

SVEUČILIŠTE U ZAGREBU
FAKULTET ELEKTROTEHNIKE I RAČUNARSTVA

Klaudio Pap

**SIMULACIJA HIBRIDNIH I DIGITALNIH
SUSTAVA SA SUČELJIMA ZA OBRADU
SLIKOVNIH ELEMENATA I RASTERA**

DOKTORSKA DISERTACIJA

Zagreb, 2004.

Doktorska disertacija je izrađena na Katedri za računala na Grafičkom fakultetu.

Mentor: Dr. sc. Vilko Žiljak, red. prof. Grafičkog fakulteta, Zagreb

Doktorska disertacija ima: 189 stranica

Disertacija br.

Povjerenstvo za ocjenu doktorske disertacije:

1. Dr. sc. Mario Žagar, red. prof. FER-a, Zagreb
2. Dr. sc. Vilko Žiljak, red. prof. Grafičkog fakulteta, Zagreb
3. Dr. sc. Željka Mihajlović, doc. FER-a, Zagreb

Povjerenstvo za obranu doktorske disertacije:

1. Dr. sc. Mario Žagar, red. prof. FER-a, Zagreb
2. Dr. sc. Vilko Žiljak, red. prof. Grafičkog fakulteta, Zagreb
3. Dr. sc. Željka Mihajlović, doc. FER-a, Zagreb
4. Dr. sc. Danko Basch, doc. FER-a, Zagreb
5. Dr. sc. Darko Agić, doc. Grafičkog fakulteta, Zagreb

Datum obrane disertacije: 25. veljače 2004.g.

SADRŽAJ

	Strana
1. UVOD	7
2. CILJ ISTRAŽIVANJA	11
2.1. Nova rasterska rješenja	13
2.2. Metode rada i eksperimentalni okvir	14
3. DIGITALNI SUSTAVI SA SUČELJIMA ZA OBRADU SLIKOVNIH ELEMENATA I RASTERA	17
3.1 Digitalne ispisne tehnologije	18
3.1.1 Elektrofotografija	18
3.1.2. Ispis tekućim mlazom	25
3.1.3. Ionografija	29
3.1.4. Magnetografija	30
3.1.5. Elkografija	32
3.2 Sučelje za obradu slikovnih elemenata i rastera (Raster Image processor - RIP)	35
3.3 Osnovni postupci rastriranja	37
4. MODELIRANJE NOVIH PROGRAMSKIH MODELA ZA RASTRIRANJE	42
4.1. Matematički model rasterskog oblika	46
4.1.1. Novi rasterski modeli	47
4.1.2. Opis PostScript rutina radi testiranja novih rasterskih modela	52
4.2. Implementacija u sučeljima za obradu slikovnih elemenata i rastera kao stohastičko rastriranje	55
4.2.1. Eksperimentalno istraživanje	56
4.2.2. Opis PostScript rutina radi vođenja eksperimenata	62

5. TEZA RAZVOJA PREMA MODULARNIM HIBRIDNIM SUSTAVIMA	73
5.1. Razlozi uvođenja modularnih hibridnih sustava.....	74
5.2. Konfiguracije hibridnih linija	77
6. MODEL HIBRIDNOG SUSTAVA ZA INDIVIDUALIZACIJU S NOVIM RASTERSKIM PROCEDURAMA	80
6.1. Model CTP sustava	82
6.1.1. Gradnja programskog simulacijskog CTP modela	86
6.1.2. Opis rutina i veza radi vođenja eksperimenata	90
6.1.3. Eksperimentiranje s CTP modelom.....	95
6.2. Model analogno digitalnog sustava za izradu mastera	99
6.2.1. GPSS programsko rješenje sustava	103
6.2.2. Opis rutina i veza radi vođenja eksperimenata	106
6.2.3. Eksperimentiranje s modelom	113
6.3. Model modularne individualizacije s novim rasterskim procedurama	115
6.3.1. Rješenje modela u GPSS-u	119
6.3.2. Opis rutina i veza radi vođenja eksperimenata	122
6.3.3. Eksperimentiranje s modelom	134
7. MODEL DIGITALNOG SUSTAVA ZA INDIVIDUALIZACIJU S NOVIM RASTERSKIM PROCEDURAMA	139
7.1 Model digitalnog sustava s jednim modulom	139
7.1.1. Rješenje modela u GPSS-u	141
7.1.2. Opis rutina i veza radi vođenja eksperimenata	143
7.1.3. Eksperimentiranje s modelom	149

7.2 Model digitalnog sustava s dva modula	151
7.2.1. Rješenje modela u GPSS-u	152
7.2.2. Opis rutina i veza radi vođenja eksperimenata	141
7.2.3. Eksperimentiranje s modelom	161
8. ZAKLJUČAK	163
9. LITERATURA	168
SUMMARY	179
ŽIVOTOPIS	183

1. Uvod

Digitalni sustavi sa sučeljima za obradu slikovnih elemenata i rastera su konfiguracije namjenjene različitim ispisnim tehnologijama. Ta sučelja (Raster Image Processor) [26, str. 3.69-3.71][53, str. 216-219] su sama za sebe računala čije arhitekture, procesne snage, kapaciteti i vrste memorijskih resursa ovise o kvaliteti, veličini i brzini željenog izlaza te o željenoj brzini i organizaciji prijema digitalnih zapisa. Ona su osnova današnjeg digitalnog tiskarstva koje je novijeg datuma i snažnog razvoja. Mnoga programska i sklopovska rješenja su u stalnim dopunama i testiranjima sa svrhom povećanja produktivnosti, dijagnosticiranja kvarova u zastoju, povećavanja protoka u komunikacijskoj mreži i racionalnog korištenja energije.

Optimalno upravljanje računarskim sustavom za obradu slike i teksta u vrijeme velikih promjena, modernizacije, uvođenja digitalnih tiskarskih tehnika, istražuje se metodama matematičkog modeliranja i simuliranja. U ovom radu predlaže se postupak modeliranja od razine konceptijskih modela do njihove realizacije kao programa za računalnu simulaciju. Konvencionalni, digitalni i hibridni tiskarski resursi planiraju se za specijalne zadatke kao linijska procedura ili kao paralelna konfiguracija. U linijskoj proceduri se često pojavljuju zastoji zbog neprilagođenosti brzina, a paralelna traži veća planiranja i poskupljuje izvedbu [1, str. ii107-ii122][2][3]. Paralelno konfiguriranje omogućava konkurentno preuzimanje poslova iz različitih područja i različitih naklada. Pokazuje se da je donošenje odluka moguće jedino iscrpnim planiranjem na bazi rezultata dobivenih simulacijom digitalnih tiskarskih sustava kao stohastičkih modela.

Riječ *sustav* upotrebljava se zato da bi se naglasila međusobna povezanost elemenata i njihova povezanost s okolinom [4]. Riječ *model* naglašava da se ne

proučava objekt interesa već samo njegova reprezentacija [4]. Fraza "modeliranje i simulacija" opisuje složene aktivnosti u svezi konstrukcije modela iz realnog svijeta i njegova simulacija na računalu [5]. Model sadržava pojednostavljenu analogiju procesa iz realnog svijeta. Model se lakše razumije; on može sadržavati uzroke i efekte procesa; on nam omogućava eksperimentalne manipulacije [6]. Ako je nekakav proces istinski stohastičan, ne postoji poznato objašnjenje za njegovu stohastiku. Ponekad, kada je za potpuno opisivanje nekog procesa potrebno previše detalja ili ima previše teškoća, proces se predstavlja kao stohastički [7].

Nakon uvodnih razmatranja postavlja se u drugom poglavlju cilj i predmet istraživanja. U njemu je definiran izvorni znanstveni doprinos ove disertacije.

U trećem poglavlju opisani su današnji digitalni sustavi sa sučeljima za obradu slikovnih elemenata i rastera. Definiran je odnos slikovnog elementa i pridruženog binarnog zapisa ovisno o ispisnoj tehnologiji. Navedeni su razlozi uvođenja programskog jezika PostScript [25, str.4][40][54] kao vodećeg jezika za opis grafičke stranice u sučeljima za obradu slikovnih elemenata i rastera. Uspoređene su današnje i nadolazeće digitalne ispisne tehnologije kao što je suha i tekuća elektrofotografija, digitalni i binarni kontinuirani ispis tekućim mlazom, piezoelektrični "kapljica na zahtjev" ispis, termalni ispis mlazom, ionografija, magnetografija i elkografija. Istražene su osnovne funkcije i faze rada sučelja za obradu slikovnih elemenata i rastera kao što je faza procesiranja i faza rastriranja. Također su navedeni današnji osnovni postupci rastriranja.

Rasterske ćelije [50, str.22-26] koje se danas upotrebljavaju u sučeljima za obradu slikovnih elemenata i rastera svode se na nekoliko modela. U četvrtom poglavlju se izvodi matematičko 3D i 2D modeliranje originalnih novih rastera kao i pripadni PostScript programski kod. Originalna implementacija u sučeljima za obradu slikovnih elemenata i rastera ostvarena je kao stohastičko rastriranje

(4.2.). Za eksperimentiranje su predloženi različiti algoritmi rastriranja. Time se upućuje na nove moguće pravce razvoja rastriranja u sučeljima za obradu slikovnih elemenata i rastera. Opis svih PostScript rutina za testiranje novih rasterskih modela i vođenja eksperimenata prikazan je u poglavljima 4.1.2. i 4.2.2. .

U petom poglavlju se istražuje prijedlog razvoja digitalnih sustava sa sučeljima za obradu slikovnih elemenata i rastera prema modularnim hibridnim sustavima. Osim same definicije modularnog hibridnog sustava i navođenja razloga njegovog uvođenja, iznose se i problemi s kojima se autor suočavao proteklih godina projektiranja takvih sustava u praksi. Prikazane su konfiguracije hibridnih linija u vidu šest samostalnih modela koji pokrivaju cjelovito digitalno ustrojstvo tiskarskih podkonfiguracija.

Dvije moguće koncepcije rješavanja istog sustava za individualizaciju s novim rasterskim procedurama prikazuju se i uspoređuju u šestom i sedmom poglavlju. Zadatak individualizacije s novim rasterskim procedurama poslužio je za sučeljavanje i ocjenjivanje hibridnog sustava nasuprot čistog digitalnog sustava. Stvaranje složenog hibridnog modela za taj zadatak prikazuje se u šestom poglavlju. Takav originalan model omogućuje optimalno projektiranje i reprogramiranje dijelova hibridnog sustava u kratkom vremenu. Pokazat će se da se zbog velike dinamike promjene ciljeva i rokova jedino modeliranjem i simuliranjem mogu donositi optimalna rješenja.

Dat je originalni programski model sustava (6.1.) koji omogućuje direktno digitalno fotoosvjetljavanje ofsetne ploče iz računala nazvano Computer to Plate (CTP) [26, str.4.5][53, str.222-224]. Mikrosimulacijama pojedinih modula CTP sustava vrši se dublje istraživanje da bi se optimirao ciljani modul sustava. Ovdje se prikazuje simulacija sučelja za obradu slikovnih elemenata i rastera kao dio CTP sustava.

Priloženo je programsko rješenje u jeziku GPSS [1, str. ii-1][52, str.ii9] (6.1.1.) čije su sve rutine i veze opisane radi vođenja eksperimenata u poglavlju 6.1.2.. Eksperimentalni rezultati postignuti s njim su prikazani u poglavlju 6.1.3..

Rezultati simulacije CTP sustava se ugrađuju u novi programski model analogno digitalnog sustava za izradu mastera (6.2.). Master je jednoznačna podloga koja se kasnije individualizira. Dat je eksperimentalni plan te rezultati simulacije (6.2.3.) s programskim GPSS modelom (6.2.1.) analogno digitalnog sustava za izradu mastera. Sve instrukcije i blokovi modela opisane su u poglavlju 6.2.2..

Model modularne individualizacije (6.3.) se naslanja na prethodni model čineći s njim hibridnu konfiguraciju. Gradnja tog modela za individualizaciju mastera kao i rješenje modela u GPSS jeziku (6.3.1.) bazirano je na mjerenjima fizičkog sustava. Eksperimentiranje s modelom (6.3.3.) se prikazuje uspoređivanjem dviju hipotetičnih konfiguracija, a za potrebe budućih eksperimentiranja programski model je detaljno opisan u poglavlju 6.3.2..

U sedmom poglavlju gradi se model digitalnog sustava za individualizaciju s novim rasterskim procedurama i to model s jednim digitalnim modulom (7.1.) i model s dva modula (7.2.). Za oba slučaja dani su programski GPSS modeli (7.1.1., 7.2.1.) na kojima je izvršeno eksperimentiranje (7.1.3., 7.2.3.). Na kraju svakog eksperimenta grafički se uspoređuju rezultati s rezultatima hibridnog modularnog modela, a opis svih programskih rutina i veza nalazi se u poglavljima 7.1.2. i 7.2.2..

2. Cilj istraživanja

Cilj istraživanja je dokazati da se s predloženim modelima postižu znatna poboljšanja vođenja protoka podataka u digitalnim grafičkim sustavima. Predložit će se novi modeli u prikazivanju i rješavanju veoma složenih zadataka varijabilne obrade slikovnih elemenata kojima bi se moglo unaprijediti i olakšati projektiranje složenih digitalnih grafičkih sustava za specijalne zadatke. Do sada su se metode modeliranja i simulacije malo koristile u upravljanju takvih sustava pa se ovim radom ujedno proširuje područje istraživanja istih te otvaraju nova poglavlja izučavanja uskih grla protoka podataka u digitalnoj grafici.

Zna se da se zadaci varijabilne obrade slikovnih elemenata trebaju rješavati određenim tehnologijama digitalnog ispisa, ali se ne zna kako ih optimalno koristiti ili ih možda hibridno povezivati s konvencionalnim metodama tiska. Postojeća mjerenja na realnim sustavima trebaju poslužiti za izradu stohastičkog modela koji će u programskoj izvedbi omogućiti eksperimentiranje s hipotetskim konfiguracijama i resursima.

Digitalni zapisi, koji se šalju prema sučeljima za obradu slikovnih elementa i rastera, su programi koji mogu sami generirati podatke za ispis ili ih eksplicitno sadrže. Prijem digitalnih podataka, iz tržišnih programskih paketa, zahtjeva konverziju u programski zapis koji jedino razumiju izlazni uređaji. Postoji ih nekoliko zasnovanih na posve različitim tehnologijama kao što je PDF [51, str. 247][57][58] na primjer. Za varijabilne podatke to je značajno usporavanje u protoku do ispisa.

Direktno pisanje programa za posebne zadatke iziskuje programiranje u specijalističkim jezicima kao što je PostScript [41][42][43][44][45][46][47][49][54]. Mogućnosti tržišnih programskih paketa i pripadnih upravljačkih programa za izlazni uređaj nisu mogle odgovoriti na nove postavljene ispisne brzine i rasterske algoritme.

Na temelju dugogodišnjeg praćenja i programiranja različitih zadataka na digitalnim i hibridnim konfiguracijama, postavlja se teza da se digitalni sustavi sa sučeljima za obradu slikovnih elemenata i rastera trebaju razvijati prema modularnim hibridnim sustavima. Tu tezu razvoja s potpuno novim neistraženim metodama rada i rasterskih algoritama treba potkrijepiti s razlozima uvođenja kao i mogućim konkretnim problemima.

Postupak istraživanja i dokaza teze provodi se metodom simulacije. Za različite zadatke očekuju se različite konfiguracije s pripadnim programskim rješenjima bez i u kombinaciji s drugim tehnikama tiska odnosno modularni digitalni i hibridni sustavi. Za njihovo optimalno projektiranje razvit će se metode ocjenjivanja ispravnosti konfiguriranja što će se prikazati stohastičkim modelom simulacije na računalu.

Da bi se pristupilo modeliranju treba kvantitativno i kvalitativno usporediti različite vrste digitalnih ispisnih tehnologija i pripadnih sučelja. Ući će se u unutarnju strukturu načina ispisivanja slikovnih elemenata. Treba analizirati njihove limite i mogućnosti primjene.

Sučelje za obradu slikovnih elemenata i rastera (Raster Image Processor - RIP)[55][56] prevodi PostScript naredbe u binarni zapis za slikovne elemente digitalnog ispisnog uređaja. Definirale su se faze i načini rada tog uređaja.

Izvorni znanstveni doprinos disertacije očekuje se u slijedećem:

- prijedlog novih rasterskih ćelija za digitalnu transformaciju slikovnog elementa te njihova implementacija u sučeljima za obradu slikovnih elemenata i rastera
- prijedlog novog stohastičkog modela, baziranog na snimanju i mjerenju protoka podataka realnog sustava, koji služi kao alat za daljnje istraživanje

- prijedlog razvoja digitalnih sustava sa sučeljima za obradu slikovnih elemenata i rastera prema modularnim hibridnim konfiguracijama u svrhu poboljšanja rješavanja novih zadataka
- razvoj metoda za ocjenjivanje i odabiranje optimalnih konfiguracija iz eksperimentalnih rezultata
- prijedlog optimalnog modularnog korištenja sučelja za obradu slikovnih elemenata i rastera i pripadnih programskih rješenja za posebne zadatke varijabilnog programiranja

2.1. Nova rasterska rješenja

Sadržaj ispisa digitalnog tiska je definiran memorijskim sadržajem i rasterskim modelom. Takvi modeli su neistraženi, a ovim radom bit će ponuđena nova PostScript rasterska rješenja koja se ne mogu primjeniti u tradicionalnoj tehnici. Nova PostScript rasterska rješenja bit će izvedena i testirana na digitalnim tiskarskim uređajima.

Istražit će se efikasnost nove predložene metodologije kreiranja novih modela rastriranja jer postojeći načini su spori i skupi. To područje je slabo istraženo jer su potrebna velika sredstva za izradu analogne realne podloge za mjerenje i ocjenjivanje.

Pokazat će se da se modeliranjem i simuliranjem rasterskih rješenja može potaknuti snažan razvoj u postojećim sučeljima za obradu slikovnih elemenata i rastera. Ponudit će se pristup izrade matematičkog modela novih rasterskih oblika i to preko matematičkog 3D i 2D modeliranja i njihova implementacija preko PostScript programskog koda.

Razvijen je PostScript program za testiranje novih rasterskih elemenata. Ponudit će se implementacija u sučeljima za obradu slikovnih elemenata i rastera kao stohastičko rastriranje. Izvest će se eksperimentiranja s cikličkim slijedom algoritama rastriranja, sa stohastičkim odabirom algoritma rasterskog elementa i rastriranje sa stohastičkim odabirom algoritma rasterskog elementa, linijature i kuta rastriranja.

Tako složene eksperimente treba programski primjeniti na istoj slici s uvećanim slikovnim elementima radi lakšeg individualnog prepoznavanja primjenjenog rasterskog elementa, a s udaljavanjem gledanja slike može se ocjenjivati uspješnost primjene.

2.2. Metode rada i eksperimentalni okvir

Eksperimentalni rezultati osnova su razrade modela stohastičke simulacije i traženje poboljšanja programa i sklopovlja. Eksperimentiranja s takvim uređajima su preskupa pa je izabrana simulacija istih kao metoda utvrđivanja uskih grla protoka informacija, provjere optimalnih konfiguracija i programskih rješenja.

Rezultati simulacijskog eksperimentiranja bi trebali dati ocjenu i svrsishodnost svakog modela. To bi prikazalo način kako se za svaki zadatak može modeliranjem i simuliranjem pravilno i optimalno izvršiti projektiranje. Takvi modeli bi bili radni modeli s kojima bi se eksperimentiralo svaki puta prilikom promjene zadatka individualizacije.

Izvest će se model i simulacijsko eksperimentiranje kao primjer mikro i makro simulacije stohastičkog sustava koje omogućuje direktno fotoosvjetljavanje ofsetne ploče iz računala (Computer to Plate - CTP). Ovaj model će ujedno prikazati sve faze u sučelju za obradu slikovnih elemenata i rastera. CTP sustav

se sastoji od tri modula: modul za impoziciju i procesiranje, modul za bitmapiranje, modul za osvjetljavanje i razvijanje. Modul za impoziciju i procesiranje i modul za bitmapiranje izvode operacije sučelja za obradu slikovnih elemenata i rastera.

Da bi se ostvario postavljen cilj istraživanja treba izvršiti mjerenja vremena procesiranja i impozicije tipičnih PostScript zapisa mastera kao i mjerenja bitmapiranja za određeni format tiskovne forme i vremena osvjetljavanja i razvijanja za definiranje stohastičnih varijabli koje bi se ugradile u model. Gradnja programskog simulacijskog CTP modela omogućiti će eksperimentiranje za traženje optimalnog broja modula za bitmapiranje kao primjer optimalnog odabira bilo kojeg modula sustava. Kao faktor odabira i ocjene bit će parametar prosječnog čekanja u redu i iskorištenje o broju poslova u vremenu. Dat će se isplativost prelaska s jednog modula na dva.

Napravljen je poseban model za izradu mastera offsetnom tehnologijom koji se može koristiti i proučavati potpuno autonomno ili kao dio većih konfiguracija. Izvršit će se mjerenja vremena pripreme peterbojnog ofsetnog stroja i njegove brzine ispisa kao i mjerenja vremena sušenja obostranog ofsetnog tiska i vremena dorade određene naklade tiska. Iz tih mjerenja će se definirati stohastičke varijable koje će se ugraditi u programski GPSS model. Eksperimentiranje s modelom ima za cilj određivanje prosječnog vremena izvršavanja sustava za izradu mastera u odnosu na različite zadane naklade.

Kao primjer individualizacije mastera s jednom digitalnom ispisnom tehnologijom uzet će se elektrofotografija unutar modularnog modela za individualizaciju. To je model gdje se preko servera distribuiraju PostScript zapisi prema modulu sa sučeljem za obradu slikovnih elemenata i rastera preko kojeg se vrši ispis na elektrofotografsku ispisnu jedinicu.

Definirat će se distribucije kvarova RIP modula, distribucije vremena rješavanja kvarova i vrijeme kontrole tiska. Sva mjerenja koristit će se za izradu stohastičkog programskog modela. U eksperimentalnom planu će se suprotstaviti dvije konfiguracije s različitim RIP modulima koji se međusobno razlikuju po brzini, cijeni i kvaliteti. Rezultati simulacijskog eksperimentiranja će se analizirati preko iskorištenja RIP modula i potrebnog vremena za izvršenje potrebne naklade.

Nasuprot prethodno opisanim modelima koji zajedno čine hibridnu konfiguraciju, izgradit će se model čistog digitalnog sustava s kojim bi se tiskao istovremeno i master i varijabla u jednom prolazu. U model ulaze dvije vrste PostScript zapisa. Jedan koji definira pozadinu (master) prednje i stražnje strane i drugi koji stvara varijabilne informacije.

Iz realnog sustava uzet će se rezultati mjerenja vremena rješavanja kvarova, distribucija pojavljivanja kvarova kao i potrebno vrijeme zamjene role jer se radi o stroju koji tiska iz role papira. Iz tih mjernih podataka će se stvoriti stohastičke varijable potrebne za gradnju programskog modela. Eksperimentiranje s programskim GPSS modelom će se voditi s ciljem određivanja vremena trajanja individualizacije s digitalnim modelom i istovremeno usporediti s vremenima proizašlim iz eksperimentiranja s hibridnim modularnim modelom.

Izgradit će se novi model s dva digitalna modula gdje će server distribuirati PostScript varijabilne zapise na oba modula. S tim novim GPSS modelom će se provesti sva eksperimentiranja kao i s jednim modulom i usporediti ga s rezultatima hibridnog modela. Na taj način će se vidjeti veličina poboljšanja izvedbe posla u odnosu na prethodni model kao i prostor moguće primjene.

3. Digitalni sustavi sa sučeljima za obradu slikovnih elemenata i rastera

Digitalni sustavi sa sučeljima za obradu slikovnih elemenata i rastera su konfiguracije različitih digitalnih ispisnih tehnologija i pripadnih sučelja. Današnje digitalne ispisne tehnologije su rasterske ispisne tehnologije. Karakteristika rasterske tehnologije je što se otisnuta ili prikazana ispisna slika sastoji od polja slikovnih elemenata koji se mogu pojedinačno adresirati pa tako i obrađivati. Svakom slikovnom elementu je pridružen binarni zapis koji određuje način prikaza slikovnog elementa. Zbog toga što se ispisi vrše iz binarnog zapisa slikovnog elementa, takvi sustavi se također nazivaju sustavi digitalnog tiska.

Ovisno o vrsti ispisne tehnologije binarni zapis slikovnog elementa može biti:

- jednobitan slikovni element (boja je ili nije prisutna)
- dvobitan, četverobitan, šesterobitan, osmerobitan, dvadesetčetverobitan i tridesetdvobitan za diskretne sive nijanse odnosno nijanse boje slikovnog elementa (digitalni tisak Xeikon [59], digitalni tisak Indigo [60], digitalni tisak Xerox Docucolor [61], Elco 400 [62])

Danas je potpuno saživio koncept nezavisnosti izlaznih uređaja od ulaznih programa i sklopovlja. To je ostvareno uvođenjem programskog jezika PostScript s kojim se opisuje grafička stranica koja se šalje, ne više kao grafički digitalni zapis, već kao niz PostScript naredbi koje se unutar sučelja za obradu slikovnih elemenata i rastera prevode u informacije koje razumije digitalni ispisni uređaj. Uređaj koji prevodi PostScript naredbe u binarni zapis za slikovne elemente digitalnog ispisnog uređaja je sučelje za obradu slikovnih elemenata i rastera (Raster Image Processor - RIP) . Raster je nakupina slikovnih elemenata koje za ljudsko oko čini sivu razinu u monokromatskim odnosno nijansu boje u kolor digitalnim ispisnim sustavima.

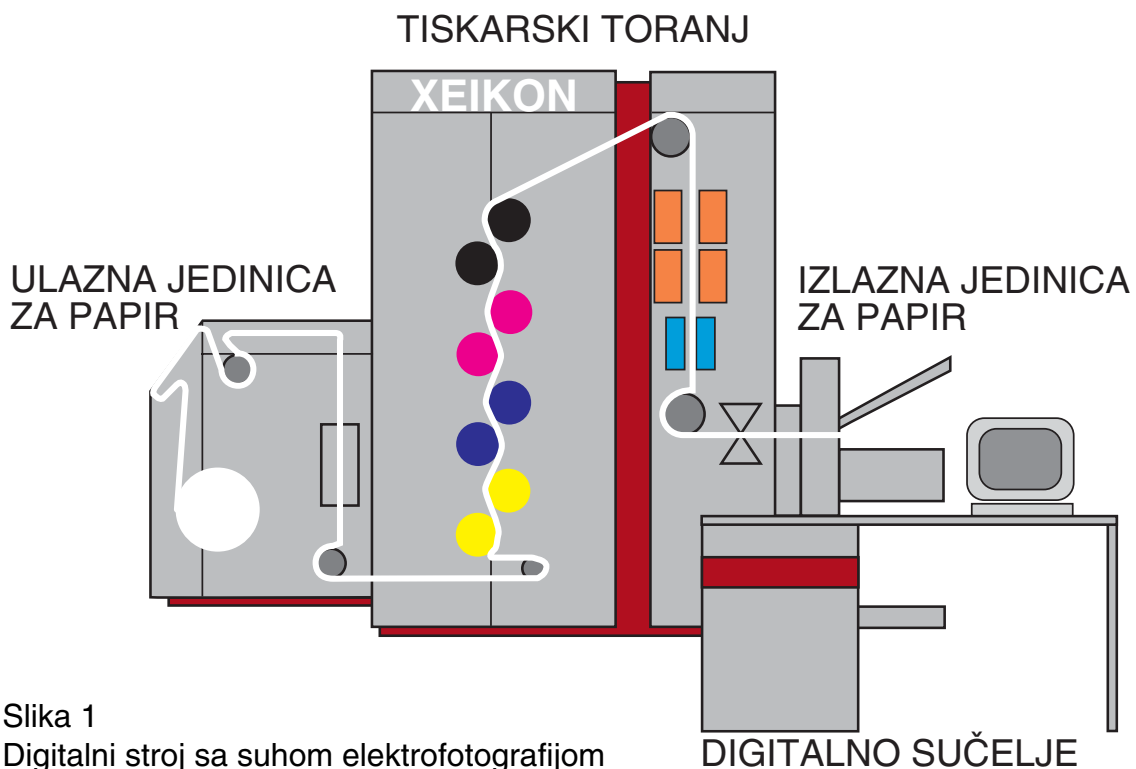
Prvi PostScript digitalni sustavi su se pojavili 1985-te godine od firme Adobe Systems. Danas je to standard za programski opis grafičke stranice za sve digitalne izlazne tehnologije i aktualna je verzija PostScript Level 3 [54].

3.1 Digitalne ispisne tehnologije

Najraširenije su tehnologije bazirane na toneru i tekućoj elektrografskoj boji. Zastupljeni su principi kao elektrofotografija, ispis tekućim mlazom, ionografija, magnetografija i nadolazeća elkografija.

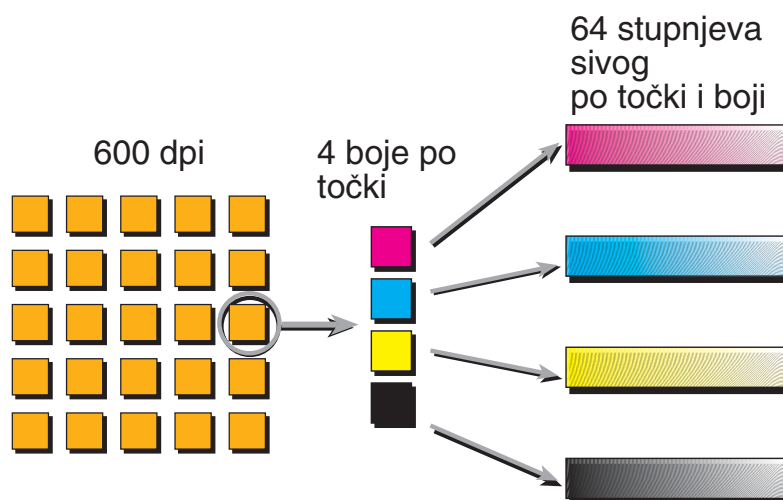
3.1.1 Elektrofotografija

Digitalni tiskarski uređaji bazirani na elektrofotografiji koriste suhi, ali i tekući toner kao boju pa se elektrofotografija dijeli na suhu i tekuću elektrofotografiju [9][10][11][12][13][14][15][16][26, str. 4.37-4.39][53, str. 328-330]. Suha elektrografija je najraširenija dok je tekuća tek u razvoju.



Slika 1
Digitalni stroj sa suhom elektrofotografijom

Na slici 1 prikazan je uređaj koji koristi suhu elektrofotografiju [37][38][39], a omogućuje reprodukciju četverbojnog obostranog tiska, zaobilazeći međufaze standardnih postupaka, kao što je izrada filma, pokusnog otiska i tiskovnih ploča koje su nezaobilazne u konvencionalnom ofsetnom tisku. Sastoji se od 8 elektrofotografskih jedinica kao tiskovnih uređaja, četiri za tisak prednje strane i četiri za tisak stražnje strane.

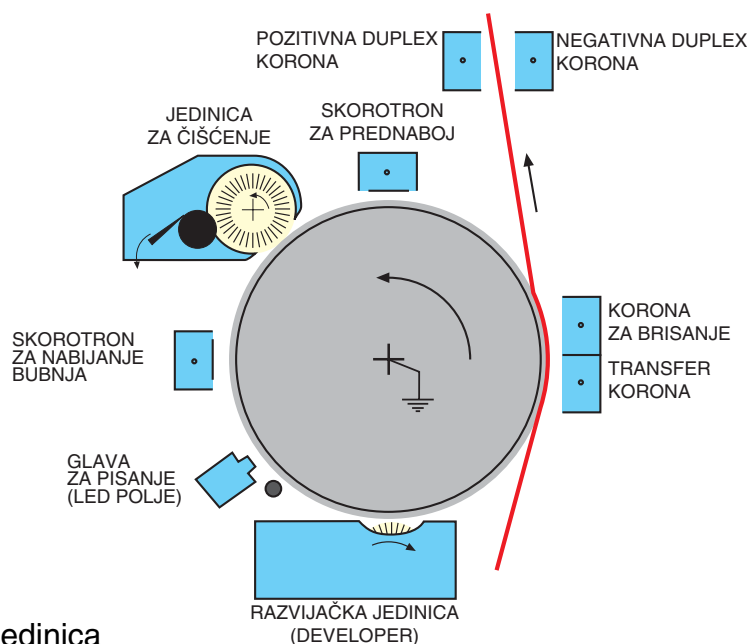


Slika 2
Raster tehnologija

Tisak se vrši direktno sa računala iz programa koji mogu stvoriti PostScript zapis [8]. Rasterizacija slike se odvija raster tehnologijom na rezoluciji od 600 točkica po inču sa 64 stupnja sive skale po točkici za svaku boju posebno (Slika 2).

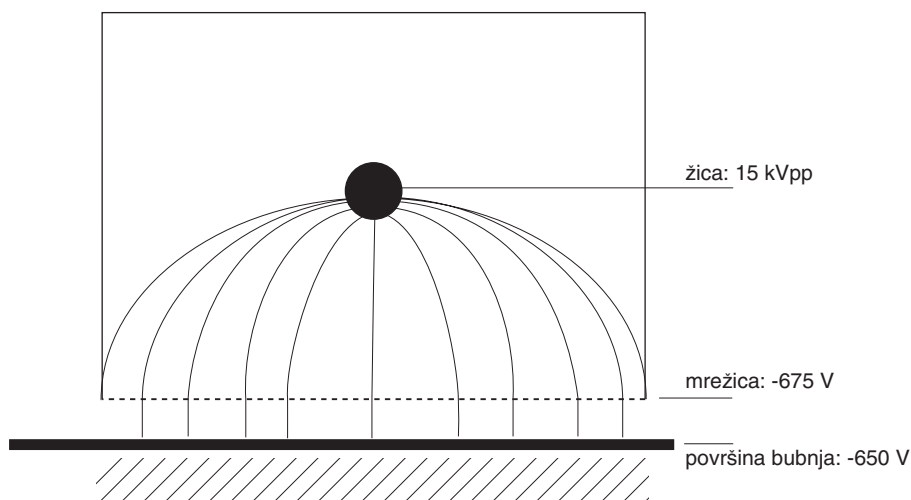
Veličina otiska je definirana količinom instalirane memorije koja je pridružena svakoj tiskovnoj jedinici (od 36MB do 252 MB). Brzina tiska je 2100 A4 obostrano otisnutih stranica u jednom satu (odnosno 4200 A4 formi).

Fizički centar tiskovne jedinice (Slika 3) je organski fotovodljivi bubanj (OPC) koji se sastoji od uzemljene aluminijske baze prekrivene sa organskim fotovodljivim slojem debljine 18 μ m. Taj sloj je nevodljiv u mraku, a postaje vodljiv na mjestu eksponiranog svjetla.



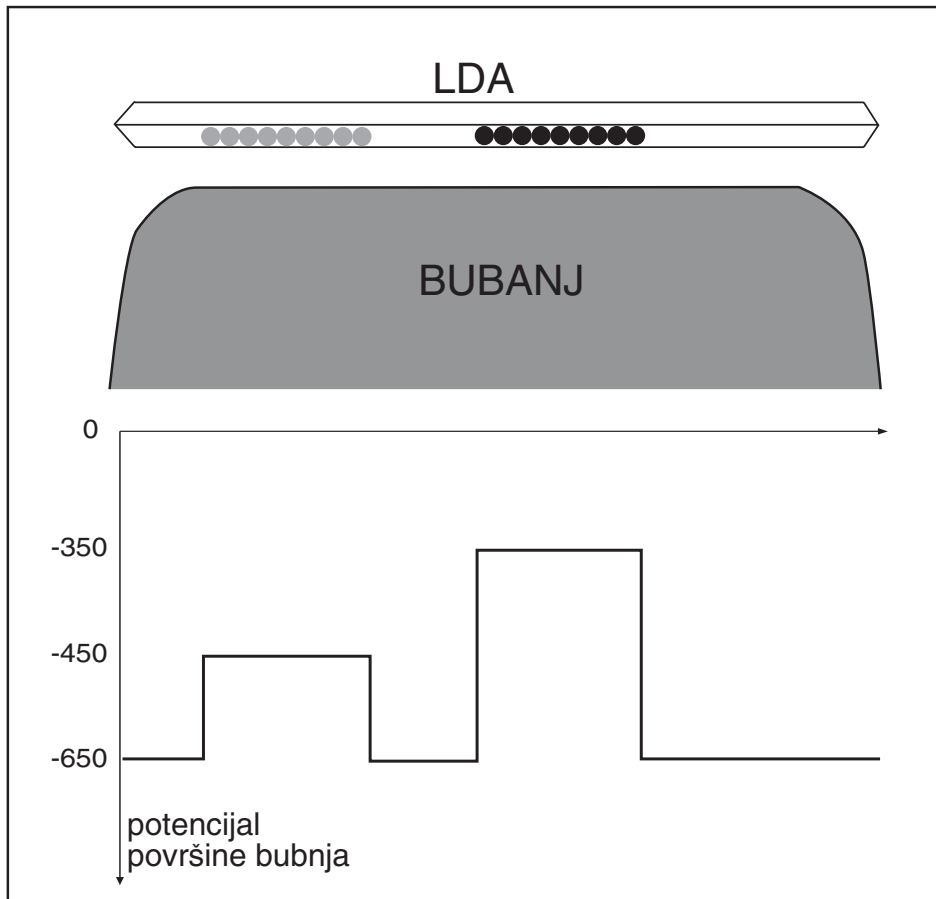
Slika 3
Tiskovna jedinica

Nabijanje bubnja (Slika 4) na radni potencijal od -650 V vrši tanka žica spojena na izmjenični napon od 15 kVpp koja je smještena unutar uzemljenog okvira sa jednim otvorom na kojem se nalazi električna mrežica na istosmjernom naponu od -675 V kroz koju se raspršuje negativan naboj na površinu bubnja. Ta žica sa svojim okvirom čini koronu, a s dodatnom mrežicom čini skorotron za nabijanje bubnja. Tako samo negativan period izmjeničnog napona proizvodi naboje koji napuštaju skorotron jer naboje koje proizvede



Slika 4
Nabijanje bubnja

pozitivni period ne propušta negativna mrežica. Teorijski bi nabijanje bubnja trebalo završiti kada bubanj dosegne potencijal mrežice (-675 V), ali u praksi će bubanj doseći -650 V. Tako se s naponom mrežice može regulirati potencijal bubnja.

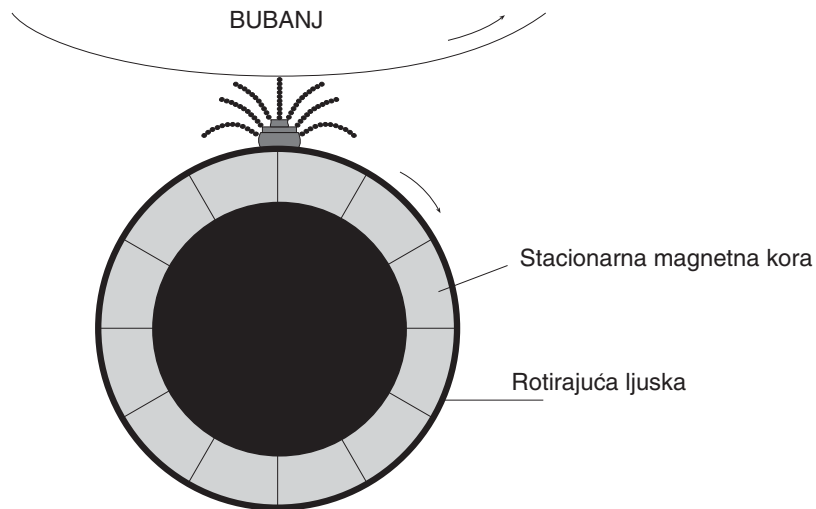


Slika 5
Zapis slike

Digitalna slika stvara se preko LED polja (Light Emitting Diode) eksponiranjem određene jačine svjetla na bubanj što uzrokuje razliku potencijala na onim mjestima gdje se očekuje otisak (Slika 5). LED polje se sastoji od 7424 dioda s razmakom od 42,3 mikrona (ili 600 dioda po linearnom inču).

Svaka dioda se može zasebno adresirati. Moduliranjem ekspozicije svake adresirane diode dobivaju se različite razlike potencijala o kojima ovisi intenzitet otiska točke. LED polje je postavljeno paralelno sa osi bubnja i tako se slika zapisuje liniju po liniju kako se bubanj okrene za 42,3 mikrona.

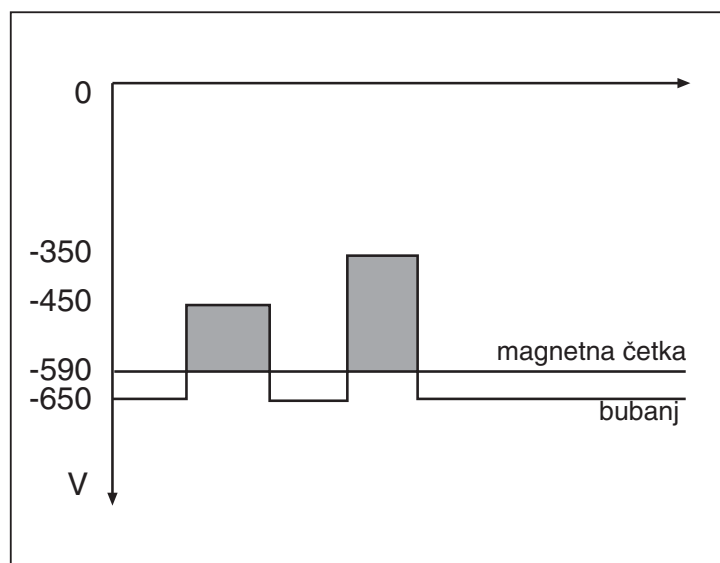
U razvijačkoj jedinici nalazi se smjesa tonera i magnetnog nosača. Suhi toner je baziran na polimernoj smoli i samljeven je u čestice prosječne veličine od 7.5 μm koje nisu vodljive, ali se mogu nabiti. Dijelovi nosača (npr. željezo) su magnetični i imaju ulogu električnog nabijanja čestice tonera kada se zajedno



Slika 6
Magnetna četka

miješaju (tribo-električni efekt). Toner će se spojiti s nosačem tako da je toner negativan, a nosač pozitivan. Razvijač (toner+nosač) nanesen je na magnetni bubanj formirajući magnetnu četku u razvijačkoj jedinici (Slika 6).

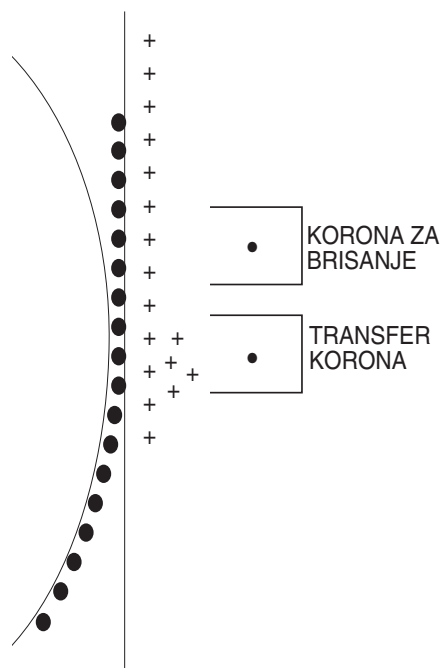
Taj magnetni bubanj je spojen na -590 V. Kada eksponirano područje na bubnju prođe pored magnetne četke, negativne toner komponente elektrostatski



Slika 7
Potencijal razvijanja

preskaču na to područje. Nosač ostaje u razvijačkoj jedinici jer je magnetna sila od magnetnog bubnja u razvijačkoj jedinici veća od elektrostatske sile stvorene prolaskom površine bubnja s "potencijalnom rupom" . Razlika između potencijala magnetne četke (-590 V) i aktualnog potencijala na površini bubnja je potencijal razvijanja (Slika 7). Veličina te razlike određuje koliko će se tonera deponirati na bubanj. Napon magnetne četke je moguće programski mijenjati.

Uporabom korone (tzv. transfer korone) s istosmjernom strujom od 100 μ A koja raspršuje pozitivne naboje na stražnju stranu papira (Slika 8), negativne toner komponente na bubnju prelaze na papir. Pošto je električno polje između



Slika 8
Transfer tonera na papir

obje strane papira jako veliko (od 2000 do 3000 V), potrebno je to polje smanjiti da ne bi došlo do lokalnog probijanja zraka u malom prostoru između bubnja i papira. Za to je zadužena korona za brisanje koja je priključena na izmjenični napon od 10kVpp.

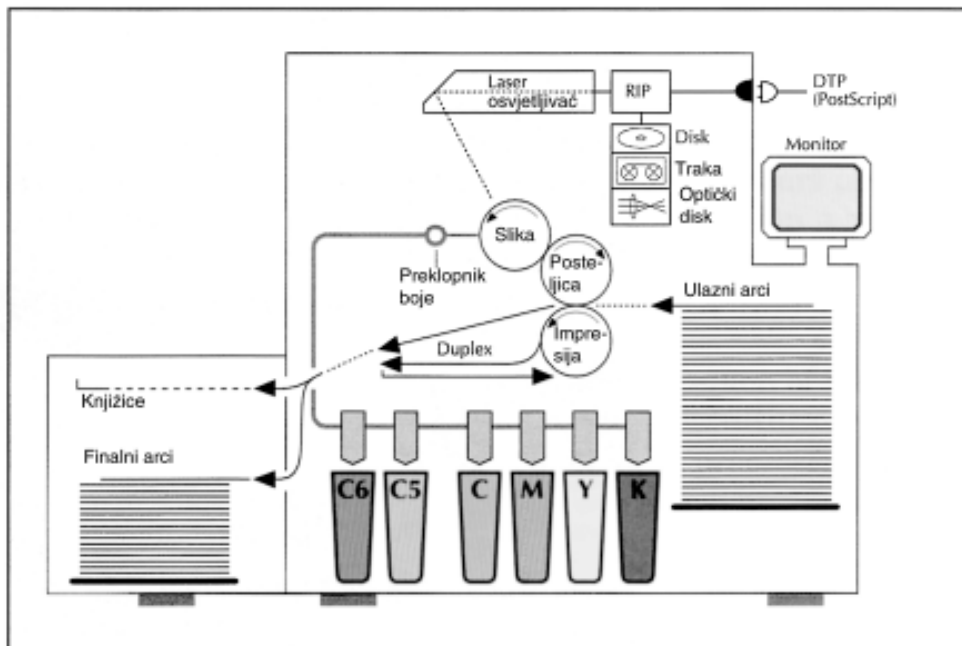
Nakon tog koraka, još je uvijek prisutna elektrostatska latentna slika na bubnju sa zaostalim tonerom koji nije uspio prijeći na papir. Prije nego što bubanj dosegne jedinicu za čišćenje, skorotron za prednaboj će neutralizirati latentnu sliku nabijajući bubanj na -500V što će pojednostaviti čišćenje

preostalog tonera i nabijanje bubnja s glavnim skorotonom na radnih -650 V (Slika 3).

Dodatne dvije korone (pozitivna duplex i negativna duplex) (Slika 3) pripremaju papir s deponiranim tonerom na sebi od ove (kao i prošle) tiskovne jedinice za slijedeću tiskovnu jedinicu. Cilj je da deponirani toner ostane na papiru ne utječući na prelaz tonera na papir u slijedećoj tiskovnoj jedinici.

Na kraju, papir s deponiranim tonerima svih boja prolazi kroz fazu fiksiranja gdje se boja stapa s papirom. Fiksiranje boje radi se beskontaktno, pa se otisak može vršiti na papirima velikog raspona gramature, ali i na materijalima kao plastika, transparentne folije, folije za naljepnice i slično. Svaki puta kada tisak miruje, mehanizam za fiksiranje boje je u zatvorenom stanju.

Digitalni tiskarski uređaji s tekućom elektrofotografijom [26, str.4.40-4.41] [53, str. 330-331] koriste tekući toner kao boju. Na slici 9 prikazan je takav stroj koji u jednom prolazu može digitalno otisnuti samo jednu boju, odnosno potrebno je



Slika 9
Digitalni stroj sa tekućom elektrofotografijom

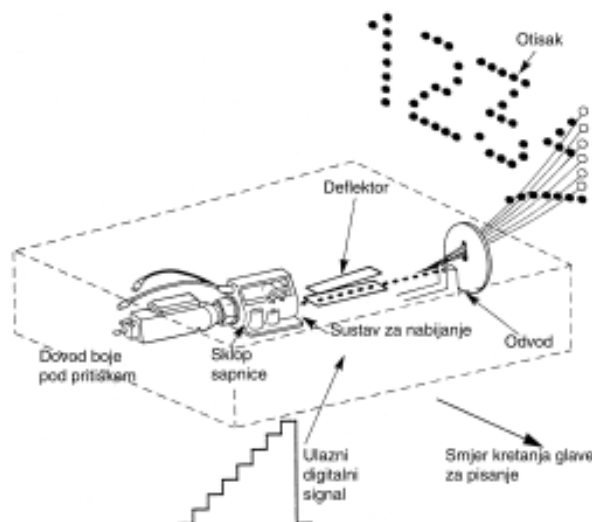
četiri prolaza za četverbojni otisak jedne strane plus okret medija za dodatnih četiri prolaza za drugu stranu. Brzina tiska je 8000 stranica po satu za jednu boju, a svaka slijedeća boja prepolavlja izlaznu brzinu. Izlazna brzina za četverbojni obostrani otisak je samo 500 stranica po satu ili 8 u minuti.

U trenutku spajanja s ispisnim medijem, tekući toner je u formi ljepljivog polimera dok se u tehnologiji sa suhim tonerom toner mora zapeći pod visokom temperaturom na ispisni medij. Pošto se koristi jedna tiskarska jedinica, potrebno je odbaciti tekuću boju s tiskarske posteljice nakon svakog okreta. Zbog tog zahtjeva razvio se tekući toner s relativno malom adhezijom, što daje za posljedicu neželjeno skidanje boje s nekih vrsta ispisnih medija.

3.1.2. Ispis tekućim mlazom

Osnovni cilj tehnologije ispisa tekućim mlazom (Ink Jet) [26, str. 4.42-4.43] [53, str. 321-333][63] je proizvesti kontrolirani mlaz individualnih kapljica elektrografske boje. Postoje dva pristupa: kontinuirani mlaz boje i pristup "kapljica na zahtjev" (drop-on-demand DOD).

