UDC: 510.646:004.8 Info M: str. 11-19

АУТОМАТСКО ПРОВЕРАВАЊЕ НЕФОРМАЛНИХ ДОКАЗА ТЕОРЕМА СРЕДЊОШКОЛСКЕ ГЕОМЕТРИЈЕ
AUTOMATED VERIFICATION OF INFORMAL PROOFS FROM HIGH SCHOOL GEOMETRY

Сана Стојановић Ђурђевић
Математички факултет у Београду, Универзитет у Београду

РЕЗИМЕ: Доказивање теорема уз помоћ рачунара је област којој се поклања све више пажње у последњих неколико десетина година. Програми за доказивање теорема се из дана у дан све више развијају, постају флексибилнији и доказују све теже теореме. Њихов развој је често вођен идејом што једноставнијег коришћења и са циљем приближавања новим корисницима. Упркос томе, употреба програма за доказивање теорема није толико распрострањена међу широм популацијом као што су ученици средњих школа и студенти. Програм приказан у овом раду има за циљ приближавање области доказивања теорема уз помоћ рачунара управо њима. Програм омогућава аутоматско проверавање неформалних доказа теорема средњошколске геометрије и генерише формални доказ теореме проверив уз помоћ рачунара као и доказ записан на српском језику. Биће приказана његова употреба на доказима десетак теорема из уџбеника за средњу школу. Веома је једноставан за коришћење тако да га могу употребљавати чак и корисници који немају никакво искуство са доказивањем теорема уз помоћ рачунара. Поред примена у образовању, програм може бити користан и математичарима приликом генерисања нових доказа.

КЉУЧНЕ РЕЧИ: аутоматско доказивање теорема, интерактивно доказивање теорема, кохерентна логика, средњошколска геометрија

ABSTRACT: Computer assisted theorem proving has been gaining a lot of attention over the past couple of decades. Programs used for such tasks have steadily been improving, gaining in flexibility and capability to tackle an ever-increasing set of more complex theorems. While the development of these software tools has been guided by the notion of user-friendly interfaces and ease of use, they have not gained much popularity in a broad population which includes high school and university students. The program we present in this paper aims to bridge this gap as it is specifically tailored for this audience. It allows for automated checking of informal proofs that are often found in high school geometry curricula and generates formal, computer verifiable proof along with a proof written in Serbian. We demonstrate its use on a handful of proofs from high school textbooks. The program is very simple to use and does not require a user to have any prior experience with automated theorem proving software. In addition to being a useful educational tool, the program could also be used by mathematicians for producing new original proofs.

KEY WORDS: automated theorem proving, interactive theorem proving, coherent logic, high school geometry

1 УВОД

У последњих неколико десетина година појављује се све више критика упућених доказима који се могу наћи у класичним математичким књигама па и у математичким уџбеницима. Испоставља се да, строго гледано, велики број тих доказа није довољно прецизно написан, а у неким случајевима чак постоје и грешке [20]. Провера исправности доказа теорема се до пре неколико деценија ослањала само на ручну проверу доказа како од стране аутора доказа тако и од стране других математичара, обично резензената. С обзиром на то да су докази теорема постајали све компликованији, дужи и тежи за праћење, појавила се потреба за рачунарским проверавањем доказа теорема [21].

Први значајнији помак у верификацији доказа уз помоћ рачунара се јавља 1967-е у оквиру пројекта Automath [10]. Основна идеја тог пројекта је била коришћење рачунара за проверавање доказа који корисник уноси. Од појаве те идеје до данас направљен је изузетан напредак. Програми намењени проверавању и писању доказа теорема различитих теорија називају се интерактивни доказивачи теорема. Њихова употреба постаје све популарнија последњих година [16,17].

Програми за аутоматско доказивање теорема, где програм самостално конструише доказ теореме, су моћни алати који се најчешће користе за доказивање тврђења која припадају специфичним доменима (верификација софтвера и хардвера, проблеми планирања и распоређивања, итд.). Намењени су корисницима који се баве доказивањем јако тешких теорема (са хиљадама променљивих) или раде на веома важним проблемима за које је неопходно бити сигуран у исправност неког тврђења.

Последњих година све више се комбинују аутоматски и интерактивни приступ доказивању теорема и користе се чак и за разматрање општих математичких проблема који су вековима присутни[1]. Осим тога, све чешће се користе и за решавање проблема свакодневне математике. Упркос свему томе и даље нису познати широј публици и математичари их слабо користе. Тренутно су програми за аутоматско и интерактивно доказивање теорема првенствено намењени експертима. Време које је потребно за савладавање таквих алата је директно пропорционално добити од употребе тих алата. У случају почетника, који желе да испробају аутоматско и интерактивно доказивање теорема без великог ула-

1 Интерактивни доказивачи теорема су коришћени за формализацију Хилбертове књиге *"Grundlagen der Geometrie"* [12,22,23] и доказивање еквивалентности између Хилбертовог аксиоматског система и аксиоматског система Тарског [9], док је комбинација аутоматских и интерактивних доказивача теорема коришћена за формализацију књиге Тарског *"Metamathematische Methoden in der Geometrie"* [25,26,40].

гања времена, савладавање таквих алата је често превише крупан залогај и знатно компликованије и захтевније него решавање оригиналног проблема који их занима.

Доказивање теорема уз помоћ интерактивних доказивача теорема је често процес који траје знатно дуже него што би се очекивало, чак и када корисник има пред собом доказ теореме написан на папиру. Разлог за то је што човек у свом природном резоновању несвесно прави одређене претпоставке и изводи закључке које није потребно експлицитно навести да би се доказ могао пратити. Приликом презаписивања доказа у језик интерактивног доказивача често се испоставља да у неформалном доказу недостају одређени кораци и закључци који су неопходни за креирање формалног доказа. Одређивање тих корака није увек тривијалан задатак.

Средњошколска геометрија, односно теореме које се у оквиру ње доказују, се често сматра једноставном и докази тих теорема се најчешће наводе са врло мало информација у њима. Међутим, чак ни ти докази се не могу увек једноставно преписати у доказе записане на језику програма за интерактивно доказивање теорема. С друге стране, студенти и ученици средњих школа те доказе савршено разумеју. Чињеница да су ти докази преживели неколико стотина година у математичким уџбеницима говори о томе да су ти докази „довољно прецизни“ за ученике и да су употребљиви у свакодневној математичкој пракси. Осим тога, додавање детаља (које формално доказивање теорема захтева) би учинило те доказе опширнијим, тежим за праћење, и умањило би могућност ученика да активно прате и слушају предавања.

Суштински, формално доказивање теорема и доказивање теорема у образовању имају различите циљеве. Од формалних доказа се очекује максимална прецизност и несумњива исправност доказа, док се од доказа у образовању очекује да дају објашњење зашто одређено тврђење (теорема или задатак) важи. Поставља се питање да ли је могуће направити спој те две области и креирати систем који би аутоматски проверио исправност доказа теореме из уџбеника, без мукотрпног задатка записивања тог доказа у оквиру програма за интерактивно доказивање теорема.

