

Optimizacija ciklusa izrade na fleksibilnom proizvodnom sustavu primjenom genetskih algoritama

Perinić, Mladen

Doctoral thesis / Disertacija

2004

Degree Grantor / Ustanova koja je dodijelila akademski / stručni stupanj: **University of Rijeka, Faculty of Engineering / Sveučilište u Rijeci, Tehnički fakultet**

Permanent link / Trajna poveznica: <https://um.nsk.hr/um:nbn:hr:188:371267>

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

Download date / Datum preuzimanja: **2024-07-13**



Repository / Repozitorij:

[Repository of the University of Rijeka Library - SVKRI Repository](#)



SVEUČILIŠTE U RIJECI
TEHNIČKI FAKULTET

**OPTIMIZACIJA CIKLUSA IZRADE NA
FLEKSIBILNOM PROIZVODNOM SUSTAVU
PRIMJENOM GENETSKIH ALGORITAMA**

Doktorska disertacija

Mladen Perinić

Mentor: Red. prof. dr. sc. Juraj Ljubetić, dipl. ing.

Rijeka, 2004.

Sveučilište u Rijeci
TEHNIČKI FAKULTET
- Fakultetsko vijeće –
Klasa: 602/04-01-01/4
Ur.br.: 2170-57-43-01-03
Rijeka, 23. veljače 2001.

Fakultetsko vijeće Tehničkog fakulteta Sveučilišta u Rijeci, na 4. sjednici (u akad. god. 2000/2001.) održanoj 22. veljače 2001. donijelo je sljedeću

O D L U K U

Prihvaća se izvješće Stručnog povjerenstva u sastavu: red.prof.dr.sc. Juraj Ljubetić, red.prof.dr.sc. Jože Balić – Fakulteta za strojništvo Univerze v Mariboru i izv.prof.dr.sc. Tonči Mikca te se utvrđuje da **mr.sc. MLADEN PERINIĆ, dipl.ing.** ispunjava uvjete iz članka 51. Zakona o visokim učilištima te da može pristupiti izradi doktorske disertacije izvan doktorskog studija.

Naziv teme doktorske disertacije glasi: **“Optimizacija ciklusa obrade na fleksibilnim proizvodnim sustavima primjenom genetskih algoritama”.**

Pristupniku se za mentora imenuje red.prof.dr.sc. Juraj Ljubetić.

~~D e k l a r a c i j a~~

Red.prof.dr.sc. Bernard Franković, dipl.ing.

- Dostaviti:

- 1.) Mr.sc. Mladen Perinić
- 2.) Mentor
- 3.) Povjerenstvo za poslijediplomske studije i doktorate znanosti
- 4.) Evidencija studija
- 5.) Opća i kadrovska služba
- 6.) Pismohrana F.V.

SAŽETAK

Cilj je rada istražiti problem simultane proizvodnje dijelova na FPS-u s različitim kombinacijama tipova dijelova i količina, koja će osigurati maksimalno iskorištenje proizvodnog sustava. Pretpostavka za dobro iskorištenje fleksibilnog proizvodnog sustava je formiranje radnog naloga s takvom strukturom tipova dijelova, koja će omogućiti odvijanje proizvodnje uz najmanje vremensko opterećenje proizvodnog sustava kao cjeline. Struktura radnog naloga po pitanju tipova dijelova i količina diktirana je tržišnim zahtjevima, tj unaprijed je zadana. Budući da struktura pojedinog radnog naloga nije u potpunosti usklađena s eksploatacijskim karakteristikama FPS-a koje proizlaze prvenstveno iz značajki strukture jezgre sustava, suočavamo se s problemom, kako u takvim uvjetima upravljati realiziranjem radnog naloga i pri tom postići glavni cilj: što manje vremensko zauzeće sustava odnosno što kraći ciklus izrade.

U radu je prikazana metoda koja se temelji na dvije faze rješavanja problema upravljanja realiziranja radnog naloga. U prvoj fazi određen je izbor optimalne kombinacije planova procesa koja pri simultanoj proizvodnji različitih dijelova zadanih količina osigurava najmanje vremensko opterećenje proizvodnog sustava u cjelini. U drugoj fazi optimiran je redoslijed ulaska pojedinih dijelova u proces izrade i redoslijed izvođenja pojedinih operacija.

Optimizacijski problem u obje faze upravljanja riješen je rabljenjem genetskog algoritma. Razvijen je računalni program za proračun i optimizaciju redoslijednog odvijanja procesa na FPS-u.

SUMMARY

The aim of dissertation is to solve the problem of simultaneous production on the flexible manufacturing system with different combination of product types and quantities that will give maximal utilization of production system. The presumption for good utilization of FMS is in forming of working order with such product type structure that will make possible of production processing with minimal time load of complete production system. Working order structure from the point of product types and quantities is dictated by market demands that are known earlier. Because the structure of particular working order is not harmonized with the exploitation characteristics of FMS, we are faced with problem how to realize working order in such conditions as well as how to achieve main goal: shorter machining cycle with less time occupation of production system.

The method based on two phases for solving problem of control working order realization is presented in the dissertation. In the first phase the selection of optimal combination of process plans which gives minimal time load of production system through simultaneous production of different products and their quantities is given. In the second phase the order of part production and the order of particular operations processing is optimized.

The optimization problem in both phases of control is solved by application of genetic algorithm approach. The software for computing and optimizing of processing order on FMS is developed.

PREDGOVOR

Cilj je ove disertacije rješenje problema optimizacije ciklusa izrade kao pokazatelja vremenskog zauzeća proizvodnog sustava pri simultanoj proizvodnji dijelova prema određenom radnom nalogu što je temeljni preduvjet za gospodarstvenu isplativost velikih ulaganja vezanih za izgradnju i korištenje FPS-a. Ostvarenje cilja rada provedeno je razvojem metode u obje faze upravljanja FPS-a i primjenom genetskog algoritma kao najopćenitije suvremene metode za rješavanje optimizacijskih problema. Konačan rezultat nisu samo riješeni pojedinačni problemi optimalnog upravljanja FPS-a izloženi u ovoj disertaciji, već i računalni program kojim je moguće riješiti bilo koji novi problem proračuna i optimizacije redosljednog odvijanja procesa na FPS-u.

Koristim priliku da se zahvalim svima koji su mi pomogli u realizaciji ovog rada.

Posebno se zahvaljujem mentoru red. prof. dr. sc. J. Ljubetiću na nizu sugestija u teoretskim razmatranjima problema te nesebičnoj pomoći i podršci koju mi je pružio pri izradi i dovršenju rada. Isto tako, zahvaljujem red. prof. dr. sc. T. Mikcu za niz konstruktivnih primjedbi, korisne savjete i podršku tijekom izrade rada.

Također se zahvaljujem izv. prof. dr. sc. S. Vuković na pomoći pri odabiru i nabavci softverskih alata, kolegi O. Bukovcu na pomoći pri raalizaciji računalnog programa za rješavanje problema optimalnog upravljanja FPS-a, te osoblju knjižnice Tehničkog fakulteta na pomoći u nabavci literature.

Autor

SADRŽAJ

1. UVOD	1
2. FLEKSIBILNI PROIZVODNI SUSTAVI	4
2.1. FLEKSIBILNE PROIZVODNE PROSTORNE STRUKTURE	5
2.1.1. Fleksibilna proizvodna ćelija	7
2.1.1.1. Prednost FPC	8
2.1.1.2. Nedostatci FPC	8
2.1.2. Fleksibilni proizvodni sustavi	8
2.1.2.1. Prednosti FPS-a	9
2.1.2.2. Nedostatci FPS-a	9
2.1.3. Fleksibilne transfer linije	10
2.2. PODSUSTAVI FPS-a	11
2.2.1. Tehnički sustav	12
2.2.1.1. Procesni sustav	13
2.2.1.2. Sustav opskrbe radnim komadima	14
2.2.1.3. Sustav opskrbe alatima	17
2.2.2. Sustav ljudi poslužioca	18
2.2.3. Informacijski sustav	18
2.3. FLEKSIBILNOST KAO KARAKTERISTIKA FPS- a	20
2.3.1. Fleksibilnost elemenata	21
2.3.1.1. Fleksibilnost stroja	21
2.3.1.2. Fleksibilnost toka materijala	21
2.3.1.3. Operacijska fleksibilnost	22
2.3.2. Fleksibilnost sustava	22
2.3.2.1. Fleksibilnost na pregradnju	22
2.3.2.2. Proizvodna fleksibilnost	22
2.3.2.3. Procesna fleksibilnost	23
2.3.2.4. Fleksibilnost toka	23
2.3.2.5. Količinska fleksibilnost	23
2.4. TENDENCIJE U RAZVOJU FPS-a	24

3.	UPRAVLJANJE RADNIM NALOZIMA	30
3.1.	CIKLUS IZRADE	32
3.2.	ALTERNATIVNI PLANOWI PROCESA.....	34
3.2.1.	Model izbora planova procesa.....	34
3.2.2.	Postavljanje funkcije cilja.....	36
3.2.3.	Postupak izbora planova procesa.....	37
3.3.	REDOSLIJED ODVIJANJA RADNIH NALOGA	39
3.3.1.	Klasični TRN model.....	40
3.3.1.1.	IP model.....	42
3.3.1.2.	LP model.....	44
3.3.1.3.	Grafički model.....	44
3.3.2.	Konvencionalna heuristika.....	46
3.3.2.1.	Heuristika prioriteta pri raspodjeli.....	46
3.3.2.2.	Heuristika slučajnog odabira pri raspodjeli.....	48
3.3.2.3.	Heuristika premještaja uskog grla.....	49
4.	RJEŠAVANJE PROBLEMA UPRAVLJANJA FPS-a PRIMJENOM GENETSKIH ALGORITAMA.....	51
4.1.	STRUKTURA GENETSKIH ALGORITAMA.....	53
4.1.1.	Predstavljanje	54
4.1.2.	Prilagođenost jedinke	54
4.1.3.	Odabir roditelja.....	56
4.1.4.	Operatori genetskih algoritama.....	58
4.1.4.1.	Križanje.....	58
4.1.4.2.	Mutacija.....	58
4.1.4.3.	Napredni operatori genetskih algoritama	
4.2.	GENETSKI ALGORITMI U OPTIMALNOM IZBORU PLANOVA PROCESA.....	60
4.2.1.	Genotip i fenotip.....	60
4.2.2.	Križanje	60
4.2.3.	Mutacija.....	61
4.2.4.	Evaluacija i selekcija.....	62
4.3.	GENETSKI ALGORITMI ZA KLASIČNI PROBLEM TRN-a	63
4.3.1.	Predstavljanje	63
4.3.1.1.	Predstavljanje na bazi operacija.....	64
4.3.1.2.	Modificirano predstavljanje na bazi operacija.....	66
4.3.2.	Križanje.....	67
4.3.2.1.	Pristup Cheng-a, Gen-a i Tsujimure.....	67
4.3.3.	Mutacija.....	69
4.3.3.1.	Mutacija bazirana na traženju susjedstva.....	69

5.	OPTIMIZACIJA REDOSLIJEDNOG ODVIJANJA PROCESA NA FPS-u.....	70
5.1.	MODEL TRN-a NA FPS-u.....	70
5.2.	GENETSKI ALGORITMI ZA PROBLEM TRN-a NA FPS-u.....	71
5.2.1.	Kodiranje i dekodiranje kromosoma.....	71
5.3.	RAČUNALNI PROGRAM GJOB.....	81
5.3.1.	Opis potprograma.....	81
5.3.2.	Primjeri.....	86
5.3.2.1.	MT06 problem.....	86
5.3.2.2.	MT10 problem.....	88
5.3.2.3.	MT20 problem.....	91
5.4.	USPOREDBA REZULTATA I POTVRDA KODA RAČUNALNOG PROGRAMA.....	94
6.	PRAKTIČNI PRIKAZ METODE NA KONKRETNOM ZADATKU	97
6.1.	PRIMJER IZBORA PLANOVA PROCESA.....	97
6.1.1.	Usporedba s rezultatima tabu tehnike.....	98
6.2.	PRIMJER TERMINIRANJA DIJELOVA NA FPS-u.....	99
6.2.1.	Slučaj 1.....	99
6.2.2.	Slučaj 2.....	105
6.2.3.	Slučaj 3.....	112
6.2.4.	Slučaj 4.....	120
7.	ZAKLJUČAK	128
	LITERATURA	130
	POPIS OZNAKA I SKRAĆENICA.....	136
	POPIS SLIKA.....	138
	POPIS TABLICA.....	141
	ŽIVOTOPIS	142

1. UVOD

Razvoj tržišnih zahtjeva ide u smislu smanjenja količina istovjetnih proizvoda u skladu s tendencijom smanjenja životnog vijeka i povećanja broja varijanti proizvoda. Zahtjevnost tržišta ogleda se i u traženju visoke stabilnosti održavanja rokova isporuke te u njihovom izuzetnom vremenskom skraćanju. Uz to, često je potrebno udovoljiti i specifičnim zahtjevima kupaca uz zadržavanje povoljne cijene proizvoda.

Ispunjavanje tržišnih zahtjeva uz prisutnu težnju za smanjenjem troškova proizvodnje i rad s minimalnim zalihama uvjetuje potrebu da se u proizvodnju lansiraju sve manje količine izradaka (jedinične serije), kako bi se osigurao što jednakomjerniji dotok pojedinih vrsta dijelova na montažu.

Udovoljavanje ovim zahtjevima moguće je ostvariti na proizvodnom sustavu, koji u skladu s očekivanim rasponom navedenih zahtjeva treba imati traženi nivo fleksibilnosti na:

- promjenu geometrije izradaka
- promjenu redoslijeda obrade
- promjenu veličine jediničnih serija
- promjenu količina

Osnovu tako koncipiranih proizvodnih sustava osigurava razvoj fleksibilne automatizacije i na njoj temeljenih proizvodnih struktura s numerički upravljanim alatnim strojevima i obradnim centrima. Fleksibilni proizvodni sustavi (FPS) navedenih značajki pokrivaju upravo područje maloserijske i srednjeserijske proizvodnje, a gospodarski su isplativi čak i kod pojedinačne proizvodnje uz osiguranje visoke kvalitete proizvoda.

Zahvaljujući svojim eksploatacijskim značajkama fleksibilni proizvodni sustavi postaju sve prisutnije proizvodne strukture. Sposobnost prihvata većeg broja različitih izradaka i visoka spremnost za isporuku i pri manjim količinama proizvoda osnovne su prednosti FPS-a, koje osiguravaju široko područje njihove primjene. Prednosti se iskazuju postizanjem visoke proizvodnosti pri obradi stalnog toka različitih izradaka uz kratak ciklus proizvodnje. Za postizanje takvih učinaka presudno je smišljeno strukturiranje i oblikovanje FPS-a u skladu s postavljenim radnim zadatkom, ali isto tako i odgovarajuće planiranje i upravljanje odvijanja proizvodnje. Temelj rješenja je visoka razina automatizacije uz potrebnu tehničku i vremensku fleksibilnost koji su poduprti organizacijskim mjerama, kako bi sustav mogao biti fleksibilno korišten.

Jedan je od primarnih ciljeva pri tom, stvaranje uvjeta za dobro iskorištenje FPS-a, kako bi se uz visoku proizvodnost osigurala zadovoljavajuća ekonomičnost proizvodnje i pored vrlo visokih ulaganja uzrokovanih velikom kompleksnošću sustava.

Cilj je ovog rada omogućiti simultanu proizvodnju dijelova s različitim kombinacijama tipova dijelova i količina, koja će osigurati maksimalno iskorištenje proizvodnog sustava.

Pretpostavka za dobro iskorištenje fleksibilnog proizvodnog sustava je formiranje radnog naloga s takvom strukturom tipova dijelova, koja će omogućiti odvijanje proizvodnje uz najmanje vremensko opterećenje proizvodnog sustava kao

cjeline. No struktura tipova dijelova može biti unaprijed zadana, npr. primjenom pravila prioriteta na temelju plana montaže.

Kada je struktura radnog naloga zadana treba tražiti načine za maksimalno iskorištenje proizvodnog sustava u datim okolnostima. Tada su od presudnog značaja mogućnost korištenja pojedinih strojeva sustava i redoslijed odvijanja poslova pri realizaciji radnog naloga. Osnovu rješenja problema tada mogu dati alternativni planovi procesa pri proizvodnji pojedinih dijelova u radnom nalogu. Alternativni planovi procesa temelje se na različitom korištenju pojedinih strojeva sustava.

Rješenje zadatka moguće je u dva koraka.

Prvi je korak, odrediti onu kombinaciju planova procesa između razvijenih planova procesa, koja će pri simultanoj proizvodnji različitih dijelova zadanih količina, osigurati najmanje vremensko opterećenje proizvodnog sustava poznatih karakteristika.

U drugom koraku potrebno je odgovarajućim upravljanjem redoslijednog odvijanja pojedinih operacija radnog naloga provesti minimizaciju ukupnog vremena čekanja strojeva i dijelova, tj. potrebno je optimizirati redoslijed ulaska pojedinih dijelova u proces izrade.

Rješenje zadatka optimizacije zahtijeva odabir optimizacijske metode. Do sada razvijene tradicionalne metode upravljanja redoslijedom odvijanja procesa ukazuju na ograničenu uspješnost pri rješavanju ovog kombinatornog problema optimizacije zbog realno vrlo velikog broja mogućih varijabli i mogućih kombinacija što je redovit slučaj kod fleksibilnih proizvodnih sustava. Uspješnijim su se pokazale do sada razvijene aproksimativne metode temeljene na lokalnoj potrazi za rješenjem kao što su: simulirano približavanje, tabu tehnika pretraživanja, metoda premještaja uskog grla i mnoge druge.

Ostvarenje cilja rada provedeno je razvojem metode temeljem provedenog istraživanja, koja osigurava osnovu za praktično rješenje problema upravljanja radnim nalogima u okviru fleksibilnog proizvodnog sustava.

U ovom radu odabran je Goldbergov genetski algoritam zbog svojih velikih potencijalnih mogućnosti. Genetski algoritmi ne zahtijevaju derivabilnost niti bilo kakvu drugu regularnost funkcije cilja i prostora parametara. Također ne ovise o početnoj procjeni ekstrema, već je uopće i ne trebaju, te uspješno nalaze globalne, a ne samo lokalne ekstreme. Izuzetno su učinkoviti već i u najjednostavnijem obliku s tri osnovna operatora odabir-križanje-mutacija. Iako matematička teorija nije do kraja razjasnila zašto genetski algoritmi tako dobro rade, primjena svakodnevno raste, jer ne postoji nikakvo ograničenje na područja i zadatke optimizacije, koji se genetskim algoritmima mogu riješiti.

Razvijena metoda uključuje programsku podršku računalom, a dobiveni rezultati mogu biti korišteni pri oblikovanju sustava, ali također i pri planiranju i vođenju odvijanja proizvodnje.

Valjanost metode pokazana je rješenjem konkretnog zadatka.

Sam rad raščlanjen je u sedam poglavlja. U uvodu je, ukratko formuliran problem koji će se u radu riješiti, te naglašen cilj rada.

U drugom poglavlju rada analizirane su pojedine prostorne strukture proizvodnih sustava, te identificiran fleksibilni proizvodni sustav i njegove značajke u svezi gradnje, korištenja i upravljanja. Ukazano je također na tendencije u razvoju.

Problem upravljanja nalogima za proizvodnju različitih proizvoda s različitim brojem i trajanjem operacija s različitim redoslijedom korištenja pojedinih radnih mjesta

u proizvodnom sustavu, obrađen je u trećem poglavlju. Osim toga dan je i pregled nekih tradicionalnih metoda, koje se uglavnom temelje na primjeni metoda matematičkog programiranja, kao i pregled nekih od heurističkih metoda, koje su se pokazale uspješnijim od tradicionalnih metoda.

Doprinos metodi rješavanja problema upravljanja FPS-a obrađen je teoretski u četvrtom poglavlju. Najprije je izvršena identifikacija i valorizacija utjecajnih faktora na upravljanje redoslijedom odvijanja procesa kod fleksibilnih proizvodnih sustava, kao i definiranje elemenata kriterija za optimalno odvijanje procesa na FPS-u, na temelju kojih je razrađena sistematika i metoda optimizacije redoslijeda izrade proizvoda i odvijanja procesa.

U petom poglavlju definirana je praktički primjenjiva, računalom podržana, pogodna metoda optimizacije redoslijeda odvijanja procesa kod fleksibilnih proizvodnih sustava, temeljena na primjeni genetskih algoritama

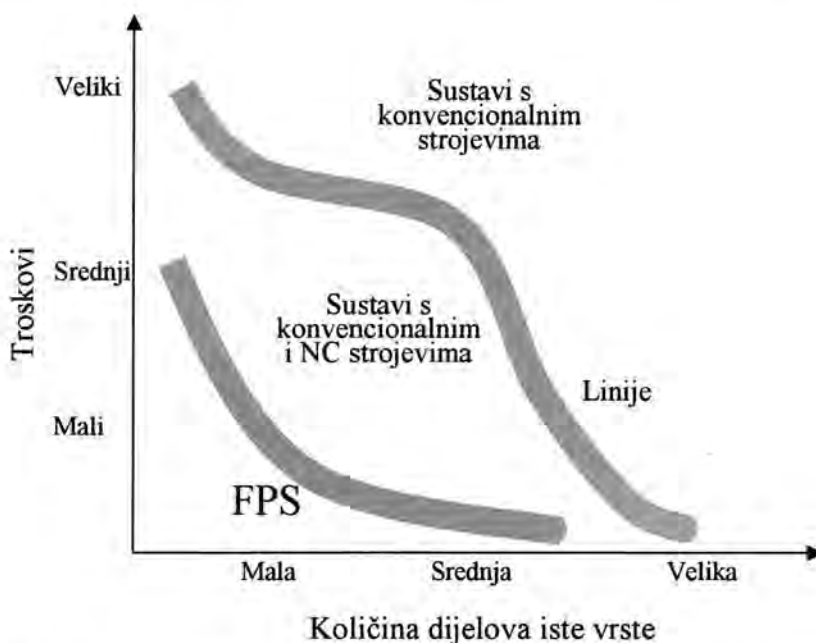
U šestom je poglavlju rješanjem konkretnog zadatka i analizom rezultata pokazana valjanost metode.

Završno, sedmo poglavlje sadrži zaključak i moguće smjernice za budući znanstveno- istraživački rad na ovom području

2. FLEKSIBILNI PROIZVODNI SUSTAVI

Razvitak fleksibilne tehnologije omogućuje povećanje proizvodnosti i gospodarstvenosti proizvodnje manjih količina proizvoda. Da bi se ovo ostvarilo, potrebni su automatizirani sustavi najčešće temeljeni na fleksibilnim obradnim ćelijama koje omogućuju veliku koncentraciju obrada uz automatsko posluživanje.

Suvremeno koncipirani alatni strojevi i obradni sustavi temeljeni na NC alatnim strojevima, postaju lako prilagodljivi za različite radne zadatke i lako se mogu povezati u fleksibilne proizvodne sustave upravljane elektroničkim računalima. Ovdje je za automatizaciju proizvodnih procesa bitno posluživanje fleksibilnog proizvodnog sustava, koje se može izvesti s industrijskim robotima i manipulatorima, uz automatiziranje transporta i spremišta izradaka i alata. Primjena industrijskih robota ima opravdanja i u nastojanju da se čovjeka oslobodi monotonog i ponavljajućeg rada, kao i u slučajevima vrlo teških uvjeta rada, posebno onih štetnih po zdravlje radnika.



Sl. 2.1 Primjenljivost pojedinih proizvodnih sustava u proizvodnji (prema [2])

Upravo težnja za smanjenjem troškova proizvodnje i zahtjevi koji proizlaze iz povećanja kompleksnosti proizvoda i širenja različitih varijanti, uz istovremeno smanjenje životnog vijeka proizvoda, uvjetuju potrebu da se u proizvodnju lansiraju sve manje količine izradaka, čak i do serija veličine jedan. Optimalnim određivanjem veličine serije osigurava se ujedno i uravnotežen dotok gotovih izradaka na montažu. To su razlozi koji upućuju na sve veću zastupljenost pojedinačnog, maloserijskog i srednjeserijskog tipa organizacije, pripreme i vođenja proizvodnje [1]. Za razliku od klasične automatizacije, koja se gospodarstveno primjenjuje kod visokoserijske i masovne proizvodnje, fleksibilna automatizacija je rentabilna i kod maloserijske, čak i pojedinačne proizvodnje [2,3,4], slika 2.1.

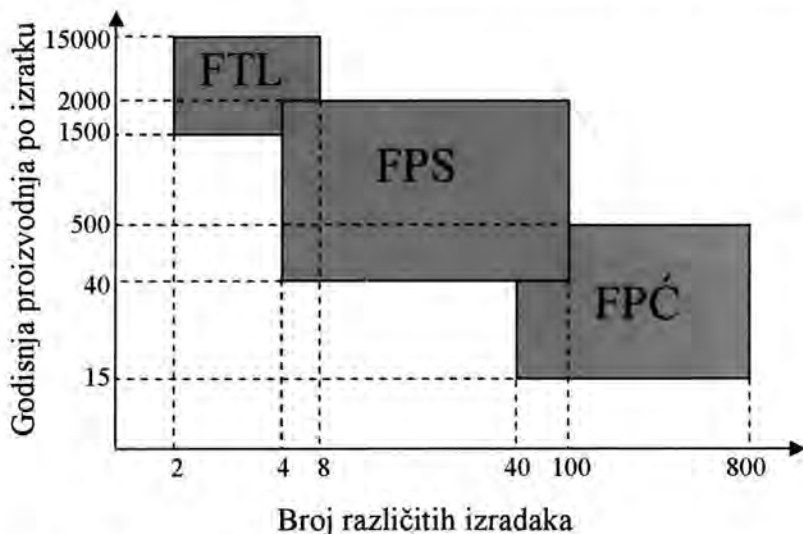
2.1. FLEKSIBILNE PROIZVODNE PROSTORNE STRUKTURE

Razvitkom automatizacije obradnih sustava temeljenih na NC¹ alatnim strojevima, a u cilju ispunjenja zahtjeva tržišta, javljaju se fleksibilne proizvodne prostorne strukture.

Uzimajući u obzir prilagodljivost u odnosu na promjene proizvodnog programa po asortimanu i količini te stupanj integracije i automatizacije, razlikujemo sljedeće proizvodne kapacitete fleksibilne tehnologije:

- fleksibilna proizvodna ćelija (FPĆ)
- fleksibilni proizvodni sustav (FPS)
- fleksibilna transfer linija (FTL)

Broj različitih elemenata koji se obrađuju i količina proizvoda u seriji su dva odlučujuća kriterija za primjenu nekih od proizvodnih kapaciteta fleksibilne tehnologije. U nastavku se daju orijentacioni podaci prema [2,3]. Fleksibilne transfer linije su izgrađene na način koji omogućuje, da se mali broj različitih proizvoda (2 do 8) izrađuje u relativno velikim količinama (1500 do 15000). FPS se koristi za 4 do 100 različitih proizvoda u količinama od 40 do 2000 komada. Fleksibilne ćelije za obradu primjenjuju se za veliki broj različitih proizvoda (40 do 800) a malu količinu (15 do 500), slika 2.2. Za svaki od slučajeva je potrebno izraditi konkretnu ekonomsku analizu. Navedeni orijentacioni podaci određuju dominantna područja korištenja pojedinih fleksibilnih struktura iako danas srećemo pojedina rješenja i znatno izvan navedenih granica.



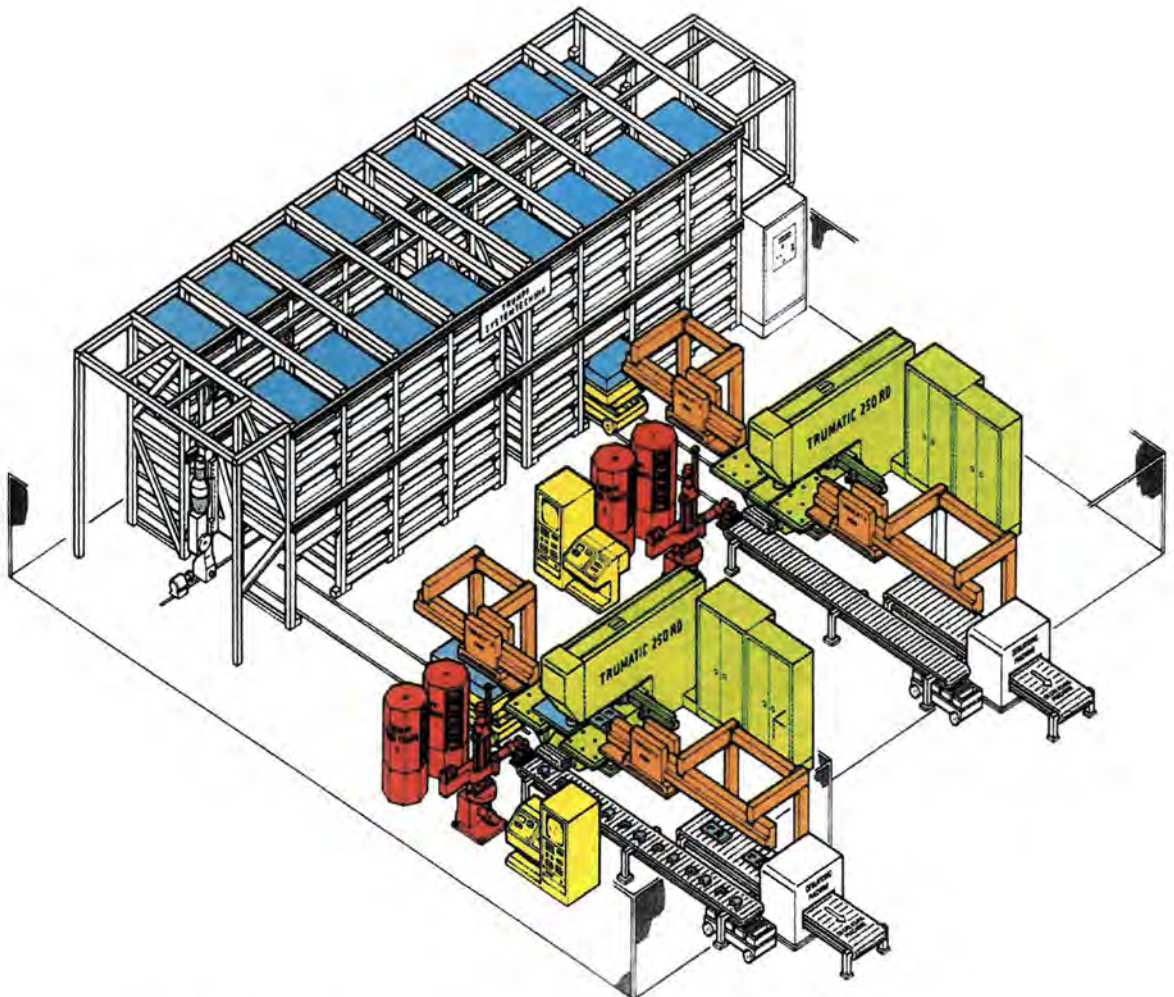
Sl. 2.2 Područja primjene pojedinih fleksibilnih proizvodnih kapaciteta

Na početku razvoja fleksibilni se proizvodni sustavi u odgovarajućim izvedbama upotrebjavaju za obradu prizmatičnih i asimetričnih izradaka kao i za rotaciono simetrične komade. Za to su potrebni, pored raznih NC-alatnih strojeva i

¹ Numerical Control

uređaja i različiti transportni sustavi. Prizmatski izratci se pretežno stežu, pojedinačno ili u grupama, na paletama i zajedno s njima transportiraju. Kod rotaciono simetričnih dijelova koristi se transportna košara s kojom se transportiraju ili se stavljaju na traku koja ih transportira. Namjesto izmjenjivača paleta kod prizmatičnih dijelova, kod rotaciono simetričnih dijelova se rabi uređaj za rukovanje koji neobrađene dijelove pojedinačno vadi iz sanduka ili transportne trake i prenosi do stezne glave, gdje dvostruki manipulator zamjenjuje obrađene izratke sa neobrađenim i stavlja obrađene komade u sanduk za gotove komade.

Primjena fleksibilne tehnologije danas je proširena na različita proizvodna područja uz primjenu različitih postupaka kao na primjer: izrada odvajanjem čestica, izrada limenih dijelova, montaža i sl.

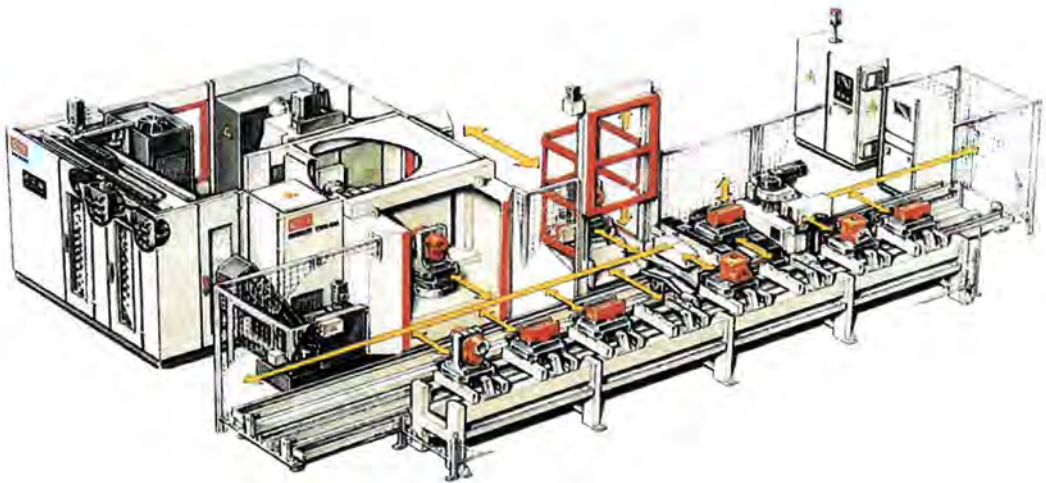


Sl. 2.3. *Fleksibilni proizvodni sustav za obradu limenih dijelova*

Na slici 2.3 prikazan je FPS za obradu limenih dijelova koji se sastoji iz dva stroja za obradu lima s automatskim sustavom izmjene alata, zatim spremnika materijala i alata i vodećeg sustava s računalom za upravljanje i nadzor postrojenja.

2.1.1. Fleksibilna proizvodna ćelija

FPĆ se definira kao obradna struktura u kojoj se mogu automatizirano obrađivati različiti prizmatični ili rotaciono simetrični izratci u pojedinačnoj ili maloserijskoj proizvodnji. Fleksibilna proizvodna ćelija zasnovana je na vođenju tehnološkog procesa na jednom CNC² alatnom stroju, a sadrži međuskladišta izradaka i alata, uređaje za automatsko rukovanje izratcima, automatsku izmjenu alata, te automatsku kontrolu i korekciju, u cilju održavanja kvalitete. Na slici 2.4 prikazana je fleksibilna proizvodna ćelija s linearnim skladištem paleta, a sastoji se iz jednog obradnog centra, transportnih kolica za palete, zatim više mjesta za stezanje i skladištenje paleta i jednog uređaja za upravljanje ćelijom. Stroj u fleksibilnoj proizvodnoj ćeliji, kao i procesi, upravljani su direktnim numeričkim upravljanjem [5,6,7,8].



Sl. 2.4. *Fleksibilna proizvodna ćelija s linearnim skladištem paleta*

FPĆ je namijenjena u pravilu kompletnoj obradi familije kompleksnih dijelova određenog nivoa tehnološke sličnosti, a proces može biti realiziran ili paralelnim vođenjem tehnoloških procesa na određenoj grupi dijelova ili sukcesivnim sekvencijalnim vođenjem procesa na određenoj količini pojedine vrste izradaka. Načinu vođenja procesa prilagođena je i izvedba ćelije.

Prijelaz s obrade jedne na obradu druge vrste dijelova vrši se uz automatsku (kod paralelnog vođenja tehnološkog procesa) ili ručnu (kod sekvencijalnog vođenja tehnološkog procesa) pripremu i raspripremu radnog mjesta [9].

Proizvodni se proces na FPĆ može odvijati bez prisustva poslužioaca što znači da se i sve pomoćne aktivnosti izvode automatski (izmjena izradaka, izmjena alata, priprema i raspriprema radnog mjesta, kontrola).

² *Computer Numerical Control*

FPC su najčešće temeljene na obradnim centrima za bušenje i glodanje i obradnim centrima za kombiniranu obradu tokarenjem, bušenjem i glodanjem [10,11].

Fleksibilna ćelija je samostalni obradni sustav koji po svojim karakteristikama i zastupljenosti danas čini ključni element fleksibilne integrirane proizvodnje.

2.1.1.1. Prednosti FPC

- ukupan tehnološki proces se izvodi na jednom radnom mjestu,
- ne postoji međuoperacijski transport,
- ciklus izrade vrlo kratak
- upravljaljivost procesom vrlo visoka
- fleksibilnost na promjenu geometrije i količina vrlo visoka
- opći uvjeti za postizanje kvalitete izvanredno visoki,
- iskoristivost opreme vrlo dobra
- moguć rad u pojedinim vremenskim periodima bez nadzora,
- efikasnost proizvodnog sustava izvanredno visoka.

2.1.1.2. Nedostaci FPC

- visoki zahtjevi na tehnološku pripremu proizvodnje
- visoki zahtjevi na kvalifikaciju i odgovornost osoblja, koje vrši poslove pripreme, nadzora i održavanja fleksibilnih proizvodnih ćelija.

2.1.2. Fleksibilni proizvodni sustavi

Povezivanjem više radnih stanica: NC alatnih strojeva ili uređaja, obradnih centara ili fleksibilnih ćelija u jednu cjelinu formiramo fleksibilne proizvodne sustave, slika 2.5.

U sklopu FPS-a redovito se nalazi i automatski mjerni uređaj koji je preko centralnog računala izravno vezan u proces omogućujući informacije o stanju kvalitete proizvedenih izradaka, a niz senzorskih uređaja omogućuje i nadzor stanja alata u toku procesa [12].

Radne stanice FPS-a imaju pored automatiziranog izvođenja operacija, automatsku izmjenu izradaka, a automatiziranim transportom te međuskладиštem materijala, izradaka i alata povezani su u jednu cjelinu, pri čemu je cijeli sustav integriran i upravljan centralnim računalom. Također postoji mogućnost povezivanja s pripremom proizvodnje: automatiziranim projektiranjem i konstruiranjem (CAD³), projektiranjem tehnologije (CAPP⁴), upravljanjem kvalitetom (CAQ⁵) i operativnim upravljanjem radnim nalogima (CAPP⁶).

Na fleksibilnim proizvodnim sustavima može se obrađivati više različitih izradaka istovremeno, pa upravo tako koncipirani proizvodni sustavi pokrivaju područje maloserijske i srednjeserijske proizvodnje osiguravajući kompromis utjecajnih

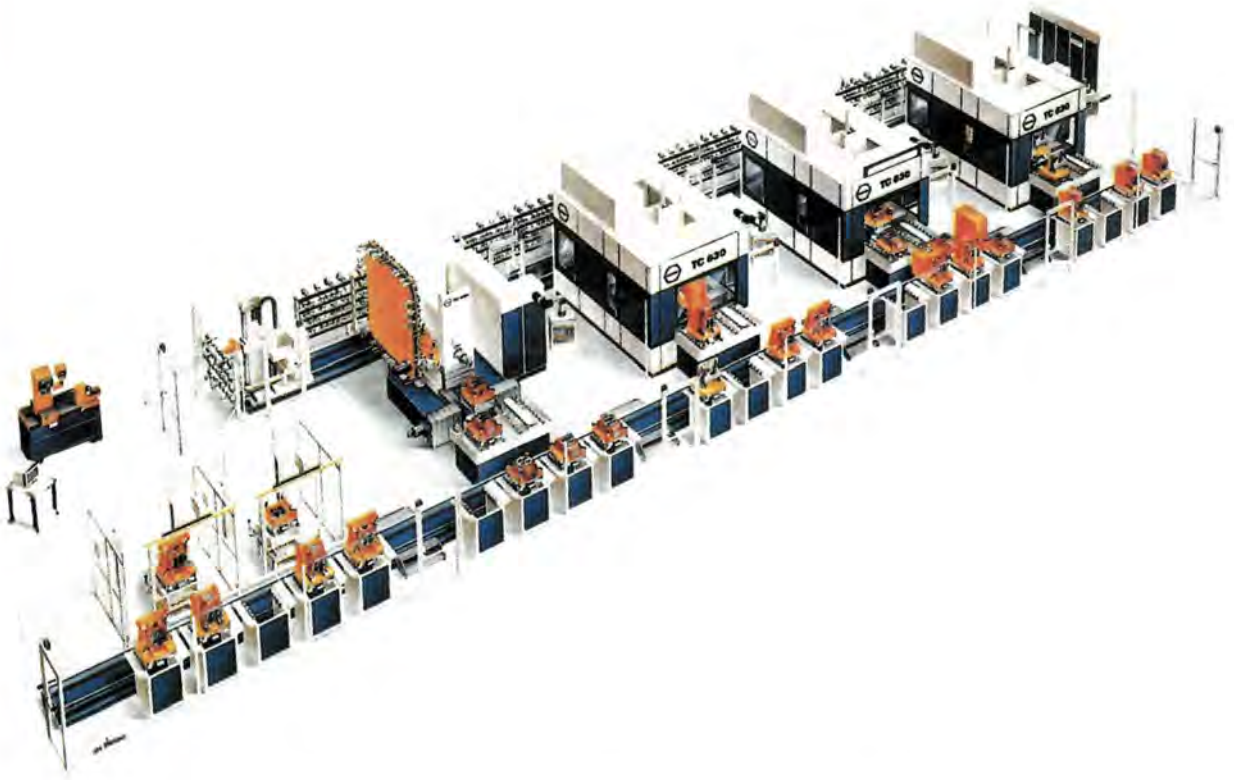
³ *Computer Aided Design*

⁴ *Computer Aided Process Planning*

⁵ *Computer Aided Quality Assurance*

⁶ *Computer Aided Production Planning and Control*

čimbenika, omogućujući istovremeno dovoljnu fleksibilnost, proizvodnost i stabilnost kvalitete, a time i efikasnost kakva se inače postiže u proizvodnji velikih količina.



Sl. 2.5 *Fleksibilni proizvodni sustav s automatskim snabdijevanjem izradaka i alata posredstvom tračničkog transportnog vozila*

2.1.2.1. Prednosti FPS-a

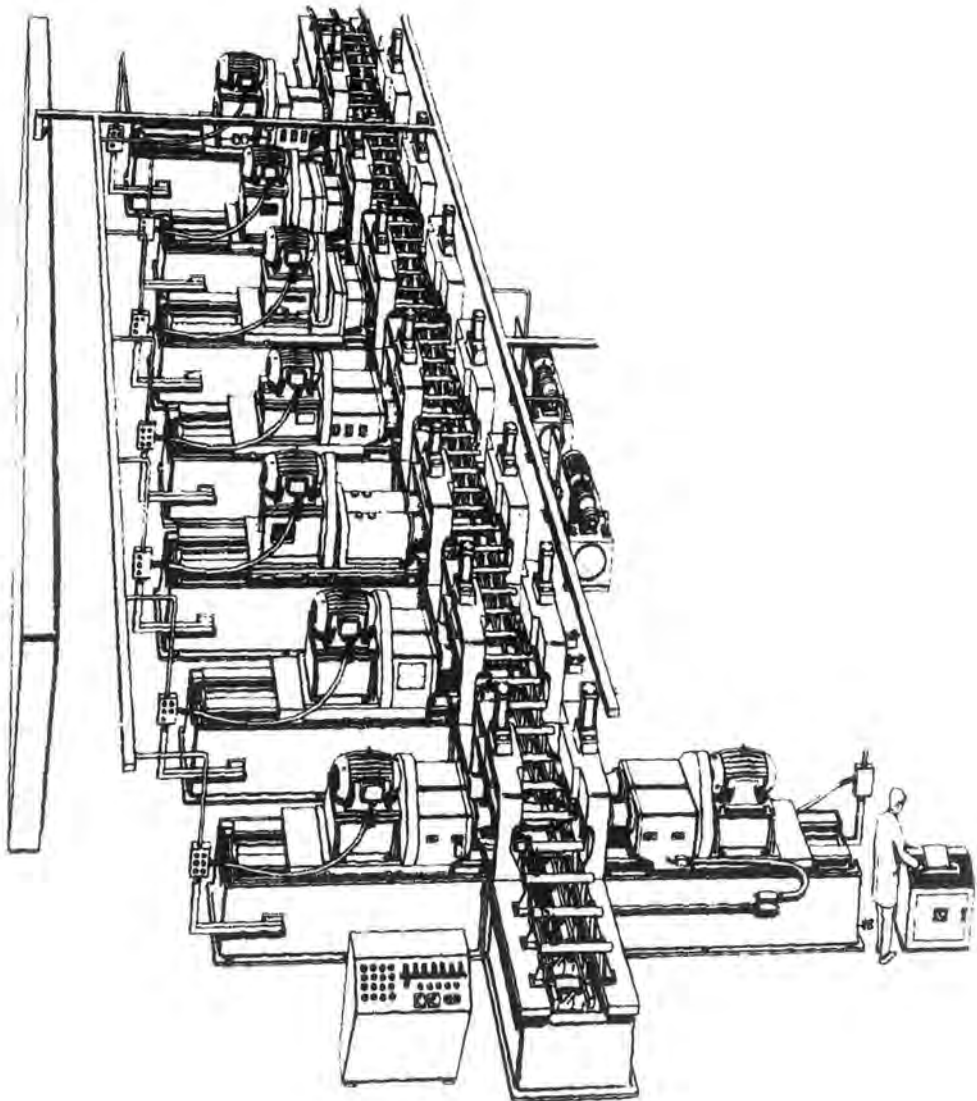
- istovremeno vođenje tehnoloških procesa na grupi različitih izradaka,
- vrlo visoka fleksibilnost na promjenu geometrije i količina,
- automatsko odvijanje proizvodnog procesa,
- kratak ciklus izrade i ciklus proizvodnje,
- mogućnost povremenog odvijanja procesa bez nadzora poslužioaca, odnosno uz smanjeni nadzor (rad u 3. smjeni),
- opći uvjeti za postizanje kvalitete vrlo visoki,
- efikasnost proizvodnog sustava vrlo visoka.

2.1.2.2. Nedostaci FPS-a

- visoki zahtjevi na tehnološku pripremu proizvodnje,
- visoki zahtjevi na sustav osiguranja i snabdijevanja alatom,
- visoki zahtjevi na kvalifikacije i odgovornost osoblja koje vrši poslove pripreme, nadzora i održavanja sustava,
- visoka cijena sustava.

2.1.3. Fleksibilna transfer linija

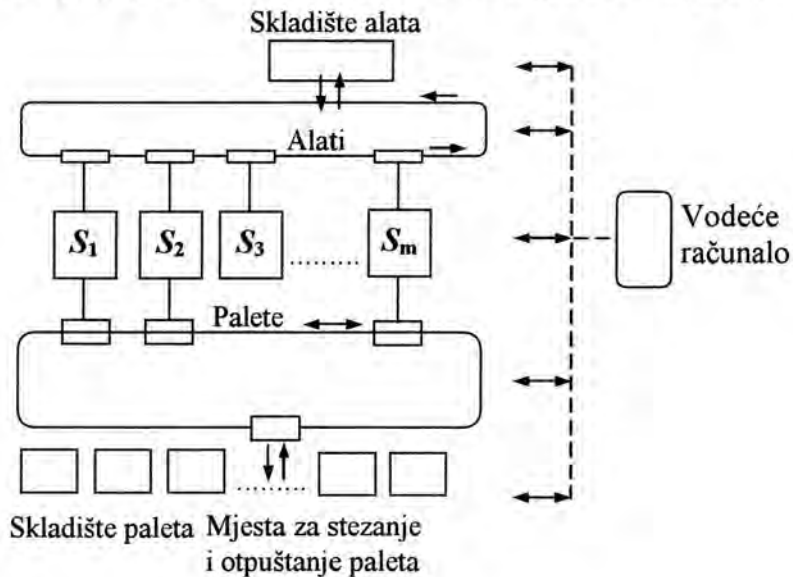
Kod fleksibilnih transfer linija, više različitih ili istovrsnih NC alatnih strojeva izvodi sve zahtijevane operacije pri obradi jedne skupine tehnološki sličnih dijelova u automatskom slijedu uz istosmjerni transport, slika 2.6. Svi dijelovi prolaze pojedine radne stanice u čvrstom redoslijedu pri čemu se pojedini dijelovi obrađuju prema odgovarajućim NC-programima. Operacije na pojedinom stroju pri obradi različitih dijelova mogu se razlikovati u granicama mogućnosti strojeva bez preopremanja radnog mjesta prema pripremljenim NC-programima. To znači, da dijelovi moraju posjedovati relativno veliku tehnološku sličnost, koja omogućuje korištenje istih radnih mjesta u istom redoslijedu.



Sl. 2.6 *Fleksibilna transfer linija*

2.2. PODSUSTAVI FPS-a

Fleksibilni proizvodni sustav obuhvaća više proizvodnih uređaja, povezanih upravljačkim i transportnim sustavom, a omogućuje automatsku obradu i izvođenje, u datom području, različitih operacija na raznim kompleksnim izradcima. Vodeće računalo koordinira odvijanje procesa, kao i za pogon sustava potrebnu logistiku, slika 2.7 [8,13].



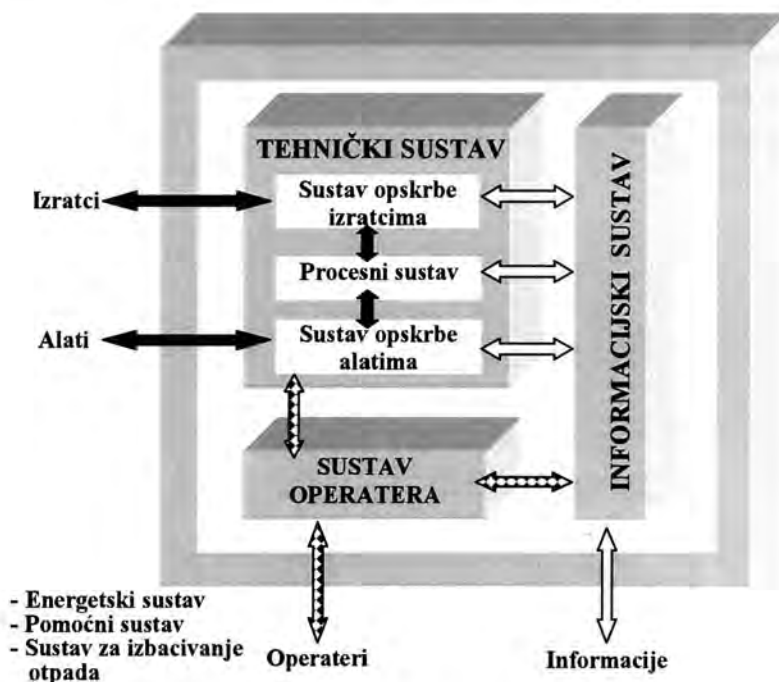
Sl. 2.7 Opći model i karakteristike FPS-a

Prema [11,13], funkcijska struktura i karakteristike FPS-a date su na slici 2.8.



Sl. 2.8 Struktura i karakteristike FPS-a

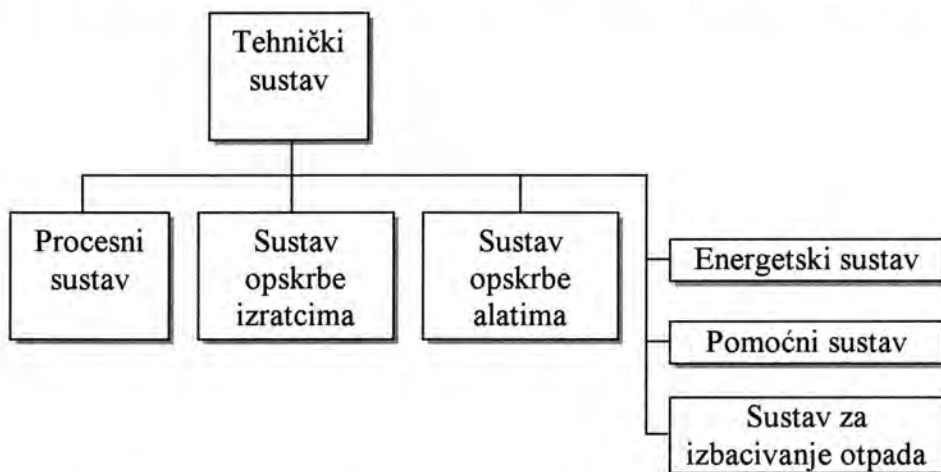
U daljnjim razmatranjima koristit će se podjela FPS-a prema [14] na nekoliko međusobno usko vezanih podsustava, kao što je prikazano na slici 2.9.



Sl. 2.9 Podsustavi FPS-a

2.2.1. Tehnički sustav

Tehnički sustav se sastoji od procesnog sustava, sustava opskrbe radnim komadima i sustava opskrbe alatima. Dodatno, tu su uključeni i specijalni logistički podsustavi kao što je energetski sustav i pomoćni sustav, odgovorni za opskrbu sredstvima za podmazivanje i hlađenje, te sustav za izbacivanje otpada, slika 2.10.



Sl. 2.10 Komponente tehničkog sustava FPS-a

2.2.1.1. Procesni sustav

Procesni se sustav FPS-a sastoji od strojeva za obradu sa sustavom za izmjenu alata i skladištenje alata, stanica za kontrolu kvalitete (mjernih stanica) i uređaja za pranje, slika 2.11. Strojevi i uređaji su numerički upravljani strojevi (CNC strojevi)[15]. Ovi strojevi rade prema slijedećem principu: nakon što je radni komad čvrsto pozicioniran, alat i radni komad se mogu pomicati relativno jedan u odnosu na drugoga što omogućuje izvođenje obrade. Svaki od ovih pomaka upravljan je posredstvom naredbi iz NC programa.



Sl. 2.11 Komponente procesnog sustava FPS-a

CNC strojevi omogućavaju, da se NC program učita u upravljačku jedinicu stroja. CNC stroj spojen je na određeno vodeće računalo FPS-a s kojega se u upravljačku jedinicu NC stroja učitava zahtijevani program. Kod promjene radnog komada, učitavanje novog NC programa traje vrlo kratko.

Alatni strojevi se, prema stupnju prilagodljivosti pri promjeni radnog zadatka, mogu podijeliti na univerzalne strojeve (često su to obradni centri) ili specijalne strojeve. Univerzalnim strojevima moguće je izvesti različite obrade (npr. bušenje, glodanje i tokarenje) na različitim radnim komadima, koji su stegnuti u radnom prostoru. Ovo je moguće uz izmjenu reznih alata prema NC programu. Takva mogućnost jako je ograničena kod specijalnih strojeva (npr. specijalni strojevi za bušenje, specijalni strojevi za brušenje, specijalni strojevi za pranje), ali zato oni imaju u pravilu znatno veću proizvodnost. Veliki broj alata pohranjenih i upotrijebljenih na CNC strojevima imaju određenu često i veliku vrijednost, no, samo se jedan alat može koristiti u datom trenutku. Univerzalni stroj je visoko iskorišten, ali alati u spremniku nisu, tako da uvijek postoji dio investiranog kapitala koji nije u uporabi. Specijalni strojevi ponekad su nužni a često su samo djelomično iskorišteni čak i unutar optimalne konfiguracije FPS-a.

Ovisno o obradama koje se trebaju izvršiti na strojevima, FPS može biti izveden kao sustav s identičnim strojevima (identični strojevi po izvedbi i po opremi alatima) ili različitim strojevima (različiti strojevi po izvedbi i opremi ili identični strojevi ali različito opremljeni), te kao kombinirani sustav. Identični strojevi mogu izvršiti identične radne zadatke. Identični strojevi mogu se koristiti za izvođenje svih operacija određenog radnog zadatka, što omogućuje slobodan izbor stroja u redoslijedu izvedbe operacije. Osnovna je prednost ovakvih sustava što kod ispada jednog stroja

ostali strojevi mogu preuzeti njegove radne zadatke, te ne može doći do ispada kompletnog sustava već samo do pada kapaciteta. Treba imati na umu da se, ipak, identične strojeve može koristiti na taj način ograničeno kod operacija s jako velikim brojem različitih alata obzirom na ograničenu veličinu spremnika alata na stroju, što onda zahtijeva sukcesivnu nadopunu. Kod toga postoji i varijanta da se određeni strojevi sustava koriste samo za npr. visokoprecizne obrade što također smanjuje fleksibilnost sustava. Isto tako može se pokazati svrsishodnim sve skupe specijalne alate koncentrirati na jednom stroju i na njemu izvoditi pripadajuće operacije. U takvom slučaju prednost je, da se takvi alati ne moraju stalno prenositi sa stroja na stroj što je inače čest slučaj, kako bi se smanjila ulaganja u skupe alate. FPS-i temeljeni na različitim strojevima predmet su zanimanja ovog rada prvenstveno iz razloga mogućnosti nejednakomjernog opterećenja pojedinih radnih mjesta, kao i mogućnosti ispada sustava zbog ispada pojedinog stroja.

Spremnik alata može biti kazeta, bubanj, disk ili lanac. Spremnik alata može sadržavati velik broj različitih alata iako ih može biti više od istog tipa. Ukoliko se, nakon izvršenja radne operacije ili po isteku vijeka trajanja alata, zatraži novi alat, upotrijebljeni se alat skida s vretena stroja i zamijeni novim alatom iz spremnika alata pomoću uređaja za automatsku izmjenu. Broj raspoloživih alata u spremniku alata ovisi najvećim dijelom o tipu i veličini spremnika. Spremnici, ovisno o izvedbi, sadrže između 20 i 200 alata. Budući da dio alata ima veću vrijednost, tendencija je kod skupljih alata, imati jedan alat (ili ograničen broj alata) iste vrste u FPS-u. Ova strategija zahtijeva kod zatupljenja češću supstituciju novim alatom iz skladišta alata. O tome treba voditi računa, kako sustav opskrbe alatima ne bi postao usko grlo FPS-a. Olakšanje problema doprinosi ograničenje broja različitih izradaka u okviru radnog naloga, koji se realizira u određenom vremenu rada sustava.

Cilj vođenja procesa na FPS-u je proizvodnja kvalitetnih dijelova sposobnih za ugradnju. Zbog toga su u pravilu sastavne komponente procesnog sustava uređaj za pranje i stanica za kontrolu kvalitete.

2.2.1.2. Sustav opskrbe radnim komadima

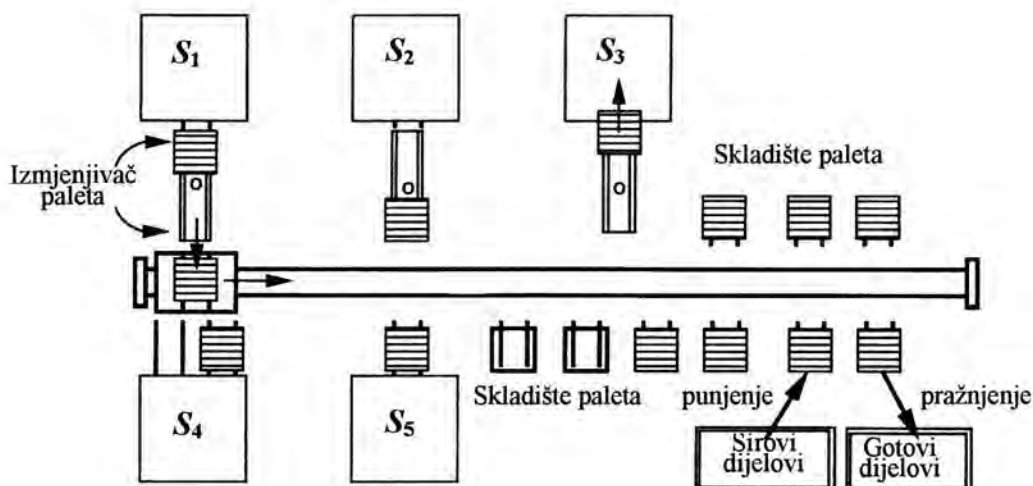
Sustav opskrbe radnim komadima sastoji se od sustava transporta, sustava skladištenja, sustava za stezanje na transportne (radne) palete i otpuštanje i sustava za rukovanje, slika 2.12.