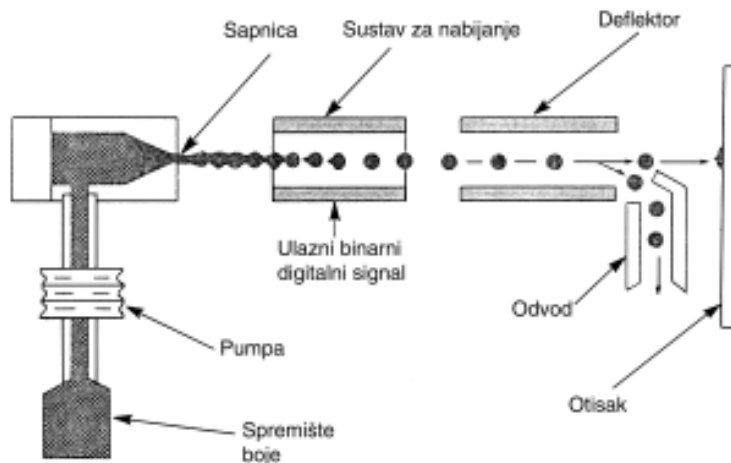
Na slici 10 prikazan je digitalni kontinuirani ispis tekućim mlazom u kojem se slikovni elementi stvaraju kontinuiranim mlazom na odgovarajućoj vertikalnoj



Slika 10
Digitalni kontinuirani ispis tekućim mlazom

poziciji dok se ispisna glava miče po horizontali. U mlaznicu ulazi tekuća boja pod kontroliranim pritiskom. Kontinuirani mlaz se pretvara u seriju individualnih kapljica preko pulsirajućih kontrakcija i ekspanzija keramičkog elementa u mlaznici. Pulsiranje može dolaziti od akustičkog ili piezoelektričnog efekta odnosno efekta da neki materijali mehanički reagiraju kada se na njihovu površinu dovede električka pobuda visoke frekvencije.

Frekvencija kapljica odgovara frekvenciji pobudnog napona i to obično u ultrazvučnom području od 100 i više kHz. Putanja kapljica boja od mlaznice do medija za tisak kontrolirana je elektrostatskim silama. Digitalni podatak svakog slikovnog elementa pretvara se u diskretni naponski nivo koji stvara odgovarajuću elektrostatsku silu za svaku kapljicu. Na slici 10 je to prikazano diskretnim brojem naponskih stepenica koja svaka odgovara ispisnoj poziciji kapljice. Taj princip je jako podložan promjeni temperature i vlage zraka.



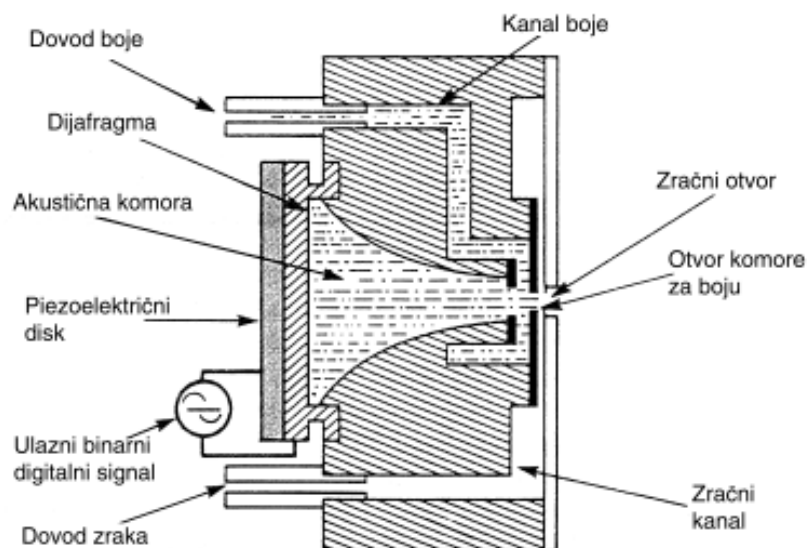
Slika 11
Binarni kontinuirani ispis tekućim mlazom

Na slici 11 prikazan je drugi princip kontinuiranog ispisivanja tekućim mlazom. On se naziva binarnim jer se jednim bitom definiše hoće li se kapljica elektrostatski nabiti ili ne. Ova metoda ne zahteva piezoelektrični efekt. Kapljice nastaju spontano u mlaznici kao rezultat efekta površinske napetosti tekućine. To se

dobija projektiranjem odnosa geometrije mlaznice i velikog tlaka u spremniku boje. Stvara se brzi i uski mlaz boje.

Elektrostatski nenabijene kapljice prolaze ravno na ispisni medij dok nabijene kapljice deflektor skreće u slivnik za recirkulaciju. Ovom metodom mogu se dobiti kapljice manje u promjeru od 0,0003 inča (0,008 mm) u frekvenciji 1.000.000 u sekundi pa je moguće ostvariti ispisnu rezoluciju od 300dpi.

U metodi kontinuiranog ispisa mlazom više od pola tekuće boje koja je izašla iz mlaznice završava u recirkulaciji dok je samo mali dio potreban za formiranje

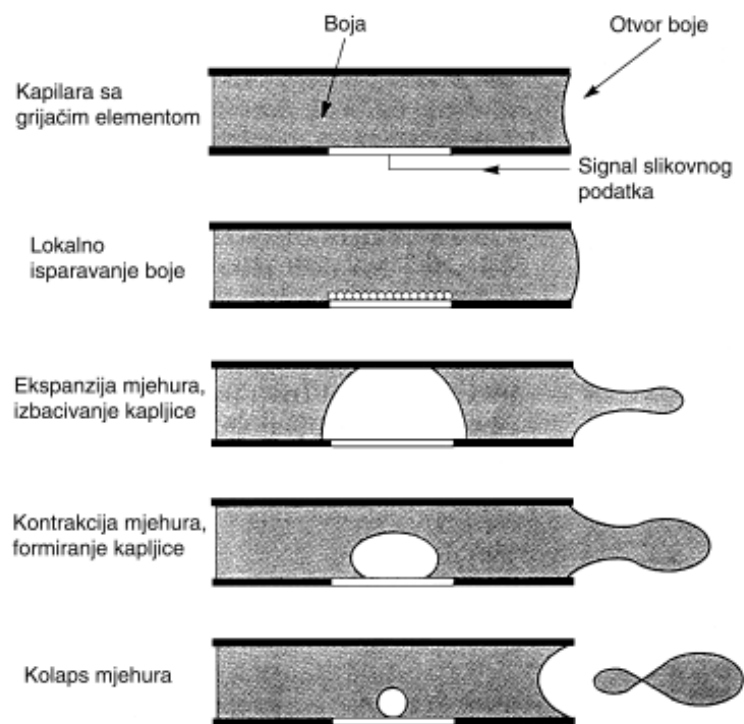


Slika 12
Piezoelektrični "kapljica na zahtjev" ispis

slike ispisa. Zbog toga se razvio princip "kapljica na zahtjev" (drop-on-demand DOD). Ovaj princip ne zahtjeva nabijanje kapljica, deflektor i recirkulaciju.

Kemijska svojstva boje su manje zahtjevna nego kod kontinuiranog principa gdje boja mora zadržati svoja svojstva duže vrijeme kroz nekoliko recirkulacijskih ciklusa.

Na slici 12 prikazan je piezoelektrični DOD princip. Piezoelektrični kristal nalazi se na stražnjoj strani mlaznice. Puls kristala stvara kapljicu koja izlazi iz mlaznice samo onda kada treba ispisati slikovni element. Izlaz mlaznice se nalazi u zračnoj komori koja svojom konstrukcijom definira oblik i veličinu izbačene kapljice. Kapljice se stvaraju frekvencijom oko 20.000 u sekundi i ostvaruje se ispisna rezolucija od 200 dpi. Ova metoda omogućuje projektiranje višestrukih ispisnih mlaznica male veličine i velike gustoće što još povećava ispisnu rezoluciju. Ovaj princip se još naziva samo piezoelektrični ispis mlazom.



Slika 13
Nastajanje kapljice u termalnom ispisu mlazom (bubble jet)

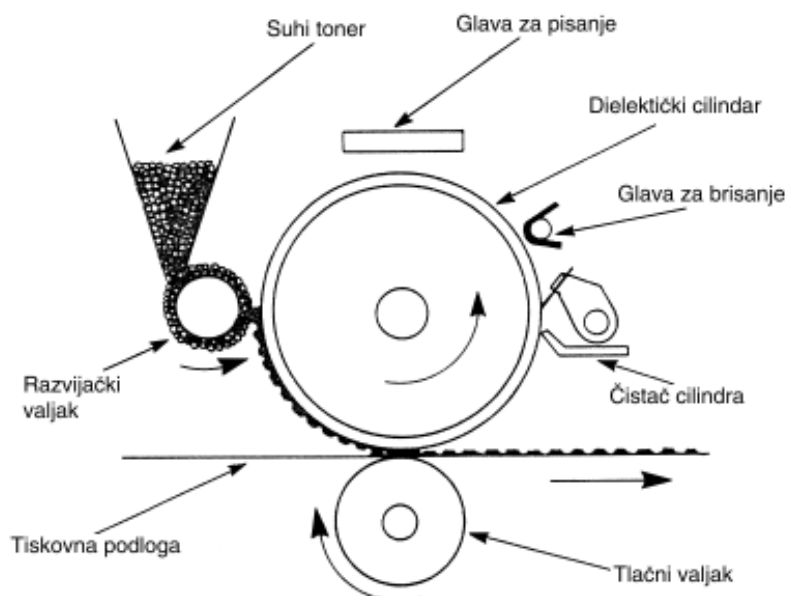
Drugi DOD princip koristi vremenski precizne pulseve toplinske energije za stvaranje kapljica pa se naziva termalni ispis mlazom . Taj princip se još naziva principom mjehurastog mlaza (bubble jet). Na slici 13 se vidi princip nastajanja kapljice boje. Osnovni element ove metode je kapilarni kanal unutar kojeg je ugrađen električki otpornik kao grijači element. Kada kroz otpornik prođe

električni strujni signal slikovnog elementa, generira se toplina koja uzrokuje lokalno isparavanje boje i formiranje mjehura unutar kapilarnog kanala. Nastajanje mjehura stvara tlak koji uzrokuje stvaranje kapljice nakon koje se urušava mjehur što vraća boju u kapilari u normalno tekuće stanje.

Prednost uređaja s termalnim ispisom mlaza u odnosu na piezoelektrične uređaje je što se signal slikovnog elementa prenosi električki blizu otvora mlaznice gdje se formira mjehur u kapilari. Piezoelektrično stvaranje kapljice zahtjeva relativno dugačku mlaznicu gdje stvaranje akustičnih valova pod različitim temperaturama daje stanovitu smetnju.

3.1.3. Ionografija

Ionografija [53, str. 326] isto koristi bubanj kao nositelj latentne slike kao i elektrofotografija samo što izvor nije svjetlo već mlaz iona odnosno nabijene molekule plina i što je bubanj dielektrik (ne provodi struju). Ioni koji dođu do površine bubnja stvaraju latentnu sliku u formi naboja (potencijalnih rupa). Ionografija koristi jednokomponentni toner za razliku od dvokomponentne



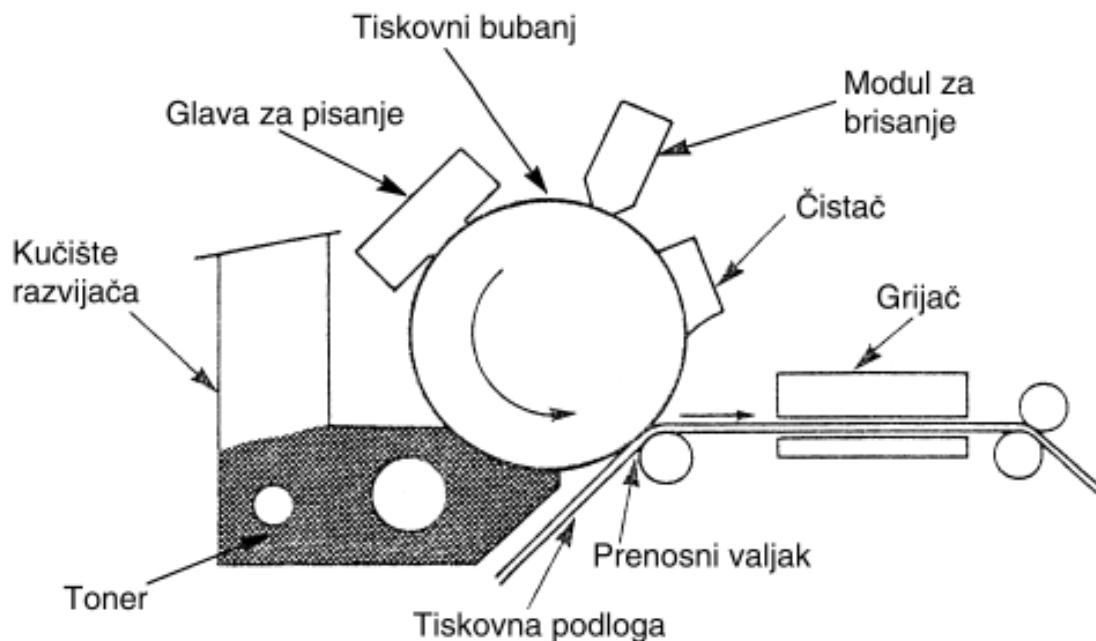
Slika 14
Tiskovna ionografska jedinica

kombinacije tonera i nosača kod elektrofotografije. Limitirana je nepostojanjem jednokomponentnih tonera u boji.

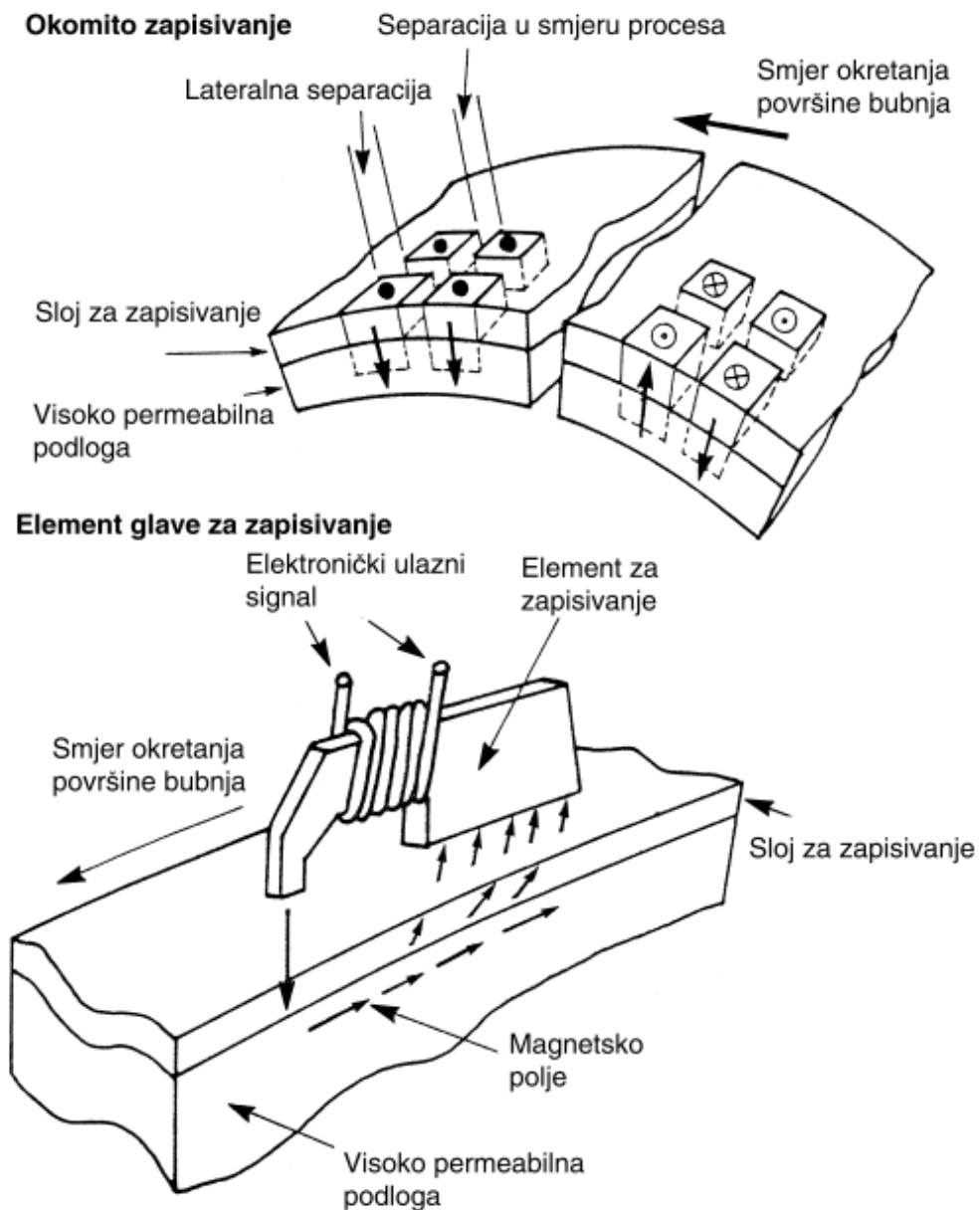
Na slici 14 je prikazana tiskovna ionografska jedinica. Kada latentna slika prođe ispod razvijачke jedinice, toner prelazi na latentne slikovne elemente na bubnju zbog električnog polja stvorenog razlikom potencijala između slikovnog bubnja i razvijачkog bubnja. Tonske komponente prenose se na ispisni medij mehaničkim pritiskom i razlikom potencijala između dielektričnog bubnja i gumenog tlačnog valjka. U cijelom procesu nema potrebe za toplinskom energijom kao kod elektrofotografije u fazi taljenja tonera na ispisni medij.

3.1.4. Magnetografija

Magnetografija [64], kao i ionografija koristi jednokomponentni toner za koji ne postoji toner u boji. Latentna slika se stvara na magnetskom metalnom



Slika 15
Tiskovna magnetografska jedinica



Slika 16
Magnetografski elementi za zapisivanje latentnih slikovnih elemenata

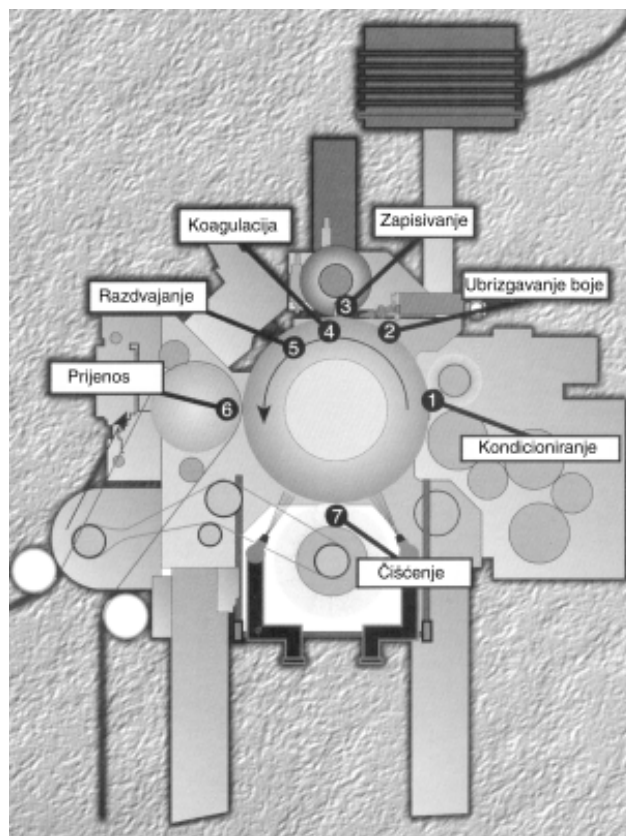
bubnju pomoću magnetskog polja. Toner u magnetografiji se sastoji od finih željeznih elemenata koji se pod utjecajem magnetske sile prenose na magnetnu latentnu sliku na bubnju.

Na slici 15 prikazana je magnetografska jedinica gdje se u ispisnoj glavi pretvara informacija slike u magnetsko polje. Kada se aktivira magnetsko polje iznad površine bubnja, dolazi do promjene magnetskog polariteta individualnog djelića metalne površine bubnja što stvara latentni slikovni element.

Magnetski zapis u magnetografiji je okomit na bubanj (slika 16) za razliku od zapisa na magnetske trake i diskove. Magnetografija daje velike brzine ispisa, veće nego elektrofotografija.

3.1.5. Elkografija

U elkografiji [53, str. 335][65][66][67] se tekuća boja elektrostatski zgrušava (elektro-grušanje, elektrokoagulacija) na poziciji slikovnog elementa pa se



Slika 17
Elkografska tiskovna jedinica

potom prenosi na ispisni medij. Elkografska boja je posebna elektrolitički osjetljiva polimerička boja (kao pasta). Da bi boja bila električki vodljiva, integrirane su elektrolitičke soli u pigmente polimeričke vodene boje. Tiskovna glava je katoda, a rotirajući bubanj na kome se stvara latentna slika je anoda.

Elkografska tiskovna jedinica prikazana je na slici 17. U prvom koraku se stvara tanki nanos ulja (faza kondicioniranja) prije nego se ubrizga boja. To omogućuje kasnije lakši prijenos boje na papir. U drugom koraku ubrizgava se boja s paralelno postavljenim ubrizgivačima.

U fazi zapisivanja stvaraju se električni impulsi iz integrirane elektroničke ispisne glave. Ona pretvara slikovni zapis poslan iz 4 GB međumemorije (buffer) u impulse električne struje. Ispisna glava ima redak od 3424 elektroda ili katoda koja ostvaruje ispisnu rezoluciju od 200 linija po inču. Elektroda je metalna žica od 50 mikrona koja šalje električnu struju kroz boju. Elkografska boja prenosi električne signale varijabilne vremenske duljine do slikovnog bubnja kao anode.

U fazi koagulacije odnosno zgrušavanja dolazi do prijanjanja (adhezije) boje kao ugrušaka boje na površini bubnja kao anode. Anoda privlači polimere iz boje koji se zgrušavaju na njezinoj površini. U cijelom periodu tiska je procjep između katode i anode od 50 mikrona potpuno popunjen sa bojom uvijek spreman za električne signale zapisivanja.

Pošto na površini bubnja postoje istovremeno ugrušci boje kao slikovni elementi i okolne tekuće boje vrši se faza razdvajanja ugrušenih točaka od tekuće boje. To se radi jednostavnim gumenim nožem jer su ugrušene točke više kohezivne nego okolna tekuća boja.

I nakon faze razdvajanja ugrušene točke još uvijek imaju u sebi vode. Zbog toga je dovoljan hladni pritisak tlačnog valjka da bi se mokra slika prenijela na

ispisni medij. Na njemu se boja osuši ishlapljivanjem. U stanici za čišćenje skidaju se ostaci boje i uljnog sloja sa četkama, sapunom i vodom pod visokim pritiskom iz mlaznica.

Jedna katoda može formirati s jednim signalom svaku točku u 256 sivih razina. To se izvodi u inkrementima vremena od 15 nanosekundi za svaku sivu razinu. Što je dulji signal to je točka ugruška deblja. Pošto je ugrušak od transparentne elkografske boje, u našim očima će biti viša siva razina ako je primjenjen dulji signal na katodi. Današnja elkografska tehnologija je limitirana elektroničkom kontrolom katoda odnosno adresiranjem istih pa je brzina ispisa 4 mikrosekunde po točki što odgovara brzini od 1 metar u sekundi ili 850 kolor stranica u minuti ako imamo u lancu 4 elkografske stanice u boji.

3.2 Sučelje za obradu slikovnih elemenata i rastera (Raster Image processor - RIP)

Standardni programski jezik s kojim komuniciramo s digitalnim ispisnim uređajima je PostScript. To je ostvareno preko PostScript interpretera [68] kao programskog jezika implementiranog u sučelje za obradu slikovnih elemenata i rastera (Raster Image processor - RIP) koji je pridružen digitalnom ispisnom uređaju. Na taj način se ostvaruje ispis iz bilo kojeg programa koji može stvoriti PostScript zapis na bilo koji digitalni ispisni uređaj. Opis ispisne stranice u PostScript jeziku je definiran u korisničkom koordinatnom prostoru. U RIP-u će se izvršiti konverzija iz korisničkog koordinatnog prostora u koordinatni prostor ispisnog uređaja s definiranom ispisnom rezolucijom. Osnovne faze koje sučelje za obradu slikovnih elemenata i rastera mora izvršiti su:

- faza procesiranja: provjera sintakse i semantike jezika, konverzija PostScript naredbi u osnovne grafičke primitive odnosno stvaranje naredbi nižeg nivoa u vidu liste grafičkih primitiva pomoću PostScript interpretera. Stvorena lista sastoji se od onih naredbi nižeg nivoa koje definiraju samo one objekte koji čine grafičku stranicu. U toj fazi koristi se RIP RAM [69] ili ako postoji i RIP disk [70].

- faza rastriranja: preuzima listu grafičkih primitiva iz faze procesiranja i stvara podatke za svaki ispisni element čitave stranice odnosno okvira (frame) u frekvenciji potrebnoj za definiranu rezoluciju ispisnog uređaja. Stvaranje takove bitmape ili pixmape, ovisno o arhitekturi RIP-a, može se izvoditi tako da se okvir rastrira u RIP RAM-u s trenutačnim ispisom i ponovnim punjenjem slijedećeg okvira ili se rastrirani okvir stvara na slikovnom tvrdom disku (RIP DISK) s trenutačnim ili naknadnim ispisom.

U fazi PostScript procesiranja mora se riješiti cijela sintaksa i semantika jezika. Moraju se izvršiti sva izračunavanja i akcije nad svim mogućim tipovima podataka kao što su brojevi, polja, znakovni nizovi i drugi. Moraju se izvršiti sva uvjetna grananja, petlje i procedure koje su moguće u PostScriptu. Također se programske krivulje segmentiraju u linijske segmente. Ti se segmenti potom podebljavaju na zadanu debljinu i dobivamo segmentiranu debelu krivulju. Grafički primitivi se stvaraju nakon upotrebe transformacijskih naredbi koordinatnog sustava kao što su naredbe scale i rotate, maskiranja s naredbom clip i drugih naredbi.

Faza rastriranja je različita za različite ispisne digitalne uređaje. Glavni zadatak te faze je kreiranje impresije sivoće ili tona boje u našim očima za ispisni digitalni uređaj spojen na RIP. Tu se vrši punjenje binarnog podatka za svaki adresirani ispisni slikovni element. Binarni podatak se nalazi na odgovarajućoj memorijskoj adresi u raster memoriji koja se još naziva i slikovna memorija, a njen cijeli sadržaj bitmapa ili pixmapapa. Svi procesi u ovoj fazi odvijaju se u koordinatnom sustavu ispisnog digitalnog uređaja.

Pretvaranje grafičkih primitiva zadanih atributa u odgovarajuće slikovne elemente može naići na probleme koji se moraju algoritamski rješavati. Na primjer, kada linija zadane debljine ne pada točno u granicama ispisnih slikovnih elemenata mora se izvršiti korekcija debljine linije, a da to naše oči što manje vide. Izvršava se konverzija ulaznog kolor prostora u ispisni kolor prostor zadanog uređaja. U ovoj fazi se moraju rješavati sve pojave karakteristične za ispisne uređaje da bi se dobio optimalno kvalitetan ispis. To su poslovi kao što je kompenzacija tonskih vrijednosti za efekt proširivanja rasterske točkice određene ispisne tehnologije (dot gain), nesavršenost preklapanja registra višebojnog tiska (trapping control)[74][75], optimalno pozicioniranje PostScript stranica po cijeloj površini ispisnog medija definiranog uređaja (impozicija) [72][73] kao i zamjena niskorezolucijskih zamjenskih slika sa visokorezolucijskim originalnim slikama (OPI control) [71].

3.3. Osnovni postupci rastriranja

Ljudsko oko vidi kontinuirane signale odnosno kontinuirane tonove boja. Digitalni signal neće nikad moći potpuno reproducirati analogni signal. On je prihvatljiv kao original sve dok se kontinuirani signal uzorkuje većom frekvencijom nego što je limit ljudske percepcije.

Konvertiranje analognog signala tona boje u digitalni odvija se preko diskretizacije sivih razina slikovnih elemenata preko definirane sive skale. Siva skala određuje diskretni broj sivih stepenica koje može poprimiti digitalni signal nakon konverzije iz analognog. Ljudsko oko može razlikovati oko 150 sivih nivoa, a to znači da nam treba najmanje 8 bitova za digitalni zapis sive razine jer je 8 najmanja potencija od 2 koja je veća od 150 ($2^8=256$).

Jednom odabrani uzorci iz originalne izvorne kontinuirane slike, u procesu uzorkovanja, moraju se prikazati na nekom uređaju s procesom rekonstrukcije koji pokušava nanovo stvoriti sliku što bliže originalnoj iz stvorenog skupa uzoraka. U tom procesu dolazi do kvantizacijske greške u tonu i prostoru. Kvantizacijska greška tona se pojavljuje u obliku vidljivih granica između tonova (stepenice tonova u tonalnim gradacijama). Prostorna kvantizacijska greška se pojavljuje kao limitirana selekcija rasterskih koordinata ispisnog uređaja. Jedan od načina smanjivanja vizualnih učinaka greške tonalne kvantizacije je difuzija greške (eng. error diffusion) na susjedne pozicije slikovnih elemenata, ali jako usporava proces rastriranja. Greške prostorne kvantizacije se pokušavaju riješiti algoritmima za skanersku konverziju (eng. scan conversion). Ti algoritmi daju metode na koji način se prisilno pridružuju ispisne koordinate ulaznim koordinatama u trenutku kada se dogodi promašaj istih.

Ako je ulazni objekt digitalna slika s koordinatama uglova svojih slikovnih elemenata onda se može primjeniti slijedeći jednostavan algoritam za

skanersku konverziju [46, str. 502]. Granice izlaznog slikovnog elementa uvijek padaju na cjelobrojne (diskretne) koordinate u ispisnom prostoru. Slikovni element je kvadratična površina označena s lokacijom svoga ugla u koordinatnom prostoru (najčešće lijevog donjeg ugla). U ispisnom prostoru ta se površina drži poluotvorena što znači da sadrži svoje donje i lijeve, ali ne i gornje i desne granice. Preciznije rečeno za bilo koju točku čije su koordinate realni brojevi (x,y) dopušten je izraz $i=\text{floor}(x)$ i $j=\text{floor}(y)$. Slikovni element koji sadrži te točke se identificira sa (i,j) . Područje koje pripada tom slikovnom elementu se definira sa skupom točaka (x',y') tako da je $i \leq x' < i+1$ i $j \leq y' < j+1$.

Većina digitalnih tiskarskih procesa koristi binarne slikovne elemente (boja je prisutna ili ne). Metode reprodukcije tonских vrijednosti binarnim načinom datiraju od vremena prvih tiskarskih strojeva i tehnika. Osnova tih metoda je stvaranje sitnih površina otisnute boje koje se ne mogu međusobno razlučiti u ljudskom oku. U oku se nakupine takvih površina u kombinaciji s bojom podloge percipiraju kao prosječna vrijednost tona boje. Tada naše oko obavlja takozvanu prostornu integraciju (eng. spatial integration). Takve metode se nazivaju metode rastresanja (eng. dithering).

Metode rastresanja se mogu klasificirati u dvije kategorije. Jedna je rastresanje po pragu kvantizacije (eng. threshold dithering), gdje se svaki slikovni element ulazne slike uspoređuje s zadanom vrijednošću praga da bi se dobila vrijednost ispisnog slikovnog elementa. Toj kategoriji pripadaju metode: prosječno rastresanje (eng. average dithering), uređeno rastresanje (eng. ordered dithering) i stohastičko rastresanje (eng. stochastic dithering).

Druga kategorija je rastresanje s difuzijom pogreške (eng. error diffusion dithering), gdje se greška zaokruživanja svakog slikovnog elementa kompenzira tako što se prenaša na susjedne slikovne elemente. Tipičan primjer je Floyd-Steinberg rastresanje.

U metodi prosječnog rastresanja se prvo izabere neka konstantna siva razina, obično prosječna vrijednost svih ulaznih slikovnih elemenata, za definiranje globalnog praga kvantizacije za 0 ili 1 ispisne vrijednosti (crno ili bijelo). Svi slikovni elementi čija siva razina leži ispod praga kvantizacije dobivaju vrijednost 0, za obrnuto 1. Nedostatak je što su vidljive konture kvantizacije.

Uređeno rastresanje (eng. ordered dithering) uvodi matricu ili masku s različitim vrijednostima pragova kvantizacije. Tu se primjenjuje periodična maska pragova kvantizacije na svaki ton ulazne slike. Ispisni slikovni elementi se konvertiraju u 0 (crno) ako su njihove sive razine ispod praga ili 1 (bijelo) u obrnutom slučaju. Tako se stvara rastrirana slika čiji slikovni elementi imaju vrijednosti od 0 do 1. Dio slike koja odgovara primjenjenoj masci zove se rasterska ćelija. Maska pragova kvantizacije od M slikovnih elemenata ima pragove $0, 1/M, 2/M, \dots, 1$, što podržava $M+1$ sivih razina u rasterskoj ćeliji. Raspored pragova u maski imat će utjecaj na vizualnu kvalitetu rastriranja. U srednjim tonovima, pola će slikovnih elemenata u rasterskoj ćeliji biti "upaljeno", a pola "ugašeno". Rastresanje s uređenim nakupinama (eng. cluster dot order dither) će stvarati nakupine na povezan način što smanjuje efekt povećanja ispisne točkice zbog penetracije boje na papiru (eng. dot gain). Metoda rastresanja s uređenim nakupinama često se naziva polutoniranje (eng. halftoning) ukoliko nije drukčije specificirano. Takve uređene nakupine u rasterskoj ćeliji stvaraju osnovni rasterski element (RE). Kada se takav digitalni polutonski raster šalje na ispisni uređaj s PostScript RIP-om potrebno je specificirati frekvenciju rastera (linijatura), kut rastera i oblik RE. PostScript dozvoljava slobodno definiranje rasterskog oblika za različite grafičke objekte što je iskorišteno u stvaranju novog načina rastriranja u ovoj disertaciji.

Idealni oblik rasterskog elementa (RE) bi trebao [50, str. 64] :

- a) minimizirati efekt povećanja ispisne točkice zbog penetracije boje na papiru (eng. dot gain) kroz cijelu sivu skalu,
- b) održanje vidljivosti boje na najmanjoj vrijednosti sive skale (2 do 3%),
- c) održanje željenog ponašanja u sivim razinama gdje se oblik RE počinje spajati,
- d) otpor tendenciji preranog zapunjenja za boju na velikim vrijednostima sive skale (80% na više).

Integritet oblika RE za sve razine sive skale je ostvaren postupkom rastuće sekvence (eng. growth sequence) [111, str. 569] gdje se vodi računa da se svaki slikovni element koji je jednom upaljen za sivu j razinu, mora ostati upaljen za sve sive razine $k > j$.

Niti jedan oblik RE ne može odgovoriti idealno na sva četiri zahtjeva pa se rade kompromisi. Upravo zato se u modernim algoritmima upotrebljava više varijacija oblika RE za različite intervale sive razine.

Kod stohastičkog rastresanja (eng. stochastic dithering) se za svaku vrijednost ulaznog slikovnog elementa generira slučajni broj 1..256 kao kvantizacijski prag. Ako je taj broj veći od vrijednosti slikovnog elementa na toj poziciji se dobiva vrijednost 1 odnosno 0. Ova metoda praktični nema regularne vizualne artefakte. Ovu metodu može degradirati loša distribucija slučajnih brojeva.

Rastresanje s difuzijom pogreške je najbolje reprezentirano s Floyd-Steinberg rastresanjem gdje se greška zaokruživanja svakog slikovnog elementa kompenzira tako što se prenaša na susjedne slikovne elemente. Postoje varijacije ove tehnike ovisno o kojim se susjednim slikovnim

elementima radi. Na primjer ako originalne vrijednosti slikovnih elemenata označimo s k_1, k_2, k_3, \dots , i kvantizirane (zaokružene) vrijednosti s r_1, r_2, r_3, \dots . Greška zaokruživanja e_1 prvog slikovnog elementa je $e_1 = r_1 - k_1$, a u greški e_i svakog slijedećeg slikovnog elementa se kompenzira greška prethodnog slikovnog elementa s $e_i = r_i - (k_i - e_{i-1})$.

Primjer rastresanja s difuzijom pogreške gdje su moguće ispisne vrijednosti 0, 3 i 6:

k:	1	1	1	2	3	4	7	1	5 ...
r:	0	3	0	3	3	3	6	3	3 ...
e:	-1	1	0	1	1	0	-1	1	-1 ...

$$r_i := \text{round}(k_i - e_{i-1}); \quad e_i := r_i - k_i + e_{i-1};$$

4. Modeliranje novih programskih modela za rastriranje

Razvoj rastriranja predloženog u ovoj disertaciji izvire iz klasične metode rastresanja s uređenim nakupinama (eng. cluster dot order dithering) ili često nazivano kao polutoniranje (eng. halftoning) koje se bazira na rasterskom elementu (RE).

Oblici rasterskih elemenata (RE), u najvećem dijelu, realiziraju se u tri oblika [43, str. 141-143]: okrugla točka, romb i linija jer su se ovi mogli proizvesti optomehaničkim postupcima. Drugačiji oblici nisu izvađani do digitalnog tiskarstva kada je ponuđena mogućnost matematičkog definiranja rasterske ćelije [29]. Današnja je praksa u većem dijelu svedena na simulaciju spomenuta tri oblika jer su njihove matematičke interpretacije rasta zacrnjenja rasterskog područja bile objavljene u PostScript literaturi. Ovdje je ponuđena metodologija kreiranja novih modela rastriranja, ponuđeni novi modeli kao dokaz o uspješnosti istraživanja te dati razlozi nužnosti otvaranja područja "individualiziranog rješenja" rastriranja. Uvađamo proceduru stohastičkog izbora rasterske ćelije s parametrima linijature, rasterskog kuta i odabira oblika. Predloženo rješenje je podržano na svim ispisnim tehnologijama koje podržavaju direktno PostScript polutoniranje (binarne ispisne tehnologije), a to znači da višebitne digitalne ispisne tehnologije treba u momentu primjene ovog algoritma podesiti na binarni način rada (za svaku boju posebno), a to sve nabrojane u ovoj radnji i mogu.

U ovom radu naglašena je prisutnost različitih rasterskih oblika. Autori o ovom pitanju kao V. Ostromoukhov [102][103][104] predlažu izrađivanje fiksne preddefinirane konture s blend tehnikom npr. u Adobe Illustratoru ili FreeHandu za svaki željeni sivi nivo pa se za svaki sivi nivo rasterizira odgovarajuća kontura. U ovom radu predložen je novi način koji je orjentiran na stvaranje polja programiranih rasterskih oblika da bi se moglo sa stohastičkim odabirom iz tog polja birati oblici, definirati kutevi od 0 do 90, kao i linijature u dozvoljenom rasponu linijatura ispisnog uređaja. Stvaranjem PostScript algoritama koji ovisno o sjemenu generatora slučajnih brojeva (SEED) implementira pseudoslučajno N individualnih rasterskih oblika, kuteva i linijatura dobila se unikatnost ispisa preko jednog jedinog broja (SEED). Taj način se može implementirati na svakom današnjem PostScript RIP-u.

U ovom radu govorimo o reprodukciji specifičnog područja koje više obraća pažnju na jedinstvenost rješenja i nešto što se ne može ponoviti. Potreba za takvom primjenom je u području reprodukcije grafika sa fiksnim brojem otisaka, reprodukciji poštanskih maraka, reprodukcija vrijednosnica i onih dokumenata koji se digitalno tiskaju u tek jednom primjerku; integrirana slika na putovnici, bedževima, propusnicama.

Područje je neistraženo zbog toga jer su potrebna prevelika sredstva za izradu analogne realne podloge za mjerenje. Još se i danas, na primjer, poštanske marke (filatelistički interes) rastriraju klasičnim rasterima [50][76] koji nemaju nikakve "zaštitne elemente". Simulacija rastriranja daje nam opsežan uvid u ponašanje moirea [20][50, str. 43-61] i kontrolu prirasta RE [77][78][79][80][81].

Stohastičko rješenje individualizacije otiska je rubno područje digitalnog tiska. Promjene u PostScript naredbama omogućuju programsku intervenciju za vrijeme otiskivanja. Prvo, na svakom otisnutom arku moguć je novi broj, druga slika ili tekst pripremljeni u izdvojenim datotekama. Drugo, individualizacija može biti generirana algoritmom (što se najčešće koristi u numeraciji), a to znači da se ispisuju podaci kojih nema u memoriji računala već nastaju logikom određenom u programu pisanom u PostScriptu. U ovom radu se postavlja tvrdnja da najviši stupanj individualizacije čine slučajno izabrani parametri linija i krivulja u zatvorenom skupu mogućih pomaka ili slučajno izabran oblik rasterskog elementa za svaki slikovni element posebno. Postoje eksperimenti sa stohastičkim promjenama Bezier krivulje u vektorskoj grafici koji su izučavani prethodnih godina i djelomično objavljivani [18].

Ponudjen je originalan pristup izrade matematičkog modela novih rasterskih ćelija pomoću programa "Mathematica" [82][83][84] jer je od ideje dala vizuelno grafičko rješenje a time osim kvantitativnog testiranja ispravnosti, i kvalitativno likovno rješenje.

Dosadašnji objavljeni radovi pojedinih autora [105][106][107][108] prikazuju metode izrada rješenja koja se ne procesiraju direktno u PostScript RIP-u već se pridruživanje preddefiniranih oblika za svaki sivi ton odvija na strani računala kao stanice za renderiranje da bi se potom izvršio ispis. Novost u ovom radu je metoda za procesiranje na strani PostScript RIP-a ispisnog uređaja. To znači da podaci ulazne slike zajedno s algoritmom putuju prema RIP-u, a SEED parametar koji individualizira sliku generira lanac slučajnih brojeva iz generatora slučajnih brojeva RIP-a, a ne ulaznog računala. Na taj način se je ispis individualizirao ne samo algoritmima novih rasterskih oblika već i RIP-om ispisnog uređaja gdje su PostScript generatori slučajnih brojeva različiti od uređaja do uređaja. Na SEED parametar može se gledati i kao elektronski potpis slike. Ako se za svaki ulazni kanal boje definiraju različite SEED vrijednosti mogu se dobiti višestruke kombinacije zaštite protiv krivotvoritelja.

Svaka od postojećih objavljenih metoda "umjetničko rastriranje" [102][103][104], "polutonski postupci bez rastera" [108], "slikovno bazirano rastriranje"[107] ima drugačije ciljeve. Ovdje je ideja vodilja bila činjenica da se moderno digitalno tiskarstvo razvilo i razvija u PostScript okolini pa se metoda razvila s PostScript mehanizmima koje "razumiju" svi moderni tiskarski uređaji. Osnovni algoritam koji se odvija na strani RIP-a bi se mogao prikazati pseudokodom na ovaj način:

```
for (vrijednost tona svakog piksla slike)
{
    postavi SEED parametar;
    generiraj slučajni broj za algoritam rasterskog oblika;
    generiraj slučajni broj za kut rastriranja;
    generiraj slučajni broj za linijaturu rastriranja;
    rastriraj sa definiranim rasterskim oblikom, kutem i linijaturom;
}
```

Rješenje ponuđeno u ovoj disertaciji se je eksperimentalno razvijalo da bi zadovoljilo ovim kriterijima idealnog oblika RE: održanje vidljivosti boje na malim vrijednostima sive skale, održanje željenog ponašanja u sivim razinama gdje se oblik RE počinje spajati i otpor tendenciji preranog zapunjenja boje na velikim vrijednostima sive skale.

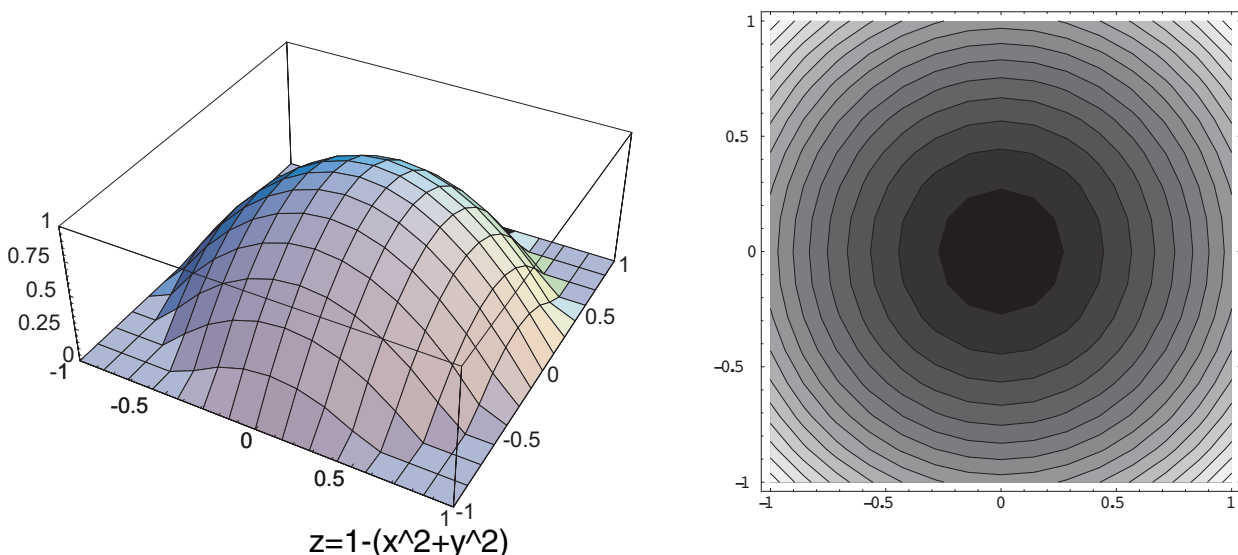
Kriterij minimiziranja efekta povećanja ispisne točkice zbog penetracije boje na papiru nije uzet u obzir prilikom eksperimentiranja jer bi se morao izvoditi na različitim ispisnim medijima i tehnologijama ispisa (vrsta boje i supstrata primjene). Taj kriterij nije posebno zanimljiv u ovom rješenju jer je cilj proizvesti potpuno nove oblike RE, a ne njihovu redukciju zbog vrste ispisnog medija i tehnologije. Za svaku tehnologiju ispisa posebno, zajedno u kombinacijama s raznim ispisnim medijima trebalo bi odabrati iz polja novih rasterskih oblika samo one koji imaju najmanji utjecaj za spomenuti neželjeni efekt.

4.1. Matematički model rasterskog oblika

Program Mathematica omogućava prikaz ispravnosti početnih prijedloga. Odluka o izboru RE donesena je nakon animacije rasterske ćelije. Mathematica omogućuje ispitivanje velikog broja oblika koji bi inače ostali samo kao idejni prijedlozi, nikada ni pokušani u primjeni. Zbog rubnih situacija deformacije rasterskog elementa, RE, koristimo dvodimenzionalan i trodimenzionalan prikaz.

Na slici 18 prikazan je model standardnog točkastog rastera. Funkcija $z=f(x,y)$ definira matematički model rasta RE. Plot3D [85] i ContourPlot [86] su funkcije programa Mathematica s kojima se definirao trodimenzionalan i dvodimenzionalan prikaz rasta pripadnog matematičkog modela RE. Prikazan je niz PostScript naredbi koje se ugrađuju kasnije u finalnu PostScript setscreen funkciju [87].

Površina zacrnjenja rasterskog elementa (RE) omeđena je u kvadratičnoj ćeliji [34]. Zbog toga nastaju izobličenja oblika već nakon prvog dodira s rubom



$$z=1-(x^2+y^2)$$

3D : `Plot3D [z,{x,-1,1},{y,-1,1},z={0,1}]`

2D : `ContourPlot[1-z,{x,-1,1},{y,-1,1},Contours=16]`

PostScript: `{dup mul exch dup mul add 1 exch sub}`

Slika 18
Model točkastog rastera

ograda i nastavljaju se do potpunog zacrnjenja. Taj događaj se, na primjer pojavljuje nakon 78% zacrnjenja kod okrugle točkice. Za ta proučavanja su najpogodniji 3D i 2D prikazi koji se jednostavno realiziraju programom Mathematica.

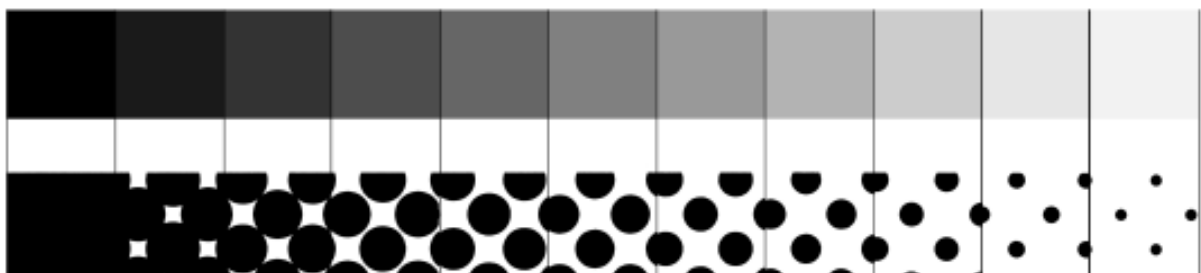
Mathematica omogućuje ispitivanje velikog broja oblika koji bi inače ostali samo kao idejni prijedlozi, nikada ni pokušani u primjeni. Kada se govori o individualizaciji oblika tada to znači da nam je na raspolaganju parametarski sustav definiranja RE: S jedne strane je to programirana numerička vrijednost kojom se djeluje na oblik (crtež) RE, a s druge strane je to mogućnost izbora dijelova matematičkih izraza.

