UDK: 35.07.65.014.510.5:861.3.01
COSATI: 14-02, 17-03

An Approach to the Design and Development of a Decentralized Data Fusion Simulator – Battlefield and COMINT System Simulator

Nadica Kozić[1)]
Predrag Okiljević[1)]
Ivan Pokrajac[1)]
Danilo Obradović[2)]
Dejan Ivković[1)]

Multisensor data fusion is very important in the defence-oriented field. Data fusion can be used to integrate the individual sensor data into a common operational picture of the battlefield. A reliable forming of a common operational picture of the battlefield requires detecting, locating, identifying, classifying and monitoring dynamic entities such as radio-emitters, different platforms, weapons, and military units. In this paper, we present an approach to design and develop two simulators for a decentralized data fusion simulator – a battlefield simulator and a simulator for the communications intelligence (COMINT) system (only direction finders). Primarily, the battlefield simulator is described as a basic part for data fusion. We have developed a fully interactive, graphical user interface based scenario generation tool for creating battlefield scenarios. A target behavior is simulated in the battlefield scenario. The proposed simulator of the COMINT system generates position measurements based on the detection of electromagnetic radiation from targets. To generate position measurements, several factors such as sensitivity, observation error, target activity, methods for position determination were considered. Three methods for target position determination are implemented in the simulator. These methods are two-step positioning techniques (indirect techniques) based on the estimation of a specified parameter such as the angle of arrival (AOA).

Key words: **simulator, data fusion, decision level, sensor, direction finder, position determination, simulation of battlefield.**

Introduction

IN the defense community, decentralized data fusion is used to integrate the individual sensor data into a common operational picture of the battlefield in real-time in the area of interest (AOI). A reliable forming of a common operational picture of the battlefield requires detecting, locating, identifying, classifying and monitoring dynamic entities such as radio-emitters, different platforms, weapons, and military units. This dynamic data is used in order to form a common operational picture, but not only to display data on a map. Users usually seek to determine the relationships among entities and their relationships with the environment and higher-level enemy organizations [1]. Some of the defence applications are battlefield intelligence, surveillance and target acquisition, and strategic warning and defense. In order to design a distributed fusion-based tracking system, it is necessary to develop a decentralized data fusion simulator. This kind of simulators should provide data from different kinds of sensors based on a simulated situation in the AOI, data association and target tracking, networking, data fusion and visualization. Generally, this simulator consists of three main elements such as a battlefield simulator, a sensor simulator and a data fusion processor. To perform data fusion, the input data consists of the positional measurement and identity information from each sensor simulator such as a simulator of the COMINT system, a simulator of the optoelectronics system for surveillance, a simulator of the ground radar, and a simulator of the acoustic sensor. These simulators generate data such as positional measurement and identity information based on a simulated battlefield scenario, and this data is an input for the data fusion simulator.

To perform data fusion in a multisensor system, it is necessary to apply one of the data fusion architectures. In the literature, there are three basic data fusion architectures: sensor level fusion, feature-level fusion and decision-level fusion [2]. The sensor-level fusion assumes that each sensor detects, classifies, identifies, and estimates the tracks of potential targets before data entry into the data fusion processor. The data fusion processor combines the information from the sensors to improve the classification, identification, or state estimate of the target or object of interest. In the feature-level fusion, some target features are extracted from each sensor or sensor channel and combined into a composite feature, representative of the object in the field of view of the sensors. The decision-level fusion assumes that the results of the initial object detection and classification by the individual sensors are inputs to a fusion algorithm. The final classification occurs in the data fusion processor using an algorithm that combines the detection, classification, and position attributes of the objects located by each sensor.

This paper primarily describes the battlefield simulator as a basic part of the decentralized data fusion simulator. Also, we present an approach to the design and development of the

[1] Military Technical Institute (VTI), Ratka Resanovica 1, 11132 Belgrade, SERBIA.
[2] General Staff of Serbian Army, Department J-2, 11000 Belgrade, SERBIA.

COMINT system equipped only with direction finders (DF) for the decentralized data fusion simulator.