Sl. 2.12 Sustav opskrbe radnim komadima

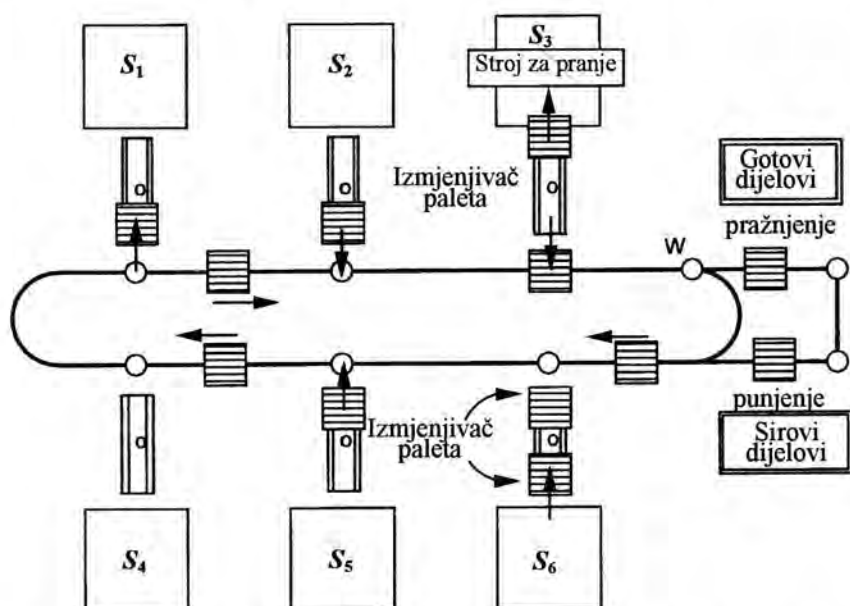
Sustav transporta radnih komada ima za cilj protok radnih komada između radnih stanica, sustava skladištenja i sustava za stezanje i otpuštanje.

Radni se komadi transportiraju na paletama. Na slici 2.13, 2.14 i 2.15 prikazani su osnovni načini transporta izradaka na paletama [16,17].



Sl. 2.13 Linijski, tračnički transportni sustav s paralelnim obostranim rasporedom strojeva

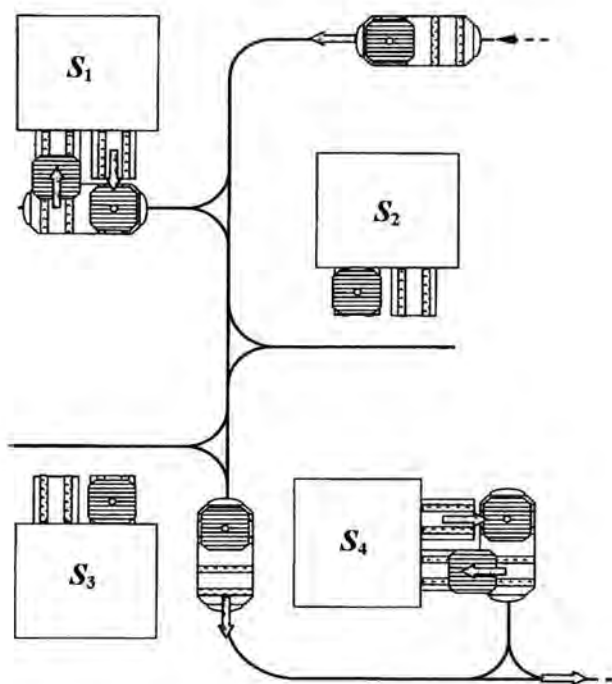
Na slici 2.13 prikazan je linijski transportni sustav. Značajke tog sustava su: transportna kolica raspolažu s jednim transportnim mjestom. Prema potrebi može biti korišteno jedno vozilo sa dva mjesta ili dodano vozilo s odgovarajućom izmjenom



Sl. 2.14 Paletni optočni sustav s paralelnim rasporedom strojeva na obje strane transportne pruge

upravljanja transportom. Napunjene palete odlažu se na skladišna mjesta odakle ih po potrebi preuzima transportno vozilo. Strojevi 1,2 i 3 su opremljeni izmjenjivačem paleta, stroj 4 ima po jedno čvrsto ulazno i izlazno mjesto, a stroj 5 je opremljen samo s jednim mjestom za izmjenu paleta.

Značajke optoćnog transportnog sustava prikazanog na slici 2.14 su: ne postoje dodatna mjesta za skladištenje paleta. Neobrađeni dijelovi kruže na transportnom sustavu dok se ne oslobodi ulazno mjesto izmjenjivača paleta odgovarajućeg stroja. Izmjenjivači paleta ispred strojeva su istovremeno ulazna i izlazna međuskladišta za jednu gotovu i jednu neobrađenu paletu. Obradene palete se po napuštanju stroja vode automatski, a nakon uređaja za pranje preko skretnice "w" prema stanicama za stezanje i otpuštanje.



Sl. 2.15 Mrežni transportni sustav s automatski vođenim vozilima

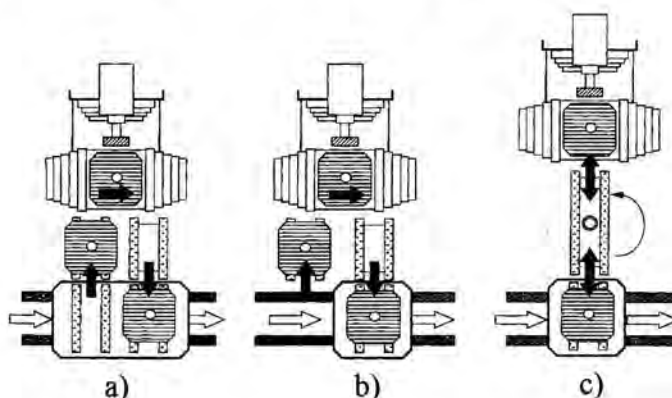
Mrežni transportni sustav prikazan na slici 2.15 koristi zadane transportne puteve. Vozila se indukcijom vode preko vodova u podu. Značajke tog sustava su: strojevi mogu biti slobodno raspoređeni na raspoloživoj površini. Vozila voze u "jednotračnom prometu" od skladišta materijala do strojeva i nazad. Tamo se pune i prazne dok je brzina vozila iz sigurnosnih razloga ograničena.

Sustav međuskladištenja radnih komada ima za zadatak da uskladišti radni komad (već učvršćen na paleti) između radnih operacija, koje se izvode na različitim strojevima. U FPS-u razlikujemo raspoloživa privremena međuskladišta prema njihovoj lokaciji kao lokalna ili centralna privremena međuskladišta. Lokalna privremena međuskladišta dodjeljuju se određenoj stanici za obradu ili stanici za stezanje i otpuštanje kao ulazno/izlazna privremena međuskladišta. Stoga su dostupna samo za uskladištenje radnih komada za određenu stanicu. S druge strane su centralna

privremena međuskладишта dostupna za skladištenje svih radnih komada. Centralna se privremena međuskладишта koriste kada su popunjeni kapaciteti lokalnih.

Sustav za stezanje i otpuštanje radnih komada je namijenjen za učvršćivanje i otpuštanje radnih komada na i sa paleta. Uz to, mjesta za stezanje-otpuštanje se često koriste za promjenu položaja djelomično obrađenog izratka da bi se stroju omogućio pristup prethodno nedostupnim površinama. Zahtijevane se radnje uobičajeno izvode ručno na stanicama za stezanje-otpuštanje. Gledajući sa stanovišta radnog komada, stanice za stezanje-otpuštanje predstavljaju ulaz i izlaz iz FPS-a. Rotaciono simetrični radni komadi, obično, ulaze u sustav u transportnoj košari ili na paleti, ali neučvršćeni. Takve komade robot – uređaj za rukovanje premješta u radno područje stroja uz automatsko stezanje. Radni komadi prizmatičnog i asimetričnog oblika se učvršćuju na paletu posredstvom specijalnih steznih radnih naprava [18]. Na paleti je ovisno o izvedbi uređaja za stezanje moguće stegnuti i veći broj istovrsnih ili različitih izradaka, dostupnih s jedne ili više strana.

Sustav za rukovanje radnim komadima ima zadatak osiguranja promjene položaja paleta između radnih mjesta za stezanje i otpuštanje, transporta i pojedinih radnih stanica. Tipična radnja kod rukovanja radnim komadom je prijenos paleta iz međuskладиšta u radno područje stroja. Na slici 2.16 prikazana su tri različita principa izmjena paleta s transportnim kolicima [16,17].



Sl. 2.16 *Tri različita principa izmjena paleta*
 a) *Transportna kolica s dva paletna mjesta*
 b) *Transportna kolica s jednim paletnim mjestom*
 c) *Okretni paletni izmjenjivač na stroju*

2.2.1.3. Sustav opskrbe alatima

Slično sustavu za opskrbu radnim komadima, sustav opskrbe alatima transportira, priprema i uskladišćuje alate zahtijevane u procesu obrade radnih komada. Alati se zamjenjuju po isteku njihova vijeka trajanja. Također se zamjenjuju ukoliko se izmijeni grupa tipova radnih komada koje je u FPS-u potrebno obraditi u određenom vremenskom razdoblju – ukoliko su različiti zahtjevi za alatom. Sustav opskrbe alatima može se razlikovati prema tome da li se opskrba spremnika alata novim alatima izvodi ručno (decentralizirana opskrba alatima) ili automatizirano (centralizirana opskrba

alatima). Kod decentralizirane opskrbe alatima periodično se, ručno, puni spremnik alata; npr. kazeta s alatima koji ne odgovaraju više zahtjevima obrade zamijeni se drugom uz pomoć vozila za transport alata. Tijekom postupka zamjene rad na stroju je često zabranjen; proces obrade se mora prekinuti. Rezultirajući gubitak procesnog vremena je veliki razlog zašto je poželjno obrađivati radne komade sličnih ili identičnih zahtjeva po pitanju alata, u okviru istog radnog naloga.

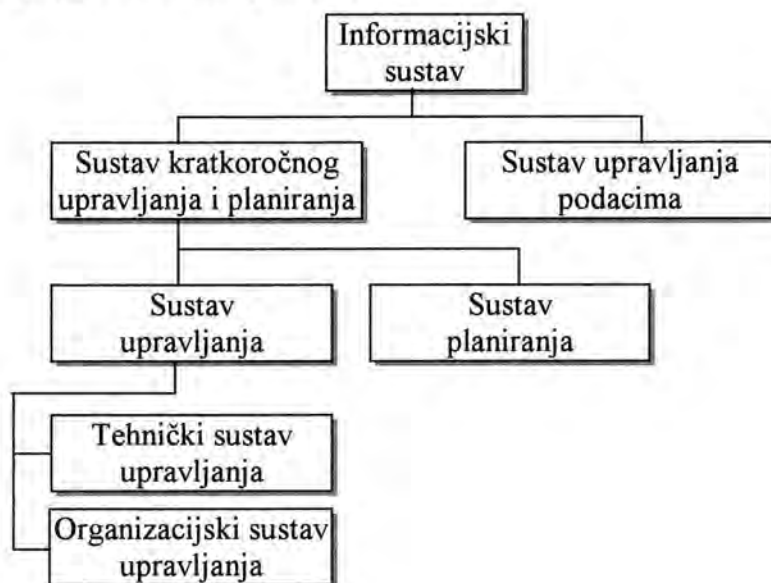
S centraliziranom opskrbom alatima izmjena alata u spremniku alata je često moguća bez prekidanja tekućeg procesa obrade na stroju. Alat se izvlači iz centraliziranog spremišta alata pomoću automatiziranog sustava za transport i rukovanje alatima. U pravilu je moguća i izmjena jednog alata. Automatizirana opskrba spremnika alata omogućava simultanu proizvodnju gotovo svih tipova radnih komada koji se mogu obrađivati u FPS-u. U ovom slučaju vremena postavljanja alata kraća su negoli u slučaju decentralizirane opskrbe. U praksi postoji opasnost da sustav opskrbe alatima postane usko grlo FPS-a u cjelini prvenstveno kod decentralizirane opskrbe alata.

2.2.2. Sustav ljudi poslužioca

Sustav ljudi poslužioca uključuje osoblje neposredno potrebno za funkcioniranje FPS-a, iako FPS može raditi i bez poslužioca u ograničenom vremenskom razdoblju. Sustav poslužioca je odgovoran za pripremu radnih komada i alata, nadzor operativnog sustava, održavanje i remont mehaničkog sustava kao i računalnog sustava za upravljanje. Trenutni standard software-a za upravljanje FPS-om također zahtijeva ručni upliv.

2.2.3. Informacijski sustav

Informacijski sustav preuzima sve funkcije potrebne za upravljanje proizvodnog procesa, slika 2.17. Može se podijeliti na sustav kratkoročnog planiranja i upravljanja i na sustav upravljanja podacima.



Sl. 2.17 Informacijski sustav

Sustav upravljanja podacima sprema i zadržava lako dostupnima sve podatke bitne za planiranje i upravljanje tokom radnih komada koji se obrađuju u FPS-u, kao što su podaci o prethodno obavljenim operacijama za svaki radni komad prisutan u FPS-u. Informacijski sustav upravlja i spremnicima alata i kontrolira preostali vijek trajanja alata. Da bi se dijagnosticirao i spriječio lom alata neophodan je i sustav automatskog nadzora procesa na radnoj stanici.

Sustav kratkoročnog planiranja i upravljanja obuhvaća sve funkcije planiranja i upravljanja koje su potrebne u svrhu dobre sinhronizacije u odvijanju procesa. Kod toga važnu ulogu igra sustav upravljanja redoslijedom odvijanja poslova u okviru zadanog radnog naloga.

Sustav upravljanja može se podijeliti na sustav tehničkog i sustav organizacijskog upravljanja. Sustav tehničkog upravljanja nadzire, između ostaloga, prijenos NC programa, upravljanje radnih komada i sustav opskrbe alatima (npr. koordinacija pozicije induktivnih transportnih vozila da bi se izbjeglo sudaranje), sinhronizaciju između upravljanja stroja i sustava transporta, i upravljanje svakog pojedinačnog stroja. Osnovna zadaća sustava organizacijskog upravljanja je sinhronizirati obavljanje svih poslova unutar zadanog radnog naloga i odgovarajuće tome sinhronizirati rad elemenata pojedinih radnih stanica, transportnih uređaja i uređaja za rukovanje.

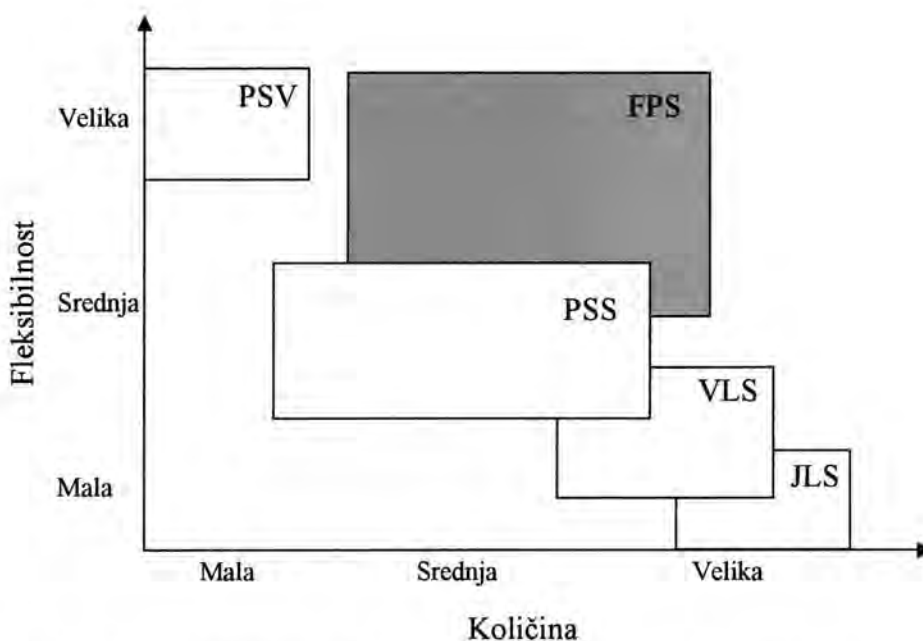
Nužnost rada takvih sustava konstantno u tri smjene, a djelomično i bez nadzora radnika, uvjetuje primjenu različitih vrsta senzora [12] i automatiziranih alatnih strojeva i ćelija sposobnih za učenje i samooptimiranje u toku rada [19] s jedne strane, te nužnost preventivnog održavanja s druge strane.

2.3. FLEKSIBILNOST KAO KARAKTERISTIKA FPS-a

Uvođenjem FPS-a omogućuje se ostvarenje brojnih gospodarskih ciljeva. Na primjer, smanjenje troškova za osoblje, povećanje iskorištenja strojeva tijekom treće smjene u kojoj se rad obavlja uz manje osoblja, te nedjeljom i praznikom. Nadalje, smanjenje vremena trajanja procesa, a time i zalihe proizvoda u procesu i pripadajući troškovi kapitala, te omogućena veća prilagodljivost kratkoročnim izmjenama. Istovremeno FPS ima i visok stupanj prilagodljivosti kod izmjena dugoročnih proizvodnih zadataka [20].

Visok stupanj prilagodljivosti (fleksibilnosti) FPS-a čini ove ciljeve mogućima. Međutim, o terminu fleksibilnosti ne možemo govoriti jednodimenzionalno; ona ima brojne aspekte. Ovaj termin nema jedinstvenu definiciju u ekonomskoj i inženjerskoj literaturi [21,22]. Fleksibilnost znači mogući stupanj prilagodljivosti određenoj situaciji. Ovakva situacija može nastati zbog ometanja tekućeg procesa ili zbog novih okolnosti u okruženju.

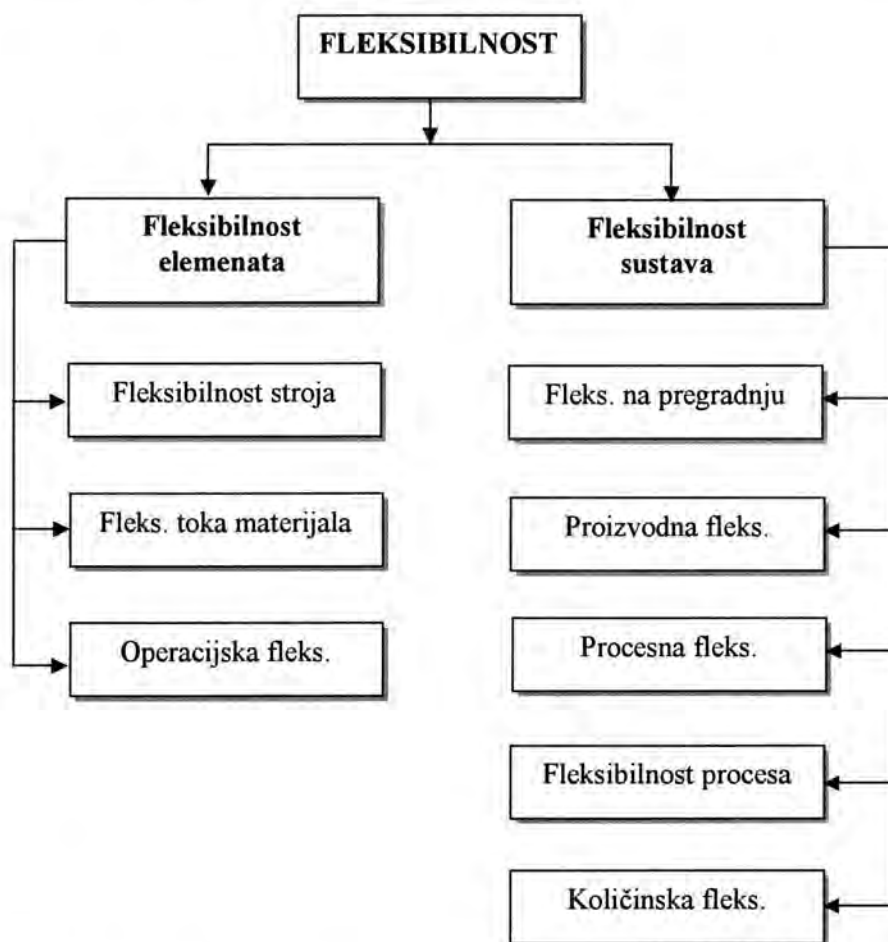
Konvencionalni modeli proizvodnih sustava kao što su proizvodni sustav prema vrsti opreme (PSV), proizvodni sustav sličnog redoslijeda korištenja (PSS) te jednopredmetni (JLS) i višepredmetni linijski sustav (VLS) imaju u osnovi različit stupanj fleksibilnosti, odnosno prilagodljivosti novim radnim zadacima, slika 2.18.



Sl. 2.18 Područje primjene i fleksibilnost konvencionalnih modela proizvodnog sustava i FPS-a

Kao što se vidi na slici 2.18, PSV je u pravilu namijenjen proizvodnji širokog asortimana različitih izradaka u manjim količinama, ima vrlo visok stupanj fleksibilnosti; PSS prilagođen toku materijala određene grupe dijelova ima ograničen stupanj fleksibilnosti dok proizvodne linije (jednopredmetne i višepredmetne) u konvencionalnoj izvedbi imaju izrazito nizak stupanj fleksibilnosti.

Kod FPS-a je cilj kombinirati prednosti konvencionalne radionice temeljene na rasporedu prema vrsti (laka promjena proizvodnog zadatka uz ograničene investicijske troškove) i efikasnost proizvodne linije (visok proizvodni kapacitet s malim jediničnim troškovima). Treba naglasiti da je FPS fleksibilan samo u okvirima one grupe proizvoda za koje je projektiran. Univerzalni FPS u okviru kojega bi se mogao proizvesti bilo koji proizvod ne postoji prvenstveno iz ekonomskih razloga. Da bi se pojasnili glavni aspekti fleksibilnosti, niže su predstavljene temeljni oblici fleksibilnosti prema dostupnoj literaturi [21,23,24].



Sl. 2.19 Oblici fleksibilnosti jednog fleksibilnog proizvodnog sustava

2.3.1. Fleksibilnost elemenata

2.3.1.1. Fleksibilnost stroja

Fleksibilnost stroja opisana je lakoćom kojom stroj može prijeći s jedne na drugu operaciju. U FPS-u se ona postiže automatskom izmjenom alata, pribora i aktiviranjem odgovarajućeg NC programa.

Fleksibilnost stroja je bitan element sposobnosti stroja, a ovisi o veličini spremnika alata [25].

2.3.1.2. Fleksibilnost toka materijala

Fleksibilnost toka materijala je sposobnost sustava opskrbe radnim komadima u FPS-u da transportira i locira radne komade različitog tipa i veličine. Što je veća fleksibilnost toka materijala bolje se strojevi mogu opskrbiti radnim komadima i na taj način fleksibilnost strojeva nije ometana ograničenjima sustava opskrbe radnim dijelovima.

Fleksibilnost toka materijala je kvaliteta sustava opskrbe radnim dijelovima. Na nju se može utjecati tehničkim rješenjima i izborom transportne putanje.

2.3.1.3. Operacijska fleksibilnost

Operacijska fleksibilnost postoji ukoliko radni komad može biti izrađen različitim tehnološkim slijedom operacija (različitim planom procesa). Što je veća operacijska fleksibilnost to je lakše balansirati opterećenje strojeva; tako se povećava produktivnost i ekonomičnost FPS-a.

Operacijska fleksibilnost vezana je za karakteristike izratka, te ju se može ostvariti generiranjem različitih planova procesa za pojedine vrste komada.

Fleksibilnost strojeva, toka materijala i operacijska fleksibilnost opisuju svojstva strojeva, sustava toka materijala i tipova radnih komada u okviru FPS-a. Svaka od navedenih fleksibilnosti tiče se samo jednog elementa FPS-a. Slijedeći tipovi fleksibilnosti se odnose na nekoliko elemenata FPS-a istovremeno.

2.3.2. Fleksibilnost sustava

2.3.2.1. Fleksibilnost na pregradnju

Fleksibilnost na pregradnju opisuje koliko i da li se FPS može proširiti ili smanjiti po pitanju broja strojeva, stanica za stezanje-otpuštanje itd. Širenje može biti neophodno zbog izmjena u zahtjevima proizvodnje. Na primjer, skup identičnih strojeva može postati usko grlo i broj im treba biti povećan. Promjena konfiguracije FPS-a također je neophodna u slučaju da se na FPS-u obrađuju novi komadi koji iziskuju novi stroj.

Ponekad, da bi se izvršila prilagodba sustava zahtjevima novog proizvoda, zamijene se komponente sustava.

Fleksibilnost na pregradnju je atribut cjelokupnog sustava. Može se osigurati, ukoliko se kod planiranja layouta FPS-a unaprijed predvide promjene (npr. ostavljanje prostora za strojeve koji će se u budućnosti ugraditi u FPS) i primjenom modularnog koncepta gradnje hardware-a i software-a.

2.3.2.2. Proizvodna fleksibilnost

Proizvodna fleksibilnost opisuje mogućnost prepremanja sustava. Proizvodna fleksibilnost ovisi o spektru različitih proizvoda koji se mogu proizvoditi u FPS-u bez dodatnih investicija u strojeve.

Proizvodna fleksibilnost je sposobnost svih komponenti FPS-a. Na nju utječe stupanj do kojega su strojevi identični (zamjenski) ili različiti (dopunjujući). FPS-i s

različitim strojevima manjeg su potencijala proizvodne fleksibilnosti od FPS-a koji se sastoje prvenstveno od identičnih strojeva.

2.3.2.3. Procesna fleksibilnost

Procesna fleksibilnost je termin koji opisuje kratkoročnu fleksibilnost FPS-a, kada se ne dozvoljava širenje ili nova preopremanja FPS-a. Procesna se fleksibilnost tiče seta različitih tipova radnih komada, koji se mogu proizvesti u FPS-u bez bitnih preopremanja. Svi komadi koji se mogu proizvesti bez preopremanja pojedinih radnih mjesta mogu odmah i biti proizvedeni. Kod visoke procesne fleksibilnosti moguće je proizvoditi različite izratke uz kontinuiran dotok na montažu što omogućuje proizvodnju s manjim zalihama i kraćim rokovima isporuke.

Procesna je fleksibilnost, primarno, karakteristika strojeva i njihovih alata u sustavu ali i izradaka.

2.3.2.4. Fleksibilnost procesa

Fleksibilnost procesa se iskazuje, kada radni komad u proizvodnji prema zadanom planu procesa slijedi različite putanje u FPS-u. Ovo je slučaj kada se nekoliko identičnih strojeva može koristiti alternativno za izvođenje operacija. Naročito je važna kada dođe do kvara stroja.

Fleksibilnost procesa je vezana za strojeve u FPS-u. Ovisi, posebno, o broju identičnih strojeva i o tome kako brzo se mogu opremiti željenim alatom.

2.3.2.5. Količinska fleksibilnost

Količinska fleksibilnost opisuje sposobnost FPS-a da radi ekonomično na različitim razinama ukupne proizvodnje pojedine vrste izradaka. FPS visokog stupnja količinske fleksibilnosti pokazuje ekstremno položenu krivulju troškova po radnom komadu kod različitih količina.

Količinska fleksibilnost je vezana za FPS u cjelini.

2.4. TENDENCIJE U RAZVOJU FPS-a

Prvi fleksibilni proizvodni sustavi pojavili su se u Japanu i SAD, a nedugo zatim i u Njemačkoj već kasnih 1960-tih godina. Analize stanja [8,26,27,28,29] pokazuju da su bili kompleksni i s relativno malom fleksibilnošću za kompletnu obradu, a porast produktivnosti pokušalo se ostvariti na temelju automatske opskrbe alatom i izracima tijekom obrade, uz integrirani tok informacija u okviru nadzorno-upravljačkog sustava .

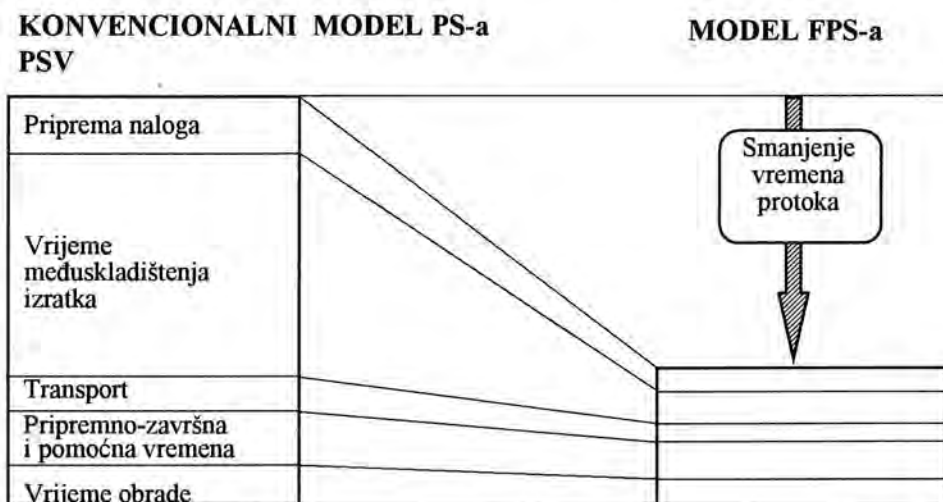
Unatoč činjenici, da je na primjer u razvijenoj industriji metaloprerade SR Njemačke u 1987. godini prema [30] udio opreme sa NC – upravljanjem iznosio svega 8 % od ukupne proizvodne opreme, tendencije u razvoju i sve veća zastupljenost fleksibilnih proizvodnih struktura temeljenih na numeričkom upravljanju očite su i vrlo izražene.

Čimbenici, koji najizraženije utječu na povećanje zastupljenosti novih proizvodnih struktura – FPC i FPS-a su pored ostalih [31]:

- potreba za povećanjem fleksibilnosti,
- zahtjev za smanjenjem vremena protoka materijala kroz proizvodni proces.

Na temelju opsežne analize primjene FPS-a, a na osnovi iskustva korisnika prema [31,32,33,34,35] može se zaključiti, da se na njima ne obrađuju jednostavni već složeni i skupi dijelovi. To uvjetuje neophodnost standardizacije, posebno u području alata, jer bi inače prevelik broj njihovih vrsta uzrokovao neekonomičnosti u radu.

Fleksibilni proizvodni sustavi zbog visoke cijene često rade u tri smjene ostvarujući visok stupanj vremenskog iskorištenja [33], pa im je raspoloživost radnog vremena veća nego kod konvencionalnih proizvodnih struktura, kao i ostalih fleksibilnih struktura. No time je intenzivnije trošenje opreme pa je vrlo važna dobra organizacija funkcije održavanja, koja je nužna i zbog potrebe osiguranja što boljeg efektivnog korištenja FPS-a, kao bitne pretpostavke isplativosti velikih ulaganja u njegovu izgradnju.



Sl. 2.20 Smanjenje ciklusa obrade primjenom FPS-a

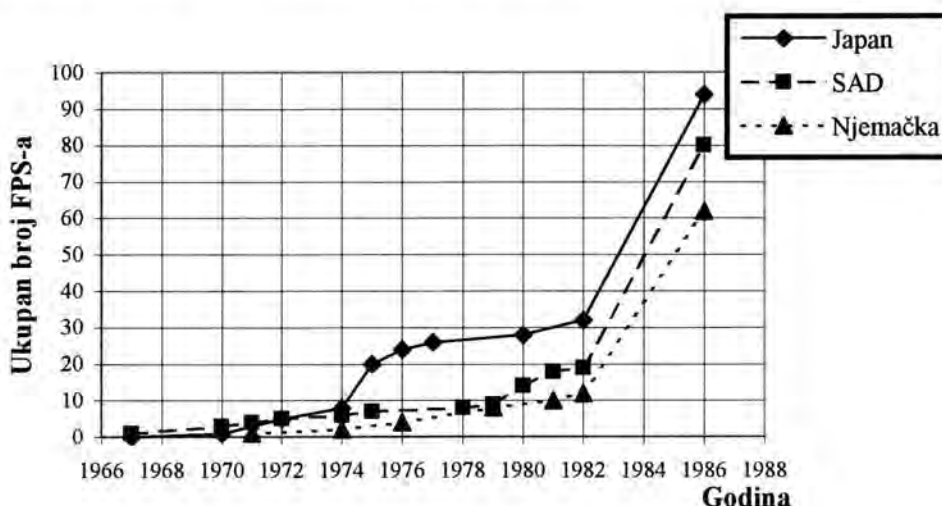
Bitno različiti uvjeti vođenja proizvodnog procesa kod FPS-a u odnosu na odgovarajuće konvencionalne proizvodne strukture rezultiraju velikim smanjenjem vremena protoka materijala kroz proces [36], te velikim promjenama u strukturi vremena pojedinih događaja, slika 2.20.

Razni su autori proveli istraživanja efekata primjene novih struktura na različitim uzorcima, pa se rezultati donekle razlikuju, ali kvalitativno svi daju istu sliku. Za ilustraciju navode se rezultati prema R. Shah-u [33] dobiveni analizom primjene 23 različita FPS-a prikazani u tablici 2.1.

Tab. 2.1 Efekti primjene FPS-a

ZNAČAJKA	PROSJEČNI EFEKTI
Ciklus izrade	↓ smanjenje za 62%
Vrijeme korištenja strojeva	↑ povećanje za 41%
Broj strojeva	↓ smanjenje za 62%
Obrtni kapital	↓ smanjenje za 45%
Broj poslužioca	↓ smanjenje za 62%
Korištenje 3. smjene, reducirano osoblje	primjena u 33% slučajeva
Troškovi obrade	↓ smanjenje za 25%
Kontrola kvalitete (obim)	↓ smanjenje za 40%

Pozitivni se efekti primjene prema [37] načelno iskazuju u povećanju proizvodnosti, povećanju kvalitete izrade i smanjenju troškova proizvodnje, skraćenju roka izrade proizvoda te povećanju iskorištenja opreme.



Sl. 2.21 Ukupan broj proizvedenih FPS-a u početnim godinama razvoja

No prisutni su i neki negativni efekti kao što su visoka investicijska ulaganja zbog zahtijevane visoke fleksibilnosti, nužna prekvalifikacija i obrazovanje kadrova [38], te velika ulaganja u reorganizaciju proizvodnje.

Uvođenje fleksibilne proizvodnje, temeljene na fleksibilnim proizvodnim ćelijama i fleksibilnim proizvodnim sustavima započela je početkom sedamdesetih godina uz relativno spor rast do 1980. godine. Nakon tog perioda započinje sve ubrzanija primjena FPS-a i FPC-a. Na slici 2.21 dat je prikaz broja instaliranih FPS-a u Japanu, SAD i SR Njemačkoj do 1986 godine [39].

Razlozi početnog sporog rasta leže u tadašnjoj relativno slaboj ponudi kao posljedici rješavanja vrlo dugačkih rokova projektiranja i realizacije uz visoke cijene, kao i filozofiji stvaranja sustava na bazi većeg broja strojeva [11].

Prema današnjem stanju i trendovima razvoja fleksibilnih proizvodnih struktura, može se konstatirati napuštanje početne ideje stvaranja velikih fleksibilnih proizvodnih sustava. Daljnji je razvoj okrenut velikoj primjeni manjih sustava [11,38,40,41], koje karakterizira manja kompleksnost, veća pouzdanost, veća preglednost i upravljivost, te lakše uključivanje u složeni proizvodni sustav, a dalje i u poslovni sustav [42].

Broj strojeva	0%	Udjeli 1990.	40%
2		27	
3		23	
4		8	
5		12	
6		10	
7		3	
8 do 9			
preko 9		17	

Sl. 2.22 Veličina sustava u ovisnosti o broju radnih jedinica (prema [42])

Vrsta strojeva	0%	Udjeli 1990.	100%
Obradni centri		69	
Brusilice		6	
Tokarilice		5	
Mjerni strojevi		5	
Uređaji za pranje		5	
Ostali strojevi		10	

Sl. 2.23 Udio vrsta alatnih strojeva u izvedenim FPS-ima (prema [42])

Prema istraživanjima provedenim 1986. godine izrazit je udio FPC i FPS-a do 2 stroja u strukturi instaliranih sustava prema veličini [11], dok istraživanja prema [42,43] pokazuju da na uzorku od 60 sustava 50 % FPS-a ima 2-3 stroja, slika 2.22.

Od primijenjenih strojeva u okviru FPS-a najčešće su zastupljeni obradni centri, slika 2.23, pri čemu se gotovo kod svih FPS-a (95%) pojavljuje barem jedan obradni centar [42].

Kao mjera fleksibilnosti realiziranih FPS-a uzima se asortiman dijelova, koji se obrađuje na jednom FPS-u (slika 2.24).

Broj dijelova / FPS	0%	Udjeli 1990.	25%
1 do 5		19	
6 do 10		8	
11 do 20		8	
21 do 50		13	
51 do 100		15	
101 do 200		13	
201 do 400		17	
preko 400		7	

Sl. 2.24 Struktura broja dijelova po jednom FPS-u, (prema [38])


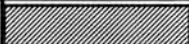

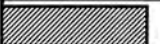
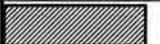
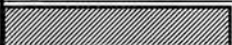

Prema stanju 1990. godine, u području od 6 do 100 različitih izradaka po sustavu nalazilo se samo 44% instaliranih sustava, što znači da se više od 50% FPS-a nalazilo izvan nekad dominirajućeg područja primjene. Očito je značajno širenje područja na FPS-e relativno visoke specijalizacije (na 19% sustava obrađuje se od 2 do 5 vrsta dijelova), ali se također na velikom broju sustava obrađuje vrlo širok spektar izradaka (na 36% sustava obrađuje se preko 100 vrsta dijelova, na pojedinim sustavima i preko 1000). Posljedica je to s jedne strane vrlo visoke fleksibilnosti, ali i sve veće produktivnosti, koje je moguće ostvariti na FPS-u. Interesantno je također upozoriti na činjenicu da je najveći broj sustava instaliran u većim poduzećima (72% od ukupnog broja instalirano je u poduzećima s više od 1000 zaposlenih).



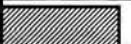



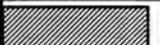

Analiza veličine pojedinačnih serija, odnosno količina, koje se obrađuju uz jednokratno preopremanje FPS-a, pokazuje tendenciju prema većim količinama, slika 2.25.

Iako FPS, zahvaljujući preopremanju sustava paralelno sa izvođenjem obrade, omogućuje rad i uz veličinu jedinične serije 1, ipak je takav način proizvodnje prisutan tek kod 5% obrađivanih dijelova. Češći su primjeri obrade većih količina. Čak u 23% slučajeva veličina jedinične serije⁷ veća je od 200 komada. Zahvaljujući visokoj fleksibilnosti prisutna je tendencija višestrukog lansiranja istovrsnih dijelova u proizvodnju. U 32% slučajeva primjene istovrsni se dijelovi obrađuju od 6 do 11 puta godišnje, dok se u 6% slučajeva oko 1990. godine radilo o prosječno svakodnevnom lansiranju pojedinih dijelova u rad, što omogućuje kontinuiran dotok dijelova na

⁷ količine po tipu dijelova u radnom nalogu




montažu. Danas je taj udio znatno veći zbog sve veće zastupljenosti JIT (just in time) filozofije.

Ponavljjanje radnog naloga / god.	0%	Udjeli 1990.	40%
1 do 2		8	
3 do 5		14	
6 do 11		32	
12 do 24		11	
25 do 48		11	
49 do 100		18	
preko 200		6	

Jedinična serija kom.	0%	Udjeli 1990.	40%
1		5	
2 do 5		10	
6 do 10		9	
11 do 20		15	
21 do 50		13	
51 do 100		13	
101 do 200		12	
preko 200		23	

Sl. 2.25 Struktura radnih naloga po jednom FPS-u (prema [42])

Kod primjene transportnih sustava tračnički transport većinom je zastupljen kod manjih sustava, a induktivna vozila kod većih FPS-a, slika 2.26. Pritom je registriran manji pad udjela induktivnih vozila na račun povećanog udjela portalnih sustava, koji su posebno pogodni za transport alata.

Vrsta	0%	Udjeli 1990.	100%
Tračnički sustavi		60	
Portalni roboti/manipulatori		30	
Induktivna vozila		10	

Sl. 2.26 Struktura primjene vrsta transportnih sustava (prema [42])

Temeljem izloženog mogu se rezimirati tendencije u razvoju gradnje i primjene fleksibilnih proizvodnih sustava:

- prevladava udio malih FPS-a
- širenje područja primjene na izradu relativno malog broja izradaka i područje velikog broja različitih izradaka
- sve veća zastupljenost obradnih centara
- ulazak u proizvodnju sa sve manjim količinama pojedine vrste dijelova čak do veličine serije 1
- veća zastupljenost tračničkih transportnih sustava kod manjih, a induktivnih vozila kod većih FPS-a.

U ovom su poglavlju kao osnova uzeti podaci koji karakteriziraju stanje primjene FPS-a u 1990. godini. U novije vrijeme ovi podaci nisu doživjeli značajnije promjene.

3. UPRAVLJANJE RADNIM NALOZIMA NA FPS-U

Složenost tržišnih zahtjeva prvenstveno se ogleda u potrebi za sve većom kvalitetom proizvoda koji postaju i sve kompleksniji, a uz to je često potrebno udovoljiti specifičnim zahtjevima kupaca uz zadržavanje povoljne cijene proizvoda. Uz i ovako zahtjevna očekivanja, tržište nameće i raznolikost naručivanih količina koje variraju od velikih do vrlo malih serija, pa čak i pojedinačne izrade. Zahtjevnost tržišta ogleda se i u traženju visoke stabilnosti održavanja rokova isporuke te u njihovom izuzetnom vremenskom skraćanju. Veliki broj novorazvijenih dijelova nekog proizvoda, kao i dinamičnost izmjena postojećih dijelova tijekom vijeka proizvodnje, predstavlja znatan problem proizvodnom ovladavanju takvog asortimana.

Težnja za smanjenjem troškova proizvodnje i radom s minimalnim zalihama, kao i zahtjevi suvremenog tržišta uvjetuju potrebu da se u proizvodnji lansiraju sve manje količine izradaka. Cilj je istraživanja omogućiti simultanu proizvodnju dijelova na FPS-u s različitim kombinacijama tipova dijelova i količina i s mogućnošću prilagodbe toka materijala prevladavajućim pogonskim uvjetima. Prilagodba toka materijala potrebna je prvenstveno zbog balansiranja opterećenja pojedinih radnih stanica proizvodnog sustava pri proizvodnji prema određenom radnom nalogu. S druge strane, promjena toka materijala nekad je potrebna radi privremenog otkazivanja nekog stroja sustava zbog kvara, što je od izuzetne važnosti kod proizvodnje dijelova na FPS-u s različitim strojevima (identični strojevi ali različito opremljeni s alatima ili različiti strojevi po izvedbi i opremi) jer u suprotnom ispad stroja iz sustava vodi do kompletnog ispada sustava pri proizvodnji pojedinih dijelova.

Temeljna pretpostavka za prilagodbu odvijanja procesa je uz potrebnu fleksibilnost jezgre proizvodnog sustava (alatni strojevi, strojevi za pranje, strojevi za kontrolu, stanice za punjenje/pražnjenje paleta, automatski transportni sustav) postojanje alternativnih planova procesa, kojima se predviđa izrada pojedinih tipova dijelova uz različit broj operacija s različitim korištenjem strojeva. Postojanje alternativnih planova procesa povećava fleksibilnost proizvodnje i učinkovitost proizvodnog sustava pri paralelnoj proizvodnji promjenjivog skupa dijelova. U stalno promjenjivom proizvodnom okruženju formiraju se radni nalozi različitog sadržaja po tipovima i količinama pojedinih izradaka. Cilj je istraživanja formirati parcijalni zadatak⁸ s takvom strukturom radnog naloga koja će omogućiti odvijanje proizvodnje uz:

- najmanje vremensko opterećenje sustava kao cjeline, odnosno najmanjeg ciklusa izrade radnog naloga
- izbalansirano opterećenje pojedinih radnih mjesta
- što manje vremensko opterećenje automatiziranog transportnog sustava
- dobru sinhronizaciju rada osoblja i korištenja alata

Minimizaciju ukupnog vremena čekanja koje osigurava najmanje vremensko opterećenje sustava moguće je provesti odgovarajućim upravljanjem redoslijednog odvijanja pojedinog radnog naloga odnosno operacija parcijalnog zadatka.

⁸ detaljnija formulacija u podpoglavlju 3.3

Uspješno rješavanje problema optimizacije ukupnog toka odvijanja procesa u okviru parcijalnog zadatka jedna je od ključnih podloga za ostvarenje uvjeta za visoku ukupnu iskoristivost FPS-a, što je temeljni preduvjet za gospodarstvenu isplativost velikih ulaganja vezanih za njihovu izgradnju i korištenje.

Upravljanje nalogima pri proizvodnji različitih proizvoda s različitim brojem i trajanjem operacija s različitim redoslijedom korištenja pojedinih radnih mjesta u proizvodnom sustavu, s ciljem stvaranja što kraćeg vremenskog zauzeća proizvodnog sustava, predstavlja jedan od najsloženijih, ali i čestih kombinatornih optimizacijskih problema.

Najbolja ekonomičnost pri proizvodnji dijelova prema određenom radnom nalogu bit će postignuta pri najmanjem vremenskom opterećenju sustava kao cjeline, odnosno pri najmanjem ciklusu izrade radnog naloga.

Obzirom da je minimizacija ciklusa izrade na FPS-u glavni predmet zanimanja ovog rada, detaljnija formulacija tog pojma dat će se u sljedećem podpoglavlju.

Ciklus izrade svakog dijela sastoji se iz vremena operacija, vremena transporta i vremena na ulaznim i izlaznim redovima čekanja. Međutim ukupno vrijeme operacija i vrijeme transporta ovisi o planovima procesa pridruženim pojedinom dijelu iz radnog naloga. Rezultat pridruživanja ovisit će o tipovima dijelova i količinama u aktualnom radnom nalogu, karakteristikama proizvodnog sustava i razvijenim alternativnim planovima procesa, koje obilježava redoslijed korištenja i vremena zauzeća pojedinih strojeva.

Primjena alternativnih planova procesa kao jedne od temeljnih pretpostavki za prilagodbu toka odvijanja procesa značajkama radnog zadatka i značajkama jezgre FPS-a obradit će se u podpoglavlju 3.2. Tu će se definirati matematički model izbora planova procesa. U ovom podpoglavlju predstaviti će se postupak izbora planova procesa koji se temelji na osnovnim postavkama metode Seo i Egbelu, ali s modificiranim pristupom u fazi vrednovanja pojedinih rješenja.

U podpoglavlju 3.3. razmotrit će se problematika terminiranja radnih naloga i u tu svrhu razmotriti sustav u kojem se proizvode dijelovi s različitim kombinacijama tipova dijelova količine 1. Za svaki izradak podrazumijeva se da je poznato vrijeme trajanja pojedine operacije na datom proizvodnom kapacitetu, te redoslijed odvijanja pojedinih operacija na određenim strojevima datog radnog naloga. Taj pojednostavljeni tip problema terminiranja radnih naloga na FPS-u važan je za postavku daljnje strategije ovog rada i usporedbe rezultata dobivenog software-a.

Problem terminiranja radnih naloga na FPS-u je među najtežim kombinatornim problemima optimizacije [44]. Posljednjih je godina ponovno oživio interes za lokalnom potragom za rješenjem na temelju razvoja probabilističkih metoda kao što su: simulirano približavanje [45], tabu tehnika pretraživanja [46], metoda premještanja uskog grla [47,48,49] i mnoge druge.

Također, u podpoglavlju 3.3. dat će se kratki osvrt na heuristička pravila upotrebljena kod terminiranja radnih naloga. Ovo je podpoglavlje važno za razumijevanje rezultata o kojima se govori u istraživačkim radovima kao i u ovom radu. Mnogo je studija o heurističkim pravilima u literaturi te se za detaljnije upoznavanje upućuje na navedene reference.

3.1. CIKLUS IZRADE

Brzina protoka proizvoda kroz proizvodni proces jedna je od najbitnijih karakteristika procesa. Stoga se kao predmet stalnog interesa i istraživanja u radovima raznih autora kao pokazatelji brzine protoka navode ciklus proizvodnje, ciklus izrade, vrijeme protoka itd.

Ciklus proizvodnje podrazumijeva kalendarski odsječak vremena u kojem se odvija proizvodni proces [50] uključujući i prekide (neradne smjene i neradni dani, zastoji zbog kvarova opreme itd...) koji nisu uvjetovani tehničkim i organizacijskim postavkama procesa [13]. Ciklus proizvodnje uključuje ciklus izrade i vremena čekanja.

Dok pri formulaciji pojma ciklusa proizvodnje postoji visoka podudarnost, definiciji ciklusa izrade razni autori pristupaju na različite načine, pa se npr. prema [51] ciklus izrade odnosi na određenu količinu izradaka, a kod nekih drugih autora [13,52,53,54] svodi se na jedan izradak.

U daljnjim razmatranjima pod ciklusom izrade podrazumijevat će se ukupno vrijeme izvođenja svih operacija (aktivnosti), kao i trajanje tehnički i organizacijski uvjetovanih prekida pri proizvodnji jednog izratka.

Ciklus je izrade ona karakteristika, koja daje pravu sliku tehnoloških i organizacijskih rješenja pri postavci i vođenju procesa u određenom proizvodnom sustavu i najbolje karakterizira tok materijala.

Prema [54,55] ciklus izrade t_{cj} može se izraziti kao:

$$t_{cj} = \sum_{i=1}^n t_i \quad (3.1)$$

gdje je: t_{cj} - ciklus izrade j -tog izratka
 t_i - vrijeme trajanja i -tog događaja (aktivnosti ili zastoja) tijekom ciklusa izrade j -tog izratka
 $i = 1, 2, 3, \dots, n$ - broj aktivnosti ili zastoja

Gornji se izraz može dalje razviti prema vrstama događaja [56,57] :

$$t_{cj} = t_{kj} + t_{tj} + t_{mj} \quad (3.2)$$

gdje je: t_{kj} - ukupno vrijeme izvođenja svih radnih i kontrolnih operacija pri izradi j -tog izratka
 t_{tj} - ukupno vrijeme svih transporata tijekom procesa izrade j -tog izratka
 t_{mj} - ukupno vrijeme svih planiranih međuskladištenja tijekom vođenja procesa j -tog izratka

Daljnjim razvijanjem određenih vremena pojedinih vrsta događaja [58,59] dolazi se do općeg izraza za ciklus izrade :

$$t_{cj} = \sum_{r=1}^n t_{rj} + \left(t_{uj} + \sum_{s=1}^k t_{sj} + t_{ij} \right) + \sum_{s=1}^k t_{rj} \cdot (q_{sj} - 1) \quad (3.3)$$

- gdje je :
- t_{rj} - vrijeme r -te operacije pri izradi dijela j
 $r = 1, 2, 3, \dots, n$
 - t_{uj} - vrijeme transporta od skladišta repromaterijala do prve operacije pri izradi dijela j
 - t_{sj} - vrijeme s -tog međuoperacijskog transporta pri izradi j -tog izratka
 - t_{ij} - vrijeme transporta od zadnje operacije do skladišta gotovih proizvoda pri izradi dijela j
 $s = 1, 2, 3, \dots, k$
 - q_{sj} - količina j -tih izradaka u s -tom međuoperacijskom transportu poslije r -te operacije

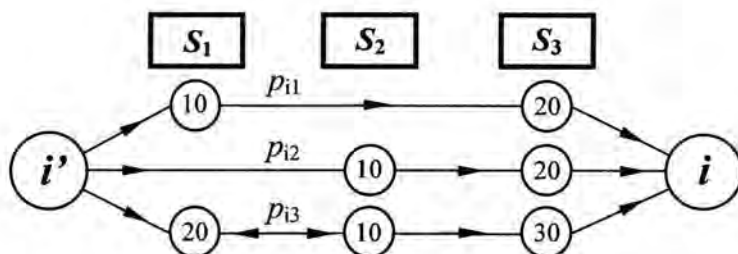
Vidljivo je da na ciklus izrade utječu različiti elementi. To su u prvom redu tehnološka rješenja izražena kroz vremena operacija ($\sum t_{rj}$), ali i elementi ovisni o prostornoj strukturi proizvodnog sustava. To su vremena transporta, u direktnoj zavisnosti o rasporedu radnih mjesta i redosljedu izvođenja operacija ($\sum t_{sj}$), te udaljenosti skladišta repromaterijala (t_{uj}) i skladišta gotovih proizvoda (t_{ij}). Od značaja je, naravno i način izvođenja transporta.

S druge strane, to su zastoji (od završetka prethodnog događaja do početka narednog događaja) [52] zavisni o načinu vođenja proizvodnog procesa odnosno o količini izradaka, koja se odjednom treba transportirati između radnih mjesta ($\sum t_{rj} \cdot (q_{sj} - 1)$).

Jedan je od osnovnih ciljeva rada postizanje što kraćeg ciklusa izrade jer njegovim skraćanjem smanjujemo zalihe a time i troškove, pri čemu se osjetno povećava i upravljivost sustava.

3.2. ALTERNATIVNI PLANOVI PROCESA

Alternativni planovi procesa omogućuju različite putove pretvorbe ulaznog materijala i' u dio određenih konstrukcijskih karakteristika i , kao što je prikazano na slici 3.1.



Sl. 3.1 Alternativni planovi procesa za izradu dijela

Iako će od razvijenih planova procesa jedan osiguravati u statičkim uvjetima najekonomičniju proizvodnju određenog dijela, kod promijenjenog mixa dijelova i drugačijih količina, njegova primjena može biti neprimjerena zbog prevelike razlike u ukupnom opterećenju pojedinih radnih mjesta sustava, odnosno zbog preopterećenja transportnog sustava. U realnom okruženju koje redovito vodi različito strukturiranim radnim nalogima za proizvodnju na istom određenom proizvodnom sustavu, postoji zbog interaktivnog djelovanja, potreba za identifikaciju i izbor one kombinacije između razvijenih planova procesa, koja će osigurati optimalnu proizvodnju svih tipova dijelova u zadanim količinama prema konkretnom radnom nalogu.

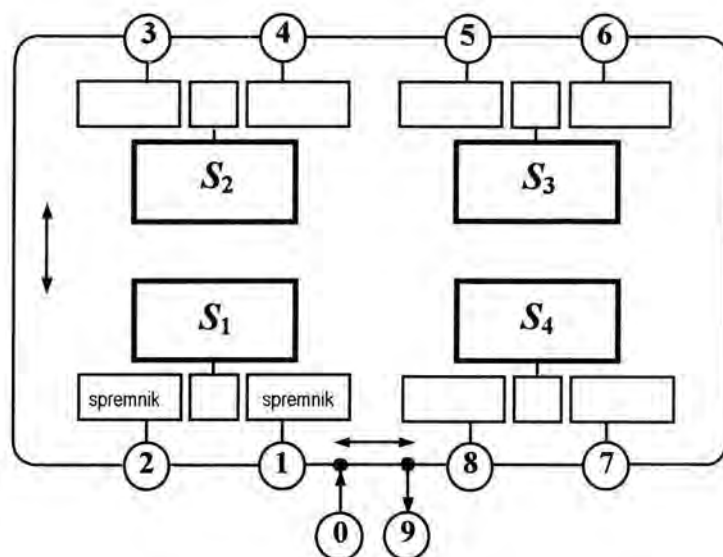
Primjena alternativnih planova procesa i problem izbora plana procesa obrađeni su u radovima niza autora (Baskaran 1990., Kusiak i Finke 1988., Seo i Egbelu 1996.). Seo i Egbelu formulirali su matematički model izbora plana procesa s optimizacijom procesnih i transportnih varijabli i razvili metodu rješavanja problema uz primjenu poznate tabu tehnike [60,61]. Rješavanje problema temelji se na zahtjevu minimalnog vremenskog opterećenja obradnog i transportnog sustava uz određivanje dozvoljenog graničnog vremenskog opterećenja pojedinih strojeva i transportnog sustava [62,63].

U sljedećem podpoglavlju prikazat će se modificirani model izbora planova procesa koji se temelji na osnovnim postavkama metode Seo i Egbelu [64,65].

3.2.1. Model izbora planova procesa

Fleksibilni proizvodni sustav pretstavlja jedinstvenu cjelinu sastavljenu iz jezgre sustava, perifernih elemenata (skladišta alata, skladišta poluproizvoda, skladišta gotovih dijelova, ...) i vodećeg računala, slika 3.2. Polazišta pri razvoju metode izbora planova procesa su:

- što manje ukupno vremensko opterećenje radnih stanica
- što manja razlika vremenskog opterećenja radnog mjesta koje predstavlja usko grlo i ostalih radnih mjesta sustava pri proizvodnji prema radnom nalogu
- što manje vremensko opterećenje automatiziranog transportnog sustava



Sl. 3.2 *Fleksibilni proizvodni sustav s 4 radne stanice i automatskim transportnim sustavom*

Razvoj modela izbora planova procesa temelji se na slijedećim varijablama i parametrima:

- i – i -ta vrsta dijelova određenog radnog naloga ($i=1,2, \dots, n$)
- j – j -ti plan procesa ($j=1,2, \dots, r_i$)
- k – k -ti stroj ($k=1,2, \dots, m$)
- q – skup količina dijelova radnog naloga za simultanu proizvodnju $\{q_1, q_2, \dots, q_n\}$
- p_{ij} – j -ti plan procesa za dio vrste i
- t_{ijk} – vrijeme operacije na stroju k u planu procesa p_{ij} ,

$$x_{ij} = \begin{cases} 1, & \text{ako je plan procesa } p_{ij} \text{ izabran} \\ 0, & \text{u ostalim slučajevima} \end{cases}$$

Plan procesa p_{ij} može se predstaviti uređenim skupom grupa parametara. Pojedina grupa sadrži parametre koji se odnose na jednu operaciju: ulazni segment putanje i vrijeme operacije. Dodatnim parametrom definira se izlazni segment putanje nakon izvršenja posljednje operacije. Čvorne točke putanje dijela predstavljaju: izlazna stanica skladišta materijala (a), ulazne i izlazne stanice strojeva, ulazna stanica skladišta dijelova (s), tako da slijedi:

$$p_{ij} = \{(d_{ab}, t_{ijk}), \dots, d_{rs}\} \quad (3.4)$$

gdje je:

- d_{ab} - najkraći razmak između čvornih točaka a i b , koji treba prevaliti paleta s izratkom prije prve operacije;
- d_{rs} - najkraći razmak između čvornih točaka r i s , koji treba prevaliti paleta s izratkom nakon izvršenja posljednje operacije.

m_{ij} – vrijeme izrade q_i dijelova vrste i prema planu procesa p_{ij}

$$m_{ij} = q_i \sum_{k=1}^m t_{ijk} \quad (3.5)$$

h_{ij} – vrijeme transporta q_i dijelova vrste i prema planu procesa p_{ij}

$$h_{ij} = \frac{q_i}{v_p} \sum_{ab}^{rs} d_{ij} \quad (3.6)$$

Ovdje je pretpostavljeno, da se istovremeno kreće brzinom v_p samo jedna paleta s jednim izratkom.