4.1.1. Novi rasterski modeli

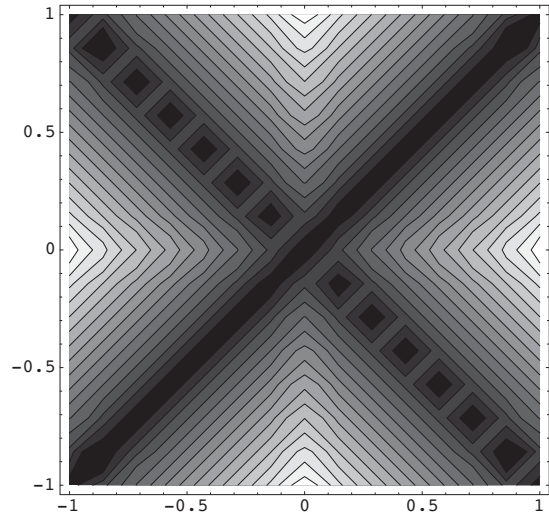
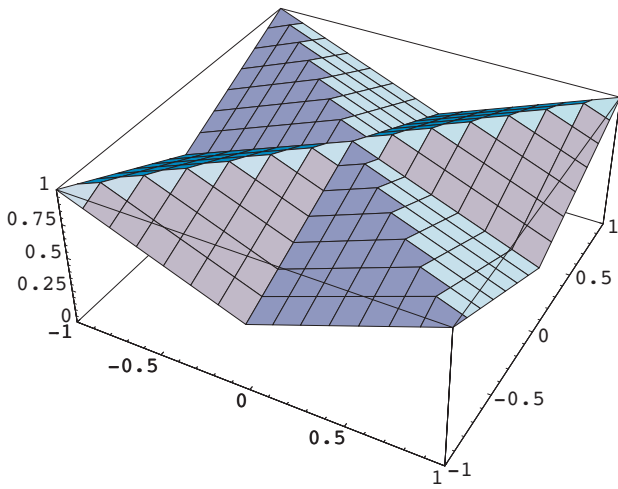
Prirast RE proučava se kvantitativno mjerenjem i kvalitativno psihološkim doživljavanjem zacrnjenja. Najbolji način za ocjenjivanje uspješnosti modela je ako novorazvijani rasterski model primjenimo u rasponu zacrnjenja od 0% do 100%.

Ta metoda prikazana je na slici 19 na primjeru standardnog točkastog rastera. RE se primjenjuje na slikovne elemente počevši od 5% zacrnjenja pa zatim od 10% do 100% s koracima po 10%. Ovdje je primjenjena gruba linijatura od 4 lpi radi boljeg pregleda razvoja rasta RE i kut od 45 stupnjeva koji se može eksperimentalno mijenjati.

Osnovna skupina originalno razvijenih modela oblika RE broji oko pedeset modela. Ovdje je na slikama 20, 21 i 22 prikazana skupina od 6 modela.

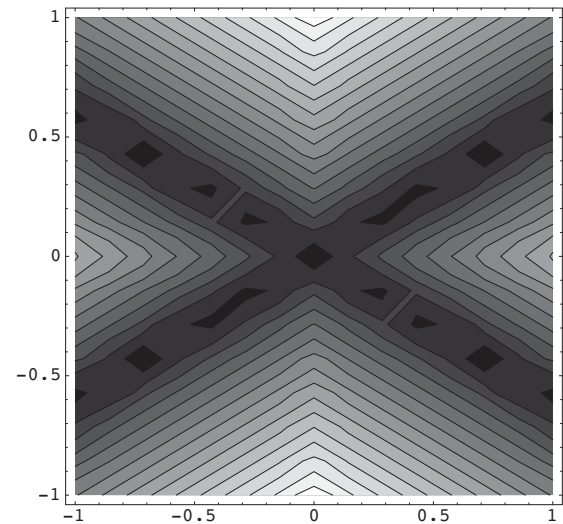
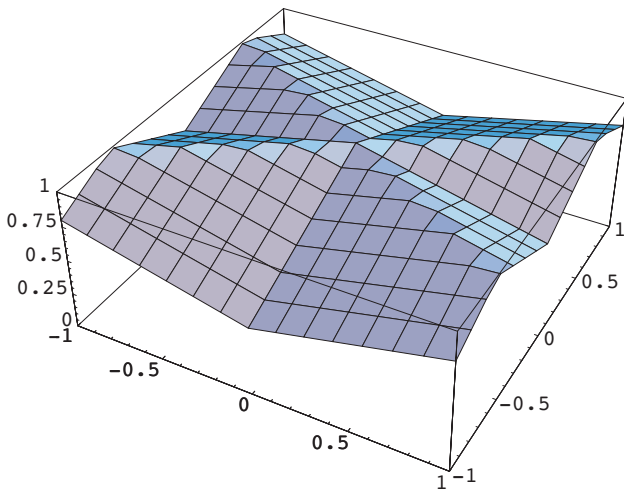


Slika 19
Test slikovni elementi s točkastim RE



$$z=1-\text{Abs}[(\text{Abs}[x]-\text{Abs}[y])/2]$$

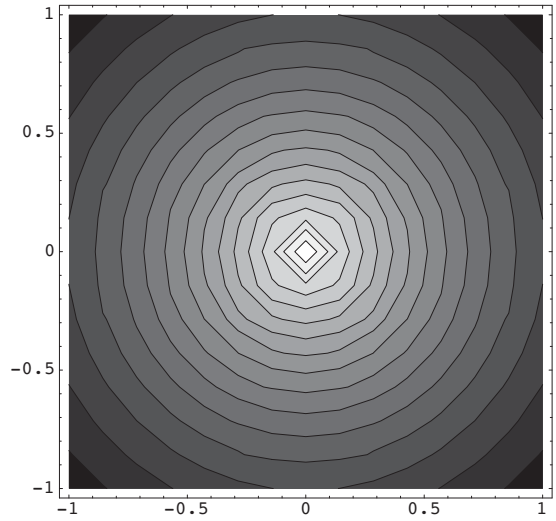
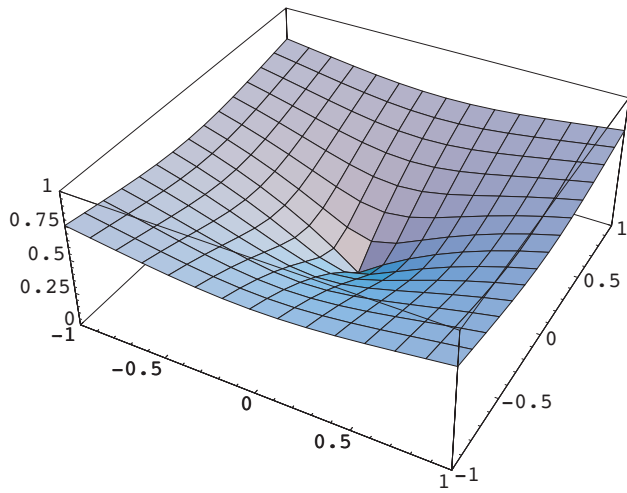
3D : `Plot3D [z,{x,-1,1},{y,-1,1},z={0,1}]`
2D : `ContourPlot[1-z,{x,-1,1},{y,-1,1},Contours=16]`
PostScript: `{abs neg exch abs add 2 div abs 1 exch sub}`



$$z=1-\text{Abs}[(\text{Abs}[x]*0.6-\text{Abs}[y])/2]$$

3D : `Plot3D [z,{x,-1,1},{y,-1,1},z={0,1}]`
2D : `ContourPlot[1-z,{x,-1,1},{y,-1,1},Contours=16]`
PostScript: `{abs neg exch abs 0.6 mul add 2 div abs 1 exch sub}`

Slika 20
Modeli r1 i r2 RE

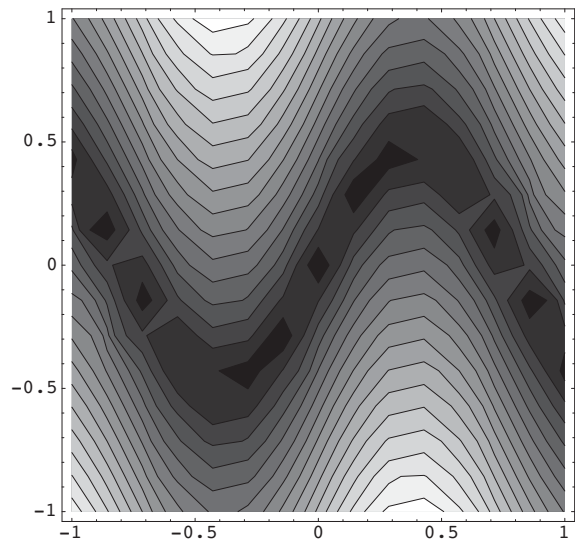
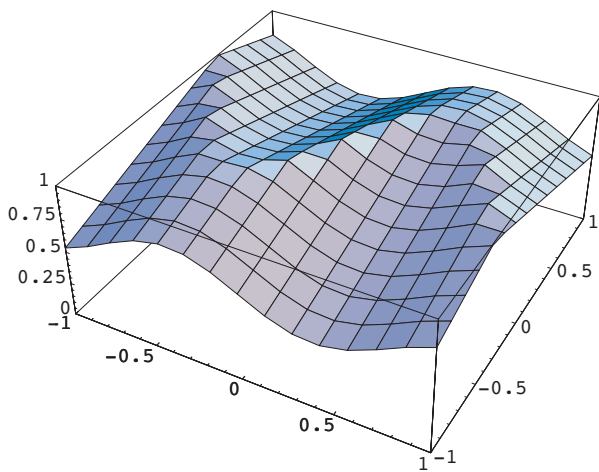


$$z=1-\text{Exp}[-\text{Sqrt}[x^2+y^2]]$$

3D : `Plot3D [z,{x,-1,1},{y,-1,1},z={0,1}]`

2D : `ContourPlot[1-z,{x,-1,1},{y,-1,1},Contours=16]`

PostScript: `{dup mul exch dup mul add sqrt neg e exch exp 1 exch sub}`



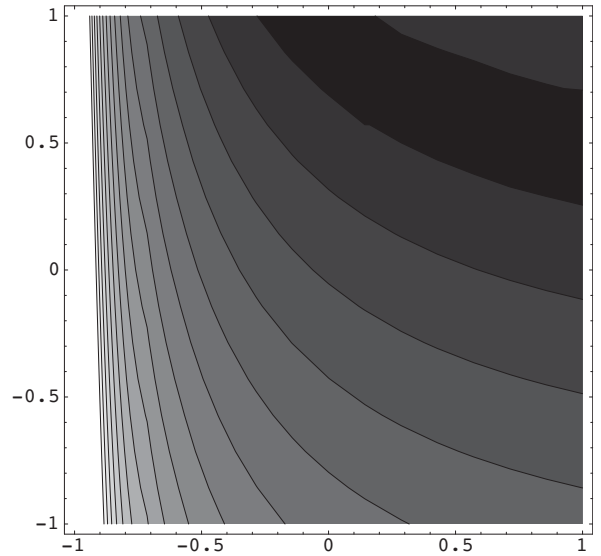
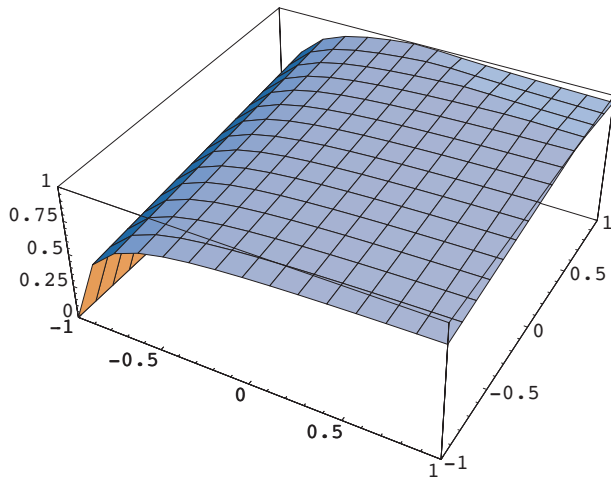
$$z=1-\text{Abs}[(\text{Sin}[180*x]/2-y)/3]$$

3D : `Plot3D [z,{x,-1,1},{y,-1,1},z={0,1}]`

2D : `ContourPlot[1-z,{x,-1,1},{y,-1,1},Contours=16]`

PostScript: `{neg exch 180 mul sin 2 div add 3 div abs 1 exch sub}`

Slika 21
Modeli r3 i r4 RE

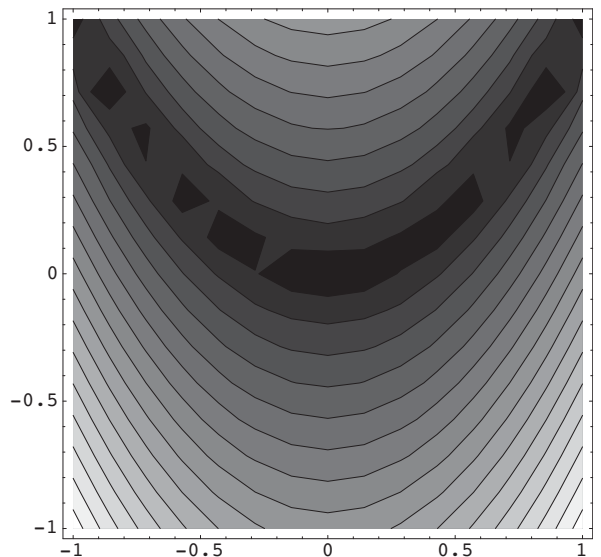
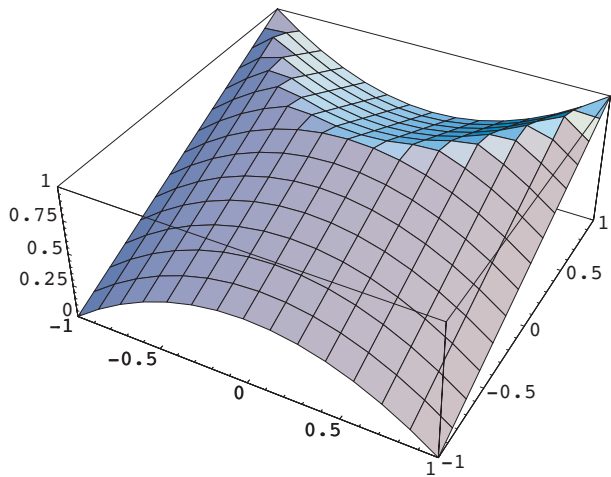


$$z=1-\text{Abs}[(1/(x+1.1)-y)/10]$$

3D : `Plot3D [z,{x,-1,1},{y,-1,1},z={0,1}]`

2D : `ContourPlot[1-z,{x,-1,1},{y,-1,1},Contours=16]`

PostScript: `{neg exch 1.1 add 1 exch div add 2 div abs 1 exch sub}`



$$z=1-\text{Abs}[(x^2-y)/2]$$

3D : `Plot3D [z,{x,-1,1},{y,-1,1},z={0,1}]`

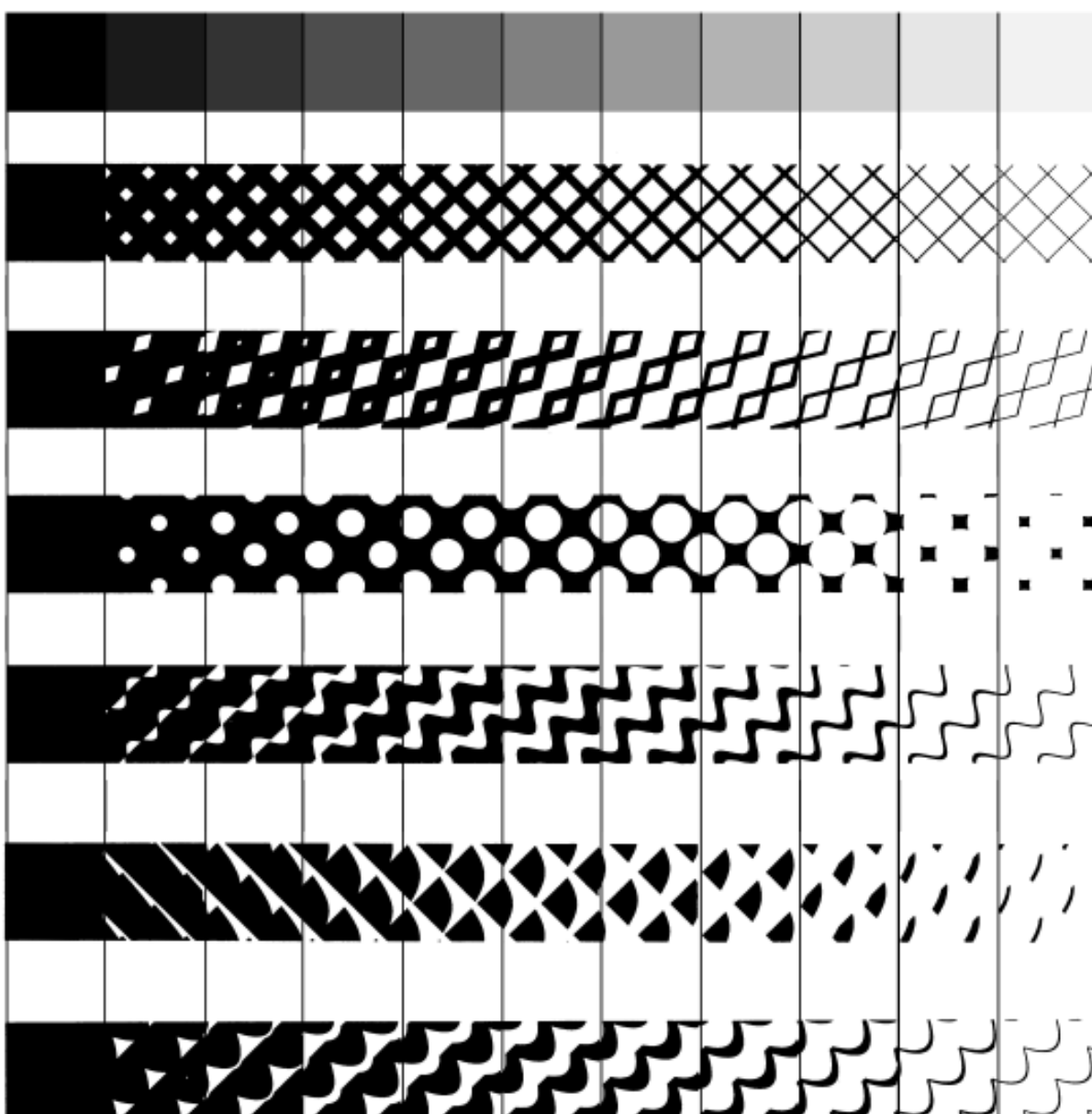
2D : `ContourPlot[1-z,{x,-1,1},{y,-1,1},Contours=16]`

PostScript: `{neg exch dup mul add 2 div abs 1 exch sub}`

Slika 22
Modeli r5 i r6 RE

Svaki model ima 3D i 2D prikaz rasta RE kao i pripadne matematičke i PostScript izraze opisane na prethodnom primjeru standardnog točkastog RE. Modelima su dati pripadne oznake od r1 do r6 radi lakšeg programskog referenciranja i kasnijeg pretraživanja.

Na slici 23 prikazana je primjena svih šest modela RE na test slikovne elemente. Primjenjena je linijatura od 4 lpi i kut od 45 stupnjeva. Pripadni



Slika 23
Test slikovni elementi s primjenjenih šest novih RE (4 lpi)

```
1 /r1 {abs neg exch abs add 2 div abs 1 exch sub} bind def
2 /r2 {abs neg exch abs 0.6 mul add 2 div abs 1 exch sub} bind def
3 /r3 {dup mul exch dup mul add sqrt neg 2.71828 exch exp 1 exch sub} bind def
4 /r4 {neg exch 180 mul sin 2 div add 3 div abs 1 exch sub} bind def
5 /r5 {neg exch 1.1 add 1 exch div add 10 div abs 1 exch sub} bind def
6 /r6 {neg exch dup mul add 2 div abs 1 exch sub} bind def
7 /L 4 def
8 /K 45 def
9 /slika1 <F2E5CCB2997F664C331900> def
10 /pixl 40 def
11 /dpixl 1 pixl div def
12 gsave
13 100 100 translate
14 L K {r5} bind setscreen
15 11 1 8 [dpixl 0 0 dpixl 0 0]
16 {slika1}image
17 grestore
18 showpage
```

Slika 24

PostScript program za testiranje novih RE

PostScript program na slici 24 ostvaruje takvo testiranje RE. Preko varijabli L i K se mijenja linijatura i kut pa se ubrzava eksperimentiranje s razvojnim modelom RE.

4.1.2. Opis PostScript rutina radi testiranja novih rasterskih modela

Instrukcije 1 do 6:

```
/r1 {abs neg exch abs add 2 div abs 1 exch sub} bind def
/r2 {abs neg exch abs 0.6 mul add 2 div abs 1 exch sub} bind def
/r3 {dup mul exch dup mul add sqrt neg 2.71828 exch exp 1 exch sub} bind def
/r4 {neg exch 180 mul sin 2 div add 3 div abs 1 exch sub} bind def
/r5 {neg exch 1.1 add 1 exch div add 10 div abs 1 exch sub} bind def
/r6 {neg exch dup mul add 2 div abs 1 exch sub} bind def
```

definiraju PostScript procedure *r1* do *r6* za definiranje rasterskog modela. Svaka procedura je pod *bind* naredbom s kojom se vrši pohrana algoritma u priručnu memoriju s čime se ubrzava proces rastriranja.

Instrukcije 7 do 8:

```
/L 4 def
```

```
/K 45 def
```

definiraju varijable *L* i *K* za linijaturu (lpi) i kut (stupanj) respektivno za proces rastriranja.

Instrukcija 9:

```
/slika1 <F2E5CCB2997F664C331900> def
```

definira heksadecimalni slikovni niz s nazivom *slika1*. Svaki heksadecimalni par opisuje sivu razinu jednog slikovnog elementa [35].

Instrukcije 10 do 11:

```
/pixl 40 def
```

```
/dpixl 1 pixl div def
```

definiraju veličinu piksla 40x40 točaka.

Instrukcija 12:

```
gsave
```

sprema tekuće grafičko stanje na stog grafičkih stanja da bi se pri kraju programa mogla napraviti njegova restauracija za slijedeće programe koje slijede. S time se postiže modularnost PostScript koda.

Instrukcija 13:

```
100 100 translate
```

translatira koordinatni sustav 100 po x-osi i 100 po y-osi u odnosu na tekuću poziciju koordinatnog sustava.

Instrukcija 14:

L K {r5} bind setscreen

definira rastriranje po proceduri r5 s linijaturom L i kutem K. S naredbom *bind* se postiže privremena pohrana već jednom rastriranog sivog nivoa. Svaki put kada se ponovo preuzme u algoritam rastriranja već jednom rastrirani sivi nivo ne pristupa se ponovnoj izradi pripadnog rasterskog modela.

Instrukcije 15 i 16:

11 1 8 [dpxl 0 0 dpxl 0 0]

{slika1}image

ispisuju rastrirane slikovne elemente iz slikovnog niza *slika1*. Prva dva parametra definiraju broj kolona odnosno broj redaka slikovnih elemenata. Na taj način se stvara željeno matrično slaganje slikovnih elemenata. Treći parametar definira broj bitova po slikovnom elementu s čime se određuje broj mogućih sivih nivoa svakog slikovnog elementa ($2^8=256$). Transformacijsko polje [dpxl 0 0 dpxl 0 0] definira sliku čiji su slikovni elementi veliki $1/dpxl * 1/dpxl$ točkaka bez transformacije [36]. Naredba *image* pokreće slaganje slikovnih elemenata. Prije njenog aktiviranja na operativnom stogu moraju postojati svi ovdje spomenuti entiteti [35].

Instrukcija 17:

grestore

restaurira grafičko stanje sa stoga grafičkih stanja.

Instrukcija 18:

showpage

pokreće ispis na ispisnom mediju.

4.2. Implementacija u sučeljima za obradu slikovnih elemenata i rastera kao stohastičko rastriranje

Parametri stohastičkog rastriranja su: slučajni izbor linijature unutar zadanih granica, slučajni izbor oblika (iz palete dozvoljenih oblika) za zadano zacrnjenje te kontrolirano programirani impuls generiranja slučajnih veličina.

Eksperimentiranje je već u prvim ispisima istog sadržaja (slike) s različitim (sjemenom) početnim generatorom slučajnog niza, otvorilo novo veliko neistraženo područje ovisnosti prirasta rasterske točke, linijature i oblika rasterskog elementa. Stohastička metoda daje na jednom mjestu mogućnost sagledavanja najrazličitijih parametara rasterske problematike bez zamornog mjerenja neciljanih situacija. Stohastički rastrirani uzorci nakon kratkog pregleda otiska nude smjer istraživanja neuralgičnih točaka, onih površina kojima treba dati veću pažnju. Budući da je sve to moguće simulirati u mnogim situacijama i ponavljati eksperimente neograničeno puta dolazimo do onih područja koja bi teško pronašli starim procedurama mjerenja, a koja počinju tek kada se izvrši tisak konvencionalnim metodama, metodama koje nisu dozvoljavale velik broj eksperimenata.

Eksperimenti su zasnovani na programskom rješenju dodjeljivanja rasterskog oblika svakom pikslu slike posebno i slučajno izabran. Da bi se to postiglo osim što su stvoreni matematički modeli koji realiziraju egzotične oblike rasterskih elemenata definirana je procedura koja će za svaki piksel pridružiti drugi oblik rastera i drugi rasterski kut. Stvoreni su programi u PostScriptu kojima se potpuno realiziraju nova rješenja individualizacije.

Ovdje se radi o pseudoslučajnim brojevima s poznatim sjemenom (seed) i procedurom generiranja. Zbog toga je moguće ponoviti niz slučajnih brojeva, a

time i realizaciju pseudoslučajne rastrirane slike. Određivanje sjemena preporuča se staviti u funkciju čvrstih podataka na dokumentu. Na primjer, ako se tu radi o osobnom dokumentu sa portretom, tada se sjeme može generirati iz podataka: datum rođenja i adresa. Ponavljanje i provjeru autentičnosti slike može napraviti samo onaj tko ima potpuni algoritam generatora slučajne varijable. Dati su originalni PostScript programi i procedure (rutine) kao baza mogućeg daljnjeg istraživanja stohastičkog izbora vrste rastera, linijature na jedinstvenoj slici, a za svaki piksel individualno rješenje.

4.2.1. Eksperimentalno istraživanje

Jednobojne slike portreta na dokumentima obično imaju 150 kB pa se isti rasterski oblik pojavljuje u prosjeku 3000 puta sa drugačijim zacrnjenjem, a to znači sa drugačijim oblikom u tisku. Na slici 25a, 25b, 25c i 25d prikazana su 4 različita modela i svaki je prikazan u 12 različitih zacrnjenja [21].

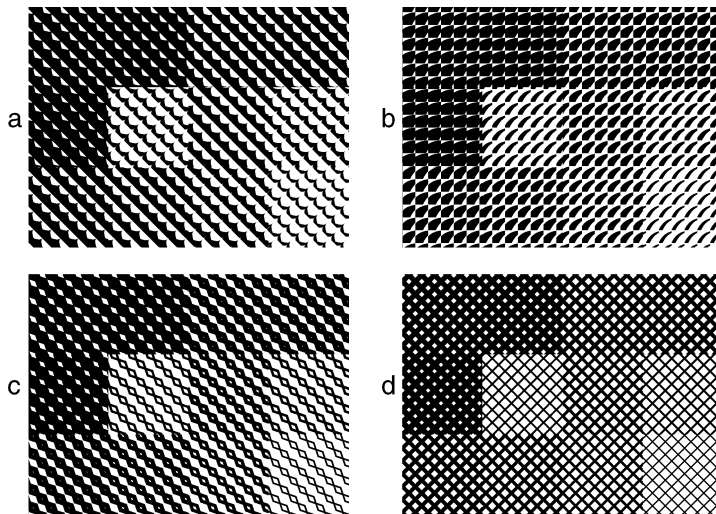
Pripadne matematičke relacije i PostScript naredbe su:

Model 1 (r7) $f(x,y) = 1 - \left| (\sqrt{|x|} - y) / 2 \right|$
{neg exch abs sqrt add 2 div abs 1 exch sub} setscreen

Model 2 (r8) $f(x,y) = 1 - \left| (\ln |x + 1.1| - y) / 3 \right|$
{neg exch 1.1 add abs ln add 3 div abs 1 exch sub} setscreen

Model 3 (r9) $f(x,y) = 1 - \left| (|x| 0.5 - |y|) / 2 \right|$
{abs neg exch abs 0.5 mul add 2 div abs 1 exch sub} setscreen

Model 4 (r10) $f(x,y) = 1 - \left| (|x| - |y|) / 2 \right|$
{abs neg exch abs add 2 div abs 1 exch sub} setscreen



Slika 25
Prikaz 4 modela na 12 različitih zacrnjenja

Svjetliji i tamniji slikovni elementi istog modela se znatno razlikuju pa se na taj način može dobiti impresija da se radi o RE iz različitih matematičkih modela (posebno vidljivo kod modela 2 i 3).

S ova 4 rasterska modela rješen je portret na slici 26. Slikovni element je povećan, njihov broj smanjen, a linijatura smanjena radi bolje prezentacije. Kompariramo četiri rješenja. Standardni raster s okruglim točkicama služi kao kontrolni uzorak (Slika 26a). Potpuna kontrola pravilnog rada različitih rastera koji se nižu jedan za drugim i potom ponavljaju daje "oblikovni moire" s periodom umnoška veličine piksela i broja različitih modela rastera u programu (Slika 26b). Moire nestaje ako se izbor rasterskog modela radi slučajno. Slučajno se još mogu birati linijatura (Slika 26c) te linijatura i kut zakretanja rastera (Slika 26d).

Implementacija u sučeljima za obradu slikovnih elemenata i rastera je realizirana preko PostScript programa prikazanih na slikama 27a, 27b, 27c, i 27d korespondiraju sa slikama 26a, 26b, 26c i 26d. Promjena vrijednosti sjemena (Slika 27c i 27d) dat će potpuno nov redoslijed rasterskih rješenja.



Slika 26 a) standardni točkasti raster
b) rastriranje sa cikličkim slijedom algoritama rastriranja
c) rastriranje sa stohastičkim odabirom algoritma oblika rasterskog elementa (SEED = 12345678), linijature i $\alpha = 0^\circ$
d) rastriranje sa stohastičkim odabirom algoritma rasterskog elementa, linijature i kuta rastriranja α (SEED = 246135)

PostScript rješenje uključuje i podatke o slici koji se inače u operativnom programu masovne primjene trebaju uzimati iz vanjske datoteke (niz "Grga" iz programa na slici 27a).

Slika 26b koristi isti slijed slikovnih elemenata, ali tako da se svakom pridružuje drugi algoritam rastera iz slijeda RE danih na slici 25. Treće rješenje uključuje generator slučajne varijable izbora algoritma oblika rasterskog elementa s kutom rastriranja α od 0° (slika 26c i 27c). Sjeme je postavljeno fiksno (SEED = 12345678). Programom na slici 27d generirana je slika 26d, ali sa drugim sjemenom (SEED = 246135) s time što je u tom primjeru ugrađen i slučajni izbor kuta rastriranja α . Dobiven je novi raspored pridruživanja oblika rasterskog elementa i rasterskog kuta istom slijedu piksla, odnosno za istu sliku.

Istraživanja s multirasterskim oblicima i stohastičkim izborom linijature i kuta može imati specifičnu primjenu na rješavanje portreta i maraka. Vibracije, izazvane različitim rješenjima rastriranja istog piksla osiguravaju neponovljivost i jaku zaštitu protiv svih tehnika skaniranja. Skaniranje takvog predloška će rezultirati Moire efektom [29]. To nije tehnika unutar kategorije tehnika zaštite s digitalnim vodenim znakom (eng. digital watermarking) niti digitalnog otiska prsta (eng. digital fingerprinting) [110] jer se ti algoritmi odvijaju na strani ulaznih slikovnih elemenata, a ne u dijelu ispisnih slikovnih elemenata.

Ocjenjivanju zacrnjenja, koje je kvantitativna veličina, dodaje se ocjena kontrasta i oštine kao vizuelni doživljaj. Zbog promjene linijature i oblika RE znatno se poremetio doživljaj slike. Sljedeći korak je ukrućivanje stohastičkih parametara. To je izvedba slike sa samo pojedinačnim oblicima rastera; zatim sa fiksnim linijaturama; ali sa istim sjemenom generatora slučajnih brojeva. Između ova dva krajnja eksperimentalna plana predlažu se simulacije postepenog oslobađanja stohastičkih parametara.

```
1 /grga <
2B372A52d6D04A005C4F162325222425211511151E25221F201F1D28362A8CE3D17C100E3316181C1519180C21312A111A20191B1D1A
223441C3C3B078410029180E0D201A1442757A774F222C1A1B1E1811162B4818110D0E110F0E09034B745B5F4E3B4A78722A211C1916
0E240F02A0E0C0B130F1A2310284C342E2E1E182D6252241A1919155E29050C0E0C0CF101027271E151114110B151013171B191718
0C0F0B0B0D0C0A0E1B23180A0B0F0B0A0D0F121C1A1C13191915140E0B0A0C0B0A070C131B12100E10110D0F161A1E2729221E1D2413
0E0C0A0C0A0D19080D11121F162027171010121D232027272B38270D0C0B0C062F5D0A0C11181E2524200E100F131D241B1B2633303A
0D0D0C0D0A1530330B161E4556493730221632343B2E212C383D410D0D0C0D0E0612221A2B37646E686D816458695E634C363A48473C
090F0F0D0D0B0903394F5671707A93ADB4B5AC98997C646F6C595E386320080E0B0A083A515A6B70859FBED0DADAD2CBB7B2B0996B6C
304F1E0A0E0C0B07264564556186ACC8DCE7E8E5DAD6D2C1A37760064554130B0D06061A375A4B5B8FB2C8D8E2E8EADDD6CEBD9E693B
0C17230E0B0E23061230545D648FB1C4CFDAE3E4D9D2CCBA9B5C300E0D0B0A0A1F5640202B63736B8EAAA9ACC4D1D3CAC6BAAA9C694D
0E0F0D0A0C53528D7A31698E9796705A7EA7C2CAB7957D7A83746A0E0E0D0C0B0C55A0813E71ABB088560072A4BDCE99520B44628477
0501000507011286BF5C61A6B5C2BFAAB5AFB1B8919D7197A8B68639656833144A083B935A52A3BBC7CBCDC8A6A2BC97AEC6C7CABB85
E5F3F4E2B6931A05586B4A9AB6CEDCE0D79DB6C4A5B4DACFC8AC81EBEBEBEDF06E00061C433C89B0CEDDD8CCBDBDB2ABBFE1DBBE9566
ECE2D2C0DCA50C0007032079A1B6B193B2C9D8BC5AFBDC9A86B57E5C08280B0BD854005030B499DAC9B694B64666A596492A27E483E
E7DCB97FB2D0EBEDBD49000B51A3B2A2404E76483483A6926B2A07DFC3A3B8CBD9E2EEFAED7905073282BD9C7B8483A3BAA7E32080E
A09F91928BC3D3E9F3F0F9890804104BBBCC4C2B3824C1C080B0F91516287A6CDECD FDEE8F0F57B060600318BB3AB70260608C0D0C
5E62777F96DDFAEE8E8E6E9F1EF750A08050A1E20315E0E0A0F0B23436C5F527DBAE5F4F9FFFC EEDFE683040A0906040F5711090B0FA7
544F49585B4067838B8C4EAF2E0B51307090807031F100B0818D05B6097A37332395155514C5988D7D72B040B0B0B0A0B0D0E080EAF
3651999DC3781C2D415D523D37458132010A0B0D0B0A0B0A0A004456606D80471807030F7A886E565B534E1E050A0D0B0A090B0A0143
2A2E3324040509080221836657545B726C21070C0B0B0A0A0B005102010003080A0909090034C04286987260781C090C0C0B0A0B0523
080708090809090908056C9BBB85897C46210B0B0B090B0A05180809080708080808090A090625728E7388605F1E090A090B0B1B55
> def
2 162 240 scale
3 27 40 8 [27 0 0 40 neg 0 40]
4 {grga} image
5 showpage
```

```
b)
1 /ros [ {r7} {r8} {r9} {r10} ] def
2 /L [ 10 15 20 25 ] def
3 /K [ 0 15 45 75 ] def
4 /m 0 def
5 /pixl 6 def
6 /dpixl 1 pixl div def
7 gsave 0 39 pixl mul translate
8 0 1 39 {/k exch def /i k 27 mul def
9 gsave
10 i 1 i 26 add {/j exch def
11 /LIN L m get def /KUT K m get def
12 LIN KUT ros m get bind setscreen
13 1 1 8 [dpixl 0 0 dpixl 0 0]
14 {grga j 1 getinterval} image
15 pixl 0 translate /m m 1 add def
16 m 3 gt {/m 0 def} if
17 } for
18 grestore 0 pixl neg translate
19 } for grestore showpage
```

Slika 27 a) program za standardni točkasti raster
b) program sa cikličkim slijedom algoritama rastriranja

```
c) 1 /ros [ {r7} {r8} {r9} {r10} ] def
2 12345678 srand
3 /m {2 31 exp 1 sub } def
4 /rn {rand m div} def
5 /L {10 20 rn mul add } def %sluc broj [10,30]
6 /R {3 rn mul} def %real sluc broj [0,3]
7 /IR {R round cvi } def %cjeli sluc broj [0,3]
8 /pixl 6 def
9 /dpixl 1 pixl div def
10 gsave 0 39 pixl mul translate
11 0 1 39 {/k exch def /i k 27 mul def
12 gsave
13 i 1 i 26 add {/j exch def
14 L 0 ros IR get bind setscreen
15 1 1 8 [dpixl 0 0 dpixl 0 0]
16 {grga j 1 getinterval} image
17 pixl 0 translate
18 } for
19 grestore 0 pixl neg translate
20 } for grestore showpage
```

```
d) 1 /ros [ {r7} {r8} {r9} {r10} ] def
2 246135 srand
3 /m {2 31 exp 1 sub } def
4 /rn {rand m div} def
5 /L {10 20 rn mul add } def %sluc broj [10,30]
6 /K {0 90 rn mul add } def %sluc broj [0,90]
7 /R {3 rn mul} def %real sluc broj [0,3]
8 /IR {R round cvi } def %cjeli sluc broj [0,3]
9 /pixl 6 def
10 /dpixl 1 pixl div def
11 gsave 0 39 pixl mul translate
12 0 1 39 {/k exch def /i k 27 mul def
13 gsave
14 i 1 i 26 add {/j exch def
15 L K ros IR get bind setscreen
16 1 1 8 [dpixl 0 0 dpixl 0 0]
17 {grga j 1 getinterval} image
18 pixl 0 translate
19 } for
20 grestore 0 pixl neg translate
21 } for grestore showpage
```

Slika 27 c) program sa stohastičkim odabirom algoritma oblika rasterskog elementa (SEED = 12345678), linijature i $\alpha = 0^\circ$
d) program sa stohastičkim odabirom algoritma rasterskog elementa, linijature i kuta rastriranja α (SEED = 246135)

U ovom disertaciji predlaže se potpuna kontrola preko numeričke vrijednosti sjemena (seed) i algoritma. Time je u potpunosti onemogućeno ponavljanje mikrostrukture reprodukcije ako se nema uvid u algoritam i vrijednost inicijalnog slučajnog broja. S druge strane, svaka rasterska ćelija slučajno je oblikovana i ne postoji periodičnost niti ponavljanje oblika. Slučajni raspored je i kod FM rastera ali tamo se radi o nekontroliranom početku te o činjenici da su sve točkice jednake veličine. U ovom prijedlogu stohastičkog rastera postoji kut rasterskih točaka i različitost veličine rasterske točkice. Oni su različiti, odnosno imaju drugačije i nezavisno rješenje za svaki slikovni element ili slikovnu strukturu.

4.2.2. Opis PostScript rutina radi vođenja eksperimenata

a) program za standarni točkasti raster

Programski model na slici 27a ima samo 5 instrukcije. Prva instrukcija definira slikovni niz *grga*. Instrukcija *162 240 scale* skalira koordinatni sustav tako da se množi svaka x koordinata s 162, a svaka y koordinata s 240. Treća instrukcija *27 40 8 [27 0 0 40 neg 0 40]* stvara slikovnu matricu od 27 x 40 slikovnih elemenata s time da je svaki velik 1/27 x 1/40 točaka. Pošto se prije ove instrukcije napravilo skaliranje 162 x 240 ispisna veličina slikovnog elementa je 6x6 točaka. S *40 neg (-40)* dobilo se slaganje slikovnih elemenata čitave slike od gornjeg lijevog kuta i dalje u definiranoj matričnoj formi 27x40. Četvrta instrukcija *{grga} image* započinje uzimati iz slikovnog niza *grga* heksadecimalne parove koje prikazuje preko prethodno opisanog algoritma. Instrukcija *showpage* pokreće ispis na definirani medij. Stoga što ne postoji *setscreen* naredba rastriranje se izvodi s točkastim rasterskim modelom koji je inicijalni model rastriranja u PostScriptu.

b) program sa cikličkim slijedom algoritama rastriranja

Instrukcija 1: `/ros [{r7} {r8} {r9} {r10}] def`

stvara polje rasterskih procedura od *r7* do *r10* koje su definirane na slici 24.

Instrukcija 2: `/L [10 15 20 25] def`

definira polje *L* parametara linijature zadane u linijama po inču.

Instrukcija 3: `/K [0 15 45 75] def`

definira polje *K* parametara kuteva zadanih u stupnjevima.

Instrukcija 4: `/m 0 def`

inicijalizira varijablu *m* koja nam služi kao pokazivač pozicije u poljima *L* i *K*

Instrukcija 5: `/pixl 6 def`

definira veličinu piksla od 6x6 točaka.

Instrukcija 6: `/dpixl 1 pixl div def`

definira parametar *dpixl* koji se namješta kao $1/pixl$ za transformacijsko polje *image* naredbe da bi slikovni element bio stvarno dimenzija $pixl * pixl$

Instrukcija 7: `gsave 0 39 pixl mul translate`

prvo sprema tekuću poziciju koordinatnog sustava prije translacije. Nakon toga translacija se koordinatni sustav za (broj redaka - 1) po y-osi za pripremu pozicije ispisa prvog slikovnog elementa.

Instrukcija 8: `0 1 39 {/k exch def /i k 27 mul def`
predstavlja glavu vanjske *for* petlje koja se vrti točno koliko slika *grga* ima redaka. Varijabla *k* preuzima tekuću vrijednost brojača petlje. Varijabla *i* se puni sa $k \cdot 27$ jer nam ona služi kao pokazivač za dohvat heksadecimalnog para slijedećeg retka slike (u retku ima 27 slikovnih elemenata).

Instrukcija 9: `gsave`
sprema tekuće grafičko stanje zbog pamćenja trenutne pozicije početka retka

Instrukcija 10: `i 1 i 26 add {/j exch def`
predstavlja glavu unutarnje *for* petlje koja se vrti točno koliko slika *grga* ima stupaca. Varijabla *j* preuzima tekuću vrijednost brojača petlje odnosno tekuću poziciju unutar tekućeg retka.

Instrukcija 11: `/LIN L m get def /KUT K m get def`
puni varijable *LIN* i *KUT* s pozicije *m* iz polje *L* odnosno *K*.

Instrukcija 12: `LIN KUT ros m get bind setscreen`
definira rastriranje s linijaturom *LIN*, kutem *KUT* i rasterskom procedurom odabranom iz polja rasterskih modela *ros*.

Instrukcije 13 do 14:

```
1 1 8 [dpixl 0 0 dpixl 0 0]
{grga j 1 getinterval} image
```

definira sliku od jednog slikovnog elementa s 8 bitnim sivim nivoima. Bez deformacije slikovni element je kvadratičan `[dpixl 0 0 dpixl 0 0]`, *image* operator prikazuje slikovni element definiran heksadecimalnim parom na *j* poziciji duljine 1 (*j 1 getinterval*).

- Instrukcija 15: `pixl 0 translate /m m 1 add def`
nakon ispisa jednog slikovnog elementa vrši translaciju za širinu slikovnog elementa po x-osi. Zatim se povećava pokazivač polja za jedan.
- Instrukcija 16: `m 3 gt {/m 0 def} if`
definira *if* grananje. Ukoliko je pokazivač *m* premašio svoj maksimum (3) tada se ponovo resetira na 0.
- Instrukcija 17: `} for`
definira kraj unutrašnje petlje.
- Instrukcija 18: `grestore 0 pixl neg translate`
restaurira poziciju početka retka i vrši translaciju za širinu slikovnog elementa po y-osi (za slijedeći redak slike).
- Instrukcija 19: `} for grestore showpage`
definira kraj vanjske petlje, restaurira grafičko stanje prije ispisa slike i aktivira ispisni mehanizam.

c) program sa stohastičkim odabirom algoritma oblika rasterskog elementa (SEED = 12345678), linijature i $a = 0^\circ$

Instrukcija 1: `/ros [{r7} {r8} {r9} {r10}] def`
stvara polje rasterskih procedura od $r7$ do $r10$ koje su definirane na slici 24.

Instrukcija 2: `12345678 srand`
sjeme pseudoslučajnog generatora brojeva koji će sa naredbom `rand` stvarati slučajne brojeve.

Instrukcija 3: `/m {2 31 exp 1 sub } def`
definiranje parametra m s kojim ćemo djeliti pseudoslučajni interval u slijedećem koraku da bi stvarali slučajne brojeve unutar intervala $[0,1]$.

Instrukcija 4: `/rn {rand m div} def`
definira pseudoslučajni broj rn u intervalu $[0,1]$. Naredba `rand` daje slučajan broj iz intervala 0 do $2^{31} - 1$ iz generatora pseudoslučajnih brojeva. Kada se takav broj podjeli sa $2^{31} - 1$ dobiva se slučajan broj unutar intervala $[0,1]$.

Instrukcija 5: `/L {10 20 rn mul add } def`
definira slučajni broj L unutar intervala $[10,30]$. Tako se definirao stohastički odabir linijature (l_{pi}) u željenim granicama. PostScript kod `10 20 rn mul add` predstavlja aritmetički izraz $10 + (20 * rn)$ gdje je rn slučajni broj iz intervala $[0,1]$.

- Instrukcija 6: `/R {3 rn mul} def`
definira realni slučajni broj R unutar intervala $[0,3]$.
- Instrukcija 7: `/IR {R round cvi } def`
definira cijeli slučajni broj IR unutar intervala $[0,3]$. Naredba `cvi` konverira realni broj u cijeli broj (convert to integer) Tako se definirao stohastički odabir indeksa polja *ros* za odabir algoritma rastriranja koji mora biti cijeli broj.
- Instrukcija 8: `/pixl 6 def`
definira veličinu piksla od 6×6 točaka.
- Instrukcija 9: `/dpixl 1 pixl div def`
definira parametar *dpixl* koji se namješta kao $1/pixl$ za transformacijsko polje *image* naredbe da bi slikovni element bio dimenzija $pixl * pixl$
- Instrukcija 10: `gsave 0 39 pixl mul translate`
prvo sprema tekuću poziciju koordinatnog sustava prije translacije. Nakon toga se translira koordinatni sustav za (broj redaka - 1) po y-osi za pripremu pozicije ispisa prvog slikovnog elementa.
- Instrukcija 11: `0 1 39 {/k exch def /i k 27 mul def`
predstavlja glavu vanjske *for* petlje koja se vrti točno koliko slika *grga* ima redaka. Varijabla *k* preuzima tekuću vrijednost brojača petlje. Varijabla *i* se puni sa $k * 27$ jer nam ona služi kao pokazivač za dohvat heksadecimalnog para slijedećeg retka slike (u retku ima 27 slikovnih elemenata).

Instrukcija 12: `gsave`

sprema tekuće grafičko stanje zbog pamćenja trenutne pozicije početka retka

Instrukcija 13: `i 1 i 26 add {/j exch def`

predstavlja glavu unutarnje *for* petlje koja se vrti točno koliko slika *grga* ima stupaca. Varijabla *j* preuzima tekuću vrijednost brojača petlje odnosno tekuću poziciju unutar tekućeg retka.

Instrukcija 14: `L 0 ros IR get bind setscreen`

definira rastriranje s linijaturom *L*, kutem *0* stupnjeva i rasterskom procedurom stohastički odabranom iz polja rasterskih modela *ros*.

Instrukcije 15 do 16:

`1 1 8 [dpxl 0 0 dpxl 0 0]`

`{grga j 1 getinterval} image`

definira sliku od jednog slikovnog elementa s 8 bitnim sivim nivoima. Slikovni element je bez deformacije i kvadratičnog je oblika $[dpxl\ 0\ 0\ dpxl\ 0\ 0]$, a *image* operator prikazuje slikovni element definiran heksadecimalnim parom na *j* poziciji duljine 1 (*j* 1 *getinterval*).

Instrukcija 17: `pixl 0 translate`

nakon ispisa jednog slikovnog elementa vrši translaciju za širinu slikovnog elementa po x-osi.

Instrukcija 18: `} for`

definira kraj unutrašnje petlje.

Instrukcija 19: `grestore 0 pixl neg translate`

restaurira poziciju početka retka i vrši translaciju za širinu slikovnog elementa po y-osi (za slijedeći redak slike).

Instrukcija 20: `} for grestore showpage`

definira kraj vanjske petlje, restaurira grafičko stanje prije ispisa slike i aktivira ispisni mehanizam.

d) program sa stohastičkim odabirom algoritma rasterskog elementa, linijature i kuta rastriranja α (SEED = 246135)

Instrukcija 1: `/ros [{r7} {r8} {r9} {r10}] def`

stvara polje rasterskih procedura od $r7$ do $r10$ koje su definirane na slici 24.

Instrukcija 2: `246135 srand`

sjeme pseudoslučajnog generatora brojeva koji će sa naredbom `rand` stvarati slučajne brojeve.

Instrukcija 3: `/m {2 31 exp 1 sub } def`

definiranje parametra m s kojim ćemo djeliti pseudoslučajni interval u slijedećem koraku da bi stvarali slučajne brojeve unutar intervala $[0,1]$.

Instrukcija 4: `/rn {rand m div} def`

definira pseudoslučajni broj rn u intervalu $[0,1]$. Naredba `rand` daje slučajan broj iz intervala 0 do $2^{31} - 1$ iz generatora pseudoslučajnih brojeva. Kada se takav broj podjeli sa $2^{31} - 1$ dobiva se slučajan broj unutar intervala $[0,1]$.

