

Sinteza programa

Anja Ivanišević, Ivan Ristović, Milana Kovačević, Vesna Katanić

4. maj 2018

Sadržaj

1	Recenzent — ocena:	2
1.1	O čemu rad govori?	2
1.2	Krupne primedbe i sugestije	2
1.2.1	Sažetak	2
1.2.2	Naslov rada	2
1.2.3	Pasusi od jedne rečenice	2
1.2.4	Jasnoća rada	3
1.3	Sitne primedbe	5
1.3.1	Formulacije rečenica	5
1.3.2	Štamparske i stilske greške	7
1.3.3	Engleski izrazi	9
1.4	Provera sadržajnosti i forme seminarskog rada	9
1.5	Ocenite sebe	10
2	Recenzent — ocena:	11
2.1	О чему рад говори?	11
2.2	Крупне примедбе и сугестије	11
2.3	Ситне примедбе	12
2.4	Провера садржајности и форме семинарског рада	14
2.5	Оцените себе	16
3	Recenzent — ocena:	17
3.1	O cemu rad govori?	17
3.2	Krupne primedbe i sugestije	17
3.3	Sitne primedbe	18
3.4	Provera sadržajnosti i forme seminarskog rada	20
3.5	Ocenite sebe	21
4	Dodatne izmene	22

Glava 1

Recenzent — ocena: 3

1.1 O čemu rad govori?

Rad govori o automatskom procesu generisanja programskog koda. Jasno su prikazane primene kao i primeri alata koji se koriste. Opisani su izazovi u ovoj oblasti - problem definisanja specifikacija i problem pretrage prostora programa. Prikazan je i CEGIS kao i njegove faze: definisanje specifikacija, sinteza kandidata, verifikacija kandidata i vraćanje podataka potrebnih za dalju sintezu programa.

1.2 Krupne primedbe i sugestije

1.2.1 Sažetak

Smatram da se sažetak nekako naglo prekida. Možda može da se završi rečenicom da će se rad najviše fokusirati na CEGIS ili da se doda neki 'zaključak'. Takođe, fale ključne reči.

[Ispravljeno](#)

1.2.2 Naslov rada

Naslov je u redu, ali smo na času došli do zaključka da treba da, pored toga što nas upućuje u temu rada, bude i privlačan - tema nije isto što i naslov rada.

[Smatramo da naslov rada odgovara tekstu i da je sam po sebi dovoljno zanimljiv.](#)

1.2.3 Pasusi od jedne rečenice

Ima par pasusa koji se sastoje od jedne rečenice. Trebalo bi ili dodati neke rečenice tim pasusima (koje podržavaju to rečeno, pružaju dodatne informacije, daju primere..) ili ih prosto pripojiti nekom od susednih pasusa.

Ovakvi pasusi se javljaju na sledećim mestima:

- Uvod: poslednji pasus.

[Promenjen je poslednji pasus.](#)

- Superoptimizacija: prvi pasus.
[Izmenjeno. Pasus je spojen sa narednim.](#)
- Induktivna pretraga: prvi pasus.
[Izmenjeno.](#)
- CEGIS: poslednji pasus.
[Izmenjeno. Pasus je spojen sa prvim pasusom.](#)
- Primene CEGIS-a: prvi pasus.
[Izmenjeno. Dodata enumeracija.](#)

1.2.4 Jasnoća rada

Par delova rada mi nije bilo jasno, čak ni u prvih par puta čitanja - da li zbog formulacije, nedostatka adekvatnog objašnjenja ili su prosto pisani tako da osoba koja čita mora imati predznanja u nekim oblastima.

Deo rada: Grafika

"Programski opis grafičkih objekata dovodi do bržih proračuna koordinata zavisnih tačaka od tačaka koje imaju slobodne koordinate, što omogućava interaktivne izmene i efikasne animacije. Tehnike sinteze programa mogu uspešno generisati rešenja geometrijskih problema srednjoškolske težine [13]."

Rad ne govori o grafici i referenca na rad na tu temu postoji, ali mislim da bi bilo lepo dodati bar jednu uvodnu rečenicu koja će opisati grafičke objekte, šta je neefikasno u trenutnim proračunima i kako ovaj programski opis to poboljšava - ukratko, ali da postoji uvod u tematiku. Prosto previše naglo počinje odeljak i meni, kao nekom ko se ne razume u računarsku grafiku, je bilo potrebno da posetim gubl radi pojašnjenja.

Ta sekcija je navedena kao primer primene programske sinteze u oblasti računarske grafike zarad dobitaka na performansama i automatizacije određenih procesa. Smatramo da je jasno šta su grafički objekti i ne vidimo razlog za dodatno objašnjenje (samo bi dovelo do više teksta bez vidljivog dobitka). Ideja je bila da se čitaocu samo da uvid u to kako preko grafičkog interfejsa automatski možemo sintetisati programski kod i kako se na osnovu već postojećih grafičkih objekata interaktivnim izmenama mogu sintetisati novi (nešto kao MS Paint), a ne detaljisati o računarskoj grafici i problemima koje ona rešava. Promenili smo redosled pasusa kako bi se čitalac polako uveo u tematiku.

Deo rada: Superoptimizacija

"Formula $\text{prosek} = \frac{x+y}{2}$ može dovesti do prekoračenja. Takođe koristi skupu aritmetičku operaciju deljenja. Alternativa je formula $(x|y) - ((x \oplus y) \gg 1)$."