Систем ArgoGeoChecker представља први корак на путу од неформалног доказивања теорема на папиру до формалног доказивања теорема уз помоћ рачунара. Корисник пред собом има доказ (који је преузео из уџбеника или је сам формулисао), из њега издваја помоћна тврђења која су у њему наведена и прослеђује их систему ArgoGeoChecker који ће проверити исправност тог доказа. Као резултат примене овог система корисник ће добити аутоматски генерисан документ који је формално проверив од стране интерактивних доказивача теорема. У наставку текста тај документ ћемо звати формални аргумент исправности доказа. Поред тога, систем генерише и документ записан на природном језику у ком је аргумент исправности доказа записан у форми читљивој за човека.

Систем ArgoGeoChecker користи неколико програма за аутоматско доказивање теорема, као и XSL подршку за генерисање формалних доказа (записаних на језицима програма за интерактивно доказивање теорема Isabelle [27,46] и Coq [43]) и доказа на природном језику (на српском и енглеском језику). У овом раду биће приказана једноставност предложеног приступа кроз његову примену на десетак теорема и задатака средњошколске геометрије.

2 ДОКАЗИВАЊЕ ТЕОРЕМА УЗ ПОМОЋ РАЧУНАРА

Два основна начина доказивања теорема уз помоћ рачунара су аутоматско доказивање теорема и интерактивно доказивање теорема. Иако се граница између та два приступа последњих година све више смањује, основна разлика се огледа у нивоу информација које корисник задаје доказивачу, као и у изгледу самог доказа који се добија.

Аутоматски доказивачи теорема

У највећем броју случајева аутоматски доказивачи теорема се користе само за проверавање исправности неког тврђења. Деле се на доказиваче општег типа [32] и на доказиваче развијене за специфичне теорије. Иако постоје аутоматски доказивачи који су намењени области геометрије, у овом раду биће коришћени само доказивачи општег типа базирани на методу резолуције који доказују теореме које припадају логици првог реда: Vampire [31] и E [33].

Информације које аутоматски доказивачи теорема пружају кориснику у току процеса доказивања се разликују од доказивача до доказивача. У неким случајевима дају само одговор да/не на питање да ли је неко тврђење теорема (у оквиру аксиоматског система који се задаје) или дају доказ теореме који је записан у формату који није читљив за човека.

Програми за аутоматско доказивање теорема нису подразумевано поуздани. Да би се могли сматрати поузданим, потребно је или доказати коректност имплементације или показати исправност доказа које ти програми генеришу (у случајевима када генеришу доказ). Из тог разлога се све чешће користе у интеракцији са интерактивним доказивачима теорема.

Интерактивни доказивачи теорема

Интерактивни доказивачи теорема су програми који се користе за конструисање и проверавање формалних доказа уз помоћ рачунара. Докази се записују на објектном нивоу у терминима аксиома и правила извођења. Процес интерактивног доказивања теорема се најчешће изводи ручним уношењем доказа и може бити временски веома захтеван. Одређен ниво аутоматизације постоји у интерактивним доказивачима теорема (кроз разне тактике и алат које ти доказивачи нуде) али је ефикасно коришћење аутоматизације (а самим тим и кратко и језгровито записивање формалних доказа) углавном резервисано за искусније кориснике. Најпопуларнији доказивачи у којима је урађено највише формализација су интерактивни доказивачи теорема Isabelle, Coq, Mizar [44] и HOL-light [18].

За разлику од аутоматских доказивача теорема, интерактивни доказивачи нису толико моћни (у смислу аутоматизације процеса), али су у потпуности поуздани и гарантују исправност написаних доказа. Њихово резоновање се заснива на малом броју правила која се могу ручно проверити.

Интерактивно доказивање теорема је, поред своје основне сврхе (за проверавање већ написаних доказа), донело нове могућности примена програма за аутоматско доказивање теорема (који морају на одређен начин бити прилагођени за рад у оквиру интерактивних доказивача теорема). Последњих година јавља се све више пројеката који се баве интеграцијом аутоматских и интерактивних доказивача теорема [7,8,11].

Кохерентни доказивач теорема ArgoCLP

Кохерентна логика је део логике првог реда [19] који се састоји од формула приказаних на слици 1.

$$A_1(x) \wedge \ldots \wedge A_n(x) \Rightarrow \exists y_1 B_1(x, y_1) \vee \ldots \vee \exists y_n B_n(x, y_n)$$

Слика 1: *Општи облик кохерентне формуле*

Формуле кохерентне логике су имплицитно универзално квантификоване формуле. Кохерентна логика не дозвољава употребу негације у свом запису. Уместо коришћења негације за сваки предикат R, који се јавља у теорији са којом се ради, уводи се нови предикат nR и аксиоме R ∧ nR => ⊥ и R ∨ nR [30]. Кохерентна логика је препозната од стране неколико аутора [2,15] као погодан домен за изражавање значајних делова стандардне математике (посебно геометрије) и све чешће се користи у оквиру пројеката формализације математичког знања [4,5,14].

Аутоматски доказивач теорема ArgoCLP [38] је генерички доказивач теорема који се базира на кохерентној логици. Има могућност рада са формулама произвољне теорије (чије су дефиниције, аксиоме и теореме записане у кохерентном формату). Подржава TPTP [42] формат улаза и генерише доказ у XML формату [37]. Опремљен је скупом XSL алата који трансформишу доказ теореме у формалне доказе записане на језицима интерактивних доказивача теорема Isabelle и Coq, као и у доказе записане на српском и енглеском језику.

3 ОПИС СИСТЕМА ARGOGEOCHECKER

Систем за проверавање неформалних доказа ArgoGeoChecker намењен је аутоматској провери доказа теорема и решења задатака средњошколске геометрије. У наставку текста нећемо правити разлику између појмова теорема и задатак, и доказа теореме и решења задатка.

Приликом доказивања теореме ученик разматра тврђење теореме и аксиоме које има на располагању, записује премисе тврђења и црта дијаграм који их приказује. Након тога записује помоћна тврђења која јасно илуструју редослед извођења у току доказа. Тако формиран скуп логичких формула зваћемо неформални доказ теореме. У неким случајевима може се десити да корисник нема доказ теореме, ако је у питању једноставна теорема или теорема коју корисник не уме да докаже. Тада се неформални доказ теореме састоји само од две формуле од којих је прва премиса теореме, а друга закључак теореме. Логичке формуле које припадају неформалном доказу теореме зваћемо кораци доказа.

Као што је већ речено, докази теорема у уџбеницима су често недовољно прецизни и записивање формалних доказа тих теорема може бити временски веома захтевно[2]. Систем ArgoGeoChecker је креиран тако да, на основу неформалног доказа теореме, аутоматски генерише документ који се може формално проверити уз помоћ интерактивних доказивача теорема. Може се рећи да је систем ArgoGeoChecker креиран за проверу неформалног доказа теореме.

Проверавање неформалног доказа се изводи тако што се проверава сваки корак доказа засебно. Систем проверава да ли се текући корак неформалног доказа може извести из претходних корака и скупа аксиома које је корисник навео приликом покретања програма. Ако систем успе да докаже текући корак, генерисаће његов формални доказ. Формални докази појединачних корака ће бити обједињени у једном документу који ће представљати формални аргумент исправности доказа. Систем генерише два формална документа који се могу верификовати доказивачима за интерактивно доказивање теорема Isabelle и Coq, као и два документа у читљивој форми записана на српском и енглеском језику.

