašli su svi metaheuristički algoritmi pri čemu srednja vrednost rezultata pokazuje vrlo malu standardnu devijaciju za 20 egzekucija. PSO algoritam pokazuje zanemarljivo mala odstupanja u odnosu na CSA i WOA i nešto veće odstupanje u odnosu na rezultate postignute pomoću GWO algoritma. U pogledu na modifikovane pristupe, jasno se može uočiti potpuna konzistentnost u postizanju minimalnih vremena TWMT kod modifikovanog mGWO. Sa druge strane, mWOA pruža bolje rezultate od klasičnog WOA algoritma pri čemu je standardna devijacija rezultata manja u odnosu na ostale algoritme. Iz literaturnih izvora preuzeti su rezultati CEAND i CENAND metoda [67] koji su ostvarili minimalnu vrednost vremena, dok su GA [60] i GP [69] pristupi pokazali za nijansu slabije performanse u odnosu na druge algoritme.

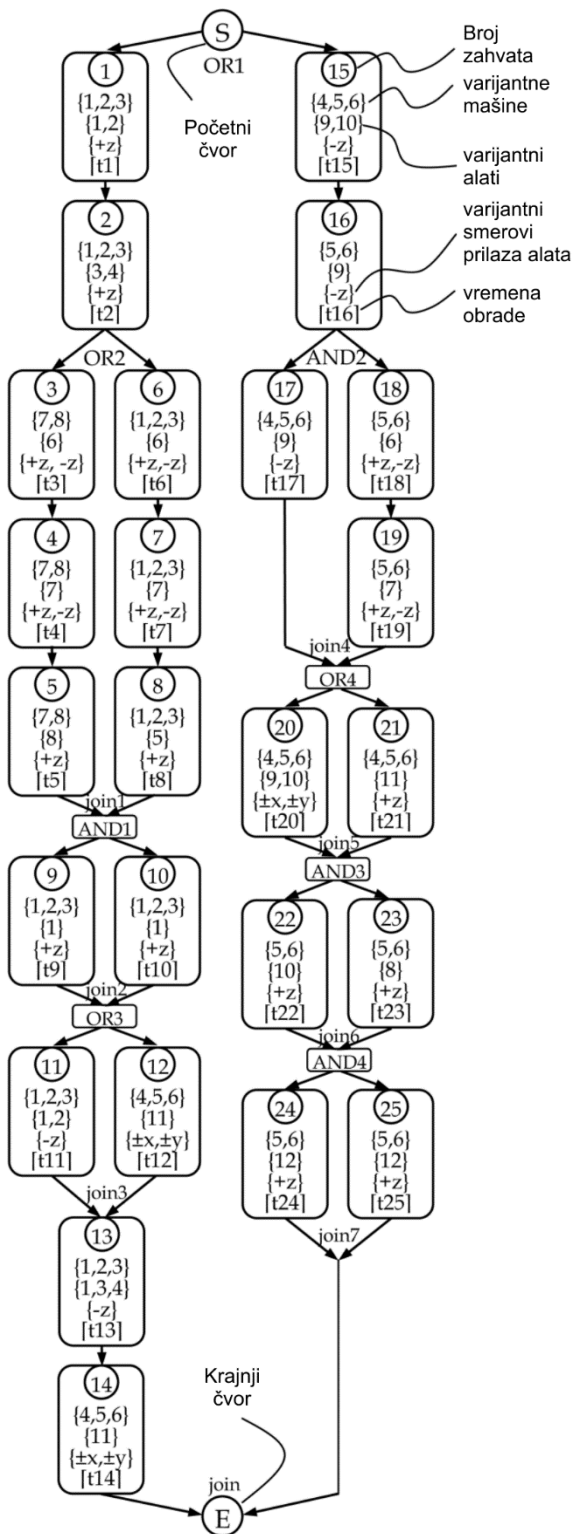
Krive konvergencije za peti eksperiment koje su date na slici 6.24 pokazuju bržu konvergenciju algoritama mGWO i mWOA u odnosu na ostale metode. Očito se može primetiti napredak pre svega u poređenju sa učinkom tradicionalnih GWO i WOA algoritama.

6.3.2 Šesta eksperimentalna studija

Šesti reprezentativni deo bio je predmet istraživanja u [2,21]. Slično prethodnom primeru izvršena je minimizacija ukupnog vremena obrade reprezentativnog dela rotacionog oblika. Radi se o delu za koji definisano 25 potencijalnih zahvata za obradu ukupno 9 tipskih tehnoloških oblika. Trodimenzionalni model ovog dela prikazan je na slici 6.25. Sa ciljem prikaza fleksibilnosti tehnoloških procesa razvijena je I/ILI mreža koja se nalazi na slici 6.26. Tehnološke specifikacije koje se tiču tipskih tehnoloških oblika, zahvata obrade i tehnoloških resursa sa kvalitetima tolerancije i hrapavosti obrađene površine dati su u tabeli 6.43. Postupak kodiranja tehnološkog procesa detaljno je opisan u odeljku 5.2.2.1 a shema kodiranja predstavljena je u tabeli 5.2. Dekodirana varijanta jednog izvodljivog tehnološkog procesa sa 10 zahvata obrade prikazana je u tabeli 5.3.



Slika 6.25. Trodimenzionalni model za šesti eksperiment



t1 = [2.3, 3.3, 4.6, 6.5, 3.8, 5.5]
t2 = [1.2, 1.0, 1.9, 1.5, 2.0, 1.6]
t3 = [5.5, 3.8]
t4 = [54.8, 38.1]
t5 = [34.3, 13.7]
t6 = [3.3, 4.8, 3.3]
t7 = [13.4, 26.6, 30.2]
t8 = [1.1, 1.5, 1.8]
t9 = [0.6, 0.7, 0.8]
t10 = [0.7, 0.8, 0.9]
t11 = [2.3, 3.3, 4.6, 6.5, 3.8, 5.5]
t12 = [5.7, 4.2, 2.8]
t13 = [0.5, 0.7, 0.4, 1.0, 1.4, 0.8, 0.8, 1.1, 0.7]
t14 = [10.8, 9.7, 7.4]
t15 = [2.7, 2.8, 2.9, 2.4, 2.8, 2.5]
t16 = [4.6, 7.2]
t17 = [4.3, 3.5, 6.8]
t18 = [3.3, 3.4]
t19 = [10.7, 24.1]
t20 = [2.7, 2.8, 2.9, 2.4, 2.9, 2.4]
t21 = [5.7, 4.2, 2.8]
t22 = [3.1, 3.2]
t23 = [30.2, 13.4]
t24 = [0.4, 0.5]
t25 = [0.5, 0.6]

Slika 6.26. I/II mreža (b) za šesti reprezentivni deo rotacionog oblika sa slike 6.25 [2]

Tabela 6.43. Informacije o tipskim oblicima, zahvatima i tehnološkim resursima za šesti reprezentativni deo [2]

Tipski oblik	Kvalitet tolerancije (Ra, μm)	Naziv zahvata	Mašine{ }	Alati{ }	SPA{ }
TO1	IT11 (6,3)	Struganje (Z1)	M1,M2,M3	A1,A2	+z
		Glodanje (Z20)	M4,M5,M6	A9,A10	+x,-x,+y,-y
		Glodanje (Z21)	M4,M5,M6	A11	+z
TO2	IT11 (6,3)	Struganje (Z2)	M1,M2,M3	A3,A4	+z
		Glodanje (Z22)	M5,M6	A10	+z
TO3	IT10 (3,2)	Bušenje (Z3)	M7,M8	A6	+z,-z
		Proširivanje (Z4)	M7,M8	A7	+z,-z
TO3	IT10 (3,2)	Bušenje (Z6)	M1,M2,M3	A6	+z,-z
		Proširivanje (Z7)	M1,M2,M3	A7	+z,-z
TO3	IT10 (3,2)	Bušenje (Z18)	M5,M6	A6	+z,-z
		Proširivanje (Z19)	M5,M6	A7	+z,-z
TO4	IT8 (1,6)	Proširivanje (Z5)	M7,M8	A8	+z
		Unutrašnje struganje (Z8)	M1,M2,M3	A5	+z
		Proširivanje (Z23)	M5,M6	A8	+z
TO5	IT11 (6,3)	Struganje (Z9)	M1,M2,M3	A1	+z
		Glodanje (Z24)	M5,M6	A12	+z
TO6	IT11 (6,3)	Struganje (Z10)	M1,M2,M3	A1	+z
		Glodanje (Z25)	M5,M6	A12	+z
TO7	IT11 (6,3)	Struganje (Z11)	M1,M2,M3	A1,A2	-z
		Glodanje (Z12)	M4,M5,M6	A11	+x,-x,+y,-y
		Glodanje (Z15)	M4,M5,M6	A9,A10	-z
TO8	IT11 (6,3)	Struganje (Z13)	M1,M2,M3	A1,A3,A4	-z
		Glodanje (Z16)	M5,M6	A9	-z
TO9	IT11 (6,3)	Struganje (Z14)	M4,M5,M6	A11	+x,-x,+y,-y
		Glodanje (Z17)	M4,M5,M6	A9	-z

Za obradu razmatranog rotacionog dela na raspolaganju je osam alternativnih mašina: M1-CNC strug, M2-Univerzalni strug, M3-Produkcioni strug, M4-Univerzalna glodalica, M5-CNC vertikalna glodalica, M6-CNC vertikalna glodalica, M7-Bušilica i M8-Koordinatna bušilica. Alternativni alati koji se koriste za formiranje tipskih oblika su sledeći: A1-Strugarski nož 1, A2-Strugarski nož 2, A3-Strugarski nož 3, A4-Strugarski nož 4, A5-Nož za unutrašnju obradu, A6-Burgija, A7-Proširivač 1, A8- Proširivač 2, A9-Glodalo, A10-Glodalo 2, A11-Glodalo 3 i A12-Glodalo 4. Transportna vremena između različitih mašina sa vremenskim indeksima koji se odnose na promene alata i SPA nalaze se u tabeli 6.44.