Smatram da bi bilo dobro prikazati na koji način je ova formula alternativa. Čitalac bi morao da ili dobro bude upoznat sa bitovskim operatorima, ili da provede mnogo vremena tumačeći formulu.

Pretpostavka ovog rada je da čitalac ima osnovna znanja iz programiranja te da nije neophodno da se pravi poseban osvrt na bitovske operatore, s obzirom da razumevanje rada zahteva poznavanje osnova računarstva i informatike. Takođe ne vidimo šta je (osim eventualno bitovskih operatora) nejasno u ovoj formuli.

Deo rada: Definisanje specifikacija

"Ovaj program vrši nama intuitivnu transformaciju niski, ali, na primer, da bi se on automatski generisao korišćenjem FlashFill [12] programa, potrebno je pretražiti prostor koji sadrži milione mogućih rešenja."

Nije baš jasno šta taj prostor sadrži. Nakon čitanja rada, postalo mi je jasno da se radi o prostoru svih mogućih programa koji daju izlaz 'Smith, J.' za ulaz 'John Smith', ali u trenutku prvog čitanja ovog pasusa, više mi je zvučalo kao da se pretražuje prostor koji sadrži milione mogućih niski.

Slazemo se da bi rečenica mogla da navede na pogrešan zaključak. Izmenili smo kako je predloženo.

Deo rada: Enumerativna pretraga

"Glavna ideja je da se prvo na neki način opiše prostor pretrage u kome se nalazi željeni program. To može da se postigne korišćenjem meta-podataka kao što su veličina programa ili njegova složenost. Kada se mogući programi numerišu po osobinama, mogu da se odmah odbace oni koji ne zadovoljavaju prethodno definisane specifikacije."

Meni iz ovog pasusa nije bilo baš najjasnije šta je enumerativna pretraga. Da li ovo znači da mi kažemo da hoćemo program od 100 linija koda, zatim sve programe u prostoru pretrage numerišemo po veličini i zatim odsečemo sve one sa preko 100 linija koda? Nije mi baš jasna poslednja rečenica - oni su numerisani i mi opet gledamo da li zadovoljavaju specifikacije? Ili nam ta numeracija govori da li zadovoljavaju ili ne. Možda bi neki primer ili malo izmenjena formulacija pomogli razumevanju.

Način opisivanja prostora programa u sklopu enumerativne pretrage je van granica ovog rada, pa ga zbog toga ovde nismo detaljnije opisali. Smatramo da je tekst dovoljno jasno napisan, te ga nismo menjali.

Deo rada: Statistička pretraga

"Mašinsko učenje - Tehnike mašinskog učenja mogu doprineti ostalim pretragama uvodeći verovatnoću u čvorove granjanja prilikom pretrage. Vrednosti verovatnoće se uglavnom generišu pre sinteze programa: tokom treninga ili na primer na osnovu datih primera ulaza i izlaza."

Pretpostavljam da verovatnoća služi da se neki delovi prostora pretrage iseku ako je ona manja od nekog broja. Smatram da bi trebalo da stoji objašnjenje kako se generišu kao i čemu te verovatnoće služe, ili da se bar stavi referenca na rad koji govori o tome.

Upotreba mašinskog učenja je van opsega našeg rada. Dodata referenca.

Deo rada: Sinteza vođena uzorom, deo Faza sinteze

"Primetimo da SSA forma jednostavno povezuje komponente međusobno, tako da ako želimo da imamo dva sabiranja moramo da obezbedimo dve komponente za sabiranje. Takođe primetimo da se vrednost o2 ne koristi, takozvani mrtvi kod. U ovom slučaju to je poželjna karakteristika, jer ne moramo unapred tačno da odredimo koliko ćemo imati kojih operacija, već možemo samo da zadamo gornju granicu, a sintezirer će generisati mrtvi kod za komponente koje se ne koriste."

Nije mi jasno šta je poželjna karakteristika - mrtvi kod ili to što imamo više komponenti za sabiranje (pretpostavljam da je kod u pitanju)? Malo je čudno formulisano. Prvo se kaže da su nam potrebne dve komponente za sabiranje a zatim da ne moramo da odredimo koliko ćemo imati kojih operacija. I kako tačno sintezirer generiše mrtvi kod? Ceo pasus je malo zbunjujuć.

[Izmenili smo tekst tako da bude jasniji.](#)

1.3 Sitne primedbe

Ovde će biti izdvojene greške u kucanju, formulacije rečenica koje mi baš nisu zvučale najbolje kao i neki sitni predlozi.

1.3.1 Formulacije rečenica

- Uvod: *'Danas skoro da ne postoji osoba koja nema pristup modernim tehnologijama.'*

Deo 'danas skoro da ne postoji osoba' mi zvuči previše neformalno za jedan seminarski rad. Možda bi bolje bilo 'U današnje vreme, pristup modernim tehnologijama je svima omogućen.' ili slično.

[Slažemo se, ispravili smo kao sto je predloženo.](#)

- Uvod: *'..dao je vetar u leđa..'*

Ovo takođe zvuči malo neformalno. Možda bi bilo bolje iskoristiti 'podstakao' umesto ove sintagme.

[Smatramo da je ova sintagma iako možda malo neformalna poželjna, jer rad čini zanimljivijim i čitljivijim.](#)

- Priprema podataka: *'Priprema podataka predstavlja proces čišćenja, transformacije i pripreme podataka...'*

Loše je formulisano jer izgleda ovako: 'Priprema podataka je priprema podataka'. Drugo 'priprema' može da bude 'prevodenje'.

