

СТРУЧНИ ЧЛАНЦИ
ПРОФЕСИОНАЛНЫЕ СТАТЬИ
PROFESSIONAL PAPERS

OPEN LOOP CONTROL OF THE FIVE-AXIS MISSILE AND TARGET FLIGHT MOTION SIMULATOR IMPLEMENTATION

Nataša D. Kljajić^a, Miloš I. Kostić^b, Miloš V. Pavić^c

Ministry of Defense of the Republic of Serbia,
Military Technical Institute, Belgrade, Republic of Serbia

^a e-mail: natasha.kljajic@yahoo.com,

ORCID iD: <http://orcid.org/0000-0001-9708-0964>

^b e-mail: mixkostic@gmail.com,

ORCID iD: <http://orcid.org/0000-0001-7132-4780>

^c e-mail: cnn@beotel.rs,

ORCID iD: <http://orcid.org/0000-0002-0013-9745>

DOI: 10.5937/vojtehg64-9493

FIELD: Electrical Engineering

ARTICLE TYPE: Professional Paper

ARTICLE LANGUAGE: English

Summary:

An important phase in missile systems development is flight testing and simulation in the environmental conditions dictated by the operation the system is made for. Since field testing of complex systems means a big financial burden and a time consuming process, hardware in the loop (HIL) simulations represent a very effective solution for saving both costs and time. This paper presents an implementation of a control application that integrates synchronized data generation and acquisition on the five-axis missile and target flight motion simulator owned by the HIL laboratory. The described simulation application is a result of a missile development engineer's need for a simple way to acquire information on a real missile response, when desired control signals are provided. The solution is realized in LabVIEW software using a robust software design architecture named "classic state machine".

Key words: tests, simulations, missiles, data acquisition, signal generation, state machine, synchronized measurements, LabVIEW, hardware in the loop.

Introduction

Surface-to-air missile systems are developed to meet specified tactical and operational requirements in environmental conditions dictated by the operation itself. From the initial formulation of the concept for a new missile system to the end of the life cycle of the missile, there is a continuous need to predict the performance of alternative designs of the missile. It is important to emphasize that computational modelling and simulations with a set of partial differential equations bring uncertainty and errors when switching to realistic application (Oberkampf, DeLand, Rutherford, Diegert, Alvin, 2001). This happens because it is not easy to incorporate all environmental interactions into a mathematical model. That is why the most valuable source of information is missile field testing followed by missile flight simulation (USA, Department of Defense, 1995). Since field testing presents enormous financial cost, especially in the missile development domain, where field testing means a destruction of a missile itself, hardware in the loop simulations have proven to be a very cost-effective solution. Also, there is more pressure nowadays to get products to market faster and to reduce design cycle time. This has led to a need for dynamic testing, where components are tested while in use with the entire system, either real or simulated, with real-time hardware and software. A missile flight simulation is a way to find the missile performance under varying dynamics that is impossible to find by flight tests or by an analytical solution, so even nowadays this is a hot research topic (Awad, Wang, 2014). So, hardware in the loop (HIL) simulations become an integrated part of the design and testing cycle (National Instruments, 2008).

The simulation application described in detail in this paper is a result of a need of missile development engineers for a simple way to acquire information on a real missile response when desired control signals are provided.

HIL simulations

A hardware-in-the-loop simulation enables a highly realistic simulation of equipment in an operational environment. A typical HIL system includes sensors to receive data from the control system, actuators to send data, a controller to process data, a human-machine interface (HMI) and a development post simulation analysis platform (National Instruments, 2008).

Hardware in the loop laboratory, shown in Figure 1, is specialized for the HIL simulation of missile guidance and control systems, with real hardware, data acquisition, and signal processing.



Figure 1 – HIL laboratory with five axis missile and target flight simulator
 Рус. 1 – ПАМ Лабораторный пятиосевой имитатор движения ракеты и цели
 Slika 1 – HIL laboratorija sa petoosnim simulatorom leta rakete i cilja

The laboratory is equipped with an S-450R-5 five-axis missile and target flight motion simulator, as well as measurement, data acquisition and signal processing systems (National Instruments, Data Translation). All Series S-450R-5 Flight Motion Simulators include a control console, hydraulic power supply with supply and air installation for high pressure working. A HIL simulation enables a development of appropriate testing and evaluation methods of guidance as well as a control systems simulation in a closed loop. Yaw, pitch and roll motions of a missile are realized by the inner three axes of the simulator. The outer two axes of the flight simulator provide the azimuth and the elevation of translational motions of the target (Acutronic, 2011). A HIL simulation is necessary in order to determine important technical and operational parameters of guided missiles. Since field testing is expensive, the usage of simulation software and hardware platforms ensures savings in guided missile and unmanned aerial vehicle (UAV) development.