Систем ће бити коришћен за проверу неформалног доказа теореме ако он постоји. У случају да не постоји доказ теореме, односно да неформални доказ садржи само премисе и закључак теореме, систем ће бити коришћен за проналажење доказа теореме.

4 УПОТРЕБА СИСТЕМА ARGOGEOCHECKER

У уџбеницима за математику, решења задатака често нису потпуна већ само садрже упутство за решавање. У таквим случајевима систем ArgoGeoChecker може бити веома користан. У наставку текста приказан је један задатак и његово решење из уџбеника за први разред средње школе [13], његов неформални доказ (записан на природном језику и у синтакси система) и излаз који се добија покретањем система ArgoGeoChecker.

У овом раду разматране су само теореме и задаци који се доказују уз помоћ аксиома припадања и аксиома распореда тако да је скуп дефиниција и аксиома које се користе унапред одређен и унет у систем[3]. Приликом провере доказа теореме корисник бира да ли жели да користи само аксиоме припадања (опција I) или аксиоме припадања заједно са аксиомама распореда (опција II).

2 Однос између дужине формалног и неформалног доказа теореме се често назива *de Bruijn* фактор [3]. Тај однос варира за различите типове теорема и система који се користе али најчешће има вредност 4.

3 Скуп аксиома није чврсто уграђен у сам систем већ се уноси кроз засебне датотеке.

Задатак. Дата је права *p* и ван ње тачка *A*. Доказати да све праве које садрже тачку *A* и секу праву *p* припадају једној равни.

Решење.

Упутство: Доказати да све ове праве припадају равни одређеној тачком *A* и правом *p*.

На основу упутства датог у решењу може се записати следећи неформални доказ:

Дата је права *p* и ван ње тачка *A* и права *q* која садржи тачку *A* и сече праву *p*.

Постоји раван *R* одређена тачком *A* и правом *p*.

Права *q* припада равни *R*.

Када се овај доказ запише у синтакси која је подржана системом добија се наредни низ формула (синтакса је детаљно објашњена у наредном поглављу). Овај низ формула представља улаз у систем.

```
! [P,A,Q] : (line(P) & point(A) & ninc_po_l(A,P) &
             line(Q) & inc_po_l(A,Q) & int_l_l(P,Q))
? [R] : (plane(R) & inc_po_pl(A,R) & inc_l_pl(P,R))
(inc_l_pl(Q,R))
```

Позивом система са опцијом I наглашавамо да се задатак решава применом аксиома припадања:

./ArgoGeoChecker I th_11_proof.txt

На сликама 2, 3 и 4 су приказани аутоматски генерисани документи (аргументи исправности доказа) записани на српском језику, и на језицима интерактивних доказивача теорема Isabelle и Coq.

Teorema 1 (th_11_01.) *Pod pretpostavkom da važi $A \notin p$ i $A \in q$ i prave p i q se seku pokazati da postoji ravan α tako da važi $A \in \alpha$ i $p \in \alpha$.*

Dokaz:

1. Postoje tačka B i tačka C tako da važi $B \neq C$ i $B \in p$ i $C \in p$ (aksioma $I3a$).
2. Na osnovu činjenica $B \neq C$ i $B \in p$ i $C \in p$ i $A \notin p$ važi $\neg col(B,C,A)$ (aksioma $D1a$).
3. Na osnovu činjenice $\neg col(B,C,A)$ važi $\neg col(C,A,B)$ (aksioma sym_ncol1).
4. Na osnovu činjenice $\neg col(C,A,B)$ važi $\neg col(A,B,C)$ (aksioma sym_ncol1).
5. Na osnovu činjenice $\neg col(A,B,C)$ postoji ravan α tako da važi $A \in \alpha$ i $B \in \alpha$ i $C \in \alpha$ (aksioma $I4a$).
6. Na osnovu činjenica $B \neq C$ i $B \in p$ i $C \in p$ i $B \in \alpha$ i $C \in \alpha$ važi $p \in \alpha$ (aksioma $I6$).
7. Zaključak teoreme sledi iz činjenica $A \in \alpha$ i $p \in \alpha$.

QED

Teorema 2 (th_11_02.) *Pod pretpostavkom da važi $A \notin p$ i $A \in q$ i prave p i q se seku i $A \in \alpha$ i $p \in \alpha$ pokazati da važi $q \in \alpha$.*

Dokaz:

1. Na osnovu činjenice prave p i q se seku postoji tačka B tako da važi $p \neq q$ i $B \in p$ i $B \in q$ (aksioma $D6$).
2. Na osnovu činjenica $p \in \alpha$ i $B \in p$ važi $B \in \alpha$ (aksioma $D11$).
3. Važi $A = B$ ili $A \neq B$.
   4. Pretpostavimo da važi: $A = B$.
      5. Na osnovu činjenica $B \in p$ i $A = B$ važi $A \in p$.
      6. Na osnovu činjenica $A \notin p$ i $A \in p$ dobijamo kontradikciju.
   7. Pretpostavimo da važi: $A \neq B$.
      8. Na osnovu činjenica $A \neq B$ i $A \in q$ i $B \in q$ i $A \in \alpha$ i $B \in \alpha$ važi $q \in \alpha$ (aksioma $I6$).
      9. Zaključak teoreme sledi iz činjenice $q \in \alpha$.
10. Teorema je dokazana u svim slučajevima.

QED

Слика 2: *Аргумент исправности доказа записан на српском језику.*

```
lemma th_11_01 :  assumes  "¬ inc_po_l A p" and "inc_po_l A q" and "int_l_l p q"
shows "∃(alpha::plane).(inc_po_pl A alpha ∧ inc_l_pl p alpha)"

proof -
obtain B::point and C::point
where "B ~= C" and "inc_po_l B p" and "inc_po_l C p" using ax_I3a [of "p"] by auto
from ‘B ~= C‘ and ‘inc_po_l B p‘ and ‘inc_po_l C p‘ and ‘¬ inc_po_l A p‘
have "¬ col B C A" by (rule ax_D1a)
from ‘¬ col B C A‘ have "¬ col C A B" by (rule ax_sym_ncol1)
from ‘¬ col C A B‘ have "¬ col A B C" by (rule ax_sym_ncol1)
from ‘¬ col A B C‘ obtain alpha::plane
where "inc_po_pl A alpha" and "inc_po_pl B alpha" and "inc_po_pl C alpha"
using ax_I4a [of "A" "B" "C"] by auto
from ‘B ~= C‘ and ‘inc_po_l B p‘ and ‘inc_po_l C p‘ and ‘inc_po_pl B alpha‘ and
‘inc_po_pl C alpha‘ have "inc_l_pl p alpha" by (rule ax_I6)
from ‘inc_po_pl A alpha‘ and ‘inc_l_pl p alpha‘ have ?thesis by auto
from this show ?thesis .
qed

lemma th_11_02 :  assumes  "¬ inc_po_l A p" and "inc_po_l A q" and "int_l_l p q" and
"inc_po_pl A alpha" and "inc_l_pl p alpha"  shows "(inc_l_pl q alpha)"

proof -
from ‘int_l_l p q‘ obtain B::point
where "p ~= q" and "inc_po_l B p" and "inc_po_l B q" using ax_D6 [of "p" "q"] by auto
from ‘inc_l_pl p alpha‘ and ‘inc_po_l B p‘ have "inc_po_pl B alpha" by (rule ax_D11)
have "A = B ∨ A ~= B" by (subst disj_commute, rule excluded_middle)
   show ?thesis
   proof(cases "A = B")
     case True
      from ‘inc_po_l B p‘ and ‘A = B‘ have "inc_po_l A p" by simp
      from ‘¬ inc_po_l A p‘ and ‘inc_po_l A p‘ have "False" by (rule notE)
      from this show ?thesis by (rule FalseE)
   next
     case False
      from ‘A ~= B‘ and ‘inc_po_l A q‘ and ‘inc_po_l B q‘ and ‘inc_po_pl A alpha‘ and
‘inc_po_pl B alpha‘ have "inc_l_pl q alpha" by (rule ax_I6)
      from ‘inc_l_pl q alpha‘ show ?thesis by assumption
   qed
qed
end
```