A typical battlefield scenario consists of ground or air targets observed by sensors on air platforms and ground platforms. Sensors are partitioned into groups called sensor networks and within each sensor network, a master sensor is chosen, using a network topology algorithm, to receive the information from other sensors in the sensor network. Optionally, a sensor can run a tracking algorithm to generate tracks for the observed targets. The master sensor provides the information about the positional measurement and some identity information. Reliable data is essential to be able to fuse sensor data in the sensor fusion system. Nevertheless, in some cases it is difficult or even impossible to build a real environment complete with deployed sensors in order to ensure reliable data for the research, development and testing of sensor data fusion systems.

The proposed simulator of the COMINT system generates position measurements based on the detection of electromagnetic radiation from targets - emitters. The behavior of each emitter is modeled and simulated in the battlefield scenario through a simulation of radio-networks. The data gathered in the COMINT system simulator is an input in the simulator of the data fusion processor. To generate position measurements, several factors such as sensitivity of DFs, observation error, target activity and the methods for position determination were considered. Three methods for target position determination are implemented in the simulator of the COMINT system. These methods belong to the two-step positioning technique (indirect technique) based on the estimation of a specified parameter such as the angle of arrival (AOA).

This paper consists of seven parts. The introduction is given in Section I. The concept of the decentralized data fusion simulator is given in Section II. The description of the battlefield simulator is given in Section III. The description of the COMINT system simulator is given in Section IV. The simulation and the results of the battlefield simulator are presented in Section V. Some simulation and results of the COMINT system simulator are given in Section VI. The conclusions are given in Section VII. Apendix A describes the basic theory of the Kalman filter tracking algorithm.

Concept of the decentralized data fusion simulator

The decentralized data fusion simulator should provide a possibility for an analysis and design of the distributed fusion based tracking system. To fulfill this task, the simulator, first of all, should provide appropriate data. Architecting the decentralized data fusion simulator is a very complex process requiring the integration of target tracking, networking, data fusion and visualization. Data from real life battle scenarios as data sources for simulation is unsuitable; therefore, it is necessary to simulate a battlefield scenario in the AOI during an observed time period [3]. However, the battlefield scenario is only one part of the data fusion simulator. In order to provide data for the process of data fusion, it is necessary to model and simulate different sensors. These sensors form useful outputs based on the knowledge of the sensor behavior and the simulated battlefield scenario. The proposed concept of the decentralized data fusion simulator is based on three main elements: battlefield simulator, sensor simulator and simulator of the data fusion processor. The proposed concept of the decentralized data fusion simulator is shown in Fig.1.

The creation of a battlefield scenario should be easy, interactive and intuitive. The user would create the scenario through the user interface**.** The generated scenario should be saved and made available for later reuse and editting. In addition to this, simulation results should be saved for performance evaluation. The simulation is closely tied with geospatial information. The terrain features and cultural artifacts such as roads, vegetation etc. influence the behavior of ground targets. Hence, it becomes imperative that a database that includes this information be integrated into the testbed to permit realistic simulation. There should be a provision for importing realistic Geographical Information System (GIS) and terrain data in testbed. The targets are ground based such as tanks, radar guns and humans. The testbed should allow the selection of these target models for scenario creation and simulation.