- Instrukcija 5: `/L {10 20 rn mul add } def`
definira slučajni broj L unutar intervala $[10,30]$. Tako se definirao stohastički odabir linijature (l_{pi}) u željenim granicama. PostScript kod `10 20 rn mul add` predstavlja aritmetički izraz $10 + (20 * rn)$ gdje je rn slučajni broj iz intervala $[0,1]$.
- Instrukcija 6: `/K {0 90 rn mul add } def %sluc broj [0,90]`
definira slučajni broj K unutar intervala $[0,90]$. Tako se definirao stohastički odabir kuta (stupanj) u željenim granicama. PostScript kod `0 90 rn mul add` predstavlja aritmetički izraz $0 + (90 * rn)$ gdje je rn slučajni broj iz intervala $[0,1]$.
- Instrukcija 7: `/R {3 rn mul} def`
definira realni slučajni broj R unutar intervala $[0,3]$.
- Instrukcija 8: `/IR {R round cvi } def`
definira cijeli slučajni broj IR unutar intervala $[0,3]$. Naredba `cvi` konverira realni broj u cijeli broj (convert to integer) Tako se definirao stohastički odabir indeksa polja *ros* za odabir algoritma rastriranja koji mora biti cijeli broj.
- Instrukcija 9: `/pixl 6 def`
definira veličinu piksla od 6×6 točaka.
- Instrukcija 10: `/dpixl 1 pixl div def`
definira parametar *dpixl* koji se namješta kao $1/pixl$ za transformacijsko polje *image* naredbe da bi slikovni element bio dimenzija $pixl * pixl$

- Instrukcija 11: `gsave 0 39 pixl mul translate`
prvo sprema tekuću poziciju koordinatnog sustava prije translacije. Nakon toga se translira koordinatni sustav za (broj redaka - 1) po y-osi za pripremu pozicije ispisa prvog slikovnog elementa.
- Instrukcija 12: `0 1 39 {/k exch def /i k 27 mul def`
predstavlja glavu vanjske *for* petlje koja se vrti točno koliko slika *grga* ima redaka. Varijabla *k* preuzima tekuću vrijednost brojača petlje. Varijabla *i* se puni sa $k*27$ jer nam ona služi kao pokazivač za dohvat heksadecimalnog para slijedećeg retka slike (u retku ima 27 slikovnih elemenata).
- Instrukcija 13: `gsave`
sprema tekuće grafičko stanje zbog pamćenja trenutne pozicije početka retka
- Instrukcija 14: `i 1 i 26 add {/j exch def`
predstavlja glavu unutarnje *for* petlje koja se vrti točno koliko slika *grga* ima stupaca. Varijabla *j* preuzima tekuću vrijednost brojača petlje odnosno tekuću poziciju unutar tekućeg retka.
- Instrukcija 15: `L K ros IR get bind setscreen`
definira rastriranje s linijaturom *L*, kutem *K* stupnjeva i rasterskom procedurom stohastički odabranom iz polja rasterskih modela *ros*.

Instrukcije 16 do 17:

```
1 1 8 [dpixl 0 0 dpixl 0 0]
```

```
{grga j 1 getinterval} image
```

definira sliku od jednog slikovnog elementa s 8 bitnim sivim nivoima. Slikovni element je bez deformacije i kvadratičnog je oblika $[dpixl\ 0\ 0\ dpixl\ 0\ 0]$, a *image* operator prikazuje slikovni element definiran heksadecimalnim parom na *j* poziciji duljine 1 (*j 1 getinterval*).

Instrukcija 18: `pixl 0 translate`

nakon ispisa jednog slikovnog elementa vrši translaciju za širinu slikovnog elementa po x-osi.

Instrukcija 19: `} for`

definira kraj unutrašnje petlje.

Instrukcija 20: `grestore 0 pixl neg translate`

restaurira poziciju početka retka i vrši translaciju za širinu slikovnog elementa po y-osi (za slijedeći redak slike).

Instrukcija 21: `} for grestore showpage`

definira kraj vanjske petlje, restaurira grafičko stanje prije ispisa slike i aktivira ispisni mehanizam.

5. Teza razvoja prema modularnim hibridnim sustavima

Današnji konvencionalni pristup ispisivanja digitalnih informacija je metoda *computer to plate*. Tu digitalna faza završava direktnim ispisom na ofsetnu ploču [88][89]. Na taj način se memorija RIP-a odteretila bitmapiranih podataka koji su na primjer sadržavali slikovnu informaciju pozadine. Analognom fazom se ispisuje naklada istog sadržaja informacija. Takvi ispisi se mogu potom tretirati individualizirano preko digitalnog sustava kao druga digitalna faza. Budući da se u njemu isprepliću digitalni ulazi, analogni konvencionalni izlazi s finalom u digitalnoj individualizaciji, takav sustav nazivamo hibridnim sustavom.

Digitalna grafička tehnika spaja se s konvencionalnim tiskom u svim odjelima, integrirajući proizvodnju u kontinuiranu cjelinu. Podjela zanimanja nestaje jer grafičar s komandnog računala upravlja i tipografijom, i reprofotografijom pa tiskom i doradnim procesima [28]. Pojavila se potreba za brzim prilagođavanjem procedurama novog načina rada, potreba za širokim znanjima i samostalnim školovanjem pomoću programskih i sklopovskih trenanžera i simulatora [27].

Postavlja se teza da se digitalni sustavi sa sučeljima za obradu slikovnih elemenata i rastera trebaju razvijati prema modularnim hibridnim sustavima. Za različite zadatke rastriranja potrebne su različite konfiguracije s pripadnim programskim rješenjima i u kombinaciji s drugim tehnikama ispisnih tehnologija odnosno modularni digitalni i hibridni sustavi. To je potpuno novo neistraženo područje. Upravo zbog neistraženosti, u praksi se takvim zadacima prilazi jako oprezno uz uspješne i neuspješne pokušaje. Za njihovo optimalno projektiranje treba razviti modele i metode ocjenjivanja ispravnosti konfiguriranja pomoću simulacije na računalu [30].

Za organiziranje hibridnih linija treba angažirati simulaciju s matematičkim modelima kako bi se povećala znanja optimalnog konfiguriranja realnog sustava tiska. S digitalnim tiskom postoje već dobra iskustva u malim nakladama. Najveća prednost takvog tiska je u označavanju svakog otiska drugim podacima, to jest individualizaciji proizvoda. Budući da se na konvencionalni tisak ne može dodavati vlastiti tekst i slika svakom otisku posebno, digitalni tisak našao je primjenu i u velikim nakladama. Iskustvo iz malih naklada ne može se ekstrapolirati i primjeniti na visoke tiraže jer su kvarovi, popunjenja memorije s masovnim podacima i tehnički zastoji posve drugačiji. Eksperimentiranja na realnim konfiguracijama za visoke naklade su preskupa pa se upoznavanje procesa izučava preko stohastičkih modela.

5.1. Razlozi uvođenja modularnih hibridnih sustava

Mora postojati jak razlog da tiskari dozvole uvođenje digitalnih ispisnih tehnologija. Primjer je rješavanje individualizacija igara na sreću gdje se kombiniraju konvencionalne metode tiska i digitalne metode tj. koncept hibridnog tiskarstva. Tu se javljaju slijedeći problemi [15][19]:

1. Problem planiranja i projektiranja potrošnje boje (tonera) u digitalnom dijelu hibridne linije jer su individualni piktogrami igre na sreću različite pokrivenosti i učestalosti
2. Problem projektiranja broja i vrste digitalnih tiskarskih uređaja ovisno o nominalnoj brzini tiska, procesnoj snazi rastriranja PostScript programa za individualizaciju ako se radi o PostScript uređajima
3. Problem stohastičkog izbora znakova sa stohastičkim pozicioniranjem što mora biti programski strogo kontrolirano radi igre

4. Problem optimizacije izgleda piktograma zbog nominalne brzine ispisa ako se individualizacija vrši sa brzim linijskim matričnim printerima (veća bjelina daje brži ispis za razliku od PostScript printera gdje je to neovisno)

Zbog navedenih problema programeri imaju puno posla, a uz to se moraju usuglasiti zahtjevi dizajnera i iskorištenje tiskarskog stroja. Važno je znati što se programom automatizira dugoročno, a što ostaje za programiranje na zahtjev što daje elastičnost u rješavanju novih zadataka.

Nakon puno izdanja velikih naklada s individualizacijom, vidjelo se da 80% ukupne cijene čini toner. Dizajneri nisu smjeli dalje razvijati svoje želje već se moralo pristupiti redizajniranju znakova dobitaka odnosno minimalizaciji informacija s ciljem manje potrošnje boje. To se jedino moglo postići mjerenjem postotka zacrnjenja svake slike dobitka.

Potpuno ista problematika vrijedi i za individualizaciju brojačima i slovnim znakovima stohastičke prirode. Optimizira se upotreba fonta ovisno o veličini na ispisu. Tu se ne smije pretjerati s odabirom previše svjetle porodice fonta jer bi se informacija mogla krivo isčitati zbog povremenog izbijanja na ispisu. Nerješavanje i neoptimiziranje pokrivenosti površine tonerom bio je glavni razlog ne ulaženja digitalnog tiska u velike naklade. Moralo se prići razvijanju programa, mjerenja i postupaka optimiranja upotrebe znakovlja - vizualni doživljaj s jedne strane i cijena tonera s druge strane.

Kada se pišu programi u PostScriptu [48], za stohastički izbor znakova dobitaka i njihovog pozicioniranja na stranici mora se voditi briga o nekoliko stvari [22]:

- s koliko digitalnih nezavisnih uređaja vršimo tisak (to je bitno zbog distribucije stohastičkog polja po jedinici uz najvažniji zahtjev kod igara na sreću, a to je zabranjeno pojavljivanje dva ista talona);

- ako se radi o igri s istim dizajnom ofsetne podloge u više kola, treba pisati program tako da se potpuno razdvoji pozicioniranje individualnog tiska od stohastičkog odabira simbola igre.

Potreban broj i vrsta uređaja za digitalni tisak kod igara na sreću ovisi samo o roku isporuke posla tj. o njihovoj nominalnoj brzini. Svaka igra ima svoje nagrade, a to zahtjeva dizajniranje i izradu novih znakova, tj. fontova [90] i pisanje novih programa.

Jedan od razloga uvođenja digitalnih ispisnih tehnologija je i tisak knjiga po nekom natječaju, npr. školskog programa. Kod takvog posla dizajnirane i prelomljene stranice dolaze sa svih danas standardnih operacijskih sustava (Windows, Mac OS, Unix itd.) Tu se radi o velikom broju varijabilnih otisaka različitih naslova knjiga, a jedna knjiga je u nakladi od samo desetak komada. Zbog nekompatibilnosti tih sustava po pitanjima pozicije hrvatskih slova [91][92][93][94], upotreba različitih fontova, upotreba različitih programa za prijelom (MSWord [95], Adobe PageMaker [96], QuarkXPress [97], WordPerfect [98], Corel Ventura [99]), zbog nepravilnosti i pogrešaka u donešenim datotekama, mora se pristupiti projektiranju načina prijema tih podataka. Iako proizvođači digitalnih ispisnih uređaja isporučuju PostScript programska sučelja za skoro sve današnje operacijske sustave [100][101], nemoguće je raditi na poznati način nedirektnog (offline) generiranja PostScript zapisa jer je onemogućena prethodna ekranska kontrola vezana uz fontove, rotacije i orijentacije stranica. Uz te probleme treba rješavati organizaciju reda čekanja ispred RIP-a, reda čekanja na RIP-u i reda čekanja iza RIP-a (finalni bitmapirani zapis). To je posebno osjetljivo kod momenta odobravanja tiska gdje se mora tiskati samo onaj dio jedne knjige za koju je momentalno prisutna nadležna osoba (na primjer matematičar kod matematičkog dijela prijeloma udžbenika).

Neizostavna su i istraživanja primjene digitalnog tiskarstva u dnevnim novinama. Potaknuti smo događajem višednevnog štrajka distribucije-transporta

i prodaje novina. Takvi slučajevi su se prije istraživali metodom simulacije, ali kada nam se to stvarno desilo svi su ostali zarobljenici u "mreži nespremnosti" tiskarstva u informatičkom okruženju. Broj zahtjeva za pristup objavljenim PDF stranicama dnevnih novina na Internetu bio je veći od uobičajene otisnute naklade tih novina. Zbog protočnosti, objavljuju se selektivne stranice, stranice bez ilustracija i stranice po ključu rubrika. Tih dana u punom pogonu su bili svi digitalni strojevi, od jeftinih kućnih printera do skupih linija digitalnog ispisa. Ovaj živi primjer dao je mnoge podatke za istraživanja, simulacije svih tokova novinske proizvodnje, od novinara kao stvaratelja vijesti, urednika, grafičara, telekomunikacija pa do ozbiljnog uključivanja digitalnih tiskarskih sustava u kariku publiciranja visokih naklada.

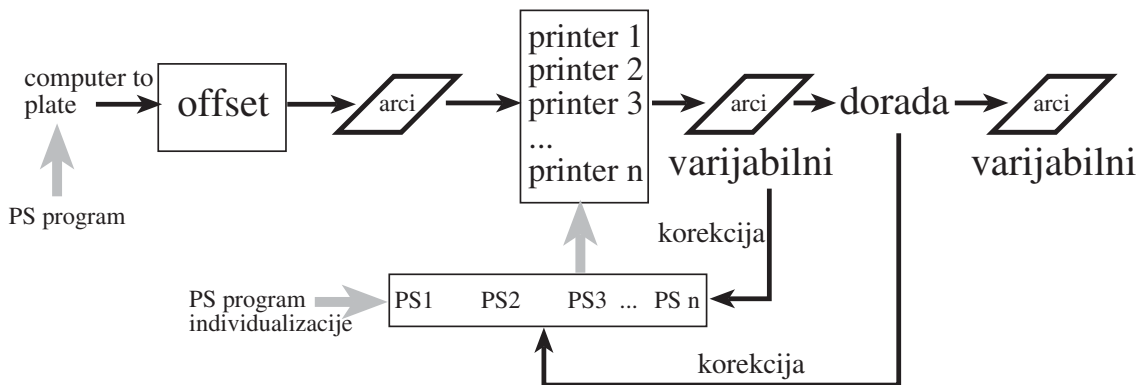
5.2. Konfiguracije hibridnih linija

Jedna od metoda izvedbe hibridne linije je nabavka sklopovskog i programskog sustava za posebnu namjenu fiksnom tehničkom metodom. Takvi sustavi puno su skuplji i njihov glavni nedostatak je nefleksibilnost i neprilagodljivost različitim poslovima.

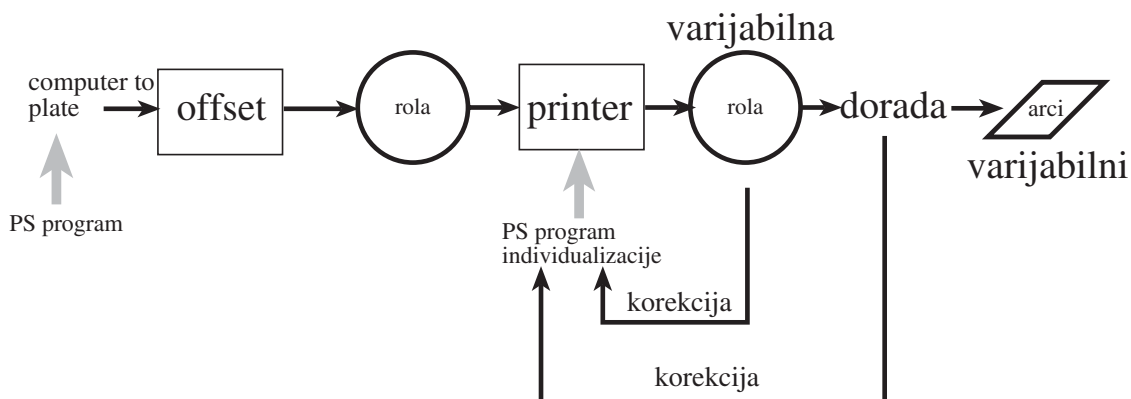
Na slici 28 i 29 prikazane su različite hibridne konfiguracije koje se mogu koristiti [24]. Ovisno o poslu odabire se konfiguracija hibridne linije. Konfiguracije 1, 2, 3 i 4 koriste offset za tisak istovjetnog dizajna, dok se kod konfiguracije 2 (rola u rola) za tu svrhu može koristiti i fleksografija. Konfiguracije 5 i 6 koriste Xeikon digitalni tisak s time da kod konfiguracije 5, Xeikon tiska samo nepromjenjeni dizajn, a kod konfiguracije 6 se nepromjenjena i varijabilna informacija tiska istovremeno. Obzirom na željenu namjenu može se postaviti na stotine hibridnih modela s neistraženom isplativošću i poštivanju rokova.

Svaka konfiguracija se treba zasebno optimizirati s obzirom na željenu nakladu i zadani rok. Optimizira se broj digitalnih uređaja s kojima se vrši individualizacija s obzirom na njihovu nominalnu propusnost, degradiranu propusnost i cijenu.

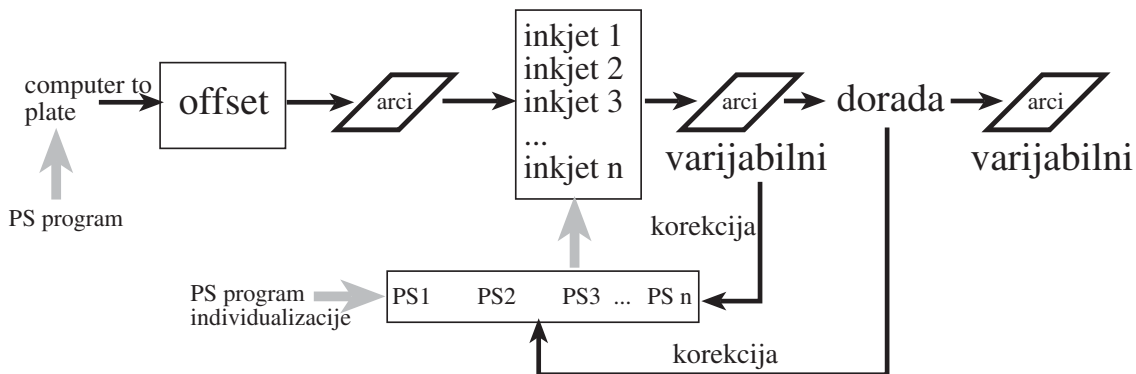
Model 1



Model 2

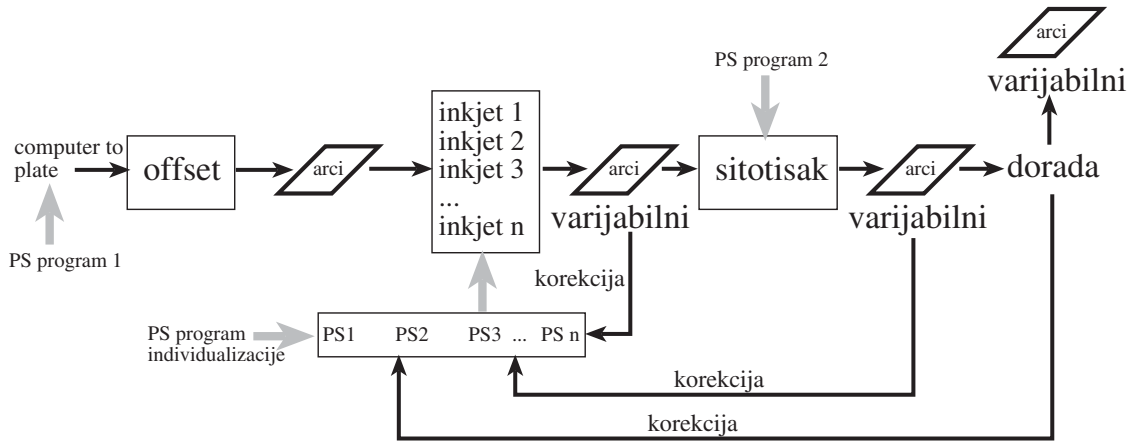


Model 3 (Individualizacija u boji)

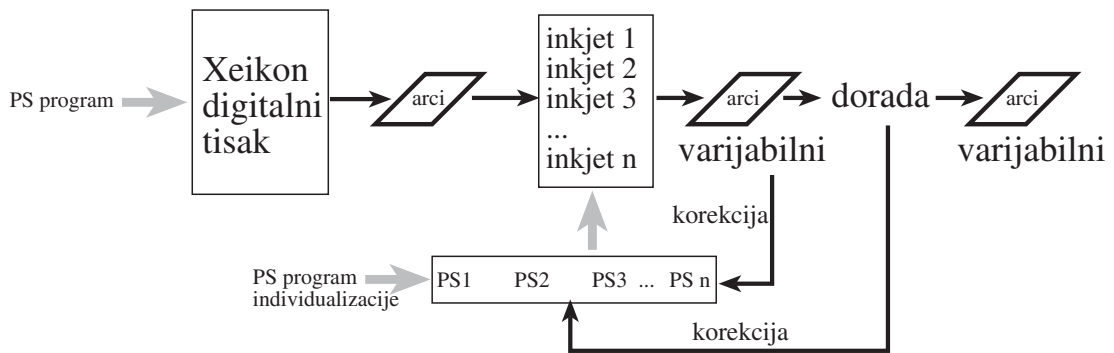


Slika 28
Hibridne konfiguracije

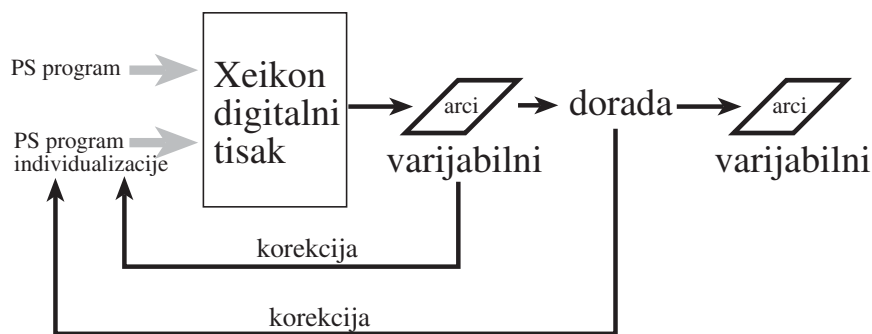
Model 4 (Individualizacija u boji)



Model 5 (Individualizacija u boji)



Model 6 (Individualizacija u boji)



Slika 29
Hibridne konfiguracije

6. Model hibridnog sustava za individualizaciju s novim rasterskim procedurama

Zadatak individualizacije s novim rasterskim procedurama je specifičan jer se za svaku vrstu posla mora model ponovo prekonfigurirati i ponovo reprogramirati sučelja za obradu slikovnih elemenata i rastera u PostScript jeziku. Zbog velike dinamike promjene cilja i rokova individualizacije, jedino se modeliranjem i simuliranjem mogu donositi pravilne odluke. Događa se da treba prekonfigurirati i reprogramirati sustav unutar jednog radnog dana. Treba napraviti takav model na kojem će se prije početka posla simulirati različite konfiguracije ovisno o zadanom roku i broju individualizacije. Na taj način ćemo dobiti fleksibilnost sustava. Takovi modeli bili bi radni modeli s kojima bi se eksperimentiralo svaki puta prilikom promjene zadatka individualizacije. Hibridna konfiguracija može uključivati i web sustave [32].

Ovdje se prikazuje stvaranje modela za ciljani zadatak od 300.000 varijabilnih talona formata A5. Svaki par talona čine formu formata A4 koja će se digitalnim ispisom jednostrano individualizirati tehnikom elektrofotografije. Nevarijabilni dio talona nazivamo masterom. U ovom slučaju master je dvostran i izrađuje se offsetnom tehnologijom preko CTP sustava i to sa 5 boja na prednjoj strani (C, M, Y, K, UV) i 4 boje na stražnjoj strani (C, M, Y, K). U ovom zadatku cijeli posao zajedno s varijabilnim dijelom mora završiti za 42 radna sata.

Napravit će se poseban model za izradu mastera offsetnom tehnologijom koji se može koristiti i proučavati nezavisno od čitavog sustava kao i modularni model za individualizaciju elektrofotografskom tehnologijom. To su sustavi različite propusnosti i različitog PostScript programiranja. Oba sustava koriste različite PostScript zapise kao svoje ulaze, jedan za obostrani master, a drugi za individualizaciju formi.

Prikazat će se izrada modela i simulacijsko eksperimentiranje kao primjer mikro i makro simulacije stohastičkog sistema tiskarskih procedura [31]. Prvo će se izraditi programski model sistema Computer to Plate (CTP) kao makrosistema koji će se mikrosimulacijama pojedinih modula CTP sustava dobro istražiti i optimirati jedan modul sustava.

Nakon simulacijskog eksperimentiranja i optimiranja, CTP sustav će se ugraditi u novi model analogno digitalnog sistema za izradu mastera. S tim makro modelom ćemo izvršiti mjerenja vremena izrade različitih naklada istog posla.

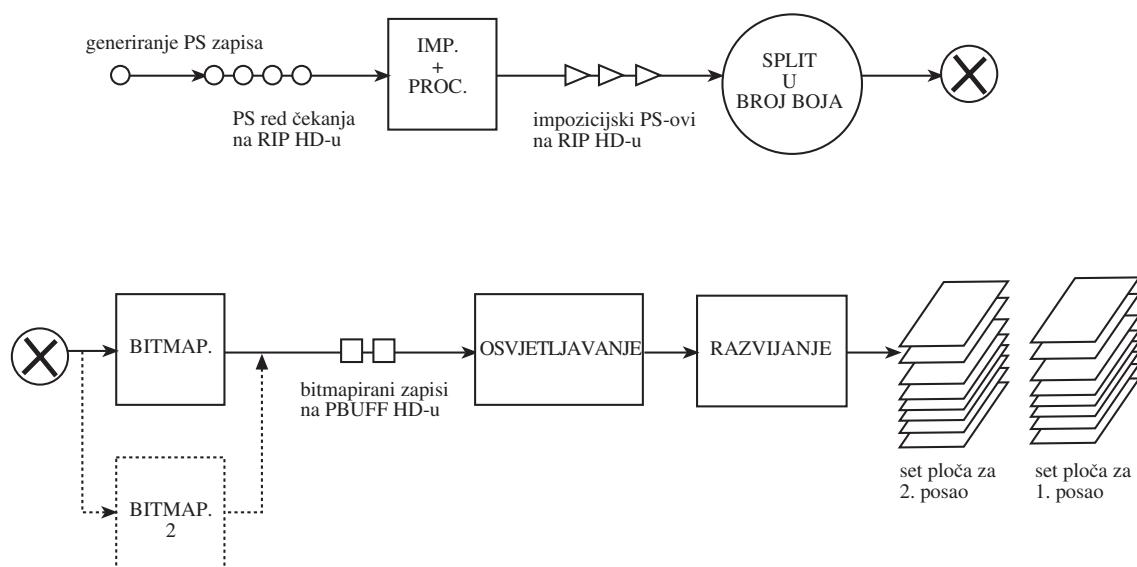
Kod CTP sustava će se dati primjer financijske razrade i odluke da li ići u nadogradnju sustava ili ne, što će ovisiti o tržištu u kojem sustav živi. Za simulaciju digitalno analognog sustava s CTP-om kao digitalnim, a ofsetom i doradom kao analognim dijelom, bit će bitni eksperimenti sa zadanim nakladama i mjerenjem vremena izvršavanja čitavog sustava radi pravilnih odluka o izvršavanju poslova u zadanom vremenu i na taj način pravilnog ugovaranja poslova od strane uprave na duži rok s fiksnim nakladama.

Sustav može servisirati poslove bez zagušenja ako mu je brzina servisiranja veća od brzine dolaženja poslova. Pošto u realnom sistemu dolazi do slučajnih poremećaja brzina dolaženja poslova, brzine servisiranja ili obojega, dolazi do zagušenja, odnosno do formiranja redova čekanja.

6.1. Model CTP sustava

CTP sustav se sastoji od tri modula: modul za impoziciju i procesiranje, modul za bitmapiranje te modul za osvjetljavanje i razvijanje. Rad modula za impoziciju i procesiranje se sastoji od faze prevođenja PostScript zapisa originalne PS datoteke u nižu objektnu PostScript razinu i faze ugradnje impozicijskih komandi za određeni izlazni zapis ofsetne ploče što je kasnije ulazni objekt za modul bitmapiranja.

Modul za bitmapiranje stvara bitmapirani zapis posla u zadanoj rezoluciji i linijaturi koji će se osvijetliti na ploči. Taj zapis jedino razumije sustav za osvjetljavanje i on se ostvaruje na velikim PageBuffer diskovima. Ove dvije faze zajedno nazivamo fazom RIP-inga koju smo namjerno rastavili na te mikromodule radi bolje izrade modela, istraživanja zastoja i moguće optimizacije.

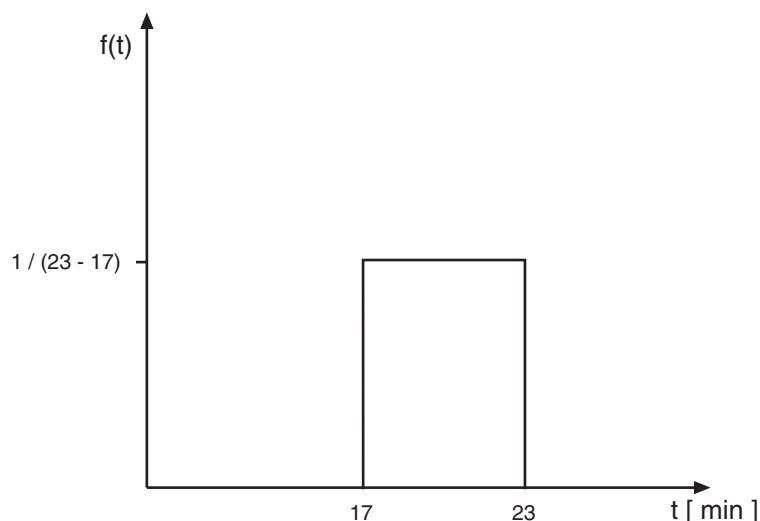


Slika 30
CTP sustav

Za proces osvjtljavanja i razvijanja ulaz je bitmapirani zapis. Faza osvjtljavanja je funkcijski ovisna o odabranoj rezoluciji i linijaturi kao i izlaznim formatom ofsetne ploče. Ovdje će se uzeti u obzir da se radi o B1 formatima ofsetnih ploča. Faza razvijanja ne ovisi o prethodno navedenim uvjetima već samo o vrsti ofsetne ploče.

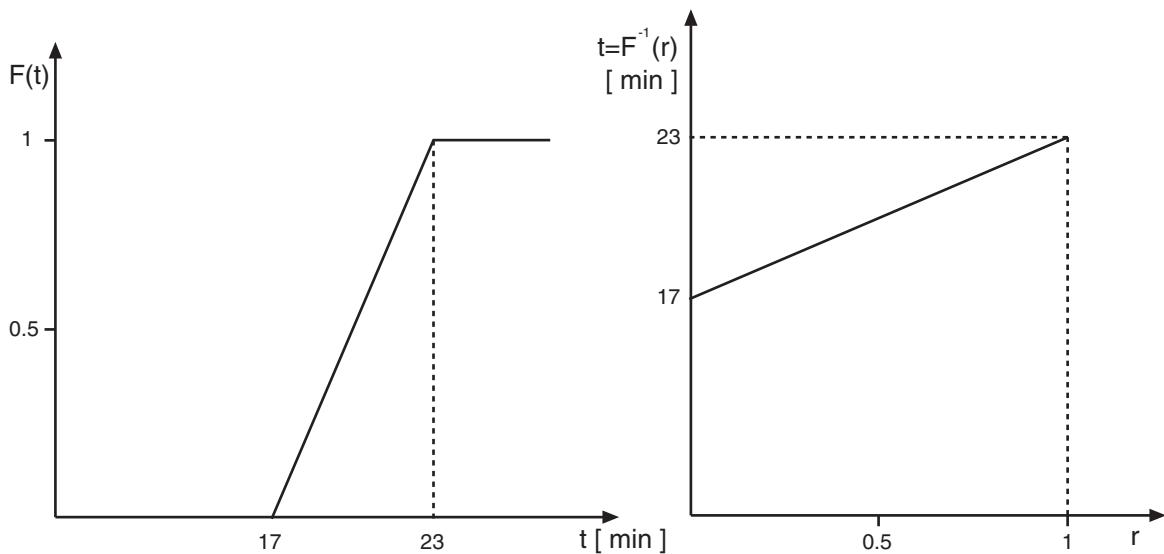
Na slici 30 prikazan je CTP sustav i pozicije formiranja redova čekanja. U sustav ulaze PostScript zapisi koji dolaze iz Internet okoline. Oni dolaze slučajno u model, odnosno dolazak sljedećeg je neovisan o prethodnom. To se najbolje simulira generiranjem transakcija eksponencijalnom distribucijom.

U model ulazi PostScript zapis koji sadrži izgled mastera prednje i stražnje strane. To je A4 PostScript obostrana digitalna forma. U fazi procesiranja stvara se lista grafičkih primitiva master forme dok se u fazi impozicije odvija umnožavanje PostScript formi u B1 format svake boje posebno. Mjerenjem vremena procesiranja i impozicije 64 tipičnih PostScript zapisa mastera dobila se funkcija vjerojatnosti [17] na slici 31.



Slika 31
Uniformna funkcija vjerojatnosti vremena procesiranja i impozicije CTP sustava

Za određivanje raspodjela stohastičkih varijabli koristit će se metoda “Monte Carlo”. Njena karakteristika je da se prvo mora generirati uniformno distribuirani slučajni niz brojeva te se pomoću njega generiraju slučajne vrijednosti bilo koje slučajne varijable. Zbog toga je potrebna kumulativna funkcija distribucije $t = F^{-1}(r)$ gdje je r slučajni broj s uniformnom raspodjelom od 0 do 1 (slika 32).

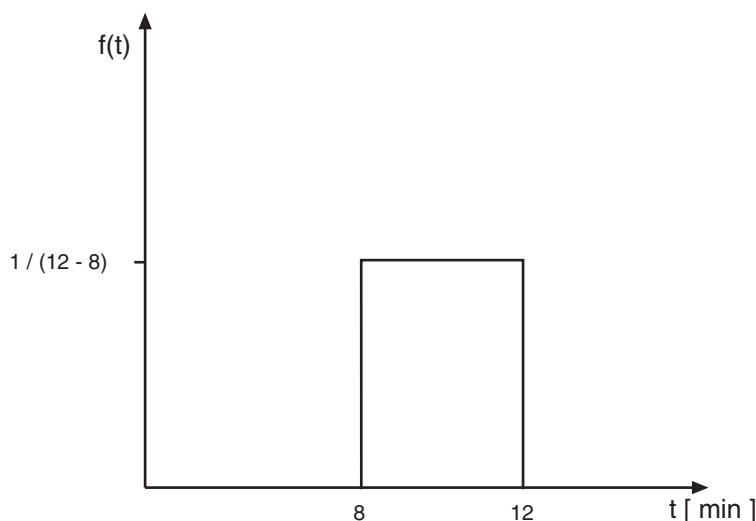


Slika 32
Kumulativna funkcija vremena procesiranja i impozicije CTP sustava

U daljnjem tekstu bit će prikazane samo funkcije vjerojatnosti umjesto pripadnih kumulativnih funkcija distribucije koje se izvode na prethodno prikazan način.

PS zapisi za B1 format mogu biti višebojni. Za svaku boju mora se formirati zasebna B1 bitmapa. Zato u modelu mora doći do razdvajanja PS zapisa u onoliko bitmapa koliko ima boja. U našem primjeru će se razmatrati posao s 9 boja, pet boja za tisak prednje strane i četiri boje za tisak stražnje strane. U tom slučaju modul za bitmapiranje mora 9 puta izvršiti servisiranje za jedan ulazni PS zapis. To se u modelu simulira splitting blokom gdje će se jedna transakcija pretvoriti u 8 transakcija tako da kroz facility bitmapiranje prolazi ukupno 9

transakcija. Vrijeme potrebno za bitmapiranje B1 forme mjereno je na 64 uzorka. Sva mjerenja su se odvijala u uvjetima rastriranja za izlaznu rezoluciju osvjetljavanja od 2400 dpi. Ta rezolucija pokriva cijeli raspon kvalitete osvjetljavanja ofsetne ploče u današnje vrijeme. Na slici 33 prikazana je funkcija vjerojatnosti vremena rastriranja podataka jedne boje za B1 osvjetljavanje. Svi podaci za jednu boju su spremljeni na tvrdim diskovima koji služe kao stranični međuspremnici (page buffer disks) prije faze fotoosvjetljavanja. Na taj način je omogućeno i uzastopno fotoosvjetljavanje istog sadržaja u nominalnoj brzini fotojedinice .



Slika 33
Uniformna funkcija vjerojatnosti vremena bitmapiranja jedne boje CTP sustava

Stvorene bitmape se spremaju na PageBuffer diskovima velikog kapaciteta. Čim se formira jedna bitmapa ona ide na proces osvjetljavanja, a zatim na online modul za razvijanje iz kojeg izlazi B1 ofsetna ploča za tisak jedne boje jednog posla. Jedan posao od 9 boja je pripremljen za tisak kada izađe na kraju 9 ofsetnih ploča. To nazivamo jednim skupom transakcija.

Faza fotoosvjetljavanja je definirana veličinom i rezolucijom ispisne forme. Vrijeme fotoosvjetljavanja za B1 veličinu ofsetne ploče pri rezoluciji od 2400 dpi

je uvijek isto i koristit će se kao konstanta u modelu. Za model koji se gradi koristit će se vrijeme fotoosvjetljavanja koje iznosi oko 2 minute po ploči, a može se mijenjati ovisno o vrsti fotojedinice i za analizu utjecaja na cijeli model.

Nakon fotoosvjetljavanja mora se ofsetna ploča kemigrafski razviti. Za model koji se gradi koristit će se vrijeme od oko 3 minute po B1 ploči nezavisno o fotoosvjetljenom sadržaju, a može se mijenjati ovisno o vrsti razvijачke jedinice.

Mjerenjima je utvđeno zajedničko vrijeme osvjetljavanja i razvijanja za B1 ploču 5+-1 minuta.

6.1.1. Gradnja programskog simulacijskog CTP modela

Eksperimentiranje na fizičkom modelu bilo bi preskupo jer se radi o skupom repromaterijalu i skupim servisnim jedinicama. Eksperimentiranje s programskim simulacijskim modelom treba omogućiti procjene isplativosti zamjene ili proširivanja postojećih resursa odnosno dinamičko konfiguriranje sustava ovisno o broju poslova i rokovima.

Iz mjerenja na fizičkom sustavu poznate su distribucije brzina CTP sustava koje su se koristile za izradu stohastičkog programskog modela. Na slici 34 prikazan je GPSS program koji predstavlja konfiguraciju s dva modula za bitmapiranje što je samo jedno stanje modela za proučavanje što će se objasniti u eksperimentalnom dijelu.

U model ulaze transakcije po eksponencijalnoj raspodjeli RVEXPO sa srednjom vrijednošću međudolazaka TM. Transakcija predstavlja ulazni PS zapis. Svakim startanjem modela učitava se varijabla TM da bi se eksperimentiralo s modelom pod različitim ulaznim brzinama dolaska poslova. Učitava se također varijabla BB što predstavlja broj boja jednog posla. Na taj

```
1      ****MODEL CTP (ComputerToPlate)  S DVA MODULA ZA BITMAPIRANJE ****
2
3      SIMULATE
4      INTEGER &N
5      INTEGER &BB,&B
6      REAL &TM
7      INITIAL LR(SPLITING)
8      LET &N=9000                      broj generiranja transakcija
9      1      BLET &B=0
10     STORAGE S(BITMAP),2
11     PUTPIC
12     0Unesi broj boja mastera (npr.9)
13     GETLIST &BB
14     PUTPIC
15     0Unesi srednje vrijeme medudolaska (min)
16     GETLIST &TM
17     2      GENERATE RVEXPO(1,&TM)
18     3      ADVANCE 0
19     4      QUEUE IMP
20     5      SEIZE IMP
21     6      DEPART IMP
22     7      ADVANCE 20,3              impozicija i procesiranje
23     8      RELEASE IMP
24
25     9      QUEUE SPLITING
26     10     GATE LR SPLITING          Da li je gotov set ploca?
27     11     DEPART SPLITING
28     12     LOGIC S SPLITING
29     13     SPLIT &BB-1,SKOK        stvaranje broja ploca koliko ima boja
30     14SKOK ENTER BITMAP
31     15     ADVANCE 10,2              bitmapiranje formata B1
32     16     LEAVE BITMAP
33     17     QUEUE OSVJ
34     18     SEIZE OSVJ
35     19     DEPART OSVJ
36     20     ADVANCE 5,1              osvjetljavanje i razvijanje formata B1
37     21     RELEASE OSVJ
38     22     BLET &B=&B+1
39     23     TEST E &B,&BB,KRAJ
40     24     BLET &B=0
41     25     LOGIC R SPLITING
42     26 KRAJ TERMINATE 1
43     START &N
44     END
```

Slika 34

GPSS model CTP sustava s dva modula za bitmapiranje

način se model može proučavati ovisno o različitom broju boja po poslu. Model ima tri Facilitya:

- IMP za impoziciju i procesiranje;
- BITMAP za bitmapiranje i
- OSVJ za osvjetljavanje i razvijanje.

Facility BITMAP se u modelu na slici 34 pretvorio u Storage sa 2 mjesta da bi se eksperimentiralo s virtualnim modelom od 2 modula za bitmapiranje. Vrijeme potrebno za servisiranje svakog od tih Facilitya odnosno Storagea je određeno u Advance blokovima.

Redovi čekanja koji su od interesa za proučavanje su formirani na tri mjesta:

- QUEUE IMP ispred faze impozicije i procesiranja koji se formira na RIP HD-u;
- QUEUE SPLITING ispred splitting faze stvaranja broja kopija ulazne transakcije prije bitmapiranja koji se formira na RIP HD-u i
- QUEUE OSVJ ispred faze osvjetljavanja i razvijanja koji se formira na PageBuffer disku.

Svaki ulazni PS na kraju modela mora proizvesti onoliko ofsetnih ploča koliko PS definira tiskarskih boja (varijabla BB). Imponirani PS ulazi u fazu bitmapiranja tek kada se završio prethodni set od BB ofsetnih ploča. To se regulira logičkom sklopkom SPLITING koja je postavljena upravo prije istoimene faze. Logička vrata GATE LR se otvaraju tek kada blok LOGIC R na

kraju resetira sklopku. To se događa tek kada brojač B dosegne vrijednost BB, odnosno kada je završen jedan set ofsetnih ploča. Na početku je logička sklopka postavljena u stanje RESET pa prva transakcija nesmetano ulazi kroz otvorena vrata u SPLIT blok.

Iza bloka GENERATE ugrađena je naredba ADVANCE 0 da se ne mijenja brzina dolazaka ulaznih transakcija ukoliko dođe do zagušenja ispred IMP facilitya što bi utjecalo na GENERATE blok tako što bi transakcija koja se treba slijedeća generirati stajala u GENERATE bloku onoliko dugo dok joj se ne oslobodi IMP. Na ovaj način transakcija stoji u ADVANCE bloku pa je model stalno pod istim opterećenjem što je važno za izlaznu statistiku i naše proučavanje modela.

Primjer modela na slici 34 projektiran je tako da iz njega izade 9000 transakcija što se može mijenjati varijablom N u START bloku. Na taj način model ima dovoljan broj transakcija za pravilno funkcioniranje svih stohastičkih varijabli modela. Ulazni broj transakcija nije jednak izlaznom broju transakcija zbog splittinga transakcija prije modula za bitmapiranje.

U modelu s dva modula za bitmapiranje je primjenjen STORAGE s dva mjesta. Na taj način transakcija ulazi u prvi slobodan bitmap modul (blok ENTER BITMAP) čime se omogućio paralelizam modula i veća propusnost servisiranja.

6.1.2. Opis rutina i veza radi vođenja eksperimenata

- Instrukcija 3: SIMULATE; izvršavanje programskog modela nakon uspješne kompilacije.
- Instrukcija 4: INTEGER &N; deklaracija cjelobrojne programske varijable &N s kojom se regulira broj izvođenja modela u bloku START radi pravilnog funkcioniranja svih stohastičkih varijabli modela.
- Instrukcija 5: INTEGER &BB,&B; deklaracija varijable &BB za broj boja jednog posla odnosno potrebnu količinu ofsetnih ploča; varijablom &B se broji momentalni broj osvjetljenih ofsetnih ploča
- Instrukcija 6: REAL &TM; deklaracija realne varijable &TM za definiranje srednje vrijednosti međudolazaka transakcije
- Instrukcije 7: INITIAL LR(SPLITING); postavljanje inicijalne vrijednosti logičkih vrata u stanje Reset (propusno stanje)
- Instrukcij 8: LET &N=9000; punjenje varijable &N; s njom se vrši optimalni odabir potrebnog broja izvođenja modela zbog potrebne stohastike naprema vremenu potrebnog za izvođenje
- Blok 1: BLET &B=0;Inicijalizacija varijable &B
- Instrukcija 10: STORAGE S(BITMAP),2; definiranje broja bitmap modula u modelu; s STORAGE-om se je dobila simulacija paralelnog rada tih modula

Instrukcije 11 do 13: PUTPIC; Ispisuje poruku "Unesi broj boja mastera (npr.9)" na ekran, a instrukcija GETLIST &BB puni varijablu &BB s tastature

Instrukcije 14 do 16: PUTPIC; Ispisuje poruku "Unesi srednje vrijeme medudolaska (min)" na ekran, a instrukcija GETLIST &TM puni varijablu &TM s tastature

Blok 2: GENERATE RVEXPO(1,&TM); generiranje transakcija u model se izvodi tako da je vrijeme međudolazaka definirano preko eksponencijalne raspodjele sa srednjom vrijednošću &TM

Blok 3: ADVANCE 0; Blok u kojem se transakcija ne zadržava ukoliko ima slobodan daljnji prolaz. Ako ne može dalje biti će u njemu zadržana kao i sve ostale koje naiđu u sličnu situaciju. Služi kao međuspremnik između GENERATE bloka i modula za Impoziciju.

Blokovi 4 do 8: QUEUE IMP
SEIZE IMP
DEPART IMP
ADVANCE 20,3
RELEASE IMP

definiraju put transakcije počevši od ulaza u red čekanja modula za impoziciju (QUEUE IMP) i njegovog zauzimanja (SEIZE IMP). Potom se registrira izlaz iz reda čekanja impozicije one transakcije koja je upravo zauzela modul za impoziciju i počinje teći vrijeme zadržavanja transakcije u njemu po uniformnoj raspodjeli 20+-3 minute. Tek nakon proteklog vremena može se osloboditi modul za impoziciju (RELEASE IMP).

Blokovi 9 do 13: QUEUE SPLITING
 GATE LR SPLITING
 DEPART SPLITING
 LOGIC S SPLITING
 SPLIT &BB-1,SKOK

reguliraju stvaranje separacija boja iz faze imponiranog PS zapisa koje bi se kasnije bitmapirale i osvijetlile. Impozicijski PS zapis kao transakcija ulazi u red čekanja za separaciju (QUEUE SPLITING). Logičkim vratima GATE LR SPLITING će se regulirati ulazak transakcija koje nadolaze. Ako je transakcija prošla logička vrata zabilježit će se njen izlazak iz reda čekanja (DEPART SPLITING) i postaviti logička vrata u SET stanje (LOGIC S SPLITING) da bi se zabranio ulazak slijedeće transakcije dok se tekuća ne izvrši. U bloku SPLIT &BB-1,SKOK stvara se &BB-1 klon transakcija kojih zajedno s originalnom transakcijom čini skup od &BB transakcija koje jedna po jedna putuju na blok sa SKOK oznakom.

Blokovi 14 do 16: SKOK ENTER BITMAP
 ADVANCE 10,2
 LEAVE BITMAP

simuliraju paralelni rad modula za bitmapiranje čiji je broj definiran s STORAGE S(BITMAP),2 u instrukciji 10 (u ovom modelu kao 2 paralelna bitmap modula). Kada prva transakcija uđe u ENTER blok zauzima se jedan bitmap modul i počinje u slijedećem bloku teći vrijeme zadržavanja transakcije po uniformnoj raspodjeli 10 ± 2 minute (ADVANCE 10,2). U ovom modelu može za to vrijeme paralelno ući još samo jedna transakcija koja će se paralelno zadržavati isto vrijeme. U bloku LEAVE BITMAP oslobađa se jedan bitmap modul pa se omogućava ulazak sljedeće transakcije u ENTER blok.

Blokovi 17 do 21: QUEUE OSVJ
 SEIZE OSVJ
 DEPART OSVJ
 ADVANCE 5,1
 RELEASE OSVJ

definiraju put transakcije počevši od ulaza u red čekanja modula za osvjetljavanje (QUEUE OSVJ) i njegovog zauzimanja (SEIZE OSVJ). Potom se registrira izlaz iz reda čekanja osvjetljavanja one transakcije koja je upravo zauzela modul za osvjetljavanje (DEPART OSVJ) i počinje teći vrijeme zadržavanja transakcije u njemu po uniformnoj raspodjeli 5+-1 minute. Tek nakon proteklog vremena može se osloboditi modul za osvjetljavanje (RELEASE OSVJ).

Blokovi 22 do 23: BLET &B=&B+1
 TEST E &B,&BB,KRAJ

testiraju da li su osvjetljene sve ploče koje zajedno čine jedan posao. Svaka transakcija svojim ulaskom povećava tekući brojač završenih ploča &B za 1. Nakon toga testira se da li je brojač dosegao zadanu &BB vrijednost. Ako je to istina, transakcija se prosljeđuje odmah slijedećem bloku iza TEST bloka, a inače se šalje u blok s oznakom KRAJ.

Blokovi 24 do 25: BLET &B=0
 LOGIC R SPLITING

pripremaju model za ulazak sljedećeg posla. Pošto se detektirao kraj prethodnog posla, brojač ploča &B se resetira na nulu. Također se resetira logička sklopka SPLITING pa će se otvoriti logička vrata u bloku 10 (GATE LR SPLITING) koja propušta transakcije samo ako je logička sklopka resetirana (upotrebljen atribut LR).

Blok 26: KRAJ TERMINATE 1

definira izlazak jednog posla iz modela. Blok TERMINATE reprezentira napuštanje jednog posla iz realnog sustava. Svaki ulazak u taj blok ujedno smanjuje vrijednost simulacijskog brojača terminacija definiran instrukcijom START. Kada brojač dosegne nulu ili manje, tada se zaustavlja simulacija.

Instrukcija 43: START &N

postavlja početnu vrijednost brojača terminiranih transakcija. S varijablom &N postignuta je mogućnost eksperimentiranja s različitim količinama transakcija. Cilj je da model ima dovoljan broj transakcija za pravilno funkcioniranje svih stohastičkih varijabli unutar modela.

Instrukcija 44: END

definira kraj GPSS programskog modela.

6.1.3. Eksperimentiranje s CTP modelom

Simulacijsko eksperimentiranje se koristi za istraživanje općih zavisnosti odziva i faktora simulacije. Ako se prosudi da je broj transakcija u redovima čekanja prevelik tada se trebaju modificirati faktori modela. To se može učiniti na tri načina: smanjivati brzinu dolaženja transakcija, povećati brzinu servisiranja pojedinih modula ili povećavanjem broja servera.