T_k – opterećenje k -tog stroja pri proizvodnji svih dijelova radnog naloga

$$T_k = \sum_{i=1}^n (q_i t_{ik}), \quad \forall k \quad (3.7)$$

T_t – opterećenje transportnog sustava pri proizvodnji svih dijelova radnog naloga prema odabranim planovima procesa

$$T_t = \frac{1}{v_p} \sum_{i=1}^n \left(q_i \sum_{ab}^{rs} d_{ij} \right) \quad (3.8)$$

Za postizanje optimalnih rezultata važno je izbalansirati opterećenje radnih mjesta sustava. Mjera nejednakomjernosti opterećenja izražena je veličinom Δm kao razlikom između opterećenja najopterećenijeg stroja i prosjeka opterećenja ostalih radnih mjesta sustava, prema izrazu:

$$\Delta m = \frac{T_{k_{\max}} m - \sum_{i=1}^m T_k}{m-1} \quad (3.9)$$

Isto tako bitno je, da transportni sustav ne uzrokuje čekanja na trenutnom uskom grlu proizvodnog sustava, odnosno da eventualna čekanja na uskom grlu budu što manja. Utjecaj transportnog sustava na rad proizvodnog sustava iskazan je veličinom Δh na način:

$$\bullet \text{ ako je } T_t \leq T_{k_{\max}}, \text{ tada je } \Delta h = 0 \quad (3.10)$$

$$\bullet \text{ ako je } T_t > T_{k_{\max}}, \text{ tada je } \Delta h = T_t - T_{k_{\max}} \quad (3.11)$$

3.2.2. Postavljanje funkcije cilja

Pri izboru plana procesa cilj je osigurati minimalne troškove, koji se mogu iskazati minimalnim vremenskim opterećenjem proizvodnog sustava kod proizvodnje

prema radnom nalogu uz što bolju balansiranost opterećenja radnih mjesta i automatskog transportnog sustava. U tom smislu funkcija cilja može se izraziti kao:

$$\varphi = \left[\sum_{i=1}^n \sum_{j=1}^{r_i} (m_{ij} + h_{ij}) x_{ij} + \Delta m + \Delta h \right], \quad (3.12)$$

Zadatak je optimizacije minimizirati funkciju cilja φ uz uvjet da je:

$$\sum_{j=1}^{r_i} x_{ij} = 1, \quad \forall i = 1, 2, 3, \dots, n \quad (3.13)$$

$$x_{ij} \in \{0, 1\}, \quad \forall x_{ij} \quad (3.14)$$

Datim ograničenjima osiguran je izbor samo jednog plana procesa za svaki dio, uz mogućnost izbora bilo kojeg od razvijenih planova procesa.

3.2.3. Postupak izbora planova procesa

Pretraživanje svih mogućnosti i pronalaženje optimalnog rješenja kod povećanog broja dijelova i broja alternativnih procesa za pojedini dio, iziskuje velik obim aktivnosti. Područje rješenja definirano je izrazom $\prod_{i=1}^n r_i$, gdje je r_i broj razvijenih procesa za dio i . Prikladnim heurističkim algoritmom postupak je učinjen praktički primjenjivim kod raznih struktura radnih zadataka. U tu svrhu korištena je tabu tehnika pretraživanja, kojom se znatno skraćuje postupak izbora skupa planova procesa u koji ujedno pretstavlja rješenje problema:

$$u = \{p_1, p_2, p_3, \dots, p_i, \dots, p_n\}, \quad (3.15)$$

gdje je p_i izabrani plan procesa za dio i , a cijeli je broj između 0 i $(r_i - 1)$.

Za provedbu postupka potrebno je svakom pretraživanom rješenju u pridružiti kodnu oznaku $C(u)$, koja pretstavlja jedinstveni broj u području rješenja. Kôd rješenja dobiven je zbrajanjem pozicijskih vrijednosti pojedinih izabranih planova procesa. Pri tom je pozicijska vrijednost izabranog plana procesa p_i za izradak i jednaka umnošku broja izabranog plana procesa p_i (0 do $r_i - 1$) i ukupnih brojeva varijantnih procesa slijedećih dijelova ($i + 1$ do n) uređenog skupa. Za izradak n pozicijska vrijednost jednaka je broju izabranog procesa p_n . Sukladno tome, kôd rješenja u dat je izrazom:

$$C(u) = \sum_{i=1}^{n-1} p_i \prod_{i+1}^n r_i + p_n \quad (3.16)$$

Mjera povoljnosti rješenja u iskazana je u skladu s funkcijom cilja kao vrijednost funkcije vrednovanja $f(u)$ prema (3.17):

$$f(u) = \sum_{i=1}^n q_i c_i + \Delta m(u) + \Delta h(u), \quad (3.17)$$

gdje pretstavlja $f(u)$ – vrijednost funkcije vrednovanja
 c_i – vrijeme izrade i transporta dijela i prema odabranom planu procesa u u rješenju u

$\Delta m(u)$ i $\Delta h(u)$ – utjecaj nejednakomjernosti opterećenja strojeva, odnosno utjecaj opterećenja automatskog transporta na vrijednost funkcije u rješenju u .

Algoritam pretraživanja starta od rješenja u_0 čiji su $p_i=0$ za sve $i \in n$ i utvrđuje vrijednost rješenja izračunom funkcije $f(u_0)$. Rješenje u_0 ujedno je početno najbolje trenutno rješenje u_c . U svakom narednom koraku utvrđuje se najprihvatljivije rješenje iz skupa rješenja $\sigma(u_c)$ izvedenog iz trenutno najboljeg rješenja u_c dobivenog u prethodnom koraku i utvrđuje kao novo trenutno najbolje rješenje. Najprihvatljivije rješenje je ono, koje ima najmanju vrijednost funkcije vrednovanja uz uvjet, da nije na tabu listi T . Sva tako razmatrana rješenja stavljaju se na tabu listu i isključuju iz ponovnog izbora i razmatranja u narednim koracima. Ovo ograničenje nije na snazi, ukoliko je vrijednost trenutno najboljeg rješenja $f(u_c)$ manja od vrijednosti $f(u_c)$ u prethodnom koraku, koja se kao vrijednost težnje a_L pridružuje rješenju u_c s kodnom oznakom $C(u_c)$ na tabu listi.

Skup rješenja $\sigma(u_c)$ generiran je u određenom koraku iz trenutno najboljeg rješenja u_c primjenom jedne od metoda. Metoda naprijed/nazad rezultira s najviše $2n$ novih rješenja. U svakom novom rješenju variran je izbor plana procesa za jedan od dijelova radnog naloga, birajući susjedni plan ispred ili iza od trenutnog rješenja. Metoda uparene izmjene rezultira s $\sum_{i=1}^n (r_i - 1)$ novih rješenja. U svakom novom rješenju variran je za jedan dio izbor između ostalih razvijenih planova procesa p_i .

3.3. REDOSLIJED ODVIJANJA RADNOG NALOGA

Skup izradaka koji predstavljaju ukupnu populaciju dijelova za izradu na određenom FPS-u ulazi u proizvodnju u podskupovima. Takve podskupove čine dijelovi definirani radnim nalogom čije odvijanje izrade na sustavu teče usporedo tijekom određenog vremenskog perioda. Obzirom na različite i promjenjive zahtjeve tržišta, skup dijelova koji predstavljaju ukupnu populaciju dijelova je varijabla promjenjiva tijekom vremena, ali se može smatrati nepromjenjivom u okviru određenog vremenskog proizvodnog razdoblja.

Sastav dijelova i količina u pojedinom radnom nalogu za određeni vremenski period montaže (1-dan, x-dana) proizlaze iz plana montaže, koji polazi od ugovorenih količina i rokova isporuke finalnih proizvoda i tehničke spremnosti sustava odnosno, kod planiranja rada sustava određuje se sastav i količina pojedinog podskupa dijelova/parcijalni zadatak/radni nalog ili operativni plan FPS-a na osnovi podataka o:

- hitnosti pojedinog radnog naloga montaže
- potrebnim *proizvodnim sredstvima*⁹; po vrsti i potrebnom vremenu korištenja
- raspoloživim proizvodnim sredstvima; po vrsti i raspoloživom vremenu korištenja
- kriterijima za izbor naloga

vodeći računa o potrebi postizanja glavnih ciljeva:

- održavanje rokova isporuke
- maksimalno iskorištenje proizvodnog sustava
- minimalne ukupne zalihe
- što veća brzina protoka
- maksimalni gospodarski efekti

Pretpostavka za dobro iskorištenje proizvodnog sustava je formiranje parcijalnog radnog zadatka s povoljnom strukturom radnih naloga. Povoljna struktura radnih naloga podrazumjeva skup dijelova, za čiju će proizvodnju trebati svi obradni strojevi sustava s njihovim približno podjednakim vremenskim opterećenjem uz najmanji utrošak vremena i dobru sinhronizaciju rada osoblja i korištenja alata. Dobro iskorištenje FPS-a, kao jedan od postavljenih glavnih ciljeva, ovisi o nizu utjecajnih faktora od kojih se mogu istaći:

- tehnička i organizacijska rješenja primijenjena pri oblikovanju i radu sustava, koja se iskazuju kao njegova tehnička raspoloživost
- značajke parcijalnog radnog zadatka sa stajališta broja dijelova, količina, broja operacija, strojnih vremena i vremena preopremanja radnih mjesta ovisno o redoslijedu provođenja
- redoslijedu odvijanja pojedinih operacija parcijalnog radnog zadatka

Budući da se na strukturu radnih naloga može samo ograničeno utjecati zbog primarnog utjecaja prioriteta, koji diktira plan montaže, uz postojeća tehnička i

⁹ *Proizvodna sredstva su skupni pojam za strojeve, pogonska sredstva, alate, palete za izratke i druga za izvođenje operacija potrebna sredstva.*

organizacijska rješenja sustava ostaje redosljed odvijanja poslova u okviru radnog naloga kao primarni element o kojem ovisi stupanj iskorištenja FPS-a.

U nastavku će se posebno razmotriti utjecaj redosljeda odvijanja operacija, koji se određuje u okviru upravljanja odvijanja rada na FPS-u. Kod toga svaka operacija predstavlja zasebni element odvijanja, koji u proces upravljanja ulazi s podacima o:

- okončanju prethodne aktivnosti;
- izboru stroja čija je priprema za izvođenje operacije obavljena tijekom aktivnosti preopremanja;
- trenutku početka operacije;
- vremenu trajanja operacije.

Između više izradaka, koji u određenom trenutku čekaju na izvođenje operacije sustav upravljanja treba koristeći fleksibilnost strukture odabrati onaj, koji će osigurati angažiranje prvog slobodnog obradnog stroja, ali na način koji daje najveće potencijalne mogućnosti korištenja strojeva pri daljnjim izborima.

Pri pridruživanju operacije radnom mjestu postoje ograničenja, koja proizlaze iz:

- broja obradnih strojeva u sustavu te stupnja njihove identičnosti/različitosti sa stajališta tehničkih značajki i opremljenosti alatima i priborom;
- obilježja grupe operacija parcijalnog zadatka FPS-a.

Da bi se postavljeni problem uspješno rješio, cilj je rada postaviti adekvatnu strategiju izbora grupe operacija i pridruživanja pojedine operacije radnom mjestu tj. provesti optimizaciju redosljeda ulaska pojedinih dijelova u proces, koja će se u ovom radu obaviti primjenom genetskih algoritama. U nastavku rada uvest će se pojednostavljen model problema terminiranja radnih naloga važan za postavku daljnje strategije ovog istraživanja uz pregled nekih od postojećih modela rješavanja.

3.3.1. Klasični TRN¹⁰ model

Klasični model terminiranja radnog naloga može se postaviti na sljedeći način [66]: n različitih izradaka/dijelova treba se obraditi na m zadanih strojeva. Svaki se plan izrade pojedinog izratka sastoji od skupa operacija s čvrstim predodređenim redosljedom. Za svaku operaciju dat je stroj na kojem je operaciju moguće izvesti i vrijeme trajanja operacije na datom stroju. Pri pridruživanju operacije radnom mjestu postoje sljedeća ograničenja:

- niti jedna operacija ne može se odvijati na više od jednog stroja istovremeno
- ne postoji ograničenje zbog prioriteta među operacijama različitih izradaka
- operacija se ne može prekinuti
- na svakom stroju sustava može se izvršavati samo jedna operacija u jedinici vremena
- rokovi isporuke proizvoda kao ni ciklus proizvodnje nisu unaprijed određeni

¹⁰ terminiranje radnog naloga

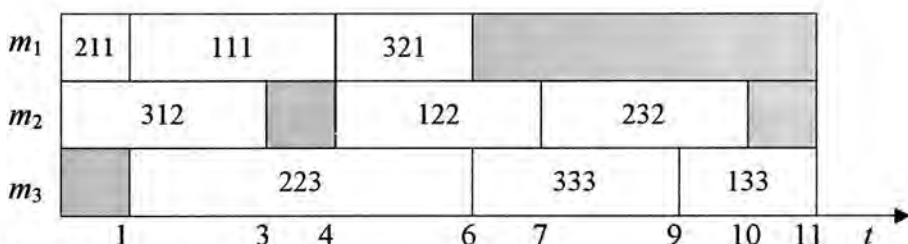
Problem je odrediti redoslijed operacija pojedinih dijelova određenog radnog naloga koji će proizvesti optimalno iskorištenje proizvodnog sustava. Kriteriji prema kojima se optimizira rješenje datog problema mogu biti sljedeći:

- minimizacija ciklusa izrade
- minimizacija broja operacija radnog naloga koje kasne u redoslijedu izvođenja ili minimizacija prosječnog vremena kašnjenja
- minimizacija ukupnih vremena čekanja
- ili razne kombinacije ovih kriterija

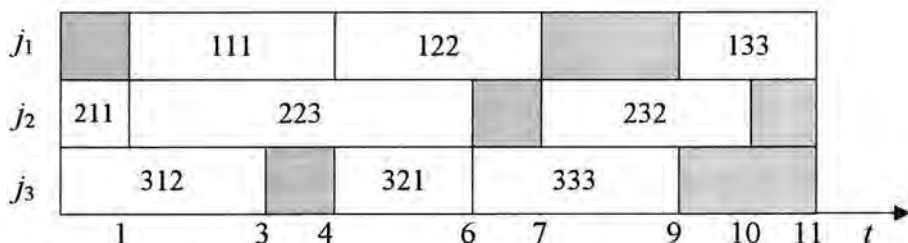
Primjer problema tri dijela – tri stroja prikazan je u tablici 3.1. Koristimo gantogram za ilustraciju mogućeg redoslijeda. Dvije su vrste gantograma: gantogram rada strojeva i gantogram dijelova. Za oznaku pojedine operacije koristi se index *jom* kod kojega *j* znači dio, *o* operaciju dijela *j* a *m* stroj na kojemu se operacija *o* izvodi. Na primjer, indeks 321 znači da će se na stroju 1 izvršiti druga operacija dijela 3. Slika 3.3 prikazuje redoslijed sa stanovišta zauzeća pojedinog stroja, dok je na slici 3.4 prikazan redoslijed sa stanovišta kada se izvršavaju operacije pojedinih dijelova.

Tab. 3.1 *Primjer problema tri izratka/dijela – tri stroja*

Vrijeme izvršavanja operacije				Redoslijed strojeva			
izradak	Operacije			izradak	Operacije		
	1	2	3		1	2	3
j_1	3	3	2	j_1	m_1	m_2	m_3
j_2	1	5	3	j_2	m_1	m_3	m_2
j_3	3	2	3	j_3	m_2	m_1	m_3



Sl. 3.3 *Gantogram rada strojeva*

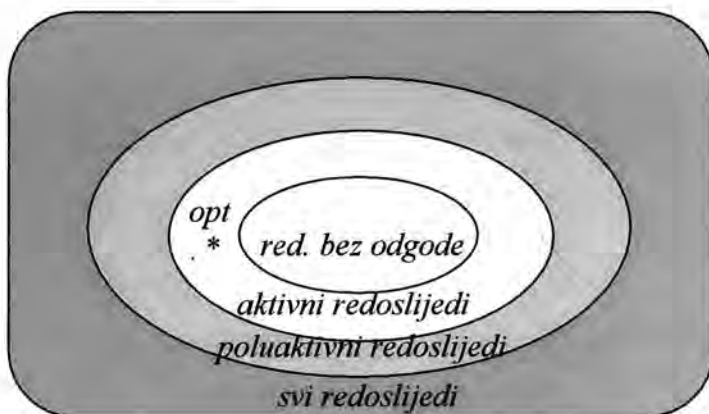


Sl. 3.4 *Gantogram izradaka/dijelova*

U principu, velik je broj izvedivih redosljeda za klasični model TRN-a zahvaljujući tome što se između dvije operacije može ubaciti neproduktivno vrijeme. Možemo operaciju pomaknuti ulijevo i zbiti sve do granice mogućnosti. Pomak u redosljedu se naziva *lokalni pomak ulijevo* ukoliko je moguće operaciju započeti bez mijenjanja redosljeda operacija. Pomak se naziva *globalni pomak ulijevo* ukoliko operacija može početi ranije bez odgađanja bilo koje druge operacije iako je s pomakom izmijenjen redosljed operacija. Razlikujemo tri vrste redosljeda na bazi ova dva koncepta, kako slijedi:

- **poluaktivan redosljed**: redosljed je poluaktivan ako ne postoji niti jedan lokalni pomak ulijevo
- **aktivan redosljed**: redosljed je aktivan ako ne postoji niti jedan globalni pomak ulijevo
- **redosljed bez odgode**: redosljed je bez odgode ako se niti jedan stroj ne održava u praznom hodu u trenutku kada bi mogao započeti s izvršenjem operacije

Odnos između skupova aktivnih, poluaktivnih i redosljeda bez odgode prikazan je na Venn-ovom dijagramu na slici 3.5. Iz slike 3.5. vidljivo je da najmanji broj redosljeda pripada skupu redosljeda bez odgode, te da je optimalan redosljed unutar skupa aktivnih redosljeda [67].



Sl. 3.5 Venn-ov dijagram odnosa među redosljedima (prema[66])

3.3.1.1. IP model

Formulaciju cjelobrojnog programiranja (IP^{11}) je razmatrao Baker [67] i, mnogo cjelovitije, Greenberg [68]. Ova se formulacija oslanja na indikator varijable za specificiranje redosljeda operacija. Cheng, Gen i Tsujimura predlažu novu formulaciju IP-a koristeći sistem linearnih nejednakosti da bi se bolje prikazala ograničenja za klasični TRN model [69].

Dvije su vrste ograničenja koje treba razmotriti za klasični problem terminiranja radnih naloga:

¹¹ *Integer Programming*

- ograničenja prioriteta operacija datog radnog naloga
- ograničenje nepreklapanja operacija za dati stroj

Neka c_{jk} označava vrijeme izvršenja operacije j na stroju k a t_{jk} označava vrijeme trajanja operacije j na stroju k . Najprije će se razmotriti ograničenja prioriteta operacija. Za operaciju i , ukoliko proces na stroju h prethodi onome na stroju k ograničenje je slijedeće:

$$c_{ik} - t_{ik} \geq c_{ih} \quad (3.18)$$

S druge strane, ukoliko proces na stroju k dolazi prvi, ograničenje je:

$$c_{ih} - t_{ih} \geq c_{ik} \quad (3.19)$$

Korisno je definirati koeficijent indikacije α_{ihk} kako slijedi:

$$\alpha_{ihk} = \begin{cases} 1, & \text{proces na stroju } h \text{ prethodi procesu na stroju } k \text{ za operaciju } i \\ 0, & \text{u svim ostalim slučajevima} \end{cases}$$

Prethodno navedena ograničenja mogu se napisati na slijedeći način:

$$c_{ik} - t_{ik} + M(1 - \alpha_{ihk}) \geq c_{ih}, \quad i=1,2,\dots,n, \quad h,k=1,2,\dots,m \quad (3.20)$$

gdje je: M proizvoljno veliki pozitivan broj.

Sada će se razmotriti ograničenje nepreklapanja operacije za dati stroj. Za dvije operacije i i j mora se zadovoljiti slijedeću nejednakost:

$$c_{jk} - c_{ik} \geq t_{jk} \quad (3.21)$$

S druge strane, ukoliko je prva operacija j , mora se zadovoljiti slijedeću nejednakost:

$$c_{ik} - c_{jk} \geq t_{ik} \quad (3.22)$$

Definira se varijablu indikacije x_{ijk} kako slijedi:

$$x_{ijk} = \begin{cases} 1, & \text{ako operacija } i \text{ prethodi operaciji } j \text{ na stroju } k \\ 0, & \text{u svim ostalim slučajevima} \end{cases}$$

Prethodno navedena ograničenja mogu se napisati na slijedeći način:

$$c_{jk} - c_{ik} + M(1 - x_{ijk}) \geq t_{jk}, \quad i,j=1,2,\dots,n, \quad k=1,2,\dots,m \quad (3.23)$$

Klasični problem terminiranja radnih naloga može se formulirati kako slijedi, s ciljem dobivanja minimalnog ciklusa izrade:

$$\min \sum_{i=1}^n \max_{1 \leq k \leq m} \{c_{ik}\} \quad (3.24)$$

uz slijedeće uvjete:

$$c_{ik} - t_{ik} + M(1 - \alpha_{ihk}) \geq c_{ih}, \quad i=1,2,\dots,n, \quad h,k=1,2,\dots,m \quad (3.25)$$

$$c_{jk} - c_{ik} + M(1 - x_{ijk}) \geq t_{jk}, \quad i,j=1,2,\dots,n, \quad k=1,2,\dots,m \quad (3.26)$$

$$c_{ik} \geq 0 \quad i=1,2,\dots,n, \quad k=1,2,\dots,m \quad (3.27)$$

$$x_{ijk} = 0 \text{ ili } 1, \quad i,j=1,2,\dots,n, \quad k=1,2,\dots,m \quad (3.28)$$

Ograničenje (3.25) osigurava da proces ide onim redoslijedom operacija koji je i propisan za određeni radni nalog. Ograničenje (3.26) osigurava da se na svakom stroju može obavljati samo jedna operacija u jedinici vremena.

3.3.1.2. LP model

Formulaciju linearnog programiranja (LP) razmatrali su Adams i dr. [47]. Neka je $N=\{0,1,2,\dots,n\}$ i neka označava skup operacija gdje se 0 i n smatraju lažnim operacijama “početka” i “kraja”, neka je $M=\{0,1,2,\dots,m\}$ i neka označava skup strojeva; neka A označava skup parova operacija koje ograničavaju relacije prvenstva za svaki posao i neka E_k označava skup parova operacija koje treba izvesti na stroju k i koji se, stoga ne mogu preklapati u vremenu. Za svaku operaciju i je, vrijeme trajanja operacije d_i fiksno, a startno vrijeme operacije t_i je varijabla koju treba odrediti tijekom optimizacije. Stoga se klasični problem TRN-a može formulirati kako slijedi:

$$\min t_n \quad (3.29)$$

uz slijedeće uvjete:

$$t_j - t_i \geq d_i \quad (i,j) \in A \quad (3.30)$$

$$t_j - t_i \geq d_i \quad \text{ili} \quad t_i - t_j \geq d_i \quad (i,j) \in E_k, \quad k \in M \quad (3.31)$$

$$t_i \geq 0 \quad (3.32)$$

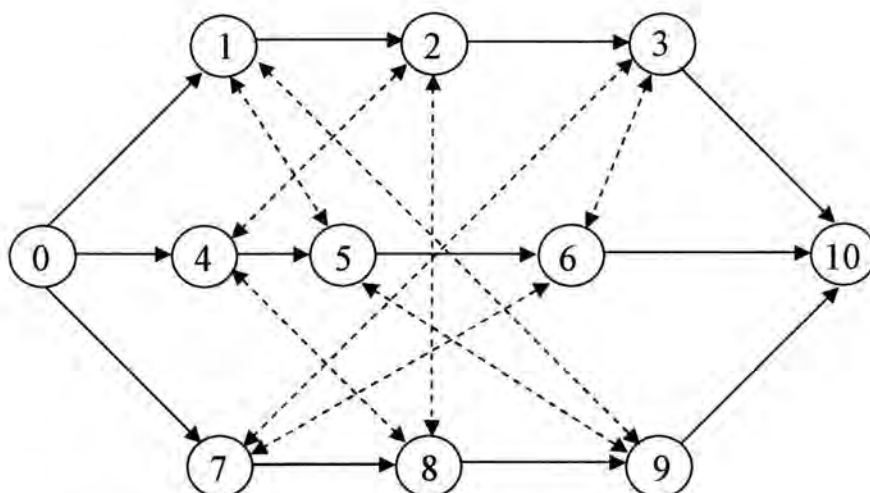
Cilj je minimiziranje ciklusa izrade. Ograničenje (3.30) osigurava da proces ide onim redoslijedom operacija koji je i propisan za određeni radni nalog. Ograničenje (3.31) osigurava da se na svakom stroju može obavljati samo jedna operacija u jedinici vremena.

3.3.1.3. Grafički model

Klasični problem TRN-a može se prikazati disjunktним grafikonom [70,71]. Disjunktни graf $G=(N, A, E)$ se definira kako slijedi: skup N sadrži čvorove koji predstavljaju operacije, skup A sadrži lukove koji spajaju niz operacija istog izratka, a E sadrži disjunktne lukove koji spajaju operacije koje se obavljaju na istom stroju. Disjunktни luk može biti postavljen sa bilo kojim od dva moguća pravca. Konstrukcija redoslijeda će postaviti orijentaciju svih disjunktних lukova tako da se odredi redoslijed operacija na jednom stroju. Jednom kada je određen redoslijed za stroj, disjunktни lukovi koji spajaju operacije koje će se izvesti na jednom stroju biti će zamijenjeni uobičajenom (orijentiranom) linijom sa strelicom ili konjuktivnim lukom. Set disjunktivnih lukova E se može raščlaniti u niz $E_k, E = E_1 \cup E_2 \cup E_3 \dots \cup E_m$, za svaki stroj.

Slika 3.6 ilustrira disjunktни graf za slučaj tri izratka – tri stroja, kod kojega svaki pojedini izradak sadrži tri operacije. Čvorovi $N = \{0,1,2,3,4,5,6,7,8,9,10\}$ odgovaraju operacijama; čvor 0 i 10 su lažne operacije “početka” i “kraja”.

Konjuktivni lukovi $A=\{(1,2), (2,3), (4,5), (5,6), (7,8), (8,9)\}$ odgovaraju ograničenjima operacija istog izratka zbog prioriteta. Disjunktivni lukovi (iscrtkane linije) $E_1=\{(1,5), (5,9), (9,1)\}$ odgovaraju operacijama koje će se izvesti na stroju 1 a disjunktivni lukovi (iscrtkane linije) $E_2=\{(4,2), (2,8), (8,4)\}$ odgovaraju operacijama koje će se izvesti na stroju 2; disjunktivni lukovi $E_3=\{(7,3), (3,6), (6,7)\}$ odgovaraju operacijama koje će se izvesti na stroju 3.



Sl. 3.6 Prikaz disjunknog grafa

Klasični problem TRN-a je naći redoslijed operacija na svakom stroju, tj. postaviti orijentaciju disjunktnih lukova tako da je rezultirajući graf acikličan (nema sukoba prioriteta između operacija) te osigurati minimalnu duljinu najdužeg puta između čvora početka i čvora završetka.

Tradicionalne metode za klasični problem TRN-a temelje se uglavnom na primjeni metoda matematičkog programiranja. Temelj za bilo koji od matematičkih modela je dodavanje varijabli tipa nema-ima (0-1) nekog dijela na pojedinom radnom mjestu. Te varijable određuju redoslijed kojim će se pojedini dijelovi izrađivati na pojedinom radnom mjestu. Osnovna formulacija problema i bilo kojeg od modela pripada klasi nedeterminističkih problema, jer se radi sa velikim brojem takvih nepoznanica. Način rješavanja problema u osnovi je kombinatoran – za unaprijed fiksirane vrijednosti nepoznanica (0 ili 1) rješava se standardni problem LP-a, te se iz dobivenog skupa različitih rješenja određuje najbolje prema definiranom cilju. Broj 0-1 nepoznanica obično je velik, te je broj modela koje treba riješiti u slučaju raspoređivanja n dijelova na m strojeva velik i iznosi između $O(n \log n)$ i $O(n^m)$. Osnovni je nedostatak ovih metoda sporo dobivanje rješenja uz velik utrošak računalnih resursa, a njihova uporaba pri rješavanju kompleksnijih problema kojima se ovaj rad bavi, bila bi neučinkovita.

3.3.2. Konvencionalna heuristika ¹²

Terminiranje radnih naloga praktičan je problem kojeg susrećemo svakodnevno. Budući da je problem TRN-a među najtežim kombinatornim problemima optimizacije, prirodno je tražiti aproksimativne metode koje mogu dati prihvatljiv redoslijed u okviru korisnog vremena. Heurističke procedure za problem TRN-a mogu se grubo podijeliti na dvije klase:

- heuristika jednog prolaza
- heuristika više prolaza

Heuristika jednog prolaza gradi jedno potpuno rješenje fiksirajući operaciju u redoslijedu u vremenu definiranom pomoću pravila raspodjele po prioritetima. Postoji mnogo pravila za odabir specifičnog podskupa koji bi se trebao rasporediti kao slijedeći po redoslijedu. Heuristika je brza i obično se pomoću nje nađe rješenje. K tome, heuristika jednog prolaza može se opetovano ponavljati da bi se izgradila mnogo sofisticiranija heuristika više prolaza a da bi se postigao bolji redoslijed uz nešto dodatnog troška računanja.

3.3.2.1. Heuristika prioriteta pri raspodjeli ¹³

Pravila prioriteta su, najčešće primjenjivana heuristika za rješavanje problema TRN-a zbog jednostavnosti njihove primjene. Giffler-Thompsonovi algoritmi se smatraju zajedničkom osnovom kompletne heuristike bazirane na pravu prioriteta [72]. Giffler i Thompson su predložili dva algoritma za generiranje redoslijeda: procedura za generiranje *aktivnog redoslijeda* i procedura za generiranje *redoslijeda bez odgoda* [73]. Vrijednost procedure redoslijeda bez odgoda je u tome što niti jedan stroj nije u praznom hodu kad imamo na raspolaganju operaciju koju treba izvršiti. Vrijednost aktivnog redoslijeda jest da niti jedna operacija ne može prije započeti bez odgađanja druge operacije. Aktivni redoslijed formira puno veći skup i uključuje redoslijede bez odgoda kao podskupove. Procedura za generiranje koju su izradili Giffler i Thompson ima pristup strukture stabla. Čvorovi u stablu odgovaraju dijelovima redoslijeda, lukovi predstavljaju moguće izbore, lišće je skup nabrojanih redoslijeda. Za zadani djelomični redoslijed, algoritam identificira sve sukobe u procesu (npr. operacije koje se "takmiče" za isti stroj) i proceduru popisa koja se koristi da bi se ovi sukobi riješili na bilo koji način u svakoj fazi procesa. Heuristika rješava ove sukobe na bazi pravila prioriteta raspodjele; tj. ona specificira pravilo prioriteta za odabir jedne operacije između operacija koje su sukobljene.

Procedure generiranja rade sa skupom operacija, kojima je moguće definirati redoslijed, na svakom od stupnjeva generiranja. Operacije kojima je moguće definirati redoslijed su operacije kojima nije definiran redoslijed ali im se zna operacija koja neposredno prethodi; ovaj skup se može jednostavno odrediti iz strukture prethodnika. Broj faza za proceduru jednog prolaza jednak je broju operacija $m \times n$. Na svakom stupnju jedna operacija je odabrana za dodavanje u djelomičan redoslijed. Sukobljavanje

¹² pristup rješavanja problema karakteriziran kod istraživanja pokusa i greške, a primjeren radu ili svojstvu koje bi moglo biti istina (ali ne garantirano)

¹³ izbor, određivanje slijeda i doznačivanje posla određenom radnom centru

operacija rješava se pomoću pravila prioriteta kod raspodjele. Na temelju Baker-ova zapažanja [66], slijedi:

- PS_t = djelomičan redoslijed koji sadrži t operacija
- S_t = skup operacija kojima je moguće odrediti redoslijed u fazi t a odgovaraju zadanom PS_t
- σ_i = najranije vrijeme kada operacija $i \in S_t$ može započeti
- Φ_i = najranije vrijeme kada operacija $i \in S_t$ može završiti

Za zadani aktivni djelomični redoslijed potencijalno vrijeme početka σ_i se određuje pomoću vremena završetka direktnog prethodnika operacije i i posljednjeg dobivenog vremena završetka na stroju koji je potreban za izvršenje operacije i . Veća od ove dvije veličine se uzima za σ_i . Potencijalno vrijeme završetka Φ_i je jednostavno zbroj $\sigma_i + t_i$ gdje je t_i procesno vrijeme za operaciju i . Procedura generiranja aktivnog redoslijeda je, kako slijedi:

Algoritam: Heuristika raspodjele prema prioritetima (generiranje aktivnog redoslijeda)

- Korak 1.** Neka je $t=0$ i počinje sa PS_t kao nultim djelomičnim redoslijedom. Inicijalni S_t uključuje sve operacije bez prethodnika
- Korak 2.** Odredimo $\Phi_t^* = \min_{i \in S_t} \{\Phi_i\}$ i stroj m^* za koji se može realizirati Φ_t^*
- Korak 3.** Za svaku operaciju $i \in S_t$ za koju je potreban stroj m^* i za koju je $\sigma_i < \Phi_t^*$ izračuna se indeks prioriteta prema specifičnom pravilu prioriteta. Pronađe se operaciju sa najmanjim indeksom i dodamo ovu operaciju na PS_t što je moguće ranije kreirajući tako novi parcijalni redoslijed PS_{t+1}
- Korak 4.** Za PS_{t+1} treba ažurirati podatke kako slijedi:
Ukloniti operacije i iz S_t
Formirati S_{t+1} dodavanjem direktnih sljedbenika operacije j u S_t
Povećati t za jedan
- Korak 5.** Povratak na korak 2 i tako sve do završetka generiranja redoslijeda

Ukoliko se zamijene korak 2 i korak 3 algoritma sa istim koracima iz algoritma koji slijedi, procedurom se generira redoslijed bez odgoda.

Algoritam: Heuristika raspodjele prema prioritetima (generiranje redoslijeda bez odgoda)

- Korak 1.** Neka je $t=0$ i počinje sa PS_t kao nultim djelomičnim redoslijedom. Inicijalni S_t uključuje sve operacije bez prethodnika
- Korak 2.** Odredimo $\sigma_t^* = \min_{i \in S_t} \{\sigma_i\}$ i stroj m^* za koji se može realizirati σ_t^*
- Korak 3.** Za svaku operaciju $i \in S_t$ za koju je potreban stroj m^* i za koju je $\sigma_i = \sigma_t^*$ izračuna se indeks prioriteta prema specifičnom pravilu prioriteta. Pronađe se operaciju sa najmanjim indeksom i dodamo ovu operaciju na PS_t što je moguće ranije kreirajući tako novi parcijalni redoslijed PS_{t+1}
- Korak 4.** Za PS_{t+1} treba ažurirati podatke kako slijedi:
Ukloniti operacije i iz S_t

Formirati S_{t+1} dodavanjem direktnih sljedbenika operacije j u S_t
 Povećati t za jedan

Korak 5. Povratak na korak 2 i tako sve do završetka generiranja redoslijeda

Problem koji ostaje je identifikacija efikasnog pravila prioriteta. Za obimniji sažetak i raspravu pogledati u Panwalkar i Iskander [74], Haupt [75] i Blackstone i dr. [76]. Tablica 3.2 sadrži neke od pravila prioriteta koji su uobičajeni u praksi.

Tab. 3.2 Lista pravila pri raspodjeli

Pravilo	Opis
SPT ¹⁴ (najkraće vrijeme trajanja operacije)	Odabir operacije s najkraćim vremenom trajanja
LPT ¹⁵ (najdulje vrijeme trajanja operacije)	Odabir operacije s najduljim vremenom trajanja
MWR ¹⁶ (najviše rada preostaje)	Odabir operacije radnog naloga s najvećim preostalim vremenom izrade
LWR ¹⁷ (najmanje rada preostaje)	Odabir operacije radnog naloga s najmanjim preostalim vremenom izrade
MOR ¹⁸ (najviše operacija preostaje)	Odabir operacije radnog naloga kod kojeg ostaje najveći broj operacija za izvršiti
LOR ¹⁹ (najmanje operacija preostaje)	Odabir operacije radnog naloga kod kojeg ostaje najmanji broj operacija za izvršiti
EDD ²⁰ (najraniji rok isporuke)	Odabir radnog naloga s najranijim rokom isporuke
FCFS ²¹ (prvi dospio, prvi opslužen)	Odabir prve operacije iz niza operacija na istom stroju
RANDOM (nasumce/slučajno)	Odabir operacije nasumce/slučajan odabir

3.3.2.2. Heuristika slučajnog odabira pri raspodjeli

Dok je heuristika jednog prolaza ograničena na konstruiranje jednog rješenja, heuristika više prolaza (znana i kao heuristika istraživanja) pokušava postići bolja rješenja generiranjem velikog broja rješenja uz, obično, veći utrošak vremena računanja. Tehnike kao što je metoda grananje – ograničavanje (branch and bound metoda) i dinamičko programiranje, mogu garantirati optimalno rješenje ali nisu praktične za probleme većeg omjera.

Heuristika slučajnog odabira je zapravo rani pokušaj osiguravanja točnijeg rješenja [67]. Ideja raspodjele na bazi slučajnog odabira započinje porodicom pravila raspodjele. Za svaku selekciju operacija koje će se pokrenuti odabire se, nasumce, pravilo raspodjele i to se ponavlja stalno kroz proces generiranja cijelog redoslijeda. Cijeli proces se ponavlja nekoliko puta i odabire se najbolji rezultat. Procedura generiranja aktivnog redoslijeda je slijedeća:

¹⁴ *Shortest Processing Time*

¹⁵ *Longest Processing Time*

¹⁶ *Most Work Remaining*

¹⁷ *Least Work Remaining*

¹⁸ *Most Operations Remaining*

¹⁹ *Least Operations Remaining*

²⁰ *Earliest Due Date*

²¹ *First Come, First Served*

Algoritam: Heuristika raspodjele prema slučajnom odabiru (generiranje aktivnog redoslijeda)

- Korak 0.** Neka je najbolji redoslijed BS nulti redoslijed
- Korak 1.** Neka je $t=0$ i počinje sa PS_t kao nultim djelomičnim redoslijedom. Inicijalni S_t uključuje sve operacije bez prethodnika
- Korak 2.** Odredimo $\Phi_t^* = \min_{i \in St} \{\Phi_i\}$ i stroj m^* na kojem se može realizirati Φ_t^*
- Korak 3.** Odabire se pravilo raspodjele nasumce iz familije pravila. Za svaku operaciju $i \in St$ za koju je potreban stroj m^* i za koju je $\sigma_i < \Phi_i$ izračuna se indeks prioriteta prema specifičnom pravilu. Pronađe se operaciju s najmanjim indeksom i dodamo ovu operaciju na PS_t što je moguće ranije kreirajući tako novi parcijalni redoslijed PS_{t+1}
- Korak 4.** Za PS_{t+1} treba ažurirati podatke kako slijedi:
Ukloniti operacije i iz S_t
Formirati S_{t+1} dodavanjem direktnih sljedbenika operacije j u S_t
Povećati t za jedan
- Korak 5.** Povratak na korak 2 i tako sve do završetka generiranja redoslijeda
- Korak 6.** Ukoliko je generirani redoslijed u prethodnom koraku bolji od najboljeg nađenog do tada, isti se sačuva kao BS . Povratak na korak 1 sve dok se ne ponovi vrijeme jednako vremenu koje je predodređeno.

Razni istraživači su pokušali poboljšati pristup slučajnog odabira. Jedna od promjena je proces učenja tako da uspješnija pravila raspodjele imaju veće šanse za odabirom u budućnosti. Morton i Pentico su predložili vođeni pristup slučajnog odabira [77]. Termin vođeni znači da je najprije potrebna izvrsna heuristika kojom bi se "istražio" problem i koja bi dala vodilju gdje se treba tražiti rješenje.

3.3.2.3. Heuristika premještanja uskog grla

Među najsposobnijim procedurama danas, u okviru heurističkih, za problem TRN-a je heuristika premještanja uskog grla od Adams-a, Balas-a i Zawack-a [47]. Njome se strojevi nizaju sukcesivno jedan za drugim uzimajući vrijeme rada na stroju kao usko grlo u skupu strojeva koji još nisu u slijedu. Svaki puta kada se u slijed stavi novi stroj, prethodno određen slijed se lokalno reoptimizira. Obje procedure, procedura prepoznavanja uskog grla i procedura lokalne reoptimizacije, se zasnivaju na ponavljanju rješavanja određenog problema redoslijeda jednog stroja. Metoda rješavanja problema jednog stroja nije novost iako su Adams, Balas i Zawack bitno ubrzali vrijeme potrebno za generiranje rješenja problema. Glavni doprinos njihovog pristupa je ovo rasterećenje koje koriste da bi odlučili o redoslijedu po kojem trebaju biti raspoređeni strojevi. Ovaj se način bazira na davanju prioriteta strojevima koji su usko grlo u slijedu. Neka je M_0 skup strojeva koji su već poredani (na početku je $M_0 = \emptyset$). Kratki izvod procedure premještanja uskog grla je, kako slijedi:

Algoritam: Heuristika premještanja uskog grla

- Korak1.** Prepoznavanje/identifikacija stroja m koji je usko grlo među strojevima $k \in MM_0$ i optimalno raspoređivanje u slijed.
Postavljanje $M_0 \leftarrow M_0 \cup \{m\}$

Korak2. Reoptimizacija slijeda za svaki kritični stroj $k \in MM_0$ naizmjenice dok se ostali dio slijeda drži fiksni. Kada je $M_0 = M$ prestaje se; inače se vraća na korak 1

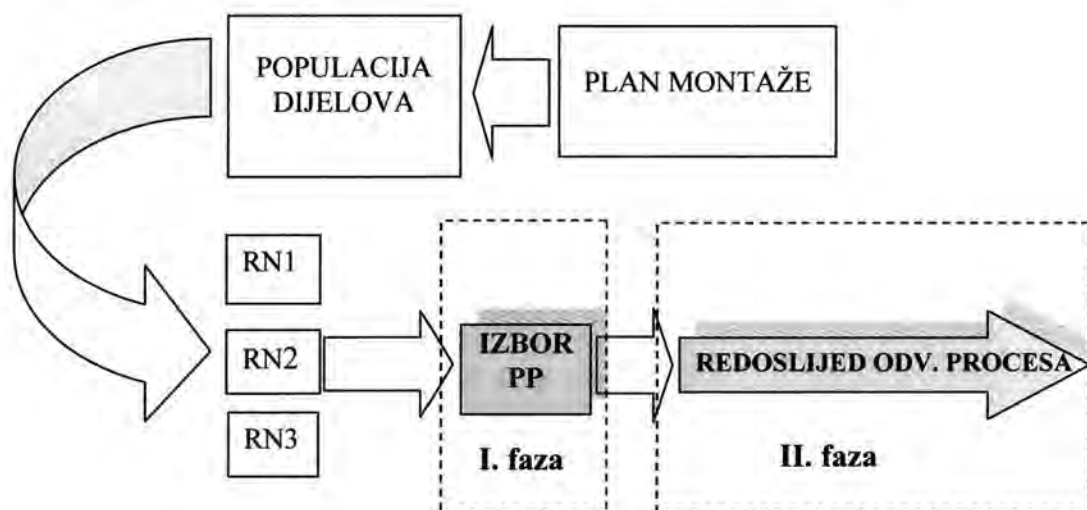
Detalji uvođenja mogu se pronaći kod Adams-a, Balas-a i Zawack-a [47] ili kod Applegate-a i Cook-a [48]. Prema iskustvima računanja Adams-a, Balas-a i Zawack-a, tipično poboljšanje njihovih procedura će biti negdje između 4% i 10%. Storer, Wu i Vaccari izvijestili su o tome da se uvođenjem procedure premještanja uskog grla nije moglo riješiti dva 10X50 problema [72]. Nedavno su Dauzere-Peres i Lasserre dali modificiranu verziju heuristike premještanja uskog grla [49].

4. RJEŠAVANJE PROBLEMA UPRAVLJANJA FPS-a PRIMJENOM GENETSKIH ALGORITAMA

Postojeće metode daju određena rješenja ali ostavljaju mnogo prostora za daljnja istraživanja i razvoj daljnjih metoda, jer su neke prikladne samo za male obime problema, neke su spore i neprikladne za operativno upravljanje, neke daju nedovoljnu razinu kvalitete rješenja. U ovom radu dat je doprinos rješenju problema upravljanja FPS-a s različitim strojevima pri realiziranju pojedinih radnih naloga čija struktura proizlazi iz planova montaže što znači, da je diktirana tržišnim zahtjevima po pitanju tipova dijelova i količina. Budući da struktura pojedinih radnih naloga nije u potpunosti usklađena s eksploatacijskim karakteristikama FPS-a koje proizlaze prvenstveno iz značajki strukture jezgre sustava, suočavamo se s problemom, kako u takvim uvjetima upravljati realiziranjem radnog naloga i pri tom postići glavni cilj: što manje vremensko zauzeće sustava odnosno što kraći ciklus izrade.

Problem je u ovom radu analiziran, a provedena istraživanja rezultirala su razvojem metode koja se temelji na dvije faze rješavanja problema upravljanja realiziranja radnog naloga, slika 4.1:

- I. faza upravljanja – izbor optimalne kombinacije planova procesa
- II. faza upravljanja – optimizacija redoslijednog odvijanja procesa



Sl. 4.1 Shematski prikaz I. i II. faze upravljanja FPS-a

U prvoj fazi određuje se izbor one kombinacije planova procesa između razvijenih planova procesa koja će pri simultanoj proizvodnji određenog mixa dijelova u zadanim količinama osigurati najmanje vremensko opterećenje proizvodnog sustava poznatih karakteristika. U podpoglavlju 3.2. postavljen je matematički model problema, te definiran postupak izbora planova procesa primjenom heurističkog algoritma temeljenog na tabu tehnici pretraživanja.

U drugoj fazi potrebno je provesti odgovarajuće upravljanje redoslijednog odvijanja pojedinih operacija parcijalnog radnog zadatka kako bi se smanjio ukupni

ciklus izrade, tj potrebno je optimirati redoslijed ulaska pojedinih dijelova u proces izrade i izvođenja pojedinih operacija vodeći pri tome računa o dva bitna kriterija:

- što kraće vremensko opterećenje sustava kao cjeline (radne stanice, automatizirani transportni sustav)
- što jednakomjernije opterećenje pojedinih radnih stanica

Pri tom je bilo nužno predložiti metodu, koja će imati veliku brzinu obrade ulaznih podataka i definiranja optimalnog izbora redoslijeda izvršenja pojedinih operacija iz cjeline radnog naloga, kako bi se osigurala primjenjivost pri operativnom upravljanju radom FPS-a i postigao glavni cilj.

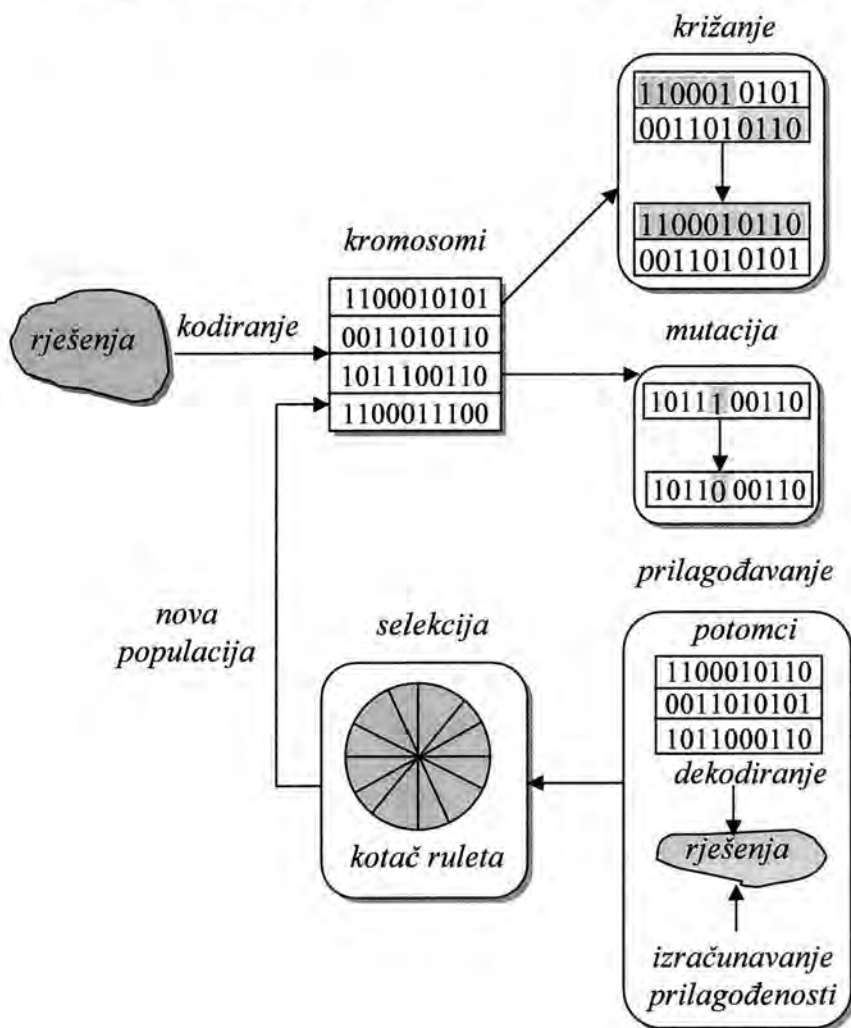
U podpoglavlju 3.4. uveden je klasični model TRN-a, kao pojednostavljen primjer stvarnog problema. Pojednostavljenje se očituje u tome što se klasični model TRN-a odnosi na terminiranje različitih vrsta dijelova količine 1 i zanemareno je opterećenje transportnog sustava, dok stvarni problem podrazumijeva terminiranje dijelova različitih tipova i količina i uključuje transportna vremena i vremena na ulaznim i izlaznim redovima čekanja. Međutim, na temelju pojednostavljenog modela formirat će se logika rješavanja optimizacije stvarnog problema, a rezultati dobiveni izrađenim software-om u ovom radu, moći će se usporediti s rezultatima iz literature.

Za rješavanje problema u obje faze, bit će korišteni genetski algoritmi kao snažni i široko primijenjeni algoritmi stohastičkog pretraživanja temeljeni na principima teorije evolucije.

Ovisno o vrsti problema, tj načinu predstavljanja problema, a time uzrokovane i primjene određenih operatora genetskog algoritma, u slijedećim podpoglavljima predstaviti će se implementacija genetskog algoritma u fazama vezanim za vrstu problema i to: u podpoglavlju 4.3. prikazat će se implementacija genetskog algoritma na izbor planova procesa, a u podpoglavlju 4.4. implementacija genetskog algoritma na klasični problem TRN-a. Razvijene tehnike za klasični problem TRN-a, opisane u podpoglavlju 4.4. iskoristit će se pri implementaciji genetskog algoritma na stvarni problem ovog rada, a to je problem TRN-a na FPS-u.

4.1. STRUKTURA GENETSKIH ALGORITAMA

Genetski algoritmi su stohastičke tehnike traženja zasnovane na mehanizmima prirodne selekcije i prirodne genetike [78,79]. Kako u prirodi, tako i u genetskim algoritmima, jedinke žive u populacijama. Postavljeni kriteriji optimalnog upravljanja čine vanjske uvjete, pod kojima jedinke žive. Jedinke u populaciji predstavljene su kromosomima koji su u okviru genetskih algoritama oblikovani kao uređeni skup gena, koji se mogu predstaviti kao binarni brojevi, realni brojevi, cijeli brojevi, simboli, permutacije elemenata itd. [78,79,80]. Za svaki kromosom može se izračunati vrijednost funkcije cilja, dakle za svaku jedinku može se znati koliko je dobro prilagođena uvjetima življenja. U prirodi, u nastajanju novog naraštaja najuspješnije jedinke imaju najveći udio, te se tako iz naraštaja u naraštaj u populaciji izloženoj stalnim uvjetima povećava prilagođenost. Operatori genetskih algoritama oponašajući procese prirodnog odabira, djeluju nad naraštajima umjetnih populacija i postižu postavljeni cilj - stvaranje najuspješnije jedinke, dakle nalaženje onog kromosoma koji ima optimalnu vrijednost funkcije cilja. Opća struktura genetskih algoritama predstavljena je na slici 4.2.



Sl. 4.2 Opća struktura genetskih algoritama

4.1.1. Predstavljanje

Predstavljanje je prvi korak u primjeni genetskih algoritama. Kako predstaviti rješenje problema u obliku kromosoma, ključno je pitanje genetskih algoritama. Predstavljanje je osnovni problem genetskih algoritama jer upravo predstavljanje povezuje realni svijet s genetskim algoritmima i tada genetski algoritmi direktno manipuliraju predstavljenim kromosomima.

Postavljanje samog zadatka optimalnog upravljanja znači odabir skupa parametara ξ_1, \dots, ξ_n , te funkcije cilja kojoj su ti parametri promjenjive veličine. Kada parametri poprimo određene vrijednosti, dobiva se točno određeno rješenje. Vrijednosti skupa parametara čine fenotip neke jedinke. No genetski algoritmi djeluju na razini gena te je potrebno poznavati genotip svake jedinke. Prema tome prvi problem je problem preslikavanja između fenotipa i genotipa, odnosno problem kodiranja skupa parametara u oblik umjetnog kromosoma.

Umjetni je kromosom niz gena, pa je problem kodiranja problem odabira onoga što se u genetskim algoritmima naziva abecedom gena te problem duljine kromosoma.

Holand [81] u svom radu, kodiranje izvodi koristeći binarnu abecedu. Međutim, za mnoge primjene genetskih algoritama posebno za probleme vezane uz inženjersku primjenu, predstavljanje kromosoma binarnim brojevima je neprimjenjivo.

Proteklih godina primjenjivale su se različite tehnike kodiranja za određene probleme:

- kodiranje realnim brojevima za optimizaciju problema sa zadanim ograničenjima
- kodiranje cijelim brojevima za optimizaciju kombinatornih problema

Kod ovih tehnika kodiranja, prilikom preslikavanja između fenotipa i genotipa, izlaze na vidjelo tri kritična pitanja:

1. Ostvarivost kromosoma
2. Legalnost kromosoma
3. Jedinственost preslikavanja

Kod problema s ograničenjima, ostvarivo područje predstavljeno je sustavom jednadžbi i nejednadžbi (linearnih ili nelinearnih). Za takve slučajeve mnoge efikasne metode predlažu rukovanje neostvarivim kromosomima [82,83,84].

Za optimizaciju mnogih kombinatornih problema korištene su specifične tehnike kodiranja koje dopuštaju ilegalne potomke nastale križanjem s jednom točkom reza. Zato što ilegalni kromosomi ne mogu biti dekodirani u rješenje to znači da se takvim kromosomima ne može procjenjivati prilagođenost, a time je za takve situacije i funkcija cilja neprimjenjiva. Usvojene su razne tehnike popravljavanja kromosoma iz ilegalnog u legalni. Problematika popravljavanja ilegalnog kromosoma jedan je od ciljeva rada te će se detaljno obraditi kasnije.

4.1.2. Prilagođenost jedinke (evaluacija)

U prirodi neka je jedinka manje ili više uspješna od drugih jedinki ovisno o tome kako joj genotip pretočen u fenotip omogućava da odgovori na uvjete u kojima živi. Za umjetnu jedinku u genetskom algoritmu također se iz njenog fenotipa - pripadnog

skupa parametara može izračunati vrijednost funkcije cilja te zaključiti da li slabije ili bolje od drugih jedinki zadovoljava kriterije koji su postavljeni. Radi kvantitativnog uspoređivanja jedinki nužna je mjera prilagođenosti jedinke. Mjera prilagođenosti ϕ mora poprimati nenegativne vrijednosti, te operatori genetskih algoritama omogućavaju nalaženje jedinke s najvećom prilagođenosti ϕ . Za razliku od toga, funkcija cilja φ zadatka optimalnog upravljanja općenito može poprimiti i pozitivne i negativne vrijednosti a optimalna vrijednost ovisno o problemu može biti najveća ili najmanja vrijednost te funkcije. Dakle, nije moguće jednostavno izjednačiti funkciju cilja φ i mjeru prilagođenosti ϕ već se postupa ovako.

Ako je zadatak optimizacije naći najveću vrijednost funkcije cilja φ jedini problem su moguće negativne vrijednosti što se može riješiti dodavanjem dovoljno velikog pozitivnog broja C , odnosno tada se uzima:

$$\phi = \max\{\varphi + C, 0\} \quad (4.1)$$

Ako je zadatak optimizacije naći skup parametara za koji je vrijednost funkcije cilja φ najmanja, tada tome odgovara najveća vrijednost funkcije $-\varphi$ te se uz dodavanje dovoljno velikog pozitivnog broja C također dobiva pogodnu mjeru prilagođenosti:

$$\phi = \max\{-\varphi + C, 0\} \quad (4.2)$$

No u genetskim alogritmima ovako dobivenu prilagođenost smatra se sirovom. Naime, jedinke ovisno o svojoj prilagođenosti sudjeluju u stvaranju novih naraštaja, točnije prema teoriji genetskih algoritama udio je neke jedinke proporcionalan eksponencijalnoj funkciji razlike između prilagođenosti promatrane jedinke i prosječne prilagođenosti u populaciji. Kako je u prvom naraštaju neke populacije velika većina jedinki slabo prilagođena i samo nekoliko jedinki izrazito odskaka od prosjeka, već bi u idućoj generaciji dominirali njihovi potomci te bi došlo do preranog gubljenja genetskog materijala. S druge strane, u kasnijim naraštajima razlika između prilagođenosti jedinki bi se smanjila te bi osrednje i bolje jedinke dobivale sličan udio u potomstvu i napredovanje populacije bi se usporilo. U oba slučaja problem je u prevelikoj, odnosno premaloj razlici između najveće i prosječne prilagođenosti u nekoj populaciji. To se rješava uvođenjem skalirane prilagođenosti koja za razliku od sirove treba u svim naraštajima zadržati sličan omjer između svoje najveće i prosječne vrijednosti.

U genetskim algoritmima moguća su ova skaliranja:

1. linearno skaliranje,
2. sigma rezanje,
3. skaliranje u obliku zakona potencija,
4. prozorsko skaliranje.

Budući da je u ovoj primjeni genetskog algoritma primijenjeno linearno skaliranje zbog svoje jednostavnosti, ostali načini skaliranja ovdje se neće objašnjavati. Kod linearnog skaliranja prikazanog na slici 4.3 iz sirove prilagođenosti računa se skaliranu pomoću izraza:

$$\tilde{\phi} = \max\left\{\phi_s + (c-1)\phi_s \frac{\phi - \phi_s}{\phi_{\max} - \phi_s}, 0\right\} \quad (4.3)$$

čime se postižu dva temeljna svojstva:

$$\tilde{\phi}_s = \phi_s \text{ i } \tilde{\phi}_{\max} = c\phi_s \quad (4.4)$$

gdje je:

c - po volji odabrani faktor, uspješno se rabe vrijednosti $1,2 < c < 2$,

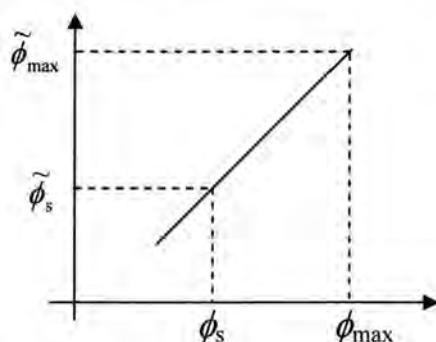
$\tilde{\phi}$ - skalirana prilagođenost,

ϕ_s - srednja prilagođenost,

ϕ_{\max} - maksimalna prilagođenost,

$\tilde{\phi}_s$ - srednja skalirana prilagođenost,

$\tilde{\phi}_{\max}$ - maksimalna skalirana prilagođenost,



SI. 4.3 Linearno skaliranje prilagođenosti

4.1.3. Odabir roditelja (selekcija)

U prirodi, jedinke koje su bolje prilagođene od ostalih jedinki iste populacije žive dulje i imaju više potomaka. Do sada je objašnjeno, kako u genetskim algoritmima populaciju čine jedinke od kojih svaka ima svoj genotip, iz kojega se može dekodirati fenotip te izračunati mjeru prilagođenosti. Sada treba pogledati na koji se način, kada se u genetskim algoritmima stvara jedinke novog naraštaja, među jedinkama starog naraštaja bira roditelje.

Jasno je da je ovo jedan od ključnih koraka genetskih algoritama, da upravo ovdje treba doći do izražaja oponašanje prirodnog odabira, te da koliko puta će neka jedinka biti birana za roditelja mora ovisiti o prilagođenosti te jedinke. Pri tome se ne zna što bi to bila apsolutno najbolja prilagođenost nego se može samo uspoređivati jedinke unutar jedne populacije. Zato se radi slijedeće: zbrajaju se prilagođenosti svih jedinki i promatra se za svaku jedinku relativna prilagođenost izražena u postocima:

$$\frac{\phi_i}{\sum_{k=1}^J \phi_k} \cdot 100\% \quad , \quad i = 1, \dots, J \quad (4.5)$$

gdje je: J - veličina populacije,
 ϕ_i - prilagođenost i -te jedinke.