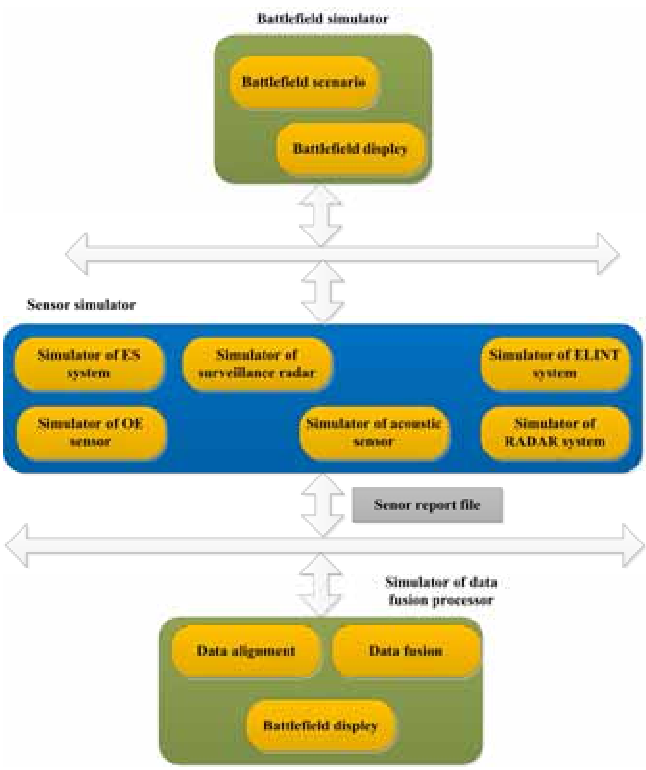

Figure 1. Concept of the decentralized data fusion simulator

In the battlefield simulator, there is a special graphical user interface (GUI) for the simulation of the insertion and placement of units of different levels on the map. When the unit is completely defined by the specified criteria, the user can input its location to the previously selected map. This is done in a way that the user enters the top left boundary of the FLOT (Forward Line of Own Troops). At this point, the general boundaries of the unit are shown on the map depending on the chosen unit type and size, type of deployment and form of combat operations.

In addition to this, there should be a means for defining a complex motion behavior for a target as well as specifying its attributes. The Kalman filter tracking algorithm was selected for all targets in the scenario (see Appendix A). The Kalman filter is a method of recursively updating an estimate of the state of a system by processing successive measurements. It takes account of target motion uncertainties. It uses state vectors, and covariance matrices to maintain bookkeeping [4].

Sensors contribute to the battlespace awareness and can be a human or a device deployed on ground- or air- based platforms. In the data fusion simulator, there are various fixed sensors as well as moving sensors.

The second part in the decentralized data fusion simulator is the sensors simulator. Ongoing efforts in the Military Technical Institute are to simulate the behavior of the set of sensors such as the COMINT system or electronic support systems, surveillance radar, acoustic sensor and optoelectronics sensors. In this paper, the simulator of the COMINT sensors is presented as well as the corresponding part of the battlefield simulator important for these sensors.

In Fig.2, the necessary inputs and functions modeled in all three parts of the decentralized data fusion simulator are listed. In order to provide the necessary data for the COMINT sensors in the battlefield simulator the activities of different radio-networks have to be simulated. The number of radio-networks, the number of participants in every radio-network, assigned frequencies, and activities of participants are defined in the battlefield simulator based on the chosen units.

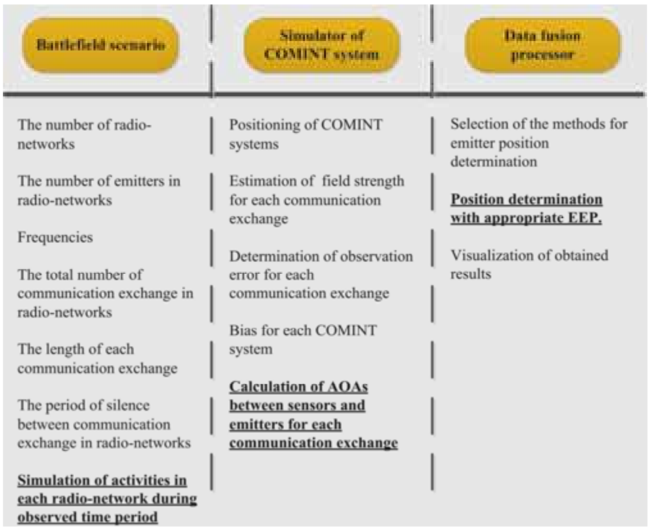

Figure 2. Necessary inputs and function modeled in all three parts of the decentralized data fusion simulator.

This data is automatically generated in the battlefield simulator for the chosen scenario. In the battlefield scenario, it is assumed that each platform is equipped with at least one radio, and this radio is a part of one or more radio-networks depending on the platform's role. For example, if a platform presents a command station, it is necessary to provide communication with the higher command and with subordinate elements.

Also, there can be more than one radio-network depending on the unit size and type. Every radio-network has its own VHF frequency which cannot be repeated in any other.

Depending on the number of the entered units in the battlefield simulator, a corresponding data matrix is automatically generated and it contains the following data: the locations (x and y coordinates) of every possible participant (platform) in radio-networks as well as the frequencies of every radio-network.

