

# Sofverski sistem za učenje ekspertskih sistema

Dražen Drašković, Vuk Batanović, Boško Nikolić

**Sadržaj** - U radu je opisan softverski sistem, koji je realizovan na Elektrotehničkom fakultetu u Beogradu, za učenje predmeta Ekspertski sistemi. Softver je razvijen kao edukacioni sistem namenjen studentima i koristi se za potrebe nastave na osnovnim i master studijama. Sistem omogućava pregled primera i zadataka korak po korak po temama kojima pripadaju i koje se rade na predavanjima i vežbama. Realizovane su sledeće simulacije: strategije pretraživanja po širini, po dubini, metodom planinarenja, metodom prvo najbolji, metodom grananja i ograničavanja, metodom A\*, pretraživanja u igrama, produkcionim sistemima, opšti rešavač problema GPS, sistem za planiranje STRIPS, fuzzy logika, rezonovanje na osnovu faktora izvesnosti i drugi načini izražavanja neizvesnosti. Studenti mogu i sami da unose svoje primere i zadatke i tako simuliraju željenu situaciju. Simulaciju mogu u svakom trenutku vratiti u prethodni korak ili ići na sledeći. Na kraju simulacije student može odštampati i detaljan postupak rešavanja zadatka. Realizovani sistem predavačima omogućava mnogo efikasniji rad, a studentima brže savladavanje gradiva.

**Ključne reči** - veštačka inteligencija, ekspertski sistemi, edukacioni sistem.

## I. UVOD

Danas u svetu postoji veliki broj ekspertskih sistema razvijenih u različitim oblastima. Najviše se koriste u medicini, ali su u velikoj meri razvijeni i u elektronici, računarstvu, matematici, fizici, za upravljanje proizvodnim i drugim procesima. Primarni cilj ekspertskih sistema je izučavanje i implementacija specijalizovanog softvera čija je svrha pronalaženje rešenja ili otklanjanje određenih neizvesnosti pri rešavanju problema, za koje bi u normalnim uslovima morali biti konsultovani stručnjaci tzv. eksperti za određenu oblast.

Nastava iz Ekspertskih sistema na Elektrotehničkom fakultetu u Beogradu počela je da se izvodi od 1987.godine. Sa razvojem savremenih informacionih tehnologija pojavila se potreba za upotrebom softverskih alata koji bi olakšali prenos znanja i pomogli studentima da na efikasan i jednostavan način usvajaju gradivo koje im je predstavljeno.

Tema ovog rada je softverski sistem za ekspertске sisteme, koji vizuelno ilustruje niz tehnika i algoritama korišćenih na predavanjima i računskim vežbama, i prikazuje detaljna rešenja odabranih problema iz zbirke zadataka. Takođe, korisnici sistema mogu i sami da definišu probleme i dobiju rešenje po koracima.

U nastavku rada, dati su razlozi za uvođenje ovog

softverskog sistema, opisan je sam sistem i korišćenje sistema kroz jednu laboratorijsku vežbu baziranu na sistemu. Na kraju rada dat je zaključak.

## II. OPIS PROBLEMA

Prema preporukama najpoznatijih svetskih strukovnih organizacija IEEE Computer Society i ACM, iz 2005.god. na listi od 40 odabranih tema koje studenti treba da obrade na svojim osnovnim i master studijama su ekspertski sistemi i veštačka inteligencija.[1] U okviru programa, koji se bave kompjuterskim naukama i softverskim inženjerstvom, preporučeno je da se teme iz ekspertskih sistema i veštačke inteligencije obrađuju na minimalno 1 ili 2 kursa, a maksimalno na 4 ili 5 kurseva. Preporuke se primenjuju i na Elektrotehničkom fakultetu u Beogradu, gde su ekspertski sistemi obavezan predmet za studente četvrtе godine osnovnih studija na Odseku za računarsku tehniku i informatiku, izborni predmet na osnovnim studijama na Odseku za softversko inženjerstvo i izborni predmet prve godine master studija na modulima Računarska tehnika i informatika i Softversko inženjerstvo.