Tabela 6.44. Transportna vremena između mašina i vremenski indeksi za šesti reprezentativni deo [2]

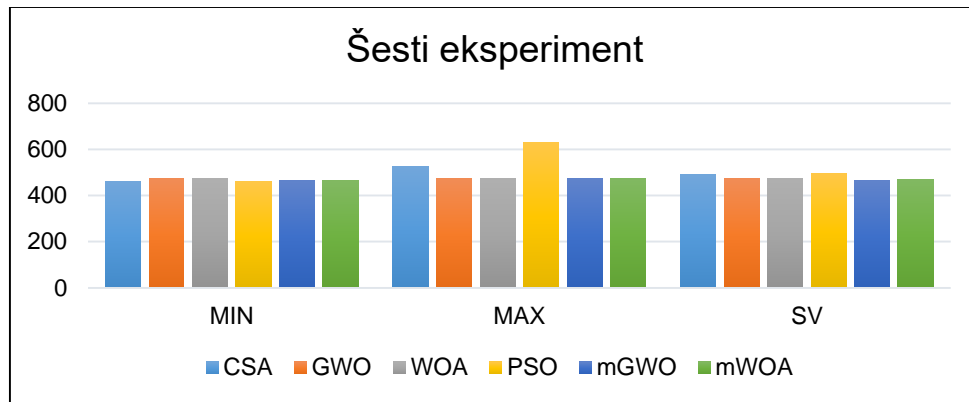
Broj Mašine	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
1	0	4	8	10	12	5	6	14	13	18	6	5	9	11	7
2	4	0	3	7	11	5	4	6	10	13	12	8	2	2	9
3	8	3	0	5	7	9	8	4	6	10	12	7	4	6	9
4	10	7	5	0	4	14	12	6	5	6	7	8	2	3	14
5	12	11	7	4	0	18	12	10	6	4	4	9	5	5	8
6	5	5	9	14	18	0	6	8	12	15	3	10	2	7	4
7	6	4	8	12	12	6	0	3	7	10	4	5	6	7	8
8	14	6	4	6	10	8	3	0	4	8	12	9	6	4	3
9	13	10	6	5	6	12	7	4	0	4	5	10	9	8	7
10	18	13	10	6	4	15	10	8	4	0	2	5	8	9	4
11	6	12	12	7	4	3	4	12	5	2	0	6	7	7	7
12	5	8	7	8	9	10	5	9	10	5	6	0	3	4	12
13	9	2	4	2	5	2	6	6	9	8	7	3	0	10	11
14	11	2	6	3	5	7	7	4	8	9	7	4	10	0	16
15	7	9	9	14	8	4	8	3	7	4	7	12	11	16	0

Troškovi promene alata: 60
Troškovi promene stezanja: 50

U skladu sa usvojenim parametrima za šesti eksperiment, modifikovani algoritmi sivog vuka i grbavog kita su razvijeni i njihove performanse su verifikovane. U uporednoj studiji analizirani su i upoređeni rezultati koje su postigli mGWO i mWOA sa rezultatima koje su postigli ostali algoritmi. Uporedni rezultati optimizacije razmatranog rotacionog dela dati su u tabeli 6.45. Dijagram sa prikazom uporednih rezultata nalazi se na slici 6.27. U uporednu analizu rezultata uključeni su algoritmi CSA, GWO, WOA, PSO, kao i modifikovani mGWO i mWOA. Rezultati pokazuju osnovne statističke parametre, minimalnu, maksimalnu vrednost ukupnog vremena TWMT kao i srednju vrednost i standardnu devijaciju rezultata za 20 egzekucija algoritama. Optimalni tehnološki procesi u dekodiranoj formi koje su pronašli savremeni metaheuristički algoritmi, CSA, mGWO i mWOA respektivno, prikazani su u tabelama 6.46 i 6.47.

Tabela 6.45. Uporedni rezultati za šesti reprezentativni deo

Algoritam	MIN	MAX	SV	STD
CSA	461,6	526,8	489,94	25,96
GWO	473,4	473,4	473,4	0
WOA	473,4	473,4	473,4	0
PSO	461,6	632,4	497,19	46,58
mGWO	464,4	473,4	465,75	3,297
mWOA	464,4	473,4	469,8	4,524



Slika 6.27. Uporedni rezultati ostvareni različitim metaheurističkim algoritmima za šesti eksperiment

Tabela 6.46. Optimalni tehnološki proces za šesti eksperiment – CSA

Zahvati	15	16	17	18	19	20	22	23	25	24
Mašine	5	5	5	5	5	5	5	5	5	5
Alati	9	9	9	6	7	10	10	8	12	12
SPA	-z	-z	-z	-z	-z	+y	+z	+z	+z	+z
Vremena	2,9	4,6	3,5	3,3	10,7	2,4	3,1	30,2	0,5	0,4
Ukupno vreme obrade: 461,6										
TW=61,6; TT=0; TCT=300; SCT=100; TWMT=461,6; $ft=0,0022$										

Tabela 6.47. Optimalni tehnološki proces za šesti eksperiment – mGWO i mWOA

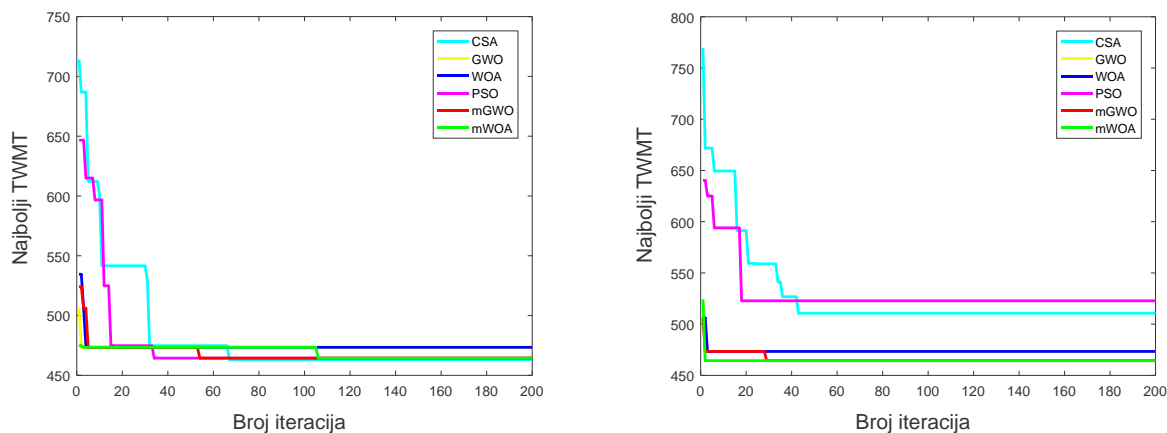
Zahvati	15	16	17	18	19	20	22	23	25	24
Mašine	6	6	6	6	6	6	6	6	6	6
Alati	9	9	9	6	7	10	10	8	12	12
SPA	-z	-z	-z	-z	-z	-y	+z	+z	+z	+z
Vremena	2,8	7,2	6,8	3,4	24,1	2,4	3,2	13,4	0,6	0,5
Ukupno vreme obrade: 464,6										
TW=64,4; TT=0; TCT=300; SCT=100; TWMT=464,4; $ft=0,0022$										

Detaljan prikaz izlaznih rezultata optimizacije tehnološkog procesa, pri čemu je izvršena minimizacija vremena TWMT za šesti reprezentativni deo rotacionog oblika, dat je u tabeli 6.48.

Tabela 6.48. Izlazni rezultati šestog eksperimenta dobijeni u 20 ponavljanja

Br.	Maksimizacija funkcije cilja f_i i minimizacija vremena TWMT																	
	CSA		GWO		WOA		PSO		mGWO		mWOA							
	f_i	TWMT	f_i	TWMT	f_i	TWMT	f_i	TWMT	f_i	TWMT	f_i	TWMT						
1.	0,0022	463,1	0,00211	473,4	0,00211	473,4	0,00217	461,6	0,00211	473,4	0,00211	473,4						
2.	0,0022	463,1	0,00211	473,4	0,00211	473,4	0,00211	474,8	0,00215	464,4	0,00211	473,4						
3.	0,002	497,8	0,00211	473,4	0,00211	473,4	0,00216	463,1	0,00215	464,4	0,00211	473,4						
4.	0,0022	461,6	0,00211	473,4	0,00211	473,4	0,00183	547,3	0,00215	464,4	0,00211	473,4						
5.	0,0019	526,8	0,00211	473,4	0,00211	473,4	0,00198	506	0,00215	464,4	0,00211	473,4						
6.	0,0022	463,1	0,00211	473,4	0,00211	473,4	0,00215	464,4	0,00215	464,4	0,00215	464,4						
7.	0,0022	464,4	0,00211	473,4	0,00211	473,4	0,00211	473,5	0,00215	464,4	0,00211	473,4						
8.	0,0022	461,6	0,00211	473,4	0,00211	473,4	0,00211	474,8	0,00215	464,4	0,00211	473,4						
9.	0,002	510,7	0,00211	473,4	0,00211	473,4	0,00211	473,4	0,00211	473,4	0,00211	473,4						
10.	0,0019	518,5	0,00211	473,4	0,00211	473,4	0,00178	561,8	0,00215	464,4	0,00215	464,4						
11.	0,002	510,7	0,00211	473,4	0,00211	473,4	0,00215	464,4	0,00215	464,4	0,00215	464,4						
12.	0,0019	518,5	0,00211	473,4	0,00211	473,4	0,00211	474,8	0,00211	473,4	0,00215	464,4						
13.	0,0019	526,8	0,00211	473,4	0,00211	473,4	0,00179	559,3	0,00215	464,4	0,00211	473,4						
14.	0,0022	463,1	0,00211	473,4	0,00211	473,4	0,00158	632,4	0,00215	464,4	0,00211	473,4						
15.	0,0021	473,5	0,00211	473,4	0,00211	473,4	0,00215	464,4	0,00215	464,4	0,00215	464,4						
16.	0,0022	461,6	0,00211	473,4	0,00211	473,4	0,00211	473,4	0,00215	464,4	0,00215	464,4						
17.	0,002	510,7	0,00211	473,4	0,00211	473,4	0,00191	522,8	0,00215	464,4	0,00215	464,4						
18.	0,002	497,6	0,00211	473,4	0,00211	473,4	0,00215	464,4	0,00215	464,4	0,00211	473,4						
19.	0,0021	487	0,00211	473,4	0,00211	473,4	0,00191	522,8	0,00215	464,4	0,00215	464,4						
20.	0,0019	518,5	0,00211	473,4	0,00211	473,4	0,00215	464,4	0,00215	464,4	0,00211	473,4						
	Statistički parametri (Standardna devijacija rezultata i srednja vrednost troškova)																	
	σ	SV	σ	SV	σ	SV	σ	SV	σ	SV	σ	SV						
	25,96	489,94	0	473,4	0	473,4	46,58	497,19	3,297	465,75	4,524	469,8						

Na slici 6.28 mogu se videti dve različite krive konvergencije algoritama pri rešavanju problema optimizacije tehnološkog procesa obrade šestog reprezentativnog dela.



Slika 6.28 Krive konvergencije za šesti eksperiment

6.3.2.1 Diskusija rezultata šeste eksperimentalne studije

Na osnovu izlaznih rezultata za šesti deo koji su prikazani u tabeli 6.45, može se zaključiti da su klasični algoritmi CSA i PSO pokazali najbolje performanse kada je u pitanju pronalaženje minimalne vrednosti vremena TWMT. Sa druge strane, najmanje odstupanje u ostvarenim rezultatima pokazuju savremeni algoritmi GWO i WOA, dok se predloženi mGWO i mWOA pre svega izdvajaju srednjim vrednostima rezultata sa neznatno većim devijacijama u odnosu na GWO i WOA. Kako se može uočiti, predloženi modifikovani algoritmi pokazuju konzistentnost u pogledu ostvarenih rezultata i pronalaženja približno optimalnih troškova TWMT od 464,4 vremenske jedinice.

U pogledu na izlazne rezultate u tabeli 6.48 u kojoj su dati izlazni rezultati za 20 ponavljanja, dobijene vrednosti funkcije cilja f_t i vremena TWMT ukazuju na to da je klasični CSA algoritam ostvario minimalnu vrednost od 461,6 vremenskih jedinica koja se pojavila tri puta u 20 ponavljanja eksperimenta. Sa druge strane, znatno bolju konzistentnost rezultata pokazuju GWO i WOA algoritmi, koji iako dostižu veću vrednost ukupnog vremena, imaju nulto odstupanje u istom broju egzekucija algoritma. Novorazvijeni mGWO i mWOA sa neznatno većim odstupanjem dostižu relativno bolju vrednost vremena TWMT koja se može smatrati blisko optimalnom u poređenju sa minimalnom vrednošću koju su ostvarili klasični CSA i PSO.