LabVIEW technology introduction

LabVIEW is graphical programming language with its own development environment. Numerous NASA scientists and engineers all over the world use the LabVIEW language as a basis for instrument

control, data acquisition, display, and data storage operations. With recent releases, LabVIEW has evolved from an instrument control and data acquisition tool to a general-purpose software development system that can be employed for nearly any application. Additionally, the ability to interface the LabVIEW source code and applications with other programming languages (such as Visual Basic and Matlab) means that software engineers can employ a combination of their favorite tools and development environments while working on any project (Don, Bradford, David, Hollis, Wei, 2001).

Problem overview and solution

The five-axis flight motion simulator is designed for high dynamic testing and evaluation of small to medium sized missile guidance systems (Don, Bradford, David, Hollis, Wei, 2001). In our case, the testing consists of specifying and giving control command signals and system response data acquisition. The given command signals are the parameters of a desired missile flight, all written in a textual document. Then, the data is processed and all the missing data is interpolated for purpose of adjusting to application, system and measurement requests. In the end, a system response is acquired and saved. It is necessary to have real time measurements and that is why there is a demand for synchronized command execution and data acquisition, which means synchronized output data generation and input data acquisition. For this realization, the chosen hardware is a National Instruments NI USB 6259 data acquisition device and the software is National Instruments LabVIEW v14.0. The acquired electrical signals are saved and available in the most convenient form for further analysis.

Global application structure and flowchart

Testing and measuring the structure organization is explained in the following paragraph. Control command signals are given through the PC application software written in LabVIEW programming language. A PC is connected via USB to the data acquisition device NI USB 6259. Analog inputs and outputs of the device are connected to corresponding inputs and outputs of the flight motion simulator. This hardware configuration is shown in Figure 2.