[Ispravljeno.](#)

- Definisanje specifikacija: *'(možda čak i da deluje komplikovanije neko pisanje samog programa).'*

[Ispravljeno.](#)

Bolje bi zvučalo da piše 'ponekad i komplikovanije od pisanja samog programa'. Ako se autori odluče da ne promene formulaciju, treba ispraviti 'neko' u 'nego'.

[Ispravljeno.](#)

- Definisanje specifikacija: *'Čak i ako bi se to nekako uspelo, opis programa bi mogao da bude toliko obiman kao i sama njegova implementacija.'*

Trebalo bi preformulisati u *'Čak i kad bi se to uspelo, opis programa.'*

Ispravljeno.

- Deduktivna pretraga: *'...sama enumerativna pretraga bi se izgubila pokušavajući da...'*

'Pretraga bi se izgubila' - zvuči malo neformalno za seminarski (naučni) rad.

Ispravljeno.

- Induktivna pretraga: *'Prilikom svake iteracije se generišu neka ograničenja, rešavačem se dođe mogućeg rešenja, a zatim se ono ispita da li je zadovoljavajuće opšte rešenje.'*

Bolje bi zvučalo: *'Prilikom svake iteracije se generišu ograničenja, rešavačem se dođe do mogućeg rešenja a zatim se ispita da li je ono zadovoljavajuće kao opšte rešenje.'* ili nešto tome slično.

Ispravljeno.

- Statistička pretraga: *'Svaka jedinka populacije se ispituje u kojoj meri zadovoljava specifikacije željenog programa.'*

Preformulisati u *'Ispituje se u kojoj meri svaka jedinka zadovoljava.'*

Ispravljeno.

- CEGIS: *'Naime, rešavač bi u svakom slučaju pronašao rešenje za datu formulu, ali kako bi se to desilo u realnom vremenu, CEGIS (eng. Counter-example-Guided Inductive Synthesis) u sebi sadrži posebne tehnike za optimizaciju.'*

Preformulisati u *'Naime, rešavač je u stanju da pronađe rešenje za bilo koju proizvoljnu ulaznu formulu, ali da bi se to desilo u realnom vremenu, ...'*. Takođe, treba izbaciti uvođenje skraćenice CEGIS jer je to već urađeno u uvodu.

Smatramo da je početak rečenice sasvim jasan i formulisan u skladu sa poentom rečenice, pa ga nismo menjali. Izbacili smo uvođenje skraćenice CEGIS.

- Sinteza vođena uzorom, faza verifikacije: *'Ako postoji rešenje mi program koji je kandidat za rešenje prosleđujemo fazi verifikacije.', 'Ako postoji nismo završili.', 'U ovom trenutku ništa ne pitamo uzor.'*

Skroz je u redu da postoji obaraćanje u radu, ali ovde mi zvuči malo nepotrebno. Mi ne prosleđujemo program fazi verifikacije sem ako faza sinteze i faza verifikacije nisu skroz odvojene pa mi moramo ručno da prenosimo izlaz jedne u ulaz druge. Zatim, nismo mi završili već program nije završio. Uzor svakako ne pitamo, ni mi ni naš program, već se ne vrši poređenje ili nešto slično tome.

Ispravljeno.

1.3.2 Štamparske i stilske greške

- Sažetak: '*popravka*' ispraviti u 'popravljanja'.
[Izmenjeno.](#)
- Uvod: '*veca*' ispraviti u 'veća'.
[Izmenjeno.](#)
- Uvod: '*dugokog*' ispraviti u 'dubokog'.
[Izmenjeno.](#)
- Priprema podataka: '*skupoća*' ispraviti u 'visoka cena'.
[Izmenjeno.](#)
- Priprema podataka: '*Alati koji koriste PBE su idealni za ovakav posao [11, 22]*' - fali tačka.
[Izmenjeno.](#)
- Popravka koda: U parčetu koda piše '*if (inhb)*' a ta promenljiva ne postoji. Trebalo bi da piše *inb*.
[Izmenjeno.](#)
- Sugestije prilikom kodiranja: '*programerska okruženja*' ispraviti u 'okruženja za rad'.
[Izmenjeno.](#)
- Grafika: '*jako neprijatno*' ispraviti u 'veoma naporno'.
[Izmenjeno.](#)
- Grafika: '*moгуće je omogućiti korisniku*' ispraviti u 'omogućava se korisniku'.
[Izmenjeno.](#)
- Definisanje specifikacija: '*napradnu*' ispraviti u 'naprednu'.
[Izmenjeno.](#)
- Definisanje specifikacija: '*gore spomenuti*' ispraviti u 'gorepomenuti'.
[Izmenjeno.](#)
- Statistička pretraga: '*..jedna po jedna, i proverava da li..*' ispraviti u '*..jedna po jedna, i proverava se da li..*'.
[Izmenjeno.](#)
- CEGIS: '*SMT*' u drugoj rečenici ispraviti u 'SMT rešavač' jer je SMT tip problema i on ne može da nalazi valuacije.
[Izmenjeno.](#)
- Arhitektura: '*kontra primer*' ispraviti u 'kontraprimer'.
[Izmenjeno.](#)