7. ZAKLJUČNA RAZMATRANJA

Tehnološka priprema proizvodnje, odnosno projektovanje tehnoloških procesa jedna je od ključnih funkcija savremenih inteligentnih tehnoloških sistema. Na osnovu nje omogućava se veza između projektovanja proizvoda i procesa njihove proizvodnje. Sve složeniji uslovi poslovanja i tržišni zahtevi nameću potrebu uvođenja savremenih tehnika i metoda projektovanja na bazi informacionih tehnologija u okviru tehnološke pripreme. Sa naglaskom na složenost razvoja sistema za projektovanje tehnoloških procesa, uz podršku savremenih informacionih tehnologija, jedan od osnovnih zadataka koji se rešava je tehnoeekonomska optimizacija pojedinih aktivnosti tehnološke pripreme. Cilj tehnoeekonomske optimizacije je da ispuni tehničke zahteve kvaliteta proizvoda sa jedne strane, i ekonomske zahteve sa druge.

Predmet istraživanja u ovoj disertaciji bila je optimizacija tehnoloških procesa obrade proizvoda koja u prvi plan izdvaja varijantnost tehnoloških procesa u pogledu redosleda tipskih zahvata obrade i varijantnosti u pogledu mašina, reznih alata i pribora kojima se definišu smerovi pristupa alata. Pažnja je posvećena pre svega određivanju optimalnih redosleda zahvata obrade i optimalnoj selekciji navedenih tehnoloških resursa pri čemu je izvršena minimizacija ukupnih troškova i vremena obrade delova prizmatičnog i rotacionog oblika.

Analizom problema optimizacije i varijantnosti tehnoloških procesa predstavljeni su uslovi i ograničenja prethođenja koji kao važan segment optimizacije direktno utiču na izvodljivost redosleda zahvata, a samim tim i tehnološkog procesa. Ograničenja prethođenja predstavljena su grafičkim i matičnim putem koristeći tzv. grafove i matrice prethođenja. U skladu sa ovim ograničenjima omogućeno je i predstavljanje tehnološkog procesa u pogledu optimizacije redosleda izvođenja zahvata obrade. Odabrana su dva prilaza u predstavljanju tehnoloških procesa obrade: predstavljanje pomoću vektora i predstavljanje pomoću I/II mreže. Prvi prilaz se odnosi na prikaz tehnoloških procesa obrade u vidu vektora, pri čemu glavni vektor predstavlja redosled zahvata obrade. Ostali vektori predstavljaju vektore mašina, alata i smerova prilaza alata. Manipulacijom ovih vektora moguće je generisanje neizvodljivih redosleda zahvata, odnosno tehnoloških procesa. Sa druge strane, I/II mreže pružaju mogućnost predstavljanja osnovnih tipova fleksibilnosti i putem odabira odgovarajuće putanje generišu izvodljive tehnološke

procesu.

U cilju optimizacije tehnoloških procesa odabrani su kriterijumi optimizacije i to: minimizacija ukupnih troškova i minimizacija ukupnog vremena obrade delova. Matematičkim modelima predstavljene su osnovne komponente ukupnih troškova i vremena obrade kao i funkcije cilja čije se vrednosti maksimizuju.

Uzimajući u obzir kompleksnost ovakvog problema koji se svrstava u polje teških problema diskretne optimizacije, nametnula se potreba za primenom jednostavnih, ali i efikasnih metoda koje će u zadovoljavajućem vremenskom periodu pronaći optimalna i/ili približno optimalna rešenja. U tu svrhu, primenjeni su savremeni metaheuristički algoritmi, metode veštačke inteligencije koje su široku primenu pronašle u okvirima inženjerske optimizacije.

Pregled dosadašnjih istraživanja u disertaciji ukazao je na popularnost razmatranog problema za čije rešavanje su primenjeni različiti modifikovani i hibridni metaheuristički algoritmi. Obzirom da ne postoji jedinstven algoritam koji će rešiti svaki mogući problem optimizacije, kao i svaku njegovu instancu, uvek će ostati aktuelna potreba za razvojem novih i modifikacijom postojećih optimizacionih algoritama. Sa iznetim pregledom stanja u oblasti razvoja metaheuristika, naglasak je postavljen na biološki inspirisane tehnike među kojima su izdvojeni algoritam vrane, algoritam sivog vuka i algoritam grbavog kita. Osnovni cilj istraživanja pri tom je ostvaren kroz modifikaciju navedenih algoritama i njihovu primenu za rešavanje problema optimizacije tehnoloških procesa obrade.

Prva predložena metoda koja je testirana sprovođenjem četiri eksperimenta u istraživanju je modifikovani algoritam vrane (GCSA). Primenom GCSA algoritma izvršena je minimizaciji ukupnih troškova obrade reprezentativnih delova prizmatičnog oblika. Izvorni kod tradicionalnog algoritma vrane (CSA) modifikovan je uvođenjem strategija genetskog algoritma, selekcije, ukrštanja i mutacije. Pristup koji je usvojen u ovim studijama je baziran na vektorskom predstavljanju tehnoloških procesa i ograničenjima prethođenja. Podešavanje parametara ove metode izvršeno je na manuelan način kao i optimizacijom pomoću metode faktornog eksperimenta. Nakon klasičnih koraka CSA usledila je primena operatora turnirske selekcije čiji zadatak je bio odabir najboljih vrana jedinki koje će biti poslate u narednu fazu ukrštanja. Novi 3SX operator ukrštanja uveo je diverzitet u proces pretraživanja kroz ukrštanje dve vrane jedinke i generisanje novih jedinki potomaka. Zatim, dva operatora mutacije vrana jedinki primenjeni su radi poboljšanja globalnih kapaciteta GCSA predložene metode. To su operator zamene i operator resursa koji uvode slučajne izmene u vektorima redosleda zahvata, mašina, alata i smerova prilaza alata. Pored toga, adaptivni koeficijenti za ukrštanje i mutaciju doprinose uravnoteženom pretraživanju sa izraženim globalnim i lokalnim svojstvima. Razvijeni algoritam koristi mehanizam popravke čiji je zadatak usklađivanje dobijenih rezultata sa pravilima, odnosno ograničenjima prethođenja. Eksperimentalni rezultati optimizacije primenom ovog algoritma pokazuju da je ostvaren doprinos razvoju ove oblasti optimizacije pri čemu su za četiri reprezentativna prizmatična dela ostvareni optimalni i/ili približno optimalni tehnološki procesi. Na osnovu navedenog, može se potvrditi polazna hipoteza istraživanja kojom se tvrdi da je *„primenom savremenih biološki inspirisanih metaheurističkih algoritama moguće u skladu sa postavljenim kriterijumom optimizacije generisati optimalan ili približno optimalan tehnološki proces obrade dela koji*

se može svesti na redosled izvođenja zahvata obrade uz pronalaženje optimalnih varijanti mašina, alata i smerova prilaza alata pri obradi“.

Drugi i treći predloženi algoritam koji je primenjen u okviru istraživanja u disertaciji su modifikovani algoritam sivog vuka (mGWO) i modifikovani algoritam grbavog kita (mWOA). Obe modifikacije ovih savremenih optimizacionih tehnika bazirane su na sličnom principu kao GCSA pri čemu su od genetskih operatora usvojeni rulet selekcija, zatim jednopozicioni operator ukrštanja i mutacije zamene i resursa. Posle osnovnih koraka GWO i WOA slede genetski operatori koji omogućavaju postizanje konzistentnijih rezultata i izbegavanje lokalnih optimuma. Prilaz koji mGWO i mWOA koriste baziran je na I/IL mrežama pomoću kojih su definisani svi tipovi fleksibilnosti tehnoloških procesa i kroz postupke kodiranja i dekodiranja vektora tehnoloških procesa osigurana su izvodljiva rešenja. Primenom dva modifikovana savremena biološki inspirisana algoritma izvršena je minimizacija drugog kriterijuma optimizacije, ukupnog vremena obrade sa fokusom na delove rotacionog oblika.

U skladu sa iznetim, može se sa sigurnošću reći da su potvrđene i postavljene pomoćne hipoteze. Sa jedne strane, zadovoljena je hipoteza koja tvrdi da je *primenom savremenih biološki inspirisanih metaheurističkih algoritmima moguće izvršiti minimizaciju ukupnih troškova i vremena obrade mašinskih delova prizmatičnog i rotacionog oblika*. Razvojem matematičkih modela za kriterijume troškova i vremena, algoritmi su testirani u ukupno šest eksperimentalnih studija pri čemu su ostvarena optimalna i približno optimalna rešenja. Sa druge strane, takođe je potvrđena hipoteza da je *optimizacija tehnoloških procesa obrade moguća primenom različitih biološki inspirisanih optimizacionih algoritama*. Tri savremena metaheuristička algoritma su modifikovana sa ciljem prilagođavanja ovom kompleksnom problemu kombinatorne optimizacije. Rezultati optimizacije i uporedne studije potvrđuju da predloženi algoritmi idu u korak sa hibridnim i savremenim algoritmima koji su predloženi od strane drugih autora.

Jedan od pravaca za dalja istraživanja može biti usmeren ka primeni predloženih biološki inspirisanih metaheurističkih algoritama u integraciji tehnoloških procesa i planova terminiranja. Nakon klasične optimizacije tehnoloških procesa usledila bi naredna faza određivanja optimalnih planova terminiranja za odgovarajuće proizvodno okruženje. Među ovim problemima najpopularniji je tzv. „*job shop*“ problem terminiranja koji takođe spada u NP teške probleme kombinatorne optimizacije.

Posmatrajući prvenstveno modifikovani algoritam vrane (GCSA), u ovom predmetnom istraživanju budući doprinosi mogu se ostvariti kroz uvođenje savremenih pravila i tehnika pri generisanju inicijalnih rešenja. Njihovom primenom na ulazu se mogu dobiti izvodljivi tehnološki procesi. Na taj način, redosled zahvata obrade ne remeti usklađenost sa pravilima prethođenja što umanjuje potrebu za mehanizmima popravke rešenja i time utiče na smanjenje vremena centralne procesorske jedinice računara na kojem je implementacija izvršena.

Takođe, kada su u pitanju modifikovani algoritmi sivog vuka i grbavog kita koji koriste različiti prilaz problemu optimizacije tehnoloških procesa, jedan od smerova za nastavak istraživanja je potencijalno kombinovanje sa drugim metaheuristikama, zatim uvođenje drugih modifikacija kao što su haotične mape, i primena u većem broju eksperimentalnih

studija koje razmatraju delove predstavljene na I/ILI mrežama.

Na kraju, može se izdvojiti potencijal primene biološki inspirisanih metaheurističkih algoritama kao inteligentnih servisa pri razvoju koncepta proizvodnje bazirane na oblak tehnologijama u okviru Industrije 4.0. Proizvodni oblak predstavlja temelj pametne proizvodnje koja se primenjuje u savremenim proizvodnim uslovima. Ovakvo fleksibilno proizvodno okruženje namenjeno je učestalim izmenama proizvodnog procesa koje zahtevaju konstantnu optimizaciju proizvodnje, kao i maksimalno iskorišćenje proizvodnih kapaciteta i tehnoloških resursa. Predstavljeni metaheuristički algoritmi, kao inteligentni alati, mogu se primenjivati u razvoju multi-agent sistema u kojima je neophodna koordinacija i kooperativnost između agenata, pri čemu se kroz komunikaciju i usaglašavanje dolazi do rešenja zajedničkog problema.