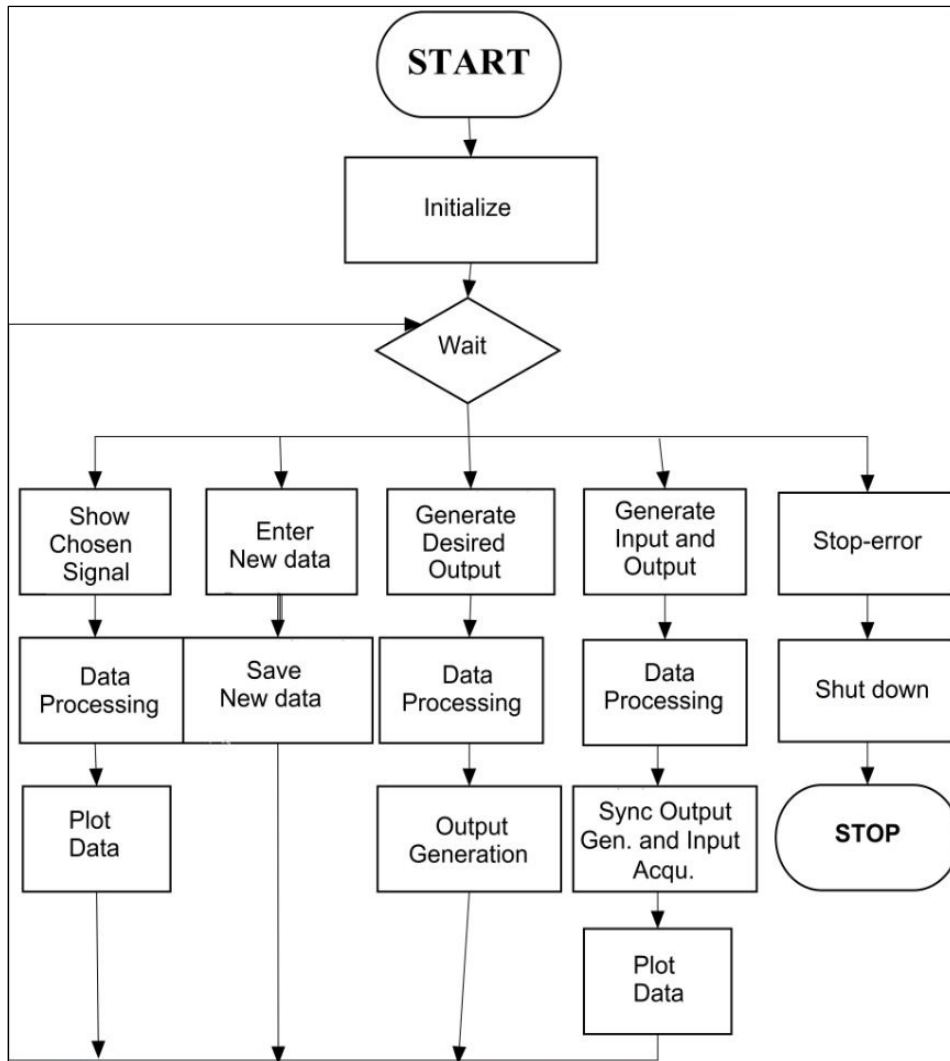


Figure 2 – Hardware configuration of the presented solution
 Рус. 2 – Конфигурация оборудования представленного решения
 Slika 2 – Hardverska konfiguracija predstavljenog rešenja

The main application flowchart is shown in Figure 3. The program options are: show control command signal, set new parameters, control commands generation, control commands generation and response data acquisition. As it can be seen in the flowchart, except for the main structure, there are also important subroutines (sub VI) explained in detail in the next paragraphs.



Figure 3 – Main flowchart
 Рус. 3 – Диаграмма набегающего потока
 Slika 3 – Dijagram toka glavnog programa

User Interface

The main task is to control the five-axis flight motion simulator, where control data is given in a textual document, and to acquire missile response data. The user interface is in Figure 4.

The simulator is controlled by the user through the main application window, which means that commands are sent to analog outputs and analog input data is acquired. The user has an option to choose the file for signal generation (which is sent to the analog output) and also to change general settings. As control command signals sent to analog outputs have to be adjusted to appropriate channel voltage levels, there are some additional settings (scale factors) for analog outputs. Possible actions for the user are: set new parameters, show signal waveform graphs, control commands generation, control commands generation and response data acquisition. The user can set and save new general parameters, or use saved values from the last measurement. Before signal generation and acquisition, the user is able to see signal waveform graphs. If there is a need to generate signals without acquisition, there is an option for this action, too, with its own interface.

By choosing the signal generation and acquisition action, the user gets synchronized signal generation and data acquisition. Simultaneous measurements involve different operations happening at the same time, such as data acquisition on the input channels while generating data on the output channels. However, these operations are not necessarily correlated to one another. Synchronized measurements, on the other hand, all need to start at the same time, as well as to share a common clock for latching data among all measurements (National Instruments, 2012).

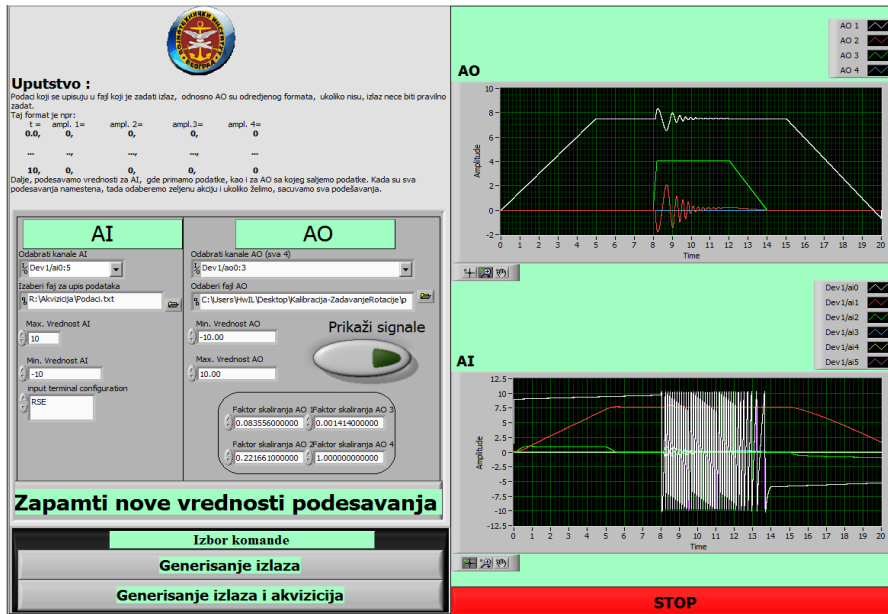


Figure 4 – User interface main window
 Puc. 4 – Главное окно пользовательского интерфейса
 Slika 4 – Glavni prozor korisničkog interfejsa