Polje veštačke inteligencije bavi se dizajnom i analizom autonomnih sistema. Veštačka inteligencija pruža skup alata za rešavanje problema, koji se mnogo teže i manje praktično rešavaju drugim metodama. Izučavanje ove oblasti uključuje heurističke algoritme za pretragu i planiranje, formalizme za reprezentaciju znanja i rezonovanja, tehnike mašinskog učenja, metode koje se primenjuju na probleme kao što su razumevanje govora i jezika, kompjuterski vid i robotika. Student treba da bude u stanju da odredi kada je pristup veštačke inteligencije pogodan za dati problem i da bi mogao da izabere i primeni odgovarajući metod veštačke inteligencije. ACM je podelio sve teme iz oblasti ekspertskih sistema i veštačke inteligencije na one koje treba obavezno uključiti u kurs i one koje su izborne, a svaka tema je izdvojena na podteme. Tri obavezne teme su fundamentalna pitanja iz oblasti veštačke inteligencije, osnovne strategije pretraživanja i znanje zasnovano na rezonovanju. U obavezne podteme koje treba obraditi na kursu spadaju primeri inteligentnih sistema, uloga heuristike i potreba za kompromisom između optimalnosti i efikasnosti, problem prostora, rešavanje problema strategijama pretraživanja, strategije *brute-force* - po širini, po dubini, po dubini sa iterativnim produblivanjem i *best-first* - generički prvo-najbolji, Dijkstra algoritam, A\*, igre sa dva igrača - minimax algoritam i minimax algoritam sa alfa-beta odsecanjem, tehnika rezolucije u dokazivanju teorema, zaključivanja direktnim, povratnim i cikličkim hibridnim ulančavanjem i Bayes-ova teorema. [2] U podteme koje se opciono izučavaju u sličnim kursovima o veštačkoj inteligenciji i ekspertskim sistemima, spadaju tehnike

Dražen Drašković, Elektrotehnički fakultet Univerziteta u Beogradu, Bul. kralja Aleksandra 73, Beograd, Srbija (tel: 381-64-1517150; e-mail: [draskovic@etf.rs](mailto:draskovic@etf.rs)).

Vuk Batanović, Elektrotehnički fakultet Univerziteta u Beogradu, Bul. kralja Aleksandra 73, Beograd, Srbija (e-mail: [vukbatanovic@sbb.rs](mailto:vukbatanovic@sbb.rs)).

Boško Nikolić, Elektrotehnički fakultet Univerziteta u Beogradu, Bul. kralja Aleksandra 73, Beograd, Srbija (tel: 381-64-1206912; e-mail: [nbosko@etf.rs](mailto:nbosko@etf.rs)).

napredne pretrage - lokalne pretrage i optimizacije, pretraživanje metodom planinarenja, genetski algoritmi, tehnike za smanjivanje kompleksnosti pretrage, tehnike naprednog rezonovanja, semantičke mreže, agenti i tipovi agenata, obrada prirodnog jezika, mašinsko učenje, planiranje sistema, robotika i percepcija.

Nabrojane teme obuhvataju algoritme koji u nekim slučajevima studentima mogu biti veoma apstraktni i nejasni. Jedna vrsta pomoći studentima može biti realizacija vizuelnog simulatora. Zadatak pri realizaciji takvog sistema bio bi zajedničko grafičko okruženje za sve simulacije algoritama, koje se po prirodi dosta razlikuju: algoritmi pretraživanja, modeli predstavljanja znanja, strategije rešavanja problema i rad ekspertskeg sistema u neizvesnom okruženju. Jedinstveno okruženje treba da podržava sledeće zahteve: unos primera za koje treba simulirati traženi algoritam, provera ispravnosti unetih primera, simuliranje odgovarajućih algoritama, omogućavanje kretanja po koracima kroz simulaciju, kao i skokovi na početak i kraj simulacije, zatim da uz svaki korak bude izloženo objašnjenje i da se vide razlike u odnosu na prethodno stanje u sistemu. Na taj način bi korisniku u svakom koraku bio jasan algoritam. Takođe, okruženje treba da omogući čuvanje simulacije i njeno ponovno učitavanje.