- Sinteza vođena uzorom: '*random*' ispraviti u 'proizvoljne'.
[Izmenjeno.](#)
- Sinteza vođena uzorom: '*dobijem*' ispraviti u 'dobijen'.
[Izmenjeno.](#)
- Sinteza vođena uzorom: '*sobzirom*' ispraviti u 's obzirom na to'.
[Izmenjeno.](#)
- Sinteza vođena uzorom: '*oracula*' i '*oraculu*' ispraviti u 'uzora' i 'uzoru' jer je ovako nekonzistentno sa ostatkom rada.
[Izmenjeno.](#)
- Sinteza vođena uzorom: '*..ponovo prolazi kroz petlju.*' ispraviti u '..ponovo se prolazi kroz petlju.'
[Izmenjeno.](#)
- Sinteza vođena uzorom: '*bi smo*' ispraviti u 'bismo'.
[Izmenjeno.](#)
- Sinteza vođena uzorom: '*Problem je da..*' ispraviti u 'Problem je to što'.
[Izmenjeno.](#)
- Sinteza vođena uzorom: '*Faza validacije treba da..*' nigde se ne pominje faza validacije, ovo treba ispraviti u 'verifikacije'.
[U ovom delu teksta uvodimo fazu validacije kao dodatnu fazu, tako da to nije greška.](#)
- Stohastička superoptimizacija: '*prtpostaviti*' ispraviti u 'pretpostaviti'.
[Izmenjeno.](#)
- Stohastička superoptimizacija: '*prilagpđenosti*' ispraviti u 'prilagođenosti'.
[Izmenjeno.](#)
- Stohastička superoptimizacija: '*nadalje*' ispraviti u 'ubuduće'.
[Izmenjeno.](#)
- Stohastička superoptimizacija: '*vraca*' ispraviti u 'vraća'.
[Izmenjeno.](#)
- Zaključak: '*dolazi se do pitanja*' ispraviti u 'postavlja se pitanje'.
[Izmenjeno.](#)
- Zaključak: '*Ali ova oblast..*' izbaciti 'ali'.
[Izmenjeno.](#)
- Zaključak: '*S obzirom..*' ispraviti u 'S obzirom na to..'.
[Ispravljen zaključak.](#)

1.3.3 Engleski izrazi

Kod engleskih skraćenica, trebalo bi da je svaka reč napisana velikim početnim slovom, bar po mom mišljenju, jer je već sama skraćenica zapisana velikim slovima. Ako se autori ne slažu sa ovim predlogom, bar treba ustaliti jedan način pisanja - u radu se na nekim mestima piše svaka reč velikim slovom a na nekim mešano. Ovo se javlja na sledećim mestima:

- Uvod: '*SAT (eng. Propositional satisfiability problem)*'
- Uvod: '*SMT (eng. Satisfiability modulo theories)*'
- Uvod: '*CEGIS-u (eng. Counterexample-guided inductive synthesis)*'
- CEGIS: '*CEGIS (eng. Counterexample-Guided Inductive Synthesis)*'

Usaglašen je način pisanja engleskih reči.

1.4 Provera sadržajnosti i forme seminarskog rada

1. Da li rad dobro odgovara na zadatu temu?
Da. Prikazano je zašto je potrebno baviti se ovom oblasti, koji izazovi postoje, šta je trenutno aktuelno u ovom polju. Mnogo se može naučiti a opet nije preopširno.
2. Da li je nešto važno propušteno?
Ne. Prikazana je motivacija ('olakšavanje života' programerima, kraće vreme pisanja koda.), izazovi (definisanje specifikacija i pretraga) a na kraju je veoma detaljno prikazan CEGIS koji predstavlja aktuelan pristup sintezi programa - što i jesu bili inicijalni zahtevi postavljeni pred autore.
3. Da li ima suštinskih grešaka i propusta?
Suštinskih - ne. Samo par stvari koje bi, po mom mišljenju, trebalo malo bolje pojasniti. Rad je jasan i informativan.
4. Da li je naslov rada dobro izabran?
Možda. Naslov se poklapa sa temom a trebalo bi da bude malo preformulisano - da bude i 'privlačan' a i informativan.
5. Da li sažetak sadrži prave podatke o radu?
Da. Sve što je rečeno da će biti opisano je i bilo opisano.
6. Da li je rad lak-težak za čitanje?
Srednje težine. Ima delova koji su laki za čitanje i veoma interesantni (što čini veći deo rada) ali ima i onih koji su konfuzni.
7. Da li je za razumevanje teksta potrebno predznanje i u kolikoj meri?
Jeste, ali ne iz oblasti sinteze programa, već nekih drugih oblasti. Mi, kao studenti MATF-a, možemo lako razumeti rad, ali na primer, neko ko nije slušao računarsku grafiku ili nije učio bitovske operatore će se zbuniti na par mesta u radu.

8. Da li je u radu navedena odgovarajuća literatura?
Jeste. Eventualno bi bilo dobro dodati literaturu za delove koji su istaknuti u delu 2 ako se autori odluče da ih ne objasne detaljnije (kao na primer za odeljak Mašinsko učenje u okviru statističke pretrage).
9. Da li su u radu reference korektno navedene?
Da, reference u radu su pravilno navedene. Sama literatura je u dobrom formatu.
10. Da li je struktura rada adekvatna?
Da. Sadrži sažetak, sadržaj (možda može da sadrži i podnaslove, ako ne prelazi prvu stranu), pravilno napisan uvod i zaključak, reference.
11. Da li rad sadrži sve elemente propisane uslovom seminarskog rada (slike, tabele, broj strana...)?
Da. Slika 1 u radu je u stvari tabela, pa bi možda to trebalo preimenovati. Ali svakako je sve ispoštovano.
12. Da li su slike i tabele funkcionalne i adekvatne?
Da. Jasne su same po sebi i na njih je referisano iz samog teksta.