8. LITERATURA

- [1] M. Petrović, Z. Miljković, B. Babić, "Veštačka inteligencija u konceptijskom projektovanju inteligentnih tehnoloških sistema – pregled stanja u oblasti istraživanja", *Tehnika - Mašinstvo*, vol. 62, no. 5, pp. 873-885, 2013.
- [2] M. Petrović, "Veštačka inteligencija u projektovanju inteligentnih tehnoloških sistema", Doktorska disertacija, Univerzitet u Beogradu, Mašinski fakultet, Beograd, 2016.
- [3] V. Todić, *Projektovanje tehnoloških procesa*. Novi Sad: Fakultet tehničkih nauka, 2004.
- [4] B. Denkena, M. Shpitalni, P. Kowalski, G. Molcho, Y. Zipori, "Knowledge Management in Process Planning", *CIRP Annals*, vol. 56, no. 1, pp. 175-180, 2007.
- [5] D. Lukić, "Razvoj opšteg modela tehnološke pripreme proizvodnje", Doktorska disertacija, Univerzitet u Novom Sadu, Fakultet tehničkih nauka, Novi Sad, 2012.
- [6] V. Todić, N. Penezić, D. Lukić, M. Milošević, *Tehnološka logistika i preduzetništvo*. Novi Sad: Fakultet tehničkih nauka, 2011.
- [7] V. Todić, J. Stanić, *Osnove optimizacije tehnoloških procesa izrade i konstrukcije proizvoda*. Novi Sad: Fakultet tehničkih nauka, 2002.
- [8] J. Wang, A. Kusiak, *Computational intelligence in manufacturing handbook*. CRC Press, 2000.
- [9] M. Petrović, J. Petronijević, M. Mitić, N. Vuković, A. Plemić, Z. Miljković, B. Babić, "The ant lion optimization algorithm for flexible process planning, *Proceedings of the 12th International Scientific Conference MMA 2015 – Flexible Technology*, pp. 125-128, Novi Sad, Serbia, 2015.
- [10] G. Nallakumarasamy, P. S. S. Srinivasan, K. Venkatesh Raja, R. Malayalamurthi, "Optimization of operation sequencing in CAPP using simulated annealing technique (SAT)", *The International Journal of Advanced Manufacturing Technology*, vol. 54, no. 5, pp. 721-728, 2011.

-
- [11] W. Li, S. Ong, A. Y. C. Nee, *Integrated and collaborative product development environment: technologies and implementations*. World Scientific, 2006.
- [12] W. Huang, Y. Hu, L. Cai, "An effective hybrid graph and genetic algorithm approach to process planning optimization for prismatic parts", *The International Journal of Advanced Manufacturing Technology*, vol. 62, no. 9, pp. 1219-1232, 2012.
- [13] Y. W. Guo, A. R. Mileham, G. W. Owen, W. D. Li, "Operation sequencing optimization using a particle swarm optimization approach", *Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture*, vol. 220, no. 12, pp. 1945-1958, 2006.
- [14] A. Gopala Krishna, K. Mallikarjuna Rao, "Optimisation of operations sequence in CAPP using an ant colony algorithm", *The International Journal of Advanced Manufacturing Technology*, vol. 29, no. 1, pp. 159-164, 2006.
- [15] T. Dereli, I. H. Filiz, "Optimisation of process planning functions by genetic algorithms", *Computers & Industrial Engineering*, vol. 36, no. 2, pp. 281-308, 1999.
- [16] S. Borojević, "Razvoj sistema za simultano projektovanje proizvoda i tehnoloških procesa," Doktorska disertacija, Univerzitet u Novom Sadu, Fakultet tehničkih nauka, Novi Sad, 2015.
- [17] Y. Su, X. Chu, Z. Zhang, D. Chen, "Process planning optimization on turning machine tool using a hybrid genetic algorithm with local search approach", *Advances in Mechanical Engineering*, vol. 7, no. 4, pp. 1-14, 2015.
- [18] W. Faheem, C. Hayes, J. Castano, D. Gaines, "What is manufacturing interaction?", in *Proceedings of DETC'98, ASME Design Engineering Technical Conferences*, USA, 1998.
- [19] D. Lukić, M. Milošević, M. Erić, M. Đurđev, J. Vukman, A. Antić, "Improving Manufacturing Process Planning Through the Optimization of Operation Sequencing", *Machine Design*, vol. 9, no. 4, pp. 123-132, 2017.
- [20] K. Lian, C. Zhang, X. Shao, L. Gao, "Optimization of process planning with various flexibilities using an imperialist competitive algorithm", *The International Journal of Advanced Manufacturing Technology*, vol. 59, no. 5, pp. 815-828, 2012.
- [21] M. Petrović, M. Mitić, N. Vuković, Z. Miljković, "Chaotic particle swarm optimization algorithm for flexible process planning", *The International Journal of Advanced Manufacturing Technology*, vol. 85, no. 9, pp. 2535-2555, 2016.
- [22] V. K. Leont'ev, "Discrete optimization", *Computational Mathematics and Mathematical Physics*, vol. 47, no. 2, pp. 328-340, 2007.
- [23] F. Rothlauf, "Optimization Problems", in *Design of Modern Heuristics: Principles and Application*, F. Rothlauf, Ed. Berlin, Heidelberg: Springer Berlin Heidelberg, pp. 7-44, 2011.
- [24] X.-S. Yang, *Engineering optimization: an introduction with metaheuristic applications*. John Wiley & Sons, 2010.
-

-
- [25] E.-G. Talbi, *Metaheuristics: From Design to Implementation*. John Wiley & Sons, Inc., 2009.
- [26] X.-S. Yang, S. Deb, Y.-X. Zhao, S. Fong, X. He, "Swarm intelligence: past, present and future", *Soft Computing*, vol. 22, no. 18, pp. 5923-5933, 2018.
- [27] X.-S. Yang, Z. Cui, R. Xiao, A. H. Gandomi, M. Karamanoglu, *Swarm intelligence and bio-inspired computation: theory and applications*. Newnes, 2013.
- [28] B. K. Panigrahi, Y. Shi, M.-H. Lim, *Handbook of swarm intelligence: concepts, principles and applications*. Springer Science & Business Media, 2011.
- [29] E. Bonabeau, D. d. R. D. F. Marco, M. Dorigo, G. Théraulaz, G. Theraulaz, *Swarm intelligence: from natural to artificial systems* (no. 1). Oxford university press, 1999.
- [30] A. Kaveh and N. Farhoudi, "A new optimization method: Dolphin echolocation", *Advances in Engineering Software*, vol. 59, pp. 53-70, 2013.
- [31] M. Dorigo, M. Birattari, T. J. I. c. i. m. Stutzle, "Ant colony optimization", vol. 1, no. 4, pp. 28-39, 2006.
- [32] S. Mirjalili, A. Lewis, "The Whale Optimization Algorithm", *Advances in Engineering Software*, vol. 95, pp. 51-67, 2016.
- [33] S. Mirjalili, "Dragonfly algorithm: a new meta-heuristic optimization technique for solving single-objective, discrete, and multi-objective problems", *Neural Computing and Applications*, vol. 27, no. 4, pp. 1053-1073, 2016.
- [34] S. Mirjalili, "The Ant Lion Optimizer", *Advances in Engineering Software*, vol. 83, pp. 80-98, 2015.
- [35] X.-S. J. I. j. o. b.-i. c. Yang, "Firefly algorithm, stochastic test functions and design optimisation", vol. 2, no. 2, pp. 78-84, 2010.
- [36] A. H. Gandomi, X.-S. Yang, A. H. Alavi, S. J. N. C. Talatahari, "Bat algorithm for constrained optimization tasks", vol. 22, no. 6, pp. 1239-1255, 2013.
- [37] W. Eversheim, J. Schneewind, "Computer-aided process planning—State of the art and future development", *Robotics and Computer-Integrated Manufacturing*, vol. 10, no. 1, pp. 65-70, 1993.
- [38] B. Babić, *Projektovanje tehnoloških procesa*. Beograd: Mašinski fakultet, 1999.
- [39] X.-j. Liu, H. Yi, Z.-h. Ni, "Application of ant colony optimization algorithm in process planning optimization", *Journal of Intelligent Manufacturing*, vol. 24, no. 1, pp. 1-13, 2013.
- [40] K. Lian, C. Zhang, X. Shao, Y. Zeng, "A multi-dimensional tabu search algorithm for the optimization of process planning", *Science China Technological Sciences*, vol. 54, no. 12, pp. 3211-3219, 2011.
- [41] P. Martin, "Some Aspects of Integrated Product and Manufacturing Process", in *Advances in Integrated Design and Manufacturing in Mechanical Engineering*, Dordrecht, 2005, pp. 215-226: Springer Netherlands.
- [42] J. Wang, B. Du, H. J. J. C. Ding, "A Modified Genetic Algorithm(GA) for Optimization of Process Planning", vol. 6, pp. 1430-1437, 2011.
-

-
- [43] M. Salehi, R. Tavakkoli-Moghaddam, "Application of genetic algorithm to computer-aided process planning in preliminary and detailed planning", *Engineering Applications of Artificial Intelligence*, vol. 22, no. 8, pp. 1179-1187, 2009.
- [44] J. Wang, X. Fan, S. Wan, "A Graph-Based Ant Colony Optimization Approach for Process Planning", *The Scientific World Journal*, vol. 2014, pp. 1-11, 2014.
- [45] S. Kafashi, "Integrated setup planning and operation sequencing (ISOS) using genetic algorithm", *The International Journal of Advanced Manufacturing Technology*, vol. 56, no. 5, pp. 589-600, 2011.
- [46] M. Milošević, D. Lukić, M. Đurđev, J. Vukman, A. Antić, "Genetic Algorithms in Integrated Process Planning and Scheduling – A State of The Art Review", *Proceedings in Manufacturing Systems*, vol. 11, no. 1, 2016.
- [47] W. Huang, W. Lin, S. Xu, "Application of graph theory and hybrid GA-SA for operation sequencing in a dynamic workshop environment", *Computer-Aided Design and Applications*, vol. 14, no. 2, pp. 148-159, 2017.
- [48] G. H. Ma, "An Automated Process Planning System for Prismatic Parts", Master's thesis, Republic of Singapore, National University of Singapore, 1999.
- [49] M. Petrović, M. Mitić, N. Vuković, Z. Miljković, "Optimizacija fleksibilnih tehnoloških procesa primenom algoritma baziranog na inteligenciji roja i teoriji haosa," in *39. Jupiter konferencija, 35. Simpozijum*, Mašinski fakultet, Beograd, 2014.
- [50] M. Petrović, Z. Miljković, "Biologically Inspired Optimization Algorithms for Flexible Process Planning", in *Proceedings of 5th International Conference on Advanced Manufacturing Engineering and Technologies*, Springer International Publishing, Cham, pp. 417-428, 2017.
- [51] Y. Yusof, K. Latif, "Survey on computer-aided process planning", *The International Journal of Advanced Manufacturing Technology*, vol. 75, no. 1, pp. 77-89, 2014.
- [52] X. Xu, L. Wang, S. T. Newman, "Computer-aided process planning – A critical review of recent developments and future trends", *International Journal of Computer Integrated Manufacturing*, vol. 24, no. 1, pp. 1-31, 2011.
- [53] R. Teti, S. R. T. Kumara, "Intelligent Computing Methods for Manufacturing Systems", *CIRP Annals*, vol. 46, no. 2, pp. 629-652, 1997.
- [54] M. Milosevic, D. Lukic, A. Antic, B. Lalic, M. Ficko, G. Simunovic, "e-CAPP: A distributed collaborative system for internet-based process planning", *Journal of Manufacturing Systems*, vol. 42, pp. 210-223, 2017.
- [55] M. Milošević, "Kolaborativni sistem za projektovanje tehnoloških procesa izrade proizvoda baziran na internet tehnologijama", Doktorska disertacija, Univerzitet u Novom Sadu, Fakultet tehničkih nauka, Novi Sad, 2012.
- [56] M. Sekulic, V. Pejic, M. Brezocnik, M. Gostimirovic, M. Hadzistevic, "Prediction of surface roughness in the ball-end milling process using response surface methodology, genetic algorithms, and grey wolf optimizer algorithm", *Advances in Production Engineering & Management*, vol. 13, no. 1, pp. 018-030, 2018.
-