Слика 3: *Формални аргумент исправности доказа записан у језику програма за интерактивно доказивање Isabelle.*

```
Theorem th_11_01 : ∀ (A:point) (p:line) (q:line), (not_inc_po_l A p ∧ inc_po_l A q ∧ int_l_l p q)
→ ∃ alpha:plane, (inc_po_pl A alpha ∧ inc_l_pl p alpha).
Proof.
 intros.
 let Hnew := fresh in assert ( Hnew : ∃ B, ∃ C, (B ≠ C ∧ inc_po_l B p ∧ inc_po_l C p)) by
applying (ax_I3a p ) ;destruct Hnew as [B[C] Hnew];decompAnd Hnew.
 assert (not_col B C A) by applying (ax_D1a B C p A ) .
 assert (not_col C A B) by applying (ax_sym_ncol1 B C A ) .
 assert (not_col A B C) by applying (ax_sym_ncol1 C A B ) .
 let Hnew := fresh in assert ( Hnew : ∃ alpha, (inc_po_pl A alpha ∧ inc_po_pl B alpha ∧
inc_po_pl C alpha)) by applying (ax_I4a A B C ) ;destruct Hnew as [alpha Hnew];decompAnd
Hnew.
 assert (inc_l_pl p alpha) by applying (ax_I6 B C p alpha ) .
 conclude.
Qed.
Theorem th_11_02 : ∀ (A:point) (p:line) (q:line) (alpha:plane), (not_inc_po_l A p ∧ inc_po_l A q
∧ int_l_l p q ∧ inc_po_pl A alpha ∧ inc_l_pl p alpha) → inc_l_pl q alpha.
Proof.
 intros.
 let Hnew := fresh in assert ( Hnew : ∃ B, (p ≠ q ∧ inc_po_l B p ∧ inc_po_l B q)) by applying
(ax_D6 p q ) ;destruct Hnew as [B Hnew];decompAnd Hnew.
 assert (inc_po_pl B alpha) by applying (ax_D11 p alpha B ) .
 assert (A = B ∨ A ≠ B) by applying (ax_g1 A B ) .
    by cases on (A = B ∨ A ≠ B).
    - {
        assert (inc_po_l A p) by substitution.
        assert (False) by applying (ax_false_inc_po_l A p ) .
        contradict.
      }
    - {
        assert (inc_l_pl q alpha) by applying (ax_I6 A B q alpha ) .
        conclude.
      }
Qed.
```

Слика 4: *Формални аргумент исправности доказа записан у језику програма за интерактивно доказивање Coq.*

5 ИМПЛЕМЕНТАЦИЈА СИСТЕМА ARGOGEOCHECKER

Приликом провере неформалног доказа теореме, систем ArgoGeoChecker проверава сваки корак доказа као засебно тврђење. Да би се ово урадило, за сваки корак сем првог (први корак доказа су увек премисе теореме и оне се не доказују) се аутоматски креирају помоћне леме у чијем закључку се налази текући корак а премиса је конјункција свих до тада изведених корака. Леме ће бити креиране у улазном формату који подржавају аутоматски доказивачи теорема који се користе у овом раду (TPTP формат). Након тога се креирају датотеке које се користе за доказивање помоћних лема. Датотеке ће садржати скуп дефиниција и аксиома (које се користе у доказу теореме чији се доказ проверава) и лему која се доказује.

У овом раду се разматрају само теореме средњошколске геометрије тако да је скуп дефиниција и аксиома унапред одређен, записан у TPTP формату и јавно је доступан [39]. У општем случају систем има могућност коришћења и других аксиоматских система, под претпоставком да су аксиоме формулисане у оквиру кохерентне логике.

Систем користи два резолуцијска доказивача (Vampire и E) и доказивач за кохерентну логику ArgoCLP. Када се користе самостално, резолуцијски доказивачи дају само информацију о томе да ли је дато тврђење теорема (тј. да ли се може доказати из аксиома примењујући дати скуп правила извођења) и враћају скуп аксиома који се користи у доказу.

Датотеке креиране за доказивање помоћних лема прво се прослеђују и једном и другом резолуцијском доказивачу. Скуп са мањим бројем аксиома, које су нашли резолуцијски доказивачи, прослеђује се кохерентном доказивачу. Кохерентни доказивач се затим користи за генерисање формалног доказа леме на основу смањеног скупа аксиома (ако је макар један од резолуцијских доказивача доказао лему).[4]

На крају процеса доказивања, формални докази помоћних лема биће обједињени у један документ који ће представљати формални аргумент исправности доказа. Тај документ може бити записан у језицима програма за интерактивно доказивање (Isabelle и Coq) и на природном језику (српском и енглеском). Архитектура система је приказана на слици 5.

Може се десити да, у оквиру задатог временског ограничења, систем није у стању да докаже неку од помоћних лема. У том случају формални аргумент исправности доказа ће опет бити генерисан али ће недостајати докази тих лема. Корисник тада може допунити документ који је систем генерисао или формулисати нов доказ теореме, дати детаљнију скицу доказа и покушати доказивање поново.

Синтакса неформалног доказа

Да би се записао неформални доказ потребно је записати његове кораке, односно логичке формуле које се у њему налазе. Релације које се јављају у формулама су релације припадности (тачка припада правој, тачка припада равни, права припада равни), релације пресека (две праве се секу, две равни се секу, права и раван се секу) колинеарност тачака, компланарност тачака и релација између. Ове релације ће бити представљене предикатима који су дати у табели 1. Предикати који означавају негације релација се добијају додавањем слова n испред имена предиката (тачка *A* не припада прави *P* се записује ninc_po_l(A,P)).