1.5 Ocenite sebe

Negde sam na granici između 'srednje upućen' i 'malo upućen'. Pročitah par članaka na ovu temu pre početka recenzije koje je profesorka predložila u mejlu.

Glava 2

Recenzent — ocena: 4

2.1 О чему рад говори?

Семинарски рад говори о томе шта представља синтеза програма, каква је њена улога у рачунарству, који су најчешћи проблеми приликом синтезе програма и како се они превазилазе. Основна замисао је да особа зада рачунару какав програм жели да добије, а специјалан програм за аутоматско генерисање кода, или тзв. синтезер програма, аутоматски генерише имплементацију жељеног програма. Међутим, то није једина примена синтезе програма. У раду је дато неколико примена синтезе програма. Такође, описани су главни потпроблеми синтезе програма, што су дефинисање спецификација жељеног програма и претраживање простора могућих програма ради проналажења оног који задовољава задате спецификације. При томе, оба потпроблема имају своје проблеме. Дефинисање жељене спецификације уопште није једноставно као што на први поглед делује, а простор програма експоненцијално расте, па се јављају проблеми како сузити тај простор, како би се цела синтеза програма завршила у реалном времену. На крају рада детаљно је описан метод CEGIS, који је заснован на следећем једноставном принципу: ако се за последњи синтетисани програм пронађе контрапример, тај програм није решење и треба наставити даље, а ако контрапримера нема, тај програм је решење. Дато је неколико примера који користи методу CEGIS, помоћу којих је јасно како она функционише.

2.2 Крупне примедбе и сугестије

Рад је веома квалитетно написан. У самој структури рада не треба вршити никакве измене. Међутим, требало би додати неколико ситница, да би рад био квалитетнији. На пример, у пододељку 2.2 Поправка кода, у примеру датом на слици 1, није дата спецификација програма, па је пример недовољно јасан. У пододељку 2.5 Супероптимизација требало би боље објаснити где је ту примена синтезе програма, пошто из тренутног текста то није јасно. Мало је јасније тек када се прочита део 4.2.2 Стохастичка супероптимизација, али би ипак требало боље објаснити у 2.5. С овим допунама, рад би био комплетиран.

[2.2 - Specifikacija je zadata tabelom ulaza i izlaza.](#) [2.5 - Ne vidimo kako to iz](#)

ovog teksta nije jasno. Data je specifikacija (naći prosek), a sintetisan efikasniji izraz koji ispunjava zadatu specifikaciju. U narednom pasusu je postavljena i prečica na deo rada koji detaljnije opisuje kako se zapravo pretražuje prostor formula kako bi se našla baš ova formula. U sekciji 2 smo navodili primene sinteze, a detaljniji opis sledi u narednim poglavljima.

2.3 Ситне примедбе

Иако је рад веома квалитетно написан, уочен је извештан број граматичких, стилских и штампарских грешака. Све уочене грешке су наведене у даљем тексту.

У сажетку, у четвртој реченици уместо речи поправка треба ставити реч поправки, због слагања у падежу.

[Ispravljeno.](#)

У уводу су примећене следеће грешке:

1. У другој реченици написана је реч „веса” уместо речи „већа”.

[Ispravljeno.](#)

2. У четвртој реченици написана је реч „имплементација” у погрешном падежу (треба „имплементацији”).

[Ispravljeno.](#)

3. У петој реченици написана је реч „дугоког”. Треба је исправити на „дубоког” или „дугог

[Ispravljeno.](#)

4. У последњем пасусу је неправилно преломљена реч „CEGIS-и”. Потребно је спречити прелом речи или додати још неку реч како не би једно слово било пренето у наредни ред.

[Ispravljeno.](#)

У пододељку 2.1 Припрема података, примећене су следеће грешке:

1. Уместо речи „полу-структурираног” треба ставити реч „полуструктурираног”.

[Ispravljeno.](#)

2. У последњем пасусу у првој реченици треба ставити „измене” уместо „изменама”, да би била у исправном падежу.

[Tekst je uredno napisan.](#)

У пододељку 2.2 Поправка кода, у коду функције `buggy` употребљена је недеklarисана променљива `inhb`. Аутори су вероватно мислили на променљиву `inb` која је аргумент функције.

[Ispravljeno.](#)

У пододељку 2.3 Сугестије приликом кодирања, примећене су следеће грешке:

1. У другој реченици уместо речи „синтезизери” треба ставити реч „синтезери”. Такође, требало би додати запету после речи „токене”.

[Ispravljeno.](#)

2. У трећој реченици би требало превести „type-directed completion” на српски, да би било у складу с претходним појмом, који је преведен.

Ispravljeno.

У пододељку 2.6 Конкурентно програмирање, у другом пасусу у четвртој реченици треба ставити реч „додајући” уместо речи „додавајући”.

Ispravljeno.

У пододељку 3.1 Дефинисање спецификација, примећене су следеће грешке:

1. У другом пасусу, у првој реченици уместо речи „неко” треба ставити реч „него”.

Ispravljeno.

2. У трећем пасусу, у другој реченици уместо речи „као” треба ставити реч „колико”, због претходно употребљене речи „толико”.

Ispravljeno.

