

# Simulaciono okruženje za bespilotnu letelicu

Mlađan Jovanović, Dušan Starčević, Željko Obrenović, Dragan Obradović

**Sadržaj** — Simulacija je dugo vremena bila korišćena kao sredstvo za unapređenje treninga i priprema za izvršavanje zadataka. U novije vreme simulacija se pojavljuje kao tehnološka podrška svih faza životnog ciklusa projekta, uključujući istraživanje, analizu i akviziciju. Sa mogućnošću da virtuelno simuliramo sisteme, procese, komunikacije i interoperabilnost svake komponente simulacionog okruženja, simulacija redukuje troškove i rizike povezane sa svim fazama projekta. Bespilotna letelica (UAV – Unmanned Aerial Vehicle) izvršava raznovrsne zadatke kao što su mobilno taktičko izviđanje, prismostra, sprovođenje zakona, pretraga i spasavanje, nadgledanje okruženja. U realizaciji svih funkcionalnosti bespilotne letelice softverski deo postaje kompleksan sistem u realnom vremenu od koga se očekuje da konkurentno izvršava zadatke. U ovom radu su izložena iskustva u dizajnu i razvoju simulatora za laku bespilotnu letelicu namenjenu za misije izviđanja i prikupljanja podataka o okruženju. Opisani su arhitektura i implementacija rešenja.

**Ključne reči** — bespilotna letelica, elevacioni model, java, simulacija.

## I. UVOD

BESPILOTNA letelica se definiše kao "vazdušno vozilo sa sopstvenim pogonom koje ne nosi ljudskog operatora, koristi aerodinamičke sile za održavanje u vazduhu, može leteti autonomno ili biti upravljano, može biti potrošno ili ponovno upotrebljivo, može nositi smrtonosni ili nesmrtonosni teret" [1]. Bespilotne letelice nalaze primenu u velikom broju različitih oblasti. Pored vojnih primena, od određivanja ciljeva i izviđanja do isporuke naoružanja [2], bespilotne letelice u poslednje vreme nalaze veliku primenu i u civilne svrhe. Specifične primene obuhvataju operacije pretrage i spasavanja, akvizicija podataka o okruženju, nadgledanje drumskog saobraćaja, šumskih požara, poplava, poljoprivredne primene i dr.. Realizacija tipične bespilotne letelice zahteva kompleksne hardverske i softverske komponente kao i strogo vremenski kritičnu koordinaciju ovih komponenti. Softverski deo obuhvata tri grupe konkurentnih zadataka – tvrdi zadaci u realnom vremenu (eng. *hard real-time tasks*), meki zadaci u realnom vremenu (eng. *soft real-time tasks*) i zadaci koji se ne

izvršavaju u realnom vremenu (eng. *non-real-time tasks*) [3]. Imajući u vidu visoku cenu izrade bespilotne letelice, računarska simulacija se intenzivno koristi za testiranje rešenja koja realizuju prethodno pomenute kritične aspekte. Ona konkretno obuhvata dizajniranje modela i simulatora sa promenljivim nivoima preciznosti i funkcionalnosti, istraživanje i razvoj mogućih rešenja i isporuku gotovih rešenja.

U radu se daje opis simulacionog okruženja za vizuelizaciju, kontrolu i simulaciju operacija prikupljanja podataka lake bespilotne letelice. U narednom odeljku se daje kratak pregled postojećih rešenja. Zatim se daje opis dizajna i prikaz arhitekture softverskog sistema. Odeljci 4 i 5 prikazuju detalje razvijenog simulacionog okruženja. Konačno, u poslednjem odeljku se daje diskusija prezentovanog rada i zaključak.