Тачка *A* припада прави *P*	inc_po_l(A,P)
Тачка *A* припада равни *R*	inc_po_pl(A,R)
Права *P* припада равни *R*	inc_l_pl(P,R)
Праве *P1* и *P2* се секу	int_l_l(P1,P2)
Равни *R1* и *R2* се секу	int_pl_pl(R1,R2)
Права *P* и раван *R* се секу	int_l_pl(P,R)
Тачке *A*, *B* и *C* су колинеарне	col(A,B,C)
Тачке *A*, *B*, *C* и *D* су компланарне	cop(A,B,C,D)
Тачка *B* се налази између тачака *A* и *C*	bet(A,B,C)

Табела 1: *Предикати и њихов запис у неформалном доказу.*

За разлику од доказа у уџбеницима - где се мала слова латинице користе за запис правих, велика слова за запис тачака и грчка слова за запис равни, у овом формату се све променљиве записују великим словима и уводе се додатни предикати *point*, *line* и *plane* за запис и дефинисање типова тих променљивих[5]. Конјункција се у формулама записује симболом &, дисјункција симболом |, универзални квантификатор симболом !, егзистенцијални квантификатор симболом ? и имена променљивих које се уводе у одређеном кораку се записују између угластих заграда [и]. Након навођења променљивих се користи симбол : (двотачка). Формуле су записане између обичних заграда (и). Ако се корак доказа односи на премисе теореме, он ће бити записан са универзалним квантификатором. Ако се кораком доказа уводе нови објекти, он ће бити записан са егзистенцијалним квантификатором. А ако се кораком доказа изводи неко својство раније уведених објеката, тај корак неће бити квантификован.

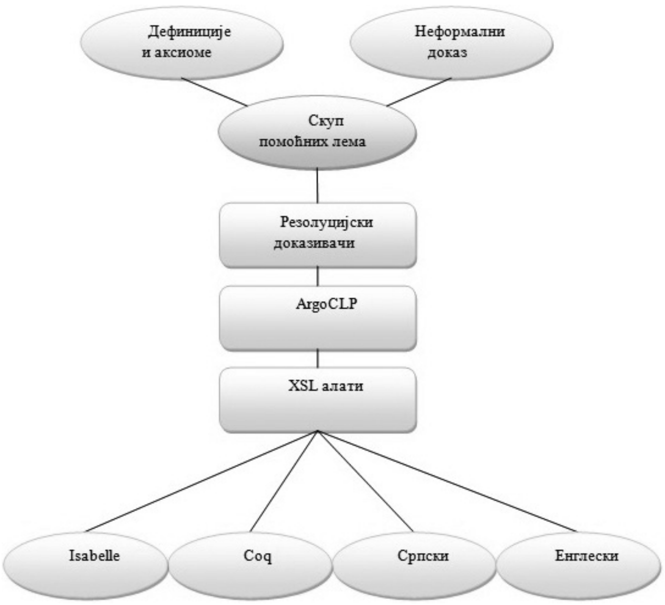
Неки од примера дозвољених корака и њихов запис у оквиру система ArgoGeoChecker се налазе у табели 2.

Тачка *A* не припада правој *P*.	! [P,A]:(line(P) & point(A) & ninc_po_l(A,P))
Тачке *A, B, C* су неколинеарне.	(ncol(A,B,C))
Постоји раван *R* која садржи тачку *A* и праву *P*.	?[R]:(plane(R) & inc_po_pl(A,R) & inc_l_pl(P,R))

Табела 2: *Примери дозвољених корака неформалног доказа.*

4 Не постоји гаранција да ће кохерентни доказивач увек успети да докаже теорему када му је прослеђен скуп аксиома које су користили резолуцијски доказивачи.

5 Логика првог реда нема подршку за типове па је увођење ових предиката неопходно.



Слика 5: *Архитектура система ArgoGeoChecker.*

6 ПРИМЕНА СИСТЕМА ARGOGEOCHECKER

Рад система ArgoGeoChecker је анализиран над осамнаест теорема и задатака из уџбеника за први разред средње школе [13,24,41]. Знатан део аксиома и теорема које се обрађују у оквиру геометрије у средњој школи припада кохерентној логици. Отуда се у великом броју случајева теореме и њихови докази могу записати у оквиру овог система без икаквих измена. У неким случајевима су докази чак у потпуности изостављени па се систему предаје сама теорема.

Неке теореме садрже тврђења типа „постоји само једна тачка“ што је израз који се не може записати у логици првог реда уз помоћ само једне формуле. Али, као што је познато, докази таквих теорема се увек изводе из два дела. Прво се покаже да одређени објекат постоји, а затим да не могу постојати два таква објекта. Управо оваквом поделом теореме и њеног доказа добијамо тврђења која припадају кохерентној логици и која се могу записати у оквиру овог система.

Теореме над којима је систем ArgoGeoChecker примењен дате су у наредној листи. Неформални докази теорема и аутоматски генерисани формални аргументи исправности доказа, као и документи записани на српском и енглеском језику су јавно доступни [39].

1. Ако тачка *A* не припада правој *p*, тада постоји раван која садржи тачку *A* и праву *p*.[6]
2. Ако тачка *A* не припада правој *p*, и ако постоје две равни које садрже тачку *A* и праву *p*, тада су те две равни идентичне.
3. Ако две разне равни имају заједничку тачку тада је њихов пресек права.
4. Ако две разне равни имају заједничку праву, тада немају заједничких тачака ван те праве.[7]
5. Ако су *A, B, C* три неколинеарне тачке, доказати да су сваке две од те три тачке различите.
6. Ако су *p, q, r* три разне праве од којих се сваке две праве секу, али не постоји тачка која припада свима трима правама, доказати да су *p, q, r* компланарне.
7. За сваке две разне тачке *A* и *B* постоји тачка *C* таква да важи распоред *A-C-B*.
8. Ако су *P, Q, R* унутрашње тачке ивица *BC, AC, AB* неког троугла, доказати да су оне неколинеарне**.**
9. Нека су *a* и *b* две различите праве. Доказати да оне имају највише једну заједничку тачку.
10. Дате су раван α и тачка *A* ван равни. Доказати да права *p*, која садржи тачку *A*, не може имати са равни α више од једне заједничке тачке.
11. Дата је права *p* и ван ње тачка *A*. Доказати да све праве које садрже тачку *A* и секу праву *p* припадају једној равни.
12. Дате су различите праве *a* и *b* и тачка *M* ван њих. Ако постоје две различите праве *m* и *n*, које садрже тачку *M* и секу обе праве *a* и *b*, тада праве *a* и *b* припадају једној равни. Доказати.
13. Дат је скуп тачака *A, B, C, D* где тачка *C* припада правој *AB,* а тачка *D* не припада правој *AB*. Доказати да се равни *ABD* и *ACD* поклапају.
14. Ван равни α дате су три неколинеарне тачке *A, B, C* такве да се права *AB* и раван α секу у тачки *M*, права *BC* и раван α секу у тачки *N* и права *AC* и раван α секу у тачки *P*. Доказати да су тачке *M*, *N* и *P* колинеарне.
15. Ако важи распоред *A-B-C* и *A-D-C*, онда су *A, B, C, D* тачке једне праве и не важи распоред *B-A-D*.
16. Ако су *O, A, B, C* тачке једне праве и ако важи распоред *A-O-B* и *A-O-C* онда не важи распоред *B-O-C*.
17. Ако важи распоред *A-O-B* и *B-O-C* и *C-O-D*, онда су тачке *O, A, B, C* и *D* тачке једне праве и важи распоред *A-O-D*.
18. Ако важи распоред *A-B-C* и *A-C-D*, онда су *A, B, C* и *D* тачке једне праве и важи распоред *A-B-D* и *B-C-D*.