Nadalje je poznato nekoliko mogućnosti:

1. stohastički odabir sa zamjenom,
2. stohastički odabir bez zamjene,
3. deterministički odabir,
4. stohastički odabir ostatka bez zamjene,
5. stohastički odabir ostatka sa zamjenom,
6. Wetzellovo rangiranje,
7. stohastički univerzalni odabir,
8. linearno rangiranje sa stohastičkim univerzalnim odabirom.

Stohastički odabir sa zamjenom naziva se i odabirom pomoću kotača ruleta. Naime, svakoj jedinki pridružuje se dio površine kotača ruleta u postocima jednake relativnoj prilagođenosti te jedinke. Sada je za odabir roditelja dovoljno zavrtjeti kotač ruleta i uzeti onu jedinku kojoj pripada područje u kojem se rulet slučajno zaustavio. Ovaj je odabir stohastički i vjerojatnost da neka jedinka bude odabrana kao roditelj jednaka je njenoj relativnoj prilagođenosti.

Stohastički odabir bez zamjene De Jongov je model očekivane vrijednosti. Kod ovog odabira računa se očekivani broj potomaka za svaku jedinku:

$$\frac{\phi_i}{\sum_{k=1}^J \phi_k / J}, \quad i = 1, \dots, J \quad (4.6)$$

Zatim se bira stohastički kao i kod prve metode, tj. bira se u skladu s istom vjerojatnosnom raspodjelom jedinki. Ako iza toga jedinka postaje roditelj jedne nove jedinke pripadni očekivani broj potomaka se smanjuje za 0,5, a ako jedinka prelazi u novi naraštaj bez zamjene pripadni očekivani broj potomaka se smanjuje za 1. Kada jedinki očekivani broj potomaka uslijed ovakvog smanjivanja padne ispod 0 jedinku se više ne može birati.

Kod metoda odabira 3., 4. i 5. počinje se tako da se svaku jedinku bira onoliko puta za roditelja koliki je cjelobrojni dio očekivanog broja potomaka (4.6). Time se naravno ne dobiva dovoljan broj parova roditelja te se javlja problem ostatka populacije koji se u ove tri metode rješava na različite načine.

Kod determinističkog se odabira poreda jedinke populacije prema veličini decimalnog dijela očekivanog broja potomaka (4.6) te se s vrha tako dobivenog popisa uzima potrebni ostatak roditelja.

Isti se ostatak roditelja primjenom stohastičkog odabira ostatka bez zamjene nalazi pomoću kotača ruleta i to tako da se svakoj jedinki pridruži dio kotača ruleta proporcionalan decimalnom dijelu očekivanog broja potomaka (4.6).

Stohastički odabir ostatka sa zamjenom znači da svaka jedinka osim što je već bila onoliko puta roditelj koliko iznosi cjelobrojni dio očekivanog broja potomaka (4.6) još može biti ponovno odabrana kao roditelj (do popunjavanja potrebnog broja) s vjerojatnošću jednakom decimalnom dijelu očekivanog broja potomaka (4.6).

Wetzellovo rangiranje slično je stohastičkom odabiru bez zamjene, ali nakon što se pomoću kotača ruleta odabere dvije jedinke, uspoređuje ih se prema njihovoj prilagođenosti i zadržava samo uspješniju kao budućeg roditelja.

Kod linearnog se rangiranja najprije poreda jedinke po prilagođenosti, tako da najbolje prilagođena jedinka postane prvom a najslabije prilagođena J -tom jedinkom populacije. Zatim se vjerojatnost da jedinka bude odabrana ne računa iz prilagođenosti nego iz položaja jedinke u poretku:

$$\frac{1}{J} \left(j_{\max} - \frac{i-1}{J-1} (j_{\max} - j_{\min}) \right) \quad (4.7)$$

gdje su $1 \leq j_{\max} \leq 2$ i $j_{\min} = 2 - j_{\max}$ po volji odabrani parametri.

Za ove i druge inačice odabira izvršena su brojna ispitivanja i poznata je njihova učinkovitost na različitim problemima. Dostupnost različitih metoda odabira kod primjene genetskih algoritama povećava mogućnosti optimalnog upravljanja. U radu su isprobane različite inačice operatora odabira genetskog algoritma. Istraženi su rezultati pokazali superiornost nekih te se pokazalo da je bilo najbolje rabiti stohastički odabir sa zamjenom.

4.1.4. Operatori genetskih algoritama

4.1.4.1. Križanje

U genetskim algoritmima tipično je da dva roditelja imaju dva potomka. Pri tom je najbitniji prijenos genetskog materijala sa starih na nove jedinke križanjem. Točnije, križanje se događa samo s nekom od korisnika odabranom vjerojatnošću, pa se može dogoditi da uopće ne dođe do križanja. U tom slučaju odabrane stare jedinke neizmijenjene ulaze u novi naraštaj populacije. Kada pak dolazi do križanja jedan potomak nasljeđuje dio gena od jednog a ostali dio gena od drugog roditelja, a drugi potomak nasljeđuje preostale dijelove kromosoma.

Prema vrsti problema (aritmetički problem, problem s ograničenjima ili kombinatorni problem) i načinu na koji se bira koje će gene naslijediti koje dijete od kojeg roditelja razlikuje se više različitih tipova križanja, o kojima će se nešto više reći kasnije.

4.1.4.2. Mutacija

Do sada proučeni operatori genetskih algoritama rade samo sa postojećim genetskim materijalom neke populacije. Ako za neki gen u početnoj populaciji nema niti jedne jedinke koja bi imala vrijednost tog gena jednaku nekoj vrijednosti iz odabrane genetske abecede, tako ostaje i u svim budućim naraštajima te populacije. Štoviše napredovanjem kroz naraštaje smanjuje se raznolikost genetskog materijala. To može dovesti do usporavanja konvergencije genetskih algoritama i nagomilavanja populacije oko nekog lokalnog, dakle lažnog optimuma. Da se to izbjegne u genetskim algoritmima ponovno se oponaša prirodu, te se povećava raznolikost ukupnog genetskog materijala populacije uvodeći mutacije.

Ovisno o vrsti problema (aritmetički problem, problem s ograničenjima ili kombinatorni problem), tj. o načinu kodiranja kromosoma (binarno ili neko drugo kodiranje), razlikuje se više različitih tipova mutacija, o kojima će se nešto više reći kasnije.

4.1.4.3. Napredni operatori genetskih algoritama

Osnovni operatori genetskih algoritama su odabir, križanje i mutacija i već su oni dovoljni za uspješno rješavanje različitih problema optimizacije. No jasno je da osnovni genetski algoritam tek vrlo pojednostavljeno oponaša procese prilagodbe, reprodukcije i razvoja živih bića. Slijedeći logiku, da priroda vrlo uspješno stvara sve savršenije i savršenije oblike života pa je u postupcima optimizacije treba samo što vjernije oponašati, razvijeni su brojni napredni operatori i tehnike: diploidnost i dominantni geni, inverzija, redukcija incesta, određivanje spola, specijalizacija, migracije između subpopulacija itd.

4.2. GENETSKI ALGORITMI U OPTIMALNOM IZBORU PLANOVA PROCESA

4.2.1. Genotip i fenotip

Kao što je ranije rečeno, prvi je korak u primjeni genetskih algoritama, a tako i kod primjene genetskih algoritama pri traženju optimalnog izbora planova procesa odrediti skup parametara tj. genotip jedinke. Kada parametri poprima određene vrijednosti, dobiva se točno određeni skup planova procesa. Skup parametara i skup planova procesa može se poistovjetiti, odnosno vrijednosti skupa parametara čine fenotip neke jedinke. Pošto genetski algoritmi djeluju na razini gena, prvi je zadatak kodiranje skupa parametara u oblik umjetnog kromosoma.

U rješavanju problema izbora planova procesa, gen predstavlja broj plana procesa za pojedini dio [85,86]. Duljina kromosoma predstavljena je brojem različitih vrsta dijelova koje treba obraditi. Označi li se broj dijelova s np dobiva se kromosom:

$$G_i = [g_1 g_2 \dots g_k \dots g_{np}] \quad (4.8)$$

gdje g_k predstavlja k -ti gen koji leži u intervalu $[1, npp_i \cdot g_L]$ gdje je npp_i ukupan broj razvijenih planova procesa za pojedini dio, a g_L gornja granica broja razvijenih planova procesa.

Kod ovog zadatka optimizacije k -ti je parametar realni broj koji ima smisao samo ako poprima vrijednost između neke najmanje i najveće moguće vrijednosti $[j_{k,\min}, j_{k,\max}]$. Tako, ako se izvrši preslikavanje intervala $[1, npp_i \cdot g_L]$ na interval $[j_{k,\min}, j_{k,\max}]$, dobiva se smisljena vrijednost k -tog parametra:

$$j_k \leftarrow (g_k - 1) / g_L + 1 \quad (4.9)$$

Kada se ovaj postupak ponovi za sve dijelove kromosoma, $k=1, \dots, np$ za jedinku je iz genotipa pročitana fenotip:

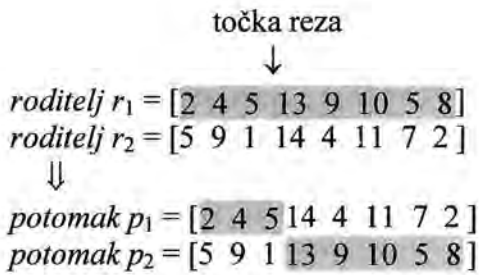
$$F_i = [f_1 f_2 \dots f_k \dots f_{np}] \quad (4.10)$$

4.2.2. Križanje

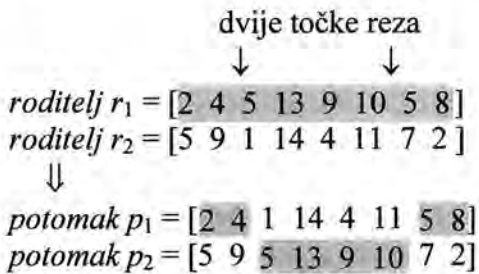
Prema načinu na koji se bira koje će gene naslijediti koji potomak od kojeg roditelja razlikuju se:

1. križanje s jednom točkom reza,
2. križanje s više točaka reza,
3. uniformno križanje,

Kod križanja s jednom točkom reza prikazanog na slici 4.4, slučajno se bira jedan položaj u kromosomu. Zatim sve gene od početka do odabrane točke u kromosomu prvog roditelja i sve gene od odabrane točke do kraja kromosoma drugog roditelja nasljeđuje prvi potomak, a sve preostale gene prvog i drugog roditelja drugi potomak.

**Sl. 4.4** Križanje s jednom točkom reza

Kod križanja s više točaka reza datog na slici 4.5, bira se više položaja za rez u kromosomu, također slučajno. Ako se radi križanje s parnim brojem točaka reza, zamišlja se da je spajanjem prvog i zadnjeg gena zatvoren kromosom u oblik prstena. Parnim brojem rezova kromosomi roditelja odvajaju se u dijelove, pa prvo dijete dobiva naizmjenično dijelove od prvog pa od drugog roditelja, a drugo dijete kao i obično preostali genetski materijal. Križanje s neparnim brojem točaka reza, svodi se na križanje s parnim brojem dodavanjem početka kromosoma kao obaveznog mjesta reza.

**Sl. 4.5** Križanje s parnim brojem točaka reza

Uniformno križanje znači da se svaki gen promatra neovisno od drugih. Pojedini gen prvo dijete nasljeđuje od prvog roditelja s vjerojatnošću 0,5, pa ako se to ne dogodi nasljeđuje ga od drugog roditelja. Drugo dijete dobiva isti gen od roditelja koji nije darovao taj gen prvom dijeteu.

Uzimajući u obzir ideju o graditeljskim cjelinama, jasno je da križanje s većim brojem točaka reza ne može biti korisno jer s većom vjerojatnosti razbija uspješne shemate. Zato se za većinu primjena genetskih algoritama preporučuje križanje s dvije točke reza. Također se obično uzima da se križanje događa s vjerojatnosti 0,6.

4.2.3. Mutacija

U ovoj primjeni pogodne su dvije vrste mutacije:

1. jednogenska mutacija,
2. višegenska mutacija.

Jednogenska mutacija djeluje tako da za svaku jedinku novog naraštaja pripadni operator prelazi preko svih gena, ali djeluje slučajno i to s vrlo malom vjerojatnosti. Kada djeluje vrijednost danog gena mijenja se u neku drugu slučajno odabranu vrijednost iz dozvoljene abecede genetskog materijala.

Višegenska mutacija razlikuje se od jednogenske samo po tome što kada pripadni operator djeluje zahvaća mutacijom više susjednih gena. Broj tih mutiranih gena slučajna je varijabla Poissonove raspodjele s vjerojatnosti obrnuto proporcionalnoj očekivanoj broju susjednih mutacija. Mutacije na rubu kromosoma rješavaju se zamišljajući kromosom u obliku prstena.

Kao i u prirodi ako je mutacija učinila jedinku slabije prilagođenom ta će jedinka biti uklonjena prirodnim odabirom. No ponekad se može dogoditi da upravo zahvaljujući mutaciji neki gen poprimi vrijednost koje prije nije bilo u populaciji, a koja bitno poboljšava prilagođenost i pospješuje nalaženje globalnog optimuma postavljenog problema.

Kod primjene najvažnije je birati vrlo malu vjerojatnost mutacije, obično oko vrijednosti 0,0333.

4.2.4. Evaluacija i selekcija

Pri izboru planova procesa, zadatak je optimizacije naći skup parametara, tj. odrediti fenotip kromosoma $v_i = \{p_1, p_2, p_3, \dots, p_k, \dots, p_n\}$, gdje je p_k izabrani plan procesa za dio k , za koji je vrijednost funkcije cilja ϕ dana izrazom (3.12) minimalna, uz ograničenja dana izrazima (3.13) i (3.14).

Mjera prilagođenosti za svaki kromosom iskazana je u skladu s funkcijom cilja kao vrijednost funkcije vrednovanja $\phi(v_i)$ prema:

$$\phi(v_i) = \max \{-\phi(v_i) + C\} \quad i = 1, 2, \dots, J \quad (4.11)$$

gdje je J broj jedinki u populaciji, $\phi(v_i)$ funkcija vrednovanja za i -ti kromosom, $\phi(v_i)$ je vrijednost funkcije cilja, a C dovoljno velik pozitivni broj.

U genetskim algoritmima ovako dobivenu prilagođenost smatra se sirovom, naime u prvim generacijama velika je većina jedinki slabo prilagođena i samo nekoliko jedinki odskače od prosjeka što bi rezultiralo samo njihovim potomcima u idućoj generaciji, a u kasnijim generacijama osrednje i bolje jedinke dobivale bi sličan udio u potomstvu i konvergencija rješenja bi se usporila. Ovaj se problem rješava uvođenjem skalirane prilagođenosti koja za razliku od sirove treba u svim generacijama zadržati sličan omjer između svoje najveće i prosječne vrijednosti. U ovoj je primjeni genetskog algoritma primijenjeno linearno skaliranje kod kojeg se iz sirove prilagođenosti računa skaliranu pomoću izraza (4.3).

Stvaranje novog naraštaja, odnosno odabir roditelja (selekcija) među jedinkama starog naraštaja ovisi o prilagođenosti jedinke. Budući da se ne zna što bi to bila apsolutno najbolja prilagođenost, odabir se radi stohastički pomoću kotača ruleta (stohastički odabir s zamjenom), na način da se svakoj jedinki pridruži dio površine kotača ruleta u postocima jednake relativnoj prilagođenosti te jedinke. Sada je za odabir roditelja dovoljno zavrtjeti kotač ruleta i uzeti onu jedinku kojoj pripada područje u kojem se rulet slučajno zaustavio. Ovaj je odabir stohastički i vjerojatnost da neka jedinka bude odabrana kao roditelj jednaka je njenoj relativnoj prilagođenosti, izraz (4.5).

4.3. GENETSKI ALGORITMI ZA KLASIČNI PROBLEM TRN-a

Problematika vezana uz uvođenje genetskih algoritama na klasične probleme TRN-a obrađivana je od mnogo autora [87,88,89,90,91,92,93,94,95,96,97,98,99,100]. Nedavni radovi na temu primjene genetskih algoritama za rješavanje problema terminiranja radnih naloga prikazani su u tablici 4.1.

U ovom će se podpoglavlju dati detaljna ilustracija nekoliko tipičnih implementacija genetskih algoritama na klasične probleme TRN-a.

Tab. 4.1 *Neki radovi o implementaciji genetskih algoritmima na problem TRN-a*

Autori	Opis
Davis (1985) [89]	Predstavljanje na bazi liste prioriteta
Nakano i Yamada (1991) [96]	Predstavljanje na bazi odnosa para dijelova r.n.
Falkenauer i Bouffoix (1991) [101]	Predstavljanje na bazi liste prioriteta
Bagchi i dr. (1991) [87]	-
Bagchi i dr. (1991) [87]	Predstavljanje prema specifičnosti problema
Nakano i Yamada (1992) [99]	Predstavljanje na bazi vremena izvršenja
Tamaki i Nishikawa (1992) [102]	Predstavljanje na bazi disjunktnog grafa
Paredis (1992) [96]	Predstavljanje na bazi odnosa para dijelova r.n.
Holsapple i dr. (1993) [94]	Predstavljanje na bazi dijelova radnog naloga
Fang i dr. (1993) [91]	Predstavljanje na bazi operacija
Gen i dr. (1994) [103]	Predstavljanje na bazi operacija
Bean (1994) [103]	Predstavljanje po ključu slučajnog odabira
Darndorf i Pesch (1995) [90]	Predstavljanje na bazi pravila prioriteta i na bazi strojeva
Croce i dr. (1995) [88]	Predstavljanje na bazi liste prioriteta
Kobayashi i dr. (1995) [104]	Predstavljanje na bazi liste prioriteta
Norman i Bean (1995) [105]	Predstavljanje po ključu slučajnog odabira

4.3.1. Predstavljanje

Zbog postojanja ograničenja koja prethode operacijama kao što su ograničenja prethodnosti, klasični problem TRN-a nije jednostavno vjerno prikazati kao problem putujućeg trgovca (TSP²²). Ne postoji dobar prikaz prethodnih ograničenja pomoću nejednadžbi. Stoga, kazneni postupak koji podrazumijeva odbacivanje neostvarivih ili nelegalnih kromosoma nije jednostavno primjeniti za rukovanje ovim tipom ograničenja. Orvosh i Davis [106] su pokazali da je za veliki broj kombinatornih problema optimizacije relativno jednostavno popraviti neostvarive ili nelegalne kromosome, te da strategija popravka ima prednost pred strategijom odbacivanja ili kazne. Većina autora, koji se bave primjenom genetskih algoritama na probleme terminiranja radnih naloga, koristi se upotrebom strategije popravka. Vrlo je važno, kod gradnje genetskih algoritama za problem TRN-a, napraviti prikladno predstavljanje rješenja zajedno s operatorima genetskih algoritama specifičnim za problem, tako da kromosomi generirani u inicijalnoj fazi ili u procesu evolucije produciraju ostvariv redosljed operacija radnog naloga. Ovo je ključna faza koja utječe na sve slijedeće korake genetskog algoritma. Tijekom posljednjih nekoliko godina predloženi su slijedeći tipovi predstavljanja klasičnog problema TRN-a:

²² *Travelling Salesman Problem*

- predstavljanje na bazi operacija
- predstavljanje na bazi dijelova radnog naloga
- predstavljanje na bazi liste prioriteta
- predstavljanje na bazi odnosa para dijelova radnog naloga
- predstavljanje na bazi pravila prioriteta
- predstavljanje na bazi disjunktne grafa
- predstavljanje na bazi vremena izvršenja
- predstavljanje na bazi strojeva
- predstavljanje po ključu slučajnog odabira

Ovi se prikazi mogu klasificirati u slijedeća dva osnovna pristupa kodiranja:

- direktan pristup
- indirektan pristup

Kod direktnog pristupa, redoslijed (rješenje za problem TRN-a) se kodira unutar kromosoma; koriste se genetski algoritmi za prilagođavanje ovih jedinki, kako bi se odredio bolji redoslijed. Predstavljanja, kao što su predstavljanje na bazi operacija, predstavljanje na bazi dijelova radnog naloga, predstavljanje na bazi odnosa para dijelova radnog naloga, predstavljanje na bazi vremena izvršenja i predstavljanje po ključu slučajnog odabira, pripadaju ovoj klasi.

Kod indirektnog pristupa kakav je predstavljanje na bazi pravila prioriteta, unutar kromosoma je kodiran slijed pravila raspodjele za određeni radni nalog (redoslijed), a genetski algoritmi se koriste za prilagođavanje ovih jedinki kako bi se odredio bolji slijed pravila raspodjele. Predstavljanje na bazi liste prioriteta, predstavljanje na bazi pravila prioriteta, predstavljanje na bazi disjunktne grafa i predstavljanje na bazi strojeva pripadaju ovoj klasi.

U ovom podpoglavlju detaljno će se analizirati samo onaj tip predstavljanja i odgovarajući pristup kodiranja koji je od važnosti u daljnjim istraživanjima ovog rada, koristeći pri tom jednostavan primjer prikazan u tablici 3.1.

4.3.1.1. Predstavljanje na bazi operacija

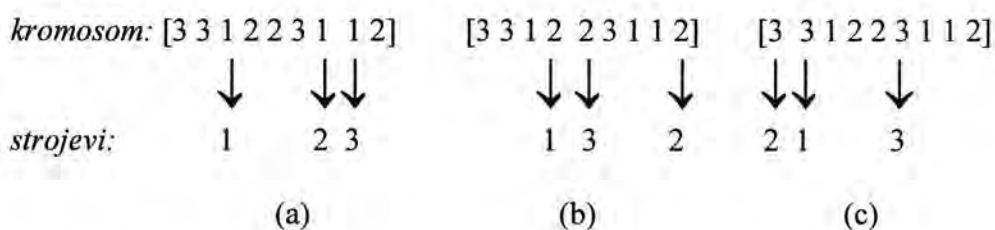
Kod ovog vida predstavljanja kromosom se sastoji od niza operacija, gdje svaki gen predstavlja jednu operaciju radnog naloga u nizu redoslijeda. Moguća su dva načina kodiranja svake operacije. Jedan od načina je imenovanje operacije stvarnim brojem, kao npr. kod permutacijskog predstavljanja za problem trgovačkog putnika. U tom slučaju za problem 3 dijela-3 stroja prikazan u tablici 3.1 kromosom se može predstaviti kao skup permutacija operacija: [3 2 5 4 7 1 6 9 8].

Nažalost zbog postojanja ograničenja prethodnosti operacija, ne daju sve permutacije ovih brojeva ostvarive redoslijede. Na primjer, broj 7 u gornjem kromosomu predstavlja prvu operaciju trećeg dijela j_3 , broj 8 drugu operaciju trećeg dijela j_3 , a broj 9 treću operaciju dijela j_3 . Da bi sve permutacije davale ostvarive redoslijede u svim permutacijama morao bi npr. broj 7 uvijek biti prije u nizu od broja 8, a ovaj prije broja 9, što je naravno neizvedivo, a zbog toga je ovaj način predstavljanja problema neprimjenjiv.

Gen, Tsujimura i Kubota predlažu alternativu. Oni sve operacije istog dijela imenuju istim brojem, a interpretiraju ih prema redu kojim se pojavljuju u nizu datog

kromosoma[107,108]. Za problem n -dijelova m -strojeva, kromosom se sastoji od $n \times m$ gena. Svaki pojedini dio pojavljuje se u kromosomu točno m puta, a broj ponavljajućih gena svakog dijela odnosi se na broj operacija određenog dijela. Lako je primjetiti da sve permutacije takvih kromosoma uvijek daju ostvarive redoslijede. Prema tome za primjer iz tablice 3.1 neka je dat kromosom [3 3 1 2 2 3 1 1 2] gdje broj 1 predstavlja dio j_1 , broj 2 predstavlja dio j_2 , a broj 3 dio j_3 . Svaki dio sadrži 3 operacije te će se iz tog razloga svaki dio pojaviti točno tri puta u kromosomu. Na primjer, za dati kromosom tri broja 2 predstavljaju 3 operacije dijela j_2 i to prvi broj 2 odgovara prvoj operaciji dijela j_2 koja se izvodi na stroju m_1 , drugi broj 2 odgovara drugoj operaciji dijela j_2 koja se izvodi na stroju m_3 i treći broj 3 odgovara trećoj operaciji dijela j_2 koja se izvodi na stroju m_2 . Može se primjetiti da su sve operacije dijela j_2 imenovane istim brojem 2 i onda protumačene prema njihovom redu pojavljivanja u nizu datog kromosoma.

Odgovarajući odnos operacija dijelova radnog naloga i strojeva na kojima se izvode operacije datih dijelova prikazan je na slici 4.6.

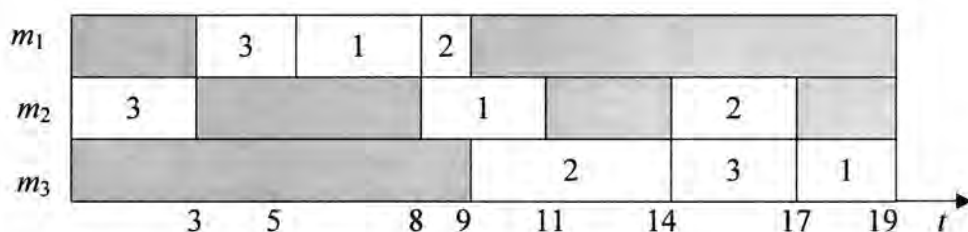


Sl. 4.6 Operacije radnog naloga i pripadajući im strojevi: (a) za dio j_1 , (b) za dio j_2 , (c) za dio j_3

Na temelju ovih relacija može se dobiti pripadajuća lista strojeva [2 1 1 1 3 3 2 3 2], koja je prikazana na slici 4.7 koja definira i redoslijed izvođenja operacija cijelog radnog naloga.



Sl. 4.7 Redoslijed izrade dijelova na stroju m_2 ; (prikazan osjenčano)



Sl. 4.8 Redoslijed izrade dijelova radnog naloga na bazi kromosoma [3 3 1 2 2 3 1 1 2]

Iz slike 4.7 se može vidjeti redosljed izrade pojedinih dijelova radnog naloga za svaki stroj i to: za stroj m_1 je $j_3 - j_1 - j_2$, za stroj m_2 je $j_3 - j_1 - j_2$ (na slici 4.7 prikazan osjenčano), te za stroj m_3 je $j_2 - j_3 - j_1$. Sumira li se sve ovo dobiva se ostvariv redosljed prikazan na slici 4.8.

4.3.1.2. Modificirano predstavljanje na bazi operacija

Cheng, Gen i Tsujimura modificiraju metodu Gen-a, Kubote i Tsujimure tako da dekodiranje kromosoma uvijek osigurava generaciju (naraštaj) aktivnog redosljeda.

Kod predstavljanja na bazi operacija Gen, Kubota i Tsujimura uzimaju u obzir samo dvije stvari: red operacija u datom kromosomu i ograničenja prethodnosti operacija pojedinog dijela. Također specificiraju red izvođenja operacija na pripadajućim strojevima. Dekodiranje takvih kromosoma može garantirati samo naraštaj poluaktivnog redosljeda, zbog postojanja dozvoljenih lijevih pomaka unutar redosljeda, te iz tih razloga ne može garantirati aktivni redosljed. Kao što je ranije rečeno optimalni je redosljed aktivni, pa je od izuzetne važnosti osigurati takvo dekodiranje koje će neminovno dati naraštaj aktivnog redosljeda.

Za primjer iz tablice 3.1 neka je kromosom [2 1 1 1 2 2 3 3 3]. Dekodiranjem kromosoma na temelju izloženog u prethodnom odjeljku, može se dobiti pripadajuća lista strojeva [1 1 2 3 3 2 2 1 3]. Na osnovi liste strojeva može se formirati redosljed prikazan u tablici 4.2 u kojoj je specificiran red operacija radnog naloga na svakom stroju.

Tab. 4.2 Redosljed dijelova radnog naloga po strojevima

Stroj	Redosljed dijelova radnog naloga		
m_1	j_2	j_1	j_3
m_2	j_1	j_2	j_3
m_3	j_1	j_2	j_3

Na slici 4.9(a) prikazan je Gantt-ov diagram poluaktivnog redosljeda dobivenog dekodiranjem datog kromosoma i ciklus izrade je 25. Može se primjetiti da postoje dva dozvoljena lijeva pomaka u poluaktivnom redosljedu:

1. prva operacija dijela j_3 može početi u vremenu 0 na stroju m_2
2. druga operacija dijela j_2 može početi u vremenu 1 na stroju m_3

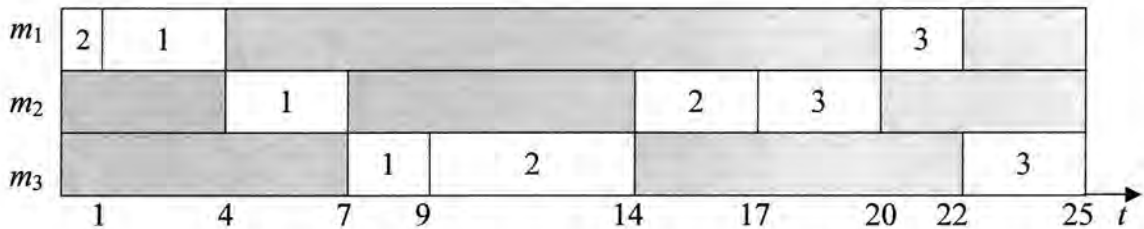
Izvedu li se ovi lijevi pomaci dobiva se aktivni redosljed prikazan na slici 4.9(b) koji ima ciklus izrade 12.

Za predstavljanje na bazi operacija, svaki gen jednoznačno pokazuje operaciju, dok modificirana metoda prvo prebacuje kromosom u listu određenih operacija. Redosljed se generira na bazi liste prioriteta heuristikom jednog prolaza. Prva operacija u listi smješta se prva u redosljedu, ovisno o vremenu njenog izvođenja i ovisno o tome na kojem se stroju izvodi. Nakon toga uzima se druga operacija itd.

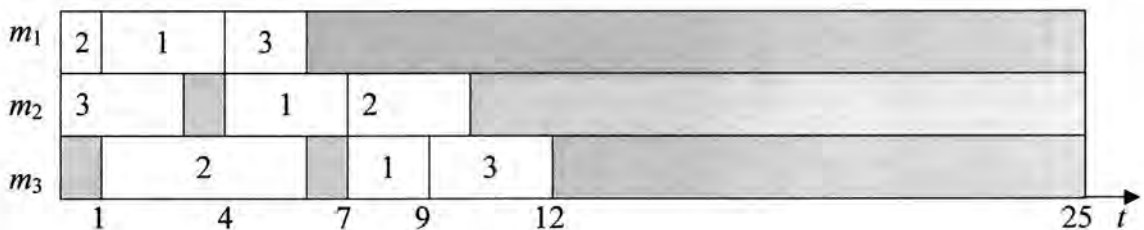
Neka pozicija operacije u listi, idući s lijeva na desno, označava prioritet operacije od najvećeg prema najmanjem. Tako će operacija s najmanjom pozicijom imati najveći prioritet. Operacija s najvećim prioritetom prva će se smjestiti u redosljed.

Neka o_{jim} označava i -tu operaciju j -tog dijela radnog naloga koja se izvodi na stroju m . Ranije spomenuti kromosom [2 1 1 1 2 2 3 3 3] može se prevesti u jednoznačnu listu određenih operacija [o_{211} o_{111} o_{122} o_{133} o_{223} o_{232} o_{312} o_{321} o_{333}].

Operacija o_{211} ima najveći prioritet i prva se smješta u redosljed, a zatim operacija o_{111} itd. Rezultat je aktivni redosljed prikazan na slici 4.9(b).



(a)



(b)

Sl. 4.9 Utjecaj lijevog pomaka u promjenjivom poluaktivnom redosljedu:
 (a) poluaktivni redosljed dobiven dekodiranjem kromosoma, i
 (b) aktivni redosljed nakon primjene lijevih pomaka na poluaktivni redosljed

4.3.2. Križanje

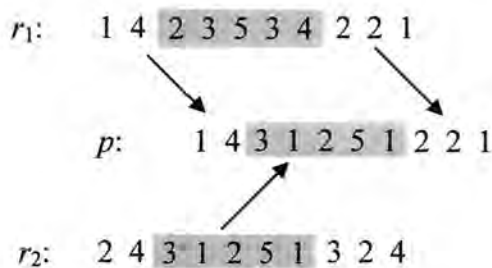
Križanje kod problema TRN-a operatorima križanja izloženim u odjeljku 4.2.2. (križanje s jednom točkom reza, križanje s više točaka reza i uniformno križanje) u ovom slučaju rezultiralo bi nelegalnim kromosomima koji bi onda dali neostvarive redosljede, tj. rezultat bi bili potomci od kojih bi jedan sadržavao veći broj ponavljajućih gena od dozvoljenog, a drugi potomak manji broj gena od dozvoljenog, a zna se da broj ponavljajućih gena svakog radnog naloga mora odgovarati točnom broju operacija datog radnog naloga.

Posljednjih godina predloženo je više tipova operatora križanja za ove probleme od mnogo autora [88,90,93,101,109], međutim ovdje će se dati detaljnija analiza samo operatora križanja bitnog za daljnja istraživanja ovog rada.

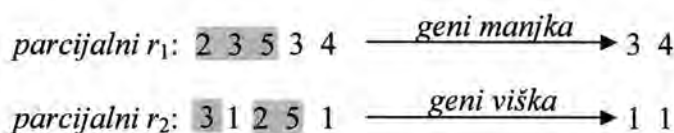
4.3.2.1. Pristup Cheng-a, Gen-a i Tsujimure

Operator križanja kojeg su definirali Cheng, Gen i Tsujimura temelji se na modificiranom predstavljanju kromosoma na bazi operacija. Modificirani pristup koristi kromosome za određivanje prioriteta svake operacije pri čemu se stvara aktivni

redosljed na temelju liste prioriteta operacija heuristikom jednog prolaza. Stoga, prilikom križanja roditelja nije nužno zadržati isti red operacija *parcijalnog redosljeda*²³ i kod potomaka.



(a)



(b)

brisanje gena viška

$p:$ 4 3 1 2 5 1 2 2

umetanje gena manjka

$p:$ 3 4 3 1 2 5 1 2 4 2

(c)

Sl. 4.10 Križanje: (a) zamjena parcijalnih redosljeda
(b) pronalaženje gena manjka i gena viška
(c) stvaranje legalnog potomka

Križanje zamjenom djelomičnih redosljeda može se opisati na slijedeći način:

1. Prvo se uzimaju dva parcijalna redosljeda od oba roditelja. Oba parcijalna redosljeda sadrže isti broj gena. Zamjenom parcijalnih redosljeda roditelja stvaraju se potomci. Na slici 4.10(a) prikazan je potomak stvoren ovom metodom

Potomak generiran na ovaj način, zamjenom parcijalnih redosljeda, može biti ilegalan, stoga je potrebno stvoriti legalan potomak brisajući gene viška i dodavajući gene manjka.

²³ Gen, Tsujimura i Kubota definiraju operator križanja zamjenom jednog dijela kromosoma roditelja, kojeg nazivaju *parcijalnim redosljedom*. Njihova je ideja bila prilikom križanja zadržati isti red operacija unutar parcijalnog redosljeda kod roditelja i potomaka.

2. Pronalaženje gena viška i gena manjka za potomak p , uspoređujući dva parcijalna redosljeda, prikazan je na slici 4.10(b).
3. Stvaranje legalnih potomaka brisanjem gena viška i slučajnim (random) umetanjem gena manjka kao što je prikazano na slici 4.10(c).

4.3.3. Mutacija

Kao što je i ranije rečeno, operator mutacije koristi se za male izmjene genetskog materijala u pravilu da se poveća raznolikost populacije, a time ubrza konvergencija genetskih algoritama. Ovdje će se objasniti tip mutacije koja će se koristiti u ovom radu.

4.3.3.1. Mutacija bazirana na traženju susjedstva

Kao što i samo ime govori, ovaj tip mutacije temelji se na tehnici traženja susjeda, a koristi se za pojačano traženje poboljšanog potomka. Za predstavljanje na bazi operacija, susjedstvo za dati kromosom može se promatrati kao skup transformiranih kromosoma datog kromosoma. Transformiranje kromosoma postiže se zamjenama nejednakih gena koji se slučajno odabiru. Optimalan je onaj mutirajući kromosom iz susjedstva koji ima najveću mjeru prilagođenosti.

Neka je dat kromosom [1 4 2 3 5 3 4 2 2 1]. Geni na pozicijama 3, 6 i 10 slučajno su odabrani. Ti geni su (2 3 1) i njihove moguće permutacije su (3 2 1), (3 1 2), (1 3 2), (1 2 3) i (2 1 3). Permutacije selektiranih gena zajedno s preostalim genima susjednih kromosoma prikazani su na slici 4.11.

<i>kromosom roditelja</i>	1 4 2 3 5 3 4 2 2 1
<i>susjedni kromosomi</i>	1 4 3 3 5 2 4 2 2 1 1 4 3 3 5 1 4 2 2 2 1 4 1 3 5 3 4 2 2 2 1 4 1 3 5 2 4 2 2 3 1 4 2 3 5 1 4 2 2 3

Sl. 4.11 Susjedni redosljedi

5. OPTIMIZACIJA REDOSLIJEDNOG ODVIJANJA PROCESA NA FPS-u

5.1. MODEL TRN-a NA FPS-u

Zadan je parcijalni zadatak/radni nalog od n dijelova različitih količina q , čije odvijanje izrade na FPS-u teče simultano tijekom određenog vremena. Za njegovu proizvodnju definiran je skup planova procesa provedbom I. faze upravljanja realiziranja radnog naloga u kojoj je proveden izbor optimalne kombinacije planova procesa primjerene strukturi radnog naloga i karakteristikama FPS-a. Svaka jedinka pojedine vrste dijelova dotičnog radnog naloga izrađuje se prema izabranom planu procesa realiziranjem skupa operacija s čvrstim predodređenim redoslijedom. Svaka operacija predstavlja zasebni element odvijanja, te u proces upravljanja ulazi sa slijedećim podacima:

- izborom stroja na kojem se operacija izvodi
- vremenom trajanja operacije
- vremenom međuoperacijskog transporta

U II. fazi upravljanja realiziranja radnog naloga potrebno je optimirati redoslijed ulaska pojedinih dijelova u proces te redoslijed izvođenja pojedine operacije na svakoj jedinki pojedine vrste dijelova prema postavljenim kriterijima optimizacije:

- što kraći ciklus izrade
- što jednakomjernije opterećenje strojeva

Pri tome treba voditi računa o slijedećim ograničenjima:

- pojedina operacija može se odvijati simultano samo na jednom stroju
- sve operacije su istog stupnja hitnosti
- operacija se ne može prekinuti
- svaki stroj može izvršavati samo jednu operaciju u datom momentu
- rokovi isporuke kao ni ciklus proizvodnje nisu predodređeni

Optimizacija problema TRN-a na FPS-u prema postavljenim kriterijima provest će se primjenom genetskog algoritma.

5.2. GENETSKI ALGORITMI ZA PROBLEM TRN-a NA FPS-u

5.2.1. Kodiranje i dekodiranje kromosoma

Od osnovne je važnosti izvršiti prikladno kodiranje kromosoma tako da kromosomi generirani u procesu evolucije produciraju ostvariv redoslijed operacija radnog naloga.

Ideja da se kromosomu kodira slijed operacija na način da svaki gen predstavlja jednu operaciju, kodiranu pomoću cjelobrojnog broja koji predstavlja indeks operacije, je odbačena. Naime, zbog postojanja ograničenja redoslijeda operacija, ne daju sve permutacije tih brojeva ostvarive redoslijede. Gen, Tsujimura i Kubota predložili su alternativu. Imenovali su sve operacije za pojedini dio radnog naloga s istim brojem koji se interpretira prema redoslijedu pojave u sekvenci kromosoma. U ovom radu, budući da postoje radni nalozi koji sadrže više jedinki dijelova iste vrste, istim brojem označit će se svaka jedinka iste vrste dijelova. Svaki od dijelova iste vrste predstavljen je operacijama s jednakim redoslijedom odvijanja i definiranim trajanjem operacija i transportnih putova.

Princip kodiranja i dekodiranja kromosoma objasniti će se na primjeru prikazanom u tablici 5.1.

Tab. 5.1 *Primjer radnog naloga s 3 vrste dijelova s različitim brojem komada*

vrsta dijelova	jedinka dijela	operacija	stroj	trajanje	transport do stroja	transport od stroja
1	1	1	1	10	5	3
	2	2	2	15	3	4
	3					
2	1	1	2	5	3	4
	2					
3	1	1	1	20	5	6

Kao što se vidi iz tablice 5.1 prva vrsta dijelova sadrži 3 jedinice od kojih svaka ima 2 operacije, druga vrsta dijelova sadrži dvije jedinice koje imaju jednu operaciju i treća vrsta dijelova ima jednu jedinku s jednom operacijom.

Svaka operacija označava se indeksom rjo gdje r označava vrstu dijela, j pojedinu jedinku dijela, a o operaciju jedinice dijela. Na primjer, indeks 123 označava treću operaciju druge jedinice prve vrste dijelova.

Ukupno postoji $3 \times 2 + 2 \times 1 + 1 \times 1 = 9$ operacija. Budući da postoje 3 vrste dijelova geni kromosoma su brojevi 1, 2 i 3. Prva vrsta dijelova r_1 sudjeluje s 6 operacija pa će u kromosomu biti 6 jedinica, dok druga vrsta dijelova r_2 sudjeluje s 2 operacije pa će u kromosomu biti 2 dvojke. Treća vrsta dijelova r_3 sudjeluje s jednom operacijom, stoga će u kromosomu biti 1 trojka. Ukupno ima 9 operacija, pa je i duljina kromosoma 9 gena.

Uzme li se npr. kromosom [1 2 1 3 2 1 1 1 1] slijedi:

- 1 – prva vrsta dijela, prva jedinka dijela, prva operacija
- 2 – druga vrsta dijela, prva jedinka dijela, prva operacija (zadnja za jedinku dijela)
- 1 – prva vrsta dijela, prva jedinka dijela, druga operacija (zadnja za jedinku dijela)
- 3 – treća vrsta dijela, prva (i jedina jedinka dijela), prva operacija (zadnja za jedinku)
- 2 – druga vrsta dijela, druga jedinka dijela, prva operacija (zadnja za jedinku dijela)
- 1 – prva vrsta dijela, druga jedinka dijela, prva operacija
- 1 – prva vrsta dijela, druga jedinka dijela, druga operacija (zadnja za jedinku dijela)
- 1 – prva vrsta dijela, treća jedinka dijela, prva operacija
- 1 – prva vrsta dijela, treća jedinka dijela, druga operacija (zadnja za jedinku dijela)

Apsolutni indeksi operacija su:

- 1 – prva vrsta dijela, prva jedinka dijela, prva operacija
- 2 – prva vrsta dijela, prva jedinka dijela, druga operacija
- 3 – prva vrsta dijela, druga jedinka dijela, prva operacija
- 4 – prva vrsta dijela, druga jedinka dijela, druga operacija
- 5 – prva vrsta dijela, treća jedinka dijela, prva operacija
- 6 – prva vrsta dijela, treća jedinka dijela, druga operacija
- 7 – druga vrsta dijela, prva jedinka dijela, prva operacija
- 8 – druga vrsta dijela, druga jedinka dijela, prva operacija
- 9 – treća vrsta dijela, prva jedinka dijela, prva operacija

Stoga nakon dekodiranja pravi izvedivi redoslijed operacija je za dani kromosom slijedeći:

Kromosom	1	2	1	3	2	1	1	1	1
Redoslijed (prioritet operacija)	1	7	2	9	8	3	4	5	6

Zatim se stvara redoslijed:

Najranije vrijeme za operaciju s indeksom 1 (budući da su svi strojevi slobodni i radi se o prvoj operaciji, jedini je uvjet transport do stroja na kojem se izvodi operacija)

Najranije vrijeme za početak operacije 7 (radi se o prvoj operaciji, jedan od uvjeta je transport do stroja na kojem se izvodi operacija 7 i ako vrsta dijela 1 i vrsta dijela 2 dijele isti stroj postoji i utjecaj zauzeća tog stroja od operacija jedinki vrste dijelova 1)

Najranije vrijeme za operaciju s indeksom 2 (budući da se radi o istom izratku, ograničavajuće vrijeme za početak operacije je završetak prethodne operacije 1 + vrijeme transporta s prethodnog stroja na kojem se izvodila operacija 1 do stroja na kojem se izvodi operacija 2) .

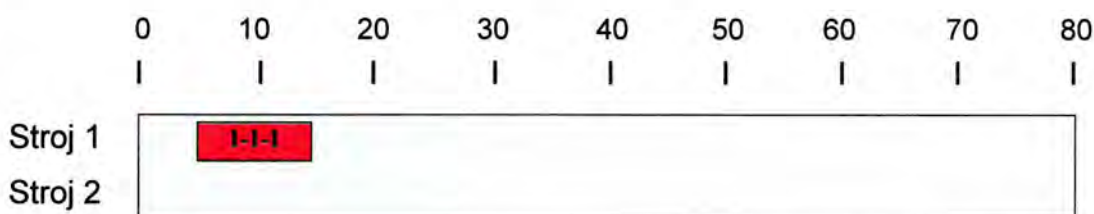
Najranije vrijeme za početak operacije 9 (radi se o prvoj i jedinoj operaciji, traži veće od vremena potrebnog za transport do stroja na kojem se izvodi operacija, i dostupnosti stroja koji je možda zauzet prethodnim operacijama).

...

Iz prethodnog slijedi, da pojedina operacija koja je u kromosomu navedena kasnije, može biti izvršena prije neke koja se u kromosomu javlja ranije. Ovim je algoritmom provedeno pomicanje ulijevo istovremeno s kreiranjem redoslijeda operacija i provedeno terminiranje navedenog radnog naloga.

1. Operacija 1

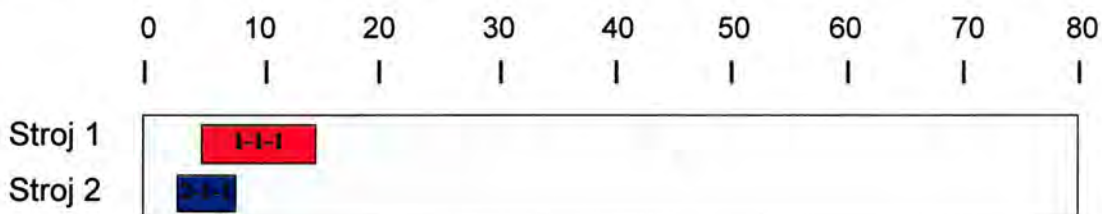
1 – prva vrsta dijela, prva jedinka dijela, prva operacija



Sl. 5.1 Terminiranje operacije s indeksom 1

2. Operacija 7

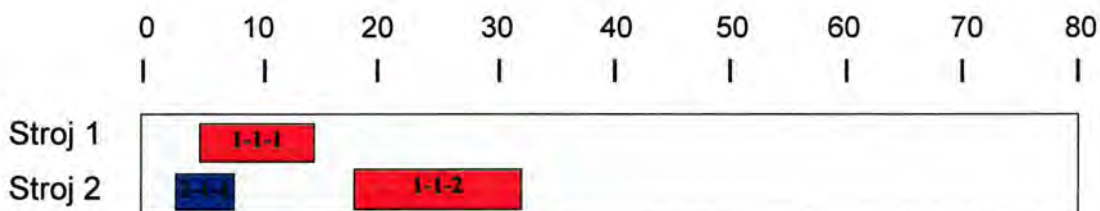
2 – druga vrsta dijela, prva jedinka dijela, prva operacija (zadnja za jedinku dijela)



Sl. 5.2 Terminiranje operacije s indeksom 7

3. Operacija 2

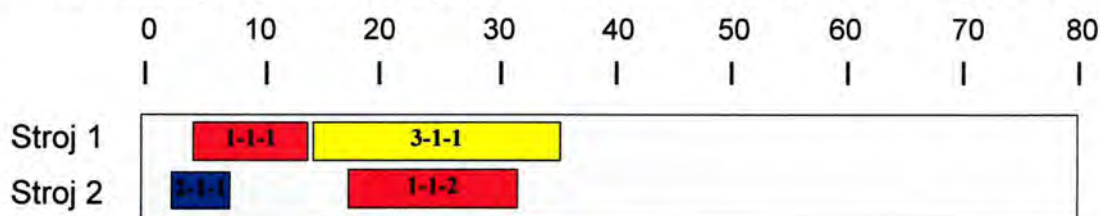
1 – prva vrsta dijela, prva jedinka dijela, druga operacija (zadnja za jedinku dijela)



Sl. 5.3 Terminiranje operacije s indeksom 2

4. Operacija 9

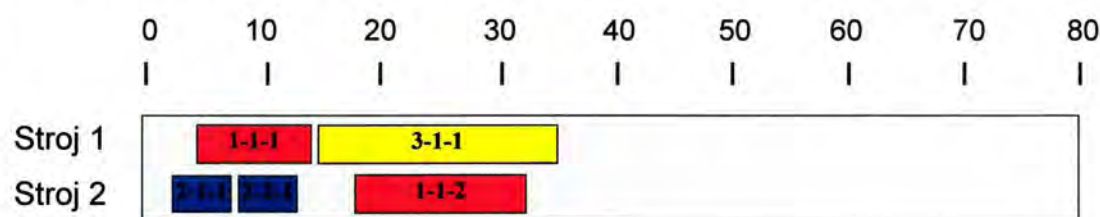
3 – treća vrsta dijela, prva (i jedina jedinka dijela), prva operacija (zadnja za jedinku)



Sl. 5.4 Terminiranje operacije s indeksom 9

5. Operacija 8

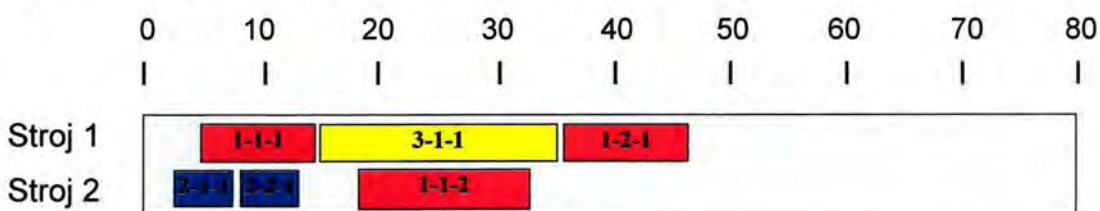
2 – druga vrsta dijela, druga jedinka dijela, prva operacija (zadnja za jedinku dijela)



Sl. 5.5 Terminiranje operacije s indeksom 8

6. Operacija 3

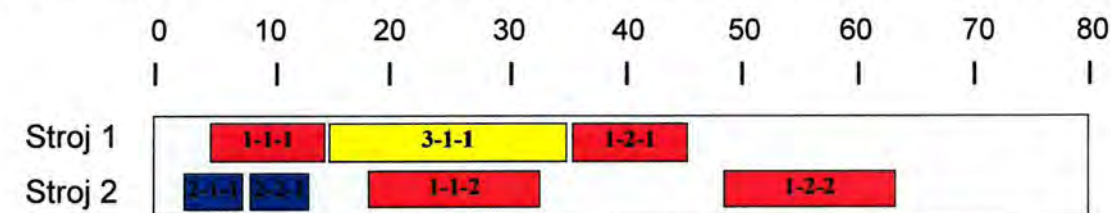
1 – prva vrsta dijela, druga jedinka dijela, prva operacija



Sl. 5.6 Terminiranje operacije s indeksom 3

7. Operacija 4

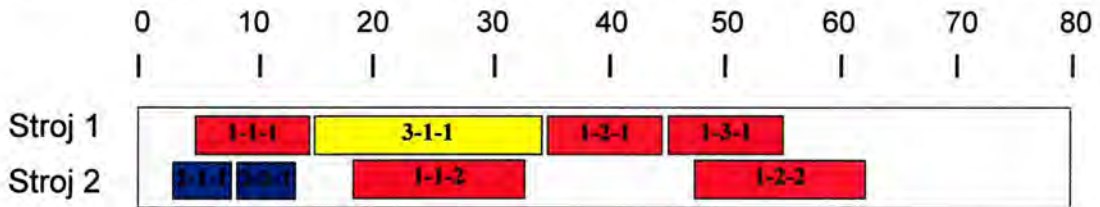
1 – prva vrsta dijela, druga jedinka dijela, druga operacija (zadnja za jedinku dijela)



Sl. 5.7 Terminiranje operacije s indeksom 4

8. Operacija 5

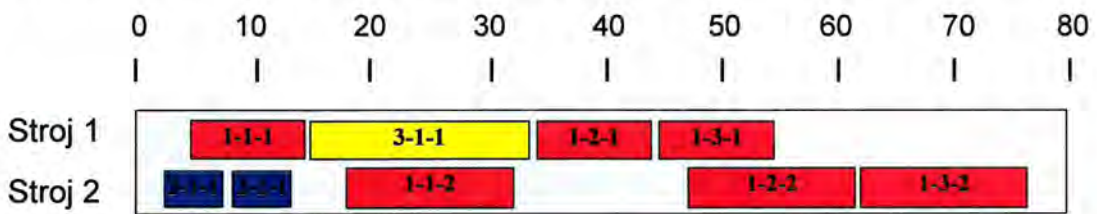
1 – prva vrsta dijela, treća jedinka dijela, prva operacija



Sl. 5.8 Terminiranje operacije s indeksom 5

9. Operacija 6

1 – prva vrsta dijela, treća jedinka dijela, druga operacija (zadnja za jedinku dijela)



Sl. 5.9 Terminiranje operacije s indeksom 6

Važno je naglasiti da je terminiranje navedenog radnog naloga, te kreiranje redoslijeda operacija za primjer prikazan u tablici 5.1 izvedeno uz temeljnu pretpostavku da prebacivanje izratka iz izlaznog međuskladišta stroja na transporter ne ovisi o spremnosti transportera, već jedino o spremnosti stroja na kojem će biti izvedena naredna operacija izratka. To bi značilo da je transporter raspoloživ u svakom trenutku, što je naravno u suprotnosti sa stvarnim slučajem.

Uzme li se u obzir ograničenje transportnog sustava koje glasi:

- transporter može transportirati samo jednu paletu s jednim izratkom u određenom trenutku
- transport se može odvijati u dva smjera
- izradak po dolasku ulazi u radni prostor stroja
- paleta s izratkom nakon izvršenja operacije ulazi u izlazno međuskladište stroja koje može prihvatiti više paleta
- redoslijed izuzimanja paleta iz izlaznog međuskladišta stroja je slobodan
- kod stroja na kojem se izvodi zadnja operacija jedinice dijela, paleta će biti prebačena na transporter u prvom slobodnom trenutku

kromosom iz primjera prikazanim u tablici 5.1 mora se proširiti na način da se dodaju *lažne* operacije za svaku jedinku dijela. To su operacije čije je vrijeme izvođenja jednako

nuli, a mjesto izvođenja ustvari skladište gotovih dijelova. Uvođenje *lažnih* operacija bitno je iz razloga računanja transportnog vremena od zadnje operacije jedinice dijela do skladišta gotovih dijelova.

Budući da postoje 3 vrste dijelova geni kromosoma su brojevi 1,2 i 3. Prva vrsta dijelova r_1 sudjeluje s 6 stvarnih operacija + 3 *lažne* operacije pa će u kromosomu biti 9 jedinica, dok druga vrsta dijelova r_2 sudjeluje s 2 operacije + 2 *lažne* operacije pa će u kromosomu biti 4 dvojke. Treća vrsta dijelova r_3 sudjeluje s jednom operacijom + 1 *lažna* operacija, stoga će u kromosomu biti 2 trojke. Ukupno ima 9 stvarnih operacija + 6 *lažnih* operacija, pa je i duljina kromosoma 15 gena.

Kromosom [1 2 1 3 2 1 1 1 1] za primjer prikazan u tablic 5.1 kad nije uzeto u obzir ograničenje transportnog sustava, proširuje se za *lažne* operacije, te za slučaj kad se uzima u obzir ograničenje transportnog sustava postaje [1 2 1 2 3 2 3 1 1 1 1 2 1 1].