U ovom radu je prikazano eksperimentiranje s mijenjanjem faktora brzine dolaženja transakcija u model i eksperimentiranje s povećavanjem broja modula za bitmapiranje. Faktor brzina ima 10 kvantitativnih razina koje se dobiju mijenjanjem prosječnog vremena međudolaska u bloku GENERATE u programu. Kada je prosječno vrijeme međudolaska 480 minuta, tada je to jedna transakcija u 8 sati, a kada transakcija ulazi prosječno svakih 48 minuta tada je to 10 transakcija u 8 sati.

U jednom određenom eksperimentu postoji jedna kombinacija razina i to po jedna za svaki faktor. Svaka ta kombinacija određuje jednu varijantu modela. Za svaku varijantu snimat će se izlazne varijable odnosno reakcije varijante modela koje se dobiju u izlaznoj statistici simulacije sa programskim modelom i to:

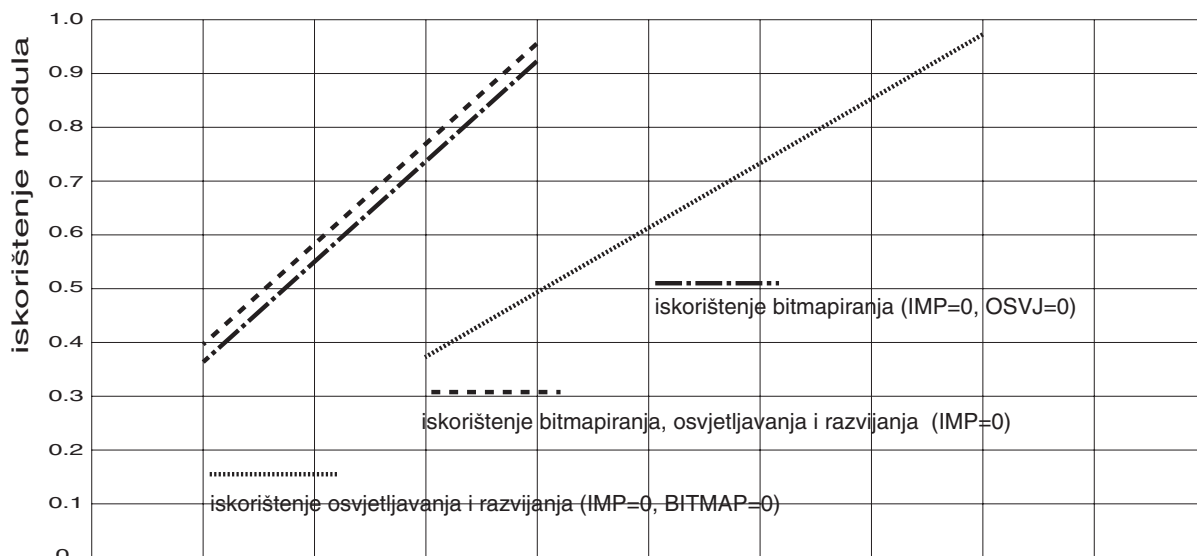
- a) Prosječno iskorištenje IMP;
parametar AVERAGE UTILIZATION u izlaznoj statistici
- b) Prosječno iskorištenje BITMAP;
parametar AVERAGE UTILIZATION u izlaznoj statistici
- c) Prosječno iskorištenje OSVJ;
parametar AVERAGE UTILIZATION u izlaznoj statistici

- d) Prosječno vrijeme čekanja IMP;
parametar \$AVER. TIME/TR u izlaznoj QUEUE statistici
- e) Prosječno vrijeme čekanja SPLITING;
parametar \$AVER. TIME/TR u QUEUE statistici
- f) Prosječno vrijeme čekanja OSVJ;
parametar \$AVERAGE TIME/TR u QUEUE statistici

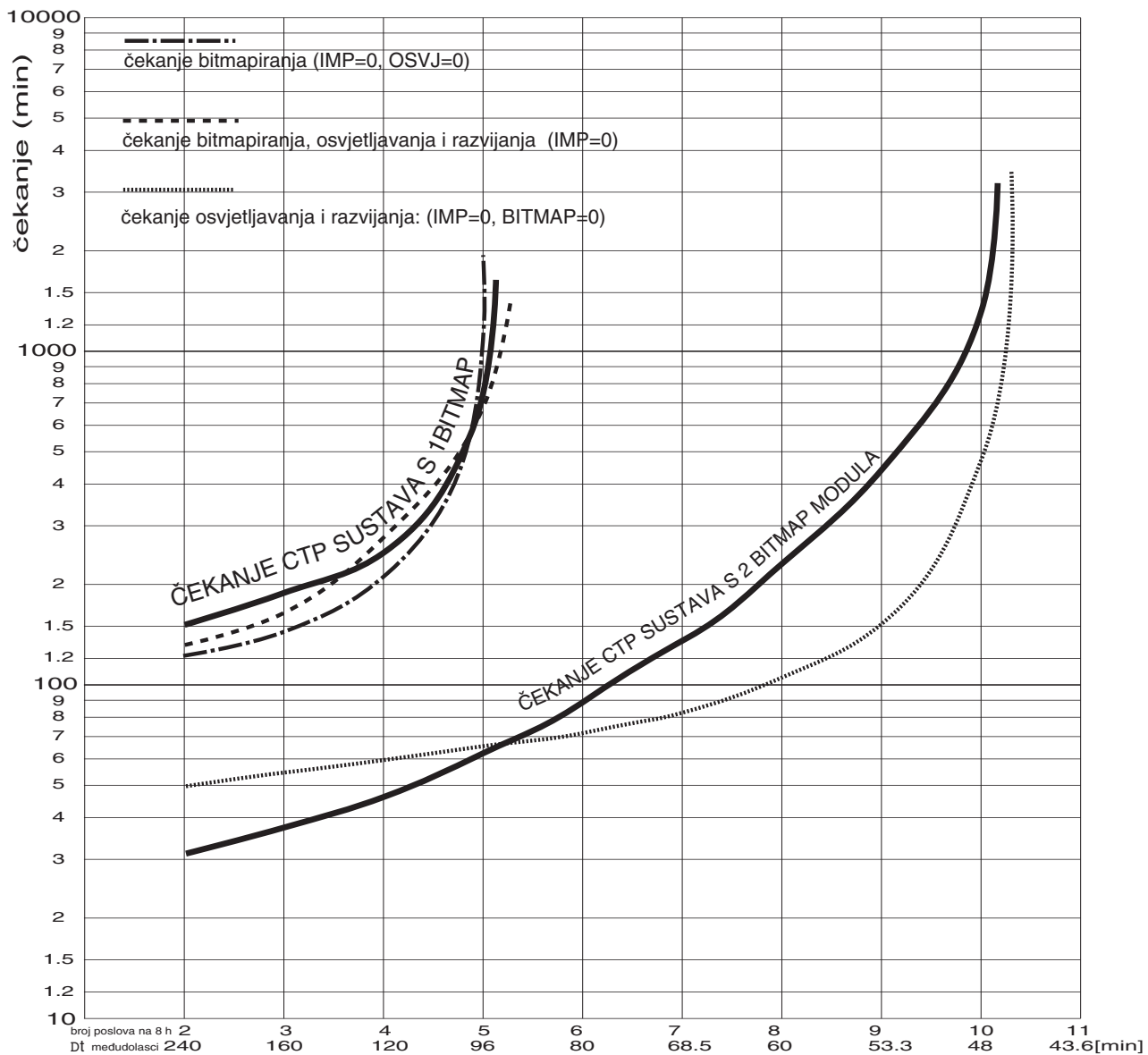
Svi rezultati eksperimentiranja prikazani su grafički, s time da su rezultati snimanja prosječnog vremena čekanja u logaritamskom mjerilu. Prvo se snima ovisnost prosječnog čekanja i iskoristivost modula za osvjetljavanje i razvijanje o broju ulaznih transakcija bez utjecaja drugih modula (IMP=0, BITMAP=0). Tako se može vidjeti s kojim brojem ulaznih transakcija modul za osvjetljavanje i razvijanje postaje usko grlo.

Snima se ovisnost čekanja i iskoristivost bitmapiranja o broju ulaznih transakcija isključivo samo zbog njega samog (IMP=0, OSVJ=0). Na taj način se vidi s kojim brojem ulaznih transakcija bitmapiranje postaje usko grlo. Tako se izvršilo snimanje i za IMP modul (BITMAP=0, OSVJ=0). To je stupnjevito simulacijsko eksperimentiranje s kojim se prvo doznaje propusnost samih modula, a znajući te zastoje snimaju se nadalje one varijante modela gdje se očekuje utjecaj zastoja pojedinog modula na čekanje i iskoristivost čitavog sistema. Odabirale su se samo one kombinacije i razine faktora koje su zanimljive za područje istraživanja broja ulaznih transakcija (do 11 u 8 sati) i za koje će iskoristivost biti između 70 i 80 %. Također se neće razmatrati kombinacije koje imaju čekanja preko 1000 minuta odnosno na putu su prema potpunom zastoju.

Na slici 35 prikazan je grafikon ovisnosti prosječnog čekanja u redu i iskorištenje modula o broju ulaznih poslova unutar 8 sati. Modul za



ZASTOJI CTP SUSTAVA



Slika 35

Ovisnosti prosječnog čekanja u redu i iskorištenje modula o broju poslova unutar 8 sati

osvjetljavanje i razvijanje ulazi u potpuni zastoј za više od 10 poslova na 8 sati, a najpovoljniji je za 7 do 8 poslova gdje mu je iskorištenje 70-80%.

Modul za bitmapiranje ulazi u potpuni zastoј kod 5 poslova na 8 sati , a najpovoljniji je za 4 poslova. On je najveće usko grlo u sistemu. Prikazan je također slučaj eksperimentiranja s modelom kojemu je otvoren samo kanal za impoziciju (IMP=0) označen s čekanje bitmapiranja i osvjetljavanja i razvijanja.

Na ovom grafikonu nisu prikazana eksperimentiranja samo s modulom za impoziciju i procesiranje jer su eksperimenti pokazali da taj modul ulazi u zagušenje tek kod 25 poslova na 8 sati tako da on nemože prouzročiti potpuni zastoј sistema pa izlazi iz daljnjeg razmatranja.

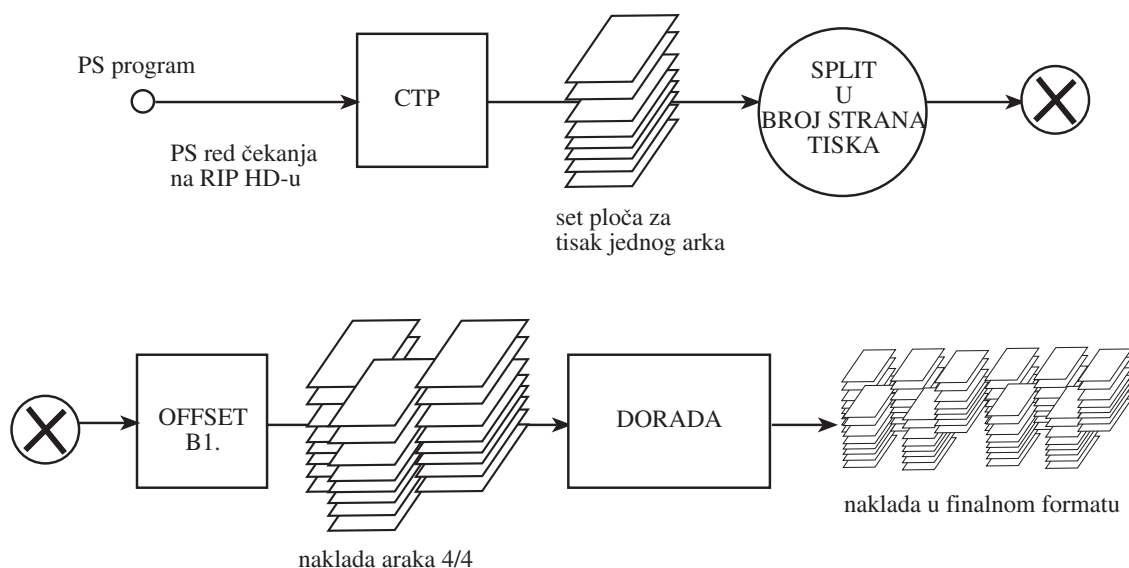
Na grafikonu su prikazane dvije krivulja čekanja CTP sustava, jedna s jednim bitmap modulom i druga s dva bitmap modula. To čekanje se mjerilo na QUEUE SPLITING odnosno prije procesa bitmapiranja i osvjetljavanja i razvijanja. Pošto impozicija ne utječe na propusnost sustava ova dva preostala modula su dovoljna za traženja optimalnog rješenja. Pošto je bitmap modul najveće usko grlo sistema napravila su se eksperimentiranja kao da postoje dva modula. U tom slučaju CTP sustav ulazi u potpuni zastoј tek kod 10 poslova na 8 sati. Ovo eksperimentiranje je pokazalo da je bilo dovoljno povećati modul za bitmapiranje, a ne zajedno s njim još jednu jedinicu za osvjetljavanje što je bilo upitno na početku.

Financijski gledano prelazak na CTP sustav s dva bitmap modula se isplati ako je umnožak vrijednosti poslova(P) i broja poslova (BP) na 8 sati veći od ulaganja (U) u novi bitmap modul za određeni period na koji bi se ulaganje odnosilo:

$$BP=480/TM \quad P \times BP > U$$

6.2. Model analogno digitalnog sustava za izradu mastera

Na slici 36 prikazan je model za izradu mastera. Analogni dio je offsetni stroj, a digitalni dio je sustav RIP-a i uređaja za fotoosvjetljavanje offsetnih ploča direktno iz računala (Computer To Plate tehnologija-CTP) koji je razmatran u prethodnim poglavljima. Taj model je ovisan o nakladi formi i broju boja otiska. Varijable naklade i broja boja utječu na propusnost offsetnog stroja, a samo varijabla broja boja utječe na propusnost CTP-a.



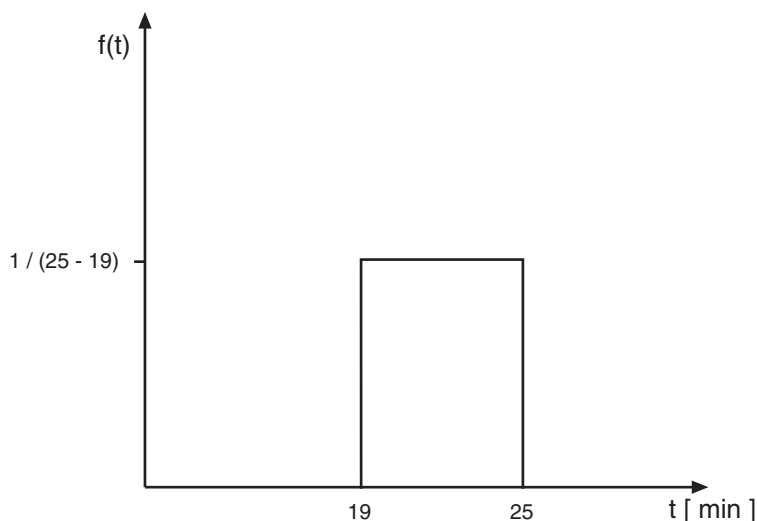
slika 36
Model analogno digitalnog sistema za izradu mastera

PostScript program s kojim je definiran izgled mastera jedne A4 forme predstavlja ulaz u model. PostScript naredbama definiran je poseban izgled svake boje za prednju i stražnju stranu (separacija boja). RIP prima PostScript program preko Ethernet veze i počinje prevoditi PostScript naredbe u bitmapu svake boje zasebno. To su A4 bitmap forme. Simulacija CTP-a mora sadržavati mogućnost impozicije bitmapirane forme na arak formata offsetne ploče. Ako je format offsetne ploče B1 tada će RIP sam napraviti novu veliku bitmap formu

koja se sastoji od 9 kopija početne A4 forme koje se pozicioniraju u koordinatnom sustavu RIP-a na način da se popuni B1 offsetna ploča. Iz svake takve bitmape B1 forme vrši se fotoosvjetljavanje offsetne ploče za tisak svake boje zasebno. Offsetne ploče za tisak boja prednje strane stavljaju se na cilindre offsetnog stroja. Nakon tiska prednje strane pere se stroj i stavljaju se offsetne ploče za tisak stražnje strane, a na ulaz se postavljaju okrenuti arci.

Nakon završetka posla na tiskovnom modulu svi arci idu u proces dorade koji sadržava i vrijeme sušenja, rezanja, brojenja i kontrole kvalitete. Model mora omogućiti promjenu broja boja na masteru, formata odnosno broja formi na arku stroja i mijenjanje naklada istih.

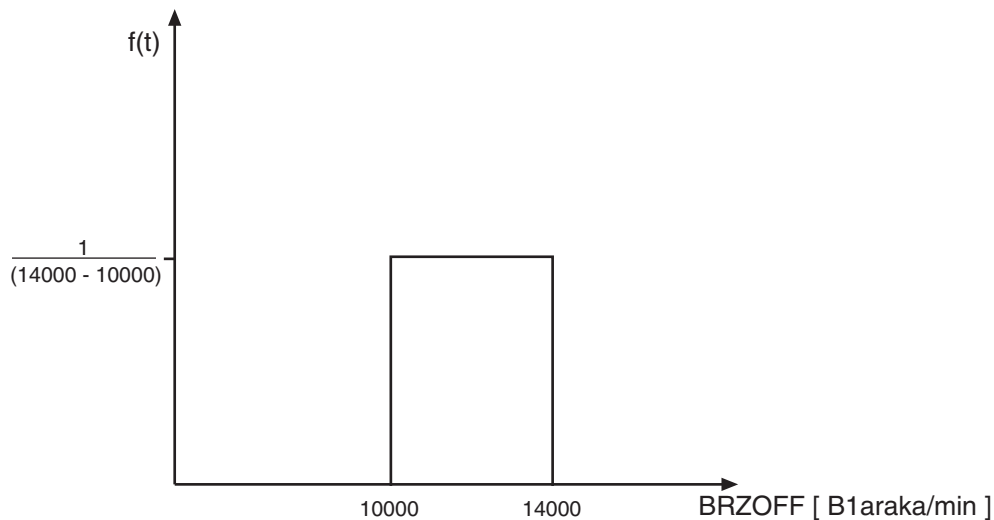
Kada se razviju sve ofsetne ploče započinje faza pripreme ofsetnog stroja. Ona zavisi od broja tiskarskih modula (cilindara) na koje se fiksiraju pripremljene ofsetne ploče. Svaki modul je zadužen za tisak jedne boje. Ovdje se proučava model sa 5 takvih modula. Posao koji treba obaviti na tom ofsetnom sustavu je dvostrani tisak gdje se u jednom prolazu tiskaju 4 boje, a u drugom 5 boja (peta je UV zaštitna boja). Vrijeme stavljanja ploča i pranje stroja odgovara uniformnoj funkciji vjerojatnosti na slici 37.



Slika 37

Uniformna funkcija vjerojatnosti vremena pripreme peterobojnog ofsetnog stroja

Nakon što se na tiskovnim modulima postave ofsetne ploče za svaku boju mastera, pokreće se faza tiska čija je brzina prikazana uniformnom funkcijom vjerojatnosti na slici 38 dobivenom mjerenjima na 324 uzorka tiska B1 formata na peterobojnog stroju. Nakon završetka tiska naklade prednje strane, arci se preokreću i stavljaju na ulaz u stroj koji se ponovo mora pripremiti s novim ofsetnim pločama za tisak stražnje strane mastera.



Slika 38
Uniformna funkcija vjerojatnosti brzine peterobojnog ofsetnog stroja B1 formata

Vrijeme potrebno za tisak jedne određene naklade B1 formata je određeno izrazom

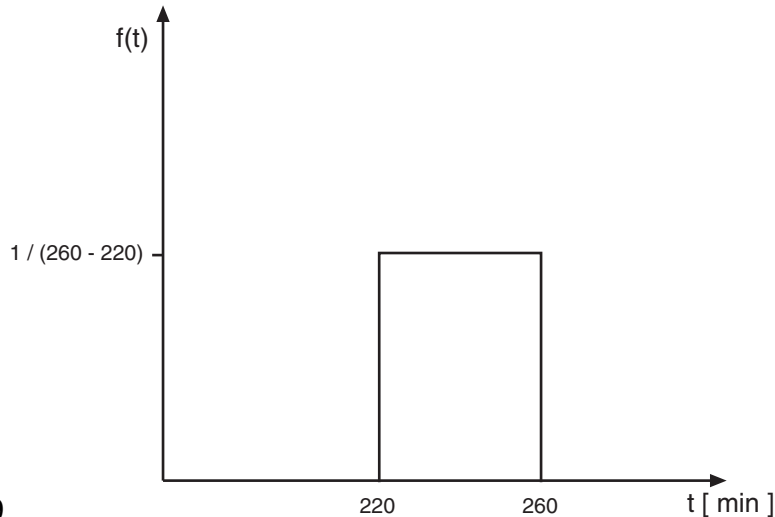
$$t = (\text{ANAKL} \times 60) / \text{FN}(\text{BRZOFF}) \quad [\text{min}]$$

gdje je ANAKL zadana naklada u B1 arcima, FN(BRZOFF) kontinuirana funkcija raspodjele brzina ofsetnog stroja izvedena iz funkcije vjerojatnosti sa slike 38. Varijabla ANAKL se izvodi iz zadane naklade mastera NAKL i broja formi mastera BA4STR na arku stroja s izrazom

$$\text{ANAKL} = (\text{NAKL} / \text{BA4STR}) \times 1.10 \quad [\text{kom}] .$$

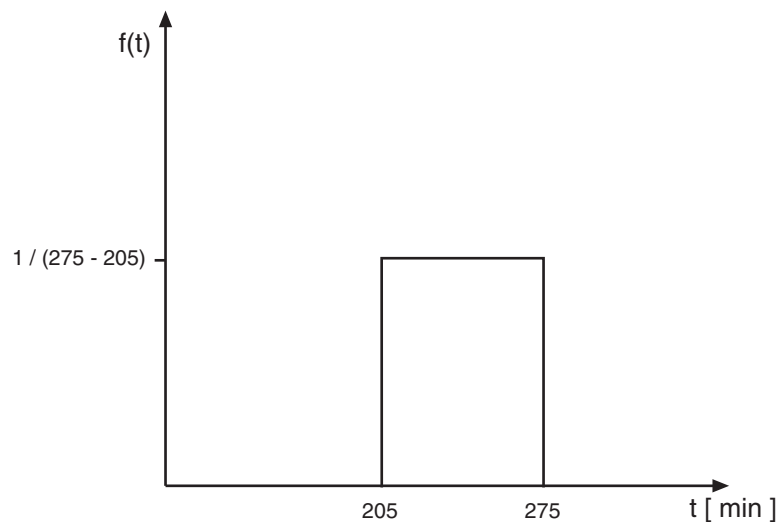
Zbog realnog škarta od 10 posto u izraz je ugrađen i faktor škarta od 1.10.

Kada se završi ofsetni tisak bilo koje naklade, potrebno je vrijeme sušenja ofsetne boje prije bilo kakve dorade. Mjerenjem sušenja u praksi na 146 uzoraka sušenja obostrano ofsetno otisnutih poslova došlo se do funkcije vjerojatnosti na slici 39.



Slika 39
Uniformna funkcija vjerojatnosti vremena sušenja obostranog ofsetnog tiska

Nakon sušenja arci se mogu početi doradivati. U vrijeme dorade uključeno je rezanje na format mastera, kontrola kvalitete, odbacivanje škarta i obavezno brojenje na strojevima za brojenje. Izvedeno je 22 mjerenja dorade naklade od 150000 mastera. Dobivena je funkcija vjerojatnosti vremena dorade na slici 40.



Slika 40
Uniformna funkcija vjerojatnosti vremena dorade naklade mastera od 150000

Vrijeme potrebno za cjelokupnu doradu jedne određene naklade NAKL je određeno izrazom

$$t = FN(SUS) + (NAKL \times FN(DOR)) / 150000 \quad [\text{min}]$$

gdje je FN(SUS) kontinuirana funkcija raspodjele vremena sušenja ofsetne boje obostranog otiska izvedena iz funkcije vjerojatnosti sa slike 39, a FN(DOR) predstavlja kontinuiranu funkciju raspodjele vremena dorade izvedenu iz funkcije vjerojatnosti sa slike 40.

Nakon faze dorade, masteri su spremni za individualizaciju koja će biti prikazana drugim modelom nezavisno od ovog.

6.2.1. GPSS programsko rješenje sustava

Dinamički element modela je jedan posao definiran svojom nakladom, brojem boja i brojem strana tiska koji je u GPSS programu predstavljen kao jedna transakcija. Ideja je da se takav model izvršava barem 1000 puta što bi bilo regulirano simulacijskom petljom. Na taj način bi sve stohastičke varijable sustava mogle ispuniti svoje stohastičke vrijednosti. Tokom izvršavanja simulacije, u dijelu simulacije CTP sustava transakcija se razdvaja u više transakcija čiji je broj definiran brojem potrebnih ofsetnih ploča za pojedinu boju mastera. Nakon završetka u dijelu CTP sustava, transakcije se stapaju ponovno u jednu transakciju. U dijelu simulacije offsetnog tiska, transakcija posla se razdvaja u dvije transakcije ukoliko se radi o obostranom tisku mastera. Nakon završetka stapaju se u jednu transakciju koja napreduje sve do kraja modela.

Statički elementi modela su RIP CTP sustava, fotoosvjetljivač CTP sustava, razvijačka jedinica CTP sustava i ofsetni stroj, dok je sustav dorade samo

vremenska jedinica zaustavljanja jedne transakcije posla. RIP, fotoosvjetljivač, razvijačka jedinica i ofsetni stroj predstavljeni su kao FACILITY i određuju jednokanalna usluživanja.

Svakim startanjem modela unosi se naklada finalnih formi preko varijable NAKL i broj boja tiska preko varijable BB. Prikazani programski model na slici 41a) i 41b) vrši ispis u zaseban file. Prikazani primjer ispisuje za određenu nakladu araka potrebno minimalno, maksimalno i prosječno vrijeme u minutama koje je potrebno za izvršenje posla za zadani sistem preko C1 systemske varijable GPSS vremenskog sata. Na taj način se ubrzava eksperimentiranje ako je samo vrijeme izvršavanja modela od našeg interesa. Inače bi nakon svakog izvršavanja modela morali gledati u standardnu izlaznu statistiku i sporo tražili željenu vrijednost od interesa.

Programski model je napisan tako da se lako mogu mijenjati svi parametri svake varijable, a mogu se lako mijenjati i konfiguracije modela.

Ovo istraživanje pokazuje da se s malim intervencijama u kreiranom programskom modelu mogu jednostavno voditi eksperimenti s različitim nakladama, različitim zamišljenim konfiguracijama ili samo mjerenjem vremena izvršavanja čitavog modela radi pravilnih odluka o izvršavanju poslova u zadanom vremenu i na taj način pravilnog ugovaranja poslova na duži rok s fiksnim nakladama.


```
1 *****MODEL ANALOGNO DIGITALNOG SUSTAVA ZA IZRADU MASTERA*****
2 *****ANALOGNO-OFFSETNI STROJ****DIGITALNO-ComputerToPlate(CTP)
3 SIMULATE
4 INTEGER &I,&N
5 INTEGER &NAKL,&BB,&BS,&BA4STR
6 REAL &ANAKL,&TRIP,&DTRIP
7 REAL &VRIJ,&MIN,&MAX
8 LET &MIN=999999.
9 LET &MAX=-999999.
10 LET &N=1000 broj izvrsavanja simulacija modela
11 PUTPIC
12 OUnesi nakladu formi mastera (npr.150000)
13 GETLIST &NAKL
14 PUTPIC
15 OUnesi broj boja mastera (npr.9)
16 GETLIST &BB
17 OUT FILEDEF 'MASTER2.OUT'
18 BRZOFF FUNCTION RN(1),C2
19 0.0,10000/1.0,14000
20 SUS FUNCTION RN(1),C2
21 0.0,220/1.0,260
22 DOR FUNCTION RN(1),C2
23 0.0,205/1.0,275
24 LET &BA4STR=9 broj formi na arku stroja B1
25 LET &ANAKL=(&NAKL/&BA4STR)*1.10 naklada u arcima stroja+10%
26 LET &TRIP=10 prosj. vrijeme CTP ripanja 1 ploce
27 LET &DTRIP=2 +-odstupanje od &TRIP
28 LET &BS=2
29
30 1 GENERATE ,,,1
31 2 ADVANCE 20,3 impozicija i procesiranje
32 3 SPLIT &BB-1,SKOK stvaranje broja ploca koliko ima boja
33 4 SKOK SEIZE CTRIP
34 5 ADVANCE &TRIP,&DTRIP
35 6 RELEASE CTRIP
36 7 SEIZE OSVJ
37 8 ADVANCE 2
38 9 RELEASE OSVJ
39 10 SEIZE RAZVIJ
40 11 ADVANCE 3
41 12 RELEASE RAZVIJ
42 13 ASSEMBLE &BB sakupljene sve ploce
43 14 SPLIT &BS-1,SKOK2 stvaranje broja prolaza kroz stroj(2zadup)
44 15SKOK2 SEIZE OFFSET
45 16 ADVANCE 22,3 vrijeme stavljanja ploca + pranje stroja
46 17 ADVANCE (&ANAKL*60)/FN(BRZOFF)
47 18 RELEASE OFFSET
48 19 ASSEMBLE &BS
49 *****DORADA
50 20 ADVANCE FN(SUS)+(&NAKL*FN(DOR)/150000) 4hsus.+rez.+br.+kontr.
51 21 TERMINATE 1
```

Slika 41a
GPSS model analogno digitalnog sustava

```
52          ****PETLJA IZVRSAVANJA MODELA
53      DO &I=1,&N
54      CLEAR
55      START 1, NP
56      LET &VRIJ=C1+&VRIJ
57      IF &MIN>C1
58      LET &MIN=C1
59      ENDIF
60      IF &MAX<C1
61      LET &MAX=C1
62      ENDIF
63      ENDDO
64      LET &VRIJ=&VRIJ/&N
65      PUTPIC FILE=OUT,LINES=5,
        (&NAKL,&ANAKL,&MIN,(&MIN/60),&MAX,(&MAX/60),&VRIJ,(&VRIJ/60))
66
67      Za nakladu od * A4 mastera odnosno * araka
68      potrebno je minimalno *.*** minuta ili *.*** sati,
69      potrebno je maksimalno *.*** minuta ili *.*** sati,
70      potrebno je prosjecno *.*** minuta ili *.*** sati,
71
72      END
```

Slika 41b
GPSS model analogno digitalnog sustava

6.2.2. Opis rutina i veza radi vođenja eksperimenata

Instrukcija 3: SIMULATE; izvršavanje programskog modela nakon uspješne kompilacije.

Instrukcija 4: INTEGER &I,&N; deklaracija cjelobrojnih programskih varijabli &I i &N. Varijabla &I predstavlja brojač petlje izvršavanja modela, a s varijablom &N se regulira broj izvođenja modela.

- Instrukcija 5: INTEGER &NAKL,&BB,&BS,&BA4STR;
deklaracija cijelobrojnih varijabli:&NAKL za nakladu formi mastera, &BB za broj boja jednog posla odnosno potrebnu količinu ofsetnih ploča, &BS za broj prolaza kroz stroj (ovisan o broju strana i broju instaliranih tiskarskih cilindara), &BA4STR za broj formi na arku.
- Instrukcija 6: REAL &ANAKL,&TRIP,&DTRIP; deklaracija realnih varijabli: &ANAKL za nakladu u arcima stroja, &TRIP za prosječno vrijeme CTP ripanja 1 ploce, &DTRIP za odstupanje od &TRIP
- Instrukcija 7: REAL &VRIJ,&MIN,&MAX; deklaracija realnih varijabli: &VRIJ za prosječno vrijeme trajanja izvršavanja modela za zadanu nakladu, &MIN za traženje minimalnog vremena izvršavanja jedne simulacije, &MAX za traženje maksimalnog vremena izvršavanja jedne simulacije
- Instrukcija 8: LET &MIN= 999999. ; inicijalizacija varijable &MIN s najvećim mogućim brojem u GPSS-u
- Instrukcija 9: LET &MAX= -999999. ; inicijalizacija varijable &MAX s najmanjim mogućim brojem u GPSS-u
- Instrukcija 10: LET &N= 1000 ; definiranje broja izvršavanja simulacija modela; s njom se vrši optimalni odabir potrebnog broja izvođenja modela zbog potrebne stohastike
- Instrukcije 11 do 13: PUTPIC; Ispisuje poruku "Unesi nakladu formi (npr.150000)" na ekran, a instrukcija GETLIST &NAKL puni varijablu &NAKL s tastature

- Instrukcije 14 do 16: PUTPIC; Ispisuje poruku "Unesi broj boja mastera (npr. 9)" na ekran, a instrukcija GETLIST &BB puni varijablu &BB s tastature
- Instrukcija 17: OUT FILEDEF 'MASTER2.OUT' ; definira izlaznu datoteku MASTER2.OUT u koju će se ispisivati željeni rezultati simulacije
- Instrukcije 18 do 19: BRZOFF FUNCTION RN(1),C2 ; kontinuirana (C) uniformna funkcija vjerojatnosti brzine ofsetnog stroja (BRZOFF) s dva parametra (2); standardni numerički atribut RN(1) generira uniformno pseudoslučajni realni broj unutar intervala (0,1) ne uključujući 0 i 1.
- Instrukcije 20 do 21: SUS FUNCTION RN(1),C2 ; kontinuirana uniformna funkcija vjerojatnosti vremena sušenja ofsetnog tiska (SUS);
- Instrukcije 22 do 23: DOR FUNCTION RN(1),C2 ; kontinuirana uniformna funkcija vjerojatnosti vremena dorade naklade mastera od 150000 (DOR);
- Instrukcija 24: LET &BA4STR=9 ; broj formi na arku
- Instrukcija 25: LET &ANAKL=(&NAKL/&BA4STR)*1.10 ; naklada u arcima stroja+10%
- Instrukcija 26: LET &TRIP=10 ; prosj. vrijeme CTP ripanja 1 ploče
- Instrukcija 27: LET &DTRIP=2 ; + - odstupanje od &TRIP
- Instrukcija 28: LET &BS=2 ; broj prolaza kroz stroj

Blok 1: GENERATE „,1 ; generiranje prve i jedine transakcije

Blok 2: ADVANCE 20,3 ; u ovom bloku počinje teći vrijeme zadržavanja transakcije po uniformnoj raspodjeli 20+-3 minute za impoziciju i procesiranje

Blok 3: SPLIT &BB-1,SKOK ; stvara se &BB-1 djece transakcija kojih zajedno s originalnom transakcijom čini skup od &BB transakcija koje jedna po jedna putuju na blok sa SKOK oznakom. Sve stvorene transakcije u SPLIT bloku zajedno s transakcijom otac postaju članovi iste porodice transakcija (Assemble Set). Na taj način se simulira stvaranje kompleta ploča.

Blokovi od 4 do 6: SKOK SEIZE CTPRIP
ADVANCE &TRIP,&DTRIP
RELEASE CTPRIP

definiraju put transakcije počevši od zauzimanja modula za CTP riping (SEIZE CTPRIP). Potom počinje teći vrijeme zadržavanja transakcije u njemu po uniformnoj raspodjeli &TRIP+-&DTRIP minute (ADVANCE &TRIP,&DTRIP). Tek nakon proteklog vremena može se osloboditi modul za CTP riping (RELEASE CTPRIP).

Blokovi od 7 do 9: SEIZE OSVJ
ADVANCE 2
RELEASE OSVJ

reguliraju put transakcije počevši od zauzimanja modula za osvjetljavanje (SEIZE OSVJ). Potom počinje teći vrijeme zadržavanja transakcije

(ADVANCE 2). Tek nakon proteklog vremena može se osloboditi modul za osvjetljavanje (RELEASE OSVJ).

Blokovi od 10 do 12: SEIZE RAZVIJ
 ADVANCE 3
 RELEASE RAZVIJ

definiraju put transakcije počevši od zauzimanja modula za razvijanje (SEIZE RAZVIJ). Potom počinje teći vrijeme zadržavanja transakcije u njemu(ADVANCE 3). Tek nakon proteklog vremena može se osloboditi modul za razvijanje za slijedeću transakciju (RELEASE RAZVIJ).

Blok 13: ASSEMBLE &BB ; U ovom bloku se spajaju sve transakcije koje pripadaju istoj porodici u jednu transakciju prije nego što se krene dalje kroz model. Tako se simulira sakupljanje svih ploča.

Blok 14: SPLIT &BS-1,SKOK2 ; stvara se &BS-1 djece transakcija kojih zajedno s originalnom transakcijom čini skup od &BS transakcija koje jedna po jedna putuju na blok sa SKOK2 oznakom. Na taj način se simulira stvaranje broja prolaza kroz stroj.

Blokovi od 15 do 18: SKOK2 SEIZE OFFSET
 ADVANCE 22,3
 ADVANCE (&ANAKL*60)/FN(BRZOFF)
 RELEASEOFFSET

definiraju put transakcije počevši od zauzimanja modula za offset stroj (SEIZE OFFSET). Potom počinje teći prvo vrijeme zadržavanja transakcije po uniformnoj raspodjeli 22+-3 minute (ADVANCE 22,3) što predestavlja vrijeme pripreme ofsetnog stroja. Nakon toga počinje teći vrijeme zadržavanja

transakcije ADVANCE (&ANAKL *60)/FN(BRZOFF) za simulaciju potrebnog vremena za tisak jedne naklade, gdje je ANAKL zadana naklada u arcima, a FN(BRZOFF) funkcija raspodjele brzina stroja definirana u instrukciji 18. Tek nakon proteklog vremena može se osloboditi modul za offset stroj (RELEASE OFFSET).

Blok 19: ASSEMBLE &BS ; U ovom bloku se spajaju sve transakcije što su bile stvorene u bloku 14 koje pripadaju istoj porodici u jednu transakciju prije nego što se krene dalje kroz model. Tako se simulira broj prolaza kroz stroj.

Blok 20: ADVANCE FN(SUS)+(&NAKL * FN(DOR)/150000) ;
Zadržavanje transakcije za simulirano potrebno vrijeme za cijelokupnu doradu jedne određene naklade. FN(SUS) je funkcija raspodjele vremena sušenja bojekoja je definirana instrukcijama 20 i 21., a FN(DOR) je funkcija raspodjele vremena dorade definirana instrukcijama 22 i 23.

Blok 21: TERMINATE 1 ; Repräsentira napuštanje jednog posla iz realnog sustava. Svaki ulazak u taj blok ujedno smanjuje vrijednost simulacijskog brojača terminacija definiran instrukcijom START. Kada brojač dosegne nulu ili manje, tada se zaustavlja simulacija.

Instrukcija 53: DO &l=1,&N ; definira glavu DO petlje gdje se brojač petlje &l vrti od 1 do &N odnosno do željenog broja izvršavanja modela.

Instrukcija 54: CLEAR ; priprema modela za novo izvršavanje tako da postavlja svu statistiku na nulu i izbacuje sve transakcije iz modela, ali tako da ne resetira generatore slučajnih brojeva u modelu. Na taj

način nova simulacija istog modela kreće sa onim nizom slučajnih brojeva koji se nastavljaju tamo gdje se stalo.

Instrukcija 55: `START 1,NP` ; postavlja početnu vrijednost brojača terminiranih transakcija. Sa vrijednosti 1 postignuto je da u model kreće samo jedna transakcija. S parametrom NP (No Print Out) postiže se izvršavanje bez standardnog ispisa statistike simulacije, jer se izgradio korisnički ispis.

Instrukcija 56: `LET &VRIJ=C1+&VRIJ` ; U svakom koraku DO petlje u varijablu `&VRIJ` nadodaje se C1 vrijeme prethodnog izvršavanja modela.

Instrukcije 57 do 59: `IF &MIN>C1`
 `LET &MIN=C1`
 `ENDIF`

nam sakupljaju uvijek tekuće najmanje izmjereno vrijeme trajanja simulacije modela u varijablu `&MIN`. Za prvi ulazak u taj segment varijabla `&MIN` se napunila sa najvećim brojem u GPSS-u 999999. u instrukciji 8.

Instrukcije 60 do 62: `IF &MAX<C1`
 `LET &MAX=C1`
 `ENDIF`

nam sakupljaju uvijek tekuće najduže izmjereno vrijeme trajanja simulacije modela u varijablu `&MAX`. Za prvi ulazak u taj segment varijabla `&MAX` se napunila sa najvećim brojem u GPSS-u -999999. u instrukciji 9.

Instrukcija 63: `ENDDO` ; kraj DO petlje gdje se brojač petlje povećava za 1.

Instrukcija 64: `LET &VRIJ=&VRIJ/&N` ; stvara prosječno vrijeme izvršavanja modela za `&N` izvršavanja.

Instrukcija 65 :

```
PUTPIC FILE=OUT,LINES=5,  
(&NAKL,&ANAKL,&MIN,(&MIN/60),&MAX,(&MAX/60),&VRIJ,(&VRIJ/60))
```

otvara ispisni kanal u datoteku OUT definiranu instrukcijom 17. Stvara se ispisni niz koji ima 5 linija (LINES=5) i 8 ispisnih varijabli odnosno aritmetičkih izraza.

Instrukcije od 66 do 70:

Za nakladu od * A4 mastera odnosno * araka
potrebno je minimalno *.*** minuta ili *.*** sati,
potrebno je maksimalno *.*** minuta ili *.*** sati,
potrebno je prosjecno *.*** minuta ili *.*** sati,

definiiraju ispisnu masku. Maska se puni redom iz definiranog ispisa iz instrukcije 65.

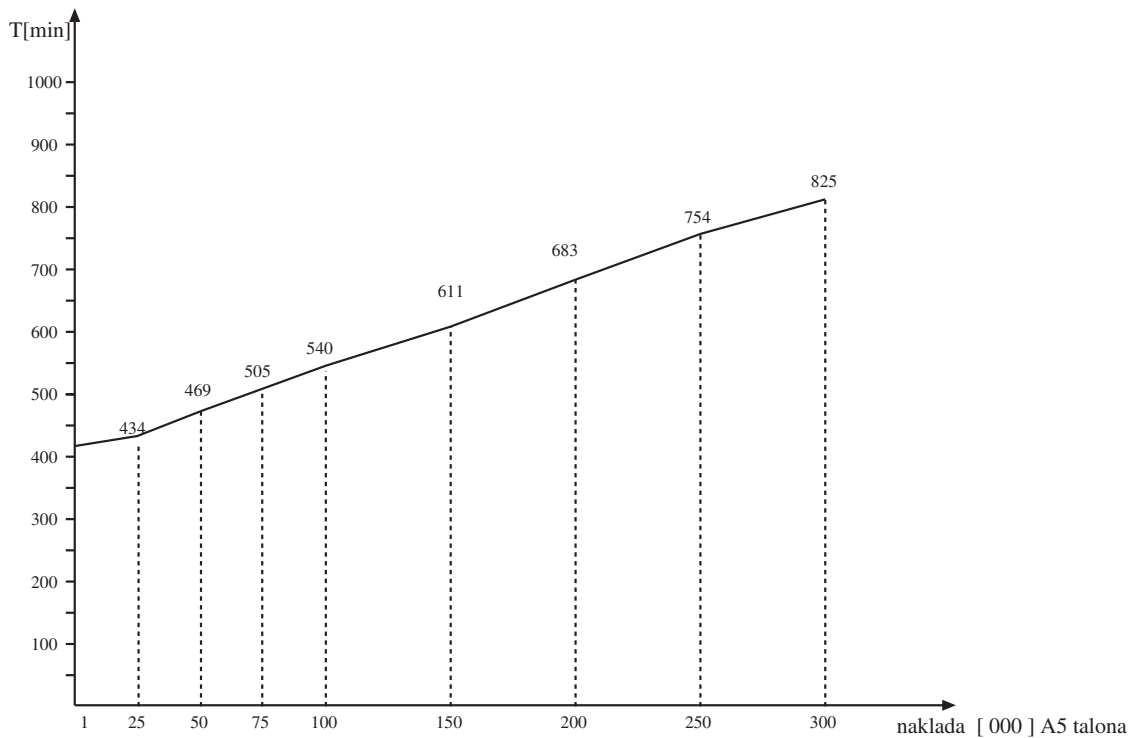
Instrukcija 72: END

definiira kraj GPSS programskog modela.

6.2.3. Eksperimentiranje s modelom

Cilj eksperimentiranja je bilo promatranje prosječne, minimalne i maksimalne brzine odziva modela za različite naklade. Na slici 42 je prikazan grafikon koji nam prikazuje prosječno vrijeme trajanja izvršavanja sustava za izradu tiska u odnosu na zadane naklade. Analizirale su se naklade A5 talona između 25.000 i 300.000.

Zbog ugrađene petlje unutar GPSS modela za svaku zadanu nakladu vrši se 1000 nezavisnih startanja modela radi zadovoljavanja statističkih uvjeta



Slika 42

Prosječno vrijeme trajanja izvršavanja sustava za izradu tiska u odnosu na zadane naklade

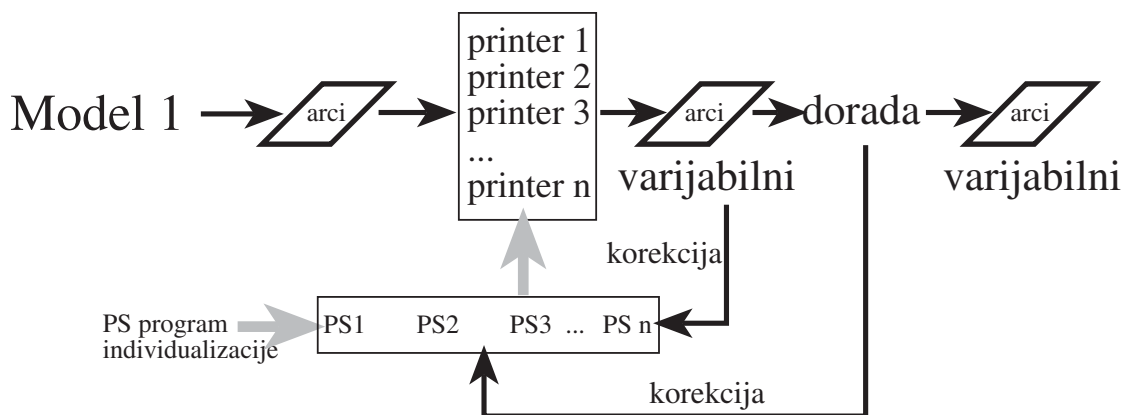
stohastičkih varijabli modela. To znači da se za rezultate prikazane grafikonom model izvršavao sveukupno 7.000 puta (7 naklada x 1000).

Na nakladama ispod 50000 vrijeme izvršavanja dolazi u zasićenje jer vrijeme trajanja pripreme stroja postaje puno veće u odnosu na vrijeme tiska male naklade. Ti rezultati se mogu dalje upotrijebiti kao ulazni podaci u modele individualizacije različitih digitalnih tehnologija ili se mogu individualno promatrati i analizirati. Za najveću nakladu od 300.000 otisaka potrebno je 825 minuta odnosno 13,75 sati. Ako analiziramo zadatak definiran na početku, da za cijeli posao zajedno s individualizacijom postoji zadani rok izrade od 42 sata, onda je za individualizaciju 300.000 talona preostalo 28 sati odnosno 1680 minuta.

6.3. Model modularne individualizacije s novim rasterskim procedurama

Sustav prikazan na slici 43 predstavlja osnovnu predloženu konfiguraciju za individualizaciju velikih naklada s novim rasterskim procedurama. Iz modela 1 za izradu mastera, dobivena je naklada otisnutih mastera spremnih za individualizaciju. Polje printera preko zasebnih RIP-ova između sebe distribuiraju nakladu mastera [33]. Za svaki RIP je napisan različit PostScript program zbog neujednačenosti preciznosti ispisnog sustava PostScript elektrofotografskih ispisnih uređaja i to bez obzira da li se radi o istim proizvođačima i modelima uređaja. Svaka otisnuta varijabilna informacija se odmah provjerava u smislu pozicioniranja i kvalitete ispisa. Ako je potrebna korekcija, mora se pristupiti ponovnom slanju PostScript zapisa koji definira varijabilnu informaciju.

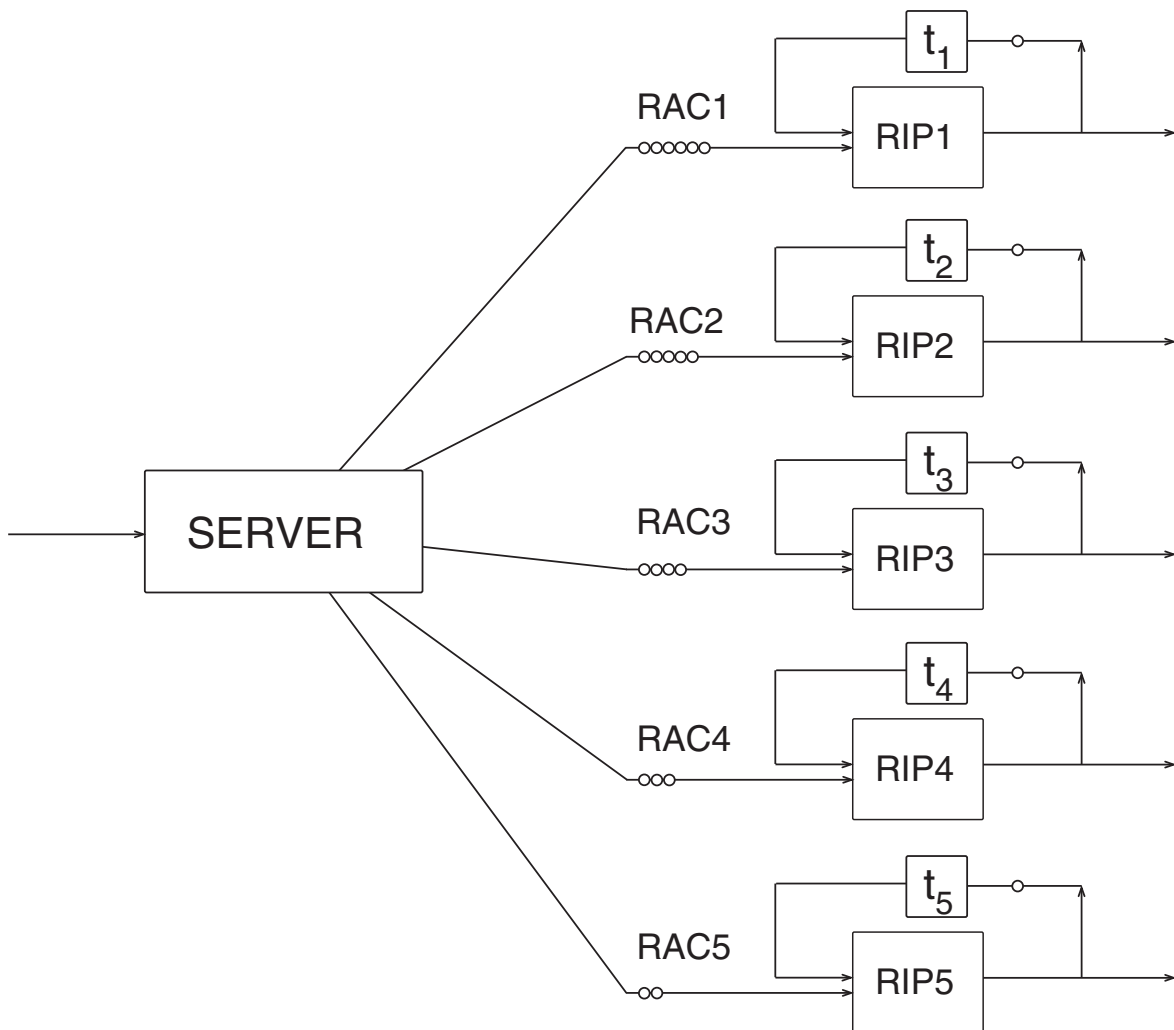
Kada cijeli sustav završi posao ulazi u fazu dorade. Dorada se sastoji od rezanja arka u osnovne varijabilne talone i vrši se ponovna provjera broja i kvalitete. Ako dođe do greške u doradi, mora se postupak individualizacije ponoviti. Pošto dorada prati cijelo vrijeme paralelno fazu individualizacije, njeno vrijeme je minimalno, odnosno zanemarivo u odnosu na vrijeme cijele



Slika 43
Model individualizacije

individualizacije pa će se izostaviti iz gradnje modela jer ne utječe na njegovu propusnost i iskorištenje.