## II. POSTOJEĆA REŠENJA

Poslednjih nekoliko godina širok spektar istraživanja se obavlja u oblasti bespilotnih letelica. Svako od njih ima za predmet specifične oblasti primene bespilotnih letelica. Modelovanje i simulacija dobijaju sve veći značaj s obzirom da se modeli i simulatori koriste kao zamene za elemente sistema koji nisu razvijeni u potpunosti. Na ovaj način se razvijeni elementi mogu inkrementalno integrisati u sistem [4]. Institut za tehnologiju vazduhoplovnih snaga vojske SAD je razvio jedan od prvih simulatora u obliku virtuelnog kokpita za distribuiranu interaktivnu simulaciju [5]. NASA laboratorije za simulaciju razvijaju širok spektar simulatora među kojima je i simulacija kontrole rojeva bespilotnih letelica [6]. Džordžija institut za tehnologiju je dizajnirao i razvio multisenzorski sistem za bespilotnu letelicu namenjen za akviziciju navigacionih podataka [7].

U okviru našeg projekta razvijena je mala bespilotna letelica. U prvoj fazi razvoja bio je dat naglasak na razvoju komunikacionog [8] i akvizicionog [9] sistema. U tekućoj fazi razvoja je implementacija softverskih sistema na zemaljskoj stanici koji se pre svega odnose na vizualizaciju primljenih podataka od letelice kao i na planiranje leta. Prikupljeni podaci sa eksperimentalnih letova su bili korišćeni u razvoju softverskih aplikacija za zemaljsku stanicu.

## III. DIZAJN I FUNKCIONALNI OPIS REŠENJA

Glavni zadatak simulatora je produkcija i prezentacija podataka o stanju okruženja i stanju letačke platforme. Ovi podaci uključuju globalne pozicione podatke (GPS), podatke vezane za misiju, podatke kontrole letelice i parametre stanja letelice. Letelicom se upravlja sa

Mlađan Jovanović, Centar za komandno-informacione sisteme i informatičku podršku, Vojska Srbije, Beograd, Srbija, (e-mail: mladjan@afrodita.rcub.bg.ac.yu).

Dušan Starčević, Fakultet organizacionih nauka, Univerzitet u Beogradu, (e-mail: starcev@fon.bg.ac.yu).

Željko Obrenović, Centrum voor Wiskunde en Informatica (CWI), Amsterdam, The Netherlands, (e-mail: Zeljko.Obrenovic@cwi.nl).

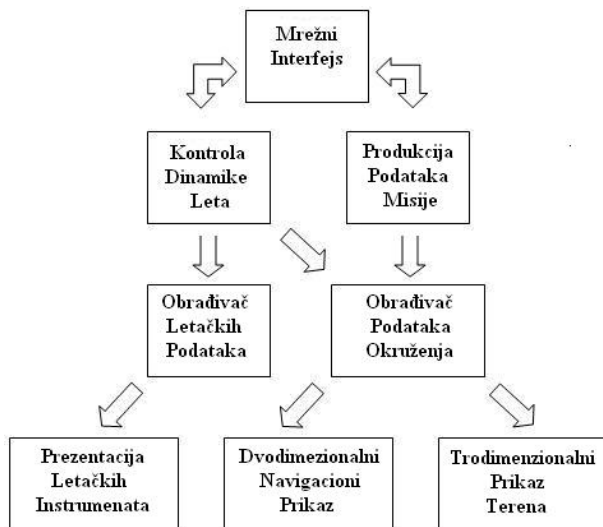
Dragan Obradović, IMTEL-Komunikacije, Bul. M. Pupina 165B, 11070 Novi Beograd, Srbija (e-mail: obrad@insimtel.com).

zemaljske kontrolne stanice u kojoj je operator odgovoran za:

- Primarni zadatak misije u kojem operator prati kontrolne tačke i izveštava o ciljevima uočenim na poznatim koordinatama.
- Sekundarni zadatak misije u kojem operator upravlja letelicom kroz simulirano prirodno okruženje.
- Zadatak nadgledanja sistema u kojem operator nadgleda letačke instrumente i detektuje moguće otkaze sistema.