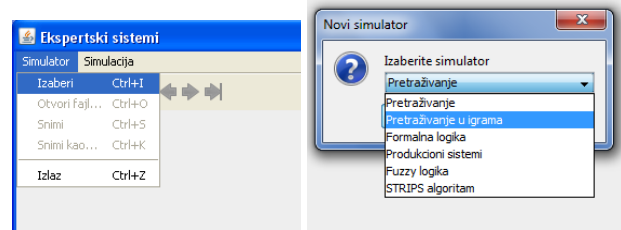
Ovakvih sistema iz oblasti veštačke inteligencije i ekspertskeg sistema, koji omogućavaju studentu da prvo definiše problem, a zatim da dobije detaljno objašnjenje, postoji vrlo malo. To su obično aplikacije koje simuliraju samo jedan algoritam ili sistemi koji simuliraju sport, ratne igre ili igre na sreću. Na Univerzitetu Delft u Holandiji, razvijen je fudbalski simulator, zasnovan na veoma pojednostavljenom modelu fudbala, koji se koristi za istraživanja o veštačkoj inteligenciji i u edukacione svrhe na uvodnom kursu o veštačkoj inteligenciji, kao i za razvoj i korišćenje prilikom davanja problemskih zadataka studentima [3]. Sličan simulator RobotSoccer, koji se takođe koristi u nastavi, implementirali su i studenti na Univerzitetu Adelaide u Australiji. Sistem čine dva algoritma, minimax i rešenje kojim je upravljao genetski algoritam [4]. Veoma značajan je i projekat MLeXAI, razvijen za nastavu iz oblasti veštačke inteligencije, sa elementima mašinskog učenja. Svaki od 26 podprojekata tog projekta, uključuje razvoj mašinskog učenja na određenoj aplikaciji. Cilj projekta je da napravi prilagodljiv sistem za prezentovanje tema iz veštačke inteligencije i tako prikaže veliku povezanost veštačke inteligencije i računarskih nauka [5].

Navedeni sistemi su uspešno primenjeni na određenim kursovima iz razmatrane oblasti, ali ne ispunjavaju postavljene zahteve. Zato su autori rada realizovali sopstveni softverski sistem koji je opisan u narednom delu rada, zajedno sa opisom laboratorijskih vežbi zasnovanih na ovom sistemu.

### III. OPIS SOFTVERSKOG SISTEMA

Rad na sistemu se obavlja preko intuitivnog grafičkog korisničkog interfejsa. Pri izradi interfejsa vodilo se računa o principima jednostavnosti, intuitivnosti i mogućnosti brzog i jednostavnog rada sa sistemom. Sve komande imaju karakterističan izgled, tako da prosečan

korisnik može bez prethodnog poznavanja sistema da pretpostavi šta koja komanda radi i kako se koristi.



Slika 1. Meni softverskog sistema i dijalog za izbor podsistema

Sličnost funkcionisanja sa tipičnim Windows aplikacijama bi trebalo da sistem još više približi krajnjim korisnicima. Sve komande zajedničke za sve podsisteme imaju i prečice na tastaturi radi lakšeg pristupa i bržeg rada sa sistemom ukoliko to odgovara korisniku, a omogućeno je i dodavanje specifičnih komandi za različite podsisteme.

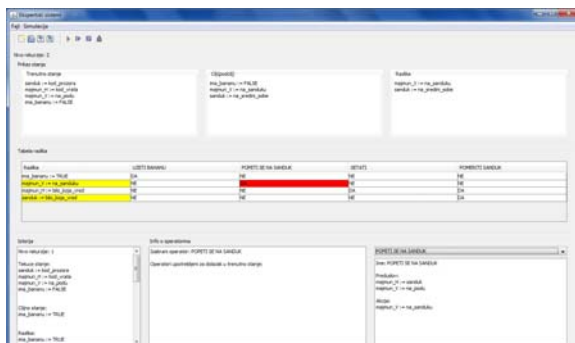
Prvi korak u okviru sistema je da se izabere podsistem koji želi da se koristi. Svaki podsistem pokriva jednu od navedenih tema: algoritmi pretraživanja, pretraživanje u igrama, formalna logika, produkcionni sistemi, sistem za planiranje STRIPS (engl. Stanford Research Institute Problem Solver), opšti rešavač problema GPS (engl. General Problem Solver), rezonovanje na osnovu faktora izvesnosti, fuzzy logika, sistem za održavanje istinitosti (engl. Truth Maintenance System), indukcioni sistemi.