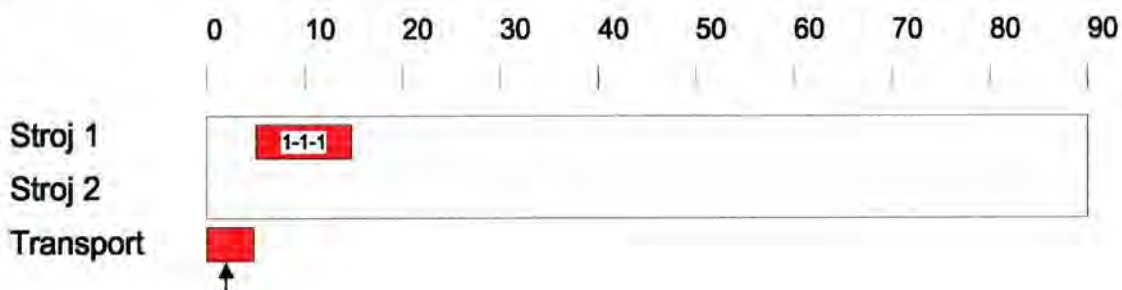
Apsolutni indeksi operacija sada su:

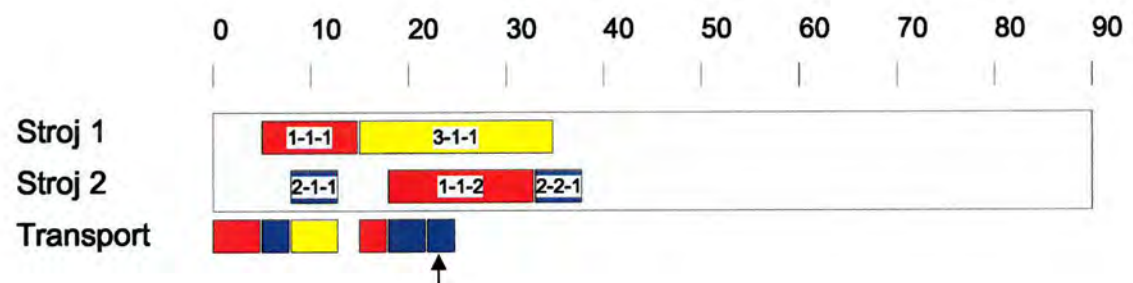
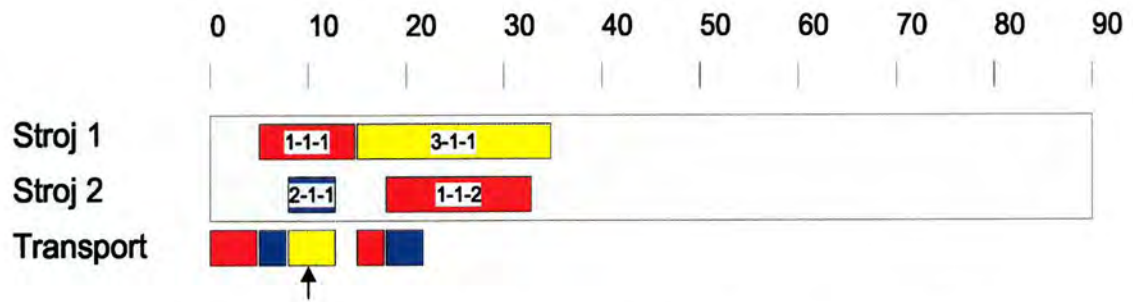
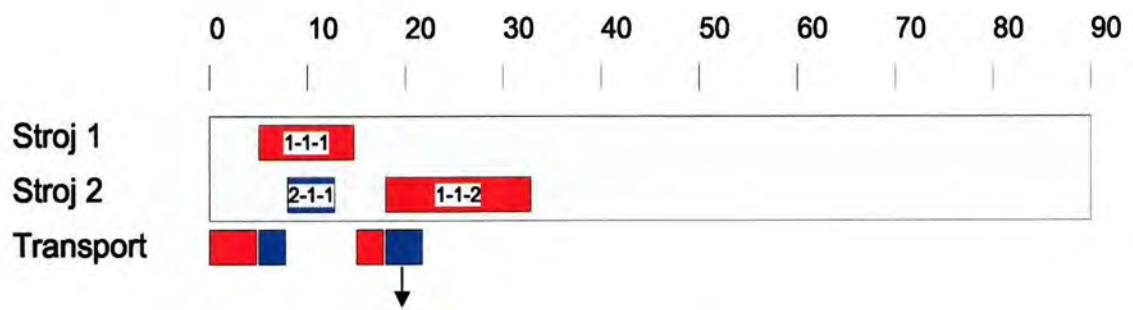
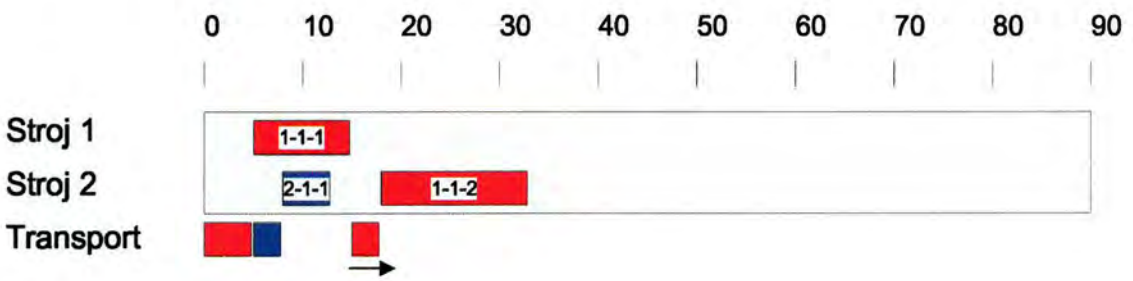
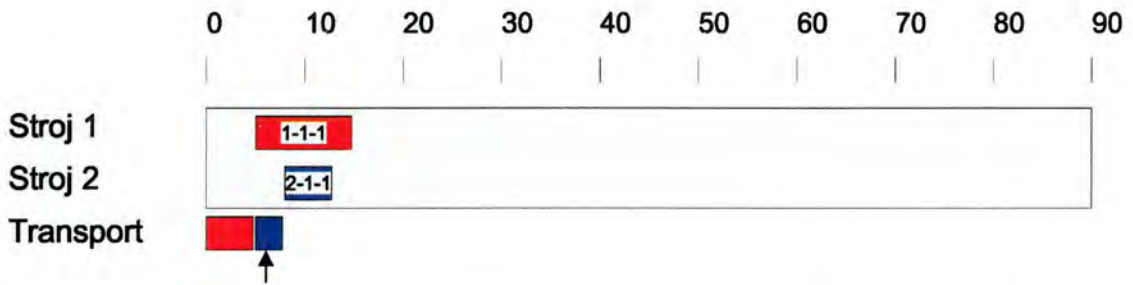
- 1 – prva vrsta dijela, prva jedinka dijela, prva operacija
- 2 – prva vrsta dijela, prva jedinka dijela, druga operacija
- 3 – prva vrsta dijela, prva jedinka dijela, treća *lažna* operacija
- 4 – prva vrsta dijela, druga jedinka dijela, prva operacija
- 5 – prva vrsta dijela, druga jedinka dijela, druga operacija
- 6 – prva vrsta dijela, druga jedinka dijela, treća *lažna* operacija
- 7 – prva vrsta dijela, treća jedinka dijela, prva operacija
- 8 – prva vrsta dijela, treća jedinka dijela, druga operacija
- 9 – prva vrsta dijela, treća jedinka dijela, treća *lažna* operacija
- 10 – druga vrsta dijela, prva jedinka dijela, prva operacija
- 11 – druga vrsta dijela, prva jedinka dijela, druga *lažna* operacija
- 12 – druga vrsta dijela, druga jedinka dijela, prva operacija
- 13 – druga vrsta dijela, druga jedinka dijela, druga *lažna* operacija
- 14 – treća vrsta dijela, prva jedinka dijela, prva operacija
- 15 – treća vrsta dijela, prva jedinka dijela, druga *lažna* operacija

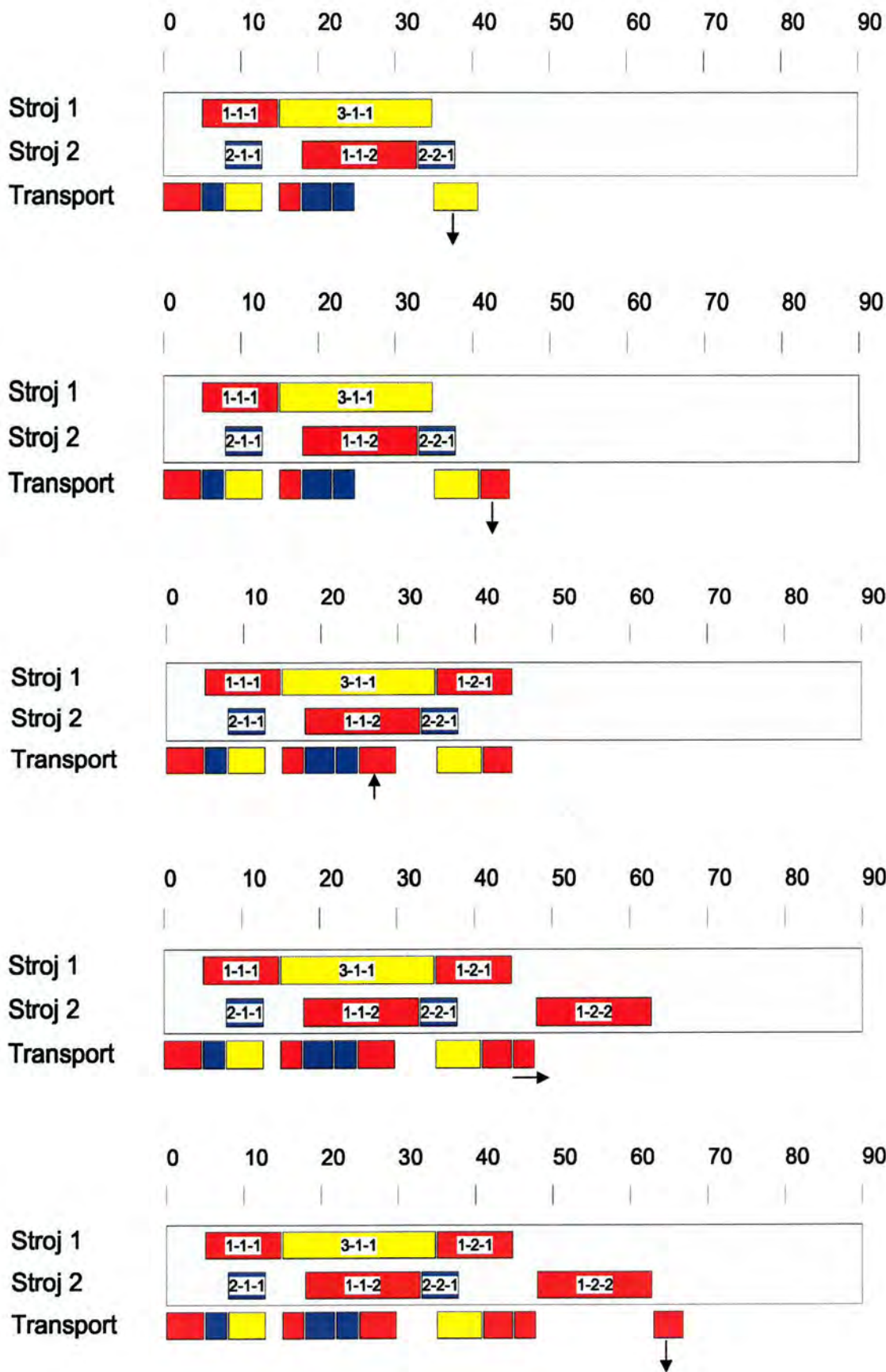
Stoga nakon dekodiranja pravi izvedivi redoslijed operacija je za dani kromosom slijedeći:

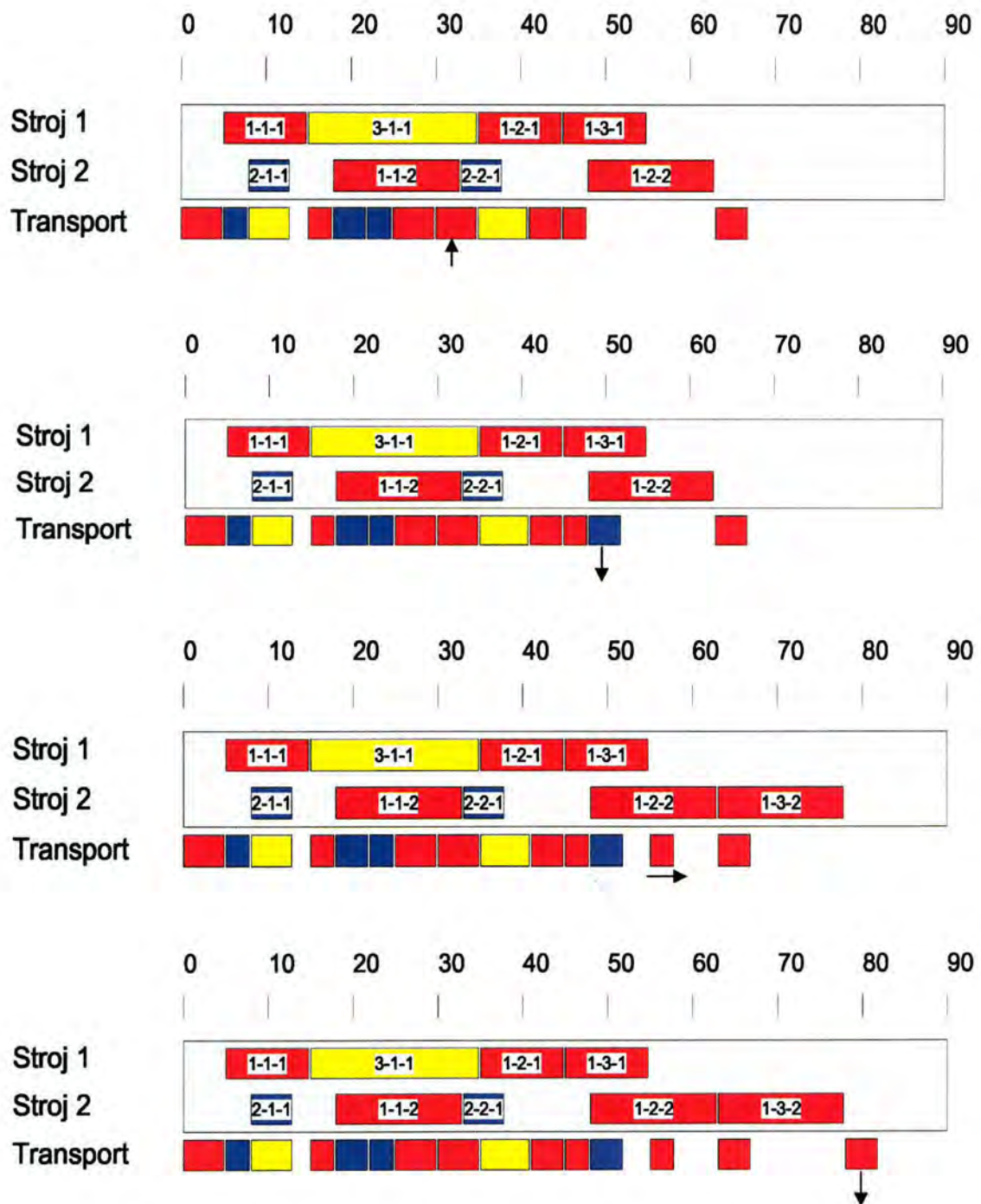
Kromosom [1 2 1 2 3 2 3 1 1 1 1 2 1 1]
 Redoslijed (prioritet operacija) [1 10 2 11 14 12 15 3 4 5 6 7 13 8 9]

Kreiranje redoslijeda operacija i terminiranje radnog naloga uzevši u obzir ograničenja transportnog sustava za primjer iz tablice 5.1, prikazano je na slici 5.10. Pojedine transportne aktivnosti na slici 5.10 prikazane su na slijedeći način: ↑ ulaz dijela, ↓ izlaz dijela, → transport dijela.









Sl. 5.10 Terminiranje operacija za dekodirani kromosom
[1 10 2 11 14 12 15 3 4 5 6 7 13 8 9]

Usporedi li se terminiranje operacija prikazano na slici 5.10 s terminiranjem operacija prikazanim na slikama 5.1, 5.2, 5.3, 5.4, 5.5, 5.6, 5.7, 5.8, 5.9 može se primjetiti da je terminiranje 1.operacije 2.jedinke 2.vrste dijelova izvedeno na drugačiji način, dok se za ostale operacije podudara za oba slučaja. Za slučaj terminiranja radnog naloga prikazan na slici 5.10, 1.operaciju 2.jedinke 2.vrste dijelova nije moguće izvoditi

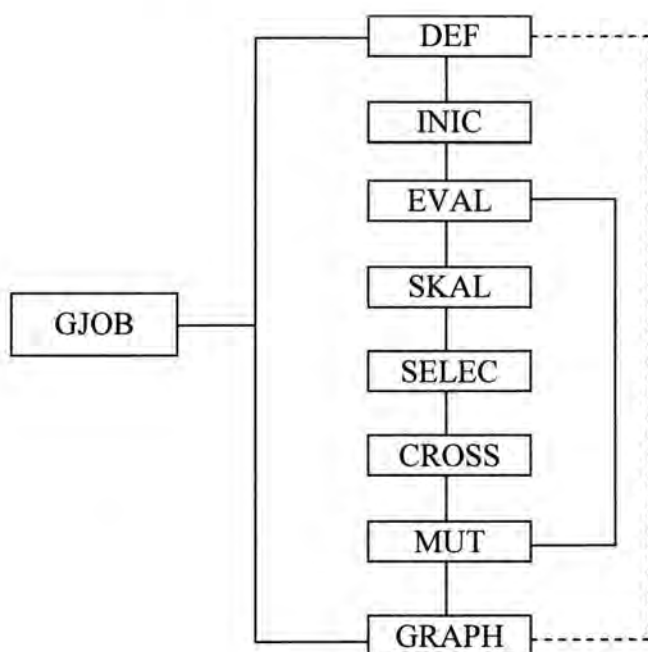
na stroju 2 odmah nakon završetka 1.operacije 1.jedinke 2.vrste dijelova kao što je to prikazano na slici 5.5 jer u tom trenutku transportno vozilo nije slobodno, te ju je moguće rasporediti tek iza 2.operacije 1.jedinke 1.vrste dijelova, što u pravilu implicira pomicanjem operacija na desno, a rezultat toga je dulji ciklus izrade.

Na temelju izloženog može se primjetiti da će dekodiranje optimalnog kromosoma u slučaju kad je uzeto u obzir ograničenje transportnog sustava rezultirati uvijek duljim ciklusom izrade.

5.3. RAČUNALNI PROGRAM GJOB

Razvijen je program GJOB za elektroničko računalo namijenjen za rješavanje odgovarajućih zadataka II. faze upravljanja realiziranjem radnog naloga odnosno optimalnog upravljanja radnih naloga na FPS-u. Program se temelji na metodi genetskih algoritama, a razvijen je pomoću Microsoft Visual C++ Developer Studio-a.

Organizacijska struktura programa GJOB koncipirana je u obliku glavnog programa i 8 potprograma, a prikazana je na slici 5.11. Dijagram tijeka programa prikazan je na slici 5.12.



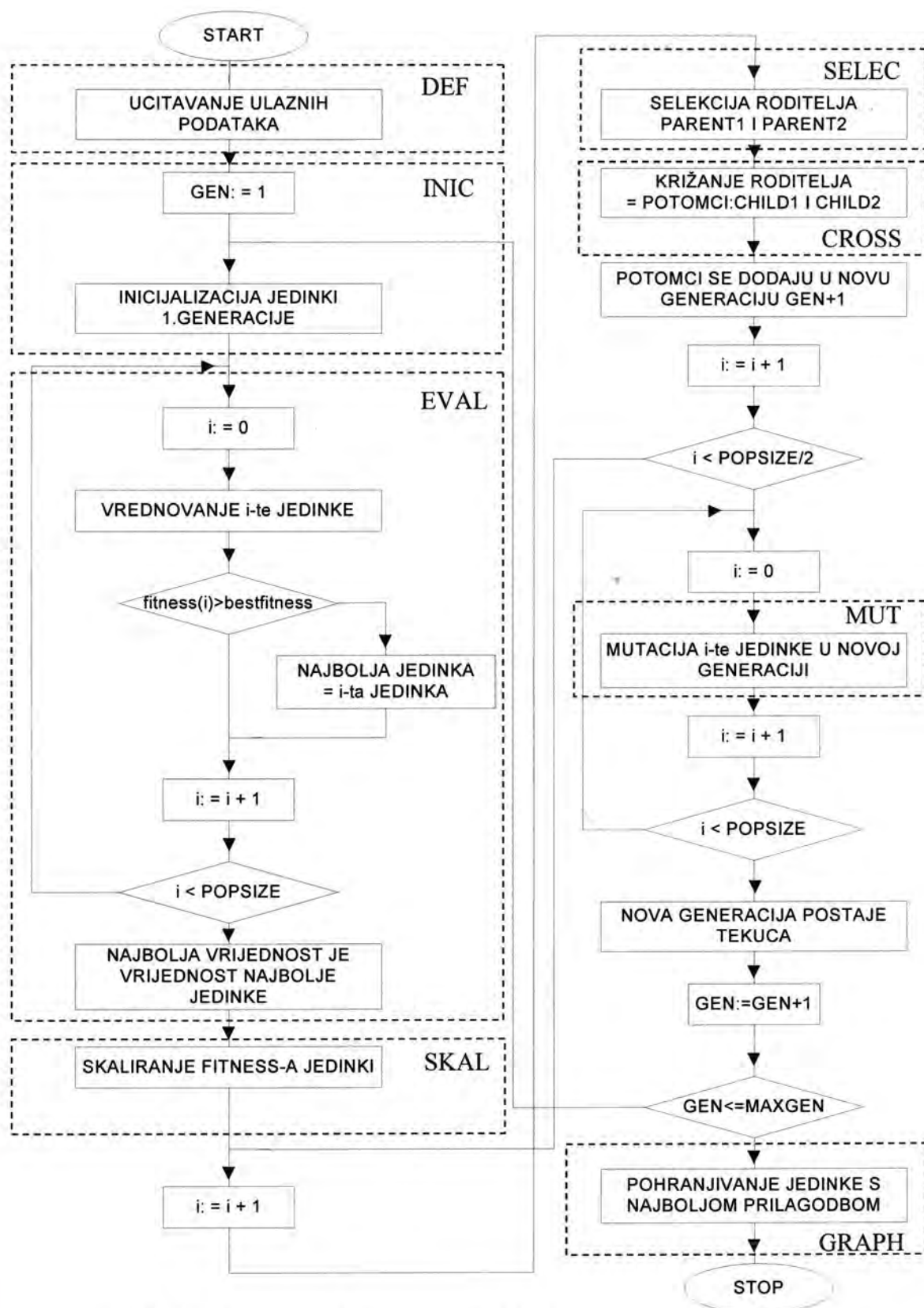
Sl. 5.11 Organizacijska struktura programa GJOB

U nastavku je dan kratki opis rada programa i potprograma.

5.3.1. Opis potprograma

Potprogramom DEF učitavaju se ulazni podaci kao što su: broj vrste dijelova, broj jedinki dijelova pojedine vrste dijelova, broj operacija pojedine jedinice dijela, vrijeme trajanja operacija, vrijeme transporta do stroja, vrijeme transporta od stroja te se definira redoslijed izvođenja operacija po strojevima pojedinog dijela za svaku vrstu dijela. Lista je ulaznih parametara:

NumberOfProducts	- broj vrsta dijelova
NumberOfParts	- broj jedinki dijelova iste vrste
NumberOfMachines	- broj strojeva FPS-a
NumberOfOperations	- broj operacija za pojedinu jedinku
OperationDuration	- vrijeme trajanja operacije
OperationTTimeBefore	- vrijeme transporta do stroja
OperationTTimeAfter	- vrijeme transporta od stroja



Sl. 5.12 Dijagram tijeka programa GJOB

Nakon učitavanja ovih podataka učitavaju se parametri genetskog algoritma:

maxgen	- maksimalni broj generacija
popSize	- veličina populacije (broj jedinki u populaciji)
pcross	- vjerojatnost križanja jedinki
pmutation	- vjerojatnost mutacije jedinki

Potprogramom INIC vrši se inicijalizacija jedinki, tj. predstavlja se rješenje problema u obliku kromosoma na način opisan u potpoglavlju 5.2.1.

Potprogram EVAL računa prilagođenost jedinke, tj. za svaku jedinku iz njenog fenotipa – pripadnog skupa parametara izračunava vrijednost funkcije cilja, te zaključuje da li slabije ili bolje od drugih jedinki zadovoljava kriterije koji su postavljeni. Radi kvantitativnog uspoređivanja jedinki računa se mjera prilagođenosti jedinke, te se pronalaze jedinke s najvećom prilagođenosti.

U genetskim algoritmima ovako dobivenu prilagođenost smatra se sirovom, naime u prvim generacijama velika je većina jedinki slabo prilagođena i samo nekoliko jedinki odskaače od prosjeka što bi rezultiralo samo njihovim potomcima u idućoj generaciji, a u kasnijim generacijama osrednje i bolje jedinke dobivale bi sličan udio u potomstvu i konvergencija rješenja bi se usporila.

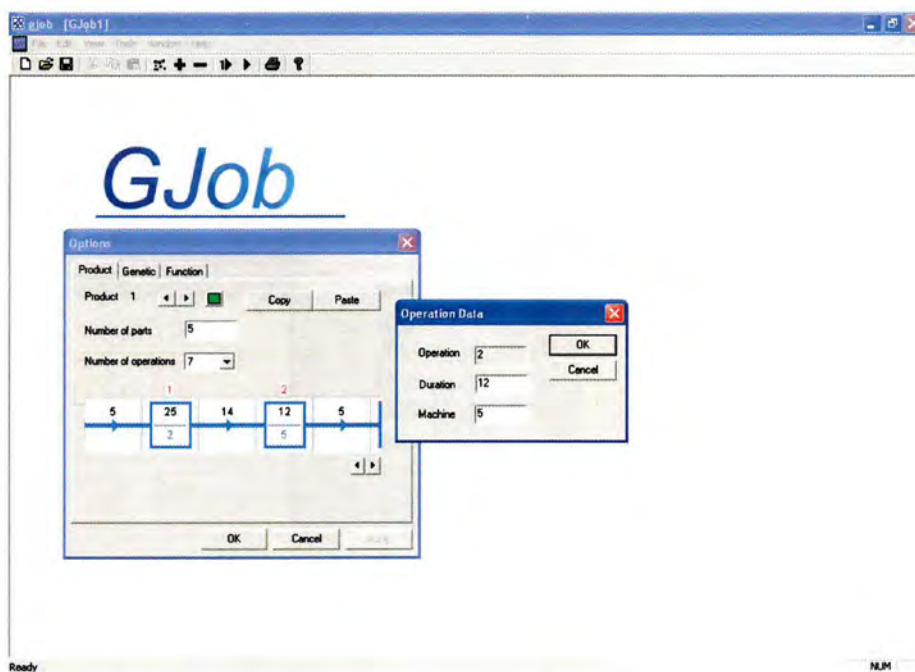
Potprogramom SKAL rješava se ovaj problem uvođenjem skalirane prilagođenosti koja za razliku od sirove treba u svim generacijama zadržati sličan omjer između svoje najveće i prosječne vrijednosti. Skalirana prilagođenost u ovom potprogramu računa se prema izrazu (4.3).

Potprogramom SELEC stvaraju se jedinke nove generacije na način da se među jedinkama stare generacije biraju roditelji. Upravo ovdje dolazi do izražaja oponašanje prirodnog odabira jer jedinke koje su bolje prilagođene od ostalih jedinki iste populacije žive dulje i imaju više potomaka. Budući da se ne zna što bi to bila apsolutno najbolja prilagođenost, odabir se radi stohastički pomoću kotača ruleta i vjerojatnost da neka jedinka bude odabrana kao roditelj jednaka je njenoj relativnoj prilagođenosti, izraz (4.5).

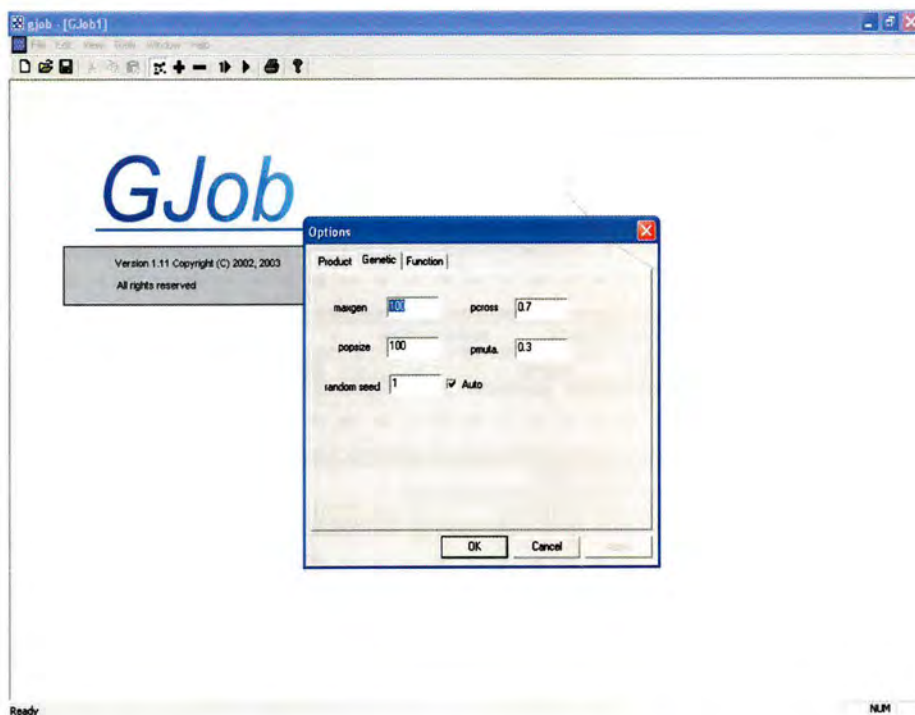
Prijenos genetskog materijala sa starih na nove jedinke križanjem obavlja se potprogramom CROSS na način opisan u potpoglavlju 4.3.2. a s odabranom vjerojatnošću križanja *pcross*.

Napredovanjem kroz generacije smanjuje se raznolikost genetskog materijala što može dovesti do usporavanja konvergencije genetskog algoritma i nagomilavanja populacije oko lažnog optimuma. Potprogram MUT služi za povećanje raznolikosti ukupnog genetskog materijala populacije uvodeći mutacije temeljene na tehnicima traženja susjedstva opisane u potpoglavlju 4.3.3.

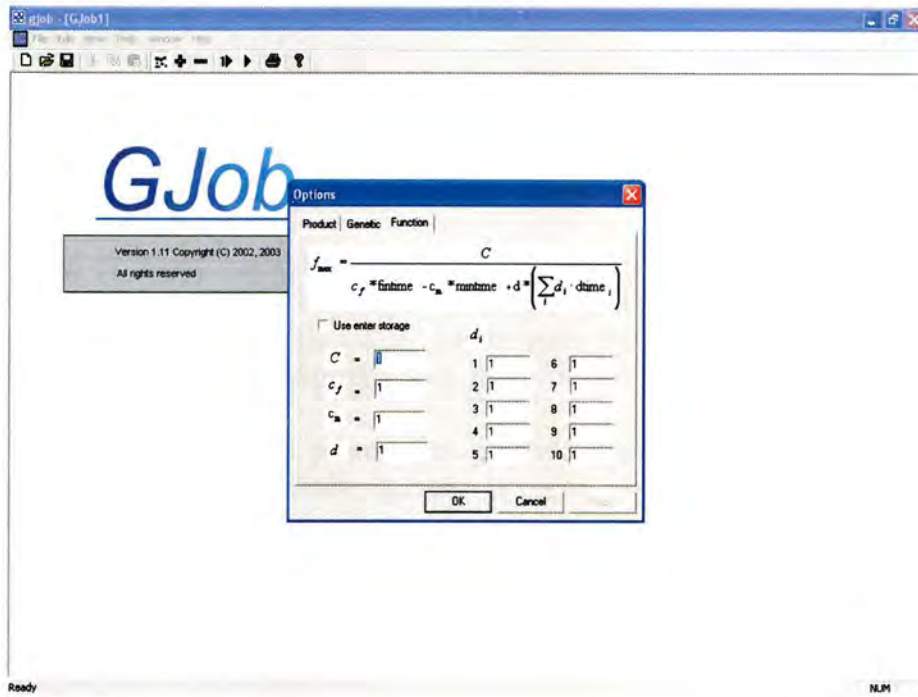
Potprogram GRAPH služi za jednostavno korištenje programa, naime omogućava korisniku jednostavno upisivanje ulaznih parametara, slika 5.13, 5.14 i 5.15, jednostavnu izmjenu različitih podataka, te pregled grafičkih rezultata, slika 5.16.



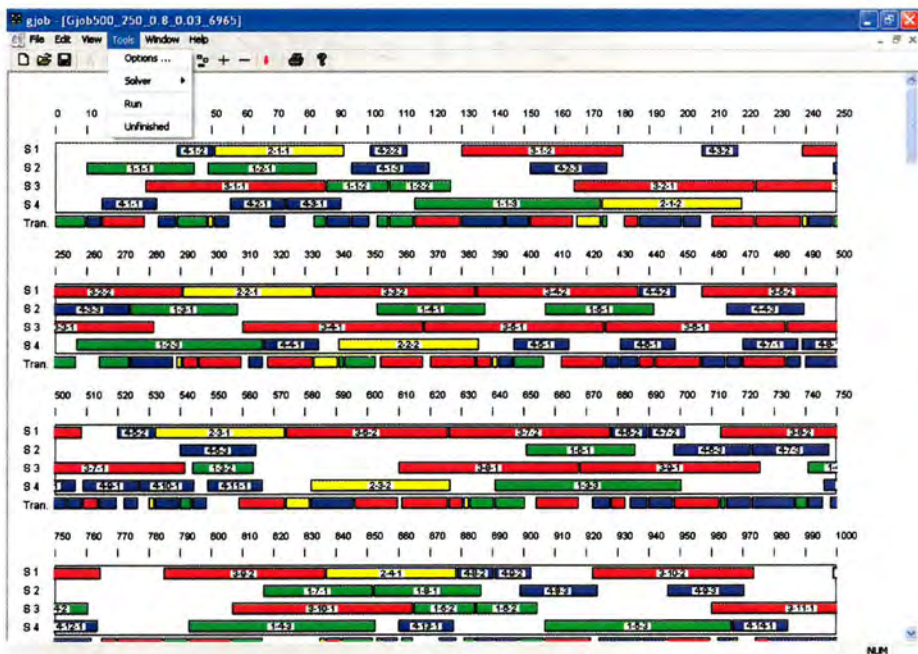
Sl. 5.13 “Options” izbornik programa GJOB (upisivanje ulaznih podataka)



Sl. 5.14 “Options” izbornik programa GJOB (upisivanje parametara GA)



Sl. 5.15 "Options" izbornik programa GJOB (odabir kriterija optimizacije)



Sl. 5.16 Prikaz rezultata programa GJOB

5.3.2. Primjeri

U cilju testiranja kvalitete genetskog algoritma prikazanog u poglavlju 5. i implementiranog u program GJOB, u ovom će se dijelu rada prikazati tri dobro poznata problema MT06, MT10, MT20 postavljena po prvi put od Muth i Thompson-a 1963. godine, a egzaktno riješena 1989. godine od Carlier i Pinson-a branch and bound metodom.

Sva tri primjera spadaju u klasične TRN probleme gdje je model postavljen na način: n dijelova koji se sastoje od skupa operacija s predodređenim redoslijedom, treba se obraditi na m strojeva tako da ciklus izrade bude što kraći. Pri tom je zanemaren problem transporta odnosno pretpostavljeno je da transport ne vodi odlaganju početka izvršenja operacije.

Na svim je primjerima primijenjen genetski algoritam uz stohastički odabir s zamjenom (odabir pomoću kotača ruleta), linearno skaliranje, križanje zamjenom parcijalnih redoslijeda opisanim u potpoglavlju 4.3.2.1., te mutacijom baziranom na traženju susjedstva opisanim u potpoglavlju 4.3.3.1.

5.3.2.1. MT06 problem

Primjer problema 6 izradaka po 1 komad - 6 strojeva prikazan je u tablici 5.2.

Tab. 5.2 MT06 problem

Vrijeme izvršavanja operacije		Redoslijed strojeva										
dio	Operacije						Operacije					
	1	2	3	4	5	6	1	2	3	4	5	6
j_1	1	3	6	7	3	6	3	1	2	4	6	5
j_2	8	5	10	10	10	4	2	3	5	6	1	4
j_3	5	4	8	9	1	7	3	4	6	1	2	5
j_4	5	3	5	3	8	9	2	1	3	4	5	6
j_5	9	3	5	4	3	1	3	2	5	6	1	4
j_6	3	3	9	10	4	1	2	4	6	1	5	3

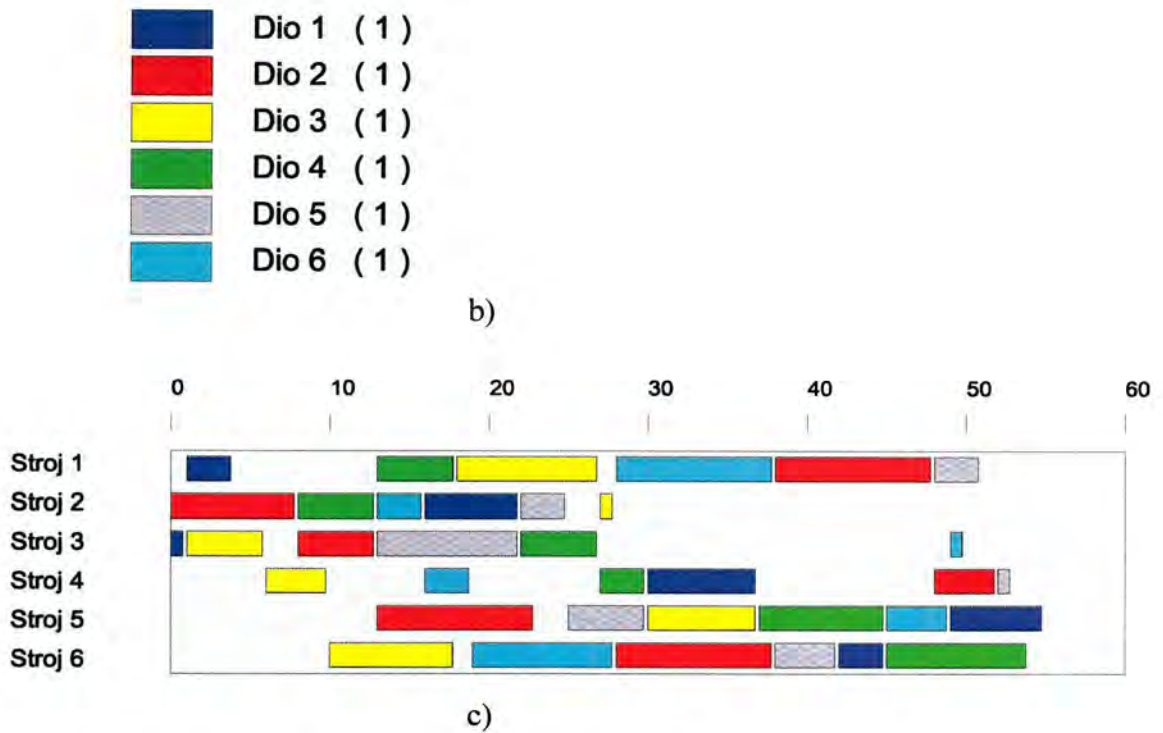
Rezultati dobiveni primjenom genetskog algoritma uz 36 gena po kromosomu, 10 kromosoma u populaciji ($popSize = 10$), jednu populaciju, 50 generacija ($maxgen = 50$), vjerojatnost križanja $pcross = 0.65$, te vjerojatnost mutacije $pmutation = 0.3$, prikazani su na slici 5.17.

Maksimalno vrijeme = 55

Broj generacija = 50

Veličina populacije = 10

a)



Sl. 5.17 Prikaz rezultata problema MT06 programom GJOB

- a) rezultat
b) legenda
c) gantogram strojeva

U tablici 5.3 prikazani su rezultati problema MT06 za različito postavljene parametre genetskog algoritma.

Tab. 5.3 Parametri genetskog algoritma za MT06 problem

	<i>popSize</i>	<i>maxgen</i>	<i>pcross</i>	<i>pmutation</i>	<i>rezultat</i>
1	10	50	0.65	0.3	55
2	10	50	0.65	0.03	61
3	10	200	0.65	0.03	61
4	20	50	0.65	0.03	55
5	10	50	1.00	0	55
6	10	50	1.00	1.00	59

Problem MT06 relativno je jednostavan i postavljanje parametara ne predstavlja neki problem da bi se dobio optimalni rezultat.

Iz tablice 5.3 može se vidjeti da smanjivanje parametra mutacije (2. red tablice) daje lošiji rezultat od optimalnog, kojeg se ne može popraviti niti s većim brojem generacija (3. red tablice), već samo tako da se poveća populacija (4. red tablice).

5.3.2.2. MT10 problem

Primjer problema 10 izradaka po 1 komad - 10 strojeva prikazan je u tablici 5.4.

Tab. 5.4 MT10 problem

dio	Vrijeme izvršavanja operacije										Redosljed strojeva									
	Operacije										Operacije									
	1	2	3	4	5	6	7	8	9	10	1	2	3	4	5	6	7	8	9	10
j_1	29	78	9	36	49	11	62	56	44	21	1	2	3	4	5	6	7	8	9	10
j_2	43	90	75	11	69	28	46	46	72	30	1	3	5	10	4	2	7	6	8	9
j_3	91	85	39	74	90	10	12	89	45	33	2	1	4	3	9	6	8	7	10	5
j_4	81	95	71	99	9	52	85	98	22	43	2	3	1	5	7	9	8	4	10	6
j_5	14	6	22	61	26	69	21	49	72	53	3	1	2	6	4	5	9	8	10	7
j_6	84	2	52	95	48	72	47	65	6	25	3	2	6	4	9	10	1	7	5	8
j_7	46	37	61	13	32	21	32	89	30	55	2	1	4	3	7	6	10	9	8	5
j_8	31	86	46	74	32	88	19	48	36	79	3	1	2	6	5	7	9	10	8	4
j_9	76	69	76	51	85	11	40	89	26	74	1	2	4	6	3	10	7	8	5	9
j_{10}	85	13	61	7	64	76	47	52	90	45	2	1	3	7	9	10	6	4	5	8

Rezultati dobiveni primjenom genetskog algoritma uz 100 gena po kromosomu, 100 kromosoma u populaciji ($popSize = 100$), jednu populaciju, 150 generacija ($maxgen = 150$), vjerojatnost križanja $pcross = 0.65$, te vjerojatnost mutacije $pmutation = 0.5$, prikazani su na slici 5.18.

Maksimalno vrijeme = 937

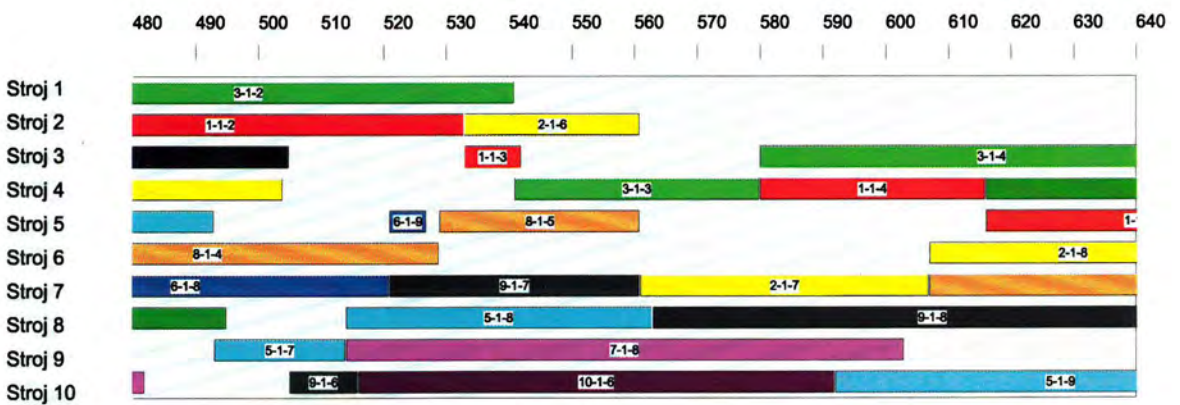
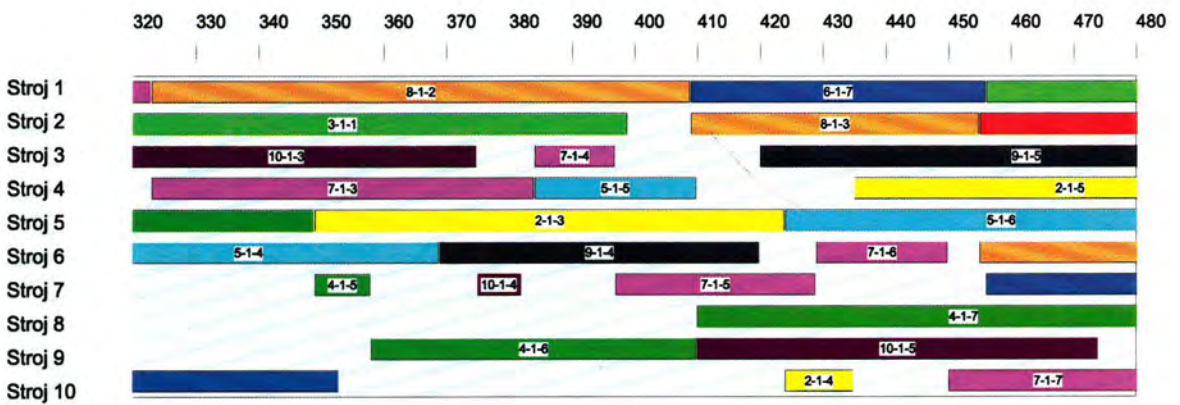
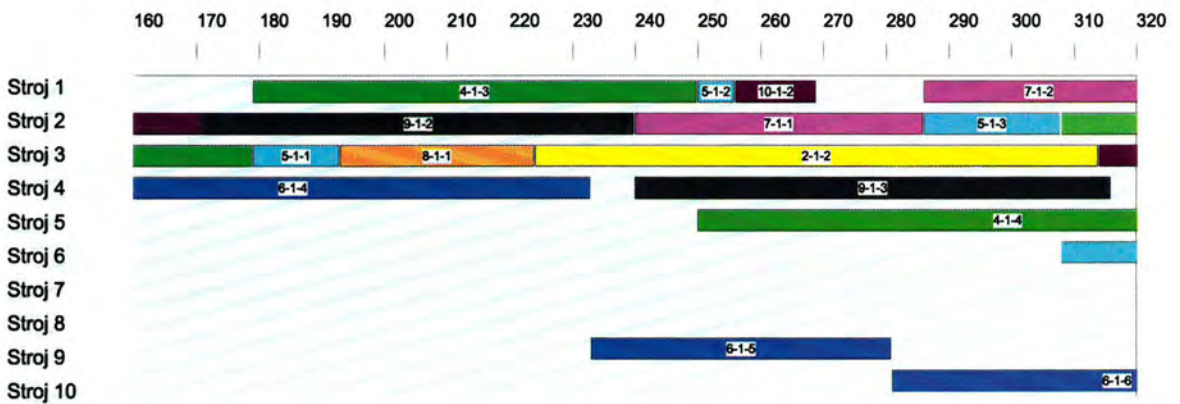
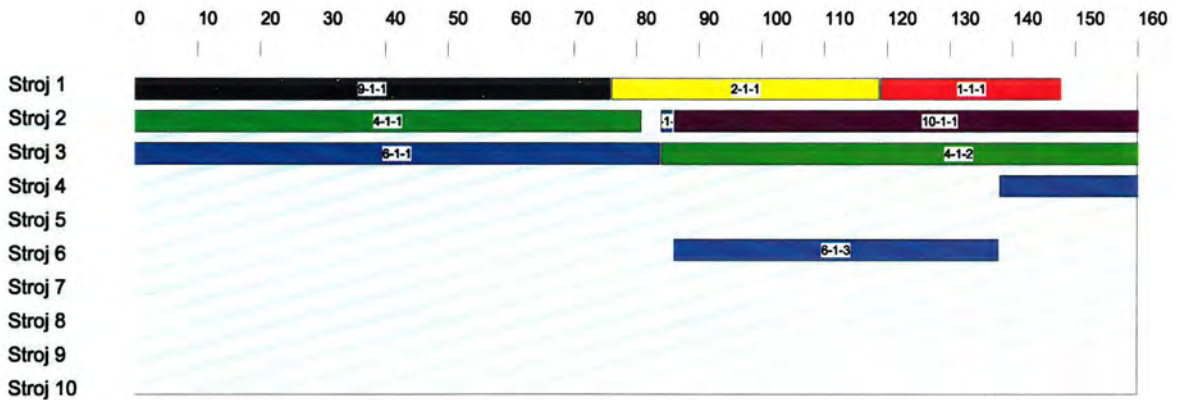
Broj generacija = 150

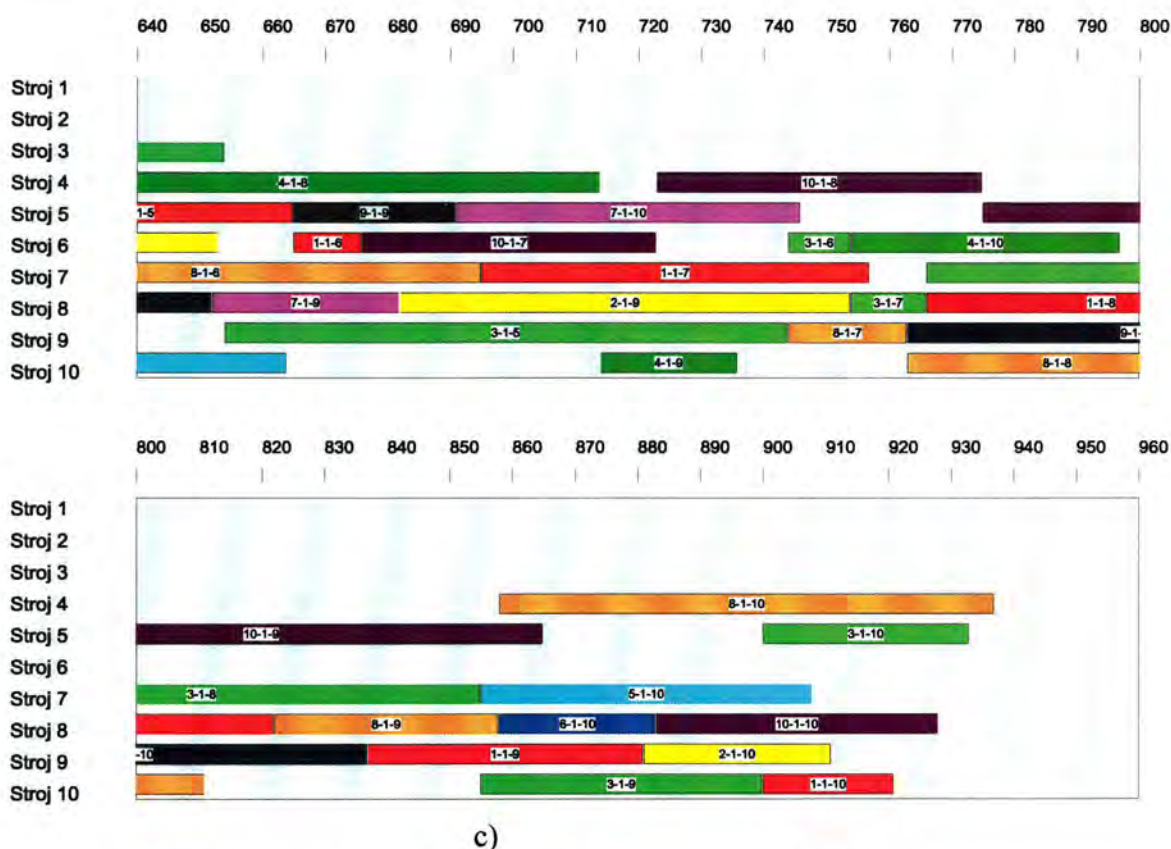
Veličina populacije = 100

a)

	Dio 1 (1)		Dio 6 (1)
	Dio 2 (1)		Dio 7 (1)
	Dio 3 (1)		Dio 8 (1)
	Dio 4 (1)		Dio 9 (1)
	Dio 5 (1)		Dio 10 (1)

b)





c)

SI.5.18 Prikaz rezultata problema MT10 programom GJOB

- rezultat
- legenda
- gantogram strojeva

U tablici 5.5 prikazani su rezultati problema MT10 za različito postavljene parametre genetskog algoritma.

Tab. 5.5 Parametri genetskog algoritma za MT10 problem

	<i>popSize</i>	<i>maxgen</i>	<i>pcross</i>	<i>pmutation</i>	<i>rezultat</i>
1	100	150	0.65	0.5	937
2	100	500	0.65	0.5	937
3	60	500	0.65	0.5	985
4	150	500	0.65	0.5	950
5	150	200	0.4	0.3	985
6	150	500	0.4	0.3	937

Iz tablice 5.5 može se primjetiti da su najbolje odabrani parametri u 1.redu tablice. U 2. redu tablice uočava se da je rješenje problema isto i za 500 generacija (nakon 150 generacija rezultat se ustaljuje). Smanjenje populacije (3. red tablice), kao i njeno povećanje (4.red tablice) uzrokuje lošiji rezultat. Povećanjem populacije moguće je dobiti dobar rezultat (6. red tablice) uz veći broj generacija i druge parametre križanja i mutacije. Primjećuje se da povećanje populacije iziskuje smanjenje parametra mutacije i križanja.

5.3.2.3. MT20 problem

Primjer problema 20 izradaka po 1 komad - 5 strojeva prikazan je u tablici 5.6.

Tab. 5.6 *MT20 problem*

Vrijeme izvršavanja operacije						Redoslijed strojeva				
dio	Operacije					Operacije				
	1	2	3	4	5	1	2	3	4	5
j_1	29	9	49	62	44	1	2	3	4	5
j_2	43	75	69	46	72	1	2	4	3	5
j_3	91	39	90	12	45	2	1	3	5	4
j_4	81	71	9	85	22	2	1	5	3	4
j_5	14	22	26	21	72	3	2	1	4	5
j_6	84	52	48	47	6	3	2	5	1	4
j_7	46	61	32	32	30	2	1	3	4	5
j_8	31	46	32	19	36	3	2	1	4	5
j_9	76	76	85	40	26	1	4	3	2	5
j_{10}	85	61	64	47	90	2	3	1	4	5
j_{11}	78	36	11	56	21	2	4	1	5	3
j_{12}	90	11	28	46	30	3	1	2	4	5
j_{13}	85	74	10	89	33	1	3	2	4	5
j_{14}	95	99	52	98	43	3	1	2	4	5
j_{15}	6	61	69	49	53	1	2	5	3	4
j_{16}	2	95	72	65	25	2	1	4	5	3
j_{17}	37	13	21	89	55	1	3	2	4	5
j_{18}	86	74	88	48	79	1	2	5	3	4
j_{19}	69	51	11	89	74	2	3	1	4	5
j_{20}	13	7	76	52	45	1	2	3	4	5

Rezultati dobiveni primjenom genetskih algoritama uz 100 gena po kromosomu, 200 kromosoma u populaciji ($popSize = 200$), jednu populaciju, 2000 generacija ($maxgen = 1000$), vjerojatnost križanja $p_{cross} = 0.85$, te vjerojatnost mutacije $p_{mutation} = 0.5$, prikazani su na slici 5.19.

Maksimalno vrijeme = 1174

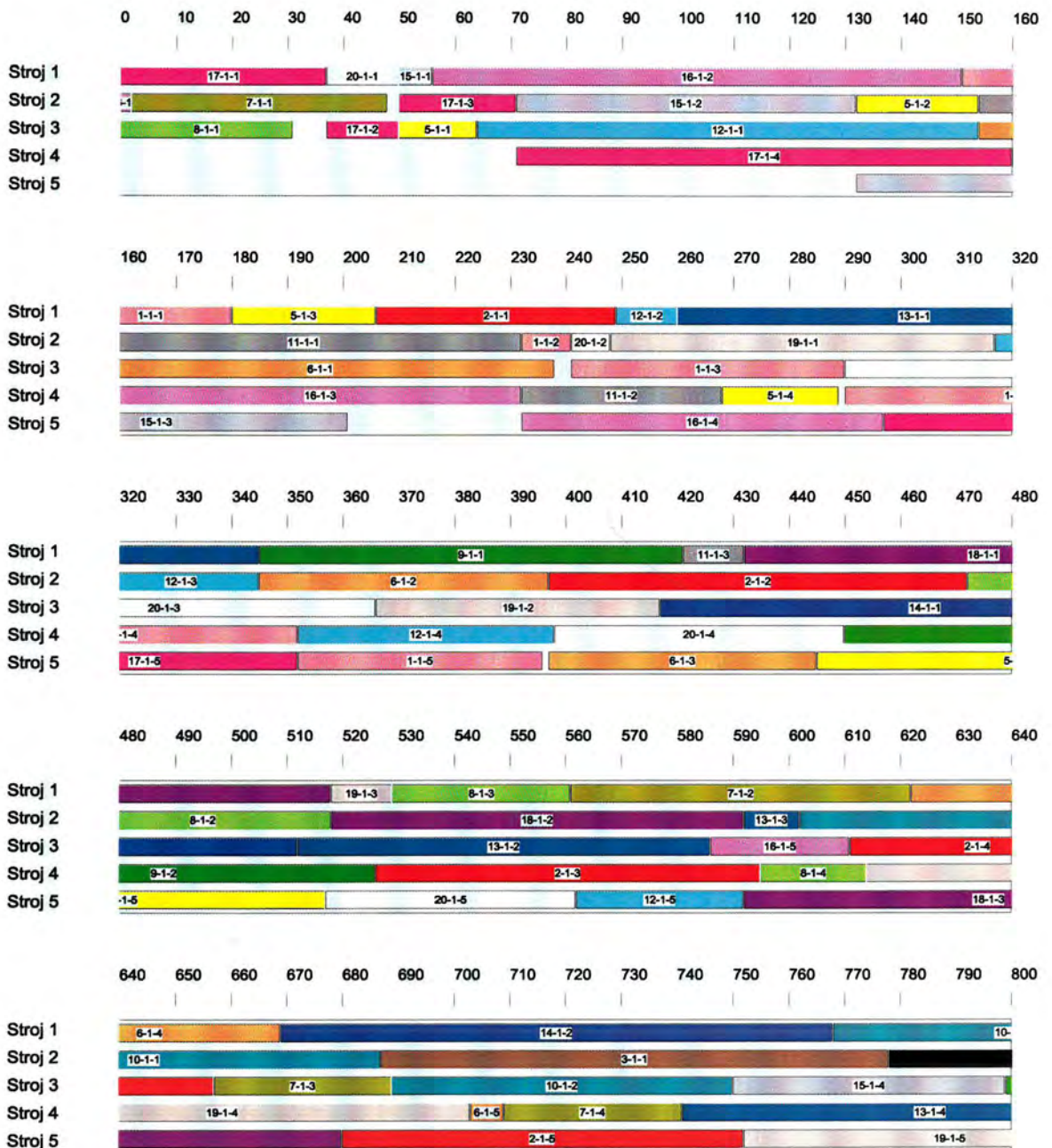
Broj generacija = 1000

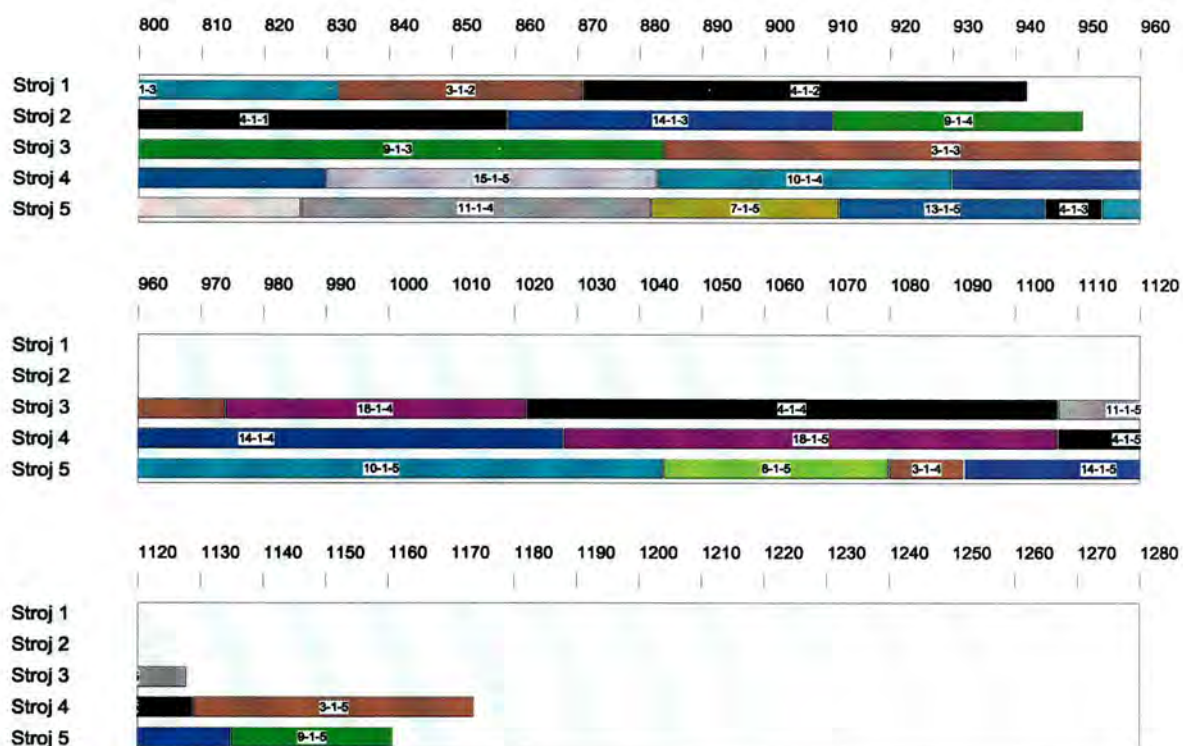
Veličina populacije = 200

a)



b)





c)

SI.5.19 Prikaz rezultata problema MT20 programom GJOB

- rezultat
- legenda
- gantogram strojeva

U tablici 5.7 prikazani su rezultati problema MT20 za različito postavljene parametre genetskog algoritma.

Tab. 5.7 Parametri genetskog algoritma za MT20 problem

	<i>popSize</i>	<i>maxgen</i>	<i>pcross</i>	<i>pmutation</i>	<i>rezultat</i>
1	200	1000	0.85	0.5	1174
2	200	2000	0.85	0.5	1174
3	150	1000	0.85	0.5	1247
4	300	1000	0.85	0.5	1194
5	300	1000	0.85	0.05	1214
6	300	1000	0.5	0.3	1279

Na temelju tablice 5.7 može se primjetiti da su najbolje odabrani parametri u 1.redu tablice. U 2. redu tablice uočava se da je rješenje problema isto i za 2000 generacija (nakon 1000 generacija rezultat se ustaljuje). Smanjenje populacije (3. red tablice), kao i njeno povećanje (4.red tablice) uzrokuje lošiji rezultat.

Usporede li se najbolji rezultati za problem MT06, MT10 i MT20, vidljivo je da složeniji problemi zahtijevaju veće populacije i veći broj generacija, a ovisno o povećanju populacije smanjuje se vjerojatnost križanja i mutacije.

5.4. USPOREDBA REZULTATA I POTVRDA KODA RAČUNALNOG PROGRAMA

Problemi MT06, MT10 i MT20 tri su referentna problema koje većina istraživača koriste za testiranje performansi svojih postupaka i algoritama, te ih se u istu svrhu koristi i u ovom radu.