-
- [57] Z. Miljković, M. Mitić, M. Lazarević, B. Babić, "Neural network Reinforcement Learning for visual control of robot manipulators", *Expert Systems with Applications*, vol. 40, no. 5, pp. 1721-1736, 2013.
- [58] X. Li, L. Gao, X. Shao, C. Zhang, C. Wang, "Mathematical modeling and evolutionary algorithm-based approach for integrated process planning and scheduling" *Computers & Operations Research*, vol. 37, no. 4, pp. 656-667, 2010.
- [59] X. Li, L. Gao, X. Shao, "An active learning genetic algorithm for integrated process planning and scheduling", *Expert Systems with Applications*, vol. 39, no. 8, pp. 6683-6691, 2012.
- [60] X. Shao, X. Li, L. Gao, C. Zhang, "Integration of process planning and scheduling—A modified genetic algorithm-based approach", *Computers & Operations Research*, vol. 36, no. 6, pp. 2082-2096, 2009.
- [61] X. Li, X. Shao, L. Gao, W. Qian, "An effective hybrid algorithm for integrated process planning and scheduling", *International Journal of Production Economics*, vol. 126, no. 2, pp. 289-298, 2010.
- [62] M. Petrović, N. Vuković, M. Mitić, Z. Miljković, "Integration of process planning and scheduling using chaotic particle swarm optimization algorithm", *Expert Systems with Applications*, vol. 64, pp. 569-588, 2016.
- [63] M. Milošević, D. Lukić, M. Đurđev, A. Antić, S. Borojević, "An overview of genetic algorithms for job shop scheduling problems", *Journal of Production Engineering*, vol. 18, no. 2, pp. 11-15, 2015.
- [64] R. Weill, G. Halevi, "Principles of Process Planning-A logical approach," ed: Chapman and Hall Inc., London, 1995.
- [65] M. Kalajdžić, L. Tanović, *Tehnologija obrade rezanjem – priručnik*. Beograd: Mašinski fakultet, 1998.
- [66] K. Lee, M. Jung, "Petri net application in flexible process planning", *Computers & Industrial Engineering*, vol. 27, no. 1, pp. 505-508, 1994.
- [67] S. Lv, L. Qiao, "A cross-entropy-based approach for the optimization of flexible process planning", *The International Journal of Advanced Manufacturing Technology*, vol. 68, no. 9, pp. 2099-2110, 2013.
- [68] Z. Miljković, M. Petrović, "Application of modified multi-objective particle swarm optimisation algorithm for flexible process planning problem", *International Journal of Computer Integrated Manufacturing*, vol. 30, no. 2-3, pp. 271-291, 2017.
- [69] X. Y. Li, X. Y. Shao, L. Gao, "Optimization of flexible process planning by genetic programming", *The International Journal of Advanced Manufacturing Technology*, vol. 38, no. 1, pp. 143-153, 2008.
- [70] J. Dou, J. Li, C. Su, "A discrete particle swarm optimisation for operation sequencing in CAPP", *International Journal of Production Research*, vol. 56, no. 11, pp. 3795-3814, 2018.
-

-
- [71] Y. Su, X. Chu, D. Chen, X. Sun, "A genetic algorithm for operation sequencing in CAPP using edge selection based encoding strategy", *Journal of Intelligent Manufacturing*, vol. 29, no. 2, pp. 313-332, 2018.
- [72] J. Wang, X. Wu, X. Fan, "A two-stage ant colony optimization approach based on a directed graph for process planning", *The International Journal of Advanced Manufacturing Technology*, vol. 80, no. 5, pp. 839-850, 2015.
- [73] J. Wang, X. Fan, A. Zhao, M. Yang, "A Hybrid Bat Algorithm for Process Planning Problem", *IFAC-PapersOnLine*, vol. 48, no. 3, pp. 1708-1713, 2015.
- [74] X.-y. Wen, X.-y. Li, L. Gao, H.-y. Sang, "Honey bees mating optimization algorithm for process planning problem", *Journal of Intelligent Manufacturing*, vol. 25, no. 3, pp. 459-472, 2014.
- [75] C. Xu, S. Zhang, R. Huang, B. Huang, X. Li, "NC process reuse-oriented flexible process planning optimization approach for prismatic parts", *The International Journal of Advanced Manufacturing Technology*, vol. 87, no. 1, pp. 329-351, 2016.
- [76] M. Salehi, A. Bahreininejad, "Optimization process planning using hybrid genetic algorithm and intelligent search for job shop machining", *Journal of Intelligent Manufacturing*, vol. 22, no. 4, pp. 643-652, 2011.
- [77] X. Li, L. Gao, X. Wen, "Application of an efficient modified particle swarm optimization algorithm for process planning", *The International Journal of Advanced Manufacturing Technology*, vol. 67, no. 5, pp. 1355-1369, 2013.
- [78] Y. W. Guo, A. R. Mileham, G. W. Owen, P. G. Maropoulos, W. D. Li, "Operation sequencing optimization for five-axis prismatic parts using a particle swarm optimization approach", *Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture*, vol. 223, no. 5, pp. 485-497, 2009.
- [79] L. Li, J. Y. H. Fuh, Y. F. Zhang, A. Y. C. Nee, "Application of genetic algorithm to computer-aided process planning in distributed manufacturing environments", *Robotics and Computer-Integrated Manufacturing*, vol. 21, no. 6, pp. 568-578, 2005.
- [80] Y. K. Kim, K. Park, J. Ko, "A symbiotic evolutionary algorithm for the integration of process planning and job shop scheduling", *Computers & Operations Research*, vol. 30, no. 8, pp. 1151-1171, 2003.
- [81] A. Lazar, "Heuristic knowledge discovery for archaeological data using genetic algorithms and rough sets", in *Heuristic and optimization for knowledge discovery*: IGI Global, pp. 263-278, 2002.
- [82] S. Russell, P. Norvig, "Artificial intelligence: a modern approach," 2002.
- [83] V. Kerleta, "Planiranje razvoja distributivnih mreža korišćenjem unapređenog heurističkog pristupa", Doktorska disertacija, Univerzitet u Novom Sadu, Fakultet tehničkih nauka, Novi Sad, 2014.
- [84] M. R. Garey, D. S. Johnson, "Computers and intractability: A Guide to the Theory of NP Completeness", Freeman, San-Francisco, 1979.
-

-
- [85] M. J. A. S. N. Sipser, "Introduction to the Theory of Computation," vol. 27, no. 1, pp. 27-29, 1996.
- [86] F. Glover, "Future paths for integer programming and links to artificial intelligence", *Computers & Operations Research*, vol. 13, no. 5, pp. 533-549, 1986.
- [87] S. Luke, "Essentials of Metaheuristics", Lulu, 2013.
- [88] S. Mirjalili, S. M. Mirjalili, A. Lewis, "Grey Wolf Optimizer", *Advances in Engineering Software*, vol. 69, pp. 46-61, 2014.
- [89] D. H. Wolpert, W. G. Macready, "No free lunch theorems for optimization", *IEEE Transactions on Evolutionary Computation*, vol. 1, no. 1, pp. 67-82, 1997.
- [90] G. Dhiman, V. Kumar, "Spotted hyena optimizer: A novel bio-inspired based metaheuristic technique for engineering applications", *Advances in Engineering Software*, vol. 114, pp. 48-70, 2017.
- [91] K. M. Passino, "Bacterial foraging optimization", *International Journal of Swarm Intelligence Research*, vol. 1, no. 1, pp. 1-16, 2010.
- [92] D. Dasgupta, *Artificial immune systems and their applications*. Springer Science & Business Media, 2012.
- [93] S. Mirjalili, J. S. Dong, A. Lewis, *Nature-inspired optimizers: theories, literature reviews and applications*. Springer, 2019.
- [94] S. Harifi, J. Mohammadzadeh, M. Khalilian, S. Ebrahimnejad, "Giza Pyramids Construction: an ancient-inspired metaheuristic algorithm for optimization", *Evolutionary Intelligence*, 2020.
- [95] X.-S. Yang, *Nature-inspired metaheuristic algorithms*. Luniver press, 2010.
- [96] K.-L. Du, M. J. T. Swamy, A. I. b. Nature, "Search and optimization by metaheuristics", 2016.
- [97] G. R. Miodragović, "Razvoj naprednih biološki inspirisanih algoritama za rešavanje optimizacionih problema primenjene mehanike," Doktorska disertacija, Univerzitet u Kragujevcu, Fakultet za Mašinstvo i Građevinarstvo, Kraljevo, 2015.
- [98] S. Saremi, S. Mirjalili, A. Lewis, "Grasshopper Optimisation Algorithm: Theory and application," *Advances in Engineering Software*, vol. 105, pp. 30-47, 2017.
- [99] G. Dhiman, V. Kumar, "Seagull optimization algorithm: Theory and its applications for large-scale industrial engineering problems," *Knowledge-Based Systems*, vol. 165, pp. 169-196, 2019.
- [100] S. H. Samareh Moosavi, V. Khatibi Bardsiri, "Satin bowerbird optimizer: A new optimization algorithm to optimize ANFIS for software development effort estimation", *Engineering Applications of Artificial Intelligence*, vol. 60, pp. 1-15, 2017.
- [101] S. Shadravan, H. R. Naji, V. K. Bardsiri, "The Sailfish Optimizer: A novel nature-inspired metaheuristic algorithm for solving constrained engineering optimization problems", *Engineering Applications of Artificial Intelligence*, vol. 80, pp. 20-34, 2019.
-