Pored same simulacije, koja će biti detaljno opisana u narednom poglavlju, sistem pruža i mogućnost snimanja simulacije, koja je trenutno u toku, u fajl i učitavanje simulacije iz fajla. Na ovaj način se svi parametri simulacije mogu podesiti, simulacija pokrenuti do određenog koraka, a zatim snimiti u fajl. Kada se kasnije taj fajl pozove, svi parametri simulacije se učitavaju i može se nastaviti rad od koraka u kome je simulacija snimljena. U slučaju da simulacija ima mnogo koraka ili student ne želi da prati simulaciju algoritma korak po korak, postoji mogućnost izvršavanja simulacije do kraja.

Unos ulaznih podataka se vrši u posebnom prozoru, i tako se tokom unosa parametara simulacije sprečava njeno pokretanje ili bilo kakva akcija nad simulacijom. Glavni prozor u zavisnosti koji je podsistem pokrenut, podeljen je na više delova i služi za praćenje relevantnih stanja toka simulacije - trenutnih stanja i ciljeva, uz mogućnost da korisnik u svakom trenutku isprati svaki korak rada algoritma.

Za lakši pristup dodatnim funkcionalnostima, svaki podsistem u traci sa alatima ima i tri dugmeta za: objašnjenje algoritma, pomoć oko korišćenja odabranog podsistema i generisanje izveštaja. Nakon prolaska kroz simulaciju određenog algoritma, sistem ima mogućnost generisanja izveštaja sa detaljnim rešenjem u vidu eksternog fajla, koji će studentima omogućiti da analiziraju rešenje nakon završene laboratorijske vežbe.

Sistem je realizovan korišćenjem programskog jezika Java, a za pravljenje korisničkog interfejsa korišćene su grafičke biblioteke AWT i Swing. Osnovna osobina arhitekture sistema je modularnost, tako da se na efikasan način može izvršiti nadogradnja sistema dodavanjem novih simulacija i algoritama.



Slika 2. Softverski sistem za učenje ekspertskih sistema, prikaz jedne simulacije

#### IV. LABORATORIJSKE VEŽBE ZASNOVANE NA SOFTVERSKOM SISTEMU

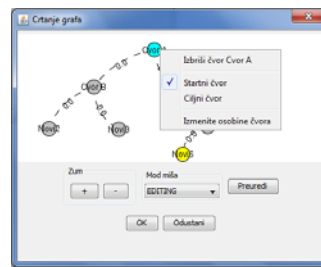
Skup laboratorijskih vežbi je zamišljen tako da pokrije sve teme koje se izučavaju na predmetu Ekspertski sistemi na Elektrotehničkom fakultetu u Beogradu. Organizovane su u sledeće celine:

- Strategije pretraživanja i algoritmi teorije igara
- Zaključivanje u formalnoj logici
- Produkcionni sistemi
- Algoritmi planiranja
- Rad u neizvesnom okruženju
- Algoritmi fuzzy logike
- Sistemi za održavanje istinitosti

Rad studenata u okviru laboratorijskih vežbi će se demonstrirati na primeru algoritama pretraživanja.