Osnovne aktivnosti povezane sa izvršavanjem softverskog sistema jesu prikupljanje ulaznih parametara, procesiranje i generisanje izlaznih parametara. U konkretnom slučaju aktivnosti se izvršavaju sinhrono za ulazima i izlazima sistema u skladu sa promenjivom periodom.

Specifikacija programskog jezika JAVA u realnom vremenu (*RTSJ – Real Time Specification for Java*) je razvijena u cilju integracije JAVA komponenti sa komponentama za rad u realnom vremenu i postizanja boljih performansi [10]. OVM (*Open Virtual Machine*) virtualna mašina predstavlja implementaciju same specifikacije [11]. Ona realizuje neke od ključnih funkcionalnosti kao što su raspoređivanje niti i upravljanje memorijom i u skladu sa tim je upotrebljena kao izvršna platforma u razvijenom simulatoru. Slika 1 prikazuje predloženu arhitekturu sistema.



Slika 1. Predlog arhitekture sistema

Simulaciono okruženje je razvijeno sa ciljem da omogući operatoru da planira i izvršava misije. Moduli kontrole dinamike leta i produkcije podataka misije (Slika 1.) simuliraju kontrolu letelice iz zemaljske kontrolne stanice i proizvode podatke o misiji, respektivno. Podaci o stanju okruženja i letačke platforme su proizvedeni kao poruke specifičnih formata. Kao takve ih obrađuju odgovarajući moduli i čine podesnim za vizuelnu prezentaciju. Moduli za navigaciju i prikaz terena prikazuju okruženje reljefa i navigaciju duž kontrolnih tačaka misije. Operator nadgleda stanje letačke platforme i identifikuje otkaze u sistemu. Ovaj zadatak obezbeđuje modul za prezentaciju letačkih instrumenata. Preko

mrežnog interfejsa sistem može primati simulacione podatke od drugih sistema i prosledivati podatke zainteresovanim sistemima. Na ovaj način sistem može biti konfigurisan sa mobilnim uređajima kao što su lični digitalni asistenti.

Korisnik najpre definiše misiju tako što specificira kontrolne tačke putanje i ciljeve od interesa za osmatranje. Zatim se izvodi virtuelno poletanje sa svim podešavanjima parametara leta. U toku leta korisnik nadgleda stanje same letačke platforme kao i status izvršenja primarnog zadatka leta, tj. stepen izvršenja misije. U narednim sekcijama je dat opis razvijenih modula.

#### IV. PREZENTACIJA LETAČKIH INSTRUMENTATA

Moderne letelice proizvode velike količine podataka koji moraju biti prezentovane operatoru u realnom vremenu. U situacijama u kojima operator treba da reaguje u ograničenom periodu vremena neophodno je prezentovati podatke o letu tako da se mogu interpretirati i procesirati imajući na umu propusnost ljudskih senzorskih i percepcijskih aparata.

Instrument tabla je prikazana na slici 2. Instrumenti sa povezanim funkcijama su razmešteni i grupisani tako da redukuju vreme skeniranja instrumenata i naprezanje operatora.

Glavni zahtevi prilikom dizajna instrument table jesu:

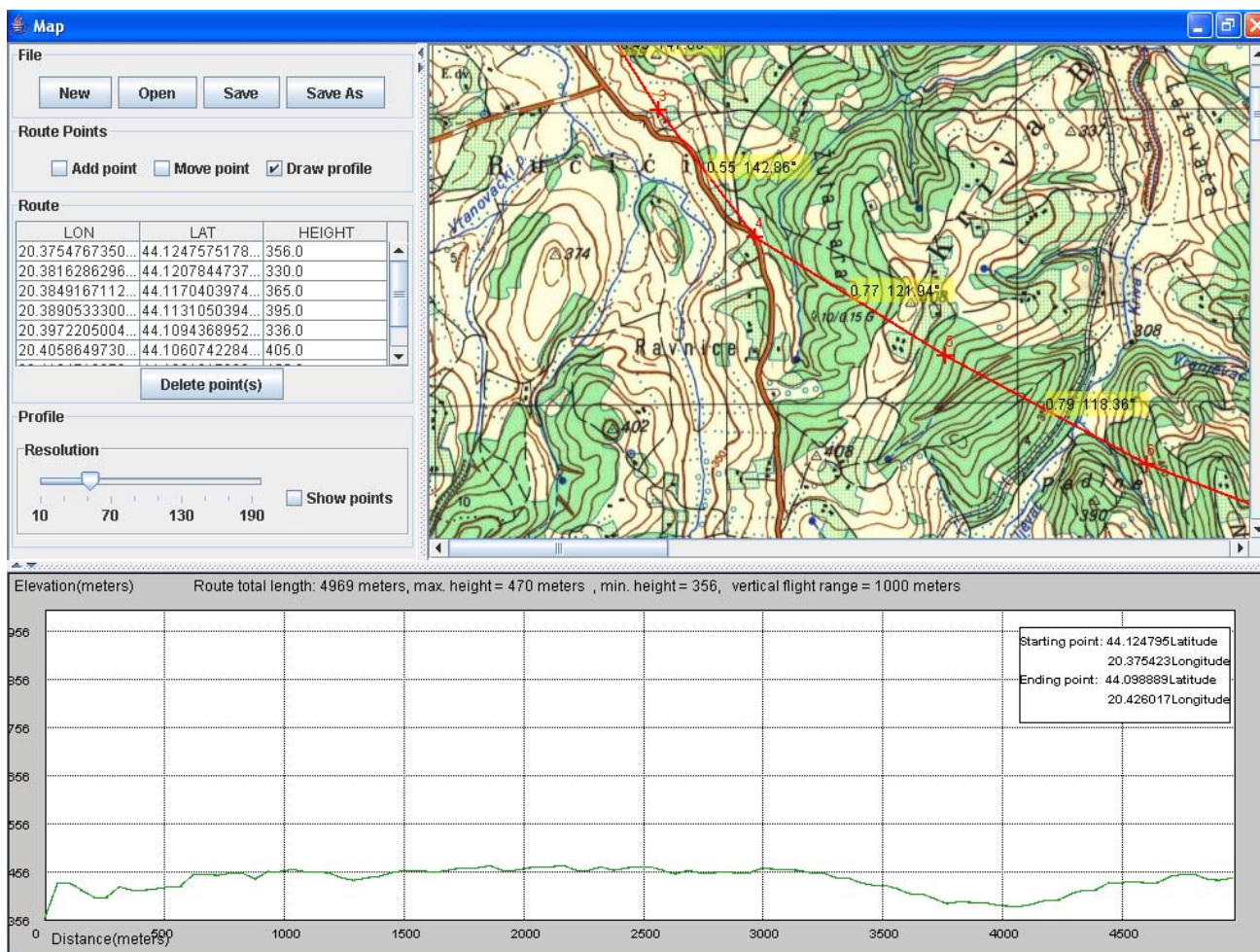
- Kognitivni cilj. Raspored i funkcionisanje kontrola treba da odgovara intuitivnom razumevanju operatora i da mu omogući jasan pregled statusa letačke platforme u svakom trenutku.
- Vreme odziva. Parametre leta je potrebno pravovremeno prikazivati tako da reflektuju stvarno stanje letelice. Ovo je postignuto tehnologijom implementacije.