For the purpose of modeling the behavior of the radio-networks, a particular communication structure has been simulated. For each radio-network the total number of communication exchange is defined and emitters are randomly chosen. The length of each communication exchange in the radio-network is defined as the Gaussian random amount of time with a selectable maximum value. Transmissions are followed by a period of silence. The duration of a period of silence is defined as the Gaussian random amount of time with a selectable maximum value.

Battlefield Simulator

The battlefield simulator defines the activities and the positions of ground targets, meteorological conditions and saves these activities into the scenario file.

In order to develop a battlefield simulator, it is necessary to known its function. Our intention was to develop the battlefield simulator that has to provide data for multisensor data fusion. Before we start to develop a battlefield simulator, it is necessary to specify the possible sensors: ES (Electronic Support) systems, Optoelectronic sensors, ground radar, acoustic based sensors. Each of these sensors detects targets in a different way. Using the ES system it is possible to detect and estimate a position based on emitters installed on targets. Optoelectronic sensors detect targets in the visual and the infrared band of the EM (Electromagnetic) spectrum. Ground radars detect targets based on moving targets. Acoustic sensors are based on acoustic signals generated by targets.

Our application has been developed using Matlab Grafical User Interface (GUI). Using Graphical User Interface allows an interactive creation of a scenario and the generation of the ground truth. The user has the capability to edit scenario entities while creating a scenario; this adds an additional flexibility to the scenario creation process.

In Matlab GUI, the user has the option to import a realistic GIS map with all terrain features and cultural artifacts. The user can chose just one region which he wants to observe. That region is then graphically displayed in one part of GUI with the 2D axes. On the map, there are also shown terrain heights which can be read from the map. This map is used further in the simulation.

After loading the map in GUI, the user must enter the data about the units. For that purpose, there is another GUI for the input of units. The following functions have been implemented in that GUI:

- loading the map from a selected folder,
- selection of the unit size (section, platoon or company),
- selection of the form of combat operations (offensive or defensive),
- selection of the types of unit deployment (formation) between a single line where the subordinate units are arranged in one row or multiple lines where the subordinate units are arranged in two rows,
- selection of the type of the unit (infantry, tank, artillery or mechanized).

The user can also choose between the platforms which carry the radars and the platforms which carry artillery. The platforms which carry the radar can move through the terrain and stand at one place and the platforms which carry artillery can also move through the terrain and stand at one place but they can shoot while standing.

The targets, in the beginning, have their positions expressed in the x and y coordinates and they have their velocities and accelerations. In every scan simulator, t the next position of every target is calculated, based on the current positions and current velocities and accelerations of targets with the Kalman filter tracking algorithm.

The Kalman filter maintains information about the target. An example could be the determination of the next position, using the current position and the rates of change (velocity, acceleration, etc.). This target data is stored in a state vector. In addition, Kalman filters remember how ”good” these determinations were. These uncertainties are stored in the covariance matrix.

As time progresses, a target moves and its rates can also change. These changes in a target's state over time have influences on the next position of target. The covariance matrix is extrapolated and a propagation matrix is used to extrapolate the estimate and the covariance matrix.

Simulator of the COMINT system

The main purpose of the COMINT system simulator is to provide an azimuth estimation for each active radio-emitter in the selected observation time period. This can be done by modeling signals from each emitter at each antenna array of DFs and after that estimating the direction of arrival (DOA) using some of the well-known techniques. Also, azimuth estimation could be performed by calculating the exact bearings between the position of the antenna array and the targets in a defined time instance and adding errors into the DOA estimation. The simulation model includes Gaussian random errors of 0° average and the root mean square (RMS) error added to the azimuth. This error includes DF accuracy, bias and GPS error in determination position of the COMINT system. Bias and GPS errors are assumed to be constant for a defined position of the COMINT system. However, DFs accuracy depends on the signal field strength.

The signal field strength depends on many factors. In order to provide this information in the COMINT system simulator, the procedure for the prediction of the signal field strength estimation is implemented based on ITU Recommendation ITU-R P.1546-1 for the point to area prediction for terrestrial services in the frequency range of 30-3000 MHz. To estimate the signal field strength, it is necessary to know the position of an emitter (in our case a platform) and the position of the receiving antenna (the COMINT sensor). In addition, it is necessary to know frequency, antenna gain, effective radiated power and terrain information. All this data is defined in the battlefield simulator for each radio-network. Based on this data, and the known position of the COMINT sensor, it is possible to predict the signal field strength E (dB(μV/m)) and the basic transmission loss L (dB) during each transmission.