Nakon pokretanja sistema korisnik dobija ekran za izbor jednog od navedenih podsistema. Nakon izbora, u zavisnosti koji algoritam ili tehnika je u pitanju, prikazuju se dodatne ikonice u traci sa alatima. Kada je izabran podsistem za algoritme pretraživanja, u okviru simulacije se iscrta novi ili učitava ranije snimljeni graf. Crtanje grafa je vrlo intuitivno, potrebno je samo mišem uneti čvorove i spojiti ih vezama. Postoje tri moda rada sa mišem kod iscrtaavanja grafa: uređivanje grafa, dohvatanje čvora i transformacija grafa. Kod uređivanja grafa, kada se klikne desnim dugmetom miša na određeni čvor, moguće je taj čvor proglasiti da je početni ili ciljni čvor. Zatim, mogu da se izmene osobine čvora kao što su ime čvora i heuristička vrednost. Svaki čvor koji se doda u graf, može i da se izbriše u svakom trenutku, što uklanja i sve njegove veze. Veze se crtaju između čvorova, povlačenjem kursora miša od jednog do drugog čvora. Za vezu se na idetntičan način mogu izmeniti osobine: cena i usmerenost. Usmerenost može biti neusmerena, od jednog čvora ka drugom i obrnuto. Veze između određenih čvorova mogu u svakom trenutku da se raskinu. Drugi mod je dohvatanje čvora, kada mogu da se označe jedan ili više čvorova i da se pomeraju kako bi se dobio lepši izgled grafa. Izgled grafa se može i automatski menjati opcijom za preuređivanje. Prilikom crtanja grafa postoje i opcije za uveličavanje i umanjenje prikaza grafa.

Kada se graf iscrta ili učitava iz fajla, potrebno je da se odredi koji će algoritam biti primenjen - po širini, po dubini, metodom planinarenja, metodom prvo najbolji, metodom grananja i ograničavanja i metodom A\*, pri čemu su obrađene i varijante sa dinamičkim programiranjem.



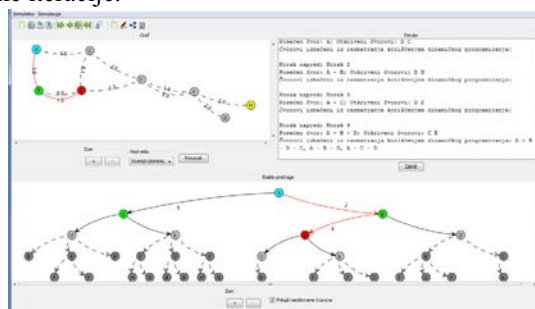
Slika 3. Crtanje grafa

Graf se učitava iz tekstualnog fajla, koji zbog parsiranja podataka treba da ima navedene prvo sve čvorove, a zatim sve veze, u sledećem formatu:

- početni čvor u grafu: A 6 start
- čvor u grafu: B 5
- ciljni čvor u grafu: H 0 end
- veza: A - B - 3

gde su A, D i H nazivi čvora, 6, 5 i 0 heurističke vrednosti, a A - B - 3 predstavlja vezu sa cenom. Usmerenost veze se označava dodavanjem strelica (<- ili ->) umesto crtice (-).

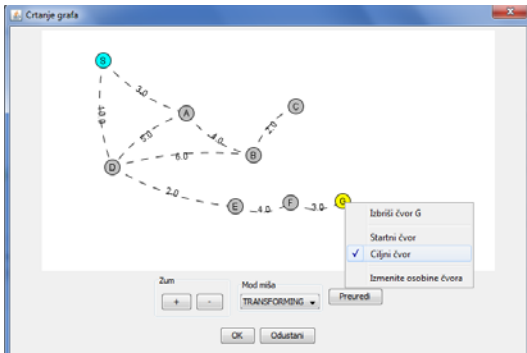
Glavni prozor podsistema za algoritme pretraživanja podeljen je na 3 dela. U gornjem delu ispod menija i trake sa alatima, nalazi se prozor sa iscrtanim grafom i prozor sa porukama, koje se ispisuju kretanjem kroz simulaciju. U donjem delu nalazi se stablo pretrage, gde se vidi stablo sa, do tog koraka, otkrivenim čvorovima ili opciono stablo i sa neotkrivenim čvorovima. Pri simulaciji algoritama pretraživanja, sadržaj u sva tri prozora se menja prilikom svake iteracije.