У делу 3.2.1 Енумеративна претрага, треба избацити цртице у речима „мета-података” и „полу-одлучива”. У делу 3.2.2 Дедуктивна претрага, треба писати „одозго на доле” уместо „одозго - на доле”.

Ispravljeno.

У делу 3.2.3 Технике са ограничењима, примећене су следеће грешке:

1. У делу о генерисању ограничења, после речи „ограничењима” треба додати запету.

Ispravljeno.

2. У делу о разрешавању ограничења, у другој реченици уместо речи „неког” треба ставити реч „неки” („за неки од решавача”).

Ispravljeno.

У делу 3.2.5 Статистичка претрага, примећене су следеће грешке:

1. У првој реченици треба додати запету после речи „програма”.
2. У другом пасусу уместо речи „грањања” треба ставити реч „гранања”, а у последњој реченици треба ставити запете пре и после синтагме „на пример”.
3. У четвртом пасусу реч „техника” треба преломити као „те-хника”. У другој реченици је пожељно додати реч „се” после речи „проверава”.

Ispravljeno.

У одељку 4 CEGIS, у другом пасусу треба ставити велико почетно слово у упитној реченици која је под наводницима и треба обрисати тачку после знака навода на крају. Такође, уочена је неконзистентност у писању речи „контрапример”. На почетку одељка је писано „контра-пример”, а касније у тексту као две речи, „контра пример”. Оба начина су неисправна и треба писати састављено „контрапример”.

Ispravljeno.

Препорука је да се слици 2 (CEGIS петља) смањи величина, како би стала на крај 8. стране. Тако ће се наћи одмах испод текста на коме се реферише на њу.

Zbog dužine teksta na prethodnoj strani slika svakako ne bi stala čak i ako bi bila manja.

Наслов подељка 4.2 је неправилно написан. Треба додати цртицу да би писало „CEGIS-a”. Такође, део 4.2.1 је писан у активном стилу, док је остатак текста писан у пасивном стилу, што нарушава конзистентност. Примећене су и следеће грешке:

1. У параграфу о фази верификације, треба искосити слово „z” и одмах након тога исправити реч „раликује” у „разликује”. Такође, у петој реченици треба заменити реч „ја” са „су”. Треба додати запете после синтагме „другим речима”, у реченици „Ако постоји, нисмо завршили” и после синтагме „на пример” у другом пасусу.

4.2.1 - Smatramo da je, zbog korišćenja aktivnog stila pisanja, tekst mnogo razumljiviji. Zbog toga ovo nismo menjali. Ostale primedbe su prihvaćene i izmenjene.

2. Треба исправити неисправну реч „собзиром” у две речи „с обзиром” и треба додати запету после „међутим”.

Ispravljeno.

3. У параграфу о повратном кораку треба превести реч „oracle”, или је у крајњој мери употребити транскрибовано на оракл.

Ispravljeno.

У делу 4.2.2 Стохастичка супероптимизација, у параграфу о фази верификације, треба исправити реч „прилагљености” у „прилагођености”.

Ispravljeno.

У делу 4.2.3 Енумеративна претрага, у параграфу о фази верификације, треба додати запете у првој реченици после синтагме „тест примерима” и у последњој реченици пре „завршили смо”.

Ispravljeno.

Коначно, у закључку треба да се исправи реченица тако да гласи овако: „Иако теоретски делује да овај приступ није довољно ефикасан, те да ће се извршати предуго, CEGIS је, као водећи представник ове групе, у пракси показао неочекивано добре резултате.”

Ispravljeno.

2.4 Провера садржајности и форме семинарског рада

1. Да ли рад добро одговара на задату тему?
Рад веома добро одговара на задату тему. Читаоцу је јасно објашњено шта је синтеза програма, како синтезери уопштено раде и како конкретно ради једна од метода (CEGIS).
2. Да ли је нешто важно пропуштено?
Није ништа важно пропуштено. Једине крупније замерке су на дате на одељак о применама синтезе програма, јер су неке примери недовољно добро објашњени.

3. Да ли има суштинских грешака и пропуста?
Суштинских грешака нема. Рад је веома квалитетно написан и сем малих пропуста у делу о применама синтезе програма, једине грешке су граматичке и штампарске природе.
4. Да ли је наслов рада добро изабран?
Наслов рада је добро изабран. Садржи сасвим довољно информација да заинтересује неког будућег читаоца да прочита рад.
5. Да ли сажетак садржи праве податке о раду?
Сажетак је коректно написан. Говори о свему што стоји у раду и шта ће читалац из њега сазнати.
6. Да ли је рад лак-тежак за читање?
Рад није тежак за читање. Из једног читања се може лепо разумети суштина рада. Тежина читања углавном потиче од тежине саме области о којој рад говори.
7. Да ли је за разумевање текста потребно предзнање и у коликој мери?
За разумевање рада потребно је извесно предзнање, углавном из алгоритмике, јер се помињу ствари као што је претрага, као и SAT и SMT решавачи. Читалац, који је упућен у те појмове, без проблема ће разумети цео рад, док ће осталим читаоцима тај део бити мало нејасан.
8. Да ли је у раду наведена одговарајућа литература?
У раду је наведена одговарајућа литература. Постоји како литература за саму синтезу програма, тако и литература за конкретне програме који се њоме баве, примене наведене у раду (поправка кода, оптимизација и друго), као и SAT и SMT решаваче, који су веома важни за примене у синтези програма.
9. Да ли су у раду референце коректно наведене?
Референце су коректно наведене. На сваком месту где се јавља неки од важних концепата синтезе или програми који се користе у њеним применама, наведена је одговарајућа референца.
10. Да ли је структура рада адекватна?
Рад је адекватно структуриран. Има сажетак, садржај, увод, разраду, закључак и литературу. Све је наведено у правилном редоследу и сваки од њих садржи потребне информације.
11. Да ли рад садржи све елементе прописане условом семинарског рада (слике, табеле, број страна...)?
Рад садржи све елементе прописане условима. Има 14 страна, што одговара условима, пошто има 4 аутора. Има 23 референце, што је скоро дупло више од најмањег траженог броја референци за 4 аутора. Такође, рад има 3 слике, на којима се налазе и табеле, те испуњава и ове услове.
12. Да ли су слике и табеле функционалне и адекватне?
Слике и табеле су функционалне и адекватне. Постоје 3 слике и две табеле у оквиру тих слика. Табеле представљају улазе и излазе програма који се у одговарајућем тексту описује и помажу лакшем