The data that the user wants to send to the simulator is in a form of textual files containing commands and corresponding time samples. The first step is to unpack the file and adjust the data for further processing in LabVIEW. Since the data is raw, which means not prepared for processing, very often it happens that time samples are not equidistant. Therefore, there is a need for the interpolation of missing data, with a previously chosen sample rate. Afterwards, the data is scaled, in order to adjust to channel voltage levels defined in the device data sheet. The processed data can now be sent to the outputs. The data acquired from the analog inputs is also saved in a form of text files, for easier analyses in any software package the user wants to use. The analysis can be conducted directly, in LabVIEW software with addition of one more subroutine for data analysis, but this request was not defined by the user. The user requirement was to pack the acquired data in a proper form to ensure an easy analysis in any kind of software used by engineers.

Software architecture

The main flowchart is shown in Figure 3. The used architecture is a classic state machine. The state machine translates state transition diagrams into a LabVIEW code that is readable, scalable and maintainable.

This way, the virtual instrument (VI) is organized into states which are general functions and features of application, and transitions that map out the ways that the VI can move from one state to another (Bress, 2013).

The described application has seven states. The states are defined based on application functional requests in a way that each state is independent and has a different function. The basic states that every variant of the state machine architecture must have are: initialize, wait and shutdown. When the application is started, the first state is “initialize state” which initializes all controls and variables. Then, the “in wait state” application waits for the user to select an action. The “shutdown state” shuts down the application due to error or stop the button click. The requested software functions, as it can be seen in the flowchart in Figure 3, are control command signal waveform graph, add new settings parameters, output generation, output generation and input data acquisition.

The basic states have a classic architecture which is almost the same for every state machine design, while functional application states have complex structures with additional subroutines, which differ from case to case. In the flowchart diagram it can be seen that, for the states control command signal waveform graph, output generation, output generation and input data acquisition, there is mutual action data processing, organized in one subroutine. This subroutine consists of unpacking textual document with desired control commands and time samples, adjusting the data for further processing, interpolation and scaling. The next integral part is a subroutine for settings parameter adding and saving. The output generation is a subroutine with its own interface. In the end, we have output generation and input data acquisition as a subroutine with its own interface, which enables synchronized output generation and input data acquisition. This is how the state machine with an error cluster data flow was formed.

Conclusion

This paper presents an implementation of a modern missile parameter testing method by using a robust software design architecture. The main advantages of this application are: it is easy to use, hardware support is compatible with software development environment used, so the probability of having hardware to software communication problems is very low and that leads to the conclusion that the testing process itself demands less time spent testing and preparing for it. The application is developed completely in accordance with the user requests, processes and saves the data in a form defined by the user. This software architecture design is readable and clear and the consequences are fewer problems with software inheritance which is certainly an advantage

The described solution is only the beginning of further automation, modernization and simplification of the flight simulation process using the five-axis flight motion simulator. The goal is to have a completely software control of the simulator axis, to be precise, its sensors and actuators, with additional applications for different HIL laboratory users. The main challenge for HIL simulation applications is the synchronization of a very large number of such I/O functions at high speeds and the rapid conversion of signal data to and from information for inputs and outputs, respectively. With the power and flexibility of today's computers, engineers and scientists are increasingly upgrading platforms for HIL simulations, so the future solution certainly means more powerful platform hardware configurations (National Instruments, 2008). The main interest of the HIL laboratory, as a part of the Military Technical Institute, for achieving this goal is cost reduction in missile systems and UAV development, as well as decreasing development time.

References

- Awad, A., Wang, H. 2014, Evaluation and Enhancing Missile performance via Real Time Flight Simulation Model, *Proceedings of the 33rd Chinese Control Conference, Nanjing, China, July 28-30.*
- Acutronic. 2011. *Datasheet-5-Axis-Flight-Motion-Simulator-S-450R-5.*
- Bress, J.T. 2013. *Effective LabVIEW Programming.*
- Don, J.R., Bradford, H.P., David, A.R., Hollis, H.J., & Wei, C. 2001. *NASA/TM-2001-210683.*
- National Instruments. 2012. *LabView Data Acquisition and Signal Conditioning, Course Manual.*
- National Instruments. 2008. *NI-Tutorial-3567, LabVIEW FPGA in Hardware-in-the-Loop Simulation Applications.*
- Oberkamp, W.L., DeLand, S.M., Rutherford, B.M., Diegert, K.V., Alvin, K.F., 2001, Error and Uncertainty in modeling and simulation, *Reliability Engineering and System Safety, 75(2002) 333-357*
- USA, Department of Defense. 1995. *Military Handbook, Missile Flight Simulation, Part One, Surface – To – Air Missiles.* MIL-HDBK-1211(MI).