U tablici 5.8 dana je usporedba najboljih rezultata tih problema dobivenih metodom premještanja uskog grla²⁴ [47,49](kolona SB), simuliranog približavanja²⁵ [45] (kolona SA), tabu tehnikom pretraživanja²⁶ [46](kolona TS) s rezultatima dobivenih programom GJOB.

Tab. 5.8 Usporedba rezultata programa GJOB s rezultatima nekih heurističkih metoda

Problem	Broj dijelova <i>n</i>	Broj strojeva <i>m</i>	Optimum	SA najbolji	TS najbolji	SB najbolji	GJOB
MT06	6	6	55	55	55	55	55
MT10	10	10	930	930	935	930	937
MT20	20	5	1165	1165	1165	1178	1174

Na slici 5.20 dana je usporedba rezultata za problem MT10 raznih heurističkih metoda dobivenih programskim paketom *LEKIN – Flexible Job Shop Scheduling System* razvijenim od Michael L. Pinedo-a i Andrew Feldman-a na Leonard N. Stern, School of Business, New York University, dok je na slici 5.21 prikazan rezultat problema MT10 gantogramom strojeva dobiven istim programom metodom simuliranog približavanja.

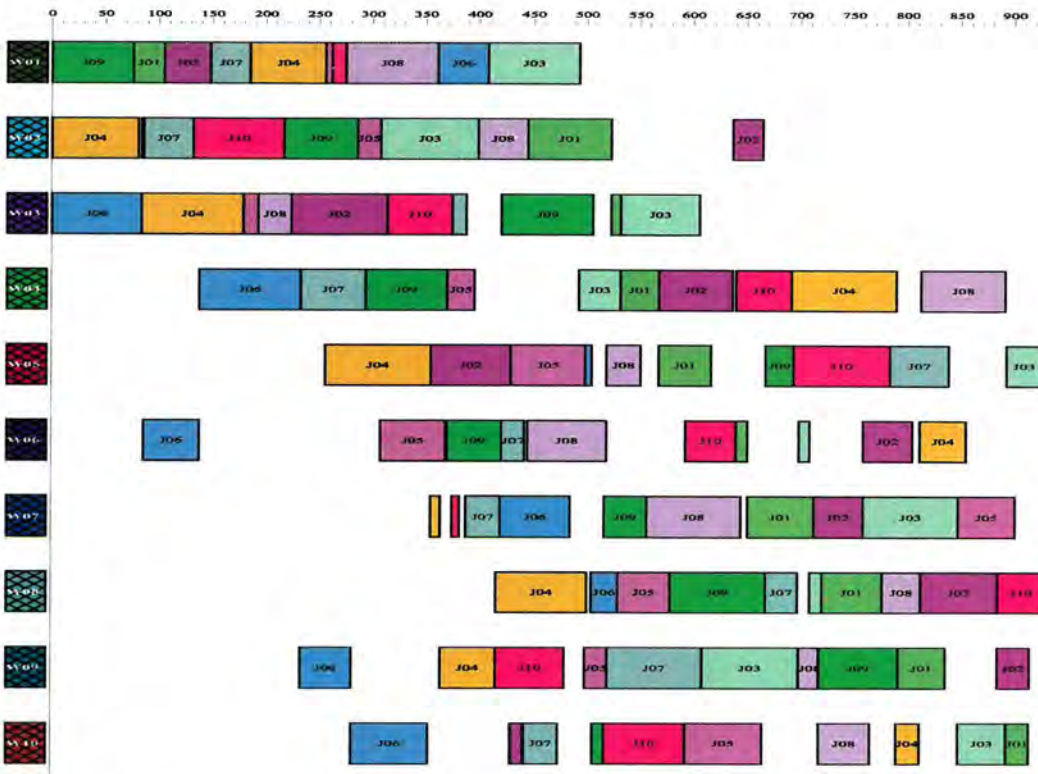
Schedule	Time	C_{max}
EDD	1	1246
LPT	1	1168
MS	1	1168
WSPT	1	1356
Simulated Annealing	64	930
Shifting Bottleneck	15	1094
Tabu Search	61	938

Sl. 5.20 Usporedba rezultata za MT10 problem raznih heurističkih metoda dobivenih programskim paketom *LEKIN*

²⁴ *Shifting Bottleneck*

²⁵ *Simulated Annealing*

²⁶ *Tabu Search*



Sl. 5.21 Gantogram strojeva za problem MT10 dobiven programom LEKIN metodom simuliranog približavanja

Heurističke metode daju određena rješenja, ali ostavljaju mnogo prostora za daljnja istraživanja jer su neke prikladne samo za male obime problema, neke su spore i neprikladne za operacijsko upravljanje, neke daju nedovoljnu razinu kvalitete rješenja.

U tablici 5.9 prikazana je usporedba rezultata nekih genetskih algoritama i GJOB-a za tri referentna problema.

Tab. 5.9 Rezultati genetskih algoritama za tri referentna problema

Autori	6 x 6	10 x 10	20 x 5
Nakano i Yamada (1991)	55	965	1215
Yamada i Nakano (1992)	55	930	1184
Paredis (1992)	-	1006	-
Gen i drugi (1994)	55	962	1175
Fang i drugi (1993)	-	949	1189
Dorndorf1 i Pesch (1995)	55	960	1249
Dorndorf2 i Pesch (1995)	55	938	1178
Croce i drugi (1995)	55	946	1178
Cheng i drugi (1995)	55	948	1196
GJOB	55	937	1174

Problem 6x6 (MT06) je relativno jednostavan i svi su genetski algoritmi postigli optimalno rješenje. Za problem 10x10 (MT10) postignuto rješenje je među najboljim, dok je za problem 20x5 (MT20) postignuto najbolje rješenje. Međutim, teško je procijeniti koji je pristup superioran nad ostalima u ovoj tablici. Svi su testirani na istim primjerima, ali ne i u istim eksperimentalnim uvjetima. Isti eksperimentalni uvjeti znače da je:

- isti ukupan broj ispitanih kromosoma
- isti ukupan broj slučajnih prolaza
- ista sredstva proračuna

Performanse genetskih algoritama za iste eksperimentalne uvjete mogu se procijeniti sa slijedećih aspekata:

- najbolje rješenje
- devijacija optimalnog rješenja od najboljeg rješenja
- frekvencija postizanja najboljeg rješenja
- distribucija rješenja i statistička analiza
- potrebna memorija i računsko vrijeme

Prednosti ili nedostaci za svaki od genetskih algoritama mogli bi se utvrditi tek nakon usporedne studije pod standardnim eksperimentalnim uvjetima koristeći ove referentne probleme i druge probleme većeg opsega.

U tablici 5.10 prikazani su parametri genetskog algoritma postavljenim za dobivanje rezultata MT10 problema kod raznih autora.

Tab. 5.10 Parametri genetskog algoritma raznih autora za problem MT10

Autori	<i>popSize</i>	<i>maxgen</i>	<i>pcross</i>	<i>pmutation</i>
Nakano i Yamada (1991)	1000	150	?	?
Yamada i Nakano (1992)	2000	100	?	?
Gen i drugi (1994)	60	5000	0.4	0.3
Dorndorf1 i Pesch (1995)	200	?	0.65	0.001
Dorndorf2 i Pesch (1995)	40	?	0.65	?
Croce i drugi (1995)	300	2971	1	0.03
Cheng i drugi (1995)	40	2000	0.4	0.4
GJOB	100	150	0.65	0.5

Iz tablice 5.10 se može vidjeti da postoje različite izvedbe za postavljanje parametara genetskih algoritama.

Na temelju analize problema MT06, MT10 i MT20 može se vidjeti da se genetskim algoritmom implementiranim u programu GJOB postižu odlični rezultati, a u usporedbi s heurističkim metodama zahtijevaju znatno manje vrijeme računanja, a veoma su prikladni za operacijska upravljanja i velike obime problema.

6. PRAKTIČNI PRIKAZ METODE NA KONKRETNOM ZADATKU

6.1. PRIMJER IZBORA PLANA PROCESA

Primjena metode genetskog algoritma na izbor plana procesa prezentirana je na konkretnom zadatku. Pri tom je korišten isti radni zadatak kao [64,65] radi kasnije usporedbe rezultata. Za radni nalog, koji sadrži četiri vrste dijelova ($N = 1,2,3,4$) u zadanim količinama ($q = 50,20,80,60$), razvijeni su alternativni planovi procesa definirani skupovima parametara:

dijelova 1	Vrsta	$p_{11} = \{(d_{01}=14, t_{111}=30), (d_{23}=8, t_{112}=15), (d_{45}=4, t_{113}=35), (d_{67}=8, t_{114}=30), (d_{89}=2)\}$
		$p_{12} = \{(d_{03}=10, t_{122}=35), (d_{47}=13, t_{124}=28), (d_{81}=6, t_{121}=45), (d_{29}=5)\}$
		$p_{13} = \{(d_{03}=10, t_{132}=35), (d_{45}=4, t_{133}=20), (d_{67}=8, t_{134}=60), (d_{89}=2)\}$
dijelova 2	Vrsta	$p_{21} = \{(d_{05}=14, t_{213}=10), (d_{67}=8, t_{214}=40), (d_{83}=15, t_{212}=25), (d_{41}=10, t_{211}=15), (d_{29}=5)\}$
		$p_{22} = \{(d_{01}=2, t_{221}=22), (d_{27}=8, t_{224}=30), (d_{83}=15, t_{222}=32), (d_{49}=14)\}$
		$p_{23} = \{(d_{05}=14, t_{233}=25), (d_{61}=15, t_{231}=18), (d_{27}=8, t_{234}=30), (d_{89}=2)\}$
		$p_{24} = \{(d_{01}=2, t_{241}=42), (d_{27}=8, t_{244}=45), (d_{89}=2)\}$
dijelova 3	Vrsta	$p_{31} = \{(d_{05}=14, t_{313}=20), (d_{63}=6, t_{312}=30), (d_{41}=10, t_{311}=40), (d_{27}=8, t_{314}=15), (d_{89}=2)\}$
		$p_{32} = \{(d_{05}=14, t_{323}=58), (d_{61}=15, t_{321}=52), (d_{29}=5)\}$
dijelova 4	Vrsta	$p_{41} = \{(d_{07}=5, t_{414}=18), (d_{81}=6, t_{411}=12), (d_{23}=8, t_{412}=25), (d_{49}=14)\}$
		$p_{42} = \{(d_{05}=14, t_{423}=20), (d_{61}=15, t_{421}=42), (d_{29}=5)\}$
		$p_{43} = \{(d_{05}=14, t_{433}=10), (d_{63}=6, t_{432}=18), (d_{41}=10, t_{431}=15), (d_{27}=8, t_{434}=12), (d_{89}=2)\}$

Ovi procesi razvijeni su za korištenje na fleksibilnom proizvodnom sustavu s četiri različite radne stanice i automatiziranim transportnim sustavom, prikazanim na slici 3.2.

Potrebno je pronaći onu kombinaciju planova procesa koja će osigurati najmanje vremensko opterećenje proizvodnog sustava, tj zadatak je naći minimum funkcije cilja, predstavljene izrazom 3.12, uz dana ograničenja, izrazi 3.13 i 3.14. Da se dobije mjera prilagođenosti u genetskom algoritmu treba primijeniti izraz 4.2.

Optimizacijsku se zadaću rješava pomoću *GALOPPS* (the Genetic Algorithm Optimized for Portability and Parallelism System) koda kojeg su razvili prof. E.D. Goodman i *GARAGE* (Genetic Algorithms Research and Applications Group), Michigan State University [110].

Genetski je algoritam primijenjen uz 4 gena po kromosomu, 20 kromosoma u populaciji ($popSize = 20$), 20 generacija ($maxgen = 20$), stohastički odabir s zamjenom, linearno skaliranje, križanje s jednom točkom reza ($pcross = 0.5$), jednogensku mutaciju ($pmutation = 0.3$) i jednu populaciju.

Dobiveno je optimalno rješenje koje se sastoji od skupa planova procesa $\{p_{13}, p_{24}, p_{32}, p_{41}\}$ s vrijednosti funkcije cilja $\varphi = 27246$.

Kod identificiranog najboljeg rješenja postignute su veličine $\Delta m = 1097$ i $\Delta h = 470$. One ukazuju na značajnu neuravnoteženost opterećenja proizvodnog sustava, kao posljedicu eksploatacijskih značajki PS-a, strukture radnog naloga i razvijenih planova procesa. Usko grlo sustava je automatski transport, čije opterećenje nadmašuje

usko grlo među radnim stanicama. Ubrza li se transport za napr. 7,5%, transport više nije usko grlo sustava, a rješenje $\{p_{13}, p_{24}, p_{32}, p_{41}\}$ i dalje ostaje najpovoljnije s vrijednošću funkcije cilja $\varphi = 26393$, istim opterećenjem strojeva, nepromijenjenim Δm i $\Delta h = 0$

6.1.1. Usporedba s rezultatima tabu tehnike

Prema [64,65] rješavanje istog zadatka uz primjenu metode tabu tehnike pretraživanja i metode uparene izmjene, provedeno je u 12 koraka. Optimalno rješenje $u = \{2310\}$ s vrijednošću funkcije vrednovanja $f(u_c) = 27246$ pronađeno je već u trećem koraku, a čini ga skup planova procesa $\{p_{13}, p_{24}, p_{32}, p_{41}\}$.

Procedura rješavanja prikazana je u tablici 6.1 prikazom samo prva 3 koraka.

Tab. 6.1 Prikaz procedure rješavanja na primjeru

Br. kor.	u_c	$C(u_c)$	$f(u_c)$	$u \in \sigma(u_c)$	$c(u)$	$f(u)$	$T-C(u_c)-(a_L)$
1	{0000}	0	29213	{0001} {0002} {0010} {0100} {0200} {0300} {1000} {2000}	1 2 3 6 12 18 24 48	30473 30113 28840 28860 28687 27954 30497 29623	0- (-)
2	{0300}	18	27954	{0301} {0302} {0310} {0200} {0100} {0000} {1300} {2300}	19 20 21 12 6 0 42 66	30333 28853 28660 28687 28860 29213 29237 28193	18- (29213) 0- (-)
3	{2300}	66	28193	{2301} {2302} {2310} {2200} {2100} {2000} {1300} {0300}	67 68 69 60 54 48 42 18	28720 28913 27246 28987 29270 29623 29237 27954	66- (27954) 18- (29213) 0- (-)

Za prikazani primjer s manjim brojem varijabli primjenom algoritma temeljenog na tabu tehnici pretraživanja kao i primjenom genetskog algoritma pronađeno je optimalno rješenje već u ranoj fazi provedbe postupka. Realno je za očekivati, da će kompleksniji radni nalozi zahtijevati provedbu postupka temeljenog na tabu tehnici s većim brojem koraka i odgovarajuće veći utrošak vremena. Za slučaj radnih naloga s većim brojem dijelova uputnije je korištenje genetskog algoritma zbog visoke razine pouzdanosti, efikasnosti numeričkog proračuna i brzine u iznalaženju optimalnih rješenja.

6.2. PRIMJER TERMINIRANJA DIJELOVA NA FPS-U

Za izabranu kombinaciju planova procesa $\{p_{13}, p_{24}, p_{32}, p_{41}\}$ potrebno je optimizirati redoslijed ulaska pojedinih dijelova u proces da bi ciklus izrade bio što kraći uz što jednakomjernije opterećenje strojeva.

Optimizacija redoslijednog ulaska i izvođenja operacija pojedinih dijelova u procesu provedeno je uz primjenu dijelom različitih korištenja i variranja provedbe procesa sa stajališta različitih karakteristika transportnog sustava u okviru FPS-a. U tu svrhu problem je riješen na 4 međusobno različita načina uz postizanje i različitih krajnjih rezultata. Te različite načine rješavanja karakteriziraju:

- Slučaj 1.** zahtjev za što kraći ciklus izrade uz pretpostavku da transport izratka ovisi jedino o spremnosti stroja za prihvrat izratka (spremnost transportnog sustava zanemarena)
- Slučaj 2.** zahtjev za što kraći ciklus izrade i što kraćim vremenom čekanja stroja na izradu uz pretpostavku da transport izratka ovisi jedino o spremnosti stroja za prihvrat izratka (spremnost transportnog sustava zanemarena)
- Slučaj 3.** zahtjev za što kraći ciklus izrade uz pretpostavku da transport izratka ovisi o spremnosti stroja za prihvrat izratka i o spremnosti transportnog sustava
- Slučaj 4.** zahtjev za što kraći ciklus izrade uz pretpostavku da transport izratka ovisi jedino o spremnosti transportnog sustava

Rezultati za pojedini slučaj dobiveni su tako da su isprobane različite vrijednosti parametara genetskog algoritma. Tako stečeno iskustvo pokazalo je superiornost nekih, te su prikazani samo najbolji rezultati.

6.2.1. Slučaj 1.

Na slici 6.1 prikazan je rezultat optimizacije prema kriteriju za što kraći ciklus izrade na FPS-u s četiri različita stroja uz temeljnu pretpostavku da transport izratka nakon izvršenja operacije ovisi jedino o spremnosti stroja za prihvrat palete izratka na stroj na kojem će biti izvedena naredna operacija, tj. ne ovisi o spremnosti transportera.


Genetski je algoritam primijenjen uz 530 gena po kromosomu, 250 kromosoma u populaciji ($popSize=250$), 500 generacija ($maxgen=500$), stohastički odabir s zamjenom, linearno skaliranje, križanje zamjenom parcijalnih redoslijeda ($pcross=0.65$), te mutaciju baziranu na traženju susjedstva ($pmutation=0.03$).

Maksimalno vrijeme = 5723

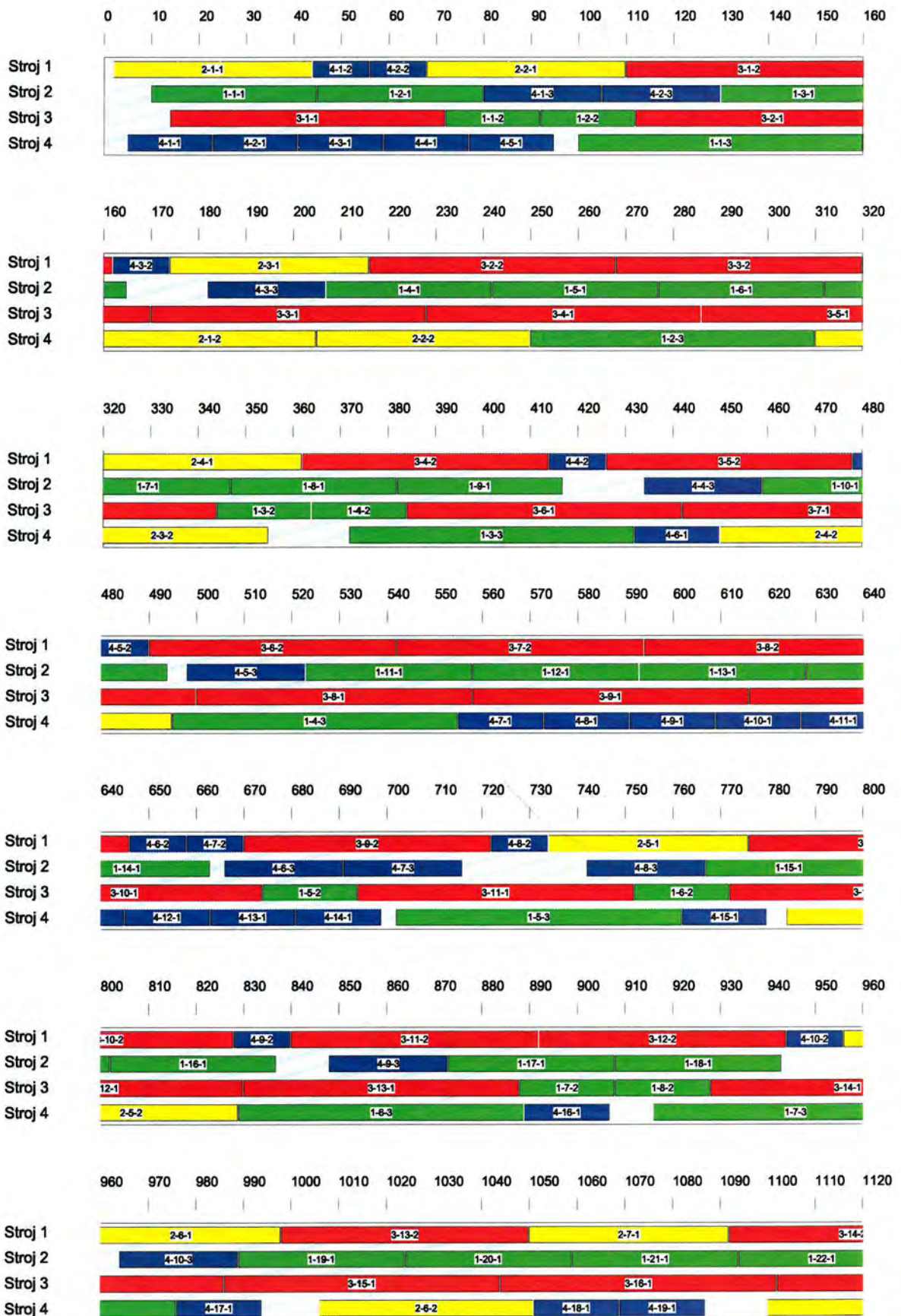
Broj generacija = 500

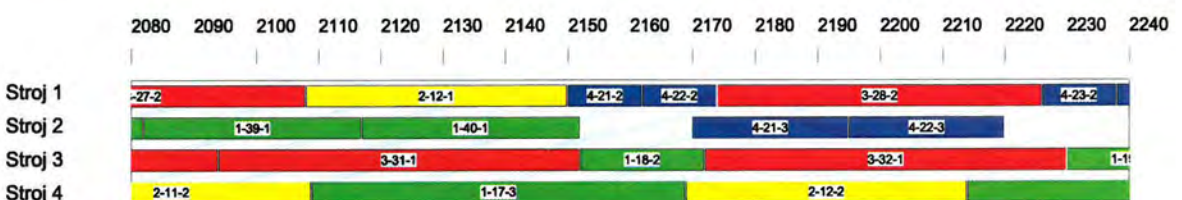
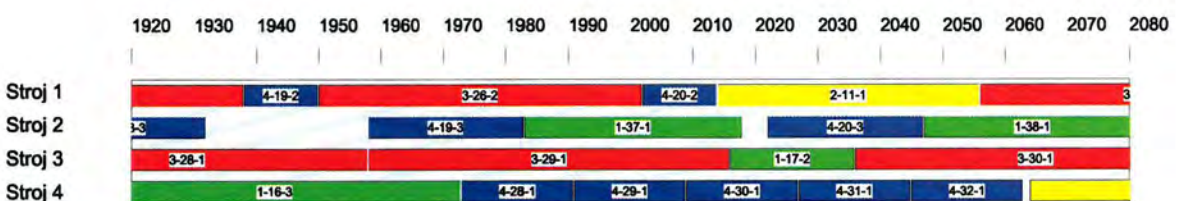
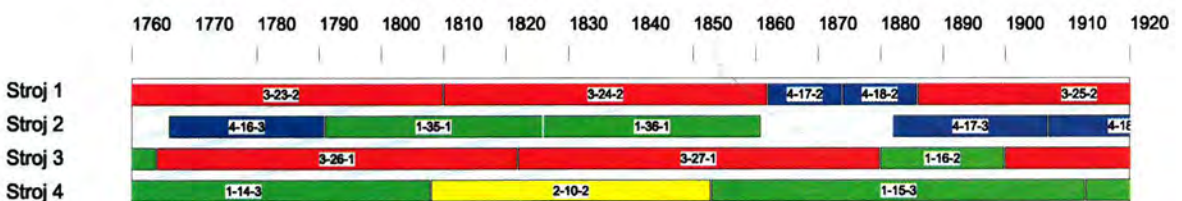
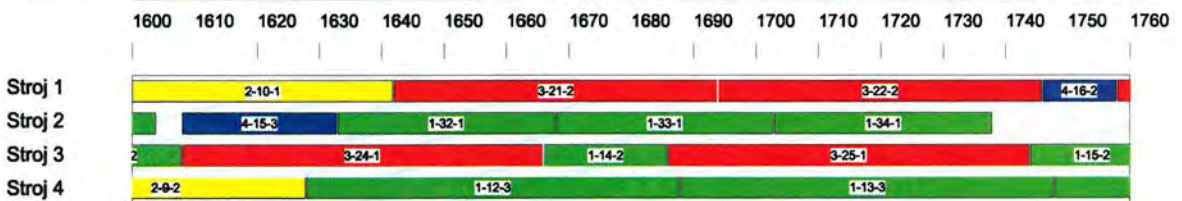
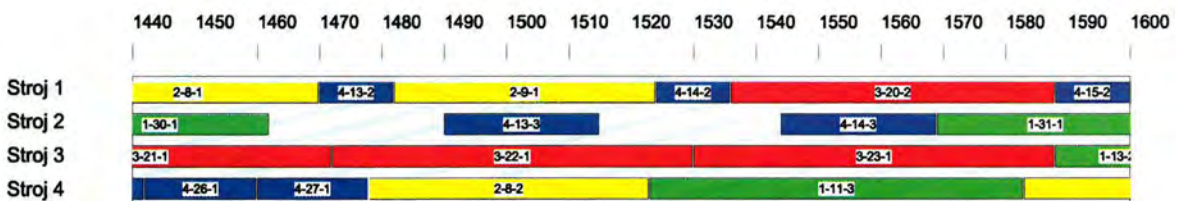
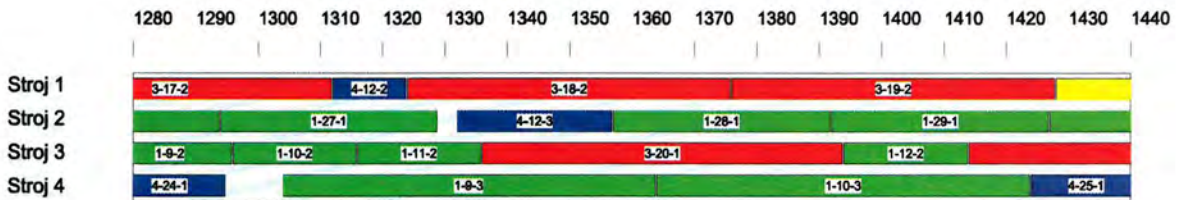
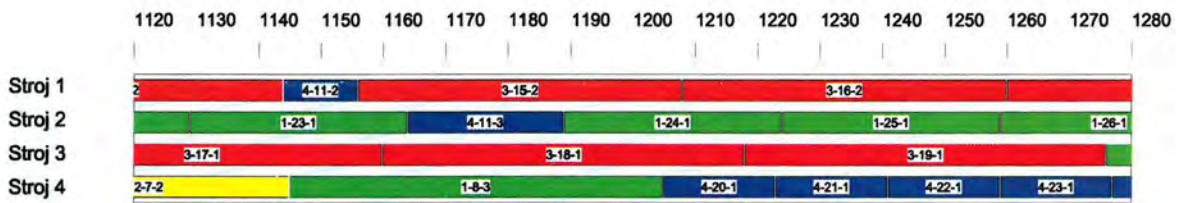
Veličina populacije = 250

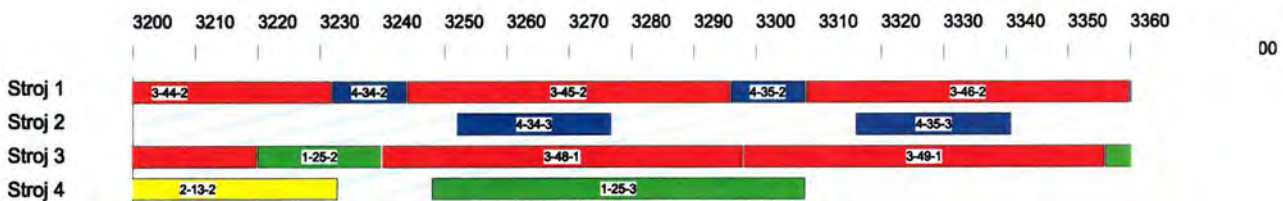
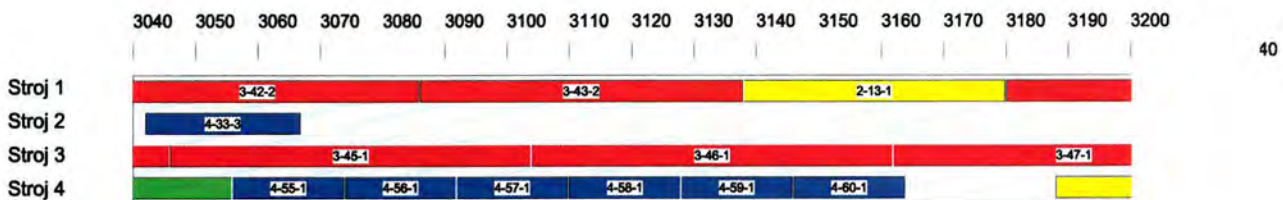
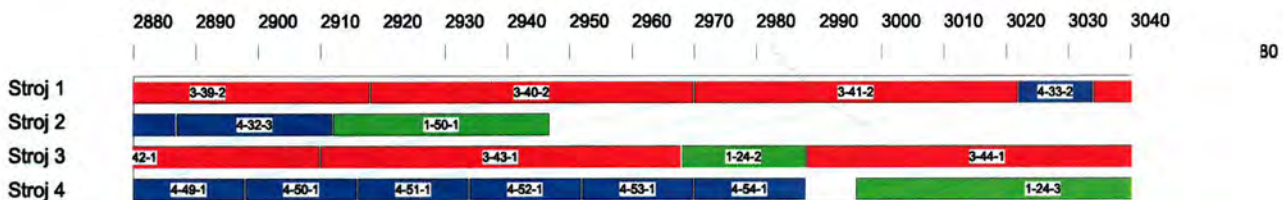
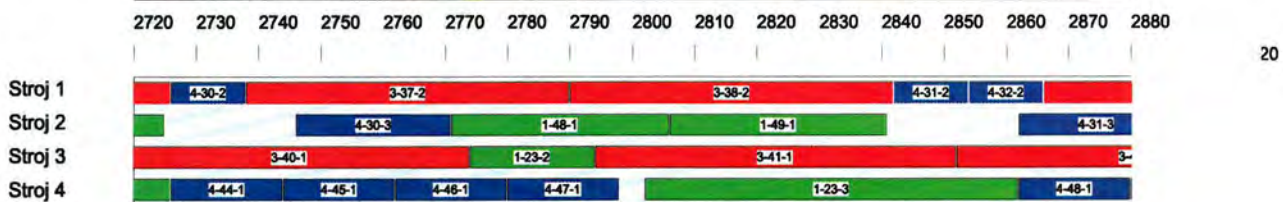
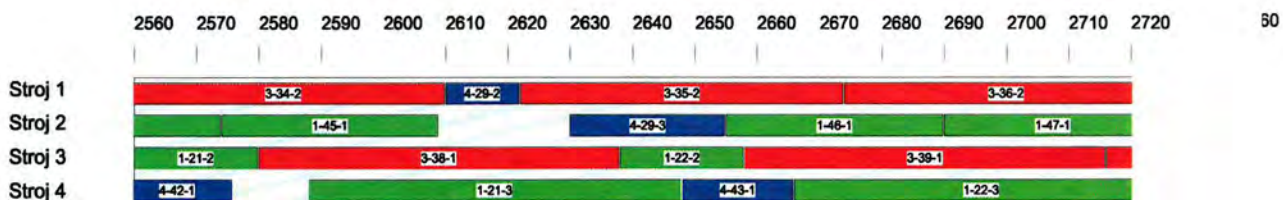
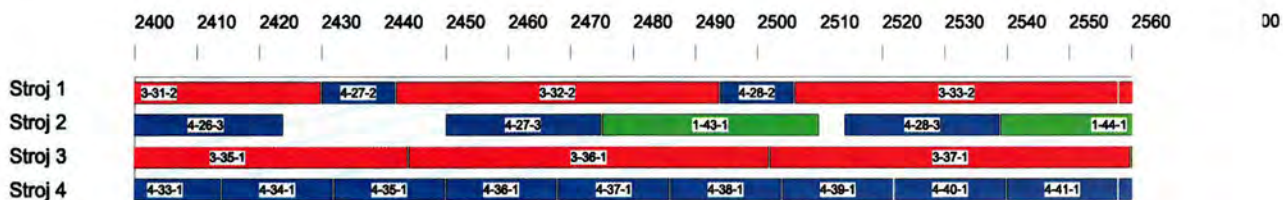
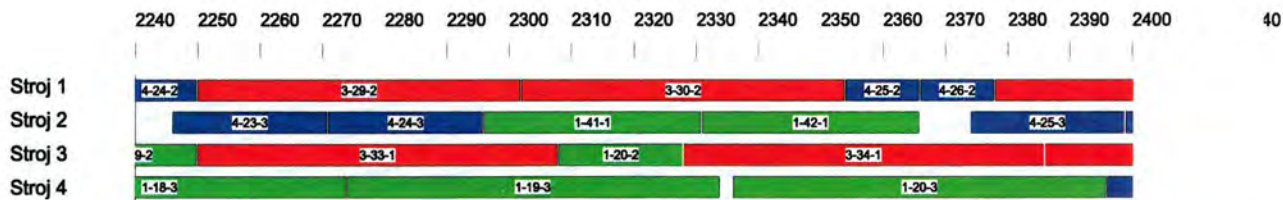
a)

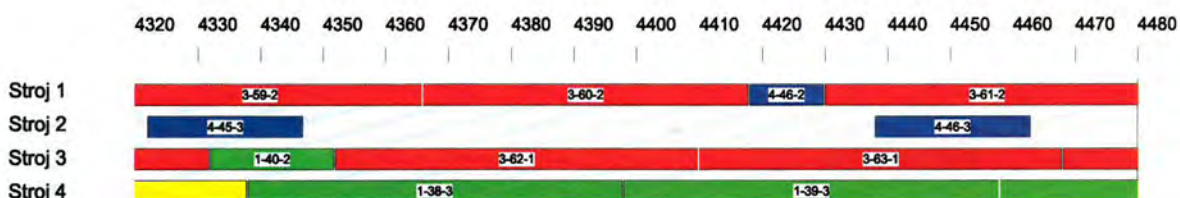
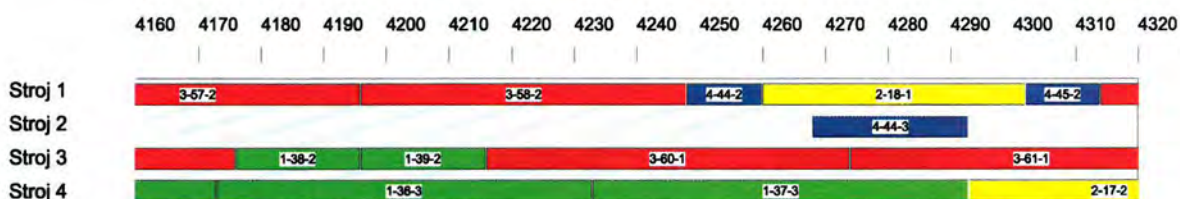
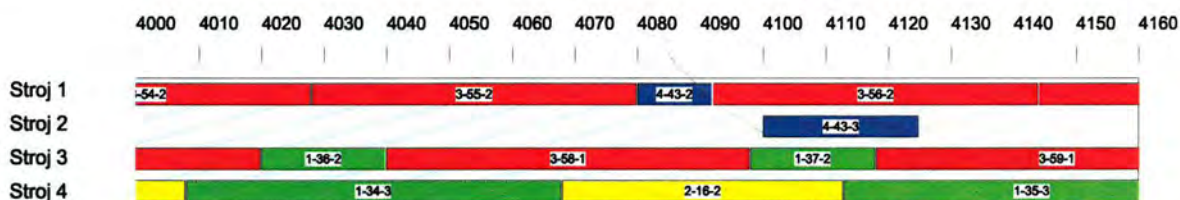
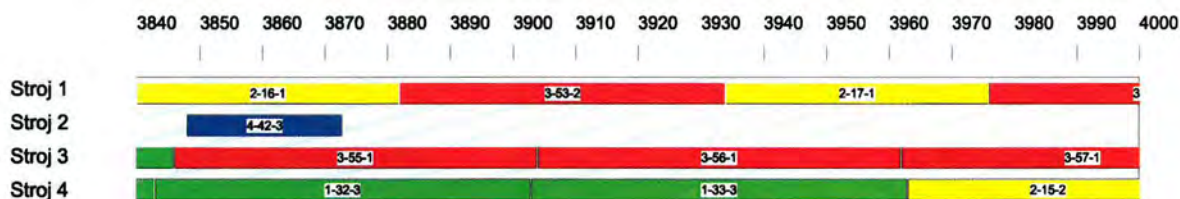
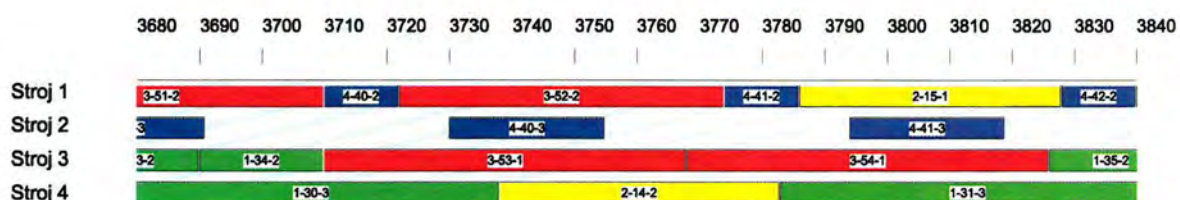
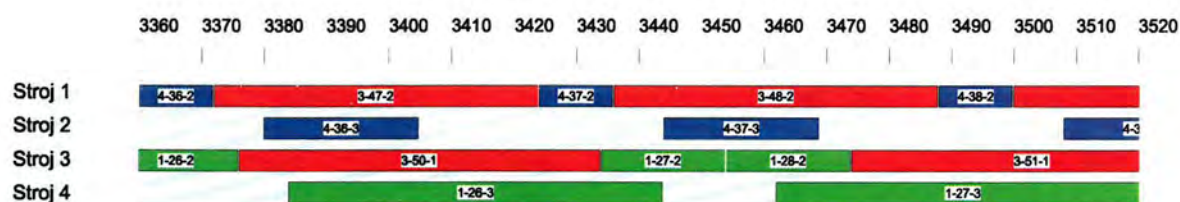
	Vrsta dijelova 1 (50)
	Vrsta dijelova 2 (20)
	Vrsta dijelova 3 (80)
	Vrsta dijelova 4 (60)

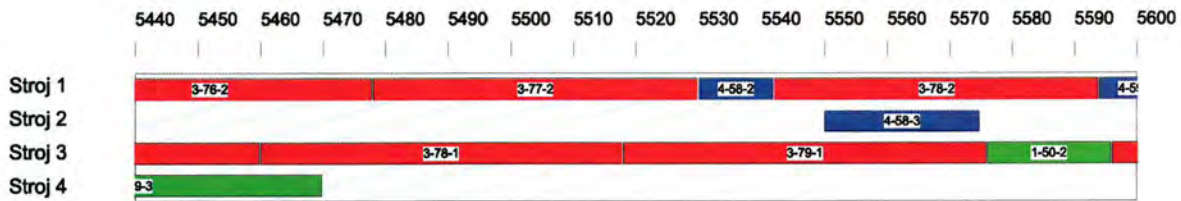
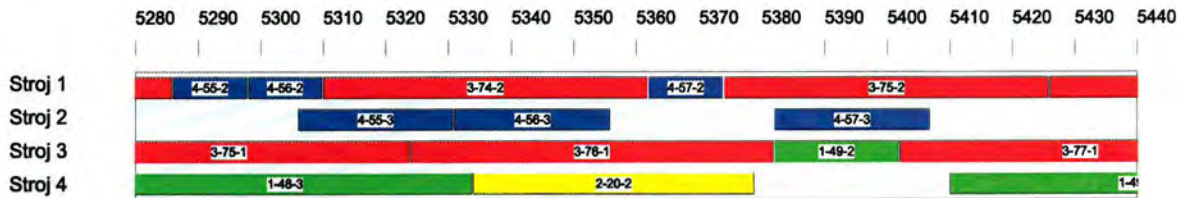
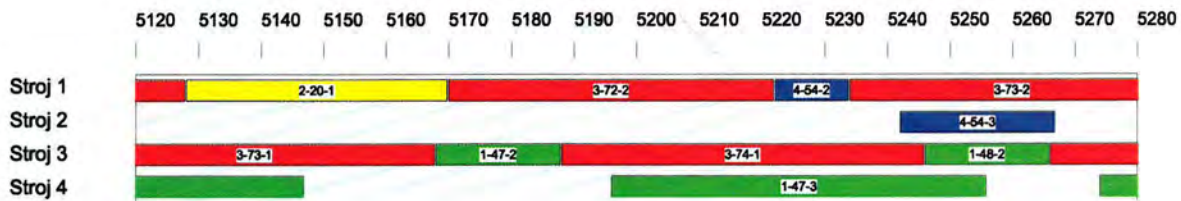
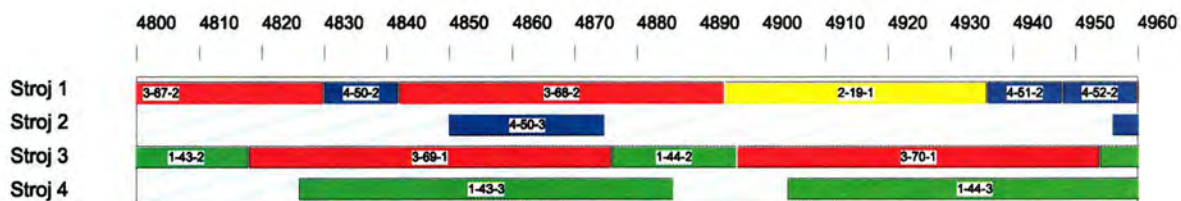
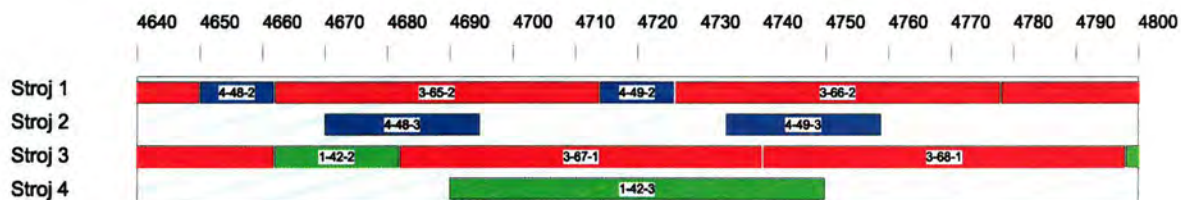
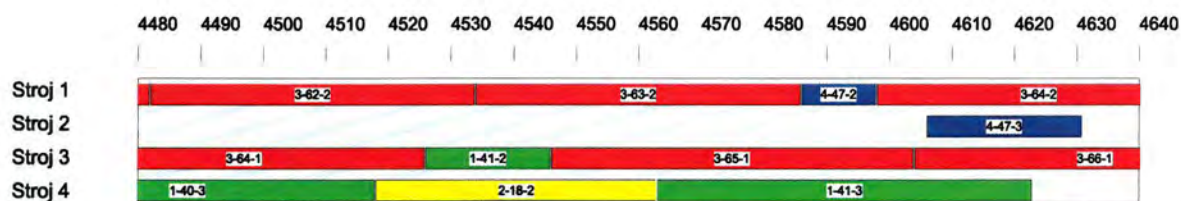
b)

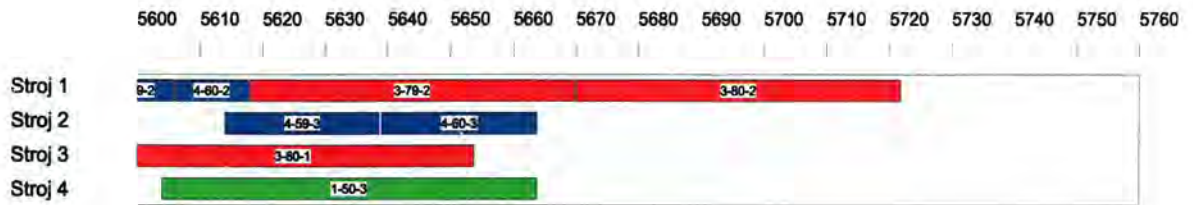












c)

Sl. 6.1 Prikaz rezultata zadatka za kriterij minimalnog ciklusa izrade – slučaj 1 programom GJOB

- rezultat
- legenda
- gantogram strojeva

Može se primijetiti da je stroj 2 dosta dugo u upotrebi (do vremena 5664 kad izradak 4 napušta proces), a njegovo iskorištenje je vrlo malo. Međutim, ovdje je samo korišten kriterij za što kraćim ciklusom izrade. Za slučaj 2 na FPS-u istih karakteristika kao u prethodnom slučaju, upotrijebit će se kombinacija kriterija za minimalnim ciklusom izrade s kriterijem za minimalnim vremenom čekanja stroja na izradu.

6.2.2. Slučaj 2.

Na slici 6.2 prikazan je rezultat optimizacije prema kriteriju za što kraći ciklus izrade i prema kriteriju za što kraćim vremenom čekanja stroja na izradu na FPS-u istih karakteristika kao u slučaju 1.

Uvođenjem dodatnog kriterija za što kraćim vremenom čekanja stroja na izradu, slučaj 2 postaje složeniji od slučaja 1, te korištenje istih parametara genetskog algoritma ne daje najbolji rezultat. Isprobavanjem različitih vrijednosti parametara genetskog algoritma pokazalo se dovoljnim povećati broj generacija.

Primijenjen je genetski algoritam uz 530 gena po kromosomu, 250 kromosoma u populaciji ($popSize = 250$), 1000 generacija ($maxgen = 1000$), stohastički odabir s zamjenom, linearno skaliranje, križanje zamjenom parcijalnih redoslijeda ($pcross = 0.8$), te mutaciju baziranu na traženju susjedstva ($pmutation = 0.03$).

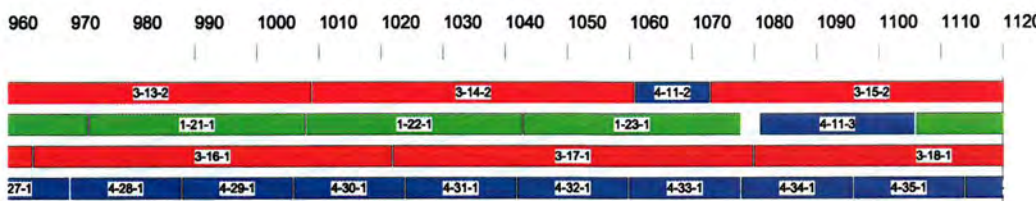
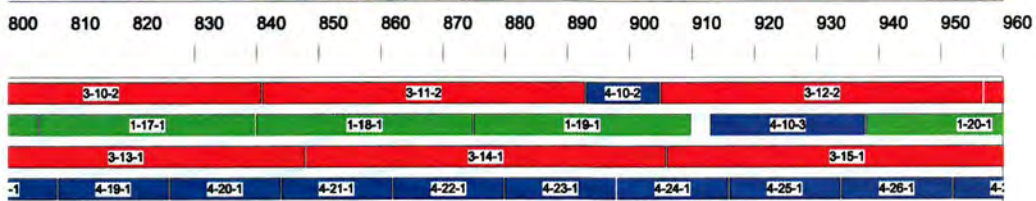
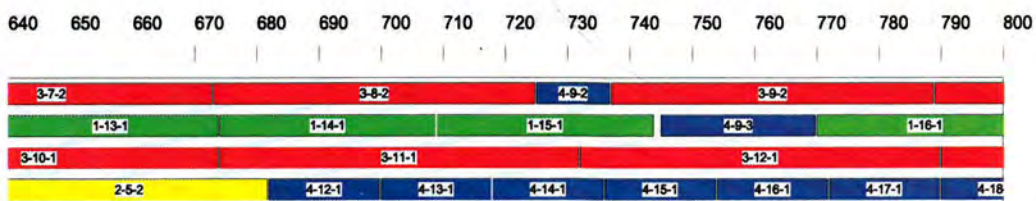
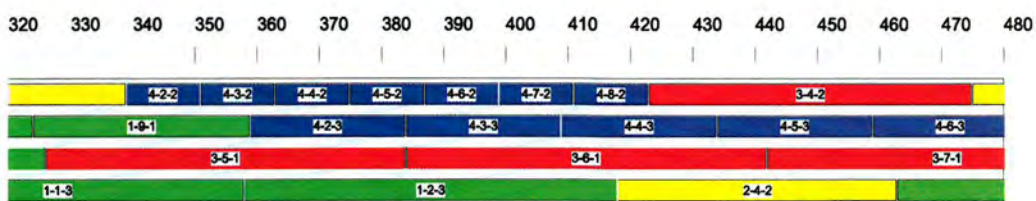
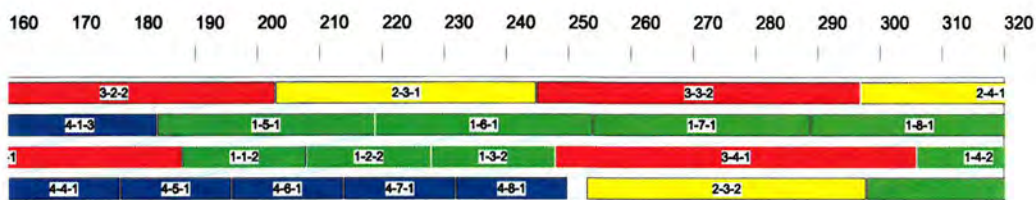
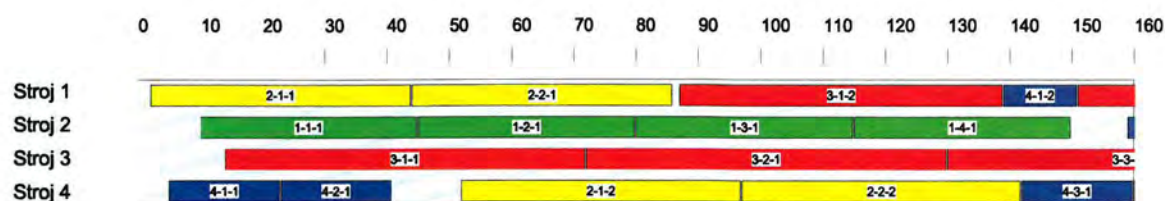
Maksimalno vrijeme = 5723

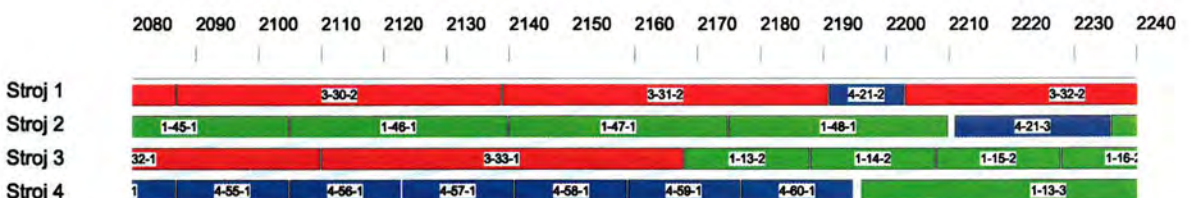
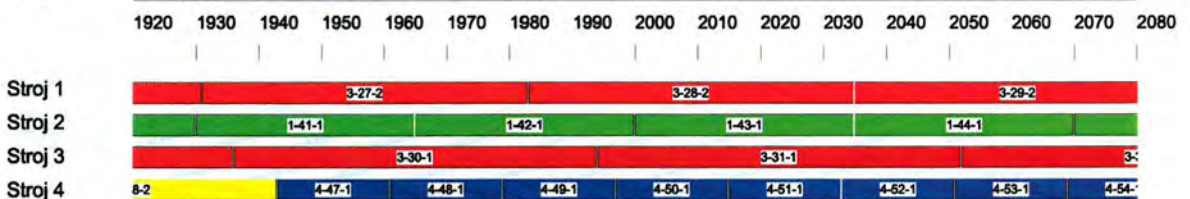
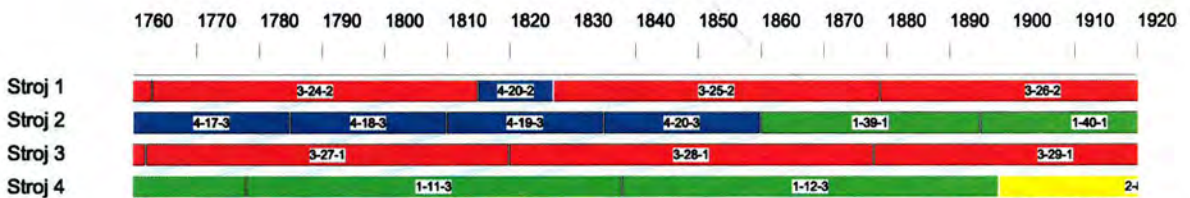
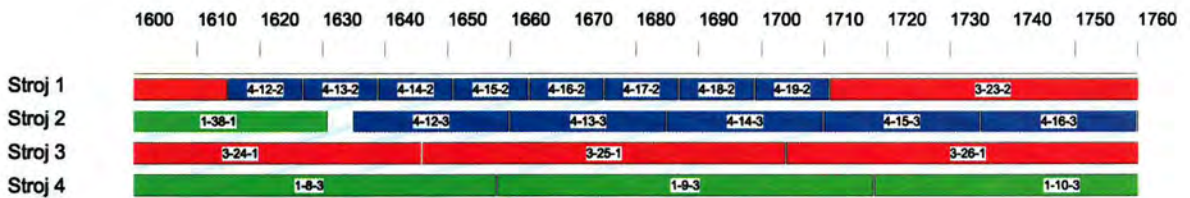
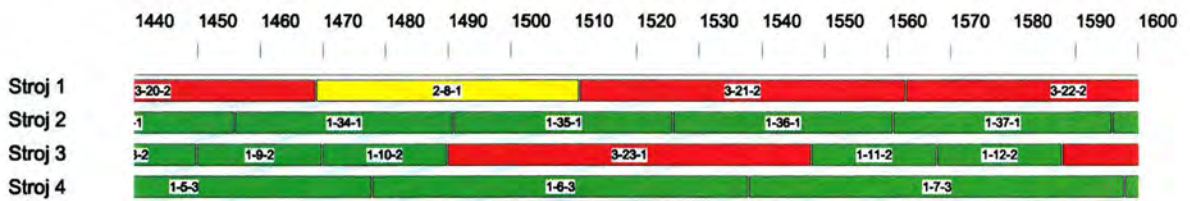
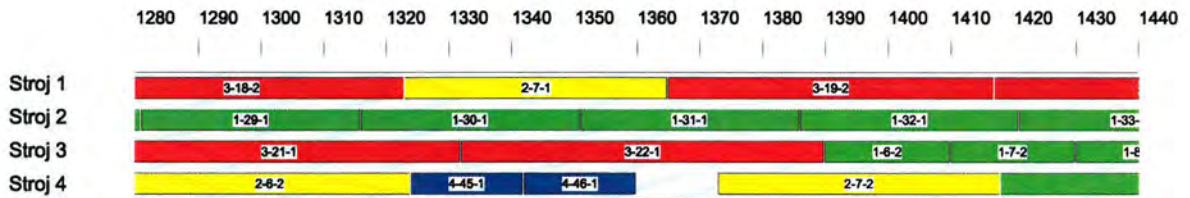
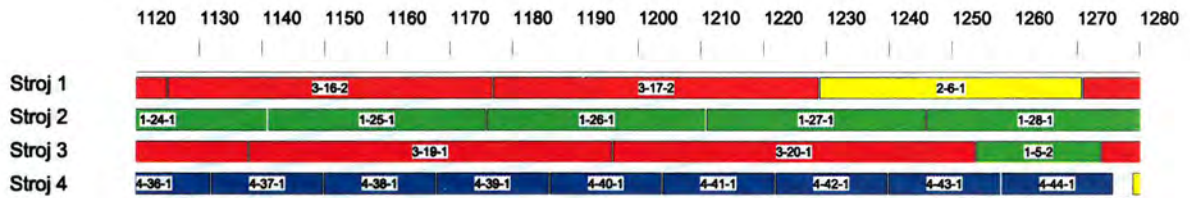
Suma vremena čekanja strojeva = 88

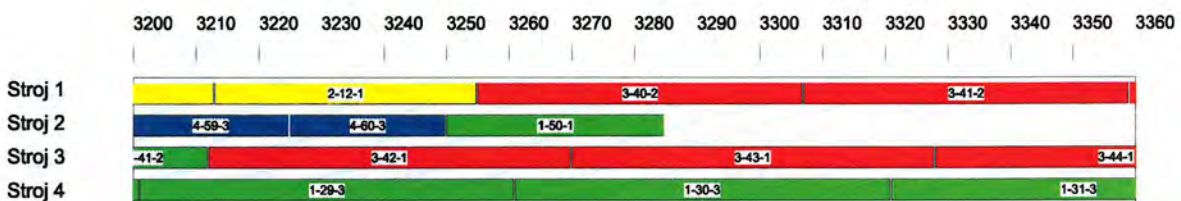
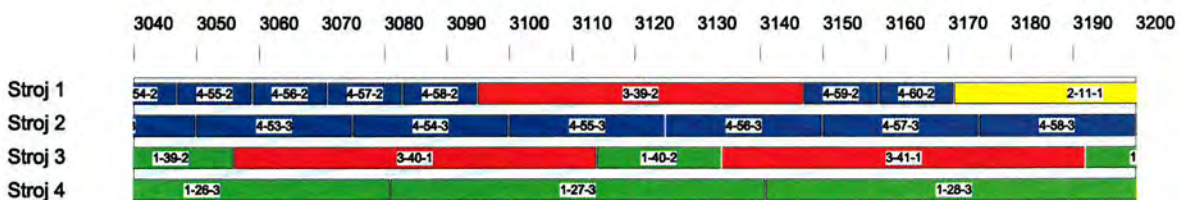
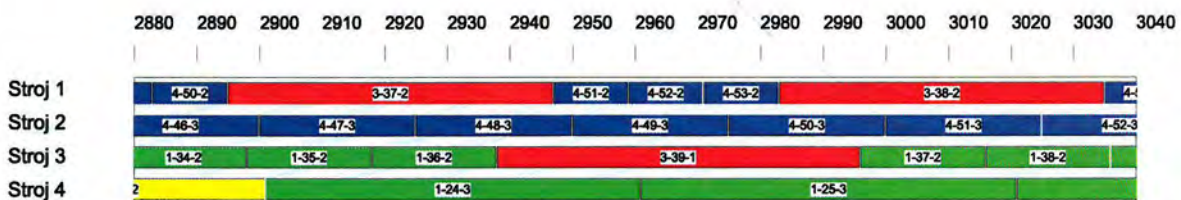
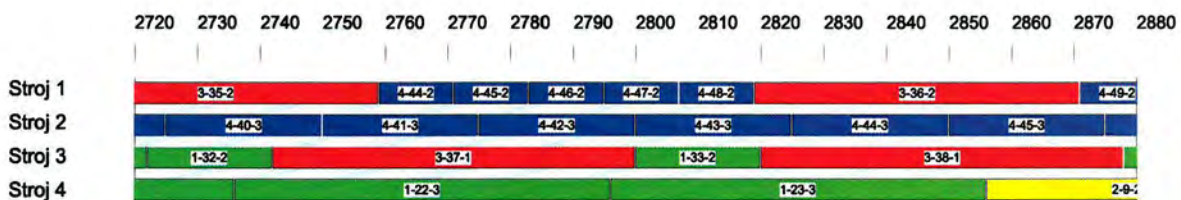
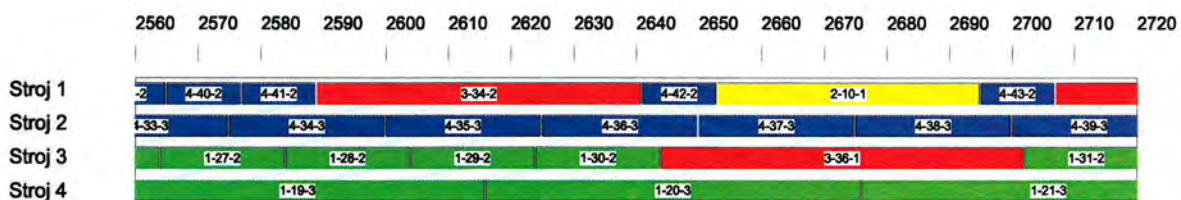
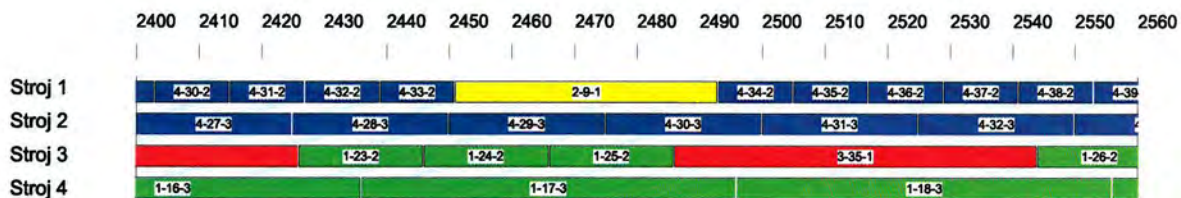
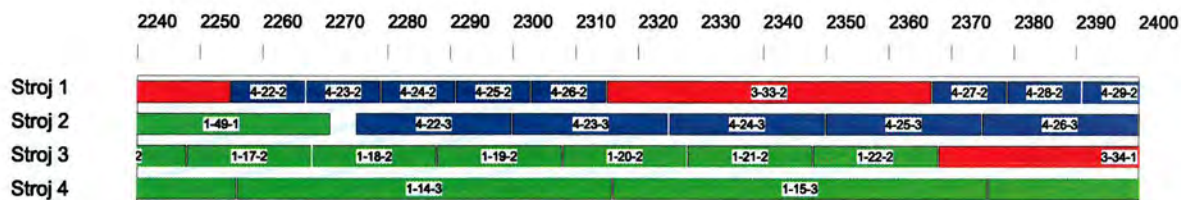
Broj generacija = 1000