-
- [102] X.-S. Yang, S. Deb, "Cuckoo search via Lévy flights", in *2009 World congress on nature & biologically inspired computing (NaBIC)*, pp. 210-214, 2009.
- [103] S. Mirjalili, "Moth-flame optimization algorithm: A novel nature-inspired heuristic paradigm", *Knowledge-Based Systems*, vol. 89, pp. 228-249, 2015.
- [104] A. H. Gandomi, A. H. Alavi, "Krill herd: A new bio-inspired optimization algorithm" *Communications in Nonlinear Science and Numerical Simulation*, vol. 17, no. 12, pp. 4831-4845, 2012.
- [105] A. Askarzadeh, "A novel metaheuristic method for solving constrained engineering optimization problems: Crow search algorithm", *Computers & Structures*, vol. 169, pp. 1-12, 2016.
- [106] D. Karaboga, "An idea based on honey bee swarm for numerical optimization", Technical report-tr06, Erciyes university, Turkey, 2005.
- [107] M. Khishe, M. R. Mosavi, "Chimp optimization algorithm", *Expert Systems with Applications*, vol. 149, p. 113338, 2020.
- [108] R. Eberhart, J. Kennedy, "Particle swarm optimization," in *Proceedings of the IEEE international conference on neural networks*, vol. 4, pp. 1942-1948, 1995.
- [109] S. Mirjalili, A. H. Gandomi, S. Z. Mirjalili, S. Saremi, H. Faris, S. M. Mirjalili, "Salp Swarm Algorithm: A bio-inspired optimizer for engineering design problems", *Advances in Engineering Software*, vol. 114, pp. 163-191, 2017.
- [110] K. Zervoudakis, S. Tsafarakis, "A mayfly optimization algorithm", *Computers & Industrial Engineering*, vol. 145, p. 106559, 2020.
- [111] A. A. Heidari, S. Mirjalili, H. Faris, I. Aljarah, M. Mafarja, H. Chen, "Harris hawks optimization: Algorithm and applications", *Future Generation Computer Systems*, vol. 97, pp. 849-872, 2019.
- [112] G. Dhiman, V. Kumar, "Emperor penguin optimizer: A bio-inspired algorithm for engineering problems", *Knowledge-Based Systems*, vol. 159, pp. 20-50, 2018.
- [113] Flickr. (Pristupio 10.06.2021.). <https://www.flickr.com/>.
- [114] Wikimedia. (Pristupio 10.06.2021.). https://commons.wikimedia.org/wiki/Main_Page.
- [115] Pixabay. (Pristupio 10.06.2021.). <https://pixabay.com/>.
- [116] Freeimages. (Pristupio 10.06.2021.). <https://www.freeimages.com/>.
- [117] Pexels. (Pristupio 10.06.2021.). <https://www.pexels.com/>.
- [118] Unsplash. (Pristupio 10.06.2021.). <https://unsplash.com/>.
- [119] O. Olorunda, A. P. Engelbrecht, "Measuring exploration/exploitation in particle swarms using swarm diversity", in *2008 IEEE Congress on Evolutionary Computation (IEEE World Congress on Computational Intelligence)*, pp. 1128-1134, 2008.
-

-
- [120] L. Carvelli, G. Sebastiani, "Improving convergence of combinatorial optimization meta-heuristic algorithms", Ph. D. thesis, Sapienza Universita di Roma, Facolta di Scienze Matematiche, 2013.
- [121] T. Davidović, T. Jakšić Krüger, "Convergence Analysis of Swarm Intelligence Metaheuristic Methods", in *Optimization Problems and Their Applications*, Cham, 2018, pp. 251-266: Springer International Publishing.
- [122] S. Mirjalili, J. S. Dong, *Multi-objective optimization using artificial intelligence techniques*. Springer, 2020.
- [123] S. Sivanandam, S. Deepa, "Genetic algorithms", in *Introduction to genetic algorithms*: Springer, pp. 15-37, 2008.
- [124] J. H. Holland, "Adaptation in natural and artificial systems", University of Michigan press, vol. 1, no. 97, pp. 5, 1975.
- [125] D. E. Goldberg, *Genetic Algorithms in Search, Optimization and Machine Learning*. Addison-Wesley Longman Publishing Co., Inc., 1989.
- [126] S. Kirkpatrick, C. D. Gelatt Jr., M. P. Vecchi, "Optimization by Simulated Annealing", *Science*, vol. 220, no. 4598, pp. 671-680, 1983.
- [127] R. C. Eberhart, S. Yuhui, "Particle swarm optimization: developments, applications and resources", in *Proceedings of the 2001 Congress on Evolutionary Computation (IEEE Cat. No.01TH8546)*, vol. 1, pp. 81-86, 2001.
- [128] S. Nakrani, C. Tovey, "On Honey Bees and Dynamic Server Allocation in Internet Hosting Centers", *Adaptive Behavior*, vol. 12, no. 3-4, pp. 223-240, 2004.
- [129] D. Pham, A. Ghanbarzadeh, E. Koc, S. Otri, S. Rahim, M. J. C. U. Zaidi, "The bees algorithm", Technical Note, Manufacturing Engineering Centre, Cardiff, UK, 2005.
- [130] X.-S. Yang, "Flower pollination algorithm for global optimization", in *International conference on unconventional computing and natural computation*, 2012, pp. 240-249: Springer.
- [131] E. Rashedi, H. Nezamabadi-Pour, S. Saryazdi, "GSA: A Gravitational Search Algorithm", vol. 179, no. 13, pp. 2232-2248, 2009.
- [132] A. Kaveh, M. Khayatazad, "A new meta-heuristic method: Ray Optimization", *Computers & Structures*, vol. 112-113, pp. 283-294, 2012.
- [133] S. Mirjalili, S. M. Mirjalili, A. Hatamlou, "Multi-Verse Optimizer: a nature-inspired algorithm for global optimization", *Neural Computing and Applications*, vol. 27, no. 2, pp. 495-513, 2016.
- [134] S. Mirjalili, "SCA: A Sine Cosine Algorithm for solving optimization problems", *Knowledge-Based Systems*, vol. 96, pp. 120-133, 2016.
- [135] A. Askarzadeh, "Bird mating optimizer: An optimization algorithm inspired by bird mating strategies", *Communications in Nonlinear Science and Numerical Simulation*, vol. 19, no. 4, pp. 1213-1228, 2014.
-

-
- [136] S. Kaur, L. K. Awasthi, A. L. Sangal, G. Dhiman, "Tunicate Swarm Algorithm: A new bio-inspired based metaheuristic paradigm for global optimization", *Engineering Applications of Artificial Intelligence*, vol. 90, p. 103541, 2020.
- [137] G. Dhiman, A. Kaur, "STOA: A bio-inspired based optimization algorithm for industrial engineering problems", *Engineering Applications of Artificial Intelligence*, vol. 82, pp. 148-174, 2019.
- [138] F. Zhang, Y. F. Zhang, A. Y. C. Nee, "Using genetic algorithms in process planning for job shop machining", *IEEE Transactions on Evolutionary Computation*, vol. 1, no. 4, pp. 278-289, 1997.
- [139] W. D. Li, S. K. Ong, A. Y. C. Nee, "Hybrid genetic algorithm and simulated annealing approach for the optimization of process plans for prismatic parts", *International Journal of Production Research*, vol. 40, no. 8, pp. 1899-1922, 2002.
- [140] S. Li, Y. Liu, Y. Li, R. G. Landers, L. Tang, "Process planning optimization for parallel drilling of blind holes using a two phase genetic algorithm", *Journal of Intelligent Manufacturing*, vol. 24, no. 4, pp. 791-804, 2013.
- [141] J. Dou, X. Zhao, C. Su, "An Improved Genetic Algorithm for Optimization of Operation Sequencing", in *2018 IEEE International Conference on Mechatronics and Automation (ICMA)*, pp. 695-700, 2018.
- [142] A. Falih, A. Z. M. Shammari, "Hybrid constrained permutation algorithm and genetic algorithm for process planning problem", *Journal of Intelligent Manufacturing*, vol. 31, no. 5, pp. 1079-1099, 2020.
- [143] Q. Liu, X. Li, L. Gao, "Mathematical modeling and a hybrid evolutionary algorithm for process planning," *Journal of Intelligent Manufacturing*, Vol. 32, pp. 781–797, 2020.
- [144] Y. Luo, Y. Pan, C. Li, H. Tang, "A hybrid algorithm combining genetic algorithm and variable neighborhood search for process sequencing optimization of large-size problem", *International Journal of Computer Integrated Manufacturing*, vol. 33, no. 10-11, pp. 962-981, 2020.
- [145] T. Xiang, X. Liao, K.-w. Wong, "An improved particle swarm optimization algorithm combined with piecewise linear chaotic map", *Applied Mathematics and Computation*, vol. 190, no. 2, pp. 1637-1645, 2007.
- [146] B. Liu, L. Wang, Y.-H. Jin, F. Tang, D.-X. Huang, "Improved particle swarm optimization combined with chaos", *Chaos, Solitons & Fractals*, vol. 25, no. 5, pp. 1261-1271, 2005.
- [147] B. Alatas, E. Akin, "Chaotically encoded particle swarm optimization algorithm and its applications", *Chaos, Solitons & Fractals*, vol. 41, no. 2, pp. 939-950, 2009.
- [148] J. F. Wang, W. L. Kang, J. L. Zhao, K. Y. Chu, "A simulation approach to the process planning problem using a modified particle swarm optimization", *Advances in Production Engineering & Management*, vol. 11, no. 2, pp. 77-92, 2016.
-

-
- [149] D. Yang, Z. Liu, P. Yi, "Computational efficiency of accelerated particle swarm optimization combined with different chaotic maps for global optimization", *Neural Computing and Applications*, vol. 28, no. 1, pp. 1245-1264, 2017.
- [150] G. I. Sayed, A. E. Hassanien, A. T. Azar, "Feature selection via a novel chaotic crow search algorithm", *Neural Computing and Applications*, vol. 31, no. 1, pp. 171-188, 2019.
- [151] M. Kohli, S. Arora, "Chaotic grey wolf optimization algorithm for constrained optimization problems", *Journal of Computational Design and Engineering*, vol. 5, no. 4, pp. 458-472, 2018.
- [152] G. Kaur, S. Arora, "Chaotic whale optimization algorithm," *Journal of Computational Design and Engineering*, vol. 5, no. 3, pp. 275-284, 2018.
- [153] Y. F. Wang, Y. F. Zhang, J. Y. H. Fuh, "A hybrid particle swarm based method for process planning optimisation", *International Journal of Production Research*, vol. 50, no. 1, pp. 277-292, 2012.
- [154] Q. Hu, L. Qiao, G. Peng, "An ant colony approach to operation sequencing optimization in process planning", vol. 231, no. 3, pp. 470-489, 2017.
- [155] S. S. Singh, S. Deb, "An intelligent methodology for optimising machining operation sequence by ant system algorithm" *International Journal of Industrial and Systems Engineering*, vol. 16, no. 4, pp. 451-471, 2014.
- [156] W. D. Li, S. K. Ong, A. Y. C. Nee, "Optimization of process plans using a constraint-based tabu search approach", *International Journal of Production Research*, vol. 42, no. 10, pp. 1955-1985, 2004.
- [157] B. Gao, X. Hu, Z. Peng, Y. Song, "Application of intelligent water drop algorithm in process planning optimization", *The International Journal of Advanced Manufacturing Technology*, vol. 106, no. 11, pp. 5199-5211, 2020.
- [158] M. Milošević, M. Đurđev, D. Lukić, A. Antić, N. Ungureanu, "Intelligent Process Planning for Smart Factory and Smart Manufacturing", in *Proceedings of 5th International Conference on the Industry 4.0 Model for Advanced Manufacturing*, Cham, 2020, pp. 205-214: Springer International Publishing.
- [159] M. Djurdjev, R. Cep, D. Lukic, A. Antic, B. Popovic, M. Milosevic, "A Genetic Crow Search Algorithm for Optimization of Operation Sequencing in Process Planning", *Applied Sciences*, vol. 11, no. 5, pp. 1981, 2021.
- [160] National Geographic. (Pristupio 20.04.2021.).
<https://www.nationalgeographic.com/animals/article/crow-as-clever-as-great-apes-study-says>.
- [161] Pexels. (Pristupio 03.06.2021.).
<https://www.pexels.com/photo/black-crow-on-green-grass-7286344/>.
- [162] Pexels. (Pristupio 03.06.2021.).
<https://www.pexels.com/photo/black-crow-eating-food-on-street-6024995/>.
-