Ulaz u sustav za individualizaciju su PostScript programski zapisi koji izvršavaju individualizaciju jednog od 1000 intervala zadane naklade. Intervali su dimenzionirani na taj način zbog kontrole koja je obavezna kod takvih poslova. Sustav se sastoji od servera koji distribuira PostScript zapise i nezavisnih RIP modula. RIP modul se sastoji od računala na kojem se formira red čekanja, RIP-a i samog tiskarskog stroja. Distribucija PostScript zapisa prikazana je na slici 44.



Slika 44
Distribucija PostScript zapisa u Modelu 2

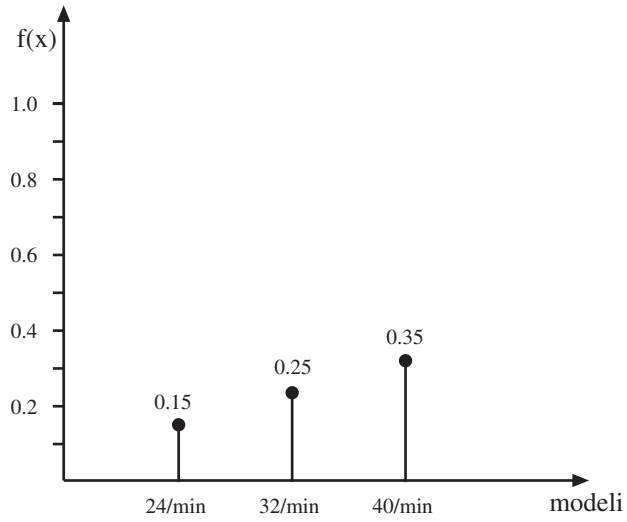
Tokom procesa treba generirati 1000 PostScript programa, čija je dinamika ulaska u model bazirana na dvogodišnjem iskustvu. Kada se PostScript zapis pošalje prema jednom modulu za ripping, onda na tom modulu treba taj zapis biti završen bez obzira na zastoje i kvarove koji se mogu dogoditi tokom tiska. Na takvoj konfiguraciji se tokom dvije godine mjere podaci koji su osnova za verifikaciju i ocjenu valjanosti stohastičkog modela realnog sustava individualizacije otisaka.

Model sustava mora funkcionirati i ako dođe do potpunih zastoja kao što su kvarovi. Tada treba usporiti ulaz PostScript zapisa u sustav i čekati rješavanje zastoja.

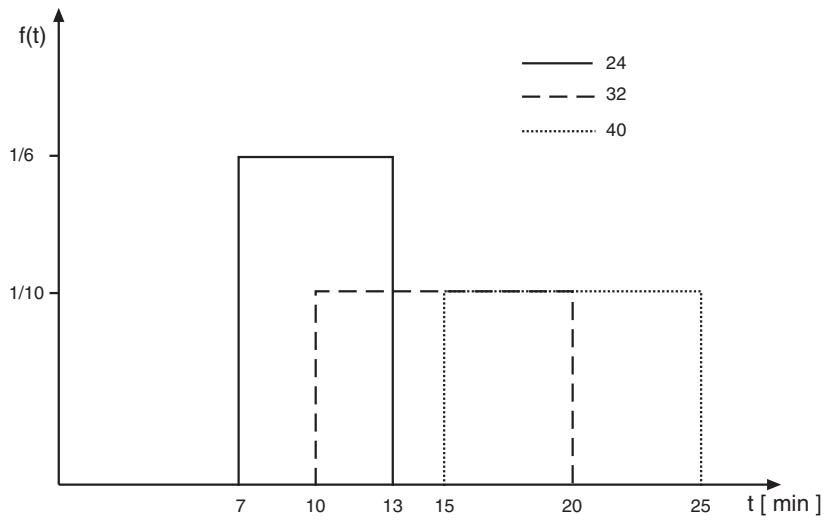
Rekonfiguriranje takvog sistema zamjenom druge vrsta RIP-a ili proširivanjem ili odbacivanjem postojećih nemoguće je, jer se naklada i rokovi (novi poslovi dolaze jedan za drugim) definiraju neposredno prije tiska. Svako eksperimentiranje na realnom sustavu moglo bi izazvati opasne zastoje, a time i neizvršenje tiska u strogo određenom vremenskom intervalu.

Ako sustav nije potpuno blokiran, PostScript zapisi ulaze s brzinom od 1 PS/min. Vrijeme generiranja PS zapisa sastoji se od ažuriranja baze, stvaranja PS zapisa i slanja PS zapisa. Ako je sustav u potpunom zastoju, tada se PostScript zapisi generiraju brzinom 1 PS/20 min.

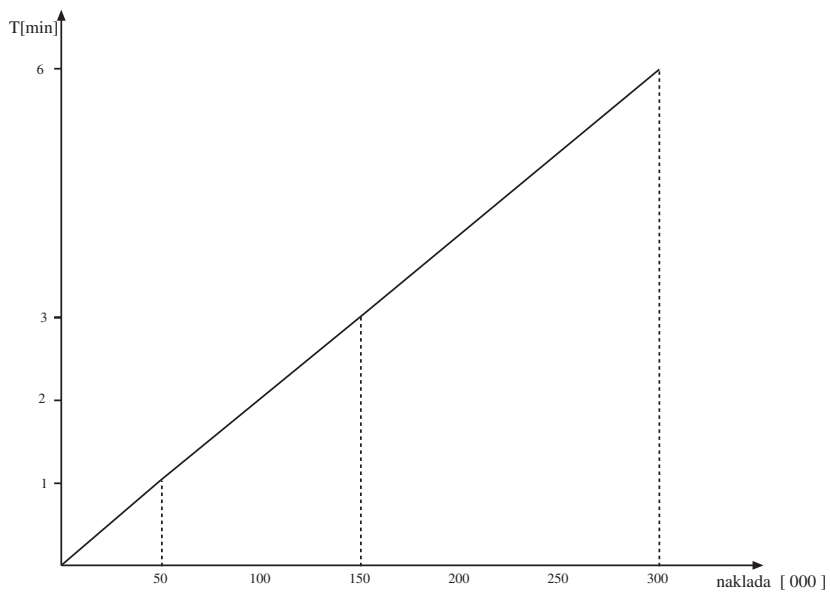
Gradnja eksperimentalnog fizičkog modela bila bi preskupa i zbog samih RIP modula, ali i zbog potrošnje repromaterijala, boje i papira pa se pristupilo stvaranju programskog simulacijskog modela. Eksperimentiranje s programskim simulacijskim modelom treba omogućiti optimalno korištenje postojećih resursa za zadanu nakladu za zadani rok, procjene isplativosti zamjene ili proširivanja postojećih resursa odnosno dinamičko konfiguriranje cjelog grafičkog sustava ovisno o nakladi i rokovima. U programskom modelu treba omogućiti i istraživanje metoda rješavanja nepotpunih i potpunih zastoja.



Slika 45 Distribucija kvarova



Slika 46 Vremena rješavanja kvarova



Slika 47 Vrijeme kontrole tiska jednog intervala (bloka)

Iz mjerenja na fizičkom sustavu unatrag dvije godine, poznate su distribucije kvarova RIP modula (slika 45), distribucije vremena rješavanje kvarova (slika 46) i vrijeme kontrole tiska (slika 47).

6.3.1. Rješenje modela u GPSS-u

Sva prethodna mjerenja korištena su za izradu stohastičkog programskog modela. Na slici 48a i 48b prikazan je GPSS program koji predstavlja tiskarsku konfiguraciju¹.

Varijablom NAKL definirana je naklada tiska talona za koju se provodi eksperiment. S varijablom BINT definira se broj intervala individualizacije zbog kontrole individualizacije otiska. Varijabla BTAL određuje broj talona na tiskovnom arku, ANAKL je naklada u arcima stroja, a BNAKL je naklada unutar jednog intervala. U dvije matrice MRIP i MKVAR definirane su raspodjele brzina, trajanja zastoja i raspodjela broja zastoja za svaki RIP modul u sustavu. To omogućava lako prekonfiguriranje modela, a indirektnim adresiranjem članova matrice ostaje sačuvan preostali dio programa. Transakcija modela je nalog za tisak.

RIP je predstavljen kao facility, a računala odnosno tvrdi disk računala gdje se formira red čekanja za zadani RIP kao storage. Svaki RIP modul ima i pripadnu logičku sklopku koja se aktivira u slučaju kvara odnosno deaktivira nakon isteka vremena zastoja. Na taj način se ne dopušta pristup RIP-u u zastoju već se transakcije preusmjeravaju.

Takvi paralelni RIP moduli zajedno sa pripadnim sklopkama stvaraju se i biraju indirektno preko parametra transakcije PH(RIP). Taj parametar nadolazeće transakcije puni se sa brojem izabranog RIP modula u dijelu

```
1      SIMULATE
2      INTEGER &NAKL,&BINT,&BTAL,&BRIP,&BTEK,&IP,&JP
3      REAL &ANAKL,&BNAKL,&PROVR,&BT24,&BT32,&BT40
4      PUTPIC
5      0Unesi nakladu individualizacija (npr. 300000)
6      GETLIST &NAKL
7      OUT FILEDEF 'DISTRIB.OUT'
8      LET &BINT=1000          broj interv. zbog kontrole indiv. otiska
9      LET &BTAL=2            broj otisaka na arku (A5 na A4)
10     LET &ANAKL=&NAKL/&BTAL  naklada u arcima (A4)
11     LET &BNAKL=&ANAKL/&BINT  naklada jednog intervala (bunta)
12     LET &PROVR=&BNAKL*3/150  vrijeme provjere za jedan interval
13     LET &BT24=&BNAKL/24+&PROVR vrijeme tipa 24A4/min+provjera za 1 interval (bunt)
14     LET &BT32=&BNAKL/32+&PROVR
15     LET &BT40=&BNAKL/40+&PROVR
16     LET &BRIP=5            potrebno za kontrolu petlje odabira
17 MRIP MATRIX ML,5,4
18     INITIAL ML$MRIP(1,1),&BT40/ML$MRIP(1,2),0/ML$MRIP(1,3),20/ML$MRIP(1,4),5
19     INITIAL ML$MRIP(2,1),&BT32/ML$MRIP(2,2),0/ML$MRIP(2,3),15/ML$MRIP(2,4),5
20     INITIAL ML$MRIP(3,1),&BT32/ML$MRIP(3,2),0/ML$MRIP(3,3),15/ML$MRIP(3,4),5
21     INITIAL ML$MRIP(4,1),&BT32/ML$MRIP(4,2),0/ML$MRIP(4,3),15/ML$MRIP(4,4),5
22     INITIAL ML$MRIP(5,1),&BT24/ML$MRIP(5,2),0/ML$MRIP(5,3),10/ML$MRIP(5,4),3
23 MKVAR MATRIX ML,5,1
24     INITIAL ML$MKVAR(1,1),35
25     INITIAL ML$MKVAR(2,1),25
26     INITIAL ML$MKVAR(3,1),25
27     INITIAL ML$MKVAR(4,1),25
28     INITIAL ML$MKVAR(5,1),15
29     STORAGE S(1),6/S(2),6/S(3),6/S(4),6/S(5),6
30     INITIAL LR(RIP)/LR(11)
31     INITIAL XL(INTER),1.0
32 1      GENERATE XL(INTER),,,&BINT
33 2      QUEUE 10
34 3      GATE LR 11
35 4      DEPART 10
36 5      TRANSFER ,TRAZI
37 6 POSAO LOGIC R 11
38 7      SAVEVALUE INTER,1.0,XL
39 8      ENTER PH(RIP)
40 9      SEIZE PH(RIP)
41 10     ADVANCE ML(MRIP,PH(RIP),1),ML(MRIP,PH(RIP),2)
42 11     TRANSFER .ML(MKVAR,PH(RIP),1),GOT,KVAR
43 12 GOT  LEAVE PH(RIP)
44 13     RELEASE PH(RIP)
45 14     TERMINATE 1
46 15 KVAR LOGIC S PH(RIP)
47 16     ADVANCE ML(MRIP,PH(RIP),3),ML(MRIP,PH(RIP),4)
48 17     LEAVE PH(RIP)
49 18     RELEASE PH(RIP)
50 19     LOGIC R PH(RIP)
51 20     TERMINATE 1
52 21 PREKID LOGIC S 11
53 22     SAVEVALUE INTER,20,XL
54 23     ADVANCE 0.1
55 24     TRANSFER ,TRAZI
56 25 TRAZI SELECT E POMOC,1,&BRIP,1,LR,PREKID
57 26     SELECT E POMOC,1,&BRIP,0,SF,PREKID
58 27     BLET &BTEK=1
59 28     BLET &IP=4
60 29 PETLJA TEST E LR(&BTEK),1,NISTA
61 30     BLET &IP=&IP+1
62 31     ASSIGN &IP,&BTEK,PH
63 32 NISTA BLET &BTEK=&BTEK+1
64 33     TEST LE &BTEK,&BRIP,SKOK1
65 34     TRANSFER ,PETLJA
```

Slika 48a

GPSS program koji predstavlja simulacijski model konfiguracije 1


```
66 35 SKOK1 BLET &JP=&IP
67 36     SELECT MAX RIP,PH(5),PH(&JP),,R
68 37     TRANSFER ,POSAO
69     START &BINT
70     PUTPIC FILE=OUT,LINES=4,(&NAKL,&ANAKL,C1,(C1/60))
71
72     Za nakladu od * individualiziranih otisaka odnosno * araka
73     bilo je potrebno *.*** minuta ili *.*** sati.
74
75     PUTPIC FILE=OUT,LINES=5,((FR(1)/10),(FR(2)/10),(FR(3)/10),(FR(4)/10),(FR(5)/10))
76
77     Iskoristenje RIP-ova:
78     RIP(1)= *.***%           RIP(2)= *.***%
79     RIP(3)= *.***%           RIP(4)= *.***%           RIP(5)= *.***%
80
81     PUTPIC FILE=OUT,LINES=3,(((FR(1)+FR(2)+FR(3)+FR(4)+FR(5))/5/10))
82
83     Ukupno iskoristenje RIP sustava je *.*** .
84
85     PUTPIC FILE=OUT,LINES=5,((SR(1)/10),(SR(2)/10),(SR(3)/10),(SR(4)/10),(SR(5)/10))
86
87     Iskoristenje STORAGE-a:
88     SR(1)= *.***%           SR(2)= *.***%
89     SR(3)= *.***%           SR(4)= *.***%           SR(5)= *.***%
90
91     END
```

Slika 48b

GPSS program koji predstavlja simulacijski model konfiguracije 1

programa iza oznake TRAZI. U tom dijelu prvo se testira potpuni zastoj zbog logičkih sklopki (jedinica u kvaru) ili zbog potpuno popunjenih STORAGE-a. Ako nije detektiran potpuni zastoj, tada se ulazi u dio PETLJA u kojem se testira stanje svakog RIP modula. Za svaki RIP koji nema aktiviranu sklopku, zapisuje se stanje njegovog STORAGE-a u zasebnom parametru transakcije. Kada su se u zadanu transakciju sa svojim stanjem STORAGE-a upisali svi RIP-ovi, upotrijebila se funkcija SELECT MAX RIP,PH(5), PH(&JP),,R koja puni parametar transakcije RIP sa brojem RIP modula koji ima najveći ostatak slobodnog prostora u svojem STORAGE-u. Stanje STORAGE-a raspoloživih RIP-ova su tako zapisani u polju parametara transakcije od 5-tog do JP parametra.

Kada transakcija preko transfer bloka ode na KVAR, jer joj se to po statistici dotičnog RIP-a moralo dogoditi, aktivira se pripadna zaštitna sklopka i počinje

teći vrijeme otklanjanja kvara koje se čita indirektno iz matrice. To vrijeme sadržava i vrijeme potrebno za završetak tiska PostScript zapisa nad kojim se kvar dogodio. Tek nakon tog vremena oslobađa se RIP i deaktivira se sklopka za ulaz u RIP modul.

Ako su svi RIP moduli u kvaru ili su im redovi čekanja puni, tada se ide na PREKID gdje se aktivira sklopka broj 11 s kojom zatvaramo dotok novih transakcija na samom početku sve dok se ne stvore radni uvjeti sistema. Istovremeno se vrši usporavanje generiranje transakcija preko savevalue INTER varijable. Čim se sistem oslobodi, deaktivira se ulazna sklopka i ubrzava se generiranje transakcija.

6.3.2. Opis rutina i veza radi vođenja eksperimenata

Instrukcija 1: SIMULATE; izvršavanje programskog modela nakon uspješne kompilacije

Instrukcija 2:

INTEGER &NAKL,&BINT,&BTAL,&BRIP,&BTEK,&IP,&JP

deklaracija cijelobrojnih varijabli:&NAKL za nakladu talona za koju se provodi eksperiment, &BINT za broj intervala individualizacije , &BTAL za broj talona na arku, &BRIP za broj RIP modula, &BTEK za tekući RIP modul u traženju najpovoljnijeg, &IP za tekući indeks polja parametara transakcije, &JP za krajnji indeks polja parametara transakcije.

Instrukcija 3: REAL &ANAKL,&BNAKL,&PROVR,&BT24,&BT32,&BT40

deklaracija realnih varijabli: &ANAKL za nakladu u arcima, &BNAKL za nakladu jednog intervala, &PROVR za potrebno vrijeme provjere za jedan interval, &BT24, &BT32 i &BT40 za potrebna vremena trajanja obrade jednog intervala modula tipa 24A4/min, 32A4/min i 40A4/min respektivno.

- Instrukcije 4 do 6: PUTPIC; Ispisuje poruku "Unesi nakladu individualizacija (npr. 300000)" na ekran, a instrukcija GETLIST &NAKL puni varijablu &NAKL s tastature
- Instrukcija 7: OUT FILEDEF 'DISTRIB.OUT' ; definira izlaznu datoteku DISTRIB.OUT u koju će se ispisivati željeni rezultati simulacije
- Instrukcija 8: LET &BINT=1000 ; broj interv. zbog kontrole indiv. otiska
- Instrukcija 9: LET &BTAL=2 ; broj talona na arku
- Instrukcija 10: LET &ANAKL=&NAKL/&BTAL ; naklada u arcima
- Instrukcija 11: LET &BNAKL=&ANAKL/&BINT ; naklada jednog intervala
- Instrukcija 12: LET &PROVR=&BNAKL*3/150 ; vrijeme provjere za jedan interval
- Instrukcija 13: LET &BT24=&BNAKL/24+&PROVR ; potrebno vrijeme obrade za RIP tipa 24A4/min +provjera za 1 interval
- Instrukcija 14: LET &BT32=&BNAKL/32+&PROVR ; potrebno vrijeme obrade za RIP tipa 32A4/min +provjera za 1 interval
- Instrukcija 15: LET &BT40=&BNAKL/40+&PROVR ; potrebno vrijeme obrade za RIP tipa 40A4/min +provjera za 1 interval
- Instrukcija 16: LET &BRIP=5 ; broj RIP modula potreban za kontrolu petlje odabira RIP-a

Instrukcija 17: MRIP MATRIX ML,5,4 ; definicija matrice MRIP s 5 redaka i 4 kolone čije su vrijednosti brojevi s pomičnim zarezom (ML- fLoating point Matrix)

Instrukcije 18 do 22:

INITIAL ML\$MRIP(1,1),&BT40/ML\$MRIP(1,2),0/ML\$MRIP(1,3),20/ML\$MRIP(1,4),5

INITIAL ML\$MRIP(2,1),&BT32/ML\$MRIP(2,2),0/ML\$MRIP(2,3),15/ML\$MRIP(2,4),5

INITIAL ML\$MRIP(3,1),&BT32/ML\$MRIP(3,2),0/ML\$MRIP(3,3),15/ML\$MRIP(3,4),5

INITIAL ML\$MRIP(4,1),&BT32/ML\$MRIP(4,2),0/ML\$MRIP(4,3),15/ML\$MRIP(4,4),5

INITIAL ML\$MRIP(5,1),&BT24/ML\$MRIP(5,2),0/ML\$MRIP(5,3),10/ML\$MRIP(5,4),3

definiraju inicijalne vrijednosti matrice MRIP i to tako da svaki redak matrice opisuje svojstva modula RIP-a koji se koristi u modelu. Prvi i drugi stupac definiraju raspodjelu brzina RIP-a (prvi stupac za srednju vrijednost, drugi za odstupanje od nje), treći i četvrti stupac definiraju raspodjelu vremena rješavanja kvarova (treći stupac za srednju vrijednost, četvrti za odstupanje od nje).

Instrukcija 23: MKVAR MATRIX ML,5,1 ; definicija matrice MKVAR s 5 redaka i 1 kolonom čije su vrijednosti brojevi s pomičnim zarezom (ML- fLoating point Matrix)

Instrukcije 24 do 28:

INITIAL ML\$MKVAR(1,1),35

INITIAL ML\$MKVAR(2,1),25

INITIAL ML\$MKVAR(3,1),25

INITIAL ML\$MKVAR(4,1),25

INITIAL ML\$MKVAR(5,1),15

definiraju inicijalne vrijednosti matrice MKVAR (kao jednodimenzionalno polje) i to tako da se u svakom retku definira postotak transakcija koji će ući u KVAR od ukupnog broja transakcija koji ulaze u taj tip RIP-a. Broj retka odgovara rednom broju RIP-a u polju RIP-ova.

Instrukcija 29: STORAGE S(1),6/S(2),6/S(3),6/S(4),6/S(5),6 ;definira veličinu reda čekanja na tvrdom disku prijemnog računala

Instrukcija 30: INITIAL LR(RIP)/LR(11) ;postavlja početna stanja logičkih sklopki: logičku sklopku s imenom RIP i logičku sklopku s brojem 11 u stanje RESET (propusno stanje).

Instrukcija 31: INITIAL XL(INTER),1.0 ; postavljanje inicijalne vrijednosti Savevalue globalne GPSS variable INTER kao broja s pomičnim zarezom (XL). U programu nam služi kao varijabla za definiranje vremenskog intervala generiranja transakcija u bloku GENERATE. Kasnije sa SAVEVALUE blokom postizemo usporavanje i ubrzavanje ulaska transakcija u model. To se događa ulaskom transakcije u SAVEVALUE blok odnosno ulazna transakcija će biti okidač u tom bloku.

Blok 1: GENERATE XL(INTER),,,&BINT ; transakcije će se generirati iz ovog bloka svakih INTER minuta. Broj generiranih transakcija je limitiran na &BINT.

Blokovi od 2 do 4: QUEUE 10
 GATE LR 11
 DEPART 10

definiraju put transakcije počevši od ulaza u red čekanja (QUEUE 10) ispred glavnih ulaznih logičkih vrata koja su regulirana logičkom sklopkom broj 11 (GATE LR 11). Tokom izvršavanja modela sklopka broj 11 može ući u stanje SET jer se tada žele zatvoriti ova logička vrata da ne mogu nove transakcije ulaziti u model. Potom se registrira izlaz iz reda čekanja ispred sklopke one transakcije koja je uspjela proći logička vrata.

Blok 5: TRANSFER ,TRAZI ; Preusmjeravanje transakcije na dio programa s oznakom TRAZI gdje će se tražiti slobodan RIP za izvršavanje posla.

Blokovi 6 do 7: POSAO LOGIC R 11
 SAVEVALUE INTER,1.0,XL

programiraju situaciju kada je transakciji već dodijeljen RIP modul pa je prosljeđena na oznaku POSAO. Ovdje se prvo resetira globalna sklopka broj 11 zbog moguće situacije da je do tada sklopka bila u SET poziciji pa se tako dozvoljava ulazak novim transakcijama na ulaznim logičkim vratima GATE LR 11. Nakon toga se definira INTER varijabla ponovno na standardno vrijeme međudolazaka transakcija ukoliko je bilo do tada usporavanje u dijelu programa PREKID.

Blokovi 8 do 10: ENTER PH(RIP)
 SEIZE PH(RIP)
 ADVANCE ML(MRIP,PH(RIP),1),ML(MRIP,PH(RIP),2)

definiraju paralelni rad polja RIP-ova. Transakcija ulazi u onaj STORAGE čiji je broj već zapisan u parametru transakcije s imenom RIP (primjeti da oni imaju različite kapacitete). Nakon toga se zauzima RIP modul kao FACILITY s istim rednim brojem kao i STORAGE. Zauzimanjem odabranog RIP modula počinje teći vrijeme zadržavanja transakcije u bloku ADVANCE ML(MRIP,PH(RIP),1), ML(MRIP,PH(RIP),2). Srednje vrijeme zadržavanja i njegovo ostupanje preuzima se iz matrice MRIP iz retka definiranog brojem u PH(RIP) parametru i stupaca 1 i 2 respektivno.

Blokovi 11 do 14: TRANSFER .ML(MKVAR,PH(RIP),1),GOT,KVAR
 GOT LEAVE PH(RIP)
 RELEASE PH(RIP)
 TERMINATE 1

reguliraju grananje putanje transakcije nakon što je prošlo vrijeme njenog zadržavanja. Grananje u dio programa KVAR će biti izvršeno statistički preko postotka transakcija za taj tip RIP-a koji se čita iz matrice MKVAR (postoji samo jedan stupac). Točka ispred predstavlja decimalnu točku jer je u matrici zapisan samo decimalni dio. Transakcije koje ne dožive statističko preusmjeravanje na KVAR idu na oznaku GOT gdje se prvo izvršava izlaz (LEAVE) pripadnog STORAGE-a znanog preko zapisanog rednog broja u parametru transakcije PH(RIP), a nakon toga se na isti način i otpušta (RELEASE) RIP s istim rednim brojem. Na kraju se transakcija uklanja iz modela sa TERMINATE 1.

Blokovi 15 do 16:

 KVAR LOGIC S PH(RIP)
 ADVANCE ML(MRIP,PH(RIP),3),ML(MRIP,PH(RIP),4)

definiraju putanju transakcije koja je proslijeđena na oznaku KVAR iz bloka broj 11. Tu se prvo postavlja logička sklopka odabranog RIP-a u stanje SET sa LOGIC S PH(RIP). Broj sklopke je odabran preko parametra transakcije PH(RIP). Na taj način se zabranjuje ulazak u taj RIP modul za ostale transakcije sve dok se ne riješi kvar. Vrijeme trajanja rješavanja kvara se simulira u bloku ADVANCE ML(MRIP,PH(RIP),3),ML(MRIP,PH(RIP),4) gdje se srednje vrijeme rješavanja kvara i njegovo ostupanje preuzima iz matrice MRIP preko retka s brojem zapisanim u parametru transakcije PH(RIP) i stupaca 3 i 4 respektivno.

Blokovi 17 do 20: LEAVE PH(RIP)
 RELEASE PH(RIP)
 LOGIC R PH(RIP)
 TERMINATE 1

reguliraju daljnji tok transakcije nakon što je proteklo vrijeme trajanja rješavanja kvara. Prvo se izvršava izlaz iz pripadnog STORAGE-a znanog preko zapisanog rednog broja u parametru transakcije PH(RIP) u bloku LEAVE PH(RIP), a nakon toga se otpušta RIP s istim rednim brojem u bloku RELEASE PH(RIP). Nakon otpuštanja RIP modula treba ponovo omogućiti ostalim transakcijama ulazak u njega jer je za vrijeme rješavanja kvara to bilo zabranjeno s logičkom sklopkom tog RIP-a u bloku broj 15. Resetiranje te sklopke se izvršava u bloku LOGIC R PH(RIP). Na kraju se transakcija uklanja iz modela sa TERMINATE 1.

Blokovi 21 do 24: PREKID LOGIC S 11
 SAVEVALUE INTER,20,XL
 ADVANCE 0.1
 TRANSFER ,TRAZI

simuliraju situaciju kada nadolazeća transakcija ne može dosegnuti niti jedan RIP zbog toga što su svi RIP moduli u kvaru ili zbog toga što su im redovi čekanja puni. Na oznaci PREKID se prvo aktivira sklopka broj 11 (LOGIC S 11) s kojom zatvaramo dotok novih transakcija na samom početku sve dok se ne promjene uvjeti u polju RIP-ova. Odmah nakon toga se vrši usporavanje generiranja transakcija u bloku 1 preko globalne varijable INTER (SAVEVALUE INTER,20,XL). Čim se sustav oslobodi deaktivira se ulazna sklopka i ubrzava se generiranje transakcije u bloku 7. Pošto transakciji koja je došla na oznaku PREKID nije pridružen niti jedan RIP zbog potpunog zastoja predloženo je da ju se zadrži kratki vremenski period od 0.1 minute (ADVANCE 0.1) i ponovo pusti

da proba pronaći prvi slobodni RIP (TRANSFER ,TRAZI). Transakcija će tako kružiti sve dok se ne promjene uvjeti u sustavu, a ostale se neće generirati na ulazu dulje vrijeme. U modelu ima ionako dovoljno transakcija koje su u RIP-ovima u kvaru ili u redovima čekanja za izvršavanje.

```
Blokovi 25 do 26:   TRAZI   SELECT E POMOC,1,&BRIP,1,LR,PREKID  
                   SELECT E POMOC,1,&BRIP,0,SF,PREKID
```

su blokovi za detekciju potpunog zastoja. U bloku 25 prvo se testira potpuni zastoj zbog logičkih sklopki (jedinica u kvaru). Tu se testiraju sklopke od rednog broja 1 do &BRIP na uvjet da li je stanje LR (Logic Reset) jednako 1 (E-Equal). Ako niti jedna sklopka nije u reset stanju (nepropusno satnje) tada se transakcija preusmjerava na oznaku PREKID inače se redni broj propusne sklopke zapiše u parametar transakcije s imenom POMOC. Pošto se ovo testiranje vrši s ciljem detekcije stanja kada su sve sklopke nepropusne odabran je pomoćni parametar transakcije POMOC koji se u modelu nadalje ne koristi, već se u tom slučaju transakcija propušta na slijedeći blok broj 26. U bloku 26 testira se potpuni zastoj zbog potpuno popunjenih STORAGE-a. Po istom principu kao i kod logičkih sklopki na oznaku PREKID preusmjerit će se tekuća transakcija kada ni za jedan RIP modul od rednog broja 1 do &BRIP ne vrijedi da je SF jednak 0 (Stanje Storage Full jednak 0). Ukoliko tekuća transakcija naiđe na stanje da barem jedan STORAGE nije potpuno popunjen redni broj tog STORAGE-a upisuje se u parametar POMOC te transakcije koja se dalje propušta na slijedeći blok.

```
Blokovi 27 do 28:   BLET &BTEK=1  
                   BLET &IP=4
```

inicijaliziraju varijable &BTEK i &IP prije ulaska u dio modela pod oznakom PETLJA u kojem se testira stanje svakog RIP modula. Varijabla &BTEK skoja

predstavlja redni broj tekućeg RIP-a koji se testira, a varijabla &IP predstavlja indeks parametra transakcije u koji će se zapisati stanje STORAGE-a RIP modula koji se testira. To znači da će tekuća transakcija imati parametre od rednog broja 4 naviše u kojima će se zapisivati sva stanja STORAGE-a u vremenskolm trenutku testiranja. Parametar s rednim brojem 4 sadržava stanje STORAGE-a za RIP modul 1, parametar s rednim brojem 5 s stanjem STORAGE-a za RIP modul 2 i tako redom dalje do zadnjeg RIP-a. Transakcija sa sobom nosi stanje sistema prema kojem će se ta transakcija dalje preusmjeravati.

```
Blokovi 29 do 31:   PETLJA   TEST E LR(&BTEK),1,NISTA  
                   BLET &IP=&IP+1  
                   ASSIGN &IP,&BTEK,PH
```

definiraju punjenje parametara transakcije s brojevima logičkih sklopki koje su propusne (redni broj RIP-a koji nije u kvaru). Testira se logička sklopka rednog broja zapisanog u varijabli &BTEK. Kada transakcija prvi put dođe na oznaku PETLJA tada varijabla &BTEK ima vrijednost 1. Ako je istina da je resetirano stanje trenutno razmatrane logičke sklopke (LR(&BTEK)) jednako 1 (propusno stanje) tada se transakcija propušta na slijedeći blok inače se upućuje na oznaku NISTA. U bloku BLET &IP=&IP+1 se priprema indeks parametra u koji će se upisati broj RIP-a koji nije u kvaru. Nakon toga se u definirani broj parametra transakcije upisuje redni broj RIP-a koji nije u kvaru preko varijable &BTEK (ASSIGN &IP,&BTEK,PH).

```
Blokovi 32 do 34:   NISTA     BLET &BTEK=&BTEK+1  
                   TEST LE &BTEK,&BRIP,SKOK1  
                   TRANSFER ,PETLJA
```

reguliraju kraj petlje za detekciju RIP-ova koji nisu u kvaru. Blok BLET &BTEK=&BTEK+1 povećava redni broj RIP-a (&BTEK) za jedan, a u bloku

TEST LE &BTEK,&BRIP,SKOK1 se testira da li je tekući redni broj RIP-a (&BTEK) i dalje manji ili jednak (LE) najvećem rednom broju RIP-a (&BRIP). Ako je istina tada se ide na blok TRANSFER ,PETLJA koji preusmjerava transakciju ponovo na oznaku PETLJA, inače upućuje transakciju na oznaku SKOK1.

```
Blokovi 35 do 37:   SKOK1   BLET &JP=&IP
                   SELECT MAX RIP,PH(5),PH(&JP),,R
                   TRANSFER ,POSAO
```

definiraju punjenje parametara transakcije s imenom RIP s brojem rednog broja RIP-a koji ima najveći slobodni ostatak STORAGE-a. Prvo se priprema indeks parametra transakcije do kojeg će se primjeniti funkcija MAX. Taj indeks je upravo zadnji iznos varijable &IP s kojom se izašlo iz dijela pod oznakom PETLJA. U bloku SELECT MAX RIP,PH(5),PH(&JP),,R će se u parametar s imenom RIP zapisati redni broj RIP-a koji ima najveći ostatak (R-rest) u svom STORAGE-u i između svih STORAGE-a čiji su redni brojevi zapisani u parametrima transakcije počevši od parametra s brojem 5 (PH(5)) do parametra s brojem &JP (PH(&JP)). Na taj način su se pretraživali samo ono STORAGE-i čiji RIP-ovi nisu bili u kvaru. Nakon punjenja parametra transakcije s imenom RIP transakcija nosi sa sobom broj RIP-a kojemu je dodijeljena. Ona se direktno upućuje na programsku oznaku POSAO.

Instrukcija 69: START &BINT; postavlja početnu vrijednost brojača terminiranih transakcija na &BINT.

Instrukcija 70 : PUTPIC FILE=OUT,LINES=4,(&NAKL,&ANAKL,C1,(C1/60)) otvara ispisni kanal u datoteku OUT definiranu instrukcijom 7. Stvara se ispisni niz koji ima 4 linije (LINES=4) i 4 ispisnih varijabli odnosno aritmetičkih izraza.

Instrukcije od 71 do 74:

Za nakladu od * individualiziranih otisaka odnosno * araka
bilo je potrebno *.*** minuta ili *.*** sati.

definiraju ispisnu masku. Maska se puni redom iz definiranog ispisa iz
instrukcije 70.

Instrukcija 75 :

```
PUTPIC FILE=OUT,LINES=5,
```

```
((FR(1)/10),(FR(2)/10),(FR(3)/10),(FR(4)/10),(FR(5)/10))
```

otvara ispisni kanal u datoteku OUT definiranu instrukcijom 7. Stvara se ispisni
niz koji ima 5 linija (LINES=5) i 5 aritmetičkih izraza.

Instrukcije od 76 do 79:

Iskorištenje RIP-ova:

```
RIP(1)= *.**%      RIP(2)= *.**%
```

```
RIP(3)= *.**%      RIP(4)= *.**%      RIP(5)= *.**%
```

definiraju ispisnu masku. Maska se puni redom iz definiranog ispisa iz
instrukcije 75.

Instrukcija 81:

```
PUTPIC FILE=OUT,LINES=3,
```

```
((FR(1)+FR(2)+FR(3)+FR(4)+FR(5))/5/10))
```

otvara ispisni kanal u datoteku OUT definiranu instrukcijom 7. Stvara se ispisni
niz koji ima 3 linije (LINES=3) i 1 aritmetički izraz.

Instrukcije od 82 do 84:

Ukupno iskoristenje RIP sustava je *.**% .

definiraju ispisnu masku. Maska se puni iz definiranog ispisa iz instrukcije 81.

Instrukcija 85:

```
PUTPIC FILE=OUT,LINES=5,  
((SR(1)/10),(SR(2)/10),(SR(3)/10),(SR(4)/10),(SR(5)/10))
```

otvara ispisni kanal u datoteku OUT definiranu instrukcijom 7. Stvara se ispisni niz koji ima 5 linija (LINES=5) i 5 aritmetičkih izraza.

Instrukcije od 86 do 89:

Iskoristenje STORAGE-a:

```
SR(1)= *.**%      SR(2)= *.**%  
SR(3)= *.**%      SR(4)= *.**%      SR(5)= *.**%
```

definiraju ispisnu masku. Maska se puni redom iz definiranog ispisa iz instrukcije 85.

Instrukcija 91: END

definira kraj GPSS programskog modela.

6.3.3. Eksperimentiranje s modelom

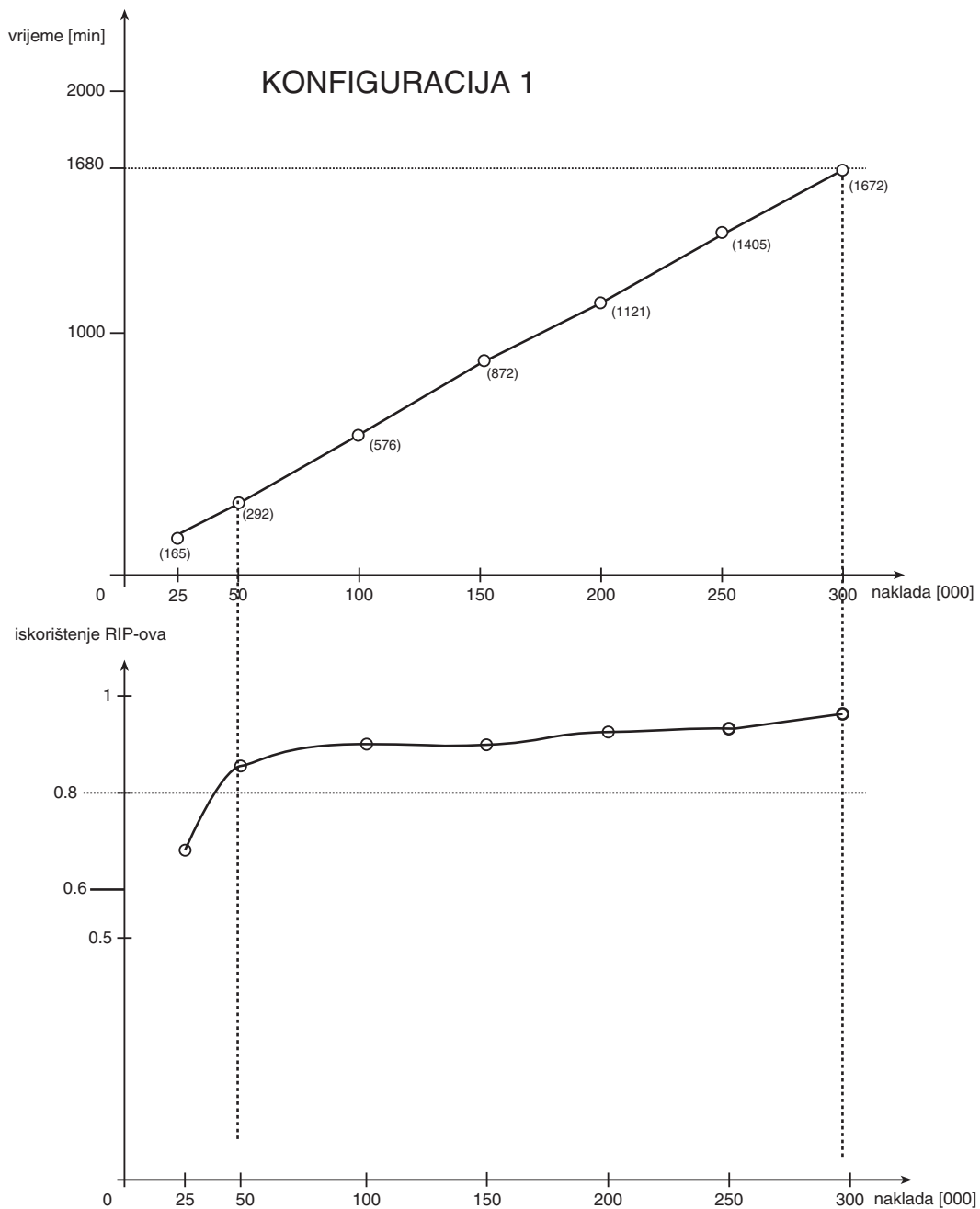
Prikazuje se eksperimentiranje nad programskim modelom koji ima za cilj naći optimalni broj i vrstu RIP modula za područje naklada tiska od 25.000 do 300.000. RIP naklada ne treba odgovarati otisnutoj nakladi jer se na jednom otisnutom arku može imati dvije i više individualne digitalne forme pa je često RIP naklada višekratnik otisnute naklade. Samo direktnim programiranjem u PostScript jeziku, koji RIP razumije, neće doći do degradacije nominalne brzine tiska jer neće biti zastoja uzrokovanih fazom programskog procesiranja.

Ovdje je prikazan eksperimentalni plan gdje su bile suprotstavljene dvije konfiguracije grafičkog sustava. Svaka ima 5 RIP modula s istom metodom distribucije PostScript zapisa:

konfiguracija 1: 1 x XEROX40, 1 x XEROX32, 2 x HP8100, 1 x HP8000;

konfiguracija 2: 1 x XEROX40, 1 x XEROX32, 3 x HP8100

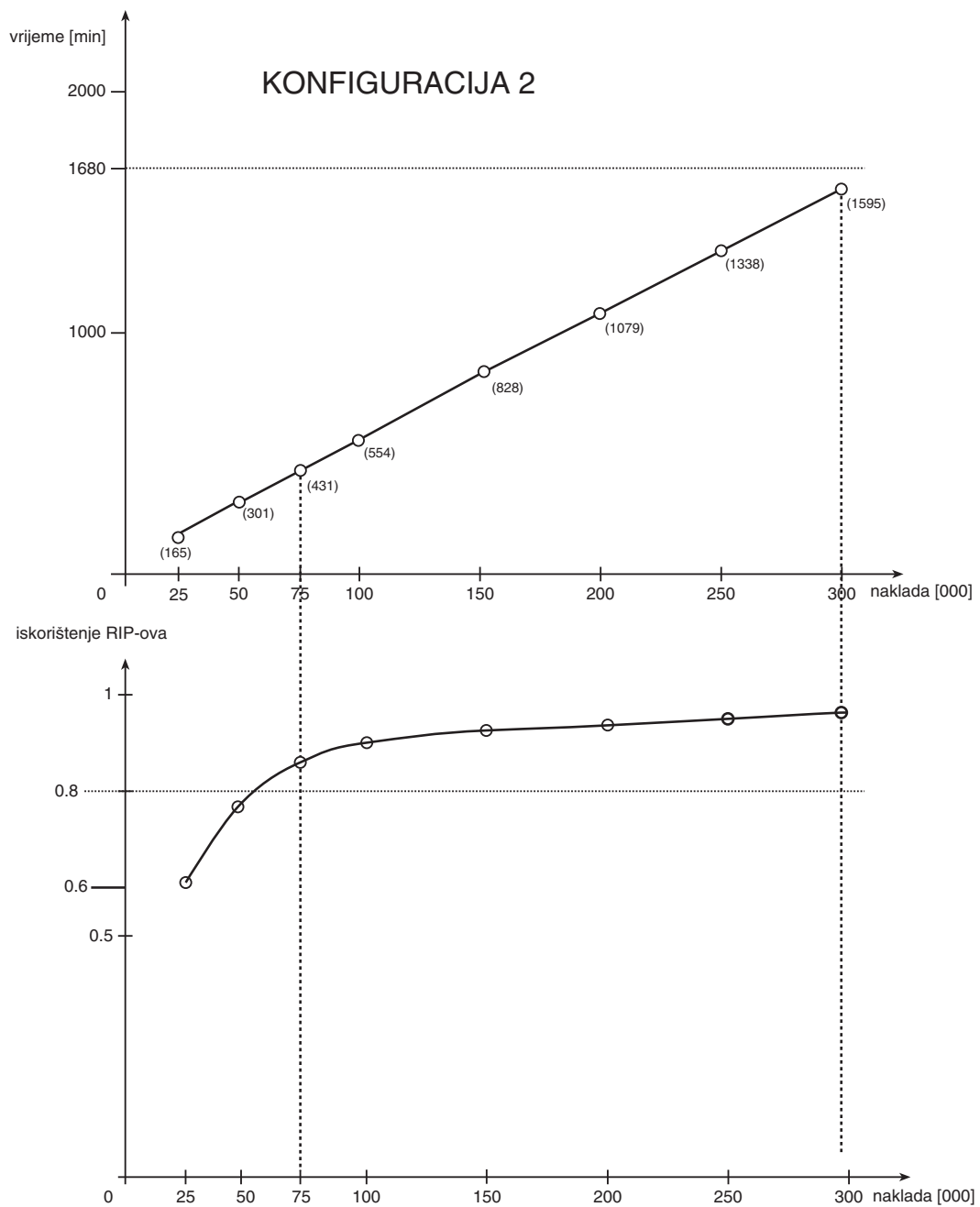
Bez obzira što je na primjer RIP modul Xerox 40 tiskarski brži od ostalih znamo da se češće kvari odnosno češće uđe u zastoj. Zbog različitih brzina, distribucija prekida i trajanja zastoja pristupilo se eksperimentima čiji su rezultati prikazani na slici 49 i 50. Željeni rok izrade svih naklada je 28 sati odnosno 1680 minuta jer je to granično vrijeme koje je preostalo nakon analogno digitalnog modela koji za nakladu od 300.000 talona troši 14 sati. Optimalni interval naklade za pojedinu konfiguraciju je određen rokom izrade i prosječnog iskorištenja svih RIP-ova.



Slika 49
Rezultati konfiguracije 1

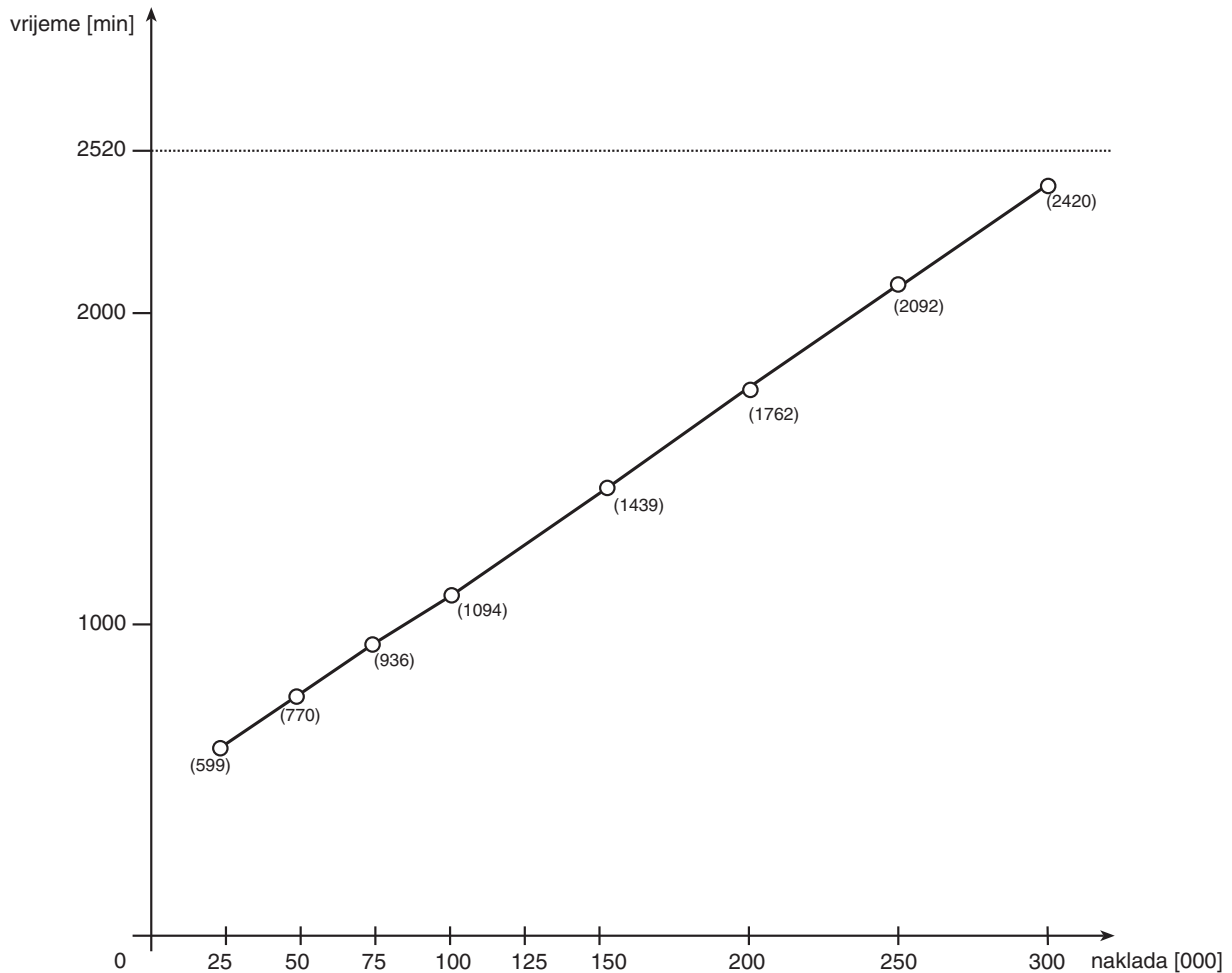
Na slici 49 se vidi da će konfiguracija 1 izvršiti posao optimalno za naklade između 50.000 do 300.000, dok za naklade ispod 50.000 treba izbaciti neki tip RIP-a i provoditi daljnje eksperimente. Naklada od 300.000 talona je upitna što se tiče roka pa je za nju očito potrebno krenuti u istraživanje proširivanja sustava.