У табели 3 су приказани резултати добијени приликом тестирања овог система. За сваку теорему је наведен број корака који се појављује у доказу теореме, затим број корака које су доказали резолуцијски доказивачи, као и број корака које је доказао кохерентни доказивач ArgoCLP. ArgoGeoChecker је тестиран на систему 48 AMD Opteron(tm) Processor 6168. У току проверавања неформалног доказа теореме сваки корак доказа се проверава независно од осталих. Временско ограничење од 60 секунди по кораку је коришћено у свим програмима за доказивање теорема.

6 Теореме 1 и 2 су настале приликом записивања теореме у кохерентну логику. Оригинална теорема гласи: (I1) Ако тачка *A* не припада правој *p*, тада постоји јединствена раван која садржи тачку *A* и праву *p*.

7 Теореме 3 и 4 су настале приликом записивања доказа теореме I2. У доказу се уочава детаљ који није јасно изражен у самој формулацији теореме, а то је да те две равни немају заједничких тачака ван те праве.

Редни број теореме	Број корака у доказу	Доказаних Vampire, E	Доказаних ArgoCLP	Време (у секундама)
1	4	4	4	9
2	4	4	4	8
3	3	3	3	8
4	4	4	4	82
5	1	1	1	16
6	2	2	2	8
7	16	13	13	136
8	1	0	0	127
9	1	1	1	7
10	1	1	1	8
11	2	2	2	73
12	6	6	5	173
13	4	3	2	128
14	8	7	6	145
15	2	1	1	128
16	1	0	0	127
17	2	1	1	134
18	3	1	1	128

Табела 3: *Резултати примене система ArgoGeoChecker.*

Систем ArgoGeoChecker је генерисао комплетне формалне аргументе исправности доказа за 9 од 18 теорема. Када се посматра укупан број корака у свим теоремама, систем је успешно доказао 51 од укупно 65 корака, односно 78% корака. С обзиром да овај систем проверава појединачне кораке доказа, тиме је дата успешност разматраног приступа. Просечно време провере доказа теореме и генерисања формалних аргумената исправности доказа у језицима интерактивних доказивача Isabelle и Coq, као и генерисања читљивих доказа на српском и енглеском језику је 80 секунди.

Одређене теореме су наведене без доказа (у табели су означене са једним кораком - само закључак теореме). У тим случајевима ArgoGeoChecker је коришћен за налажење доказа теореме. Од пет теорема које су дате без доказа, систем је успео да нађе потпун доказ (без усмеравања од стране корисника) за три теореме.

Сматрамо да је комплетан аргумент исправности доказа генерисан само за оне доказе код којих су сви кораци успешно доказани. У пракси би вештији корисник могао користити чак и непотпун аргумент исправности доказа, тако што би га допунио доказима недостајућих корака.

У оквиру ових експеримената извршена је провера само оригиналних доказа записаних у уџбеницима. Нису разматране различите верзије доказа једне исте теореме нити су постојећи докази проширени додатним корацима. Допуњавање неформалних доказа новим корацима би свакако побољшало успешност овог приступа.

7 РАЗВОЈ СИСТЕМА ARGOGEOCHECKER И СЛИЧНИ ПРИСТУПИ

Основу система ArgoGeoChecker представља доказивач за кохерентну логику ArgoCLP који је креирао аутор овог рада у сарадњи са Весном Маринковић, рођеном Павловић, и Предрагом Јаничићем. Рад доказивача ArgoCLP је анализиран кроз доказивање теорема геометрије са неколико различитих аксиоматских система за еуклидску геометрију [38] и приказан је начин коришћења доказивача који омогућава аутоматску трансформацију аксиоматских система [36].

Систем за аутоматско доказивање теорема, који користи резолуцијске доказиваче и кохерентни доказивач ArgoCLP, је осмислио и развио аутор овог рада. Формат за запис доказа кохерентне логике [37], чији је опис дат у XML-у заједно са скупом XSL алата који омогућавају превођење доказа у различите (формалне и природне) језике, је креирао аутор овог рада у сарадњи са Марк Беземом (Marc Bezem), Жилиен Нарбуом (Julien Narboux) и Предрагом Јаничићем. Комбинација ових алата коришћена је за доказивање скупа теорема из књиге о заснивању геометрије *"Metamathematische Methoden in der Geometrie"* [34] чији су аутори Волфрам Швабхојзер (Wolfram Schwabhauser), Ванда Шмилев (Wanda Szmielev) и Алфред Тарски (Alfred Tarski). Аутоматски је доказано 37% теорема из првих 12 поглавља књиге које припадају геометрији равни без икаквог навођења система [40]. Том приликом није рађена провера појединачних доказа приказаних у књизи, већ је систему предат само списак свих аксиома, дефиниција и теорема из књиге, редоследом којим се појављују у књизи.

Систем за проверавање неформалних доказа теорема ArgoGeoChecker је осмислио аутор овог рада и први пут га применио на доказе теорема средњошколске геометрије и представио у овом раду. Систем комбинује претходне приступе и користи их на нов и оригиналан начин. Ово је први систем, колико је аутору познато, који генерише доказе на српском језику који у потпуности одговарају формалним доказима који се могу проверити уз помоћ рачунара.

Слични приступи

Пре неколико година развијен је нов аксиоматски систем E намењен доказивању постулата Еуклидових Елемената [2]. Систем су развили Џереми Авигад (Jeremy Avigad), Едвард Дин (Edward Dean) и Џон Мума (John Mumma). Основна идеја која стоји иза овог система је аутоматско проверавање неформалних доказа налик доказима из Елемената. Бенџамин Нортроп (Benjamin Northrop) [28] је имплементирао извођења описана тим системом [1] и креирао програм E-proof-checker [29]. Скуп аксиома система E је подељен на правила за конструисање (construction rules) и правила извођења (inference rules). Корисник на улазу задаје објекте који се користе у доказу користећи правила за конструисање објеката. Објекти се задају навођењем њихових својстава (тачка припада правој, тачка је пресечна тачка двеју правих, итд.). Након тога, примењујући правила извођења, аутоматски доказивач покушава да докаже коначно својство (описано тврђењем теореме) над до тада креираним објектима. За разлику од система описаног у овом раду, E-proof-checker нема могућност рада са другим теоријама. Креиран је искључиво за рад са Авигадовим аксиоматским системом, подржава само једну теорију и има фиксиран скуп аксиома. Генерисање формалних доказа није подржано. Колико је познато аутору, ово је једини програм који омогућава аутоматску проверу неформалних доказа теорема.