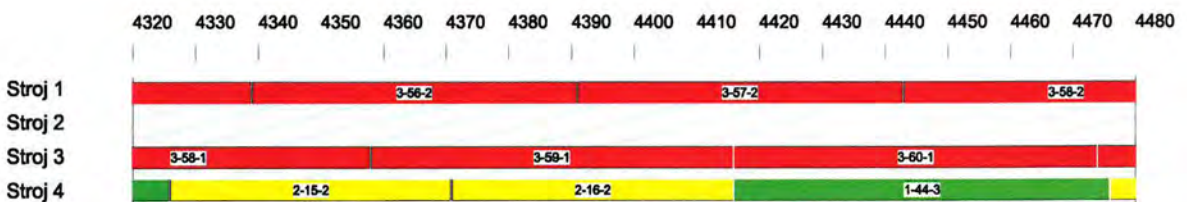
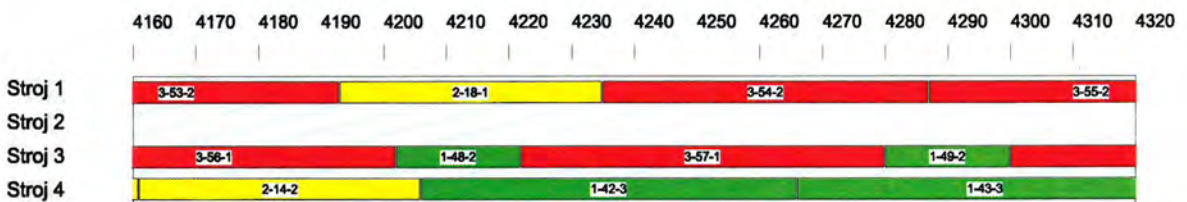
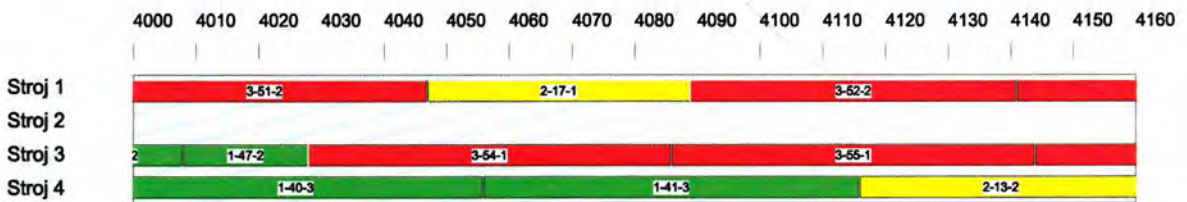
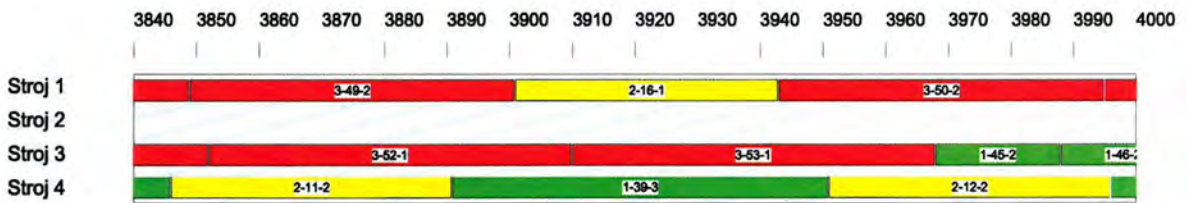
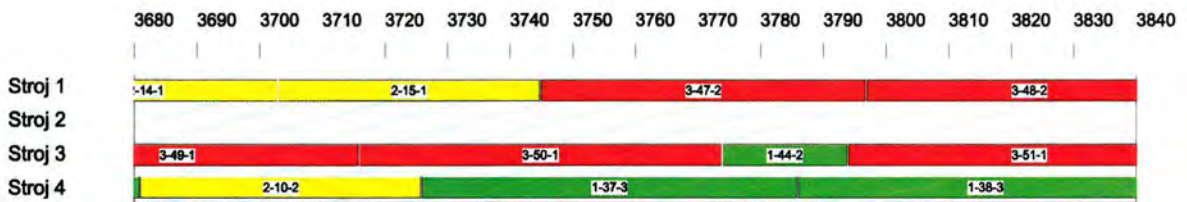
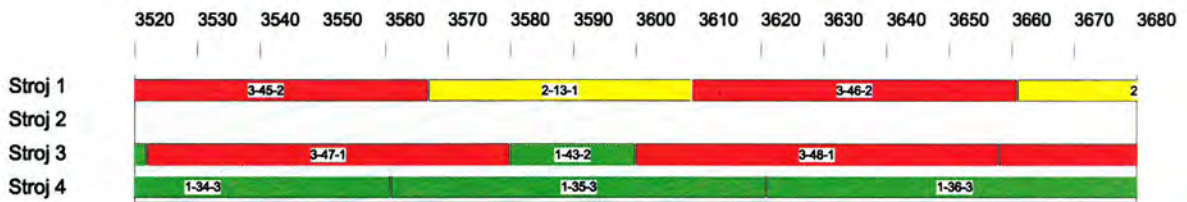
Veličina populacije = 250

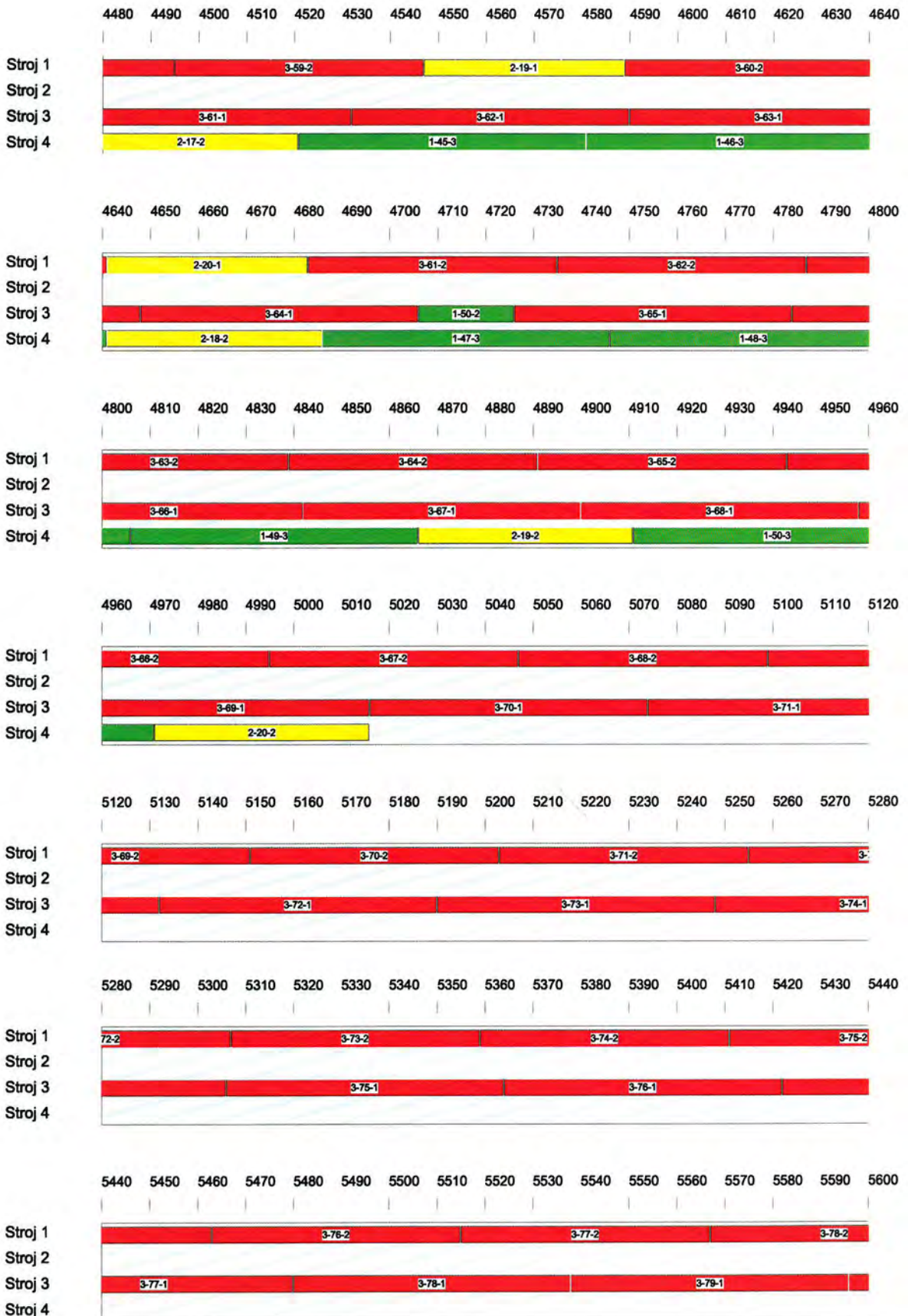
a)

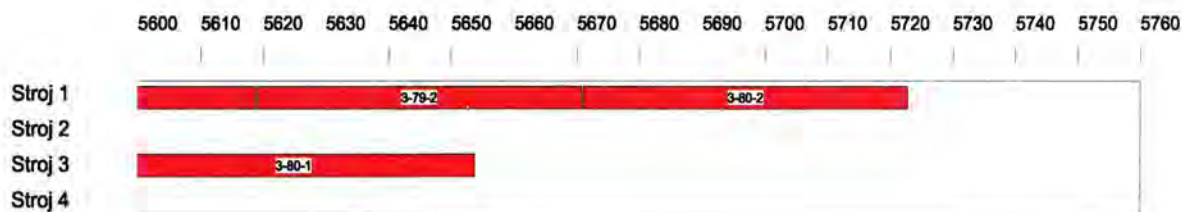












b)

- Stroj 1 započeo 2 završio 5723 radio 5720 (99.86%)
 vrsta dijelova 2 radio 840 (14.66%)
 vrsta dijelova 3 radio 4160 (72.63%)
 vrsta dijelova 4 radio 720 (12.57%)
 prazan 8 (0.14%)
- Stroj 2 započeo 10 završio 3285 radio 3250 (56.74%)
 vrsta dijelova 1 radio 1750 (30.55%)
 vrsta dijelova 4 radio 1500 (26.19%)
 prazan 2478 (43.26%)
- Stroj 3 započeo 14 završio 5654 radio 5640 (98.46%)
 vrsta dijelova 1 radio 1000 (17.46%)
 vrsta dijelova 3 radio 4640 (81.01%)
 prazan 88 (1.54%)
- Stroj 4 započeo 5 završio 5016 radio 4980 (86.94%)
 vrsta dijelova 1 radio 3000 (52.37%)
 vrsta dijelova 2 radio 900 (15.71%)
 vrsta dijelova 4 radio 1080 (18.85%)
 prazan 748 (13.06%)

c)

Sl. 6.2 Prikaz rezultata zadatka za kombinaciju kriterija – slučaj 2
 programom GJOB

- a) rezultat
 b) gantogram strojeva
 c) statistika strojeva

Iz slike 6.2 vidljivo je da je za ovu kombinaciju kriterija za minimalni ciklus izrade s kriterijem za minimalnim vremenom čekanja stroja na izradu uz isti ciklus izrade postignuto znatno bolje iskorištenje stroja 2 (98,9%) u smislu da je stroj 2 u vremenu 3285 slobodan i može ga se iskoristiti u druge svrhe. Također, ova kombinacija kriterija omogućuje znatno manji ciklus izrade dijelova 4.

Prethodne analize napravljene su uz pretpostavku da prebacivanje palete s izratkom iz izlaznog međuskladišta stroja na transporter ne ovisi o spremnosti transportera, već jedino ovisi o spremnosti stroja za prihvata palete izratka na stroj na kojem će biti izvedena naredna operacija izratka, što je naravno u suprotnosti sa stvarnim slučajem pri radu FPS-a. U *slučaju 3* definirat će se stvarno stanje pri radu FPS-a.

6.2.3. Slučaj 3.

Na slici 6.3 prikazan je rezultat optimizacije redosljednog ulaska pojedinih dijelova u proces prema kriteriju za što kraći ciklus izrade na FPS-u uz uključivanje novog kriterija o spremnosti transportnog sustava. U tu svrhu definirane su slijedeće značajke transportnog sustava:

- transport se može odvijati u oba smjera
- na jednoj paleti može biti samo jedan izradak
- transporter može transportirati samo jednu paletu u određenom trenutku
- izradak po dolasku ulazi u radni prostor stroja (ne postoji ulazno međuskladište stroja)
- u vrijeme transporta ulazi i vrijeme transporta i vrijeme ukrcavanja paleta na transporter i vrijeme iskrcavanja s transportera na ulaznu poziciju stroja
- paleta s izratkom nakon izvršenja operacije ulazi u izlazno međuskladište stroja koje može prihvatiti više paleta
- redosljed izuzimanja paleta iz izlaznog međuskladišta stroja je slobodan
- kod stroja na kojem se izvodi zadnja operacija jedinke dijela, paleta će biti prebačena na transporter u prvom slobodnom trenutku

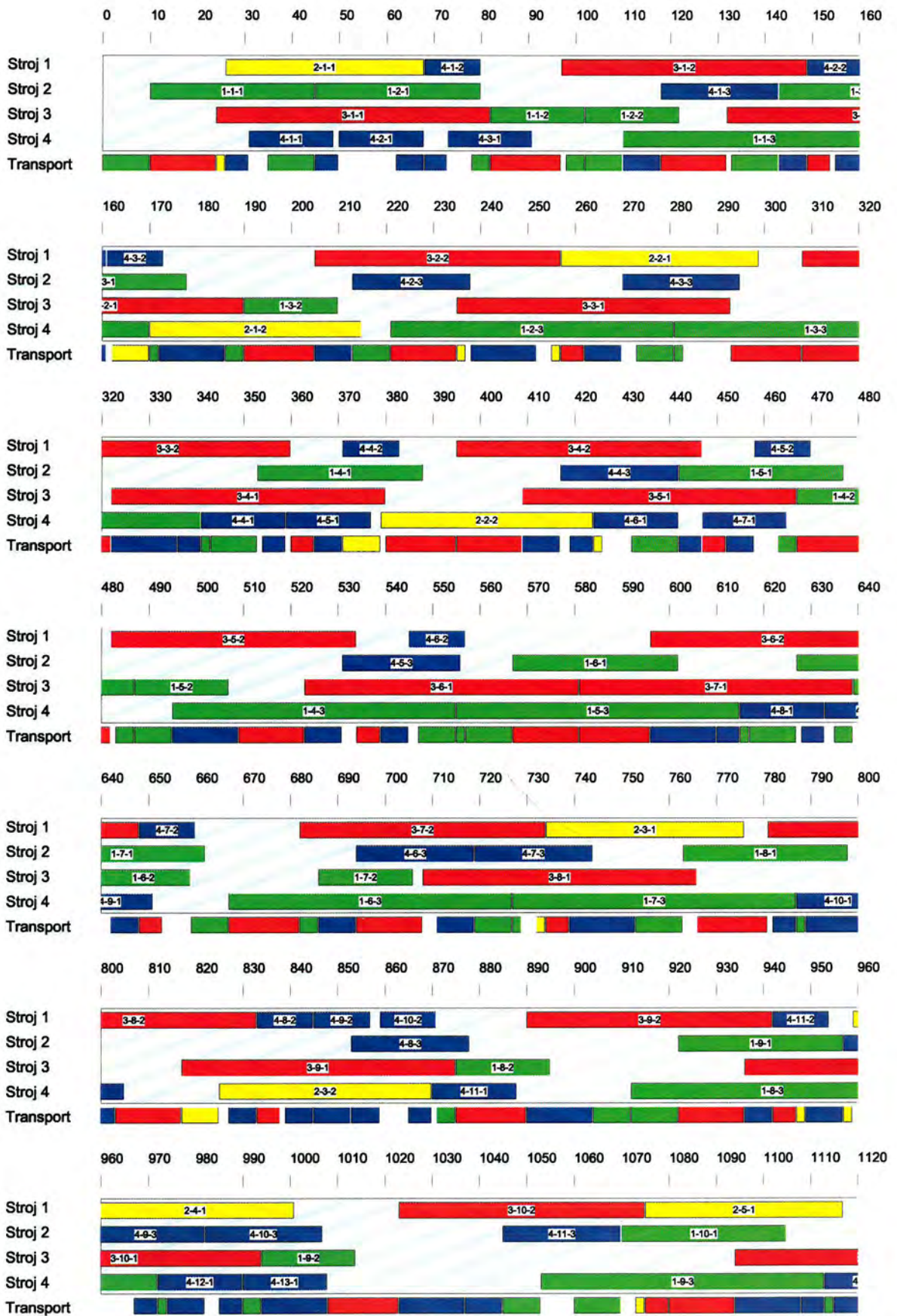
Primijenjen je genetski algoritam uz 740 gena po kromosomu, 250 kromosoma u populaciji ($popSize = 250$), 1500 generacija ($maxgen = 1500$), stohastički odabir s zamjenom, linearno skaliranje, križanje zamjenom parcijalnih redosljeda ($pcross = 0.8$), te mutaciju baziranu na traženju susjedstva ($pmutation = 0.003$).

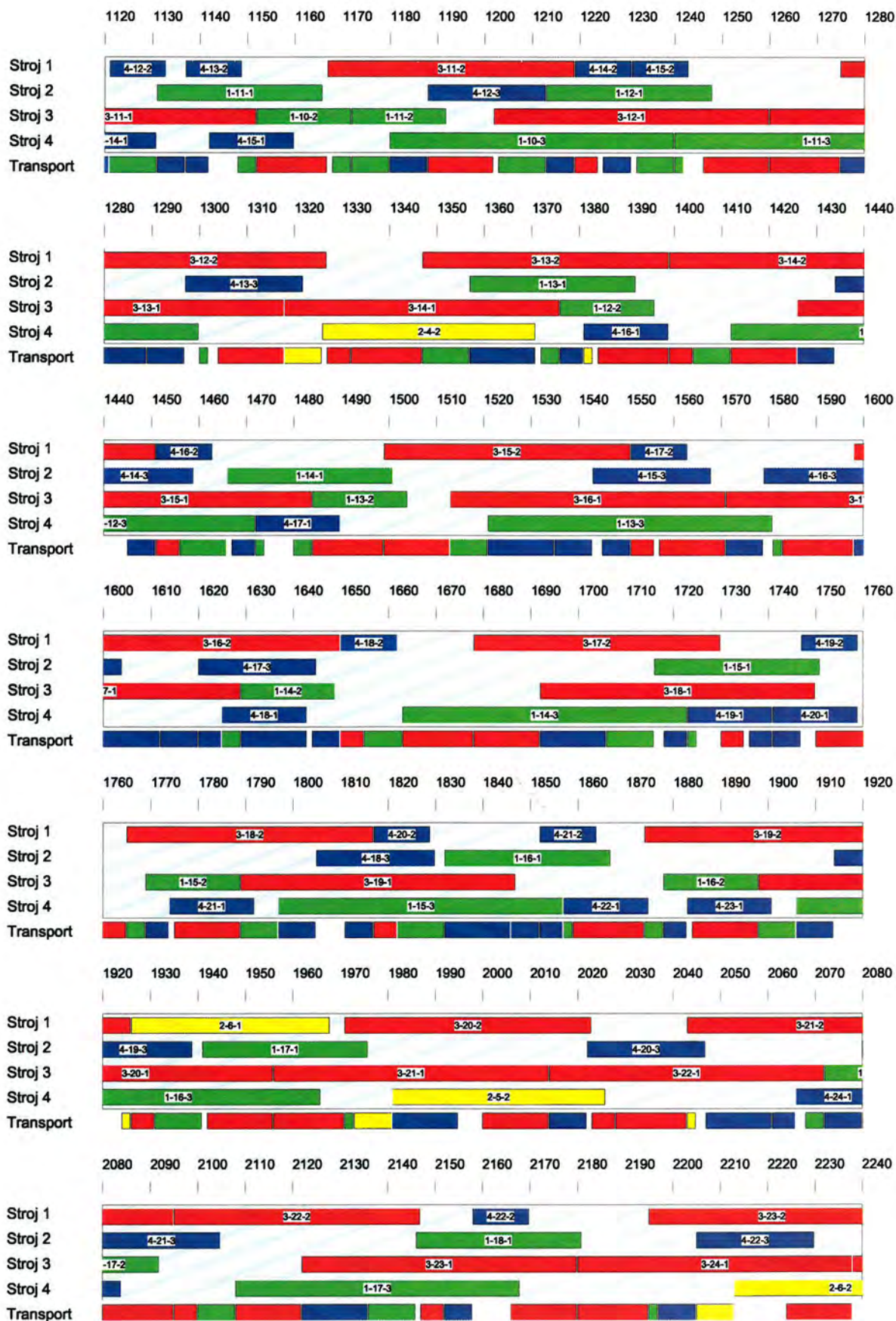
Maksimalno vrijeme = 6922

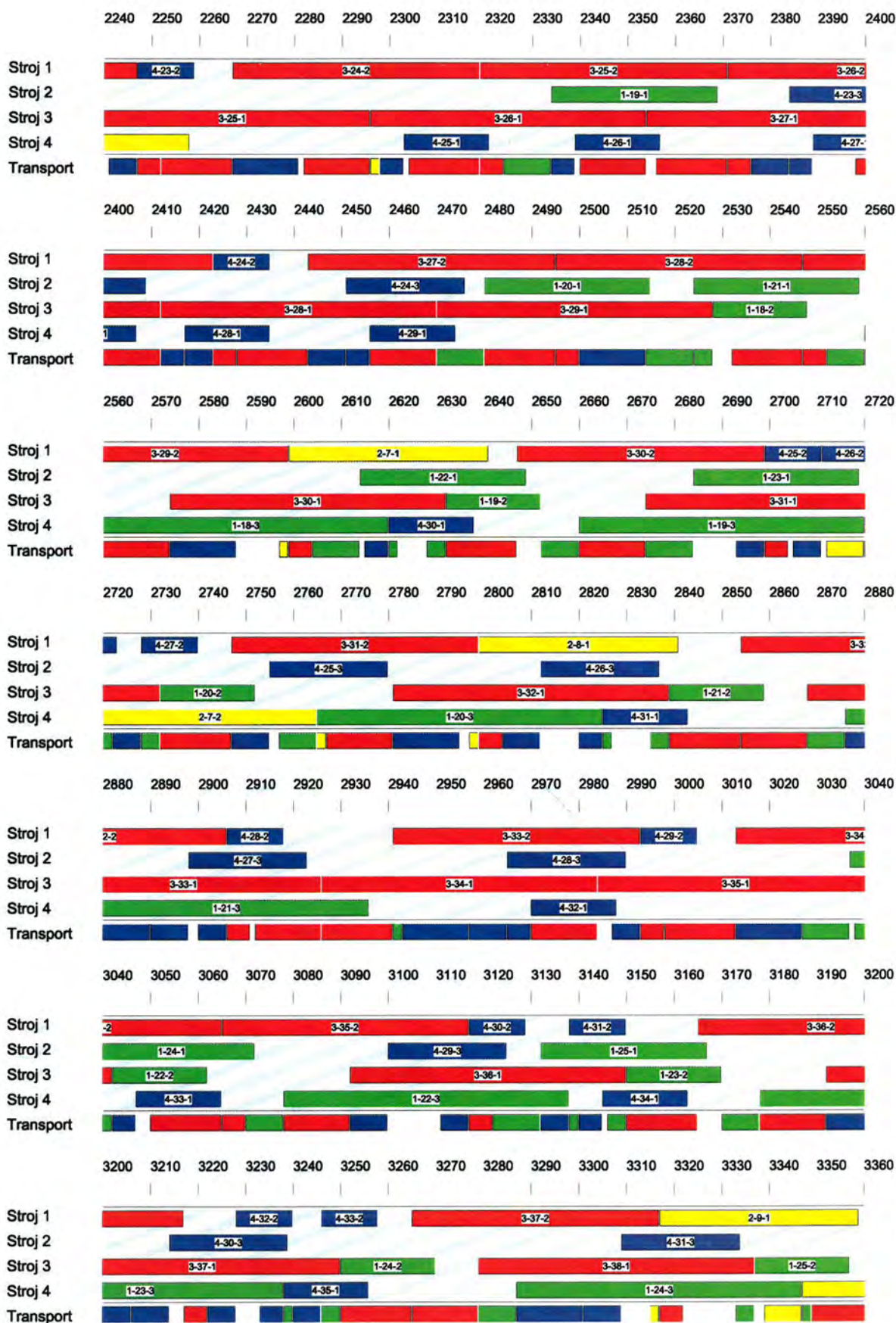
Broj generacija = 1500

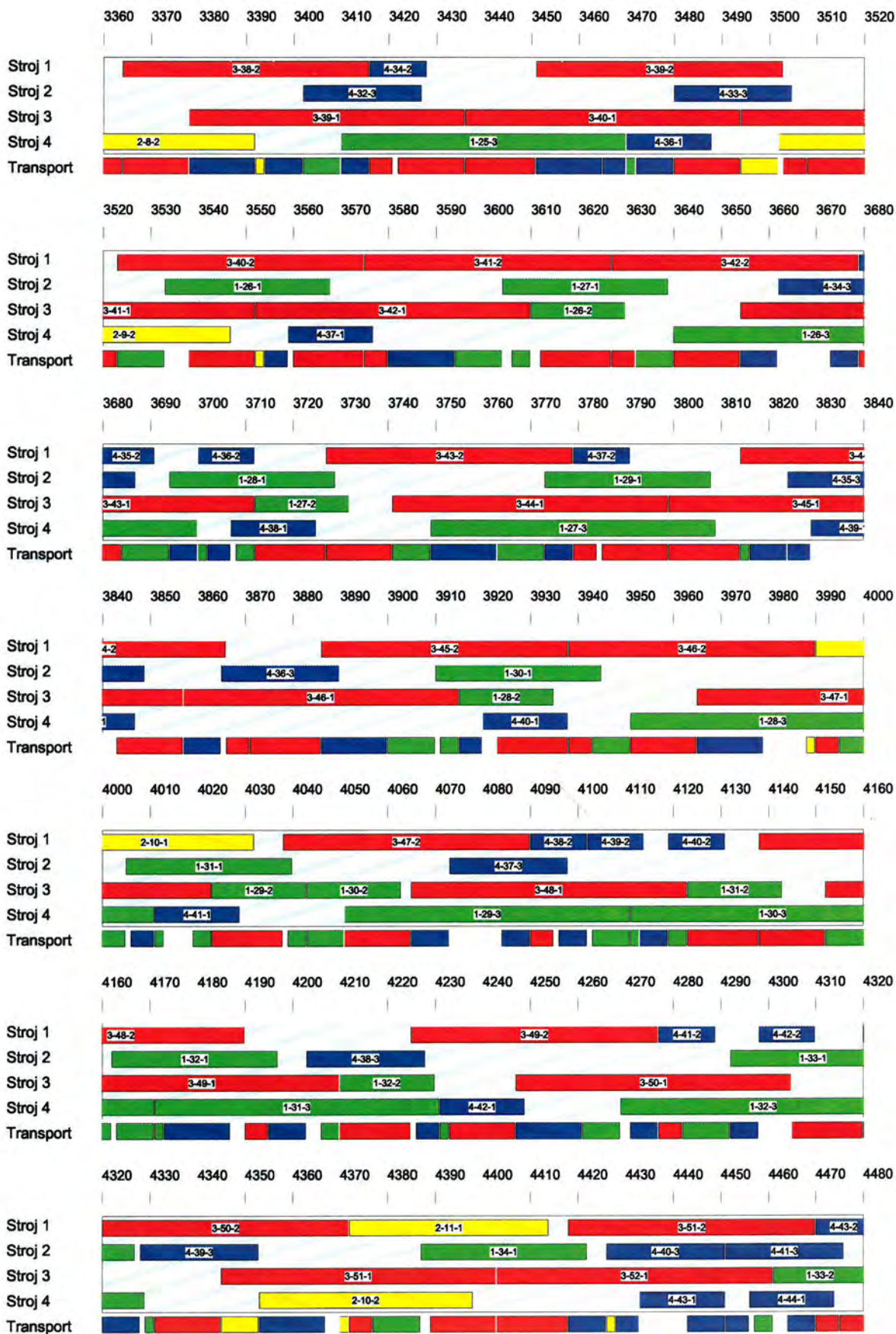
Veličina populacije = 250

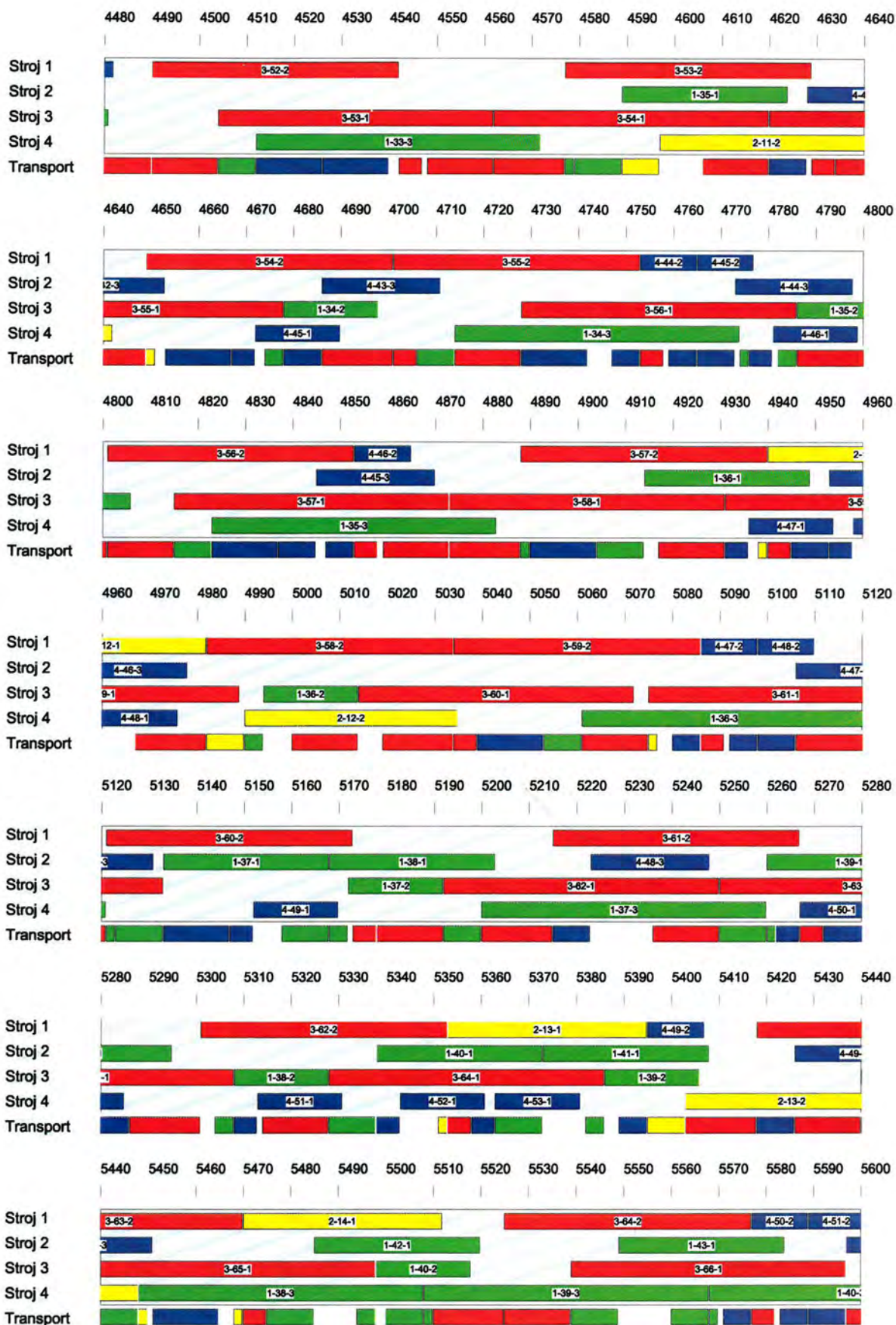
a)

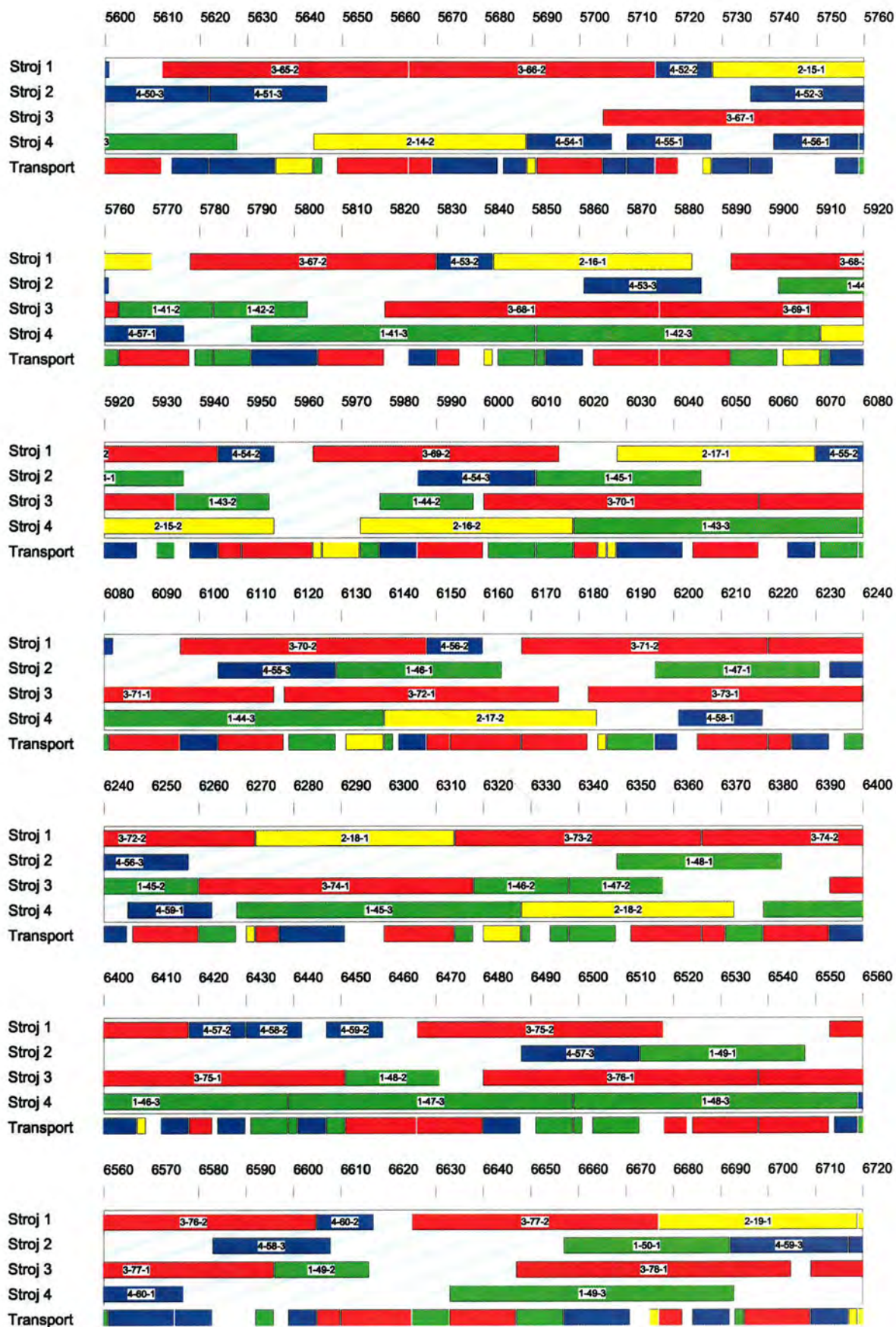


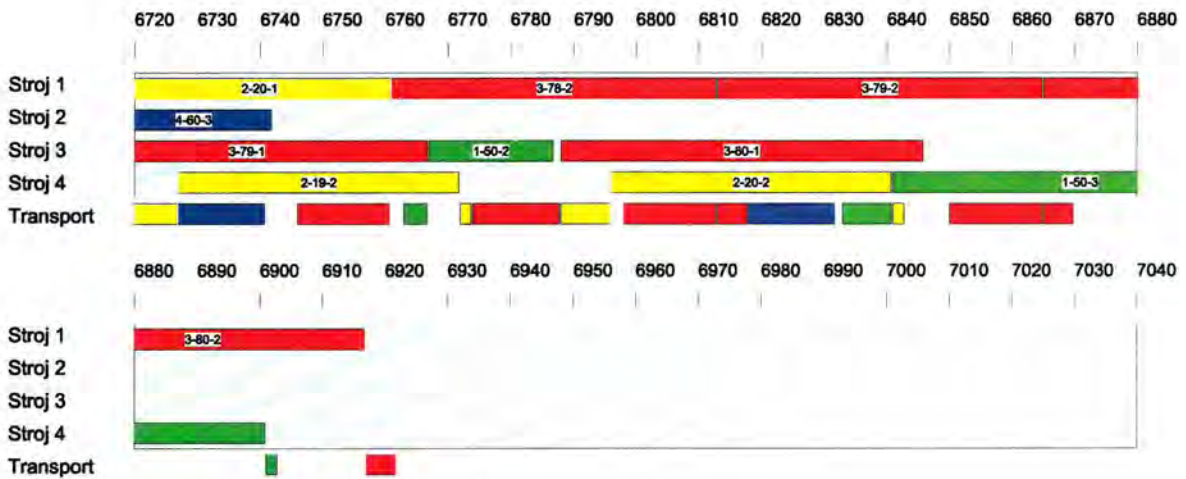












b)

Stroj 1 započeo 26 završio 6917 radio 5720 (82.64%)
 vrsta dijelova 2 radio 840 (12.14%)
 vrsta dijelova 3 radio 4160 (60.10%)
 vrsta dijelova 4 radio 720 (10.40%)
 prazan 1202 (17.36%)

Stroj 2 započeo 10 završio 6742 radio 3250 (46.95%)
 vrsta dijelova 1 radio 1750 (25.28%)
 vrsta dijelova 4 radio 1500 (21.67%)
 prazan 3672 (53.05%)

Stroj 3 započeo 24 završio 6846 radio 5640 (81.48%)
 vrsta dijelova 1 radio 1000 (14.45%)
 vrsta dijelova 3 radio 4640 (67.03%)
 prazan 1282 (18.52%)

Stroj 4 započeo 31 završio 6901 radio 4980 (71.94%)
 vrsta dijelova 1 radio 3000 (43.34%)
 vrsta dijelova 2 radio 900 (13.00%)
 vrsta dijelova 4 radio 1080 (15.60%)
 prazan 1942 (28.06%)

Transporter započeo 0 završio 6922 radio 6140 (88.70%)

c)

Sl. 6.3 Prikaz rezultata zadatka za kombinaciju kriterija – slučaj 3 programom GJOB

a) rezultat

b) gantogram strojeva

c) statistika strojeva i transportnog sustava

Usporedi li se maksimalno vrijeme na slici 6.3 s maksimalnim vremenom na slici 6.1 i slici 6.2 može se vidjeti da je ono na slici 6.3 znatno duže što je i bilo realno za očekivati.

Na temelju slike 6.3 može se vidjeti da je ciklus izrade prilično velik, strojevi dosta čekaju na izradu, a automatizirani transportni sustav nedovoljno je iskorišten. Kombinacija kriterija za minimalnim ciklusom izrade s kriterijom za minimalnim čekanjem stroja na izradu u ovom slučaju daje znatno duži ciklus izrade, a time težnja za što boljim iskorištenjem pojedinog stroja ili smanjenje ciklusa izrade pojedine vrste dijelova, nije opravdana.

Poboljšanje je moguće postići uvođenjem ulaznih međuskладишта stroja koji mogu prihvatiti više paleta izradaka. Time se postiže da prebacivanje palete s izratkom iz izlaznog međuskладишта stroja na transporter ne ovisi o spremnosti stroja za prihvrat izratka na stroj na kojem će biti izvedena naredna operacija, već ovisi samo o spremnosti transportera. U svrhu poboljšanja definirat će se *slučaj 4*.

6.2.4. Slučaj 4.

Na slici 6.4 prikazan je rezultat optimizacije redosljednog odvijanja procesa prema kriteriju za što kraći ciklus izrade na FPS-u istih karakteristika kao u prethodnom slučaju s jednim dodatkom vezanim za transportni sustav:

- ulazno međuskладиšte stroja može prihvatiti više paleta

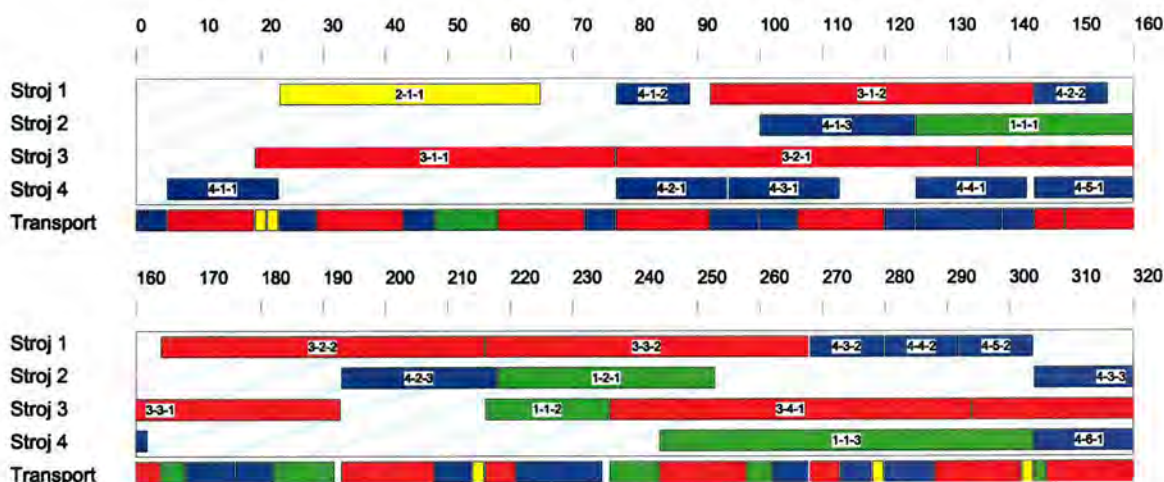
Uvođenjem ulaznog međuskладишта stroja koje može prihvatiti više paleta, slučaj 4 znatno je pojednostavljen u odnosu na *slučaj 3*. Isprobavanjem različitih vrijednosti parametara genetskog algoritma pokazalo se dovoljnim znatno smanjiti broj generacija. Primijenjen je genetski algoritam uz 740 gena po kromosomu, 250 kromosoma u populaciji ($popSize = 250$), 150 generacija ($maxgen = 150$), stohastički odabir s zamjenom, linearno skaliranje, križanje zamjenom parcijalnih redosljeda ($p_{cross} = 0.8$), te mutaciju baziranu na traženju susjedstva ($p_{mutation} = 0.003$).

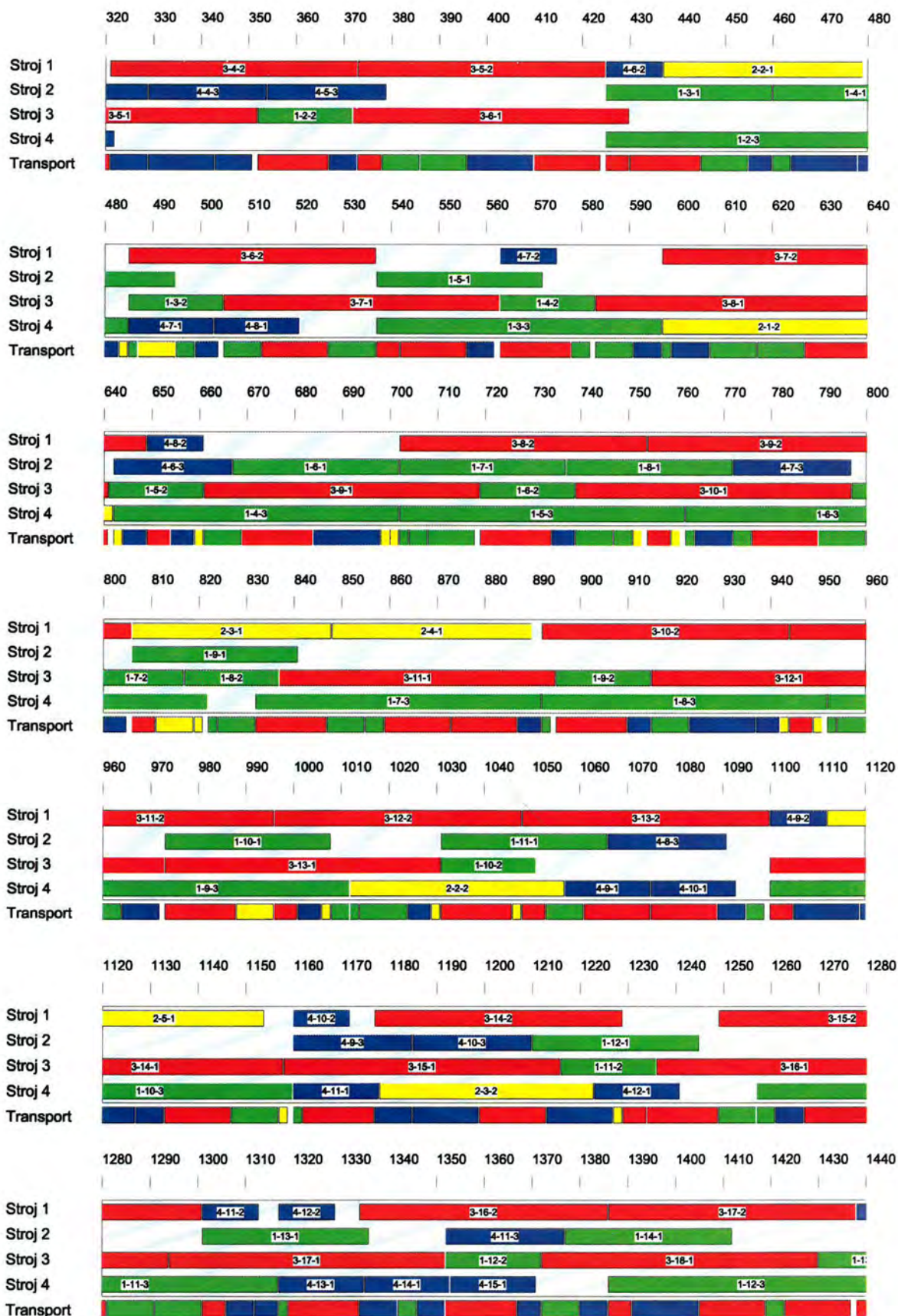
Maksimalno vrijeme = 6361

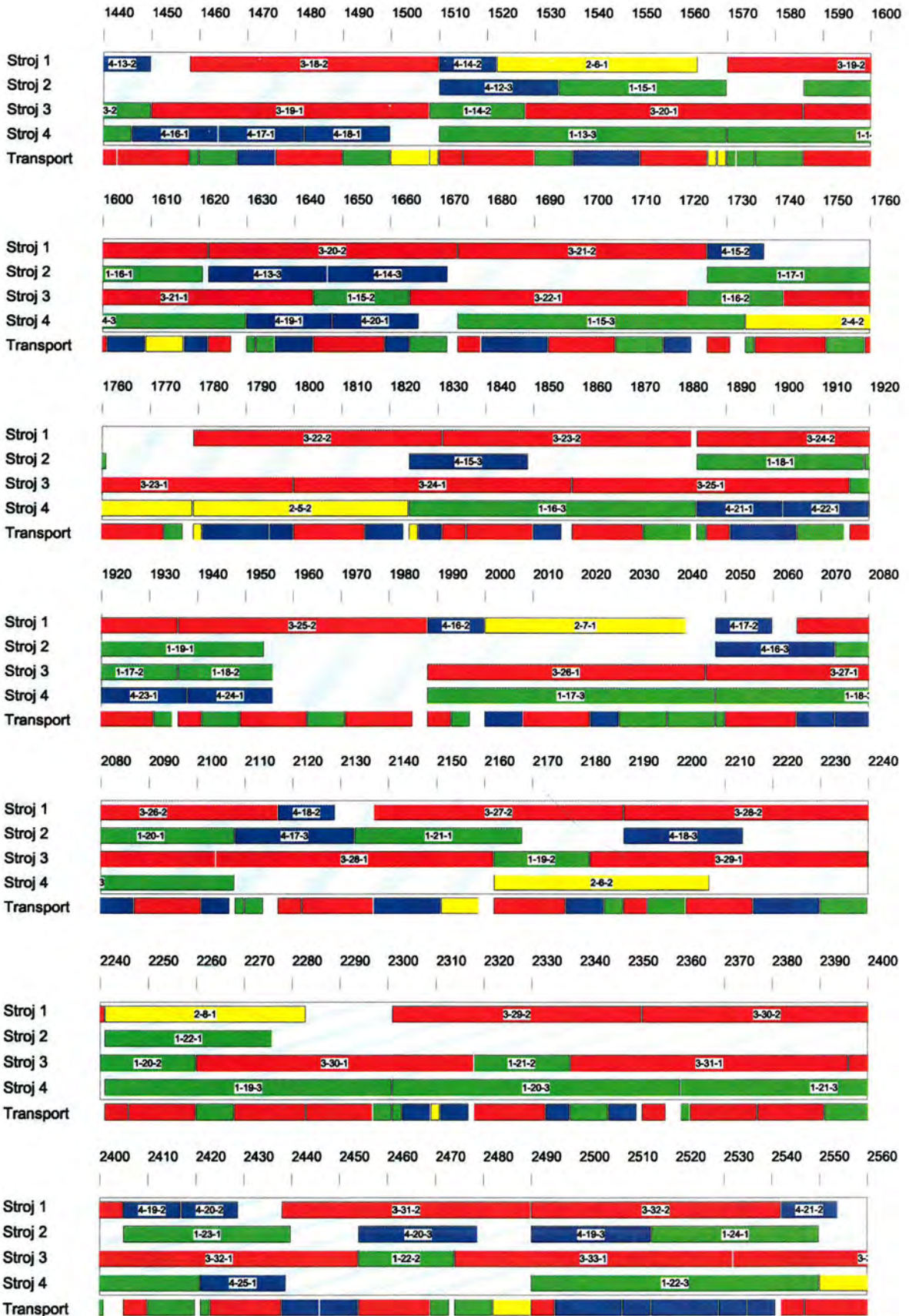
Broj generacija = 150

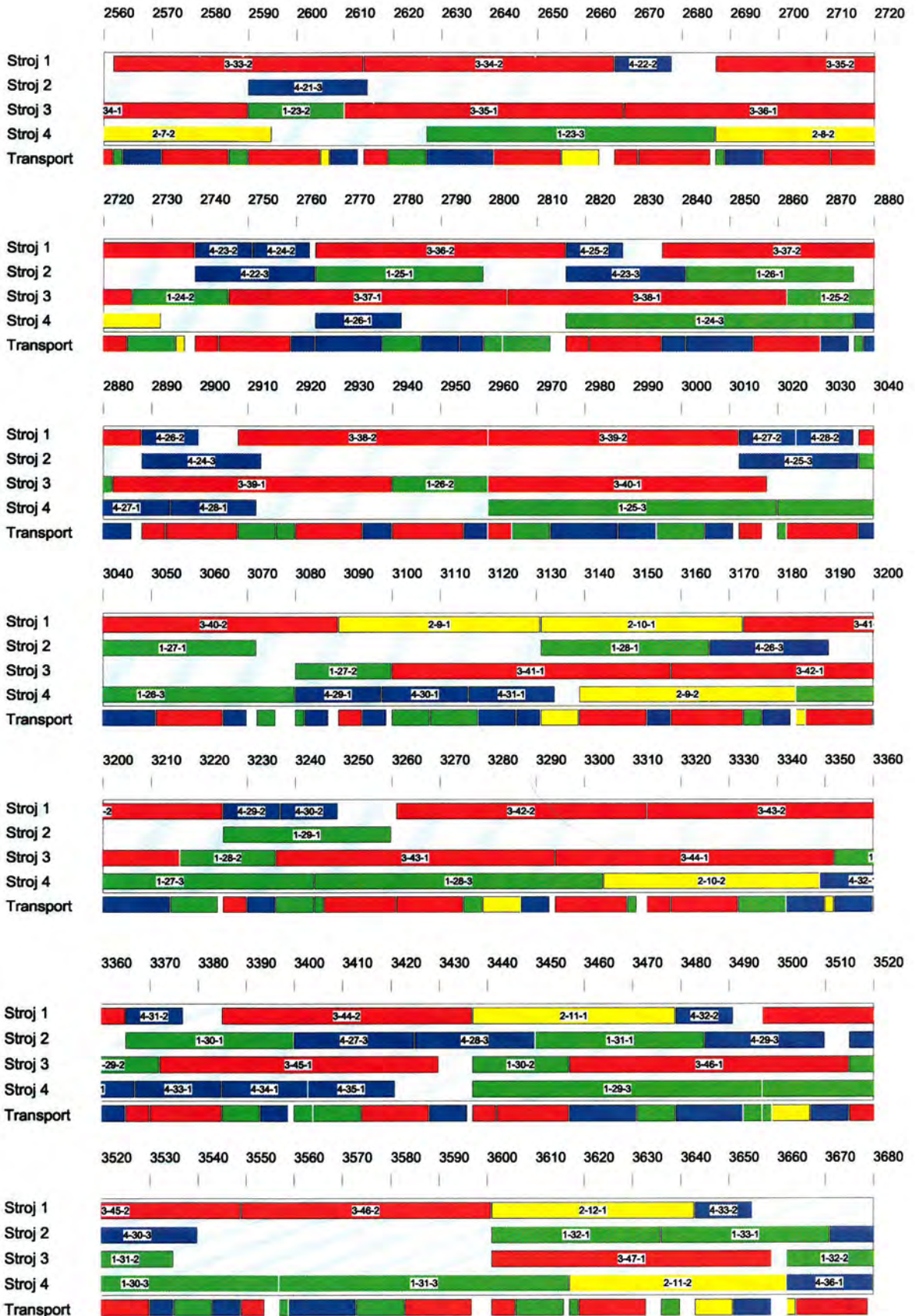
Veličina populacije = 250

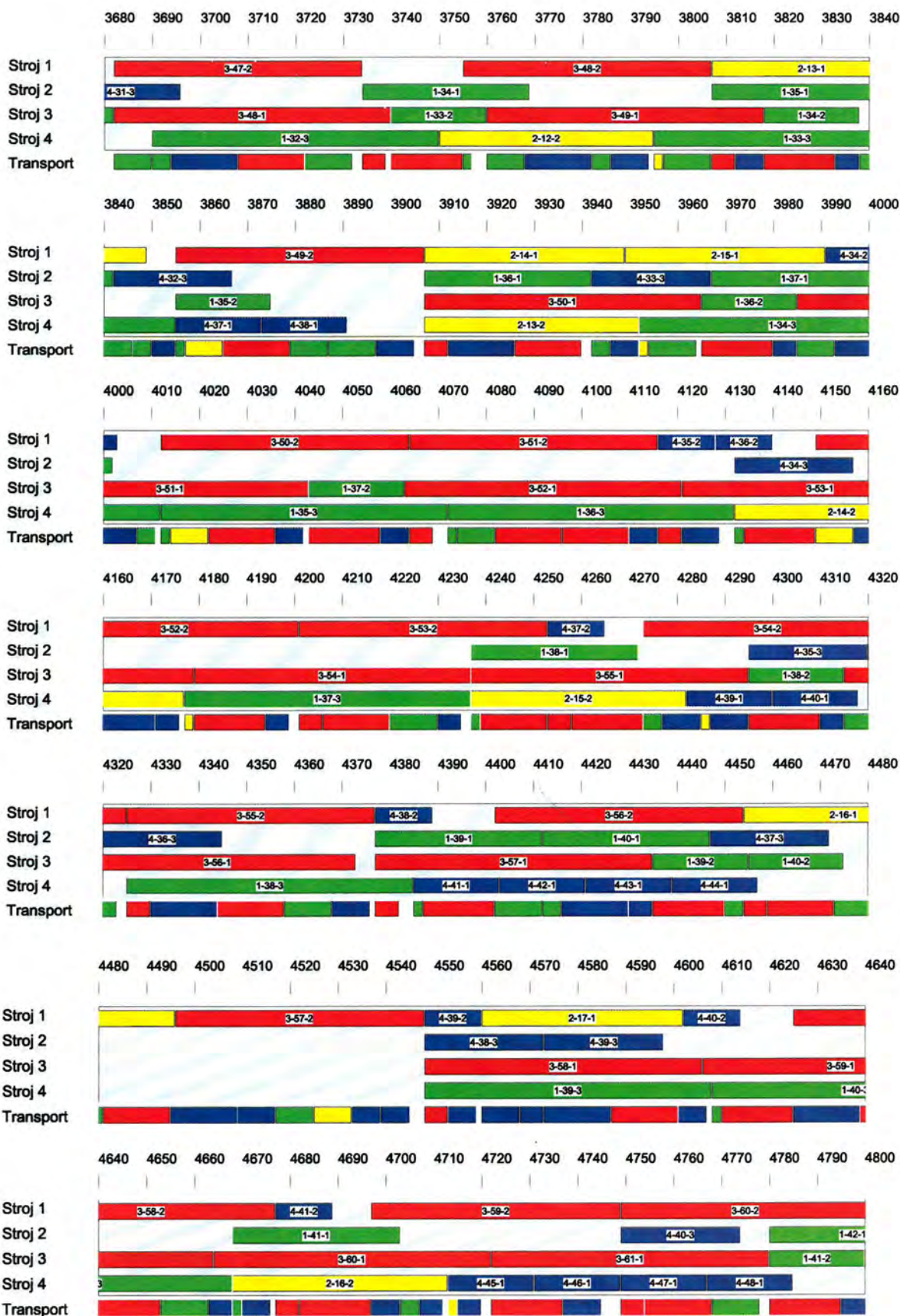
a)

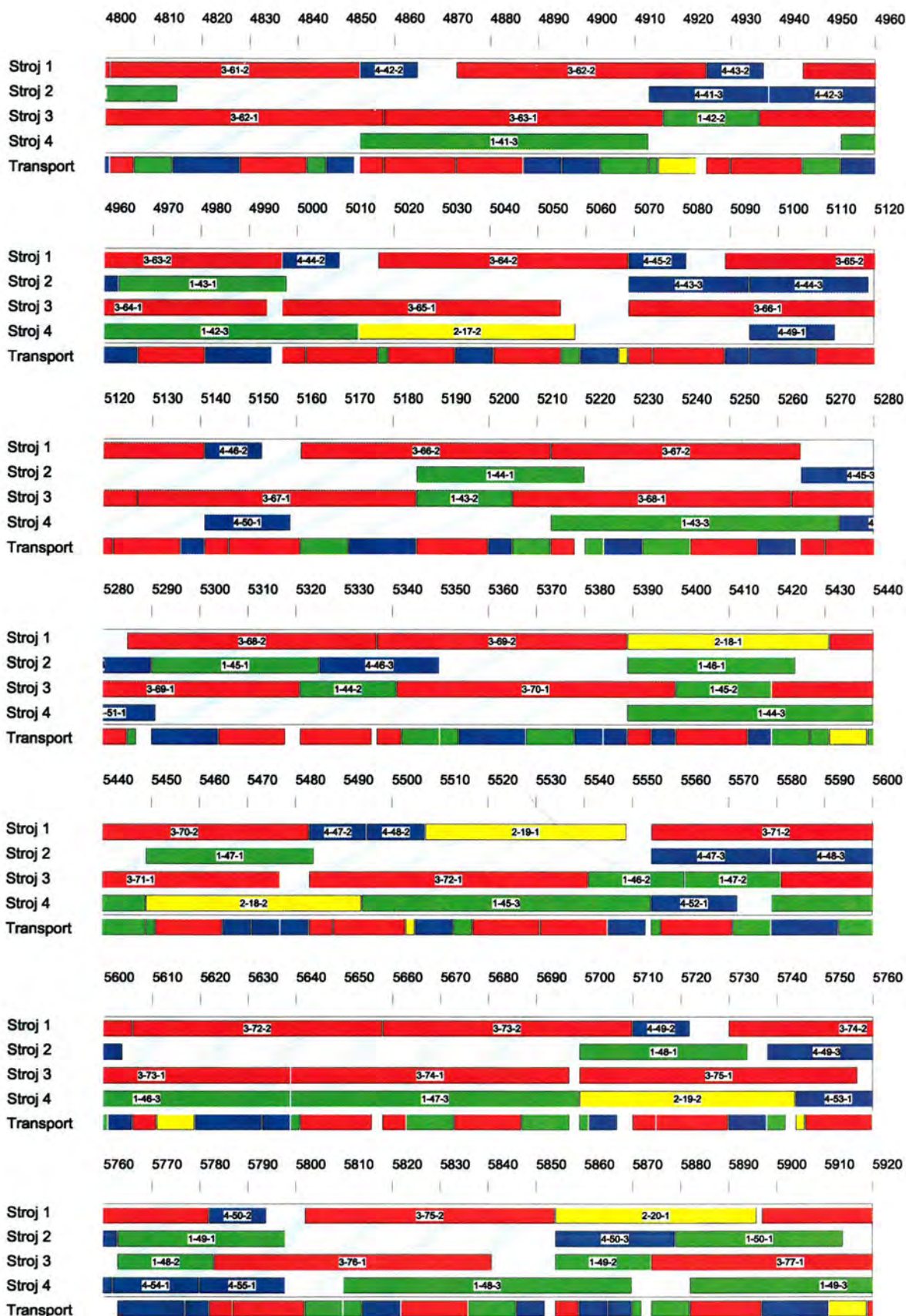


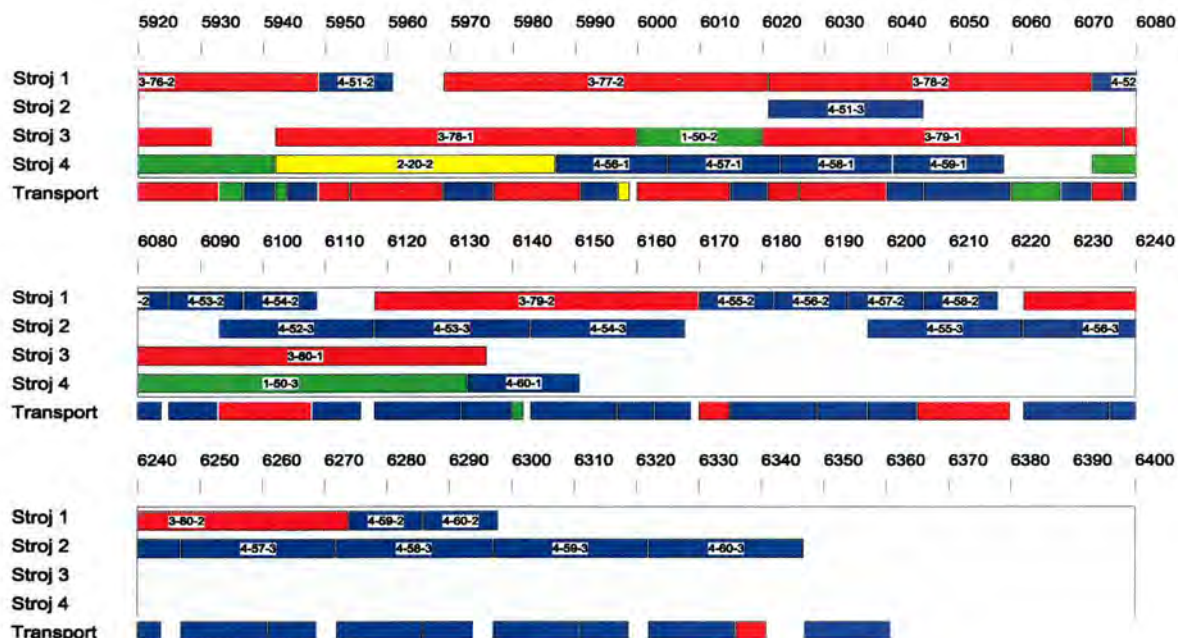












b)

Stroj 1 započeo 23 završio 6298 radio 5720 (89.92%)

vrsta dijelova 2 radio 840 (13.21%)

vrsta dijelova 3 radio 4160 (65.40%)

vrsta dijelova 4 radio 720 (11.32%)

prazan 641 (10.08%)

Stroj 2 započeo 100 završio 6347 radio 3250 (51.09%)

vrsta dijelova 1 radio 1750 (27.51%)

vrsta dijelova 4 radio 1500 (23.58%)

prazan 3111 (48.91%)

Stroj 3 započeo 19 završio 6136 radio 5640 (88.67%)

vrsta dijelova 1 radio 1000 (15.72%)

vrsta dijelova 3 radio 4640 (72.94%)

prazan 721 (11.33%)

Stroj 4 započeo 5 završio 6151 radio 4980 (78.29%)

vrsta dijelova 1 radio 3000 (47.16%)

vrsta dijelova 2 radio 900 (14.15%)

vrsta dijelova 4 radio 1080 (16.98%)

prazan 1413 (21.71%)

Transporter započeo 0 završio 6361 radio 6140 (96.53%)

c)

Sl. 6.4 Prikaz rezultata zadatka za kombinaciju kriterija – slučaj 4
programom GJOB

a) rezultat

b) gantogram strojeva

c) statistika rada strojeva i transportnog sustava

Usporedi li se maksimalno vrijeme na slici 6.4 koje iznosi 6361 s maksimalnim vremenom na slici 6.3 koje iznosi 6922 vidljivo je da je ono na slici 6.4 znatno kraće, a iskorištenje transporterera znatno bolje (96,53%), dok je u slučaju dobivenim na slici 6.3 iznosilo 88,7%.