разумевању текста. Једна слика која не садржи табелу на адекватан начин описује како CEGIS функционише.

2.5 Оцените себе

У област коју рецензирам сам потпуно неупућен, с обзиром на то да ми до момента читања овог рада није ни било познато да постоје програми који се баве аутоматским генерисањем кода. Читањем овог рада сазнао сам за једну нову, занимљиву и корисну област рачунарства и стекао знање о томе колико се до данас постигло у тој области.

Glava 3

Recenzent — ocena: 4

3.1 O cemu rad govori?

U radu "Sinteza programa" je objašnjeno sta predstavlja pojam sinteze programa, kao i cime se bavi oblast. Objašnjena je motivacija za razvitak ove oblasti, kako bi se programerima olakšao posao, a takode kako bi i "obicni" korisnici mogli da pišu neke programe. Opisane su i primene oblasti kao što su primena podataka, popravka koda itd. Objašnjeno je da specifikacija programa koji sintezirer treba da generiše mora da bude adekvatna io detaljna da bi to bilo moguće. Takode je pomenut problem prostora programa koji se mogu generisati i kako je taj prostor jako teško pretražiti. Opisane su i tehnike pomocu kojih se to može postici. Jedna od njih (CEGIS) je detaljno opisana.

3.2 Krupne primedbe i sugestije

- U sažetku ili uvodu dodati da li je sama oblast slicna mašinskom ucenju ili automatskom rezonovanju. Da li je možda podoblast neke od ovih, ili samo koristi neke od ideja i tehnika istih.
Ove oblasti imaju nekih zajedničkih odlika (npr. SAT), ali suštinski nisu slične. Možda je moguće primeniti mašinsko učenje u algoritmima za pretrage ili eventualno optimizacije, ali to je van granica rada (ideja je prikazati osnovne ideje i primene programske sinteze).
- sekcija 2.6 Konkurentno programiranje - Možda dodati, malo objasniti (kratko), na koji nacin sam AGS algoritam određuje da li se menja apstrakcija ili se menja program. Da li su u pitanju metode mašinskog ucenja ili nesto drugo?
Navedeno je da se to određuje nedeterministički, najčešće slučajnim izborom se bira jedno od ta dva ali ne postoji jedinstveni pristup već zavisi od algoritma. Smatramo da je izraz "nedeterministički" šavim na mestu.
- sekcija 3.1 Definisane specifikacije - Da li postoje alati koji omogućavaju da se program "sam" generiše do neke tacke, a da tek onda korisnik definiše ostatak? Da ne bude opisane situacije gde na pocetku korisnik radi, a zatim sintezirer, vec obrnuto. Mislim da je ukoliko je to moguće lepo pomenuti i dati referencu i ka tom alatu.

Sintezeru je neophodno dati neka početne specifikacije jer inače on ne bi imao nikakve smernice kuda da vrši pretragu za rešenjem. Ukoliko bi alat sam imao definisane specifikacije za uopštene primere programa, to bi bilo moguće. Međutim, istraživanjem ove oblasti nismo došli do ovakvih alata.

- sekcija 3.2 Pretraživanje prostora programa - Mislim da bi bilo dobro dodati referencu ka prostoru mogućih programa. Ili u jednoj recenici opisati šta je. Pretpostavke se mogu napraviti, ali bi bilo dobro da postoji opis ili referenca ka pojmu. Referenca ili objašnjenje se mogu naći i u sekciji 3 Izazovi gde je pojam prvi put pomenut.

Dodata je definicija prostora programa u 3.2.

- sekcija 3.2.3 Tehnike sa ograničenjima - Pri dodavanju pretpostavki o rešenju u koraku generisanja ograničenja, u kojoj meri se daju ove pretpostavke? Da li se može desiti da davanjem velikog broja pretpostavki o rešenju dode do preprilagodavanja i u koraku razrešavanja ograničenja nademo samo jedno rešenje?

Ideja je da se ograničenjima opiše šta željeni program treba da zadovoljava. U ovom slučaju ne postoji preprilagođavanje, jer je korisniku baš potreban program koji zadovoljava sva ograničenja. Prilikom davanja velikog broja ograničenja, verovatnije je da korisnik da neka pogrešna i da zato ne dobije očekivani rezultat (ukoliko da ograničenja koja se međusobno isključuju).

- sekcija 3.2.5 Statistička pretraga - Nadogradnju programa vrši sintezer ili korisnik?

Probabilističko zaključivanje je stvar koju vrši sam sintezer. Međutim, uz dodatne optimizacije, korisnik bi mogao da prati ovo nadograđivanje i svojim akcijama ga usmerava.