Based on the estimated the signal field strength from a corresponding look-up table, is possible to read the root mean square (RMS) error. In the order to estimate the emitter position in the COMINT sensor simulator, three different methods are implemented. These methods belong to the two-step positioning technique (indirect technique) based on the estimation of a specified parameter such as the angle of arrival (AOA). The first method is the Discrete Probability Density (DPD) method. The DPD method combines sensor measurements taken from different locations for the spatial location determination. This method is based on the assumption that the sensor measurement can be modeled by some probability density function over the range of possible values. The second method is the fusion of multiple bearing lines (FMBL) method. This method is based on the aggregation of multiple fixes obtained through the fusion of two bearing lines (AOAs). The third method is the mean-squared distance algorithm (MSDE) based on minimizing the square of the miss distance of the best point estimate from the measured lines of the position. These methods are described in [5].

Simulation results of the battlefield simulator

The positions of the targets are shown in the map which is import to the GUI. The user is able to see how targets are moving through the map. Every target is presented with another colour so it is easy to separate targets on the map.

In the GUI, the actions of the platforms which carry the radars are presented as well as those of the platforms which carry the artillery. These actions are shown in two graphics: one graphic shows the platforms carrying the radars and another graphic shows the platforms carrying the artillery. The user is able to see in every scan if the platforms are moving or standing. Also, when the platform carries artillery, it is shown if there was a shooting.

This is all shown in the following figures.