Bolji rezultat u smislu kraćeg ciklusa izrade i boljeg iskorištenja transportnog sustava posljedica je ulaznog međuskladišta stroja, jer u takvom slučaju transport izratka sa izlaznog međuskladišta stroja u kojem je redosljed izuzimanja izradaka slobodan, ovisi samo o spremnosti transporterera, dok u slučaju kad ulazno međuskladište ne postoji ovisi i o spremnosti stroja za prihvrat izratka što je jedno ograničenje više.

Zbog toga, primjena genetskog algoritma kod FPS-a koji nema ulazna međuskladišta stroja, a time ima i jedno ograničenje više, zahtijevala je puno veći broj generacija (1500) od analize FPS-a sa ulaznim međuskladištem stroja (150).

Usporede li se rezultati dobiveni na slici 6.1 (*slučaj 1*), slici 6.2 (*slučaj 2*), slici 6.3 (*slučaj 3*) i slici 6.4 (*slučaj 4*) vidljivo je, da je u sva četiri slučaja dosta velika populacija (250 jedinki), parametar križanja je isti, dok parametar mutacije i broj generacija varira ovisno o složenosti problema.

Rješavanje *slučaja 1* i *slučaja 4* je najjednostavnije, tj. ima najmanje ograničenja, a to su:

- transport izratka sa izlaznog međuskladišta stroja ovisi o spremnosti stroja (problem prikazan na slici 6.1),
- transport izratka sa izlaznog međuskladišta stroja ovisi o spremnosti transporterera (problem prikazan na slici 6.2)

pa je potreban i manji broj generacija za dobivanje optimalnog rezultata.

Rješavanje *slučaja 2* je malo složenije u odnosu na rješavanje *slučaja 1* jer postoji jedno ograničenje više, a to je:

- što bolje iskorištenje stroja 2 ,

pa je potreban i veći broj generacija (1000) za dobivanje optimalnog rezultata

Rješavanje *slučaja 3* je najsloženije, tj. ima najviše ograničenja, a to je:

- transport izratka sa izlaznog međuskladišta stroja ovisi o spremnosti narednog stroja,
- transport izratka sa izlaznog međuskladišta stroja ovisi o spremnosti transporterera,

pa je potreban i velik broj generacija (1500) za dobivanje optimalnog rezultata.

7. ZAKLJUČAK

Uspješno rješavanje problema optimizacije ukupnog toka odvijanja procesa u okviru radnog naloga jedna je od ključnih podloga za ostvarenje primjerene ukupne proizvodnosti i ekonomičnosti korištenja proizvodnog sustava. Kad su u pitanju fleksibilni proizvodni sustavi, značaj optimizacije ciklusa izrade kao pokazatelja vremenskog zauzeća proizvodnog sustava pri simultanoj proizvodnji dijelova prema određenom radnom nalogu je posebno izražen, obzirom na naglašenu potrebu stvaranja uvjeta za visoku ukupnu iskoristivost FPS-a, što je temeljni preduvjet za gospodarstvenu isplativost velikih ulaganja vezanih za njihovu izgradnju i korištenje.

Pretpostavka za dobro iskorištenje proizvodnog sustava je formiranje parcijalnog radnog zadatka s povoljnom strukturom radnih naloga. Povoljna struktura radnih naloga podrazumjeva skup dijelova, za čiju će proizvodnju trebati svi obradni strojevi sustava s njihovim približno podjednakim vremenskim opterećenjem uz najmanji utrošak vremena i dobru sinhronizaciju rada osoblja i korištenja alata. Dobro iskorištenje FPS-a, kao jedan od postavljenih glavnih ciljeva, ovisi o nizu utjecajnih faktora od kojih se mogu istaći:

- tehnička i organizacijska rješenja primijenjena pri oblikovanju i radu sustava, koja se iskazuju kao njegova tehnička raspoloživost
- značajke parcijalnog radnog zadatka sa stajališta broja dijelova, količina, broja operacija, strojnih vremena i vremena prepremanja radnih mjesta ovisno o redoslijedu provođenja
- redoslijedu odvijanja pojedinih operacija parcijalnog radnog zadatka

Budući da se na strukturu radnih naloga može samo ograničeno utjecati zbog primarnog utjecaja prioriteta, koji diktira plan montaže, uz postojeća tehnička i organizacijska rješenja sustava ostaje redoslijed odvijanja poslova u okviru radnog naloga kao primarni element o kojem ovisi stupanj iskorištenja FPS-a.

U ovom radu dat je doprinos rješenju problema upravljanja FPS-a s različitim strojevima pri realiziranju pojedinih radnih naloga, čija struktura proizlazi iz planova montaže što znači, da je diktirana tržišnim zahtjevima po pitanju tipova dijelova i količina. Budući da struktura pojedinih radnih naloga nije u potpunosti usklađena s eksploatacijskim karakteristikama FPS-a koje proizlaze prvenstveno iz značajki strukture jezgre sustava, suočavamo se s problemom, kako u takvim uvjetima upravljati realiziranjem radnog naloga i pri tom postići glavni cilj: što manje vremensko zauzeće sustava odnosno što kraći ciklus izrade.

Problem je u ovom radu analiziran, a provedena istraživanja rezultirala su razvojem metode koja se temelji na dvije faze rješavanja problema upravljanja realiziranja radnog naloga 4.1:

- I. faza upravljanja – izbor optimalne kombinacije planova procesa
- II. faza upravljanja – optimizacija redoslijednog odvijanja procesa

U prvoj fazi određen je izbor one kombinacije planova procesa između razvijenih planova procesa koja pri simultanoj proizvodnji određenog mixa dijelova u zadanim količinama osigurava najmanje vremensko opterećenje proizvodnog sustava poznatih karakteristika.

U drugoj fazi provedeno je odgovarajuće upravljanje redosljednog odvijanja pojedinih operacija parcijalnog radnog zadatka kako bi se smanjio ukupni ciklus izrade, tj. optimiran je redosljed ulaska pojedinih dijelova u proces izrade i izvođenja pojedinih operacija vodeći pri tome računa za čim kraće vremensko opterećenje sustava kao cjeline (radne stanice, automatizirani transportni sustav) i što jednakomjernije opterećenje pojedinih radnih stanica.

Pri tom je bilo nužno predložiti metodu, koja će imati veliku brzinu obrade ulaznih podataka i definiranja optimalnog izbora redosljeda izvršenja pojedinih operacija iz cjeline radnog naloga, kako bi se osigurala primjenjivost pri operativnom upravljanju radom FPS-a i postigao glavni cilj.

Ostvarenje cilja rada provedeno je razvojem metode temeljene na primjeni genetskog algoritma u obje faze upravljanja. Poznavajući svojstva genetskog algoritma, može se zaključiti da su dobivena rješenja optimizacijskih zadataka vrlo blizu točnih optimuma, a optimizacija II. faze upravljanja zbog svoje složenosti i regularnosti ili se uopće ne bi mogla riješiti nekom drugom metodom ili bi bila riješena manje učinkovito.

Vrijednost ne čine samo riješeni pojedinačni problemi optimalnog upravljanja FPS-a, već i razvijeni računalni program koji se može primjeniti na bilo koji novi problem proračuna i optimizacije redosljednog odvijanja procesa na FPS-u.

Usporedbom rezultata za 4. slučaja prikazanih u podpoglavlju 6.2. pokazano je kako veličinu populacije prilagoditi broju gena po kromosomu tj. veličini kromosoma jedinke, te kako broj generacija varira ovisno o složenosti problema. Mijenjanje veličine populacije i broja generacija iziskuje i promjenu vrijednosti parametara križanja (*pcross*) i mutacije (*pmutation*). Svi rezultati su dobiveni tako da su isprobavane različite veličine populacije, broja generacija, a time i različite vrijednosti parametara križanja i mutacije. Na temelju tako stečenog iskustva prikazani su samo najbolji rezultati.

U budućim istraživanjima moglo bi se pokušati poboljšati taj dio isprobavanja na način da se unutar algoritma programira procedura koja bi davala optimalne vrijednosti veličine populacije, broja generacija i parametara križanja i mutacije. Pored toga svakako treba pratiti razvoj naprednih operatora i novih mogućnosti genetskog algoritma te kvalitetnim novim rezultatima unaprijeđivati računalni program.

LITERATURA

- [1] Burbidge, J.L.: *Budućnost serijske proizvodnje*, Zbornik radova XIX Savjetovanja proizvodnog strojarstva jugoslavije, Kragujevac, pp.67-74, 1985.
- [2] Balič, J.: *Contribution to Integrated Manufacturing*, DAAAM International, Vienna, 1999.
- [3] Balič, J.: *Flexible Manufacturing System, Development – Structure – Operation – Handling – Tooling*, DAAAM International, Vienna, 2001.
- [4] Balič, J.: *Manufacturing Systems for the Third Millenium*, University of Maribor, Faculty of Mechanical Engineering, Maribor, 2001.
- [5] Treffert, H.: *Neue technologien - Herausforderung an das betriebliche Bildungswesen*, Werkstatt und Betrieb, 122 (1989) 3
- [6] Kovačević, R.: *Numerički upravljane mašine, alatke i njihovo programiranje*, Naučna knjiga, Beograd, 1987.
- [7] Troxler, F., Grundmann, M., Baumgartner, U.: *Flexible Fertigungszellen mit hoher Rentabilitaet*, Werkstatt und Betrieb, 122 (1989) 8
- [8] Schmidt, H., Erkes, K.F.: *Flexible Fertigung*, VDI Zeitschrift 129 (1987) 8, pp. 48-67.
- [9] Storn, H.: *Losgroessoptimierung bei flexiblen Fertigungszellen*, VDI Zeitschrift 128 (1986) 9
- [10] Benzinger, K., Kirchheim, A., Paluncic, Z.: *Bearbeitungszentren*, VDI Zeitschrift 128 (1986) 11
- [11] Fix-Sterz, J., Lay, G., Shulz-Wild, R.: *Flexible Fertigungssysteme und Fertigungszellen*, VDI Zeitschrift 128 (1986) 11, pp. 369-379.
- [12] Mikac, T.: *Istraživanja primjene senzora za trošenje i lom alata kod fleksibilnih obradnih sistema*, Magistarski rad, Tehnički fakultet Rijeka, Rijeka, 1991.
- [13] Ljubetić, J.: *Optimalizacija postupaka pri projektiranju višepredmetnih proizvodnih sistema*, Doktorska disertacija, Tehnički fakultet Rijeka, Rijeka, 1991.
- [14] Kuhn, H., Tempelmeier, H.: *Flexible Manufacturing Systems*. John Wiley & Sons, New York, 1993.
- [15] Talavage, J., Hannam, R.G.: *Flexible Manufacturing Systems in Practice-Applications, Design and Simulations*, New York (Marcel Dekker) 1988.
- [16] Kief, H.B.: *FFS – Handbuch: Einführung in Flexible Fertigungssysteme und deren Komponenten*, Carl Hanser Verlag, München, 1998.
- [17] Kief, H.B.: *NC/CNC – Handbuch 2000*, Carl Hanser Verlag, München, 1999.
- [18] Bidanda, B., Muralikrishnan, C.K.: *Flexible fixturing for intelligent manufacturing*, in: Kusiak, A. (Ed.), *Intelligent Design and Manufacturing: A Reference Book*, New York (Wiley), pp. 205-232, 1992.
- [19] Westkamper, E.: *“Inteligente” Werkzeugmaschinen für die Produktion 2000.*-VDI-Z, 135 (1993) 9, pp. 14-18
- [20] Murata, K., Tajima, M.: *Rationalisierung, Flexibilisierung und Mechatronisierung der Produktion in japanischen Unternehmen*, in Adam, D., Backhaus, K., Meffert, H. and Wagner, H. (Eds.), *Integration und Flexibilität - Eine Herausforderung für die Allgemeine Betriebswirtschaftslehre*, Wiesbaden (Gabler), pp. 249-268, 1990.

- [21] Sethi, A.K., Sethi, S.P.: *Flexibility in manufacturing*, A survey, in: IJFMS 2, pp.289-328, 1990.
- [22] Rao, P.P., Mohanty, R.P.: *Searching for definitions and boundaries in flexible manufacturing systems*, in: Production Planning & Control 2, pp.142-154, 1991.
- [23] Browne, J., Dubois, D., Rathmill, K., Sethi, S.P., Stecke, K.E.: *Classification of flexible manufacturing systems*, in: The FMS Magazine (1984) April, pp.114-117.
- [24] Kochikar, V.P., Narendran, T.T.: *A framework for assessing flexibility of manufacturing systems*, in: IJPR 30 (1992), pp. 2873-2895.
- [25] Köhler, R.: *Produktionsplanung für flexible Fertigungszellen*, Münster (Lit Verlag) 1988 (in German).
- [26] Erkes, K.F., Schonheit, M., Wieggershaus, U.: *Flexible Fertigung*, VDI Zeitschrift 130 (1988) 9, pp. 62-79.
- [27] Iwata, K.: *FMS in Japan*, Bulletin Japan Society of Precision Engineering, 18 (1984) 2, pp. 186-192.
- [28] Mertins, K.: *Entwicklungsstand flexible Fertigungssysteme in den USA*, ZWF, 76 (1981) 2, pp. 81-85.
- [29] Gottschalk, E.: *Inergrierte Fertigungen in der DDR*, VDI Zeitschrift 128 (1986) 11, pp. 381-388.
- [30] Schonheit, M., Wieggershaus, U.: *Flexible Fertigung*, VDI Zeitschrift 131 (1989) 9, pp. 49-64.
- [31] Shah, R.: *Flexible Fertigungssysteme in Europa*, Erfahrungen der Anwender, VDI Zeitschrift 129 (1987) 10, pp. 13-21.
- [32] Shah, R.: *Flexible Fertigungssysteme in Europa*, Erfahrungen der Anwender, VDI Zeitschrift 127 (1985) 17, pp. 639-648.
- [33] Shah, R.: *Flexible Fertigungssysteme in Europa*, Erfahrungen der Anwender, VDI Zeitschrift 133 (1991) 6, pp. 16-30.
- [34] Budenbender, W., Scheller, T.: *Flexible Fertigungssysteme in der Praxis*, VDI Zeitschrift 129 (1987) 10, pp. 22-28.
- [35] Meretz, H.: *Flexible Fertigungssysteme in der Praxis-eine unendliche geschichte*, Werkstatt und Betrieb, 122 (1989) 2, pp. 143-147.
- [36] Vits, R.: *Aufgabenangepasste fertigungsstrukturen*, VDI Zeitschrift 129 (1987) 8, pp. 63-69.
- [37] Cebalo, R.: *Nova filozofija proizvodnog strojarstva*, Zbornik radova Suvremeni trendovi proizvodnog strojarstva, Vol.2, Zagreb, 1992.
- [38] Ljubetić, J.: *Neki pogledi na primjenu fleksibilnih proizvodnih sustava*, Zbornik radova Tehničkog fakulteta Rijeka, Rijeka, 10 (1989), pp. 185-195.
- [39] Cebalo, R.: *Obradni sustavi – Fleksibilni obradni sustavi*, Zagreb, 2000.
- [40] Schonheit, M., Wieggershaus, U., Kiesewetter, S.A.: *Flexible Fertigung*, VDI Zeitschrift 132 (1990) 10, pp. 92-109.
- [41] Maschke: *Mit flexiblen Fertigungstechniken in die 90er Jahre*, Europa Seminar'88-Neuentwicklung eurpaeischer Werkzeugmaschinen und flexibler Fertigungssysteme: Computerintegrierte Fertigung – CIM – CAD - CAM Anwendungen, 1988.
- [42] Ljubetić, J.: *Analiza tendencija u razvoju primjene fleksibilnih proizvodnih sustava*, Strojarstvo, 34 (1992) 3/5, pp. 155-158

- [43] Hirt, K., Reineke, B., Sudkamp, J.: Einsatzbedingungen von flexiblen Fertigungssystemen, VDI Zeitschrift 133 (1991) 1, pp. 41-44.
- [44] Garey, M., Johnson, D., Sethi, R.: *The complexity of flowshop and jobshop scheduling*, Mathematics of Operations Research, vol. 1, pp. 117 – 129, 1976.
- [45] Van Laarhoven, P., Aarts, E., Lenstra, J.: *Job shop scheduling by simulated annealing*, Operations research, vol. 40, no. 1, pp. 113 – 125, 1992.
- [46] Dell' Amico, M., Trubian, M.: *Applying tabu search to the job shop scheduling problem*, Annals of Operations Research, vol. 40, pp. 231 – 252, 1993.
- [47] Adams, J., Balas, E., Zawack, D.: *The shifting bottleneck procedure for job shop scheduling*, - International Journal of Flexible Manufacturing Systems, vol. 34, no. 3, pp. 391 – 401, 1987.
- [48] Applegate, D., Cook, W.: *A computational study of the job shop scheduling problem*, ORSA Journal of Computing, vol. 3, no. 2, pp. 149 – 156, 1991.
- [49] Dauzere – Pers, S., Lasserre, J.: *A modified shifting bottleneck procedure for job – shop scheduling*, International Journal of Production Researches, vol. 31, pp. 923 – 932, 1993.
- [50] Domainko, D.: *Ekonomika i organizacija industrijskih poduzeća*, Zagreb: Narodne novine, 1970.
- [51] Burbidge, J. L.: *Production Planning*. – London: Heineman Ltd, 1971.
- [52] Selaković, M.: *Metode određivanja optimalnog proizvodnog procesa mehaničke obrade u ranoj fazi projektiranja*: Disertacija. – Rijeka, Tehnički fakultet Rijeka, 1976.
- [53] Fratassi, A. R.: *Manuale di organizzazione della produzione*. Milano: Etas, 1974.
- [54] Opitz, H., Wiendahl, H. P.: *Group Technology and Manufacturing Systems for Small and Medium Quantity Production* – International Journal of Production Research, 9 (1971) 1, pp. 181 – 203.
- [55] Selaković, M.: *Metode za određivanje optimalnog proizvodnog procesa mehaničke obrade dijelova*. - Strojarstvo, 20 (1978) 1, pp. 5-11.
- [56] Ljubetić, J.: *Definicija proizvodnog sistema u uvjetima srednjoserijskog proizvodnog programa*. – Zbornik radova međunarodnog savjetovanja o novim proizvodnim sistemima i tehnologiji, Opatija, pp. 263 – 271, 1987.
- [57] Selaković, M.: *Sinteza međuoperacijskog transporta u korelaciji redosljeda operacija i prostornog rasporeda opreme*, Zbornik radova 9. stručnog skupa o transportnim procesima u industriji, Beograd, pp. 9 – 16, 1986.
- [58] Ljubetić, J.: *Projekt proizvodnog sistema za obradu glavnih dijelova traktora*. – Rijeka : Tehnički fakultet Rijeka, 1986.
- [59] Selaković, M.: *Korelacija dijelova obrade i proizvodne opreme preko operacija*. – Tehnika, 34 (1984) 11, pp. 1399 – 1404.
- [60] Glover, F.: *Tabu search – Part I*, ORSA Journal on Computing, Vol.1, No.3, pp. 190 – 206, 1989.
- [61] Glover, F.: *Tabu search – Part II*, ORSA Journal on Computing, Vol.2, No.1, pp. 4 – 32, 1990.
- [62] Seo, Y., Egbelu, P.J.: *Process plan selection based on product mix and production volume*, International Journal of Production Research, Vol.34, No.9, pp. 2639 – 2655, 1996.

- [63] Ljubetić, J.: *Sequence proceeding of working orders and structure of FMS – Influence on Efficacy*, Bulletins for Applied Mathematics, Budapest BAM 1191/'96, pp. 125 – 134.
- [64] Ljubetić, J.: *Process plans selection in dynamic production environment*, Proceedings of the 9th DAAAM International Symposium, Katalinić, B. (Ed), Cluj – Napoca, pp. 275 – 276, 1998.
- [65] Ljubetić, J.: *Process plans selection for simultaneous production of different parts*, Proceedings of the 5th International Scientific – Technical Conference on Internal Combustion Engines and Motor Vehicles, Symposium, MOTAUTO 98, Volume II, Sofia – Vitosha, pp. 119 – 124, 1998.
- [66] Askin, R.G., Standridge, C.R.: *Modelling and Analysis of Manufacturing Systems*, John Wiley & Sons, New York, 1993.
- [67] Baker, K.: *Introduction to Sequencing and Scheduling*, John Wiley & Sons, New York, 1974.
- [68] Greenberg, H.: *A branch – and – bound to the general scheduling problem*, Operations Research, vol. 16 pp. 353 – 361, 1968.
- [69] Cheng, R., Gen, M., Tsujimura, Y.: *A tutorial survey of job – shop scheduling problems using genetic algorithms: part 1. representation*, International Journal of Computers and Industrial Engineering, vol. 30, no. 4, pp. 983 – 997, 1996.
- [70] Balas, E.: *Machine sequencing via disjunctive graphs – an implicit enumeration algorithm*, Operations Research, vol. 17, pp. 941 – 957, 1969.
- [71] Roy, B., Sussmann, B.: *Les problemes d'ordonnancement avec contraintes disjonctives*, Technical report 9, SEMA, note D. S. Paris, 1964.
- [72] Storer, R., Wu, S., Vaccari, R.: *New search spaces for sequencing problems with application to job shop scheduling*, Management Science, vol. 38, no. 10, pp. 1495 – 1510, 1992.
- [73] Giffler, B., Thompson, G.: *Algorithms for solving production scheduling problems*, Operations Research, vol. 8, no. 4, pp. 487 – 503, 1960.
- [74] Panwalkar, S., Iskander, W.: *A survey of scheduling rules*, Operations Research, vol. 25, pp. 45 – 61, 1977.
- [75] Haupt, R.: *A survey of priority – rule based scheduling problem*, OR Spektrum, vol. 11, pp. 3 – 16, 1989.
- [76] Blackstone, J., Phillips, D. and Hogg, G.: *A state – of – the – art survey of dispatching rules of manufacturing job shop operations*, International Journal of Production Research, vol. 20, pp. 26 – 45, 1982.
- [77] Morton, T., Pentico, D.: *Heuristic Scheduling Systems – With Applications to a Production Systems and Project Management*, John Wiley & Sons, New York, 1993.
- [78] Davis, L.: *Handbook of Genetic Algorithms*, Van Nostrand Reinhold, New York, 1991.
- [79] Goldberg, D.: *Genetic Algorithms in Search, Optimization and Machine Learning*, Addison – Wesley, Reading, MA, 1989.
- [80] Gen, M., Cheng, R.: *Genetic Algorithms and Engineering Design*, John Wiley & Sons, New York, 1997.
- [81] Holland, J.: *Adaptation in Natural and Artificial Systems*, University of Michigan Press, Ann Arbor, 1975.

- [82] Gen, M., Cheng, R.: *A survey of penalty techniques in genetic algorithms*, Proceedings of the Third IEEE Conference on Evolutionary Computation, Fogel, D. (Ed), IEEE Press, Nagoya, Japan, pp. 804 – 809, 1996.
- [83] Michalewicz, Z.: *A survey of constraint handling techniques in evolutionary computation methods*, Evolutionary Programming IV, McDonnell, J. et al. (Ed), MIT Press, Cambridge, MA, pp. 135 – 155, 1995..
- [84] Smith, A., Tate, D.: *Genetic optimization using a penalty function*, Proceedings of the 5th International Conference on genetic Algorithms, Forrest, S. (Ed), Morgan Kaufmann Publishers, San Mateo, CA, pp. 499 – 505, 1993.
- [85] Perinić, M., Ljubetić, J.: *A genetic algorithm for working orders sequencing*, Proceedings of the 8th International Scientific – Technical Conference on Internal Combustion Engines and Motor Vehicles, Symposium, MOTAUTO 01, Volume I, Varna, pp. 83 – 85, 2001.
- [86] Mikac, T., Perinić, M., Ljubetić, J.: *Applying genetic algorithm in activities of process plan selection*, Proceedings of the 12th DAAAM International Symposium, Katalinić, B. (Ed), Vienna, pp. 357 – 358, 2001.
- [87] Bagchi, S., Uckun, S., Miyabe, Y., Kawamura, K.: *Exploring problem – specific recombination operators for job shop scheduling*, Proceedings of the 4th International Conference on Genetic Algorithms, Belew, R., Booker, L. (Ed), Morgan Kaufmann Publishers, San Mateo, CA, pp. 10 – 17, 1991.
- [88] Croce, F., Tadei, R., Volta, G.: *A genetic algorithm for the job shop problem*, Computers and Operations Research, vol. 22, pp. 15 – 24, 1995.
- [89] Davis, L.: *Job shop scheduling with genetic algorithms*, Proceedings of the First International Conference on Genetic Algorithms, Grefenstette, J. (Ed), Lawrence Erlbaum Associates, Hillsdale, NJ, pp. 136 – 140, 1985.
- [90] Dorndorf, U., Pesch, E.: *Evolution based learning in a job shop scheduling environment*, Computers and Operations Research, vol. 22, pp. 25 – 40, 1995.
- [91] Fang, H., Ross, P., Corne, D.: *A promising genetic algorithm approach to job – shop scheduling, rescheduling, and open – shop scheduling problems*, Proceedings of the 5th International Conference on Genetic Algorithms, Forrest, S. (Ed), Morgan Kaufmann Publishers, San Mateo, CA, pp. 375 – 382, 1993.
- [92] Gen, M., Kobayashi, T.: *Proceedings of the 16th International Conference on Computers and Industrial Engineering*, Ashikaga, Japan, 1994.
- [93] Gen, M., Tsujimura, Y., Kubota, E.: *Solving job – shop scheduling problem using genetic algorithms*, Proceedings of the 16th International Conference on Computers and industrial Engineering, Gen, M., Kobayashi, T. (Ed), Ashikaga, Japan, pp. 576 – 579, 1994.
- [94] Holsapple, C., Jacob, V., Pakath, R., Zaveri, J.: *A genetics – based hybrid scheduler for generating static schedules in flexible manufacturing contexts*, IEEE Transactions on Systems, Man, and Cybernetics, vol. 23, pp. 953 – 971, 1993.
- [95] Nakano, R., Yamada, T.: *Conventional genetic algorithms for job – shop problems*, Proceedings of the 4th International Conference on Genetic Algorithms, Belew, R., Booker, L. (Ed), Morgan Kaufmann Publishers, San Mateo, CA, pp. 477 – 479, 1991.
- [96] Paredis, J.: *Exploiting constraints as background knowledge for genetic algorithms: a case study for scheduling*, Parallel Problem Solving from Nature:

- PPSN II, Männer, R., Manderick, B. (Ed), Elsevier Science Publishers, North-Holland, pp. 271 – 280, 1992.
- [97] Sakawa, M., Kato, K., Mori, T.: *Flexible scheduling in machining center through genetic algorithms*, Computers and Industrial Engineering, vol. 30, no. 4, pp. 931 – 940, 1996.
- [98] Shi, G., Lima, H., Sannomiya, N.: *A method for constructing genetic algorithm in job shop problems*, Proc. of 8th SICE Symposium Decentralized Autonomous System, pp. 175 – 178, 1996.
- [99] Yamada, T., Nakano, R.: *A genetic algorithm applicable to large – scale job – shop problems*, Parallel Problem Solving from Nature: PPSN II, Männer, R., Manderick, B. (Ed), Elsevier Science Publishers, North-Holland, pp. 281 – 290, 1992.
- [100] Tsujimura, Y., Gen, M.: *Genetic algorithms for solving multi – processor scheduling problems*, Proceedings of the First Asia – Pacific Conference on Simulated Evolution and Learning, Yao, X., Kim, J.H., Furuhashi, T. (Ed), Taejon, 1996.
- [101] Falkenauer, E., Bouffoix, S.: *A genetic algorithm for job – shop*, in Proceedings of the IEEE International Conference on Robotics and Automation, pp. 824 – 829, 1991.
- [102] Tamaki, H., Nishikawa, Y.: *A paralleled genetic algorithm based on a neighborhood model and its application to the job shop scheduling*, Parallel Problem Solving from Nature: PPSN II, Männer, R., Manderick, B. (Ed), Elsevier Science Publishers, North-Holland, pp. 573 – 582, 1992.
- [103] Bean, J.: *Genetic algorithms and random keys for sequencing and optimization*, ORSA Journal on Computing, vol. 6, no. 2, pp. 154 – 160, 1994.
- [104] Kelley, J.: *The critical path method: resource planning and scheduling*, Industrial Scheduling, Muth, J., Thompson, G. (Ed), Prentice Hall, Englewood Cliffs, NJ, pp. 347 – 365, 1963.
- [105] Norman, B., Bean, J.: *Random keys genetic algorithm for job – shop scheduling: unabridged version*, Technical report, University of Michigan, Ann Arbor, 1995.
- [106] Orvosh, D., Davis, L.: *Using a genetic algorithm to optimize problems with feasibility constraints*, Proceedings of the First IEEE Conference on Evolutionary Computation, Fogel, D. (Ed), IEEE Press, Orlando, FL, pp. 548 – 552, 1994.
- [107] De Jong, K.: *Genetic algorithms: a 25 year perspective*, Computational Intelligence: Imitating Life, Zurada, J. et al. (Ed), IEEE Press, New York, pp. 125 – 134, 1994.
- [108] Kubota, A.: *Study on Optimal Scheduling for Manufacturing System by Genetic Algorithms*, Master's thesis, Ashikaga Institute of Technology, Ashikaga, Japan, 1995.
- [109] Yamada, T., Nakano, R.: *Genetic algorithm and job – shop scheduling problem*, Systems, Control and Information, vol. 37, no. 8, pp. 484 – 489, 1993.
- [110] Goldberg, E.D.: *An Introduction to Genetic Algorithm Optimized for Portability and Parallelism System*, GARAGE Technical Report, Michigan State University, 1995.

POPIS OZNAKA I SKRAĆENICA

OZNAKE

C	- proizvoljno veliki pozitivni broj
c_i	- vrijeme izrade i transporta dijela i prema odabranom planu procesa u rješenju u
c_{jk}	- vrijeme izvršenja operacije j na stroju k
d_{ab}	- najkraći razmak između čvornih točaka a i b , koji treba prevail paleta s izratkom prije prve operacije
d_{rs}	- najkraći razmak između čvornih točaka r i s , koji treba prevaliti paleta s izratkom nakon izvršenja posljednje operacije
$f(u)$	- vrijednost funkcije vrednovanja
g_L	- gornja granica broja razvijenih planova procesa i -tog dijela
h_{ij}	- vrijeme transporta za q_i dijelove vrste i prema planu procesa p_{ij}
i	- i -ta vrsta dijelova određenog radnog naloga
j	- j -ti plan procesa
J	- veličina populacije
k	- k -ti stroj
M	- proizvoljno veliki pozitivni broj
m_{ij}	- vrijeme izrade za q_i dijelove vrste i prema planu procesa p_{ij}
nm	- broj strojeva FPS-a
np	- broj dijelova za simultanu izradu
npp_i	- ukupan broj razvijenih planova procesa i -tog dijela
p_{ij}	- j -ti plan procesa za dio vrste i
q	- skup količina dijelova radnog naloga za simultanu proizvodnju
q_{sj}	- količina j -tih izradaka u s -tom međuoperacijskom transportu poslije r -te operacije
r	- broj operacija za izradu j -tog izratka
s	- broj međuoperacijskih transporta
t_{cj}	- ciklus izrade j -tog izratka
t_i	- vrijeme trajanja i -tog događaja (aktivnosti ili zastoja)
t_{ij}	- vrijeme transporta od zadnje operacije do skladišta gotovih proizvoda pri izradi j -tog izratka
t_{ijk}	- vrijeme operacije na stroju k u planu procesa p_{ij} .
t_{jk}	- vrijeme trajanja operacije j na stroju k
T_k	- opterećenje stroja pri proizvodnji svih dijelova radnog naloga
t_{kj}	- ukupno vrijeme izvođenja svih radnih i kontrolnih operacija pri izradi j -tog izratka
t_{sj}	- vrijeme s -tog međuoperacijskog transporta pri izradi j -tog izratka
t_{rj}	- vrijeme r -te operacije pri izradi j -tog izratka
T_1	- opterećenje transportnog sustava pri proizvodnji svih dijelova radnog naloga prema odabranim planovima procesa
t_{ij}	- ukupno vrijeme svih transporta tijekom procesa izrade j -tog izratka

t_{uj}	- vrijeme transporta od skladišta repromaterijala do prve operacije pri izradi j -tog izratka
u	- skup planova procesa
v_p	- brzina palete
φ	- funkcija cilja
$\tilde{\phi}$	- skalirana prilagođenost
ϕ_s	- srednja prilagođenost
ϕ_{\max}	- maksimalna prilagođenost
$\tilde{\phi}_s$	- srednja skalirana prilagođenost
$\tilde{\phi}_{\max}$	- maksimalna skalirana prilagođenost
ϕ	- prilagođenost
ϕ_i	- prilagođenost i -te jedinice

SKRAĆENICE

CAPP	- računalom podržano projektiranje tehnoloških procesa
CAPPC	- računalom podržano planiranje i upravljanje proizvodnje
CAD	- računalom podržano konstruiranje
CAQ	- računalom podržana kontrola kvalitete
CNC	- numeričko upravljanje računalom
FPC	- fleksibilna proizvodna ćelija
FPS	- fleksibilni proizvodni sustav
FTL	- fleksibilna transfer linija
GA	- genetski algoritmi
JIT	- pravovremena (just in time) proizvodnja
JLS	- jednopredmetni linijski sustav
NC	- numeričko upravljanje
PP	- plan procesa
PPK	- pojedinačni proizvodni kapacitet
PS	- proizvodni sustav
PSS	- proizvodni sustav sličnog redoslijeda
PSV	- proizvodni sustav prema vrsti opreme
RN	- radni nalog
S	- stroj
SGP	- skladište gotovih proizvoda
SRM	- skladište repromaterijala
TRN	- terminiranje radnog naloga
TSP	- problem putujućeg trgovca
VLS	- višepredmetni linijski sustav

POPIS SLIKA

Sl. 2.1	<i>Primjenjivost pojedinih proizvodnih sustava u proizvodnji</i>	4
Sl. 2.2	<i>Područja primjene pojedinih fleksibilnih proizvodnih kapaciteta</i>	5
Sl. 2.3	<i>Fleksibilni proizvodni sustav za obradu limenih dijelova</i>	6
Sl. 2.4	<i>Fleksibilna proizvodna ćelija s linearnim skladištem paleta</i>	7
Sl. 2.5	<i>Fleksibilni proizvodni sustav s automatskim snabdijevanjem izradaka i alata posredstvom tračničkog transportnog vozila</i>	9
Sl. 2.6	<i>Fleksibilna transfer linija</i>	10
Sl. 2.7	<i>Opći model i karakteristike FPS-a</i>	11
Sl. 2.8	<i>Struktura i karakteristike FPS-a</i>	11
Sl. 2.9	<i>Podsustavi FPS-a</i>	12
Sl. 2.10	<i>Komponente tehničkog sustava FPS-a</i>	12
Sl. 2.11	<i>Komponente procesnog sustava FPS-a</i>	13
Sl. 2.12	<i>Sustav opskrbe radnim komadima</i>	14
Sl. 2.13	<i>Linijski, tračnički transportni sustav s paralelnim obostranim rasporedom strojeva</i>	15
Sl. 2.14	<i>Paletni optočni sustav s paralelnim rasporedom strojeva na obje strane transportne pruge</i>	15
Sl. 2.15	<i>Mrežni transportni sustav s automatski vođenim vozilima</i>	16
Sl. 2.16	<i>Tri različita principa izmjena</i> <i>a) Transportna kolica s dva paletna mjesta</i> <i>b) Transportna kolica s jednim paletnim mjestom</i> <i>c) Okretni paletni izmjenjivač na stroju</i>	17
Sl. 2.17	<i>Informacijski sustav</i>	18
Sl. 2.18	<i>Područje primjene i fleksibilnost konvencionalnih modela proizvodnog sustava i FPS-a</i>	20
Sl. 2.19	<i>Oblici fleksibilnosti jednog fleksibilnog proizvodnog sustava</i>	21
Sl. 2.20	<i>Smanjenje ciklusa obrade primjenom FPS-a</i>	24
Sl. 2.21	<i>Ukupan broj proizvedenih FPS-a u početnim godinama razvoja</i>	25
Sl. 2.22	<i>Veličina sustava u ovisnosti o broju radnih jedinica</i>	26
Sl. 2.23	<i>Struktura broja dijelova po jednom FPS-u</i>	26
Sl. 2.24	<i>Udio vrsta alatnih strojeva u izvedenim FPS-ima</i>	27
Sl. 2.25	<i>Struktura radnih naloga po jednom FPS-u</i>	28
Sl. 2.26	<i>Struktura primjene vrsta transportnih sustava</i>	28

SI. 3.1	<i>Alternativni planovi procesa za izradu dijela</i>	34
SI. 3.2	<i>Fleksibilni proizvodni sustav s 4 radne stanice i automatskim transportnim sustavom</i>	35
SI. 3.3	<i>Gantogram rada strojeva</i>	41
SI. 3.4	<i>Gantogram izradaka/dijelova</i>	41
SI. 3.5	<i>Venn-ov diagram odnosa među redosljedima</i>	42
SI. 3.6	<i>Prikaz disjunktog grafa</i>	45
SI. 4.1	<i>Shematski prikaz I. i II. faze upravljanja FPS-a</i>	51
SI. 4.2	<i>Opća struktura genetskih algoritama</i>	53
SI. 4.3	<i>Linearno skaliranje prilagođenosti</i>	56
SI. 4.4	<i>Križanje s jednom točkom reza</i>	61
SI. 4.5	<i>Križanje s parnim brojem točaka reza</i>	61
SI. 4.6	<i>Operacije radnog naloga. i pripadajući im strojevi:</i> <i>(a) za dio j_1,</i> <i>(b) za dio j_2,</i> <i>(c) za dio j_3</i>	65
SI. 4.7	<i>Redosljed izrade dijelova na stroju m_2; (prikazan osjenčano)</i>	65
SI. 4.8	<i>Redosljed izrade dijelova radnog naloga na bazi kromosoma [3 3 1 2 2 3 1 1 2]</i>	65
SI. 4.9	<i>Utjecaj lijevog pomaka u promjenjivom poluaktivnom redosljedu;</i> <i>(a) poluaktivni redosljed dobiven dekodiranjem kromosoma, i</i> <i>(b) aktivni redosljed nakon primjene lijevih pomaka na poluaktivni redosljed</i>	67
SI. 4.10	<i>Križanje: (a) zamjena parcijalnih redosljeda</i> <i>(b) pronalaženje gena manjka i gena viška</i> <i>(c) stvaranje legalnog potomka</i>	68
SI. 4.11	<i>Susjedni redosljedi</i>	69
SI. 5.1	<i>Terminiranje operacije s indeksom 1</i>	73
SI. 5.2	<i>Terminiranje operacije s indeksom 7</i>	73
SI. 5.3	<i>Terminiranje operacije s indeksom 2</i>	73
SI. 5.4	<i>Terminiranje operacije s indeksom 9</i>	74
SI. 5.5	<i>Terminiranje operacije s indeksom 8</i>	74
SI. 5.6	<i>Terminiranje operacije s indeksom 3</i>	74
SI. 5.7	<i>Terminiranje operacije s indeksom 4</i>	74
SI. 5.8	<i>Terminiranje operacije s indeksom 5</i>	75

Sl. 5.9	<i>Terminiranje operacije s indeksom 6</i>	75
Sl. 5.10	<i>Terminiranje operacija za dekodirani kromosom [1 10 2 11 14 12 15 3 4 5 6 7 13 8 9]</i>	79
Sl. 5.11	<i>Organizacijska struktura programa GJOB</i>	81
Sl. 5.12	<i>Dijagram tijeka programa GJOB</i>	82
Sl. 5.13	<i>“Options” izbornik programa GJOB (upisivanje ulaznih podataka)</i>	84
Sl. 5.14	<i>“Options” izbornik programa GJOB (upisivanje parametara GA)</i>	84
Sl. 5.15	<i>“Options” izbornik programa GJOB (odabir kriterija optimizacije)</i>	85
Sl. 5.16	<i>Prikaz rezultata programa GJOB</i>	85
Sl. 5.17	<i>Prikaz rezultata problema MT06 programom GJOB</i>	
	<i>a) rezultat</i>	
	<i>b) legenda</i>	
	<i>c) gantogram strojeva</i>	87
Sl. 5.18	<i>Prikaz rezultata problema MT10 programom GJOB</i>	
	<i>a) rezultat</i>	
	<i>b) legenda</i>	
	<i>c) gantogram strojeva</i>	90
Sl. 5.19	<i>Prikaz rezultata problema MT20 programom GJOB</i>	
	<i>a) rezultat</i>	
	<i>b) legenda</i>	
	<i>c) gantogram strojeva</i>	93
Sl. 5.20	<i>Usporedba rezultata za MT10 problem raznih heurističkih metoda dobivenih programskim paketom LEKIN</i>	94
Sl. 5.21	<i>Gantogram strojeva za problem MT10 dobiven programom LEKIN metodom simuliranog približavanja</i>	95
Sl. 6.1	<i>Prikaz rezultata zadatka za kriterij minimalnog ciklusa izrade – slučaj 1 programom GJOB</i>	
	<i>a) rezultat</i>	
	<i>b) legenda</i>	
	<i>c) gantogram strojeva</i>	105
Sl. 6.2	<i>Prikaz rezultata zadatka za kombinaciju kriterija – slučaj 2 programom GJOB</i>	
	<i>a) rezultat</i>	
	<i>b) gantogram strojeva</i>	
	<i>c) statistika strojeva</i>	111
Sl. 6.3	<i>Prikaz rezultata zadatka za kombinaciju kriterija – slučaj 3 programom GJOB</i>	
	<i>a) rezultat</i>	
	<i>b) gantogram strojeva</i>	
	<i>c) statistika strojeva i transpornog sustava</i>	119

Sl. 6.4	<i>Prikaz rezultata zadatka za kombinaciju kriterija – slučaj 4 programom GJOB</i>	
	<i>a) rezultat</i>	
	<i>b) gantogram strojeva</i>	
	<i>c) statistika rada strojeva i transportnog sustava</i>	126

POPIS TABLICA

Tab. 2.1	<i>Efekti primjene FPS-a</i>	25
Tab. 3.1	<i>Primjer problema tri izratka/dijela – tri stroja</i>	41
Tab. 3.2	<i>Lista pravila pri raspodjeli</i>	48
Tab. 4.1	<i>Neki radovi o implementaciji genetskih algoritmima na problem TRN-a</i>	63
Tab. 4.2	<i>Redoslijed dijelova radnog naloga po strojevima</i>	66
Tab. 5.1	<i>Primjer radnog naloga s 3 vrste dijelova s različitim brojem komada</i>	71
Tab. 5.2	<i>MT06 problem</i>	86
Tab. 5.3	<i>Parametri genetskog algoritma za MT06 problem</i>	87
Tab. 5.4	<i>MT10 problem</i>	88
Tab. 5.5	<i>Parametri genetskog algoritma za MT10 problem</i>	90
Tab. 5.6	<i>MT20 problem</i>	91
Tab. 5.7	<i>Parametri genetskog algoritma za MT20 problem</i>	93
Tab. 5.8	<i>Usporedba rezultata programa GJOB s rezultatima nekih heurističkih metoda</i>	94
Tab. 5.9	<i>Rezultati genetskih algoritama za tri referentna problema</i>	95
Tab. 5.10	<i>Parametri genetskog algoritma raznih autora za problem MT10</i>	96
Tab. 6.1	<i>Prikaz procedure rješavanja na primjeru</i>	98

ŽIVOTOPIS

Mladen Perinić rođen je 28. lipnja 1967. godine u Zadru, gdje završava osnovnu, a zatim i srednju školu matematičko - informatičkog usmjerenja s odličnim uspjehom. Nakon odsluženja vojnog roka 1987. godine upisuje se na sveučilišni studij strojarstva Tehničkog fakulteta Sveučilišta u Rijeci. Diplomirao je 06. srpnja 1992. godine s temom diplomskog rada *Numeričko modeliranje strujanja fluida u kaskadi* pod vodstvom mentora prof. dr. sc. Luke Sopte.

Tijekom studija bio je četiri puta nagrađivan za postignuti uspjeh, a akademske godine 1992/93. postao je i član sveučilišnog kluba "4+".

Krajem 1992. godine upisuje se na poslijediplomski studij *Računarska mehanika*, na modul *Strukturna analiza i optimizacija konstrukcija* na Tehničkom fakultetu Sveučilišta u Rijeci, te radi kao znanstveni novak na projektu *"Numeričko modeliranje strujanja fluida"* pod vodstvom prof. dr. sc. Zorana Mrše i prof. dr. sc. Luke Sopte, a od 07. studenog 1994. godine radi u stalnom radnom odnosu na Zavodu za tehničku mehaniku istog fakulteta kao asistent iz sljedećih predmeta: *Tehnička mehanika I* sveučilišnog studija strojarstva i brodogradnje, *Mehanika I* stručnog studija strojarstva i brodogradnje, *Mehanika i elementi konstrukcija* stručnog studija elektrotehnike, te od 1999. godine na Zavodu za proizvodno strojarstvo iz predmeta: *Osnove tehnoloških procesa*, *Projektiranje tehnoloških procesa*, *Tehnološki procesi montaže* sveučilišnog studija strojarstva, *Tehnološki procesi* stručnog studija strojarstva.

Kao istraživač bio je uključen na znanstvenom projektu *Strukturne analize objekata za optimalnu iskoristivost*, br 2-08-011, Ministarstva znanosti i tehnologije Republike Hrvatske, pod vodstvom glavnog istraživača prof. dr. sc. Josipa Brnića, a sada je uključen na znanstvenom projektu *Numerička optimizacija u projektiranju i proizvodnji*, čiji je glavni istraživač i voditelj također prof. dr. sc. Josip Brnić, a sam je projekt također financiran od Ministarstva znanosti i tehnologije Republike Hrvatske.

Magistarski je rad pod naslovom *Numeričko modeliranje potresnog opterećenja ravninskih konstrukcija* pod mentorstvom prof. dr. sc. Josipa Brnića uspješno obranio 19. lipnja 1997. godine. Fakultetsko vijeće Tehničkog fakulteta Sveučilišta u Rijeci, na svojoj 4. sjednici održanoj u veljači 2001. godine, odobrilo mu je temu doktorske disertacije pod naslovom *Optimizacija ciklusa izrade na fleksibilnom proizvodnom sustavu primjenom genetskih algoritama*.

Godine 1997. proveo je mjesec dana na usavršavanju na Fakulteta za strojništvo-Ljubljana, a 2002. godine mjesec dana na Institute for Production Engineering, Intelligent Manufacturing Systems, Vienna University of Technology.

Autor je ili koautor deset objavljenih znanstvenih radova. Govori i piše engleski. Oženjen je i otac dvoje djece.

PODACI O AUTORU I DOKTORSKOJ DISERTACIJI

1. AUTOR

Ime i prezime: Mladen Perinić

Datum i mjesto rođenja: 28. lipnja 1967. Zadar

Naziv fakulteta, studija i godina završetka dodiplomskog studija: Tehnički fakultet Sveučilišta u Rijeci, Strojstvo, 1992.

Naziv fakulteta, smjera i godina završetka poslijediplomskog studija: Tehnički fakultet Sveučilišta u Rijeci, Računarska mehanika, 1997.

Sadašnje zaposlenje: asistent

2. DOKTORSKA DISERTACIJA

Naslov: Optimizacija ciklusa izrade na fleksibilnom proizvodnom sustavu primjenom genetskih algoritama

Broj stranica, slika, tabela i bibliografskih podataka: 142, 68, 16, 110

Znanstveno polje i grana: Strojstvo, Proizvodno strojarstvo,

Voditelj rada: Red. prof. dr. sc. Juraj Ljubetić, dipl. ing.

Fakultet na kojem je rad obranjen: Tehnički fakultet Sveučilišta u Rijeci

3. OBRANA I OCJENA

Datum odobrenja teme: 22. veljače 2001.

Datum predaje rada: 01. travnja 2004.

Datum prihvatanja ocjene rada: 30. travnja 2004.

Sastav Povjerenstva za ocjenu: Red. prof. dr. sc. Tonči Mikac, dipl. ing.
Red. prof. dr. sc. Juraj Ljubetić, dipl. ing.
Red. prof. dr. sc. Jože Balič, dipl. ing.

Datum obrane: 15. srpnja 2004.

Sastav Povjerenstva za obranu: Red. prof. dr. sc. Tonči Mikac, dipl. ing.
Red. prof. dr. sc. Juraj Ljubetić, dipl. ing.
Red. prof. dr. sc. Jože Balič, dipl. ing.

Datum promocije:

OPTIMIZACIJA CIKLUSA IZRADE NA FLEKSIBILNOM PROIZVODNOM SUSTAVU PRIMJENOM GENETSKIH ALGORITAMA

Mladen Perinić

Sveučilište u Rijeci
Tehnički fakultet
Hrvatska

Ključne riječi: fleksibilni proizvodni sustav
ciklus izrade
genetski algoritam
simultana proizvodnja

Sažetak:

Cilj je rada omogućiti simultanu proizvodnju dijelova na FPS-u s različitim kombinacijama tipova dijelova i količina prema određenom radnom nalogu. Struktura radnog naloga po pitanju tipova dijelova i količina diktirana je tržišnim zahtjevima, tj unaprijed je zadana. Budući da struktura pojedinog radnog naloga nije u potpunosti usklađena s eksploatacijskim karakteristikama FPS-a koje proizlaze prvenstveno iz značajki strukture jezgre sustava, suočavamo se s problemom, kako u takvim uvjetima upravljati realiziranjem radnog naloga i pri tom postići glavni cilj: što manje vremensko zauzeće sustava odnosno što kraći ciklus izrade. U radu je prikazana metoda koja se temelji na dvije faze rješavanja problema upravljanja realiziranja radnog naloga. U prvoj fazi određen je izbor optimalne kombinacije planova procesa, a u drugoj fazi optimiran je redosljed ulaska pojedinih dijelova u proces izrade i izvođenja pojedinih operacija. Optimizacijski problem u obje faze upravljanja riješen je rabljenjem genetskog algoritma. Razvijen je računalni program za proračun i optimizaciju redosljednog odvijanja procesa na FPS-u.

Rad nije objavljen.

Mentor: Red. prof. dr. sc. Juraj Ljubetić, dipl. ing.

Povjerenstvo za ocjenu: Red. prof. dr. sc. Tonči Mikac, dipl. ing.
Red. prof. dr. sc. Juraj Ljubetić, dipl. ing.
Red. prof. dr. sc. Jože Balič, dipl. ing.

Povjerenstvo za obranu: Red. prof. dr. sc. Tonči Mikac, dipl. ing.
Red. prof. dr. sc. Juraj Ljubetić, dipl. ing.
Red. prof. dr. sc. Jože Balič, dipl. ing.

Datum obrane: 15. srpnja 2004.

Datum promocije:

Rad je pohranjen na Tehničkom fakultetu Sveučilišta u Rijeci.
(142, 68, 16, 110, hrvatski)

DD

UDK 658.512.001.26:658.513.4:004.421(043)

1. Optimizacija ciklusa izrade na fleksibilnom proizvodnom sustavu primjenom genetskih algoritama

I Mladen Perinić

II Sveučilište u Rijeci
Tehnički fakultet
HRVATSKA

Ključne riječi:

fleksibilni proizvodni sustav
ciklus izrade
genetski algoritam
simultana proizvodnja

Code: DD No.

UDC 658.512.001.26:658.513.4:004.421(043)

MACHINING CYCLE'S OPTIMIZATION ON FLEXIBLE MANUFACTURING SYSTEM APPLYING GENETIC ALGORITHMS

Mladen Perinić

University of Rijeka
Faculty of Engineering
Croatia

Key words: flexible manufacturing system
machining cycle
genetic algorithm
simultaneous production

Summary:

The aim of dissertation is to solve the problem of simultaneous production on the flexible manufacturing system with different combination of product types according to specified working order. Working order structure from the point of product types and quantities is dictated by market demands that are known earlier. Because the structure of particular working order is not harmonized with the exploitation characteristics of FMS, we are faced with problem how to realize working order in such conditions as well as how to achieve main goal: shorter machining cycle with less time occupation of production system. The method based on two phases for solving problem of control working order realization is presented in the dissertation. In the first phase the selection of optimal combination of process plans. In the second phase the order of part production and the order of particular operations processing is optimized. The optimization problem in both phases of control is solved by application of genetic algorithm approach. The software for computing and optimizing of processing order on FMS is developed.

This thesis has not been published.

Mentor: Prof. D.Sc. Juraj Ljubetić

Advisors: Prof. D.Sc. Tonči Mikac
Prof. D.Sc. Juraj Ljubetić
Prof. D.Sc. Jože Balič

Reviewers: Prof. D.Sc. Tonči Mikac
Prof. D.Sc. Juraj Ljubetić
Prof. D.Sc. Jože Balič

Presentation:

Degree conferred:

This thesis is deposited at the library of the University of Rijeka, Faculty of Engineering.
(142, 68, 16, 110, croatian language)