Zamjenom jednog sporijeg RIP modula bržim kako je prikazano konfiguracijom 2 optimalni interval naklade je između 75.000 i 300.000 što je prikazano na slici 50. Vidi se da je posao od 300.000 talona s konfiguracijom 2



Slika 50
Rezultati konfiguracije2

završen ranije za 77 minuta u odnosu na konfiguraciju 1 što daje dovoljnu sigurnost stizanja roka.



Slika 51
Grafikon ukupnog potrebnog vremena za pojedine naklade cjelokupnog hibridnog modularnog sustava

Na slici 51 prikazan je grafikon ukupnog potrebnog vremena za pojedine naklade cjelokupnog hibridnog modularnog sustava. To je suma vremena s grafikona za model analogno digitalnog sustava za izradu mastera (slika 42) i modela modularne individualizacije s konfiguracijom 2 (slika 50).

Demonstrirala se prva realna primjena matematičkog modeliranja i simuliranja varijabilnog digitalnog tiskarskog sustava. Istražene su 2 konfiguracije digitalnog tiska za naklade od 25 do 300 tisuća otisaka u krajnjem vremenu od 1680 minuta. Traži se bolja varijanta u tom intervalu interesa, a da se ne ulazi u nove investicije rezervnih strojeva. U obzir su uzeti zastoji kao stohastička varijabla prema podacima mjerenja tokom dvije godine rada. Dva sustava su postavljena kako bi u slučaju većeg kvara bilo moguće prijeći na suprotnu konfiguraciju.

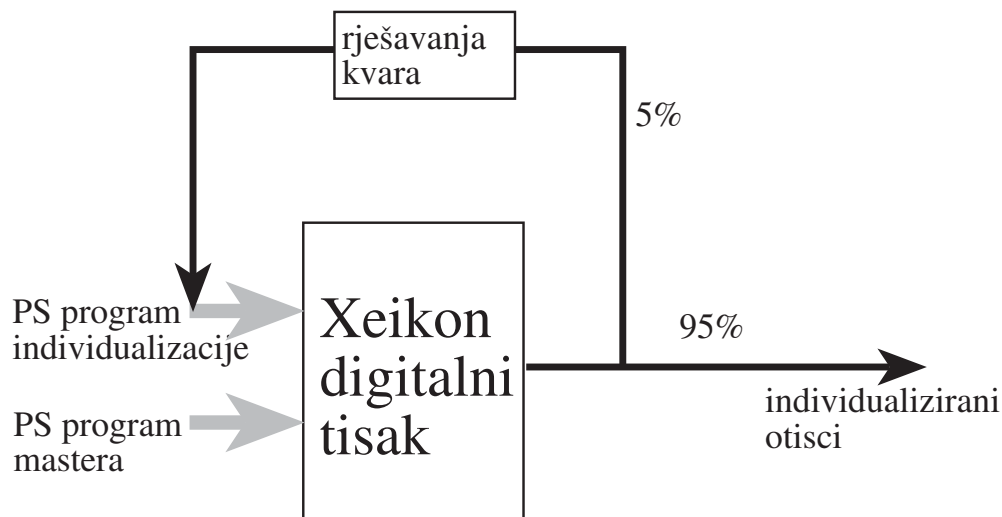
Ovi rezultati se mogu odmah primjenjivati, a najveća dobit je u stvaranju sigurnosti da će proizvodnja uspješno završiti u zadanom periodu. Ponuđeni model i programsko rješenje koristi se za izučavanje još veće tiraže s drugim parametrima vremena i dodatnim RIP sustavima. To su pripreme za još veći tisak individualizacije, ali s minimalnim ulaganjem u nove strojeve s ciljem da se odredi konkurentna ponuda za nove projekte.

7. Model digitalnog sustava za individualizaciju s novim rasterskim procedurama

Individualizacija forme može se izvoditi istovremeno kada i master. To je jedino moguće u digitalnom tisku koji tiska istovremeno četiri boje obostrano. Napraviti će se model konfiguracije koja koristi tehnologiju prikazanu na slici 1. Radi se o stroju koji koristi rolu širine 50 cm. Zadatak je isti kao kod hibridnog modela. Digitalni tisak treba raditi što kontinuiranije bez zastoja, kako bi iskoristivost bila maksimizirana [23].

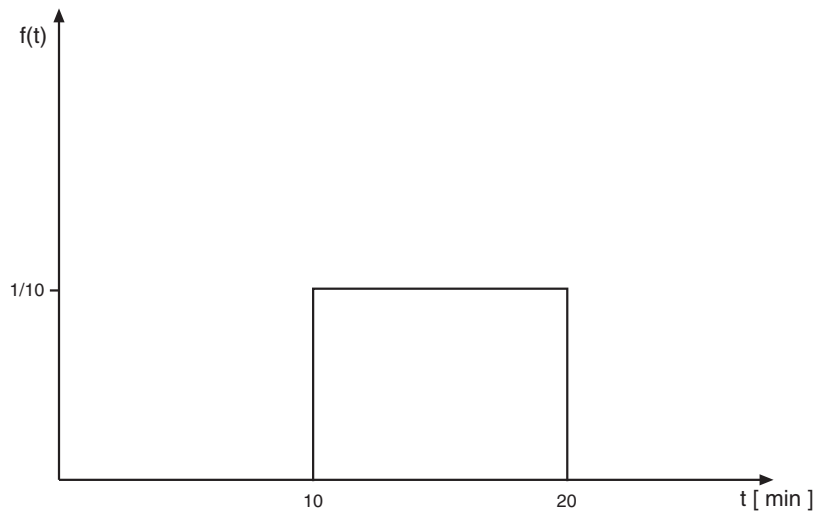
7.1 Model digitalnog sustava s jednim modulom

Na slici 52 prikazana je shema konfiguracije s jednim modulom. U model ulaze dvije vrste PostScript zapisa. Jedan koji definira master prednje i stražnje strane i drugi koji generira varijabilne informacije koje će ići na jednu stranu forme.



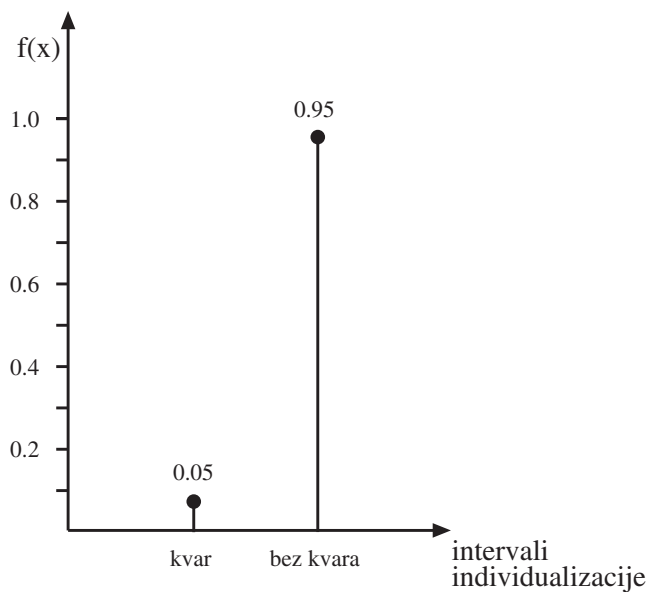
Slika 52
Model digitalnog sustava s jednim modulom

Prevođenje master PS zapisa u bitmapirani zapis događa se jedanput. Stvoreni bitmapirani zapis ispisuje se na LED polje uređaja konstantno za cijelo vrijeme tiska. PS zapisi za individualizaciju razbijeni su na PS zapise koji definiraju interval po interval individualizacije. Na taj način sačuvana je kontrola kao i jednostavnost ponavljanja uslijed kvara.



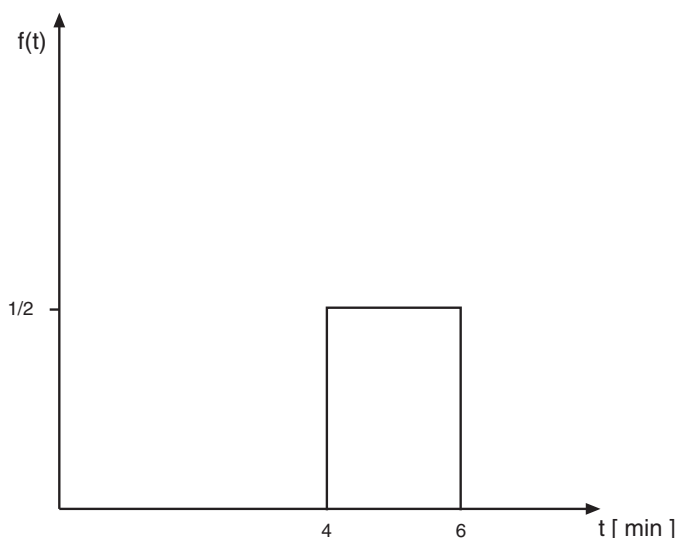
Slika 53
Vrijeme rješavanja kvara

Ukoliko dođe do zastoja stroja zbog kvara, tada treba početi teći vrijeme rješavanja kvara. Snimanjem takve konfiguracije dvije godine došlo se do uniformne funkcije vjerojatnosti vremena rješavanja kvara prikazane na slici 53.



Slika 54
Distribucija pojavljivanja kvarova

Distribucija pojavljivanja kvarova prikazana je na slici 54. Nakon rješavanja kvara treba nastaviti tisak ili ponoviti tisak do kraja tekućeg intervala individualizacije ukoliko ima uništenih talona.



Slika 55
Vrijeme zamjene role

Svaka rola ima zadanu dužinu. Kada se dođe do kraja role, mora proteći vrijeme zamjene role. Snimanjem se dobila uniformna funkcija vjerojatnosti vremena zamjene role prikazane na slici 55.

7.1.1. Rješenje modela u GPSS-u

Programski model će poslužiti eksperimentiranju i mjerenju odziva sustava s različitim nakladama. Na slici 56 prikazan je GPSS programski model s jednim Xekon modulom. Korisnik programa puni varijablu NAKL sa željenim brojem individualiziranih formi (talona). Varijablom BINT definiran je željeni broj intervala individualizacije zbog potreba kontrole. Ujedno je i to broj transakcija koji ulaze u model. Varijabla BTAL definira broj formi (ovdje A5) na arku u

```
1          *MODEL SA JEDNIM XEIKONOM
2  SIMULATE
3  INTEGER &NAKL,&BINT,&BTAL,&I
4  REAL &ANAKL,&BNAKL,&BRZCM,&DUZAR,&BTXEIK,&DROLE,&BPR
5  PUTPIC
6  OUnesi nakladu individualiziranih formi (npr.300000)
7  GETLIST &NAKL          naklada individualiziranih formi (A5)
8  OUT  FILEDEF 'XEIKON1.OUT'
9  LET &BINT=1000          broj ontervala= broj transakcija
10 LET &BTAL=3            broj formi A5 (21cm) na arku u smjeru role
11 LET &ANAKL=&NAKL/&BTAL  naklada u arcima cijele naklade
12 LET &BNAKL=&ANAKL/&BINT  naklada u arcima 1 intervala
13 LET &BRZCM=735        735 cm/min brzina Xeikona50
14 LET &DUZAR=21         duljina arka (cm) u smjeru tiska
15 LET &BTXEIK=(&BNAKL*&DUZAR)/&BRZCM vrijeme potrebno za 1 interval
16 LET &DROLE=84000      efekt. duljina role (cm)
17 LET &BPR=&DROLE/(&BNAKL*&DUZAR) broj intervala po jednoj roli
18 LET &I=0
19 STORAGE S(DISK),50
20 INITIAL XL(INTER),1
21 1          GENERATE XL(INTER),,,&BINT
22 2          ENTER DISK
23 3          SEIZE XEIK
24 4          ADVANCE &BTXEIK
25 5          BLET &I=&I+1
26 6          TEST LE &I,&BPR,PAUZA
27 7          TRANSFER .05,,KVAR
28 8          RELEASE XEIK
29 9          LEAVE DISK
30 10         TERMINATE 1
31 11 KVAR    ADVANCE 15,5
32 12         RELEASE XEIK
33 13         LEAVE DISK
34 14         TERMINATE 1
35 15 PAUZA   BLET &I=0
36 16         ADVANCE 5,1
37 17         RELEASE XEIK
38 18         LEAVE DISK
39 19         TERMINATE 1
40 START &BINT
41 PUTPIC FILE=OUT,LINES=4,(&NAKL,C1,(C1/60))
42
43 Za nakladu od * individualiziranih formi
44
45 bilo je potrebno *.*.* minuta ili *.*.* sati.
46
47 PUTPIC FILE=OUT,LINES=3,((FR(XEIK)/10))
48
49 Iskoristenje Xeikona: *.*.*%
50
51 PUTPIC FILE=OUT,LINES=3,((SR(DISK)/10))
52
53 Iskoristenje STORAGE-a DISK na PrintStreameru: *.*.*%
54
55 PUTPIC FILE=OUT,LINES=3,(N(PAUZA)+1)
56
57 Potrosilo se * rola.
58
59      END
```

Slika 56

GPSS programski model s jednim Xeikon modulom

smjeru role, ANAKL nakladu u arcima čitavog posla, BNAKL nakladu u arcima jednog intervala individualizacije, BRZCM brzinu u centimetrima po minuti i DROLE definira duljinu role.

Transakcija koja ulazi u model predstavlja PostScript program jednog intervala individualizacije. PostScript zapis koji definira master nije predstavljen u modelu jer jednom izvedena bitmapa mastera ostaje tokom čitavog procesa individualizacije nepromijenjena i ne konzumira vrijeme koje bi utjecalo na simulacijsko vrijeme. Tiskovni disk se napuni s bitmapom prvog arka (3 individualizacijske forme) i prazni se na LED polje stroja. Nakon toga dolazi na red slijedeći arak i tako sve do kraja tekućeg intervala individualizacije. Sve se izvodi nominalnom brzinom tiska bez zastoja. To je rezultat prve transakcije nakon koje slijedi slijedeća.

Varijabla I cijelo vrijeme vodi evidenciju koliko se intervala otisnulo na roli da bi se aktivirao programski kod na labeli PAUZA za zamjenu role. S naredbom TRANSFER .05, KVAR smo transferirali 5% transakcija na labelu KVAR gdje će se izvršiti zadržavanje transakcije dok se kvar ne riješi s uračunatim vremenom završetka tiska intervala.

7.1.2 Opis rutina i veza radi vođenja eksperimenata

Instrukcija 2: SIMULATE; izvršavanje programskog modela nakon uspješne kompilacije.

Instrukcija 3: INTEGER &NAKL,&BINT,&BTAL,&I

deklaracija cijelobrojnih varijabli:&NAKL za nakladu talona za koju se provodi eksperiment, &BINT broj intervala što je ujedno i broj transakcija koje ulaze u model, &BTAL za broj formi na arku, &I za brojač broja intervala po jednoj roli.

Instrukcija 4:

REAL &ANAKL,&BNAKL,&BRZCM,&DUZAR,&BTXEIK,&DROLE,&BPR

deklaracija realnih varijabli: &ANAKL za nakladu u arcima, &BNAKL za nakladu u arcima jednog intervala, &BRZCM za brzinu u centimetrima po minuti, &DUZAR za duljinu arka (cm) u smjeru role, &BTXEIK za vrijeme tiska potrebno za 1 interval, &DROLE za efektivnu duljine role u centimetrima, &BPR za broj intervala po jednoj roli.

Instrukcije 5 do 7: PUTPIC; Ispisuje poruku "Unesi nakladu individualiziranih formi (npr.300000)" na ekran, a instrukcija GETLIST &NAKL puni varijablu &NAKL s tastature

Instrukcija 8: OUT FILEDEF 'XEIKON1.OUT'; definira izlaznu datoteku XEIKON1.OUT u koju će se ispisivati željeni rezultati simulacije

Instrukcija 9: LET &BINT=1000; broj intervala što je ujedno i broj transakcija koje ulaze u model;

Instrukcija 10: LET &BTAL=3; broj formi A5 (21cm) na arku u smjeru role

Instrukcija 11: LET &ANAKL=&NAKL/&BTAL; naklada u arcima cijele naklade

Instrukcija 12: LET &BNAKL=&ANAKL/&BINT; naklada u arcima 1 intervala

Instrukcija 13: LET &BRZCM=735; 735 cm/min je brzina Xeikona50

Instrukcija 14: LET &DUZAR=21; duljina arka (cm) u smjeru tiska

Instrukcija 15: LET &BTXEIK=(&BNAKL*&DUZAR)/&BRZCM;
vrijeme potrebno za tisak 1 intervala

Instrukcija 16: LET &DROLE=84000 ; efektivna duljina role (cm)

Instrukcija 17: LET &BPR=&DROLE/(&BNAKL*&DUZAR)
broj intervala po jednoj roli

Instrukcija 18: LET &l=0; inicijalizacija brojača broja intervala po jednoj roli

Instrukcija 19: STORAGE S(DISK),50 ; definiranje kapaciteta reda čekanja
PostScript zapisa na RIP disku

Instrukcija 20: INITIAL XL(INTER),1
postavljanje inicijalne vrijednosti variable INTER kao broja s
pomičnim zarezom (XL). U programu nam služi kao varijabla za
definiranje vremenskog intervala generiranja transakcija u bloku
GENERATE.

Blok 1: GENERATE XL(INTER),,,&BINT ; transakcije će se
generirati iz ovog bloka svakih INTER minuta. Broj generiranih
transakcija je limitiran na &BINT.

Blokovi od 2 do 4: ENTER DISK
 SEIZE XEIK
 ADVANCE &BTXEIK

simuliraju ulazak transakcije u STORAGE DISK (ENTER DISK) gdje je definiran kapacitet reda čekanja PostScript zapisa za svaki interval. Ako ima slobodog mjesta transakcija potroši jednu lokaciju na disku i zauzima FACILITY Xeikon (SEIZE XEIK). Nakon zauzimanja transakcija se zadržava u bloku ADVANCE &BTXEIK točno &BTXEIK minuta što je ujedno vrijeme potrebno za tisak jednog intervala.

Blokovi od 5 do 6: BLET &l=&l+1
 TEST LE &l,&BPR,PAUZA

reguliraju situaciju kada za vrijeme tiska treba promjeniti rolu. Varijabla l vodi evidenciju koliko se intervala otisnulo na roli da bi se u datom trenutku počelo trošiti vrijeme za zamjenu role. Čim je transakcija izašla iz bloka 4 na bloku 5 se povećava brojač intervala za jedan (BLET &l=&l+1). U bloku TEST LE &l,&BPR,PAUZA da li je brojač intervala &l manji ili jednak (LE) mogućem broju untervala na jednoj roli. Ukoliko je to istina, tada se transakcija propušta na slijedeći blok inače se prosljeđuje na programsku oznaku PAUZA.

Blok 7: TRANSFER .05,,KVAR ; definira stohastičko grananje na oznaku KVAR. Pet posto transakcija koje uđu u ovaj blok odlaze na dio programa KVAR, a ostale nastavljaju na slijedeći blok.

Blokovi od 8 do 10: RELEASE XEIK
 LEAVE DISK
 TERMINATE 1

definiraju da transakcija otpusti FACILITY XEIK, a zatim oslobodi jednu lokaciju iz STORAGE DISK. Na kraju dolazi do terminiranja transakcije što umanjuje brojač terminiranih transakcija na bloku START pri kraju.

Blokovi od 11 do 14: KVAR ADVANCE 15,5
 RELEASE XEIK
 LEAVE DISK
 TERMINATE 1

simuliraju situaciju kada je transakcija stohastički došla na oznaku KVAR gdje prvo počinje teći vrijeme zadržavanja transakcije za rješavanje kvara 15+-5

minuta. Nakon toga se otpušta FACILITY XEIK, a zatim osloboda jedna lokacija iz STORAGE-a DISK (RELEASE XEIK). Na kraju dolazi do terminiranja transakcije.

```
Blokovi od 15 do 19: PAUZA  BLET &l=0
                        ADVANCE 5,1
                        RELEASE XEIK
                        LEAVE DISK
                        TERMINATE 1
```

simuliraju situaciju kada je transakcija usmjerena na oznaku PAUZA jer je dosegnut najveći mogući broj intervala na jednoj roli. Prvo se ponovo inicijalizira brojač intervala po roli na nulu (BLET &l=0). Nakon toga počinje teći vrijeme potrebno za zamjenu role 5+-1 minuta (ADVANCE 5,1), otpušta se modul Xeikon (RELEASE XEIK), napuštanje jednog mjesta iz STORAGE-a diska (LEAVE DISK) i konačno napuštanje transakcije iz modela (TERMINATE 1).

Instrukcija 40: START &BINT

postavlja početnu vrijednost brojača terminiranih transakcija. Sa varijablom &BINT postignuta je mogućnost eksperimentiranja s različitim količinama transakcija. Cilj je da model ima dovoljan broj transakcija za pravilno funkcioniranje svih stohastičkih varijabli unutar modela.

Instrukcija 41: PUTPIC FILE=OUT,LINES=4,(&NAKL,C1,(C1/60))
otvara ispisni kanal u datoteku OUT definiranu instrukcijom 8.
Stvara se ispisni niz koji ima 4 linije (LINES=4) i 3 ispisnih varijabli odnosno aritmetičkih izraza.

Instrukcije 42 do 46:

Za nakladu od * individualiziranih formi
bilo je potrebno *.*** minuta ili *.*** sati.

definiraju ispisnu masku. Maska se puni redom iz definiranog ispisa iz
instrukcije 41.

Instrukcija 47: `PUTPIC FILE=OUT,LINES=3,((FR(XEIK)/10))`
otvara ispisni kanal u datoteku OUT definiranu instrukcijom 8.
Stvara se ispisni niz koji ima 3 linije (LINES=3) i jedan
aritmetički niz.

Instrukcije 48 do 50:

Iskoristenje Xeikona: `***.**%`

definiraju ispisnu masku. Maska se puni iz definiranog ispisa iz instrukcije 47.

Instrukcija 51: `PUTPIC FILE=OUT,LINES=3,((SR(DISK)/10))`
otvara ispisni kanal u datoteku OUT definiranu instrukcijom 8.
Stvara se ispisni niz koji ima 3 linije (LINES=3) i jedan
aritmetički niz.

Instrukcije 52 do 54:

Iskoristenje STORAGE-a DISK na PrintStreameru: `***.**%`

definiraju ispisnu masku. Maska se puni iz definiranog ispisa iz instrukcije 51.

Instrukcija 55: PUTPIC FILE=OUT,LINES=3,(N(PAUZA)+1)
otvara ispisni kanal u datoteku OUT definiranu instrukcijom 8.
Stvara se ispisni niz koji ima 3 linije (LINES=3) i jedan
aritmetički niz.

Instrukcije 56 do 58:

Potrosilo se * rola.

definiraju ispisnu masku. Maska se puni iz definiranog ispisa iz instrukcije 55.

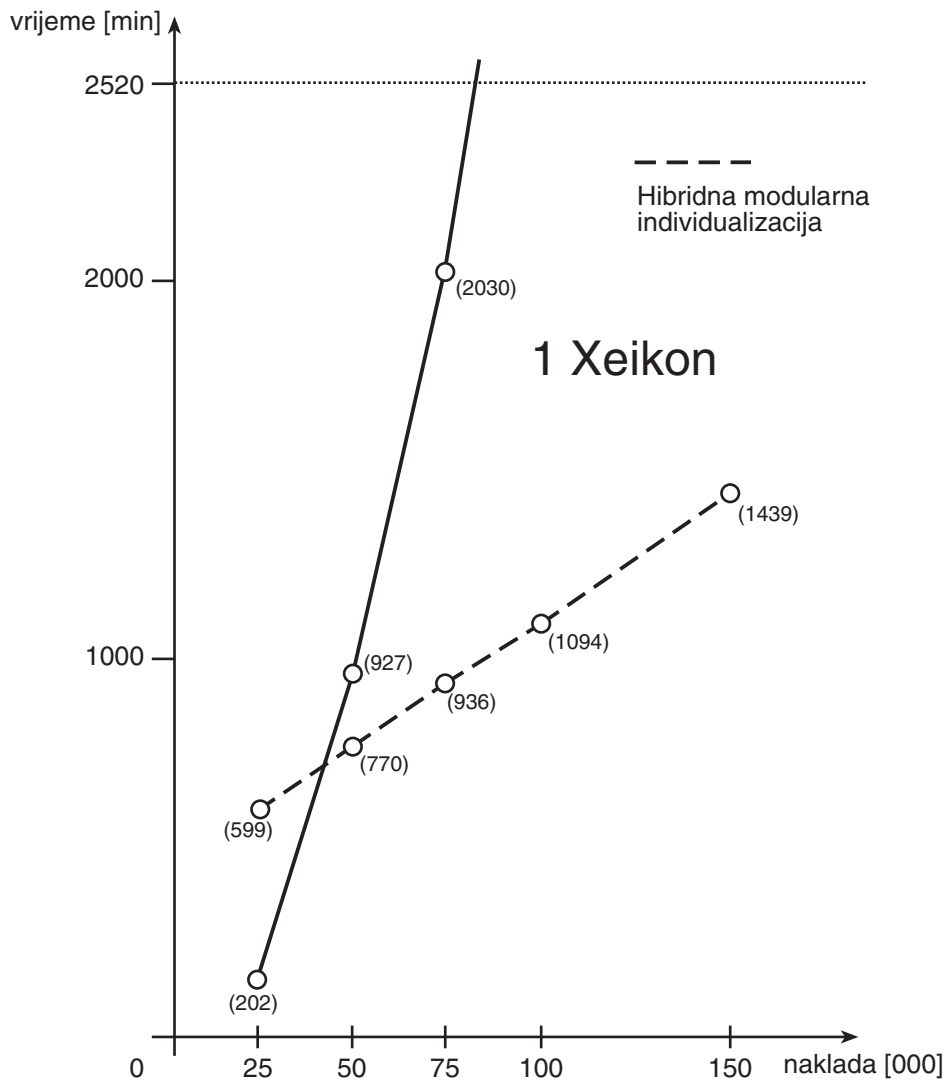
Instrukcija 59: END ; definira kraj GPSS programskog modela.

7.1.3 Eksperimentiranje s modelom

Svaki puta kada se starta programski model, unosi se naklada finalnih individualiziranih formi preko varijable NAKL, a željene rezultate ispisuje u zasebnu datoteku. Za naše eksperimentiranje interesantno je vrijeme izvršavanja posla za zadanu nakladu. To se dobiva preko C1 systemske varijable GPSS vremenskog sata koja se ispisuje nakon svakog startanja programskog modela.

Eksperimentiranja se izvode unutar zadanog vremena od 2520 minuta odnosno 42 sata. Na slici 57 prikazan je grafikon vremena trajanja individualizacije s digitalnim modelom s jednim Xeikon modulom usporedno s hibridnim modularnim modelom. Naklade do 75.000 bi se mogle izvoditi u zadanom vremenu od 42 sata ovim modelom s time da je za naklade do 50.000 ovaj model brži od hibridnog.

Osim mjerenja vremena program ispisuje i iskorištenje Xeikona i iskorištenje diskova na kojem se formira red čekanja. To služi za bolje vođenje pravaca eksperimentiranja, ali najviše je služilo u toku gradnje modela kada se nije znalo



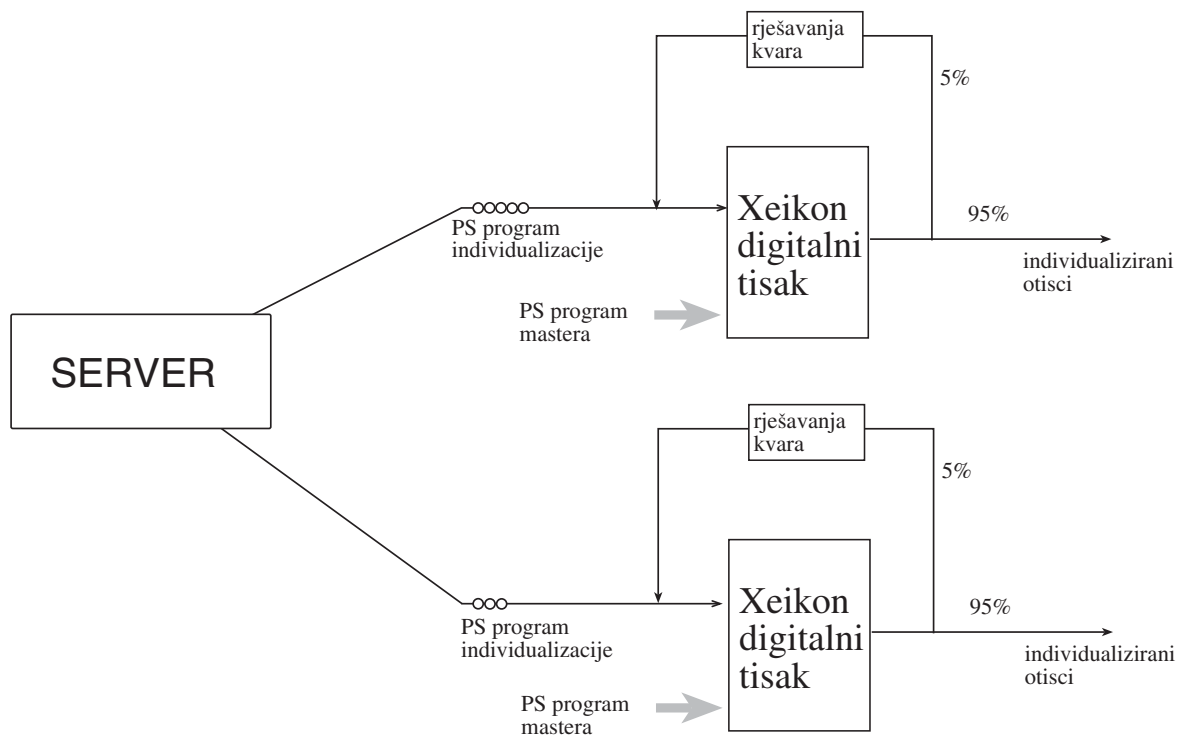
Slika 57

Vrijeme trajanja individualizacije s modelom s jednim XEIKON modulom usporedno s modelom hibridnog modularnog sustava

kojom brzinom puniti model i gdje treba ugrađivati logičke sklopke za usporavanje ulaza u model. Program ispisuje i broj potrošenih rola za lakše obračunavanje troška i predviđanje narudžbe za takve projekte.

7.2 Model digitalnog sustava s dva modula

Za potrebe daljnjeg istraživanja napravio se digitalni model s 2 Xeikon modula prikazan na slici 58. Na server računalu se generiraju PostScript zapisi za svaki interval individualizacije.



Slika 58
Model digitalnog sustava s dva modula

U gradnji modela master PostScript zapis se generira nezavisno od servera i to samo jedanput na početku posla na svakom modulu zasebno s istovrsnim bitmapiranim zapisom u master segmentu memorije. U ovom hipotetskom modelu je za svaki Xeikon primjenjena ista stohastika za vrijeme rješavanja kvara, distribuciju pojavljivanja kvarova i vrijeme zamjene role.

7.2.1. Rješenje modela u GPSS-u

Programski model će poslužiti istom eksperimentiranju i mjerenju odziva sustava s različitim nakladama kao i kod modela s jednim modulom. Na slici 59 prikazan je GPSS programski model s dva Xeikon modula.

Programski model je pisan tako da se u budućnosti lako dodaju novi moduli. Transakcija modela je nalog za tisak jednog intervala individualizacije koji se mora zbog kontrole i završiti na modulu na kojem je i započet. To se postiglo tako da se u parametar transakcije s imenom XEIK upisuje broj modula na koji će transakcija otići. Za svaki modul se matično vode brojači intervala po roli koji aktiviraju promjenu role nezavisno svaki za svoji modul.

U dijelu programa s oznakom TRAŽI, detektira se koji je modul slobodan, odnosno koji je u prekidu. Svaki put kada transakcija prođe kroz RELEASE PH(XEIK) postavi se sistemska Facility varijabla F u stanje 0 pa će naredba SELECT odabrati prvi s liste koji ima F jednak nuli. Ukoliko su svi moduli zauzeti, ide se na oznaku PREKID gdje se aktivira sklopka SERVER i aktivira čekanje od jedne minute i ponovo šalje transakciju na oznaku TRAŽI. Aktiviranjem sklopke SERVER smo zaustavili generiranje transakcija iz bloka GENERATE s namjerom sprječavanja zagušenja u toj točki modela. Svaki puta kada se izlazi iz regularno završene individualizacije, iz rješavanja kvara ili pauze za promjenu role, resetira se logička sklopka SERVER.

U matrici BRIN su implementirani brojači intervala po roli jer se na taj način zna kada treba doći do promjene role. S naredbom MSAVEVALUE se pripadni brojači unutar matrice BRIN povećavaju za jedan nakon svakog završetka intervala individualizacije. Nakon toga se odmah testira da li je upravo povećani brojač premašio dozvoljeni broj intervala po roli definiran varijablom BPR. Ako je, transakcija se šalje na oznaku PAUZA gdje se resetira tekući brojač i zadrži transakcija za potrebno vrijeme promjene role. Tek onda se terminira transakcija tekućeg intervala tiska.


```
1          *MODEL SA DVA ILI VISE XEIKONA
2 SIMULATE
3 INTEGER &NAKL,&BINT,&BTAL,&BXEIK
4 REAL &ANAKL,&BNAKL,&BRZCM,&DUZAR,&BTXEIK,&DROLE,&BPR
5 PUTPIC
6 UNesi nakladu individualiziranih formi (npr.300000)
7 GETLIST &NAKL          naklada formi (A5)
8 OUT   FILEDEF 'XEIKON2.OUT'
9 LET &BINT=1000          broj intervala=broj transakcija
10 LET &BTAL=3            broj formi A5 (21cm) na arku u smjeru role
11 LET &ANAKL=&NAKL/&BTAL  naklada u arcima cijele naklade
12 LET &BNAKL=&ANAKL/&BINT  naklada u arcima 1 intervala
13 LET &BRZCM=735        735 cm/min brzina Xeikona50
14 LET &DUZAR=21         duljina arka (cm) u smjeru tiska
15 LET &BTXEIK=(&BNAKL*&DUZAR)/&BRZCM vrijeme potrebno za 1 interval
16 LET &DROLE=84000      efekt. duljina role (cm)
17 LET &BPR=&DROLE/(&BINT*&DUZAR) broj intervala po jednoj roli
18 LET &BXEIK=2          broj Xeikona
19 BRIN  MATRIX MH,2,1   matrica brojaca interv.za promjenu role
20 INITIAL  MH$BRIN(1,1),0
21 INITIAL  MH$BRIN(2,1),0
22 INITIAL XL(INTER),1
23 1          GENERATE XL(INTER),,,&BINT
24 2          GATE LR SERVER
25 3 TRAZI   SELECT E XEIK,1,&BXEIK,0,F,PREKID
26 4          SEIZE PH(XEIK)
27 5          ADVANCE &BTXEIK
28 6          MSAVEVALUE BRIN+,PH(XEIK),1,1,MH
29 7          TEST LE MH(BRIN,PH(XEIK),1),&BPR,PAUZA
30 8          TRANSFER .05,,KVAR
31 9          RELEASE PH(XEIK)
32 10         LOGIC R SERVER
33 11         TERMINATE 1
34 12 KVAR   ADVANCE 15,5
35 13         RELEASE PH(XEIK)
36 14         LOGIC R SERVER
37 15         TERMINATE 1
38 16 PAUZA  MSAVEVALUE BRIN,PH(XEIK),1,0,MH
39 17         ADVANCE 5,1
40 18         RELEASE PH(XEIK)
41 19         LOGIC R SERVER
42 20         TERMINATE 1
43 21 PREKID LOGIC S SERVER
44 22         ADVANCE 1
45 23         TRANSFER ,TRAZI
46 46        START &BINT
47 PUTPIC FILE=OUT,LINES=4,(&NAKL,C1,(C1/60))
48
49 Za nakladu od * individualiziranih formi
50 bilo je potrebno *.*.*** minuta ili *.*.*** sati.
51
52 PUTPIC FILE=OUT,LINES=4,((FR(1)/10),(FR(2)/10))
53
54 Iskoristenje Xeikona1: *.*.***%
55 Iskoristenje Xeikona2: *.*.***%
56
57 PUTPIC FILE=OUT,LINES=3,(N(PAUZA)+&BXEIK)
58
59 Potrosilo se * rola.
60
61 END
```

Slika 59

GPSS programski model s dva Xeikon modula

7.2.2. Opis rutina i veza radi vođenja eksperimenata

Instrukcija 2: SIMULATE; izvršavanje programskog modela nakon uspješne kompilacije.

Instrukcija 3: INTEGER &NAKL,&BINT,&BTAL,&BXEIK

deklaracija cijelobrojnih varijabli:&NAKL za nakladu talona za koju se provodi eksperiment, &BINT broj intervala što je ujedno i broj transakcija koje ulaze u model, &BTAL za broj formi na arku, &BXEIK za broj Xeikon modula.

Instrukcija 4:

REAL &ANAKL,&BNAKL,&BRZCM,&DUZAR,&BTXEIK,&DROLE,&BPR

deklaracija realnih varijabli: &ANAKL za nakladu u arcima, &BNAKL za nakladu u arcima jednog intervala, &BRZCM za brzinu u centimetrima po minuti, &DUZAR za duljinu arka (cm) u smjeru role, &BTXEIK za vrijeme tiska potrebno za 1 interval , &DROLE za efektivnu duljine role u centimetrima, &BPR za broj intervala po jednoj roli.

Instrukcije 5 do 7: PUTPIC; Ispisuje poruku "Unesi nakladu individualiziranih formi (npr.300000)" na ekran, a instrukcija GETLIST &NAKL puni varijablu &NAKL s tastature;

Instrukcija 8: OUT FILEDEF 'XEIKON2.OUT' ; definira izlaznu datoteku XEIKON2.OUT u koju će se ispisivati željeni rezultati simulacije;

Instrukcija 9: LET &BINT=1000 ; broj intervala što je ujedno i broj transakcija koje ulaze u model;

Instrukcija 10: LET &BTAL=3 ; broj formi A5 (21cm) na arku u smjeru role

Instrukcija 11: LET &ANAKL=&NAKL/&BTAL; naklada u arcima cijele naklade

Instrukcija 12: LET &BNAKL=&ANAKL/&BINT; naklada u arcima 1 intervala

Instrukcija 13: LET &BRZCM=735; 735 cm/min je brzina Xeikona50

Instrukcija 14: LET &DUZAR=21; duljina arka (cm) u smjeru tiska

Instrukcija 15: LET &BTXEIK=(&BNAKL*&DUZAR)/&BRZCM;
vrijeme potrebno za tisak 1 intervala

Instrukcija 16: LET &DROLE=84000 ; efektivna duljina role (cm)

Instrukcija 17: LET &BPR=&DROLE/(&BNAKL*&DUZAR)
broj intervala po jednoj roli

Instrukcija 18: LET &BXEIK=2; broj Xeikon modula

Instrukcija 19: BRIN MATRIX MH,2,1 ; definicija matrice MRIP s 2
redaka i 1 kolone čije su vrijednosti brojevi s pomičnim
zarezm (ML- floating point Matrix)

Instrukcije 20 do 21: INITIAL MH\$BRIN(1,1),0
INITIAL MH\$BRIN(2,1),0

definiraju inicijalne vrijednosti matrice BRIN i to tako da svaki redak matrice
definira inicijalne vrijednosti brojača intervala na jednoj roli za svaki Xeikon
modul posebno (postoji samo jedan stupac).

Instrukcija 22: INITIAL XL(INTER),1

postavljanje inicijalne vrijednosti variable INTER kao broja s pomičnim zarezom (XL). U programu nam služi kao varijabla za definiranje vremenskog intervala generiranja transakcija u bloku GENERATE.

Blok 1: GENERATE XL(INTER),,,&BINT ; transakcije će se generirati iz ovog bloka svakih INTER minuta. Broj generiranih transakcija je limitiran na &BINT.

Blok 2: GATE LR SERVER

definira glavna ulazna logička vrata koja su regulirana logičkom sklopkom s imenom SERVER. Tokom izvršavanja modela sklopka SERVER može ući u stanje SET jer se tada žele zatvoriti ova logička vrata da ne mogu nove transakcije ulaziti u model.

Blok 3: TRAZI SELECT E XEIK,1,&BXEIK,0,F,PREKID

služi za detekciju slobodnog Xeikon modula , odnosno koji je u prekidu. Ukoliko je FACILITY s PH(XEIK) zauzet tada je sistemska varijabla F postavljena na 1. Čim kasnije u modelu transakcija prođe kroz blok RELEASE PH(XEIK) tada se varijabla F pripadnog rednog broja Xeikona postavlja na 0. Ovaj SELECT blok ispituje, redom od prvog do &BXEIK, koji prvi FACILITY ima F varijablu jednaku (E) nuli. Ako je nađen tada se njegov redni broj upisuje u parametar tekuće transakcije PH(XEIK) i ona se propušta u slijedeći blok. Ukoliko niti jedan nema to stanje tekuća se transakcija usmjerava na oznaku PREKID.

Blokovi 4 do 5: SEIZE PH(XEIK)
 ADVANCE &BTXEIK

simuliraju zauzimanje FACILITY-a zapisanog u parametru transakcije SEIZE PH(XEIK) i njeno zadržavanje za potrebno vrijeme tiska jednog intervala ADVANCE &BTXEIK.

Blok 6: MSAVEVALUE BRIN+,PH(XEIK),1,1,MH

definira povećanje brojača intervala na roli za jedan unutar matrice BRIN (BRIN+) čiji je redak definiran preko broja zapisanog u parametru transakcije PH(XEIK), a stupac eksplicitno 1.

Blok 7: TEST LE MH(BRIN,PH(XEIK),1),&BPR,PAUZA

testira da li je upravo povećani brojač iz matrice BRIN premašio dozvoljeni broj intervala po roli definiran varijablom BPR. Ako je, transakcija se šalje na oznaku PAUZA. Ako je promatrani brojač manji ili jednak (LE) od &BPR transakcija se propušta na slijedeći blok.

Blok 8: TRANSFER .05,,KVAR

preusmjerava 5 posto transakcija, od ukupnog broja koji su ušli u ovaj blok, na oznaku KVAR. Ostale transakcije koje nisu doživile stohastičko preusmjeravanje idu na slijedeći blok.

Blokovi 9 do 11: RELEASE PH(XEIK)
 LOGIC R SERVER
 TERMINATE 1

definiraju prvo otpuštanje FACILITY-a s rednim brojem PH(XEIK), zatim resetiranje logičke sklopke SERVER i na kraju terminiranje transakcije iz modela.

Blokovi 12 do 15: KVAR ADVANCE 15,5
 RELEASE PH(XEIK)
 LOGIC R SERVER
 TERMINATE 1

simuliraju situaciju kada je Xeikon u kvaru. U bloku ADVANCE 15,5 se transakcija zadržava 15+-5 minuta što je potrebno vrijeme za rješavanje kvara i izvršavanje intervala do svojeg kraja. Nakon toga se otpušta FACILITY s rednim brojem PH(XEIK) i resetira se logička sklopka SERVER. U bloku TERMINATE 1 transakcija se izbacuje iz modela.

Blokovi 16 do 20: PAUZA MSAVEVALUE BRIN,PH(XEIK),1,0,MH
 ADVANCE 5,1
 RELEASE PH(XEIK)
 LOGIC R SERVER
 TERMINATE 1

reguliraju situaciju kada se treba promijeniti rola papira na Xeikonu. Na oznaci PAUZA resetira se brojač intervala na roli na nulu. On je zapisan u matrici BRIN čiji je redak definiran preko parametra transakcije PH(XEIK), a stupac je eksplicitno jednak 1. U bloku ADVANCE 5,1 transakcija se zadržava 5+-1 minuta što predstavlja vrijeme potrebno za zamjenu jedne role. U bloku

RELEASE PH(XEIK) se otpušta FACILITY rednog broja PH(XEIK), a odmah nakon toga restira se sklopka SERVER. Blok TERMINATE 1 terminira tekuću transakciju iz modela.

Blokovi 21 do 23: PREKID LOGIC S SERVER
 ADVANCE 1
 TRANSFER ,TRAZI

simuliraju situaciju kada transakcija nije našla slobodni FACILITY u bloku 3. Prvo se setira logička sklopka SERVER s čime su se aktivirala logička vrata na početku u bloku 2. U bloku ADVANCE 1 transakcija se zadržava 1 minutu i ponovo se šalje na oznaku TRAZI (TRANSFER ,TRAZI).

Instrukcija 46: START &BINT

postavlja početnu vrijednost brojača terminiranih transakcija. Sa varijablom &BINT postignuta je mogućnost eksperimentiranja s različitim količinama transakcija. Cilj je da model ima dovoljan broj transakcija za pravilno funkcioniranje svih stohastičkih varijabli unutar modela.

Instrukcija 47: PUTPIC FILE=OUT,LINES=4,(&NAKL,C1,(C1/60))
 otvara ispisni kanal u datoteku OUT definiranu instrukcijom 8.
 Stvara se ispisni niz koji ima 4 linije (LINES=4) i 3 ispisnih
 varijabli odnosno aritmetičkih izraza.

Instrukcije 48 do 51:

 Za nakladu od * individualiziranih formi
 bilo je potrebno *.*** minuta ili *.*** sati.

definiiraju ispisnu masku. Maska se puni iz definiranog ispisa iz instrukcije 47.

Instrukcija 52: PUTPIC FILE=OUT,LINES=4,((FR(1)/10),(FR(2)/10))
otvara ispisni kanal u datoteku OUT definiranu instrukcijom 8.
Stvara se ispisni niz koji ima 4 linije (LINES=4) i 2 aritmetička
izraza.

Instrukcije 53 do 56:

Iskoristenje Xeikona1: ***.**%

Iskoristenje Xeikona2: ***.**%

definiraju ispisnu masku. Maska se puni iz definiranog ispisa iz instrukcije 52.

Instrukcija 57: PUTPIC FILE=OUT,LINES=3,(N(PAUZA)+&BXEIK)
otvara ispisni kanal u datoteku OUT definiranu instrukcijom 8.
Stvara se ispisni niz koji ima 4 linije (LINES=4) i 1 aritmetički
izraz. Aritmetički izraz definira koliko se puta prošlo kroz blok
PAUZA, a dodaje se još 2 (2 modula Xeikona) jer je svaki
Xeikon modul počeo sa punim rolama

Instrukcije 58 do 60:

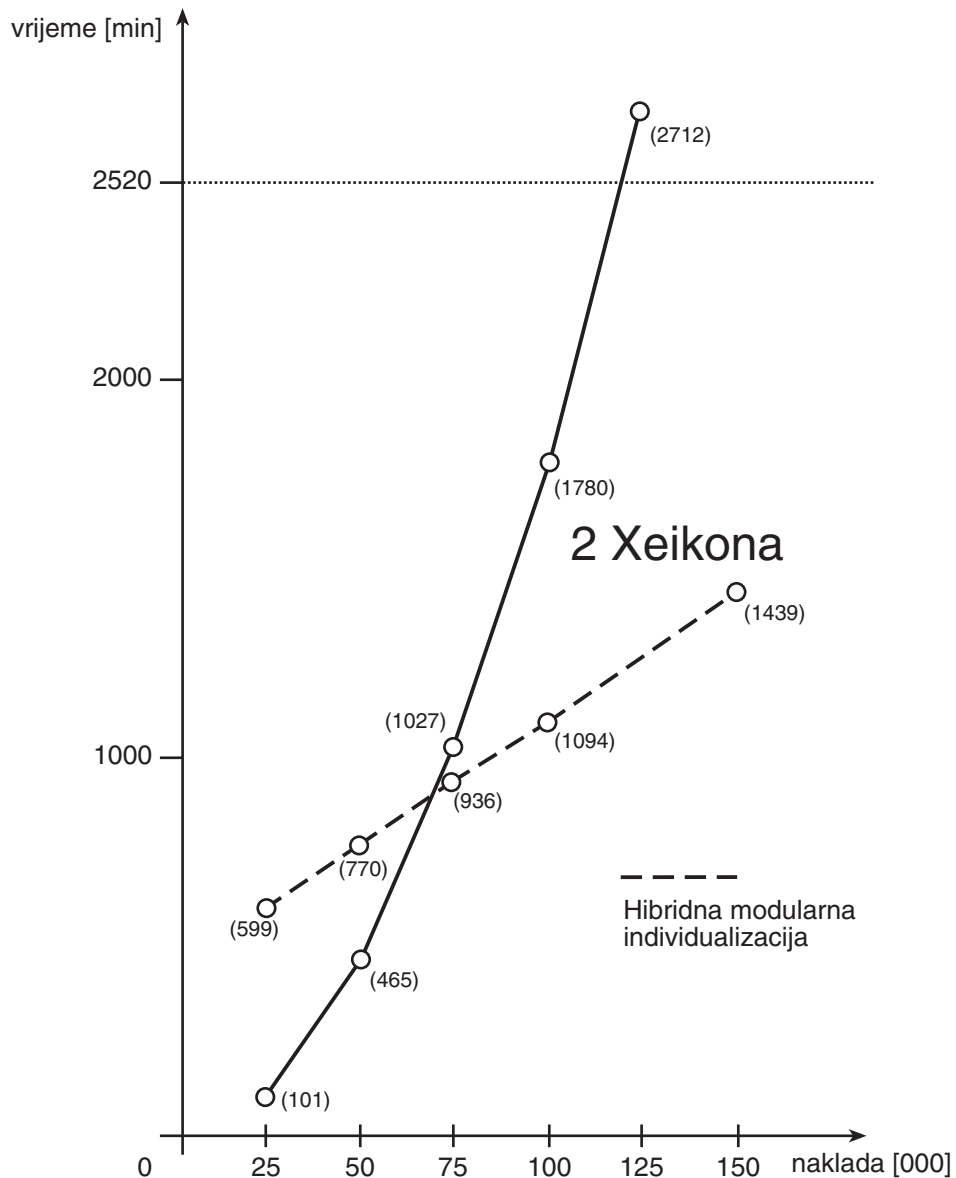
Potrosilo se * rola.

definiraju ispisnu masku. Maska se puni iz definiranog ispisa iz instrukcije 57.