-
- [163] S. Shirke, R. Udayakumar, "Evaluation of Crow Search Algorithm (CSA) for Optimization in Discrete Applications", in *2019 3rd International Conference on Trends in Electronics and Informatics (ICOEI)*, pp. 584-589, 2019.
- [164] K.-W. Huang, A. S. Girsang, Z.-X. Wu, Y.-W. Chuang, "A Hybrid Crow Search Algorithm for Solving Permutation Flow Shop Scheduling Problems", vol. 9, no. 7, pp. 1353, 2019.
- [165] S. Laabadi, M. Naimi, H. E. Amri, B. Achchab, "A Binary Crow Search Algorithm for Solving Two-dimensional Bin Packing Problem with Fixed Orientation", *Procedia Computer Science*, vol. 167, pp. 809-818, 2020.
- [166] F. Mohammadi, H. Abdi, "A modified crow search algorithm (MCSA) for solving economic load dispatch problem", *Applied Soft Computing*, vol. 71, pp. 51-65, 2018.
- [167] N. Sivarami Reddy, V. D. Ramamurthy, K. Prahlada Rao, M. Padma Lalitha, "Integrated scheduling of machines, AGVs and tools in multi-machine FMS using crow search algorithm", *International Journal of Computer Integrated Manufacturing*, vol. 32, no. 11, pp. 1117-1133, 2019.
- [168] R. M. Rizk-Allah, A. E. Hassanien, S. Bhattacharyya, "Chaotic crow search algorithm for fractional optimization problems", *Applied Soft Computing*, vol. 71, pp. 1161-1175, 2018.
- [169] C. Muro, R. Escobedo, L. Spector, R. P. Coppinger, "Wolf-pack (*Canis lupus*) hunting strategies emerge from simple rules in computational simulations", *Behavioural Processes*, vol. 88, no. 3, pp. 192-197, 2011.
- [170] D. R. MacNulty, A. Tallian, D. R. Stahler, D. W. Smith, "Influence of Group Size on the Success of Wolves Hunting Bison", *PLOS ONE*, vol. 9, no. 11, pp. 1-8, 2014.
- [171] S. Mirjalili. *Google Scholar*. (Pristupio 11.06.2021.):
<https://scholar.google.com/citations?user=TJHmrREAAA&hl=en>
- [172] C. Khan. (Pristupio 11.06.2021.):
https://commons.wikimedia.org/wiki/File:Humpback_whale_bubble_net_feeding_Christin_Khan_NOAA.jpg
- [173] G. Candan, H. R. Yazgan, "Genetic algorithm parameter optimisation using Taguchi method for a flexible manufacturing system scheduling problem" *International Journal of Production Research*, vol. 53, no. 3, pp. 897-915, 2015.
- [174] M. Sekulic, P. Kovac, M. Gostimirovic, D. Kramar, "Optimization of high-pressure jet assisted turning process by Taguchi method", *Advances in Production Engineering & Management*, vol. 8, no. 1, pp. 5-12, 2013.
- [175] M. Sekulić, M. Hadžistević, Z. Jurković, P. Kovač, M. Gostimirović, "Application of Taguchi Method in the Optimization of Face Milling Parameters", presented at the 34th International Conference on Production Engineering, 29. - 30. September 2011, Niš, Serbia, 2011.
-

- [176] D. Lukic, R. Cep, J. Vukman, A. Antic, M. Djurdjev, M. Milosevic, "Multi-Criteria Selection of the Optimal Parameters for High-Speed Machining of Aluminum Alloy Al7075 Thin-Walled Parts", *Metals*, vol. 10, no. 12, pp. 1-22, 2020.
- [177] D. Rodić, "Optimizacija procesa elektroerozivne obrade savremenih inženjerskih materijala," Doktorska disertacija, Fakultet tehničkih nauka, Univerzitet u Novom Sadu, Novi Sad, 2019.

9. PRILOZI

Prilog 1. Funkcija cilja – *fc* (Izvorni kod u Matlab-u, prvi deo)

```
function output = gcsa_funkcija_troskovi(Vrane, ZahvatNo, brojJedinki)

MCI = [10 40 100 60];           % Troškovi rada mašina
TCI = [7 5 3 8 7 10 15 30 15 20]; % Troškovi korišćenja alata

MCCI = 160;                     % Troškovi promena mašina
TCCI = 20;                      % Troškovi promena alata
SCI = 100;                      % Troškovi promena stezanja

% Inicijalizacija
output = Vrane;

for i = 1 : brojJedinki

    %% Ukupni troškovi rada (TMC)
    TMC = zeros (1, ZahvatNo);
    for j = 1 : ZahvatNo
        TMC(j) = MCI(Vrane(i).VektoriPolozaja.Masine(j));
    end
    output(i).TMC = sum(TMC);

    %% Ukupni troškovi promena (TMCC)
    % Broj promena mašina (NMC)
    NMC = zeros(1, ZahvatNo - 1);
    for j = 1 : ZahvatNo - 1
        if Vrane(i).VektoriPolozaja.Masine(j) ==
            Vrane(i).VektoriPolozaja.Masine(j+1)
            C1 = 0;
        else
            C1 = 1;
        end

        NMC(j) = C1;
    end
    output(i).NMC = sum(NMC);
    output(i).TMCC = sum(NMC) * MCCI;

    %% Ukupni troškovi korišćenja alata (TTC)
    TTC = zeros (1, ZahvatNo);
    for j = 1 : ZahvatNo
        TTC(j) = TCI(Vrane(i).VektoriPolozaja.Alati(j));
    end
    output(i).TTC = sum(TTC);

    %% Ukupni troskovi promena alata (TTCC)
    % Broj promena alata (NTC)
    NTC = zeros(1, ZahvatNo - 1);
    for j = 1 : ZahvatNo - 1
        if Vrane(i).VektoriPolozaja.Masine(j) ==
            Vrane(i).VektoriPolozaja.Masine(j+1)
            C1_masine = 0;
        else
            C1_masine = 1;
        end
    end
end
```

Prilog 1. Funkcija cilja – f_c (Izvorni kod u Matlab-u, drugi deo)

```

    if Vrane(i).VektoriPolozaja.Alati(j) == Vrane(i).VektoriPolozaja.Alati(j+1)
        C1_alati = 0;
    else
        C1_alati = 1;
    end

    if C1_masine == 0 && C1_alati == 0
        C2 = 0;
    else
        C2 = 1;
    end

    NTC(j) = C2;
end
output(i).NTC = sum(NTC);
output(i).TTCC = sum(NTC)*TCCI;

%% Ukupni troškovi promene stezanja (TSC)
% Broj promena stezanja (NSC)
NSC = zeros(1, ZahvatNo - 1);
for j = 1 : ZahvatNo - 1
    if Vrane(i).VektoriPolozaja.Masine(j) ==
        Vrane(i).VektoriPolozaja.Masine(j+1)
        C1_masine = 0;
    else
        C1_masine = 1;
    end

    if Vrane(i).VektoriPolozaja.SPA(j) == Vrane(i).VektoriPolozaja.SPA(j+1)
        C1_spa = 0;
    else
        C1_spa = 1;
    end

    if C1_masine == 0 && C1_spa == 0
        C2 = 0;
    else
        C2 = 1;
    end

    NSC(j) = C2;
end
output(i).NSC = sum(NSC) + 1;
output(i).SC = (sum(NSC) + 1) *SCI;

%% Ukupni troskovi obrade (TWMC)
output(i).TWMC = w1*output(i).TMC + w2*output(i).TMCC + w3*output(i).TTC + ...
    w4*output(i).TTCC + w5*output(i).SC;

% Funkcija cilja
output(i).Fitness = 1 / output(i).TWMC;

end

```

Prilog 2. Mehanizam najbliže mašine (Izvorni kod u Matlab-u)

```
function output = najblizi_resurs(kandidat, ZahvatNo, Masine, Alati, PPA)

for i = 1 : ZahvatNo

    Zahvat = kandidat.VektoriPolozaja.Zahvati(i);

%% Mehanizam najblize masine
% prvi korak je provera da li je selektovana masina najbolji izbor; ukoliko je
% izbor los, zameniti postojeću masinu najblizom alternativnom masinom koja je
% raspoloživa iz skupa "Masine" prema izvodljivom redosledu alternativa

    postojeciKandidat = kandidat.VektoriPolozaja.Masine(i);
    diff = sort(abs(Masine{Zahvat}- postojeciKandidat));
    if all(postojeciKandidat > Masine{Zahvat})== true
        for j = 1: size(diff,2)
            if ~isempty(find(Masine{Zahvat} == abs(current-diff(j)), 1)) == 1
                kandidat.VektoriPolozaja.Masine(i) = abs(postojeciKandidat -diff(j));
                break;
            end
        end
    elseif all(current < Masine{Zahvat})== true
        for j = 1: size(diff,2)
            if ~isempty(find(Masine{Zahvat} == abs(postojeciKandidat +diff(j)), 1))
                == 1
                kandidat.VektoriPolozaja.Masine (i) = abs(postojeciKandidat
+diff(j));
                break;
            end
        end
    else
        for j = 1: size(diff,2)
            if ~isempty(find(Masine{Zahvat} == abs(postojeciKandidat -diff(j)), 1))
                == 1
                kandidat.VektoriPolozaja.Masine (i) = abs(postojeciKandidat -
diff(j));
                break;
            elseif ~isempty(find(Masine{Zahvat} == abs(postojeciKandidat +diff(j)),
1)) == 1
                kandidat.VektoriPolozaja.Masine (i) = abs(postojeciKandidat
+diff(j));
                break;
            end
        end
    end
end
```

Prilog 3. Mehanizam popravke rešenja (Izvorni kod u Matlab-u)

```

function output = popravka_resenja(Vrane, problem, brojJedinki)

Matrica = problem.Matrica;
output = Crows;

for i = 1: brojJedinki
    pt = size(Vrane(i).Polozaj,2);
    ptOper = Vrane(i).Polozaj (1,pt);
    for j = 1 : pt-1
        if sum(Matrica(ptOper,:), 'omitnan') == 0
            Matrica(ptOper,:) = nan;
            Matrica(:,ptOper) = nan;
        else
            p = pt-1;
            pOper = Vrane(i).Polozaj(1,p);
            if sum(Matrica(pOper,:), 'omitnan') == 0
                Vrane(i).Polozaj(:, [pt p]) = Vrane(i).Polozaj(:, [p pt]);
                Matrica(pOper,:) = nan;
                Matrica(:,pOper) = nan;
            else
                p=p-1;
                pOper = Vrane(i).Polozaj(1,p);
                while p>0
                    if sum(Matrica(pOper,:), 'omitnan') == 0
                        Vrane(i).Polozaj(:, [pt p]) = Vrane(i).Polozaj(:, [p pt]);
                        Matrica(pOper,:) = nan;
                        Matrica(:,pOper) = nan;
                        break;
                    else
                        p = p-1;
                        pOper = Vrane(i).Polozaj(1,p);
                        continue;
                    end
                end
            end
        end
    end
    pt = pt - 1;
    ptOper = Vrane(i).Polozaj(1,pt);
end
Matrica = problem.Matrica;
Vrane(i).Polozaj = Vrane(i).Polozaj;
Vrane(i).VektoriPolozaja.Zahvati = Vrane(i).Polozaj(1,:);
Vrane(i).VektoriPolozaja.Masine = Vrane(i).Polozaj(2,:);
Vrane(i).VektoriPolozaja.Alati = Vrane(i).Polozaj(3,:);
Vrane(i).VektoriPolozaja.SPA = Vrane(i).Polozaj(4,:);
output = Vrane;
end