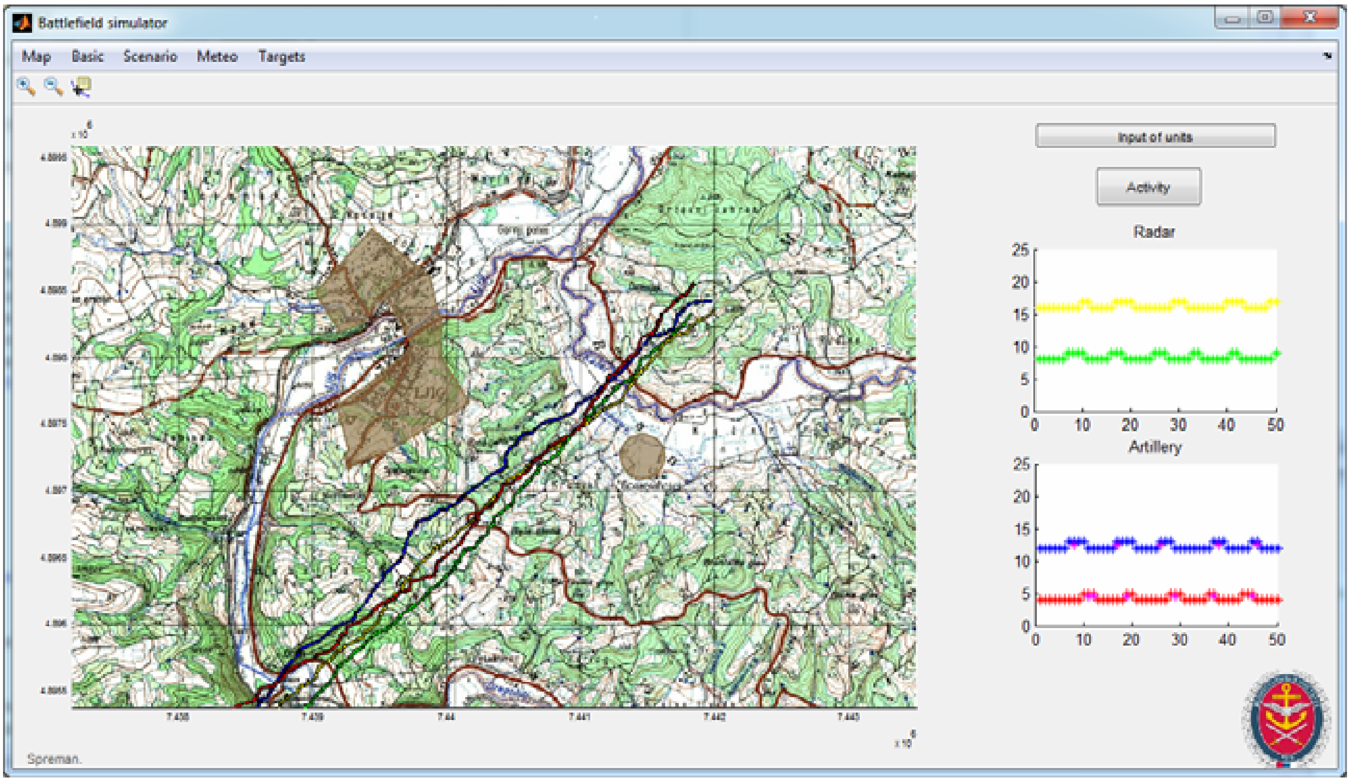

Figure 3. Battlefield simulator

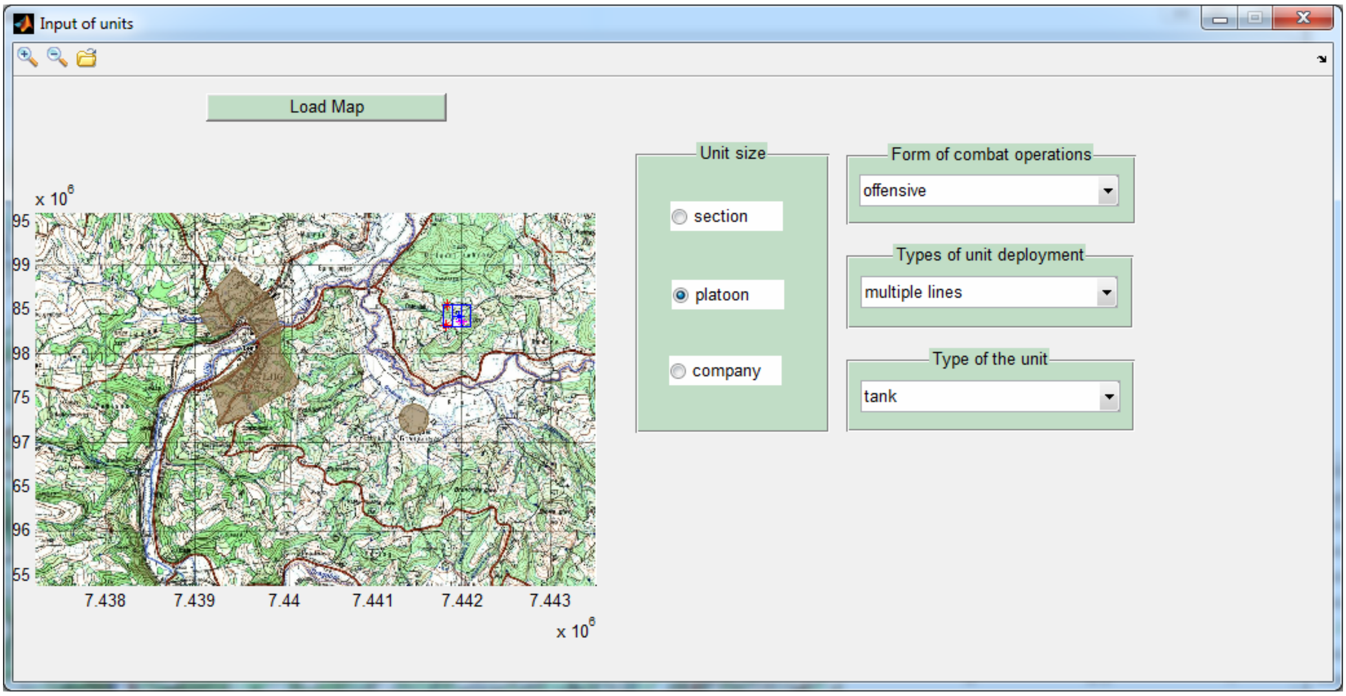

Figure 4. GUI representation of the input of units

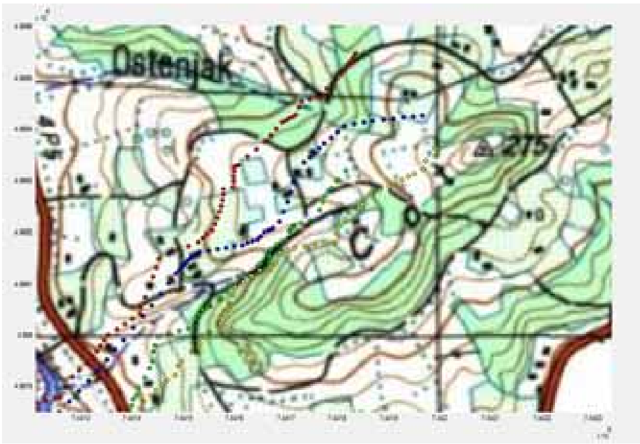

Figure 5. Targets moving through the map

In Fig.3, a representation of the GUI in Matlab after a completed simulation is shown. It could be concluded that there are four targets, ground based. Their trajectories, which show how targets are moving through the map, are represented with different colours, Fig.5.

A special GUI for the input of units is run from the GUI for the battlefield simulator and is shown in Fig.4.

Referring to the Fig.3, we can analyze the activities of the platforms which carry the radars during the observed time interval. These activities are shown on the graphic under the title "Radar". The activities are shown with stars where a lower value means that the platform is moving and a greater value means that the platform is standing at one place. We limit the time during which the platform can move and also we limit the time during which platform can stay at one place.

In the same figure, we analyze the activities of the platforms which carry the artillery. They are shown on the graphic under the title "Artillery". These platforms also have the ability to move through the terrain or stand at one spot. This is shown with stars where a lower value means that the target is moving and a greater value means that the platform is standing at one place. Also, in this graphic it is shown if the platforms which carry the artillery are shooting while in position of standing at one place. This is shown with violet stars. Similarly to the case with the platforms which carry the radars, we limit the time during which the platform can move and also we limit the time during which the platform can stay at one place.

Simulation results of the COMINT system simulator

This COMINT simulator was also implemented in the MATLAB software packet. In the simulated scenario, it is assumed that there are three units (two sections and one platoon) in the selected AOI. In this scenario, there are seven ground based platforms-targets and each target is equipped with radio-emitters. The platforms were placed in a scenario with their motion without constrains.

The position of the COMINT systems and the movement of the targets are shown in Fig. 6. All targets were assumed to have an effective radiated power of 50 W with vertical polarization and the transmitter antenna heights were 5m. The signal losses and the signal field strength are the estimated base on ITU Recommendation ITU-R P.1546-1.

The test scenario created in the battlefield scenario simulator was opened in the COMINT system simulator. In this scenario, four COMINT systems are deployed. The fusion of the multiple bearing lines method is used in order to determine the position of targets. Fig.7 and Fig.8 show the screenshots of the simulation window.