Instrukcija 61: END ; definira kraj GPSS programskog modela.

7.2.3. Eksperimentiranje s modelom

Eksperimentiranja se izvode unutar zadanog vremena od 2520 minuta odnosno 42 sata kao i kod prethodnog modela. Na slici 50 prikazan je grafikon vremena trajanja individualizacije s digitalnim modelom s dva Xeikon modula usporedno s hibridnim modularnim modelom.



Slika 60
Vrijeme trajanja individualizacije s modelom s dva Xeikon modula usporedno s modelom hibridnog sustava

Naklade do 125.000 bi se mogle izvoditi u zadanom vremenu od 42 sata ovim modelom s time da je za naklade do 75.000 ovaj model brži od hibridnog. U odnosu na model s jednim Xeikonom, brzina izvođenja individualizacije se pravilno povećala za dva puta. Cijena tog idealnog udvostručenja je optimalna distribucija PostScript zapisa intervala individualizacije od strane servera kao upravitelja dijeljenjem transakcija prema modulima za individualizaciju, kao i kontrola ispisa koja i ovdje mora biti udvostručena u ljudskom faktoru. Radi se o mogućim oštećenjima koje vidi samo ljudsko oko ili opip (npr. nedovoljno fiksirana boja).

Prednost ovog modela je u povećanoj sigurnosti, jer uvijek postoji mogućnost potpunog kvara jednog modula što bi ipak, uz cijenu probijanja zadanih rokova, završilo projekt.

8. Zaključak

U ovoj disertaciji prikazan je način na koji se mogu izvoditi brza prilagođavanja novim izazovima i zadacima u današnjim i nadolazećim digitalnim ispisnim tehnologijama i njihovim mogućim integriranjem u hibridne sustave.

Sa simulacijom rješavamo računarske probleme u digitalnom tiskarstvu. U polju računarstva dat je doprinos, među ostalima, u stvaranju programskog računarskog sustava za modularnu individualizaciju (6.3). Da bi novostvoreni sustav funkcionirao bilo ga je potrebno optimalno testirati metodom modeliranja i simuliranja. Stvoreni računarski sustav se sastoji od n polja RIP-ova i jednog servera, a na njemu n posebno napisanih programa u PostScript jeziku.

Napravljene su procedure novog načina rada baziranog na modeliranju i simuliranju. Stvoreni su programski GPSS simulacijski modeli s kojima se jednostavno mogu projektirati i optimalno koristiti digitalni tiskarski sustavi s pripadnim sučeljima za obradu slikovnih elemenata i rastera (Raster Image Processor - RIP) i kada ih treba hibridno povezivati s konvencionalnim metodama tiska.

Znanstveni doprinos u ovoj disertaciji je prijedlog novog načina na koji se može modeliranjem i simuliranjem rasterskih rješenja pokrenuti snažan razvoj u postojećim sučeljima za obradu slikovnih elemenata i rastera. Ponudio se originalan pristup izrade matematičkog modela novih rasterskih oblika (4.1.) kao i njihova implementacija u sučeljima (4.2.). Zbog rubnih situacija deformacije rasterskog elementa (RE), koristi se dvodimenzionalan i trodimenzionalan prikaz. Napravljen je program s kojim se ubrzava eksperimentiranje s razvojnim modelom RE (4.1.1.).

Primjer moguće implementacije u sučeljima za obradu slikovnih elemenata i rastera prikazan je kao stohastičko rastriranje. Definirali su se parametri stohastičkog rastriranja i to parametar slučajnog izbora linijature unutar zadanih granica, parametar slučajnog izbora oblika iz palete dozvoljenih oblika i parametar slučajnog izbora kuta rastriranja. Istraživanje s multirasterskim oblicima i stohastičkim izborom linijature i kuta može imati specifičnu primjenu u projektima sa zaštitnim slikovnim elemenata. Ovom metodom osigurava se neponovljivost i jaka zaštita protiv svih tehnika skaniranja.

Predlaže se potpuna kontrola preko numeričke vrijednosti sjemena i algoritma stohastičkog rastriranja. Svi eksperimenti prikazani su na primjeru iste slike radi lakšeg međusobnog uspoređivanja algoritama.

Tiskarska tehnologija prerasta u hibridnu tehnologiju konvencionalnog tiskarstva, digitalnih tiskarskih tehnika i informatičkog inženjeringa. Ovom disertacijom pokazao se način na koji se može takav razvoj bolje osmišljavati, voditi i kontrolirati. Preko modela je racionalnije objašnjavati i dokazivati kako se treba ponašati u novim situacijama. Modeli su idealni za ulazak u ekstremne situacije te testiranja svih procedura koje su u znaku pitanja i nesigurnosti.

Postojeća mjerenja na realnim sustavima poslužila su za izradu stohastičkih modela koji u programskoj izvedbi omogućuju eksperimentiranje s hipotetskim konfiguracijama i resursima bez ikakvog financijskog troška.

Nakon proučavanja digitalnih sustava sa sučeljima za obradu piksla i rastera (3.1., 3.2.) došlo se do potrebe projektiranja novih rasterskih modela s pripadnim programskim rješenjima za posebne zadatke.

Definirani zadatak individualizacije s novim rasterskim procedurama je jedan od najtežih jer se za svaku vrstu posla mora izvoditi nova simulacija i

novo reprogramiranje sučelja za obradu slikovnih elemenata i rastera u PostScript jeziku. Predloženi su GPSS programski modeli na kojima se prije početka svakog posla vrši simulacija različitih konfiguracija ovisno o zadanom roku i broju individualizacije.

Da bi se ocijenila odluka da li ovaj zadatak rješavati hibridnom konfiguracijom ili čistom digitalnom, izgrađen je hibridni model koji će se suprotstaviti čistom digitalnom modelu sustava.

Simulacija hibridnog rješavanja postavljenog zadatka započeta je izradom programskog modela sustava koji omogućuje direktno digitalno fotoosvjetljavane ofsetne ploče iz računala (6.1.) nazvano Computer to Plate (CTP). Nakon proučavanja i eksperimentiranja nad modelom pristupilo se izradi programskog modela analogno digitalnog sustava za izradu mastera (6.2.) u koji su se ugradili rezultati CTP modela. Nakon eksperimentiranja i analize tog modela, izgradio se modularni model individualizacije (6.3.) kao digitalni dio hibridnog sustava za individualizaciju s novim rasterskim procedurama s čime i završava modeliranje takve složene hibridne konfiguracije.

Za isti zadatak izrađen je programski model koji predstavlja čisti digitalni način za individualizaciju s novim rasterskim procedurama. Kao primjer uzeta je elektrofotografija kao tehnologija digitalnog ispisa kroz digitalni tisak najmodernijim Xeikon digitalnim tiskarskim sustavom. Izrađen je programski model s jednim modulom (7.1.) i poseban model s dva modula (7.2.). Uspoređeni su eksperimentalni rezultati hibridnog sustava i digitalnog sustava i objašnjene optimalne odluke za određene naklade individualizacije. Za postavljeni zadatak pokazalo se da se već za naklade individualizacije veće od 50×10^3 mora krenuti u hibridno rješavanje problema, ali istovremeno za manje naklade je čisto digitalno rješenje bolje. Svi izgrađeni programski modeli mogu se lako mijenjati preko parametara modela i tako prilagođavati novim zadacima i novim tehnologijama.

Model CTP sustava građen je na stvorenim funkcijama vjerojatnosti svih faza CTP sustava proizašlim mjerenjima realnih sustava. U disertaciji je prikazano eksperimentiranje s mijenjanjem faktora brzine dolaženja transakcija u model i eksperimentiranjem s povećavanjem broja modula za bitmapiranje. Svi rezultati eksperimentiranja prikazani su grafički. Točno se može vidjeti s kojim brojem ulaznih transakcija moduli ulaze u zagušenja. Ovo eksperimentiranje je pokazalo da je za zadani broj transakcija bilo dovoljno dodati još jedan modul za bitmapiranje, a ne zajedno s njim još jednu jedinicu za osvjetljavanje što je bilo upitno na početku. Također je dat način procjene kada se isplati prelazak sa CTP sustava s jednim bitmap modulom na dva.

Gradnja modela analogno digitalnog sustava za izradu mastera uključivala je analizu CTP modela i svih faza tradicionalne analogne metode tiska. Taj složeni model baziran je na stohastičkim funkcijama i varijablama baziranim na mjerenjima na realnim instalacijama. U programski model (6.2.1.) su se ugrađivali parametri na način da korisnik može koristiti model za druge tradicionalne tehnologije, a ne samo ofsetne. Opisan je način eksperimentiranja s definiranim ciljem (6.2.3.), ali se može koristiti i kao način rješavanja za druge probleme i zadatke.

Drugi dio hibridnog sustava čini izgrađeni model modularne individualizacije (6.3.) bazirane na monokromatskoj elektrofotografiji različitih brzina. Završeni masteri iz prethodne analogno digitalne faze ovdje ulaze u proces individualizacije. Ovaj model je izgrađen na temelju velikog broja podataka mjerenih u različitim uvjetima upravljanja i vođenja na individualizaciji s novim rasterskim procedurama.

Upravo zbog pojava mnogih otvorenih pitanja u praksi i nesavršenih odluka projektantske i programske naravi, pristupilo se izradi programskog modela koji

bi odgovorio zadanim rokovima i nakladama. Na bazi mjerenja u postojećim instalacijama došlo se do vrijednih stohastičkih funkcija kao što su distribucije kvarova i njihovih rješavanja.

Prikladnim programiranjem sučelja za obradu slikovnih elemenata i rastera u PostScript jeziku, što znači programiranja na najnižoj razini, došlo se do prijedloga načina distribuiranja PostScript zapisa između tih sučelja. Na taj način se u ovoj disertaciji demonstrirala prva primjena modeliranja i simuliranja varijabilnog digitalnog tiskarskog sustava.

Istražene su dvije konfiguracije digitalnog tiska (7.3.) za različite naklade za zadani rok. Ovi rezultati se već eksperimentalno primjenjuju, a najveća je dobit u stvaranju sigurnosti da će proizvodnja uspješno završiti u zadanom roku. Stvoreni model koristit će se za izučavanje još veće naklade s drugim parametrima vremena i dodatnim RIP sustavima. To su pripreme za još veće individualizacije sa specijalnim zaštitnim rastriranjem, ali s minimalnim ulaganjem u nove strojeve odnosno s izradom konkurentnih ponuda za nove projekte.

U digitalnom tisku moguće je izvršiti individualizaciju forme istovremeno kada i podloge odnosno mastera. Izgradio se model za isti zadatak kao i za hibridni sustav. Napravljen je poseban model s jednim modulom kao i model s dva digitalna modula. Eksperimentiranja nad oba modela daju nam mogućnost njihovog međusobnog uspoređivanja kao i njihovo suprotstavljanje hibridnom sustavu.

9. LITERATURA

1. T. J. Schriber: "A GPSS primer", University of Michigan, 1972.
2. V. Žiljak, G. Smiljanić: "Modeliranje i simuliranje s računalima, (skripta ETF), 1980., str. 118-121
3. V. Žiljak: "Simulacija računalom", Školska knjiga, Zagreb, 1982.
4. J.P.C. Kleijnen: "Statistical Techniques in Simulation", Marcel Dekker Inc., New York, 1974., ISBN 0-8247-6157-X, str. 1-5
5. B.P.Zeigler: "Theory of Modelling and Simulation", John Wiley & Sons Inc., USA, 1976., ISBN 0-471-98152-4, str. 3
6. H. Maisel, G. Gnugnoli: "Simulation of Discrete Stochastic Systems", Science Research Associates Inc., USA, 1972., ISBN 72-807-61, str. 5
7. G. Gordon: "System Simulation", Prentice-Hall Inc., USA, 1978., ISBN 0-13-881797-9, str. 4
8. K. Pap, V. Žiljak, T. Kosić: "Digitalni obostrani tisak u boji", , Acta Graphica 4 (1994.)
9. "Xeikon DCP-1", Preliminary Service Manual, Xeikon N.V., Mortsel, Belgium, 1994.
10. "Xeikon DCP-1", Operator Manual Version 3, Xeikon N.V., Mortsel, Belgium, 1994.
11. "Xeikon 1995 Kickoff Meeting", Information Paper, Xeikon N.V., Mortsel, Belgium, 1995.

12. Lucien De Schamphelaere, Xeikon Team: "Short Run Digital Color Printing", Keynote Paper, Eleventh Congress on Advances in Non Impact Printing, South Carolina, 1995.
13. Koen Van Hulle: "The Xeikon DCP-1: Increased productivity through an all digital workflow", Materials Seminar, Sixth Color Imaging Conference & Tutorial, Santa Barbara, California, 1995.
14. "Xeikon DCP-1", Product Description, Xeikon N.V., Mortsel, Belgium, 1995.
15. K. Pap, V. Žiljak: "Smanjenje potrošnje boje u digitalnom tisku", , Acta Graphica 4 (1995.)
16. "Firmware V2.01", Xeikon N.V., Mortsel, Belgium, 1995
17. I. Pavić: "Statistička teorija i primjena", Tehnička knjiga, Zagreb, 1988. ISBN 86-7059-076-X, str. 92
18. K. Pap, V. Žiljak: "Stochastic Change of Bezier Curve in PostScript for the Security papers", 5th International Design Conference, Product Design Development & Design Education, Dubrovnik, 1998. UDK 7.05 (063), ISBN 953-6313-20-0
19. V. Žiljak, K. Pap: "Optimization of individualized reproduction in long-run digital printing", IARIGAI 26th research Conference, Munich, 1999.
20. V. Žiljak, K. Pap: "Investigation of document and security digital printing procedures", IARIGAI 26th research Conference "Advances in Printing Science and Technology", Munich, 1999.

21. V. Žiljak, K. Pap: "Mathematical model of the stochastic algorithm in digital printing", IARIGAI 26th research Conference "Advances in Printing Science and Technology", Munich, 1999.
22. V. Žiljak, K. Pap, "PRODUCTION MANAGEMENT FOR THE LONG RUN DIGITAL PRINT WITH INDIVIDUALIZATION BASED ON DINAMIC MODULAR PRINT, 30 th annual Conference of the IC, Stockholm, 1999.
23. K. Pap, V. Žiljak, Z. Sabati: "Simulation of Raster Image Processor in DTP and Digital Printing System", Information and Intelligent Systems - IIS'99, Varaždin, 1999.
24. K. Pap, V. Žiljak: "Design of High Volume Digital Print", 6th International Design Conference-Design 2000, , Dubrovnik, 2000.
UDK 7.05 (063), ISBN 953-6313-38-3
25. V. Žiljak, K. Pap: "PostScript, programiranje grafike", FS d.o.o., Zagreb, 1998, ISBN: 953-199-000
26. Bernard Michael, 2000, Print and production Manual, Pira International, BPIF Publishing, ISBN 185802238X,
27. V. Žiljak, K. Pap: "Simulation Models in Printing Education", 6th International Design Conference-Design 2000, , Dubrovnik, 2000.
UDK 7.05 (063), ISBN 953-6313-38-3
28. V. Žiljak, K. Pap: "Elaboration of simulators as a basis of developing the digital printing appliance in package production", International Symposium printing on Packaging, Zagreb, 2001.

29. V. Žiljak, K. Pap, I. Žiljak: "Color Reproduction by Individualized Half-tone Shapes", 28. International Research Conference of IARIGAI, Montreal, Canada, 2001.
30. Z. Nježić, V. Žiljak, K. Pap: "Design of Digital Graphic System", International Design Conference DESIGN 2002, pp 876, Dubrovnik 2002.
31. V. Žiljak, V. Simovic, K. Pap: "Simulation of Stochastic System of Printing Procedures", The International Conference on Modeling and Simulating of Complex System, ICMSCS 2002, Chengdu, Sichuan, China
32. V. Žiljak, K. Pap, D. Agić, I. Žiljak: "Modelling and Simulation of Integration of Web system, Digital and Conventional Printing", 29th International Research Conference of IARIGAI, Lake of Lucerne, Switzerland, 2002 .
33. K. Pap, V. Žiljak: "Model simulacije dinamičkog konfiguriranja grafičkih sustava", IV simpozij Modeliranje u znanosti, tehnici i društvu., Rijeka 2000. UDK 519.8(082), ISBN 953-6065-00-2
34. K. Pap, V. Žiljak: "A suggestion of new definition of halftone cells in the print with the interpretation by means of Mathematica", PrimMath (2001), 1. Znanstveno-stručni skup Programski sustav Mathematica u znanosti, tehnologiji i obrazovanju, Zagreb, 2001.
35. V. Žiljak, K. Pap: "PostScript", (engleski)
Print & Publishing International Verlagsges m. b. H., Wien, 1999,
ISBN: 3-9501090-0-5
36. V. Žiljak, K. Pap, Matematički model transformacije piksla, Grafički fakultet, Zagreb, 1999.
37. K. Pap, V. Žiljak, T. Kosić: "Optimizam digitalne generacije", Vidi, br.3, Zagreb, 1995.

38. K. Pap, V. Žiljak, T. Kosić: "Digitalni tisak, sve je u dobroj pripremi", Vidi, br.4, Zagreb, 1995.
39. K. Pap, V. Žiljak, T. Kosić: "Modularne mogućnosti Xeikona", Vidi, br.5, Zagreb, 1995.
40. I. Adobe Systems: „PostScript Language Reference Manual”, Addison-Wesley, 1985, str.1
41. I. Adobe Systems: „PostScript Language Tutorial and Cookbook”, Addison-Wesley, 1985
42. I. Adobe Systems, G. C. Reid: „PostScript Language Program Design”, Addison-Wesley, 1988
43. Roth S. F., 1988, Real World PostScript, Addison-Wesley, ISBN 0-201-06663-7
44. D. Holzgang: „PostScript Programmer’s Reference Guide”, Scott, Foresman and Company, 1989
45. R. Smith: „Learning PostScript: A Visual Approach”, Peachpit Press, 1990
46. I. Adobe Systems: „PostScript Language Reference Manual”, Addison-Wesley, 1990
47. C. R. Glenn: „Thinking in PostScript”, Addison-Wesley, 1990
48. I. Adobe Systems: „Adobe Type 1 Font Format”, Addison-Wesley, 1990
49. H.McGilton, M.Campione: „PostScript by Example”, Addison-Wesley, 92.
50. P.Fink: „PostScript Screening:Adobe Accurate Screens”, AdobePress, 92

51. T. Merz: „PostScript & Acrobat, PDF”, Springer, 1996
52. J. Banks, J. S. Carson, J.N. Sy: "Getting Started with GPSS/H", Wolverine Software Corporation, 1989
53. P. Green: "Understanding Digital Color", Second Edition, GattPress USA, Pira International UK, 1999, ISBN 1-85802-450-1
54. Adobe Inc.: "Adobe PostScript 3", 1998
<http://www.adobe.com/products/postscript/pdfs/PostScript3Whitepaper.pdf>
Uloga PostScripta jezika i osnovne značajke PostScript Level 3 verzije
55. J. H. Bear <http://desktoppub.about.com/cs/rip/>
Jedna od definicija RIP-a (Raster Image Processor)
56. G. S. Owen: "Computer Graphics Hardware Overview", 1999. <http://www.siggraph.org/education/materials/HyperGraph/hardware/hardware.htm>
57. Océ-Technologies: "PDF: supporting document interchange in the collaborative engineering environment", 2002.
http://www.adobe.com/products/postscript/pdfs/wp_oce.pdf
58. InternetBiz.net: "PDF Toolbox", 2003
<http://www.pdfzone.com/toolbox/>
Sve informacije o PDF alatima
59. Xeikon Int.: "Xeikon DCP 320D"
<http://www.xeikon.com/ComWebsite/productsfiles/DCP320D%20EN.pdf>
Opis i tehničke karakteristike uređaja Xeikon DCP 320D
60. Hewlett-Packard Com.: "HP indigo press 1000 Overview", 2003
http://h30011.www3.hp.com/Products/1000/prodinfo_overview.html

61. West Products Com.: "Docutech Technical Information", 2003.
<http://www.westproducts.com/site/database/dt6180t.html>
62. Elcorsy Technology Inc.: "ELCO 400 PRESS", 2003
<http://www.elcorsy.com/an/products.html>
Opis i tehničke karakteristike uređaja ELCO 400 PRESS
63. Narrele Jarry, Computer Imaging Technology: The Process of Identification, The Book and Paper Group, Annual, Vol. 15. 1996.
<http://aic.stanford.edu/conspec/bpg/annual/v15/bp15-08.html>
64. D. Dunn: "Research & Engineering Council of the Graphic Arts", R&E, 2001
<http://www.recouncil.org/presentations/pressroom2001/28-dunn.pdf>
65. P. Castegnier: "Elcography, a New digital printing alternative", Elcorsy Technology Inc., <http://www.elcorsy.com/an/press/gatft3.html>
66. P. Castegnier: "New High Speed Digital Press Platform using Electrocoagulation Printing", TAGA 1998.
<http://www.elcorsy.com/an/press/TAGA98.PDF>
67. Nepp: "Elcorsy Technology", Digital Printing Report, Vol. 5, br. 10, 1998
<http://www.rit.edu/~spmswww/frank/dprint/9810.pdf>
68. R. S. Engelschall: "The gallery of Open Source Unix software from the net"
<http://www.netsw.org/graphic/postscript/interpreter>
Indeks dostupnih PostScript interpretera
69. A. L. Oakley, A. C. Norris: "Page description languages: development, implementation and standardization", ELECTRONIC PUBLISHING, VOL. 1(2), 79-96 (SEPTEMBER 1988), str. 82-83,
<http://cajun.cs.nott.ac.uk/compsci/epo/papers/volume1/issue2/epalo012.pdf>

70. PrePressSolutions,Inc:"Panther RIP, Panther II RIP User's Guide",Ch.6
Operativni manual RIP-a sa page buffer tehnologijom
71. Adobe Systems Inc.:"Adobe PostScript Extreme",1998,
Tehnologija za optimizaciju protoka podataka, koristi OPI
www.adobe.com/products/extreme/pdfs/extremewp.pdf
72. Adobe Systems Inc.:"PDF for Prepress Workflow and Document
Delivery",1997
<http://www.adobe.com/products/postscript/pdfs/PDFps3.pdf>
73. The Sheridan Press: "Digital Workflow: Managing the Process
Electronically", www.sheridanpress.com/PDF_docs/digwrkflw.pdf
74. Adobe Systems Inc.:"How to trap using Adobe trapping technologies",
2002, <http://www.adobe.com/products/extreme/pdfs/trapping.pdf>
75. Adobe Systems Inc.:"Adobe Trapping Technology", 2000,
<http://www.adobe.com/products/extreme/pdfs/trappingwp1.pdf>
76. M.Rundle:"Stochastic Screening", Kodak Polychrome Graphics,2002
www.kpgraphics.com/pdf/info/whitepapers/stochastic.pdf
77. S. Gooran, B. Kruse:"Colour Halftoning in DigitalPrinting"
http://www.pira.co.uk/admin/_private/TechnicalArticles/00024.pdf
78. B. Lawler: "Know thy enemy: understanding dot gain and its effects",97.
<http://www.imaging-resource.com/TIPS/LAWLER/DOTGAIN.PDF>
79. P.Fink:"Halftones & Dot Gain",1995.
<http://www.imagequality.com/dtp/htmspg18.htm>
80. PrePress Solution Inc.:"Densitometry and Dot Gain", 2001
<http://www.prepress.pps.com/TechReports/densitytr.html>

81. G. L. Rogers:"Optical Dot Gain in Halftone Print",Journal of Imaging Science and Technology, 41, 643-656, (1997)
82. Wolfram OnLine Documentation "Mathematica"V5, <http://documents.wolfram.com/v5/TheMathematicaBook/MathematicaReferenceGuide/index.html>
83. L. J. D'Andria:"Introduction to the Mathematica FrontEnd: Version 3.0",1998.
84. J. M. Novak:"Programming the V3.0 FrontEnd",1998
85. Wolfram OnLine Documentation:"Three-Dimensional Surface Plots"
<http://documents.wolfram.com/v4/MainBook/1.9.7.html>
86. Wolfram OnLine Documentation:"Contour and Density Plots"
<http://documents.wolfram.com/v4/MainBook/1.9.6.html>
87. Adobe Systems Inc.:"Level 2 Compatibility:The setscreen and current screen Operators", Technical Note #5119,1992.
<http://partners.adobe.com/asn/developer/pdfs/tn/5119.setscreen.pdf>
88. T.King:"Technology benchmarking Comparative testing of thermal and silver CTP technology",Agfa White Paper
89. L.Brunner: "When Big is Beautiful",PrintingWorld,October 11, 1999
<http://www.printingworld.com/ctp/ctp99a.htm>
90. Turner Fenton Secondary School:"Basic Typography",The Basics:Lesson 9,
<http://www.turnerfenton.com/lessons/programming/dic2ax/basics/dtp.htm>
91. Yamada Language Center:"Croatian Fonts", University of Oregon,
<http://babel.uoregon.edu/yamada/fonts/croatian.html>

92. L. Devroye:"The Croatian font scene", School of Computer Science McGill University ,Montreal, <http://cgm.cs.mcgill.ca/~luc/croatia.html>
93. R.Gospodnetic, Z. Kurjan, D. Šimunović:"Norme za kodne sustave u Hrvatskoj - Prijedlog", Hrvatska Informatička Zajednica, 1997.
http://www.open.hr/hiz/kodsus/hiz_norme_02_3.01.pdf
94. Linguist Software:"Fonts for the Whole World:LaserSerbo-Croatian"
<http://www.linguistsoftware.com/l Srb-cro.htm>
95. On-line manual:"Word 2000 Tutorial",Florida Gulf Coast University
<http://www.fgcu.edu/support/office2000/word/>
96. Adobe Inc.:"Adobe PageMaker 7.0 Overview",2001
<http://www.adobe.com/products/pagemaker/pdfs/pm7ov.pdf>
97. Quark Inc.:"QuarkXPress Overview",2003
<http://www.quark.com/products/xpress/overview.html>
98. R. Drouillard, D. Thistle:"Computer Concepts for End-Users: WordPerfect",School of Computer Science,2002.
<http://electron.cs.uwindsor.ca/60-104/wordperfect.html>
99. Corel Corp.:"Corel Ventura 10 for Windows:Reviewers Guide",2002.
http://www.corel.com/futuretense_cs/ccurl/Reviewers_Guide,4.pdf
100. Adobe Inc.:"Adobe Printer Drivers",2003
<http://www.adobe.com/products/printerdrivers/main.html>
101. Ovid Bell Press:"PostScript Assistance:How to use Postscript Drivers", 2003, <http://www.ovidbell.com/pdf/postscript.html>

102. V. Ostromoukhov, R.D. Hersch: "Artistic Screening", Proceedings of SIGGRAPH 95, Annual Conference Series, str. 219-228, 1995.
103. V. Ostromoukhov, R.D. Hersch: "Multi-Color and Artistic Dithering", Proceedings of SIGGRAPH 99, Annual Conference Series, str. 425-432, 1999.
104. V. Ostromoukhov: "Artistic Halftoning-Between Technology and Art", SPIE Vol. 3963, str. 489-509, 2000.
105. V. Ostromoukhov, N. Rudaz, I. Amidror, P. Emmel, R.D. Hersch: "Anti-Counterfeiting Features of Artistic Screening", Int. Symposium on Advanced imaging and Network Technologies, 1996. Berlin, SPIE Vol. 2951, str. 126-133
106. L. Streit, J. Buchanan: "Importance Driven Halftoning", Proceedings of EuroGraphics 98, str. 207-217
107. O. Verevka, J. Buchanan: "Halftoning with Image-Based Dither Screens", GI99, Online Paper: www.graphicalinterface.proceedings/1999/106
108. Y. Pnueli, A.M. Bruckstein: "Gridless Halftoning: A Reincarnation of the Old Method", Graphical Models and Image Processing Vol. 58, 1996, str. 38-64
109. S. Spannenburg: "Frequency Modulation of Printed Gratings as a Protection against Copying", Conf on Holographic Optical Security Systems, SPIE Vol 1509 (W.F. Fagan, Ed.), 1991, str. 88-103
110. H. Berghel, L. O'Gorman: "Digital Watermarking", 1997
http://www.acm.org/~h1b/publications/dig_wtr/dig_watr.html
111. J. Foley, A. van Dam, S. Feiner, J. Hughes: "Computer graphics, principles and practice", 2nd ed., Addison-Wesley, 1993., ISBN:0-321-21056-5

SAŽETAK

U ovoj disertaciji predlaže se postupak modeliranja i simuliranja hibridnih i digitalnih sustava sa sučeljima za obradu slikovnih elemenata i rastera od razine konceptijskih modela do njihove realizacije kao programa za računalnu simulaciju. Pokazuje se da je donošenje odluka moguće jedino iscrpnim planiranjem na bazi rezultata dobivenih simulacijom.

Opisani su današnji digitalni sustavi sa sučeljima za obradu slikovnih elemenata i rastera. Definiran je odnos slikovnog elementa i pridruženog binarnog zapisa ovisno o ispisnoj tehnologiji. Istražene su osnovne funkcije i faze rada sučelja za obradu slikovnih elemenata i rastera kao što je faza procesiranja i faza rastriranja. Navedeni su razlozi uvođenja programskog jezika PostScript kao vodećeg jezika za opis grafičke stranice u sučeljima za obradu slikovnih elemenata i rastera.

Izvodi se matematičko 3D i 2D modeliranje originalnih novih rastera kao i pripadni PostScript programski kod. Izvedena je originalna implementacija u sučeljima za obradu slikovnih elemenata i rastera kao stohastičko rastriranje. Za eksperimentiranje su predloženi različiti algoritmi rastriranja. Time se upućuje na nove moguće pravce razvoja rastriranja u sučeljima za obradu slikovnih elemenata i rastera. Opisane su sve PostScript rutine za testiranje novih rasterskih modela i vođenja eksperimenata.

Predložen je razvoj digitalnih sustava sa sučeljima za obradu slikovnih elemenata i rastera prema modularnim hibridnim sustavima. Iznose se problemi s kojima se autor suočavao proteklih godina projektiranja takvih sustava u praksi. Prikazane su konfiguracije hibridnih linija u vidu šest samostalnih modela koji pokrivaju cjelovito digitalno ustrojstvo tiskarskih podkonfiguracija.

Razvijene su dvije moguće koncepcije rješavanja istog sustava za individualizaciju s novim rasterskim procedurama. Zadatak individualizacije s novim rasterskim procedurama poslužio je za sučeljavanje i ocjenjivanje hibridnog sustava nasuprot čistog digitalnog sustava.

Stvoren je složen hibridni model za taj zadatak koji omogućuje optimalno projektiranje i reprogramiranje dijelova hibridnog sustava u kratkom vremenu. Dat je originalni programski model sustava koji omogućuje direktno digitalno fotoosvjetljavanje ofsetne ploče iz računala nazvano Computer to Plate. Mikrosimulacijama pojedinih modula CTP sustava vrši se dublje istraživanje da bi se optimirao ciljani modul sustava. Priloženo je programsko rješenje u jeziku GPSS kao i eksperimentalni rezultati. Rezultati simulacije CTP sustava se ugrađuju u novi programski model analogno digitalnog sustava za izradu mastera - jednoznačne podloge koja se kasnije individualizira. Model modularne individualizacije se naslanja na prethodni model čineći s njim hibridnu konfiguraciju. Gradnja tog modela za individualizaciju mastera kao i rješenje modela u GPSS jeziku bazirano je na mjerenjima fizičkog sustava. Eksperimentiranje s modelom se prikazuje uspoređivanjem dviju hipotetičnih konfiguracija.

Za drugu koncepciju izgradio se model digitalnog sustava za individualizaciju s novim rasterskim procedurama i to model s jednim digitalnim modulom i model s dva modula. Za oba slučaja dani su programski GPSS modeli na kojima je izvršeno eksperimentiranje. Na kraju svakog eksperimentiranja grafički su uspoređeni rezultati s rezultatima hibridnog modularnog modela.

Ključne riječi:

Hibridni sustav, digitalni sustav, sučelje za obradu slikovnih elemenata i rastera, RIP, PostScript rastriranje, modeliranje i simuliranje, stohastički model, GPSS, digitalni tisak, Mathematica

Simulation of Hybrid and Digital Systems with Raster Image Processors

SUMMARY

The thesis proposes a procedure of modelling and simulation of hybrid and digital systems with raster image processors from concept model level to their realization as program for computer simulation. It is apparent that decision making is possible only by thorough planning based on the results obtained through simulation.

Described are modern digital systems with raster image processors. The relation between pixels and the associated binary codes has been defined in regard to printing technology. The thesis explores the basic functions and working stages of the raster image processor, such as processing and rasterization. It also states the reasons for introducing PostScript as the main language for graphical page description in raster image processors.

It contains original procedure for developing mathematical 3D and 2D modelling of original new raster elements as well as the associated PostScript program code. The author has performed an original implementation in raster image processors as stochastic rasterisation. Various algorithms of rasterisation are suggested for experimentation, thus pointing out new possible directions for the development of rasterisation in raster image processors. All PostScript routines for testing new raster models and supervising experiments have been described.

The thesis also contains suggestions for the development of digital systems with raster image processors according to modular hybrid systems. The author discusses problems he faced during the actual design of such systems. He also presents configurations of hybrid lines in form of six individual models covering the entire digital system of printing sub-configurations.

There are developed two possible concepts of the same system for individualization with new raster procedures. The problem of individualization with new raster procedures has aided the interfacing and evaluation of hybrid systems as opposed to purely digital systems.

For the task of individualisation a complex hybrid model has been created that facilitates quick optimal designing and reprogramming of parts of the hybrid system. The author creates an original program model for a system called Computer-To-Plate (CTP) that enables direct digital phototypesetting of the offset plate from the computer. Microsimulations of individual modules of the CTP system present a more detailed research in order to optimize target module of the system. Attached are a program solution written in GPSS, as well as experimental results. The simulation results for the CTP system are built into a new program model of a analogous digital system for the production of a master which is to be individualized later on. The model of modular individualization leans upon the previous model, thereby creating a hybrid configuration. The design of this model, as well as the GPSS model solution, are based on measurements of the real system. Experiments with the model are shown by comparing two hypothetical configurations.

For the other concept, the author has designed two models of the digital individualization system with new raster procedures, one with one digital module and one with two modules. In both cases the experimental GPSS program models are given. At the end of each experiment, the results were compared graphically with the results of the hybrid modular model.

Key words:

Hybrid System, Digital System, Raster Image Processor, RIP, PostScript rasterisation, Modelling and Simulation, Stochastic Model, GPSS, Digital Printing, Mathematica

ŽIVOTOPIS

Rođen sam 20. ožujka 1963. godine u Zagrebu. Nakon završene matematičke gimnazije studirao sam na Elektrotehničkom fakultetu Sveučilišta u Zagrebu gdje sam i diplomirao u smjeru Računarske tehnike 1988. godine. Magistrirao sam 1997. godine na istom fakultetu (sada Fakultet elektrotehnike i računarstva) s temom "Simulacija protoka informacija između računala i digitalnog tiskarskog stroja" u smjeru Računarskih znanosti.

Od 1988. do danas zaposlen sam na Grafičkom fakultetu Sveučilišta u Zagrebu te sam biran putem FER-a u asistenta na katedri Elektronička računala. Na Grafičkom fakultetu sam također biran za asistenta za predmete Računarski slog i Računarska grafika od 30. rujna 1999. Kao asistent radim i na Studiju informatičkog dizajna Sveučilišta u Zagrebu i Studiju dizajna pri Arhitektonskom fakultetu Sveučilišta u Zagrebu. Izabran sam na FER-u 24.4.2003. u naslovno nastavno zvanje viši predavač za područje tehničkih znanosti, polje računarstvo, za grupu predmeta Digitalni tisak i Grafički programski jezici.

U toku svog rada bavio sam se istraživanjem, razvojem i primjenom računala. Područja rada su: računarska grafika, procesiranje slike i teksta, modeliranje i simuliranje s računalom, web tehnologije, digitalni tisak, grafički programski jezici. Sudjelovao sam na uvođenju novih tehnologija o čemu sam održavao niz seminara, predavanja i tečajeva.

U ovoj biografiji su navedena 57 naslova. Objavio sam sedam (7) znanstvenih članaka u domaćim časopisima, četrnaest (14) znanstvenih članaka u zbornicima međunarodnih skupova, četiri (4) znanstvena članka u zbornicima s domaćih skupova, dva (2) stručna članka, koautor sam dviju (2) knjiga, tri (3) znanstveno popularizacijska članka, četiri (4) pregledna članka, aktivno sam sudjelovao i sudjelujem na šest (6) istraživačkih projekata.

MR. SC. KLAUDIO PAP

1. PUBLIKACIJE

1.1. KNJIGE

2

knjige na hrvatskom jeziku

1. V. Žiljak, K. Pap: "PostScript, programiranje grafike", FS d.o.o., Zagreb, 1998, ISBN: 953-199-000

knjige u inozemstvu

2. V. Žiljak, K. Pap: "PostScript", (engleski)
Print & Publishing International Verlagsges m. b. H., Wien, 1999,
ISBN: 3-9501090-0-5

1.2. POGLAVLJA U KNJIZI

1

1. V. Žiljak, K. Pap: "Grafička priprema računalom", Sto godina Grafičke škole u Zagrebu, Zagreb, 1994 (str161-168) ISBN 953-96265-01

1.3. ZNANSTVENI ČLANAK U DOMAĆEM ČASOPISU

7

1. Đ. Črnjak, K. Pap: "Računarski modeli izrade osobnih dokumenata", Acta Graphica 4 (1992.), UDK 655.3.026.9:351.755:681.3
2. V. Žiljak, Đ. Črnjak, K. Pap: "Novi postupci oblikovanja dokumenata i vrijednosnih papira", Acta Graphica 4 (1992.), UDK (336.74+351.755):(655.3+667.52+676.224)
3. K. Pap, V. Žiljak, T. Kosić: "Digitalni obostrani tisak u boji", Acta Graphica 4 (1994.), Zagreb, UDK: 681.3:655.39
4. V. Žiljak, K. Pap, M. Josipović, D. Ferdebar: "Matematički model transformacije piksla i izvornog dizajna", Acta Graphica 3 (1995.), UDK/UDC 655.26:681.3
5. K. Pap, V. Žiljak: "Smanjenje potrošnje boje u digitalnom tisku", Acta Graphica 4 (1995.), UDK/UDC 681.3:655,39.022.3.003
6. K. Pap, V. Žiljak: "Deformacija piksla preko inverzne transformacije", Acta Graphica 11 (1999.), UDK/UDC 667.5.036:655.344
7. K. Pap, V. Žiljak: "Minimalizacija sličnosti CMYK modela i Spot2 separacije", Acta Graphica 4-99 11 (1999.)4, 161-206 UDK/UDC 655(05) Coden: Aggrer ISSN 0353-4707 pp: 161-171

1.4 ZNANSTVENI ČLANAK U ZBORNIKU

znanstveni članak u zborniku međunarodnog skupa 17

1. V. Žiljak, K. Pap, T. Kosić: "Neue PostScript Linien Raster und Möglichkeiten im Sicherheitsdruck", Symposium Aufbruch in neue Dimensionen der Farbproduktion, Wien, Austrija, 1993.
2. K. Pap, V. Žiljak, J. Žiljak: "Forming a Personal Handwriting for UNICODE", 5th International Design Conference, Product Design/ Development & Design Education, Dubrovnik, 1998.
UDK 7.05 (063), ISBN 953-6313-20-0
3. K. Pap, V. Žiljak: "Stochastic Change of Bezier Curve in PostScript for the Security papers", 5th International Design Conference, Product Design Development & Design Education, Dubrovnik, 1998.
UDK 7.05 (063), ISBN 953-6313-20-0
4. K. Pap, Ž. Kabljanac: "A system for the Digital Recording of Static Objects", 5th International Design Conference, Product Design Development & Design Education, Dubrovnik, 1998.
UDK 7.05 (063), ISBN 953-6313-20-0
5. K. Pap, V. Žiljak, Đ. Črnjak: "A New Linear Grid in the Design of Securities", 5th International Design Conference, Product Design Development & Design Education, Dubrovnik, 1998.
UDK 7.05 (063), ISBN 953-6313-20-0
6. V. Žiljak, K. Pap: "Optimization of individualized reproduction in long-run digital printing", IARIGAI 26th research Conference, Munich, 1999.
7. V. Žiljak, K. Pap: "Investigation of document and security digital printing procedures", IARIGAI 26th research Conference, Munich, 1999.
8. V. Žiljak, K. Pap: "Mathematical model of the stochastic algorithm in digital printing", IARIGAI 26th research Conference, Munich, 1999.
9. V. Žiljak, K. Pap, "PRODUCTION MANAGEMENT FOR THE LONG RUN DIGITAL PRINT WITH INDIVIDUALIZATION BASED ON DINAMIC MODULAR PRINT, 30 th annual Conference of the IC, Stockholm, 1999.
10. K. Pap, V. Žiljak, Z. Sabati: "Simulation of Raster Image Processor in DTP and Digital Printing System", Information and Intelligent Systems - IIS'99, Varaždin, 1999.
11. K. Pap, V. Žiljak: "Design of High Volume Digital Print", 6th International Design Conference-Design 2000, , Dubrovnik, 2000.
UDK 7.05 (063), ISBN 953-6313-38-3
12. V. Žiljak, K. Pap: "Simulation Models in Printing Education", 6th International Design Conference-Design 2000, , Dubrovnik, 2000.
UDK 7.05 (063), ISBN 953-6313-38-3

13. V. Žiljak, K. Pap: "Elaboration of simulators as a basis of developing the digital printing appliance in package production", International Symposium printing on Packaging, Zagreb, 2001.
14. V. Žiljak, K. Pap, I. Žiljak: "Color Reproduction by Individualized Half-tone Shapes", 28. International Research Conference of IARIGAI, Montreal, Canada, 2001.
15. Z. Nježić, V. Žiljak, K. Pap: "Design of Digital Graphic System", International Design Conference DESIGN 2002, pp 876, Dubrovnik 2002.
16. V. Žiljak, V. Simovic, K. Pap: "Simulation of Stochastic System of Printing Procedures", The International Conference on Modeling and Simulating of Complex System, ICMSCS 2002, Chengdu, Sichuan, China
17. V. Žiljak, K. Pap, D. Agić, I. Žiljak: "Modelling and Simulation of Integration of Web system, Digital and Conventional Printing", 29th International Research Conference of IARIGAI, Lake of Lucerne, Switzerland, 2002

znanstveni članak u zborniku s domaćeg skupa

4

1. V. Žiljak, K. Pap, D. Žiljak: "DESKTOP PUBLISHING - DESKTOP REPRO", 11. Znanstveni stručni simpozij Intergrafika 91., UDK 655:681.3
2. V. Žiljak, K. Pap, T. Kosić: "Transformacija računarske pripreme novinskog sloga", Znanstveno stručni simpozij Intergrafika 93., UDK 655.28.026.12:681.3
3. K. Pap, V. Žiljak: "Model simulacije dinamičkog konfiguriranja grafičkih sustava", IV simpozij Modeliranje u znanosti, tehnici i društvu., Rijeka 2000. UDK 519.8(082), ISBN 953-6065-00-2
4. K. Pap, V. Žiljak: "A suggestion of new definition of halftone cells in the print with the interpretation by means of Mathematica", PrimMath (2001), 1. Znanstveno-stručni skup Programski sustav Mathematica u znanosti, tehnologiji i obrazovanju, Zagreb, 2001.

1.5. STRUČNI ČLANAK

u domaćem časopisu

2

1. K. Pap, Novi programi i novi kolegiji u visokom obrazovanju grafičara, III simpozij hrvatskih grafičara, Senj, 1999.
2. K. Pap, J. Kancir, Interaktivna učionica računalne grafike, IV simpozij hrvatskih grafičara, Matica hrvatska, Senj, 2000.

1.6 ZNANSTVENO POPULARIZACIJSKI ČLANAK 3

1. K. Pap, V. Žiljak, T. Kosić: "Optimizam digitalne generacije", Vidi, br.3, Zagreb, 1995.
2. K. Pap, V. Žiljak, T. Kosić: "Digitalni tisak, sve je u dobroj pripremi", Vidi, br.4, Zagreb, 1995.
3. K. Pap, V. Žiljak, T. Kosić: "Modularne mogućnosti Xeikona", Vidi, br.5, Zagreb, 1995.

1.7 PREGLEDNI ČLANAK 4

1. V. Žiljak, K. Pap, D. Žiljak POSTSCRIPT FONTOVI, Fotosoft, Zagreb, 1991.
2. V. Žiljak, K. Pap, PROGRAMIRANJE GRAFIKE U POSTSCRIPTU, FS, Zagreb, 1998.
3. V. Žiljak, K. Pap, OBLIKOVANJE FONTOVA RAČUNALOM, Studij dizajna pri Arhitektonskom fakultetu, Zagreb, 1998.
4. V. Žiljak, K. Pap, Matematički model transformacije piksla, Grafički fakultet, Zagreb, 1999.

2. AKTIVNO SUDJELOVANJE U REALIZACIJI ZNANSTVENIH PROJEKATA

Upisan pri Ministarstvu znanosti br. 162136 6

1. Istraživač na znanstveno istraživačkom projektu
Šifra projekta: 2-16-322 Glavni istraživač: Husein Pašagić
Projekt: Kvantitativne i informatičke metode u grafičkoj tehnologiji
Naziv ustanove: Grafički fakultet, Zagreb (128)
Trajanje projekta: 3g, 1991-1994
2. Voditelj projekta
Projekt: Digitalni udžbenik PostScript grafike
Šifra projekta: 00-41
Naziv ustanove: Arhitektonski fakultet
Ime odjela: Studij dizajna
Trajanje projekta: 1g, 2000-2001
3. Istraživač na znanstveno istraživačkom projektu
Projekt: Digitalni sustavi u tiskarstvu
Šifra projekta: 128003 Glavni istraživač: Vilko Žiljak
Naziv ustanove: Grafički fakultet, Zagreb
Trajanje projekta: 1g, 2001-2002
4. Istraživač na znanstveno istraživačkom projektu
Projekt: Digitalni sustavi u tiskarstvu
Šifra projekta: 128009 Glavni istraživač: Vilko Žiljak
Naziv ustanove: Grafički fakultet, Zagreb
Trajanje projekta: 3g, 2002-

5. Istraživač na tehnologijskom istraživačkom razvojnom projektu
Projekt: Softverski alati za izradu holografskih prototipova
Šifra projekta: TP-02/0128-09 Glavni istraživač: Vilko Žiljak
Naziv ustanove: Grafički fakultet, Zagreb
Trajanje projekta: 3g, 2003-
6. Istraživač na tehnologijskom istraživačkom razvojnom projektu
Projekt:: Metoda projektiranja hibridnih konfiguracija tiskarskih tehnologija
Glavni istraživač: Vilko Žiljak
Naziv ustanove: Grafički fakultet, Zagreb
Trajanje projekta: 3g, 2003-

3. NASTAVA

9

DODIPLOMSKA NASTAVA (predavanja, vježbe, seminari)

Grafički fakultet:

1. Elektronička računala
2. Računarski slog
3. Računarska grafika

Arhitektonski fakultet, studij dizajna:

4. Računarska tipografija, (1991. -)
5. Računarska reprofotografija, (1991. -)

Studij poslovne informatike, Sveučilište u Zagrebu:

6. Računarska tipografija. (1994. - 1999)

Veleučilište u Zagrebu, Informatički odjel

7. Pretraživanje i navigacija na WEB-u (2001. -)
8. Grafički programski jezici
9. Digitalni tisak
10. Interaktivno programiranje Weba

Suautorstvo udžbenika

1. V. Žiljak, K. Pap: "PostScript, programiranje grafike", FS d.o.o., Zagreb, 1998, ISBN: 953-199-000

Mentorstvo diplomskog rada

- Četiri diplomanta na Tehničkom veleučilištu

4. ISTRAŽIVAČKA I RAZVOJNA DOSTIGNUĆA

4.1. NOVI PROIZVOD

3

1. V. Žiljak, K. Pap i grupa suradnika, Razvoj i izvedba originala dokumenata Republike Hrvatske (putovnica, domovnica, osobna iskaznica i ostalo, Hrvatski tiskarski zavod, Zagreb, 1991.
2. V. Žiljak, K. Pap i grupa suradnika, Razvoj metodologije i izvedba originala osnovnog državnog topografskog zemljovida 1:25000, 600 listova, 1:100000, 46 listova, Državna geodetska uprava i FS, Zagreb, 1995-1998
3. V. Žiljak, K. Pap i grupa suradnika, Državni topografski zemljovid 1:200000, 26 lista, Državna geodetska uprava i FS, Zagreb, 2000.-2001

4.2. SOFTVER

2

1. V. Žiljak, K. Pap: Programski paket tiska zaštitnog sustava (dionice, vrijednosni papiri, zaštićeni dokumenti, HTZ i FS, 1991, drugo izdanje 1998.
2. V. Žiljak, K. Pap: Programski paket novih rastera, FS, 2000.

5. OSTALO - OSNIVANJE LJETNIH ŠKOLA, TEČAJEVA

5

1. Seminar: "DTP/DTR, obrada slike računalom", Stubičke toplice, 1990. g.
2. Seminar: "Razvoj grafičke pripreme s računalom u grafičkoj proizvodnji", Stubičke toplice, 1991.
3. Seminar: "Sadašnji trenutak modernizacije grafičke industrije", Zagreb 1992
4. Seminar: "Nove tehnologije u grafičkoj pripremi", Krapinske toplice, 1993.
5. Seminar: "Digitalni tisak", Zagreb, 1995.

6. ZNANSTVENE I STRUČNE NAGRADE I PRIZNANJA

1

1. Dijamantna nagrada za digitalni tisak, Brussel, 1996. Xeikon