ВНЕДРЕНИЕ СИСТЕМЫ УПРАВЛЕНИЯ ПЯТИОСЕВЫХ ИМИТАТОРОВ ПОЛЕТА РАКЕТЫ И ВОЗДУШНОЙ ЦЕЛИ В РАЗОМКНУТОМ ЦИКЛЕ

Наташа Д. Кляич, Милош И. Костич, Милош В. Павич
 Министерство обороны Республики Сербия,
 Военно-технический институт, г. Белград, Республика Сербия

ОБЛАСТЬ: электротехника
 ВИД СТАТЬИ: профессиональная статья
 ЯЗЫК СТАТЬИ: английский

Резюме:

Важнейшим фактором в развитии ракетных систем является этап моделирования и симуляции движения ракеты в соответствующих операционным параметрам условиях. Учитывая высокую стоимость полевых испытаний подобных систем, большие финансовые расходы и продолжительность подготовительных мероприятий, моделирование в рамках программно-аппаратного моделирования (*Hardware in the loop - HIL*) является наиболее эффективным испытательным решением.

В данной статье описан процесс внедрения системы управления, то есть, синхронизированного генерирования и активации данных пятиосевого имитатора движения ракеты и воздушной цели, разработанного в лаборатории программно-аппаратного моделирования. Описанное в работе приложение имитатора обеспечивает эффективный контроль и безопасную среду, так как используя данную систему инженер-испытатель может сосредоточиться на функциональности устройства и быть уверенными в реальных параметрах движения ракеты. Решение разработано на платформе *LabVIEW* программного комплекса, с робастной архитектурой программного обеспечения, под названием «классический конечный автомат»

Ключевые слова: испытания, моделирование, ракеты, сбор данных, генерирование сигналов, конечный автомат, синхронизированные измерения, *LabVIEW*, ПАМ.

**IMPLEMENTACIJA UPRAVLJANJA PETOOSNIM SIMULATOROM
LETA RAKETE I CILJA U OTVORENOJ PETLJI**

Nataša D. Kljajić, Miloš I. Kostić, Miloš V. Pavić
Ministarstvo odbrane Republike Srbije, Vojnotehnički institut,
Beograd, Republika Srbija

OBLAST: elektrotehnika
VRSTA ČLANKA: stručni članak
JEZIK ČLANKA: engleski

Sažetak:

Razvoj raketnih sistema podrazumeva važne faze testiranja i simulacije leta rakete u uslovima okruženja koje diktira operacija za koju je sistem izrađen. Kako terenska ispitivanja ovako složenih sistema zahtevaju veliki utrošak finansijskih resursa, a i mnogo vremena, simulacije koje podrazumevaju hardver u petlji (*Hardware in the loop – HIL*) predstavljaju vrlo efikasno rešenje za smanjenje troškova razvoja i testiranja. U ovom radu predstavljena je implementacija upravljanja, odnosno sinhronizovanog generisanja i akvizicije podataka na petoosnom simulatoru leta rakete i cilja laboratorije za hardver u petlji (*HIL*)

laboratorije). Opisana aplikacija za simulaciju javila se kao rezultat potrebe inženjera razvoja raketa za jednostavnim načinom akviziranja informacija o odzivu realne rakete na željena i zadata upravljanja. Rešenje je realizovano u LabVIEW softverskom paketu primenom robusne arhitekture softvera, takozvane klasične mašine stanja.

Uvod

Kako su terenska ispitivanja ogroman finansijski trošak, pogotovo prilikom razvoja raketnih sistema čije poligonsko ispitivanje znači uništavanje same rakete, simulacije koje podrazumevaju hardver u petlji (*Hardware in the loop – HIL*) predstavljaju vrlo efikasno rešenje za smanjenje troškova.

HIL simulacije

Laboratorija za HIL specijalizovana je za poslove istraživanja i razvoja u oblastima simulacije sistema vođenja i upravljanja realnim hardverom i akviziciju i procesiranje realnih signala. Unutar laboratorije instaliran je petoosni simulator leta rakete i cilja S-45OR-5 sa sistemima za merenje, akviziciju i procesiranje signala. HIL simulacija omogućava razvoj pogodnih načina testiranja i evaluacije prototipova hardvera sistema vođenja i upravljanja raketom u zatvorenoj simulacionoj petlji. Simulacija kretanja rakete po visini, pravcu i valjanju obezbeđena je upravljanjem po tri ose. Spoljašnje dve ose simulatora predstavljaju azimut i elevaciju simuliranog translatornog kretanja cilja. HIL simulacija potrebna je radi određivanja neophodnih tehničkih i operacionih parametara vođenih raketa.