У пољу доказивања теорема тренутно се доста пажње посвећује комбиновању аутоматских и интерактивних доказивача теорема. Најуспешнији пример комбиновања је алат Sledgehammer [7,8] креиран у интерактивном доказивачу теорема Isabelle. Користи неколико резолуцијских доказивача теорема (Vampire, E и SPASS [45]) и интерни доказивач Metis који служи за креирање верификованог доказа [6]. Може се користити за доказивање одређених теорема, али и за добијање списка аксиома који се користе у доказу добијеном уз помоћ аутоматских доказивача теорема (у случају да корисник жели да креира свој доказ). Алат Sledgehammer је далеко моћнији од система описаног у овом раду али је ограничен на интерактивни доказивач теорема Isabelle и не гарантује добијање читљивих доказа. Осим тога његово коришћење захтева познавање језика интерактивног доказивача Isabelle.

8 ЗАКЉУЧЦИ И ДАЉИ РАД

Систем ArgoGeoChecker представљен у овом раду је интуитиван, једноставан за коришћење, аутоматски проверавач доказа. Систем на улазу очекује неформални доказ тореме (задат као низ помоћних тврђења која се користе у доказу) и генерише формалне аргументе исправности доказа у језицима програма за интерактивно доказивање теорема Isabelle и Coq и читљиве аргументе исправности доказа на природном језику (српском и енглеском). Осим документа записаног на природном језику, чак је и формални аргумент исправности доказа теореме записан у читљивом облику тако да га корисник може разумети и ако никада није користио програме за интерактивно доказивање теорема.

Приказана је његова примена на доказима и решењима десетак теорема и задатака из средњошколских уџбеника. Систем је успео да потврди исправност доказа за 9 од 18 теорема и успешно је доказао 51 од укупно 65 корака, односно 78% корака. У овим експериментима анализирани су постојећи докази који се налазе у уџбеницима, без икаквих измена или додавања нових корака, што показује да систем и у свом основном облику има потенцијала за практичну употребу.

У неким случајевима, кораци који нису доказани у себи крију примене претходно доказаних теорема. У оквиру овог експеримента, приликом провере доказа коришћене су само аксиоме, није подржано коришћење претходно доказаних теорема. Проширење система тако да подржава и ову опцију је могуће, чиме би се још боље илустровала реална примена овог система у образовању. Ово проширење је један од будућих праваца рада.

За разлику од доказивања уз помоћ програма за интерактивно доказивање теорема, систем ArgoGeoChecker не захтева никакво познавање специфичних језика за интерактивно доказивање (нити тактика које се користе у њима, без којих је интерактивно доказивање практично немогуће), што га чини погодним за коришћење међу почетницима у формалном доказивању теорема. Ипак, систем захтева формулацију геометријских тврђења у прецизној синтакси што је свакако савладиво за мотивисане ученике средњих школа.

Иако има одређена ограничења (која се тичу ефикасности и изражајности), систем описан у овом раду је далеко једноставнији за употребу у поређењу са другим моћнијим и изражајнијим системима и представља добру основу за приближавање аутоматског и формалног доказивања студентима и ђацима.

Како постоји аутоматска процедура превођења произвољне формуле логике првог реда у кохерентну форму [30], могуће је проширење овог система за рад са логиком првог реда. Аутоматско превођење формула у кохерентну логику може у неким случајевима генерисати већи број кохерентних формула и увести нове предикате што може утицати на читљивост добијених доказа. Проширење система и увођење подршке за рад са формулама логике првог реда је планирано као наставак овог рада.

Приликом доказивања теорема и креирања неформалног доказа, корисник често има информације о самим аксиомама које се користе у доказу. Тренутно систем не подржава коришћење тих информација али је у плану увођење подршке за навођење система тако да форсира примену аксиоме која се наводи у тексту. Осим тога неформални доказ који корисник задаје могао би се записати чак и на природном језику ограниченог речника. Оваква проширења система су планирана у будућности.

Такође у плану је тестирање овог система над другим аксиоматским системима као што је Авигадов аксиоматски систем и примена над постулатима Еуклидових Елемената.

Захвалност: Аутор се посебно захваљује Предрагу Јаничићу и Филипу Марићу на коментарима приликом дискусија на тему овог рада.

ЛИТЕРАТУРА

[1] Jeremy Avigad, The Role of the Diagram in Euclid's Elements, online at: http://www.phil.cmu.edu/~avigad/formal/paris2_merged.pdf

[2] Jeremy Avigad, Edward Dean, and John Mumma. A Formal System for Euclid's Elements. *The Review of Symbolic Logic*, 2009.

[3] Henk Barendregt and Freek Wiedijk. The Challenge of Computer Mathematics. *Philosophical Transactions of the Royal Society*, 363(1835):2351-2375, 2005.

[4] Marc Bezem and Thierry Coquand. Automating Coherent Logic. In Geoff Sutcliffe and Andrei Voronkov, editors, *12th International Conference on Logic for Programming, Artificial Intelligence, and Reasoning - LPAR 2005*, volume 3835 of *LNCS*. Springer-Verlag, 2005.

[5] Marc Bezem and Dimitri Hendriks. On the Mechanization of the Proof of Hessenberg's Theorem in Coherent Logic. *Journal of Automated Reasoning*, 40(1), 2008.

[6] Jasmin Christian Blanchette. Redirecting Proofs by Contradiction. In Jasmin Christian Blanchette and Josef Urban, editors,*Third International Workshop on Proof Exchange for Theorem Proving, PxTP 2013,*volume 14 of *EPiC Series*. EasyChair, 2013.

[7] Jasmin Christian Blanchette, Sascha Bohme, and Lawrence Paulson. Extending Sledgehammer with SMT Solvers. *Journal of Automated Reasoning*, 51(1):109-128, 2013.

[8] Jasmin Christian Blanchette, Lukas Bulwahn, and Tobias Nipkow. Automatic Proof and Disproof in Isabelle/HOL. In Cesare Tinelli and Viorica Sofronie-Stokkermans, editors, *Frontiers of Combining Systems, 8th International Symposium, Proceedings,* volume 6989 of *LNCS*, pages 12-27. Springer, 2011.

[9] Gabriel Braun and Julien Narboux. From Tarski to Hilbert. In Tetsuo Ida and Jacques Fleuriot, editors, *Automated Deduction in Geometry 2012*, Edinburgh, United Kingdom, September 2012.

[10] Nicolaas Govert de Bruijn. A survey of the project Automath. In J. P. Seldin and J. R. Hindley, editors, To H. B. Curry: *Essays on Combinatory Logic, Lambda Calculus and Formalism, Academic Press,* 1980.

[11] Kaliszyk Cezary and Urban Josef. HOL(y)Hammer: Online ATP Service for HOL Light. *Mathematics in Computer Science*, 2014.

[12] Christophe Dehlinger, Jean-Francois Dufourd, and Pascal Schreck. Higher-Order Intuitionistic Formalization and Proofs in Hilbert's Elementary Geometry. In *Automated Deduction in Geometry*, volume 2061 of *LNCS*. Springer, 2001.

[13] Radivoje Despotović, Ratko Tošić, Branimir Šešelja, Matematika za I razred srednje škole, *Zavod za udžbenike i nastavna sredstva*, Beograd, 2000.

[14] John Fisher and Marc Bezem. Skolem Machines and Geometric Logic. *4th International Colloquium on Theoretical Aspects of Computing - ICTAC 2007*, volume 4711 of *LNCS*. Springer-Verlag, 2007.