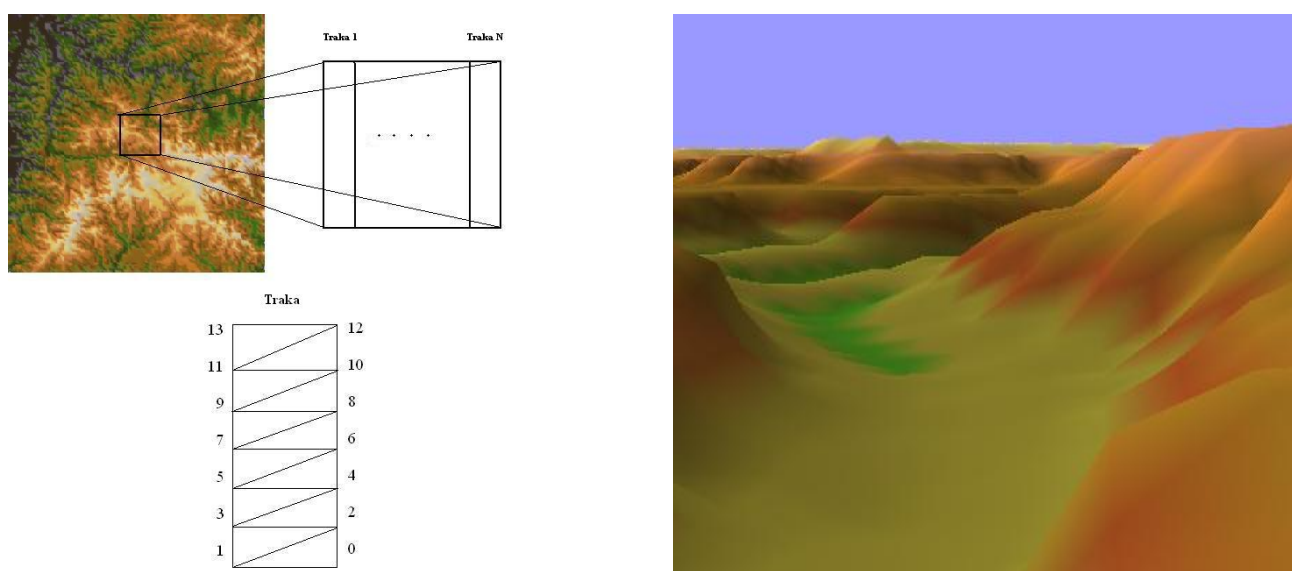
Slika 2. Instrument tabla

#### V. MODELOVANJE OKRUŽENJA

Nadgledanje i kontrola misije bespilotne letelice se obavljaju preko dvodimenzionalnog navigacionog prikaza i trodimenzionalnog prikaza reljefa terena. Navigacioni prikaz sadrži pogled na profil terena i pogled za navigaciju duž putanje definisane rute leta (Slika 3). Misija se sastoji



Slika 3. Navigacioni prikaz za kontrolu misije



Slika 4. Digitalni elevacioni model (a) i generisani prikaz (b)

od određenog broja ruta koje sadrže kontrolne tačke koje treba posetiti i opisati njihovo stanje. Operatoru je omogućeno da ručno unosi pozicije kontrolnih tačaka. Na osnovu unetih pozicija vrši se računanje parametara rute. Letelicom se može ručno upravljati, a takođe je realizovana funkcionalnost autopilota sa mogućnošću korekcije putanje.

Reprezentacija terena je kreirana na osnovu digitalnog elevacionog modela predstavljenog u obliku pravilno raspoređene pravougaone mreže. Sami podaci su organizovani kao trake trouglova (Slika 4a). JAVA 3D predstavlja grafičku biblioteku sa ugrađenim mehanizmima za efektivno prikazivanje ovakvih struktura podataka. Parametri su sadržani po referenci i deljeni između grafičkih rutina i korisničkog koda te je postignuto efikasno korišćenje memorije. Funkcionalnost prikazivanja nivoa detalja omogućava operatoru izbor preciznosti trodimenzionalnog prikaza terena. Slika 4b daje generisani prikaz terena.

## VI. ZAKLJUČAK

U ovom radu je opisano simulaciono okruženje za bespilotnu letelicu. Sa mogućnošću da simuliramo sisteme, procese, komunikacije i interoperabilnost svake komponente u simuliranom okruženju, simulacija omogućava redukovanje cene i rizika povezanih sa svim fazama projekta.