- sekcija 4.2.1 Sinteza vođena uzorom - Šta znaci da u fazi validacije treba proveriti da li program zadovoljava sve ulaze? Na koji način se generišu "svi ulazi"? Generalno, šta se podrazumeva pod ovim pojmom? Sve moguće ulaze ili njihove kombinacije nije moguće generisati za malo veće programe. O velikim softverima za kakve bi sinteza mogla jednog dana da se koristi da i ne govorimo.

U tekstu nije nigde rečeno ni naznačeno da faza validacije treba da generiše sve moguće ulaze. Potrebnu validaciju je moguće izvršiti dokazivanjem da je određena formula zadovoljiva ili na neki drugi način u zavisnosti od vrste problema. Detaljan postupak ove validacije nismo uspeali da nađemo pa tekst ostaje neizmenjen.

3.3 Sitne primedbe

- Sažetak - linija 6, popravki umesto popravka.
Izmenjeno. Taj tekst se ne nalazi u novoj verziji.
- Uvod - linija 2, veka umesto veka.
Izmenjeno.
- Uvod - linija 8, dubokog umesto dugokog.
Izmenjeno.

- sekcija 2.3 Sugestije prilikom kodiranja - linija 3, sintezeri umesto sintesizeri. Možda i nije greška, ali u celom tekstu se javlja rec sintezeri, nigde više sintesizeri.
[Izmenjeno.](#)
- sekcija 2.6 Konkurentno programiranje - pasus 2, linije 4 i 6, mislim da je bolje "kršenje" nego "prekrčenje".
[Izmenjeno.](#)
- sekcija 2.6 Konkurentno programiranje - pasus 2, linija 8 dodajuci umesto dodavajuci.
[Izmenjeno.](#)
- sekcija 3.1 Definisanje specifikacija - linija 1, na nacin na koji to korisnik definiše umesto na nacin koji to korisnik definiše.
[Izmenjeno.](#)
- sekcija 3.1 Definisanje specifikacija - linija 5, definisanjem umesto definisanje.
[Izmenjeno.](#)
- sekcija 3.1 Definisanje specifikacija - linija 7, nego umesto neko.
[Izmenjeno.](#)
- sekcija 3.1 Definisanje specifikacija - poslednji pasus, linija 1, naprednu umesto napradnu.
[Izmenjeno.](#)
- sekcija 3.2.3 Tehnike sa ogranicenjima - stavka Razrešavanje ogranicenja, linija 3, neki umesto nekog.
[Izmenjeno.](#)
- sekcija 3.2.4 induktivna pretraga - linija 6, rešavacem se dode do moguceg rešenja umesto rešavacem se dode moguceg rešenja.
[Izmenjeno.](#)
- sekcija 3.2.5 Statisticka pretraga - stavka Genetsko programiranje, linija 3, mogucih umesto mogucih.
[Izmenjeno.](#)
- sekcija 4 CEGIS - linija 5 koji umesto koje.
[Izmenjeno.](#)
- sekcije 4 CEGIS i 4.1 Arhitektura - u prvoj sekciji piše kontra-primer, a u drugoj kontra primer. Nekonzistentno.
[Izmenjeno.](#)
- sekcija 4.2.1 Sinteza vodena uzorom - podsekcija Faza verifikacije, pasusu ispod slike, linija 5, programi umesto program.
[Izmenjeno.](#)
- sekcija 4.2.1 Sinteza vodena uzorom - podsekcija Faza verifikacije, pasusu ispod slike, linija 7, s obzirom umesto sobzirom.
[Izmenjeno.](#)

- sekcija 4.2.2 Stohasticka superoptimizacija - linija 3, pretpostaviti umesto prtpostaviti.
[Izmenjeno.](#)
- sekcija 4.2.2 Stohasticka superoptimizacija - podsekcija Faza verifikacije, linija 4, prilagodenosti umesto prilagpdenosti.
[Izmenjeno.](#)

3.4 Provera sadržajnosti i forme seminarskog rada

1. Da li rad dobro odgovara na zadatu temu? Da.
2. Da li je nešto važno propušteno? Ne.
3. Da li ima suštinskih grešaka i propusta? Ne.
4. Da li je naslov rada dobro izabran? Da.
5. Da li sažetak sadrži prave podatke o radu? Da.
6. Da li je rad lak-težak za citanje? Lak. Jako lepo napisano.
7. Da li je za razumevanje teksta potrebno predznanje i u kolikoj meri? Ne previše znanja, ali bi neke delove bilo možda malo lakše razumeti. Za delove koji nisu detaljno opisani postoje odgovarajuće reference, tako da je moguće sve pronaći.
8. Da li je u radu navedena odgovarajuća literatura? Da.
9. Da li su u radu reference korektno navedene? Da.
10. Da li je struktura rada adekvatna? Da.
11. Da li rad sadrži sve elemente propisane uslovom seminarskog rada (slike, tabele, broj strana...)? Slika 3, "Primer rada faze verifikacije" je u formi tabele. Ukoliko se ona računa kao tabela postoji sve, ukoliko ne, fali tabela. Takođe Slika 1 poseduje deo koji lici na tabelu.
12. Da li su slike i tabele funkcionalne i adekvatne? Da.

3.5 Ocenite sebe

Rekao bih da je moj nivo upucenosti u datu temu "skoro neupucen".

Glava 4

Dodatne izmene