[15] Mohan Ganesalingam and William Timothy Gowers. A fully automatic problem solver with human-style output. *CoRR*, abs/1309.4501, 2013.

[16] Georges Gonthier, Andrea Asperti, Jeremy Avigad, Yves Bertot, Cyril Cohen, Francois Garillot, Stephane Le Roux, Assia Mahboubi, Russell O'Connor, Sidi Ould Biha, Ioana Pasca, Laurence Rideau, Alexey Solovyev, Enrico Tassi, and Laurent Thery. A Machine-Checked Proof of the Odd Order Theorem. In Sandrine Blazy, Christine Paulin, and David Pichardie, editors*, ITP 2013, 4th Conference on Interactive Theorem Proving*, volume 7998 of *LNCS*, France, 2013. Springer.

[17] Thomas Hales. Introduction to the flyspeck project. In *Mathematics, Algorithms, Proofs,* volume 05021 of *Dagstuhl Seminar Proceedings*. Internationales Begegnungs und Forschungszentrum fur Informatik (IBFI), Schloss Dagstuhl,Germany, 2006.

[18] John Harrison. Hol light: A tutorial introduction. In Mandayam K. Srivas and Albert John Camilleri, editors, *Formal Methods in Computer-Aided Design*, volume 1166 of *LNCS*. Springer, 1996.

[19] Predrag Janičić, Matematička logika u računarstvu, Izdavač *Matematički fakultet*, Beograd 2008.

[20] Maurice Lecat. Erreurs des Mathematiciens. *Philosophical Review*, volume 46, 1937.

[21] John McCarthy. Computer programs for checking mathematical proofs. In *Recursive Function Theory, Proceedings of a Symposium in Pure Mathematics*. American Mathematical Society,1962.

[22] Laura Meikle and Jacques Fleuriot. Formalizing Hilbert's Grundlagen in Isabelle/Isar. In *Proceedings of TPHOLs*, volume 2758 of *Lecture Notes in Computer Science*. Springer, 2003.

[23] Laura Meikle and Jacques Fleuriot. Mechanical theorem proving in computation geometry. In *Proceedings of Automated Deduction in Geometry*, volume 3763 of *LNCS*. Springer-Verlag,2005.

[24] Milan Mitrović, mr Srđan Ognjanović, mr Mihailo Veljković, Ljubinka Petković, Nenad Lazarević, Geometrija za prvi razred Matematičke gimnazije, Izdavač *KRUG*, Beograd, 1998

[25] Julien Narboux. Formalisation et automatisation du raisonnement geometrique en Coq. PhD thesis, *Universite Paris Sud*, September 2006.

[26] Julien Narboux. Mechanical Theorem Proving in Tarski's Geometry. In *Proceedings of Automated Deduction in Geometry* 2006, volume 4869 of *LNCS*. Springer, 2007.

[27] Tobias Nipkow, Lawrence Paulson, and Markus Wenzel. Isabelle/HOL - A Proof Assistant for Higher-Order Logic, volume 2283 of *Lecture Notes in Computer Science*. Springer, 2002.

[28] Benjamin Northrop. Automated Diagrammatic Reasoning: a proof checker for the language of E. Master thesis, *Department of Philosophy, Carnegie Mellon University*, 2011.

[29] Benjamin Northrop. E Proof Checker - a software program which verifies Euclidean-style diagrammatic proofs written in E. on-line at: http://www.bennorthrop.com/e/e-proof-checker.php

[30] Andrew Polonsky. Proofs, Types and Lambda Calculus. PhD thesis, *University of Bergen*, 2011.

[31] Alexandre Riazanov and Andrei Voronkov. The design and implementation of Vampire. *AI Communications*, 15(2-3):91-110, 2002.

[32] John Alan Robinson. A machine oriented logic based on the resolution principle. *Journal of ACM*,1965.

[33] Stephan Schulz. E - a brainiac theorem prover. *AI Communications*, 15(2-3):111-126, 2002.

[34] Wolfram Schwabhauser, Wanda Szmielew, and Alfred Tarski. Metamathematische Methoden in der Geometrie. Springer-Verlag, Berlin, 1983.

[35] Phil Scot. Mechanising Hilbert's Foundations of Geometry in Isabelle. Master thesis, *University of Edinburgh*, 2008.

[36] Sana Stojanović. Preprocessing of the axiomatic system for more efficient automated proving and shorter proofs. In Tetsuo Ida and Jacques Fleuriot, editors, *Automated Deduction in Geometry: 9th International Workshop, ADG 2012. Revised Selected Papers*, pages 181-192. Springer, 2013.

[37] Sana Stojanović, Julien Narboux, Marc Bezem, and Predrag Janičić. A vernacular for coherent logic. In Stephen Watt et.al., editor, *Intelligent Computer Mathematics - CICM* 2014, volume 8543 of *Lecture Notes in Computer Science*. Springer, 2014.

[38] Sana Stojanović, Vesna Pavlović, and Predrag Janičić. A Coherent Logic Based Geometry Theorem Prover Capable of Producing Formal and Readable Proofs. In Pascal Schreck, Julien Narboux, and Jurgen Richter-Gebert, editors, *Automated Deduction in Geometry*, volume 6877 of *Lecture Notes in Computer Science*. Springer, 2011.

[39] Sana Stojanović Đurđević, Automated verification of informal proofs, 2016. on-line at: http://argo.matf.bg.ac.rs/downloads/formalizations/ArgoGeoChecker.zip

[40] Sana Stojanović Đurđević, Julien Narboux, and Predrag Janičić. Automated generation of machine verifiable and readable proofs: A case study of Tarski's geometry. *Annals of Mathematics and Artificial Intelligence*, 2015.

[41] Vladimir Stojanović, Matematiskop 3 - Odabrani zadaci (sa rešenjima) za učenike srednjih škola, *Naučna knjiga*, Beograd 1988.

[42] Geoff Sutcliffe. The TPTP Problem Library and Associated Infrastructure: The FOF and CNF Parts, v3.5.0. *Journal of Automated Reasoning*, 43(4):337-362, 2009.

[43] The Coq development team. The Coq proof assistant reference manual, Version 8.2. *TypiCal Project,* 2009.

[44] Andrzej Trybulec. Mizar. In Freek Wiedijk, editor, *The Seventeen Provers of the World*, volume 3600 of *Lecture Notes in Computer Science*. Springer, 2006.

[45] ChristophWeidenbach, Dilyana Dimova, Arnaud Fietzke, Rohit Kumar, Martin Suda, and Patrick Wischnewski. Spass version 3.5. In *Automated Deduction - CADE-22 Proceedings*, volume 5663 of *LNCS*. Springer, 2009.

[46] Markus Wenzel. Isar - A Generic Interpretative Approach to Readable Formal Proof Documents. In *Theorem Proving in Higher Order Logics (TPHOLs'99),* volume 1690 of *LNCS*. Springer,1999.

Sana Stojanović Đurđević, Matematički fakultet u Beogradu.
Kontakt: sana@matf.bg.ac.rs

Oblasti interesovanja: Automatsko i interaktivno dokazivanje teorema, matematička logika, geometrija