Slika 4. Glavni prozor podsistema za algoritme pretraživanja

Zadatak koji studenti izvršavaju je sledećeg tipa. Data je mreža puteva označena dužinama puteva u kilometrima i heuristička funkcija koja prikazuje vazдушna rastojanja od pojedinih gradova do grada G u kilometrima. Na ovom primeru studenti realizuju stablo pretrage i razmatraju redoslede obilaženja čvorova. Konkretni zadatak je da se prikaže pretraga za nalaženje odgovarajućeg puta između gradova S i G, ako se koriste različiti algoritmi pretraživanja [6].

Za stablo pretrage po dubini, redosled obilaženja čvorova je sledeći: S, A, B, C, D, E, F, G. Nađeni put je S-A-B-D-E-F-G, zato što je iz čvora C kontrola vraćena na čvor D, drugi sledbenik čvora B. Ovaj put ima dužinu 22 km, što nije najkraća dužina puta između gradova S i G. Stablo pretrage po širini samo je malo manje od kompletnog stabla pretrage za ovaj problem, jer se prvo prolaze čvorovi na istom nivou, pa tek onda po dubini. Nađeni put dužine 19 km, S-D-E-F-G, prolazi kroz najmanji broj gradova, ali nije najkraći mogući.



Slika 5. Izgled grafa iz prikazanog zadatka

U slučaju pretrage metodom planinarenja gde se koristi heuristička funkcija vazdušno rastojanje, nalazi se ista putanja kao pri pretrazi po širini. Ovaj primer ilustruje nedostatak metode izbora najboljeg čvora lokalno (po vrednosti heurističke funkcije procene), tačnije među sledbenicima tekućeg čvora: po razvijanju čvora S, bira se čvor D, u skladu sa heurističkom funkcijom koja prikazuje vazdušna rastojanja. Po razvijanju čvora D, bira se B kao najbolji, a zatim se razvijanjem čvora B dobijaju čvorovi A i C. Oba ova čvora su lošiji od već otkrivenog E, sledbenika čvora D, ali čvor E neće biti izabran dok se ne obiđu i A i C, jer je izbor čvorova lokalna. Čvor A na putnji S-D-B-A nije dalje razvijan u stablu pretrage, jer bi to dovelo do stvaranja zatvorenih petlji.

Primenom metode prvo najbolji, nađeno rešenje je isto kao u prethodnom slučaju. Jedina razlika je, što se kod ovog metoda najbolji čvor bira globalno, među svim otkrivenim, a neobrađenim čvorovima, izbegava se obilaženje sledbenika A i C čvora B, što znači da se manje skreće sa ciljne putanje nego u prethodnom slučaju.

Stablo pretrage po metodu grananja i ograničavanja, nalazi optimalno rešenje problema S-A-D-E-F-G, putanju dužine 17 km. Stablo pretrage je slično kao i u slučaju pretrage po širini odnosno obuhvata skoro kompletno stablo pretrage. Kod ovog algoritma bira se parcijalna putanja najmanje dužine. Pre završetka pretrage potrebno je bilo produžiti sve otkrivene parcijalne putanje, kraće od 17, koliko iznosi dužina nađene putanje do G. Nekad je moguće da se ovakvim produžavanjem nađe nova, kraća putanja do ciljnog čvora, što se u ovom slučaju nije desilo. Pretraživanjem metodom A\* dobija se ista minimalna putanja kao i kod metoda grananja i ograničavanja. Za razliku od tog metoda, odstupanje od ciljne putanje pri pretrazi stabla je relativno malo zahvaljujući korišćenju heurističke funkcije. Prilikom izvršavanja algoritma se koristi i dinamičko programiranje, koje sistem podržava i za ostale navedene algoritme pretraživanja.

Rešavajući navedeni simulator studentima je vizuelno ilustrovan rad algoritama pretraživanja i prednosti i nedostaci pojedinih metoda, kako onih u klasi traženja bilo kog rešenja, tako i onih u klasi traženja optimalnog rešenja. Za određeni problem, bolji su algoritmi koji za isti graf pretrage i uz korišćenje istih funkcija procene stanja i cene operatora promene stanja, dolaze do rešenja obilaženjem manjeg broja čvorova stabla pretrage.