```

Prilog 4. 3SX operator ukrštanja (Izvorni kod u Matlab-u)

```
function [potomak1, potomak2] = 3SX_ukrstanje(roditelj1, roditelj2, Pc, ZahvatNo)

%3SX strategija ukrštanja jedinki

% slučajno selektovati jednu poziciju ukrštanja
pozicija_ukrstanja = randi([1 ZahvatNo-1]);

potomak1_levo = roditelj1(:,1:pozicija_ukrstanja);
potomak1_ostatak(1,:) = roditelj1(1, pozicija_ukrstanja+1:end);
for i = 1: size(potomak1_ostatak,2)
    potomak1_ostatak(2:4,i) = roditelj2(2:end, roditelj2(1,:) == ...
        roditelj1(1,pozicija_ukrstanja+i));
end

potomak2_levo = roditelj2(:,1:pozicija_ukrstanja);
potomak2_ostatak(1,:) = roditelj2(1, pozicija_ukrstanja+1:end);
for i = 1: size(potomak2_ostatak,2)
    potomak2_ostatak(2:4,i) = roditelj1(2:end, roditelj1(1,:) == ...
        roditelj2(1,pozicija_ukrstanja+i));
end

potomak1 = [potomak1_levo potomak1_ostatak];
potomak2 = [potomak2_levo potomak2_ostatak];

R1 = rand();
if R1 <= Pc
    potomak1 = potomak1;
else
    potomak1 = roditelj1;
end

R2 = rand();
if R2 <= Pc
    potomak2 = potomak2;
else
    potomak2 = roditelj2;
end
```

Prilog 5. Mutacija resursa (vektora mašina) (Izvorni kod u Matlab-u)

```
function [mutant] = mutacija_resursa(mutant, brojJedinki, Mašine, Alati, SPA)

%% Mutacija vektora mašine alatke
% selektovati slucajan broj koji ce predstavljati indeks masine alatke

mutOp = randi([1,brojZahvata]);
% selektovati gen jednog vektora mašina
zahvat = mutant.VektoriPolozaja.Zahvati(mutOp);
% zameniti postojeću masinu nekom drugom ("slučajno selektovana alternativa")
% pronaci druge gene u vektoru masina koji imaju istu postojeću masinu i isto tako
% istu "slučajno selektovanu alternativu" i izvršiti mutaciju
if numel(Masine{zahvat}) > 1 % proveriti da li selektovani zahvat ima vise
    % alternativa
    postojeciGen = mutant.VektoriPolozaja.Masine(mutOp); % dodeliti postojeću
    % masinu
    alternative = setdiff(Masine{zahvat}, postojeciGen); % pronaci sve moguće
    % alternative masina
    if numel(alternative) > 1
        altGen = randsample(alternative, 1); % dodeliti "slučajno selektovanu
        % alternativu"
        istiGen = setdiff(find(mutant.VektoriPolozaja.Masine == postojeciGen),
mutOp); % pronaci indekse ostalih gena koji imaju istu postojeću masinu
        mutant.VektoriPolozaja.Masine(mutOp) = altGen;
        for j = 1:numel(istiGen)
            % ako zahvati sa selektovanim alatima koji su isti kao i postojeci
            % imaju isti alternativni alat, izvršiti mutaciju
            if any(find(Masine{mutant.VektoriPolozaja.Zahvati(istiGen(j))}==
altGen)) == true
                mutant.VektoriPolozaja.Masine(istiGen(j)) = altGen;
            end
        end
    else % ako postoji samo jedna alternativna masina
        altGen = alternative;
        istiGen = setdiff(find(mutant.VektoriPolozaja.Masine == postojeciGen),
mutOp); % pronaci indekse ostalih gena koji imaju istu postojeću masinu
        mutant.VektoriPolozaja.Masine(mutOp) = altGen;
        for j = 1:numel(istiGen)
            % ako zahvati sa selektovanim alatima koji su isti kao i postojeci
            % imaju isti alternativni alat, izvršiti mutaciju
            if any(find(Masine{mutant.VektoriPolozaja.Zahvati(istiGen(j))}==
altGen)) == true
                mutant.VektoriPolozaja.Masine(istiGen(j)) = altGen;
            end
        end
    end
end
end
end
```


Овај Образац чини саставни део докторске дисертације, односно докторског уметничког пројекта који се брани на Универзитету у Новом Саду. Попуњен Образац укоричити иза текста докторске дисертације, односно докторског уметничког пројекта.

План третмана података

Назив пројекта/истраживања
<i>Примена савремених метахеуристичких алгоритама у оптимизацији технолошких процеса обраде производа</i>
Назив институције/институција у оквиру којих се спроводи истраживање
а) <i>Универзитет у Новом Саду, Факултет техничких наука, Департман за производно машинство</i> б) <i>Универзитет у Новом Саду, Технички факултет „Михајло Пупин“ Зрењанин</i>
Назив програма у оквиру ког се реализује истраживање
<i>Програм докторских студија из области машинства</i>
1. Опис података
1.1 Врста студије <i>Укратко описати тип студије у оквиру које се подаци прикупљају</i> <i>Докторска дисертација</i>
1.2 Врсте података а) квантитативни б) квалитативни
1.3. Начин прикупљања података а) анкете, упитници, тестови б) клиничке процене, медицински записи, електронски здравствени записи в) генотипови: навести врсту _____ г) административни подаци: навести врсту _____ д) узорци ткива: навести врсту _____ ђ) снимци, фотографије: навести врсту _____

- е) текст, навести врсту Литературни извори
- ж) мапа, навести врсту _____
- з) остало: описати Експериментална испитивања

1.3 Формат података, употребљене скале, количина података

1.3.1 Употребљени софтвер и формат датотеке:

- а) Ехсел фајл, датотека: .xlsx
- б) SPSS фајл, датотека _____
- с) PDF фајл, датотека: .pdf
- д) Текст фајл, датотека: .docx
- е) JPG фајл, датотека: .jpg
- ф) Остало, датотека: .m, .mat

1.3.2. Број записа (код квантитативних података)

- а) број варијабли: Велики број
- б) број мерења (испитаника, процена, снимака и сл.): Велики број

1.3.3. Поновљена мерења

- а) да
- б) не

Уколико је одговор да, одговорити на следећа питања:

- а) временски размак између поновљених мера је 5 минута
- б) варијабле које се више пута мере односе се на: Израчунавање трошкова и времена обраде призматичних и ротационих делова
- в) нове верзије фајлова који садрже поновљена мерења су именоване као _____

Напомене: _____

Да ли формати и софтвер омогућавају дељење и дугорочну валидност података?

- а) Да
- б) Не

Ако је одговор не, образложити _____

2. Прикупљање података

2.1 Методологија за прикупљање/генерисање података

2.1.1. У оквиру ког истраживачког нацрта су подаци прикупљени?

а) експеримент, навести тип:

Експериментална студија примене метахеуристичких алгоритама

б) корелационо истраживање, навести тип _____

ц) анализа текста, навести тип:

Прикупљање података анализом доступне литературе

д) остало, навести шта _____

2.1.2 Навести врсте мерних инструмената или стандарде података специфичних за одређену научну дисциплину (ако постоје).

2.2 Квалитет података и стандарди

2.2.1. Третман недостајућих података

а) Да ли матрица садржи недостајуће податке? Да **Не**

Ако је одговор да, одговорити на следећа питања:

а) Колики је број недостајућих података? _____

б) Да ли се кориснику матрице препоручује замена недостајућих података? Да **Не**

в) Ако је одговор да, навести сугестије за третман замене недостајућих података

2.2.2. На који начин је контролисан квалитет података? Описати

Квалитет података је контролисан селекцијом адекватних улазних параметара савремених метахеуристичких алгоритама

2.2.3. На који начин је извршена контрола уноса података у матрицу?

Контрола уноса података у матрицу је изведена поређењем добијених података са подацима из литературних извора

3. Третман података и пратећа документација

3.1. Третман и чување података

3.1.1. Подаци ће бити депоновани у Репозиторијуму докторских дисертација на Универзитету у Новом Саду.

3.1.2. URL адреса: <https://cris.uns.ac.rs/searchDissertations.jsf>

3.1.3. DOI _____

3.1.4. Да ли ће подаци бити у отвореном приступу?

а) Да

б) Да, али после ембарга који ће трајати до _____

в) Не

Ако је одговор не, навести разлог _____

3.1.5. Подаци неће бити депоновани у репозиторијум, али ће бити чувани.

Образложење

3.2 Метаподаци и документација података

3.2.1. Који стандард за метаподатке ће бити примењен? _____

3.2.1. Навести метаподатке на основу којих су подаци депоновани у репозиторијум.

Ако је потребно, навести методе које се користе за преузимање података, аналитичке и процедуралне информације, њихово кодирање, детаљне описе варијабли, записа итд.

3.3 Стратегија и стандарди за чување података

3.3.1. До ког периода ће подаци бити чувани у репозиторијуму? _____

3.3.2. Да ли ће подаци бити депоновани под шифром? **Да** **Не**

3.3.3. Да ли ће шифра бити доступна одређеном кругу истраживача? **Да** **Не**

3.3.4. Да ли се подаци морају уклонити из отвореног приступа после извесног времена?

Да **Не**

Образложити

4. Безбедност података и заштита поверљивих информација

Овај одељак МОРА бити попуњен ако ваши подаци укључују личне податке који се односе на учеснике у истраживању. За друга истраживања треба такође размотрити заштиту и сигурност података.

4.1 Формални стандарди за сигурност информација/података

Истраживачи који спроводе испитивања с људима морају да се придржавају Закона о заштити података о личности (https://www.paragraf.rs/propisi/zakon_o_zastiti_podataka_o_licnosti.html) и одговарајућег институционалног кодекса о академском интегритету.

4.1.2. Да ли је истраживање одобрено од стране етичке комисије? **Да** **Не**

Ако је одговор Да, навести датум и назив етичке комисије која је одобрила истраживање

4.1.2. Да ли подаци укључују личне податке учесника у истраживању? **Да** **Не**

Ако је одговор да, наведите на који начин сте осигурали поверљивост и сигурност информација везаних за испитанике:

- а) Подаци нису у отвореном приступу
- б) Подаци су анонимизирани
- ц) Остало, навести шта

5. Доступност података

5.1. Подаци ће бити

а) **јавно доступни**

б) доступни само уском кругу истраживача у одређеној научној области

ц) затворени

Ако су подаци доступни само уском кругу истраживача, навести под којим условима могу да их користе:

Ако су подаци доступни само уском кругу истраживача, навести на који начин могу приступити подацима:

5.4. Навести лиценцу под којом ће прикупљени подаци бити архивирани.

Ауторство – некомерцијално – без прераде

6. Улоге и одговорност

6.1. Навести име и презиме и мејл адресу власника (аутора) података

Мића Ђурђевић, mica.djurdjevic@tfzr.rs

6.2. Навести име и презиме и мејл адресу особе која одржава матрицу с подацима

Мића Ђурђевић, mica.djurdjevic@tfzr.rs

6.3. Навести име и презиме и мејл адресу особе која омогућаје приступ подацима другим истраживачима

Мића Ђурђевић, mica.djurdjevic@tfzr.rs