Sistemom je prvenstveno simuliran proces akvizicije podataka o okruženju. Sam proces podrazumeva definisanje misija koje sadrže ciljeve koje treba detektovati i opisati. Operator nadgleda sistem kroz tri prikaza. Prvi predstavlja instrument tablu za prezentaciju letaćkih podataka i stanje letelice. Drugi prikazuje mapu terena zajedno sa profilom i služi za praćenje misije. Treći prikaz generiše trodimenzionalni prikaz terena i služi za precizniju navigaciju kroz scenu.

Simulator je testiran na različitim konfiguracijama terena i misijama različite kompleksnosti. U toku svake misije posmatrani su broj otkrivenih ciljeva i performanse izvršavanja sistema. Na osnovu testova i dobijenih rezultata pristupilo se razvoju realnog sistema za upravljanje bespilotnom letelicom.

## LITERATURA

- [1] E. Bone and C. Bolkcom, "Unmanned aerial vehicles: Background and issues for congress," Report for Congress, 2003.
- [2] Unmanned Aircraft Systems Roadmap 2005, Office of the Secretary of Defence.
- [3] W. E. Hong, J. S. Lee, L. Rai and S. J. Kang, "RT-Linux based Hard Real-Time Software Architecture for Unmanned Autonomous Helicopters," Proceedings of the 11th IEEE Conference on Embedded and Real-Time Computing Systems and Applications (RTCSA 2005).
- [4] E. Theunissen, A. A. H. E. Goossens, O. F. Bleeker and G. J. M. Koeners, "UAV Mission Management Functions to Support Integration in a Strategic and Tactical ATC and C2 Environment," Proceedings of the AAIA Modeling and Simulation Technologies Conference, Paper number 6310, August 2005.
- [5] W. D. McCarty, S. Sheasby, P. Amburn, M. R. Stytz and C. Switzer, "A Virtual Cockpit for a Distributed Interactive Simulation," IEEE Computer Graphics and Application, pp. 49-54, January 1994.

- [6] S. R. Flaherty and R. J. Shively, "Manned-Unmanned Teaming Simulation with Control of Multiple Unmanned Aerial Vehicles", NASA Ames Simulation Laboratories 2005 Year in Review, <http://www.simlabs.arc.nasa.gov>
- [7] J. S. Dittich and E. N. Johnson, "A Multi-Sensor Navigation System for an Autonomous Helicopter", Proceedings of the AAIA Digital Avionics Conference, Paper number 377, October 2002.
- [8] Smiljaković V, Golubičić Z, Manojlović P, Živanović Z, Microwave Autonomous Angular Position Finding System for Middle Range Unmanned Aerial Vehicle, Proceedings of XXXVIII ICESS, Sofia, 16-18 October 2003.
- [9] Obradović D., Smiljaković V, Popović N., Manojlović P., Nešić A., Obrenović Ž., Starčević D. Akvizicioni sistem male bespilotne letelice, TELFOR 2005, 22-24. XI 2005., Beograd
- [10] G. Bollella, J. Gosling, B. Brosgol, P. Dibble, S. Furr and M. Turnbull, "The Real-Time Specification for Java", Addison-Wesley, 2000.
- [11] The Ovm virtual machine, <http://www.ovmj.org>

## ABSTRACT

Simulation has long been applied to enhance training and preparation. Nowadays, however, simulation is being used as a technological enabler to support all phases of a project lifecycle, including research, analysis, and acquisition. With the ability to virtually simulate systems, processes, communications and the interoperability of each in a simulation environment, simulation helps reduce the cost and risk associated with all phases of a project. UAV (Unmanned Aerial Vehicle) performs various kinds of missions such as mobile tactical reconnaissance, surveillance, law enforcement, search and rescue, environmental monitoring. In order to realize all functionalities of the UAV, the software part becomes very complex real-time system expected to execute real-time tasks concurrently. This paper reports experiences of designing and developing simulator for lightweight UAV purpose-built for reconnaissance and environment data acquisition missions. The overall system architecture and implementation are described.

## UAV-BASED SIMULATION ENVIRONMENT

Mlađan Jovanović, Dušan Starčević, Željko Obrenović,  
Dragan Obradović