## V. ZAKLJUČAK

U ovom radu predstavljen je softverski sistem koji

omogućava simulaciju algoritama iz oblasti ekspertskih sistema. Katedra za Računarsku tehniku i informatiku, Elektrotehničkog fakulteta u Beogradu, već nekoliko godina ima iskustvo u realizaciji i korišćenju ovakvih sistema u nastavi na nekoliko predmetima, jer studentima pruža efikasnije praćenje gradiva. Takođe, novi svetski trendovi u obrazovanju i primena Bolonjske deklaracije, sve više insistiraju na uvođenju učenja uz pomoć računara (engl. e-learning).

Korist koju sistem donosi predavačima je mogućnost vizuelnog prikaza predavanih koncepata. Tako predavač apstraktne i složene algoritme približava studentima na moderan i efekatan način. Studenti imaju mogućnost da simuliraju algoritme u različitim uslovima, sa različitim inicijalnim podacima i sa različitim primenom rešenja. Analiza algoritama je na taj način sistematičnija i racionalnija, pa studenti jednostavnije mogu da uoče prednosti i mane razmatranih algoritama. Moguća nadogradnja sistema se može razmatrati u nekoliko pravaca. Moguće su implementacije dodatnih algoritama i tehnika, koje se uče prema programu predmeta. Sa korisničke strane moguće je omogućiti paralelno učitavanje i praćenje više simulacija istovremeno, na primer kod navedenog podsistema za algoritme pretraživanja. Autori planiraju da korišćenje sistema učine dostupnim na Internetu.

## LITERATURA

- [1] [http://www.acm.org/education/education/curric\\_vols/CC2005-March06Final.pdf](http://www.acm.org/education/education/curric_vols/CC2005-March06Final.pdf), accessed on September 8<sup>th</sup>, 2010
- [2] <http://www.acm.org/education/curricula/ComputerScience2008.pdf>, accessed on September 8<sup>th</sup>, 2010
- [3] I.J.Borm, The Design and Implementation of a Multi-Agent Soccer Simulator as a Tool for AI Research and Education, Master's Thesis in Media & Knowledge Engineering, Faculty of Electrical Engineering, Mathematics, and Computer Science, Delft University of Technology, September 2006
- [4] R. Hill, A. van den Hengel, Experiences with simulated robot soccer as a teaching tool, Information Technology and Applications, ICITA 2005, 3rd International Conference on Volume: 1, Page(s): 387 - 390
- [5] Z. Markov, I. Russell, T. Neller, S. Coleman, Enhancing undergraduate AI courses through machine learning projects, Frontiers in Education, 2005, FIE '05, Proceedings 35th Annual Conference; Publication Year: 2005, Page(s): T3E - 21
- [6] D. Bojic, M. Gligoric, B. Nikolic, Zbirka zadataka iz ekspertskih sistema, Akademski misao, Beograd, 2009, ISBN: 978-86-7466-362-2

## ABSTRACT

This paper describes the software system for expert system learning which was realized at the Faculty of Electrical Engineering in Belgrade. The software has been developed as an educational system for students and will be used for teaching the subject Expert Systems within undergraduate and Master studies. The system guides students through examples and step-by-step tasks related to topics to which they belong and which are done as part of lectures and tutorials. The following issues are covered: breadth-first and depth-first strategies, hill-climbing method, best-first method, branch and bound methods, method A\*, search in games, production systems, general problem solver, STRIPS planning system and undetermined environment operation systems – fuzzy logic, probability factor reasoning and other ways of expressing uncertainty. Students may enter they own examples and tasks and thus obtain correct solutions. At every moment, there is a possibility of going one step back or forward. At the very end, a student may also print the detailed how-to procedure of solving the task. The implemented system improves lecturer's efficiency and enhances knowledge acquisition of innovative curricula.

## Software System for Expert Systems Learning

Dražen Drašković, Vuk Batanović, Boško Nikolić