Pregled postavke problema i rešenje

Ispitivanje u konkretnom slučaju predstavlja zadavanje komandi simulatoru i praćenje, odnosno snimanje odziva sistema. Zadate komande predstavljaju parametre željenog leta rakete koji se zadaju u vidu tekstualnog dokumenta. Zatim se podaci obrađuju, interpoliraju oni koji nedostaju, odnosno izvršava se prilagođenje podataka traženim zahtevima sistema i samog merenja. Na kraju je potrebno snimiti odziv sistema. Neophodno je da se merenja izvršavaju u realnom vremenu, pa je zbog toga bitno obezbediti sinhronizovano izvršavanje komandi koje se šalju simulatoru i snimanje odziva, odnosno sinhronizovano generisanje izlaza i akvizicija ulaza. Za realizaciju je odabrana akvizicijska kartica National Instruments NI USB 6259, a kao softverski paket korišćen je National Instruments LabView v14.0. Akvizirani električni signali memorišu se i pakuju u format koji najviše odgovara kasnijoj obradi.

Zaključak

U radu je predstavljena implementacija savremene metode testiranja parametara rakete korišćenjem robusne arhitekture za dizajn softvera. Prednost opisane aplikacije jeste što je vrlo jednostavna za korišće-

nje. Pored toga, hardverska podrška je kompatibilna softverskom paketu koji se koristi, što sve dovodi do znatnog skraćenja vremena potrebnog za pripremu i izvršenje opita. Aplikacija je prilagođena traženim zahtevima korisnika, dakle obrađuje i čuva podatke u formatu koji je sam korisnik odredio, a zatim i vrši obrade nad podacima po zahtevanim kriterijumima. Još jedna prednost usvojenog rešenja jeste činjenica da je opisana i korišćena arhitektura dizajna softvera vrlo čitljiva i jasna, tako da se umanjuju problemi prilikom nasleđivanja softvera.

Opisano rešenje predstavlja samo početnu stavku u daljoj automatizaciji, modernizaciji i olakšavanju procesa simulacije leta rakete petoosnim simulatorom. Cilj predstavlja kompletno softversko upravljanje osama simulatora, odnosno svim njegovim senzorima i aktuatorima, sa dodatkom aplikacija po željama raznih korisnika ove laboratorije. Krajnji interes za ostvarenje zadatog cilja ove laboratorije, kao i samog Vojnotehničkog instituta kojem ova laboratorija pripada, predstavlja umanjene velikih troškova razvoja, kako raketnih sistema, tako i bespilotnih letelica, a sa druge strane i uštedu vremena potrebnog za razvoj.

Ključne reči: test, simulacije, rakete, akvizicija podataka, generisanje signala, mašina stanja, sinhronizovana merenja, LabVIEW, hardver u petlji.

Paper received on / Дата получения работы / Datum prijema članka: 12. 11. 2015.
Manuscript corrections submitted on / Дата получения исправленной версии работы / Datum dostavljanja ispravki rukopisa: 23. 04. 2016.
Paper accepted for publishing on / Дата окончательного согласования работы / Datum konačnog prihvatanja članka za objavljivanje: 25. 04. 2016.

© 2016 The Authors. Published by Vojnotehnički glasnik / Military Technical Courier (www.vtg.mod.gov.rs, втг.мо.унр.срб). This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution license (<http://creativecommons.org/licenses/by/3.0/rs/>).

© 2016 Авторы. Опубликовано в "Военно-технический вестник / Vojnotehnički glasnik / Military Technical Courier" (www.vtg.mod.gov.rs, втг.мо.унр.срб). Данная статья в открытом доступе и распространяется в соответствии с лицензией "Creative Commons" (<http://creativecommons.org/licenses/by/3.0/rs/>).

© 2016 Autori. Objavio Vojnotehnički glasnik / Military Technical Courier (www.vtg.mod.gov.rs, втг.мо.унр.срб). Ovo je članak otvorenog pristupa i distribuirano se u skladu sa Creative Commons licencom (<http://creativecommons.org/licenses/by/3.0/rs/>).