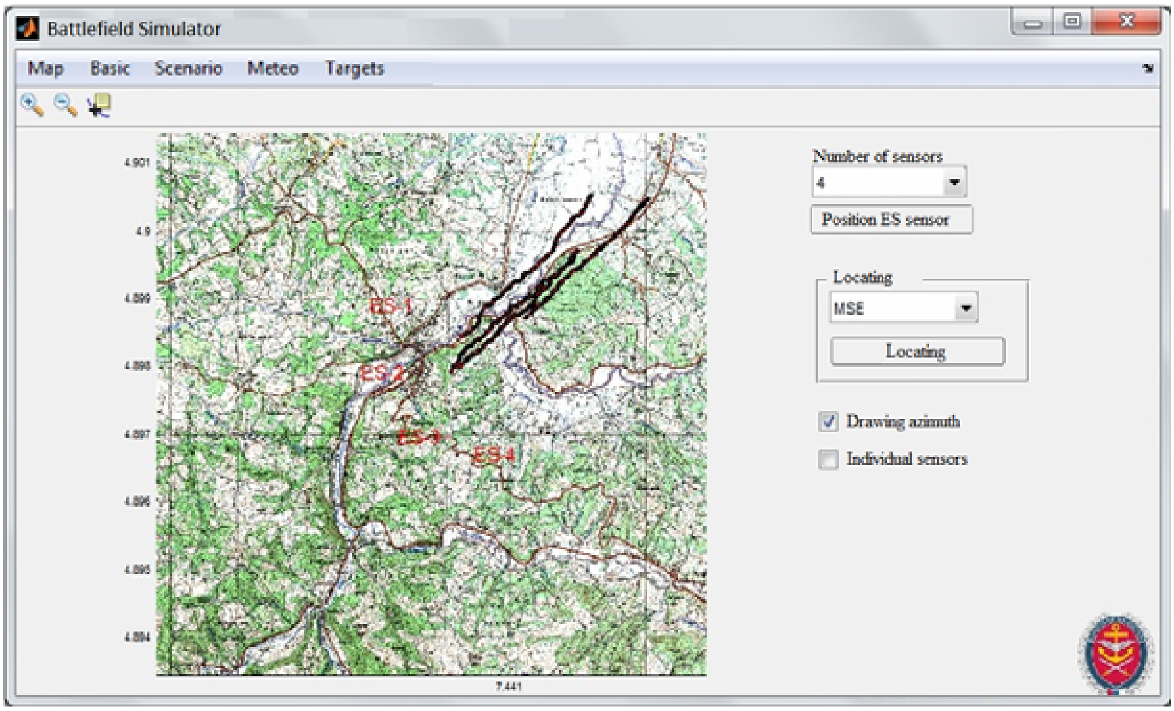

Figure 6. Graphical user interface for the COMINT system simulator

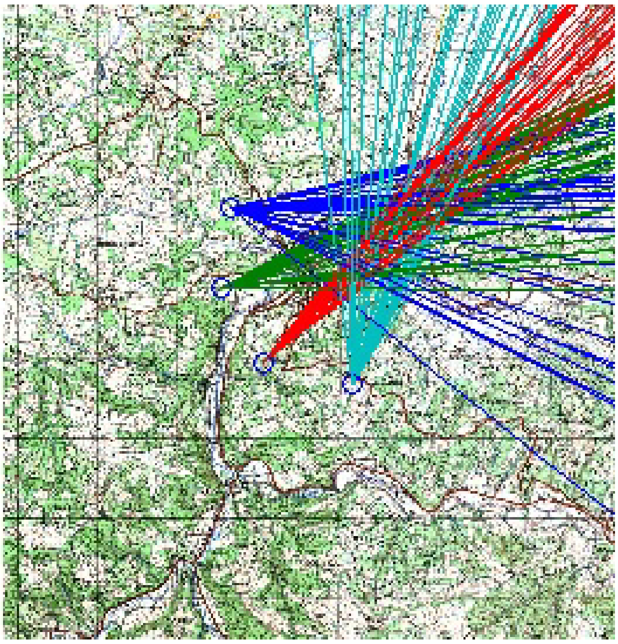

Figure 7. COMINT system simulator window showing the estimated azimuths from each sensor

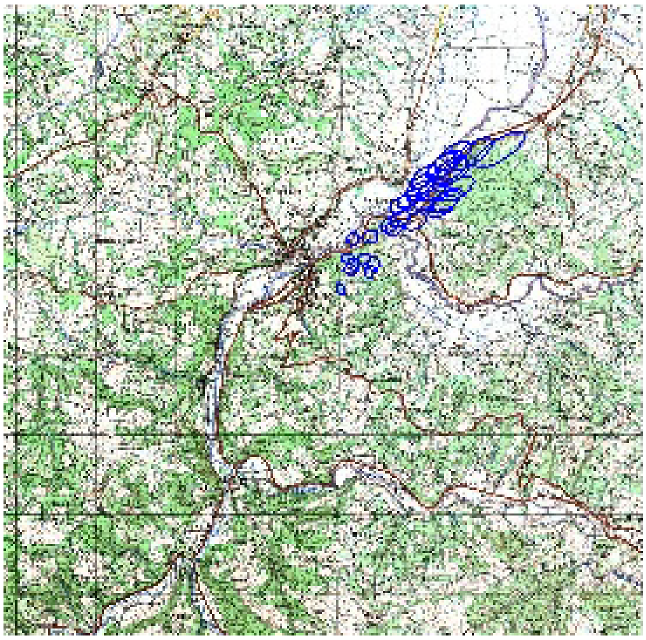

Figure 8. COMINT system simulator window showing the determined positions placed in the centers of the ellipses that are sensors covariance

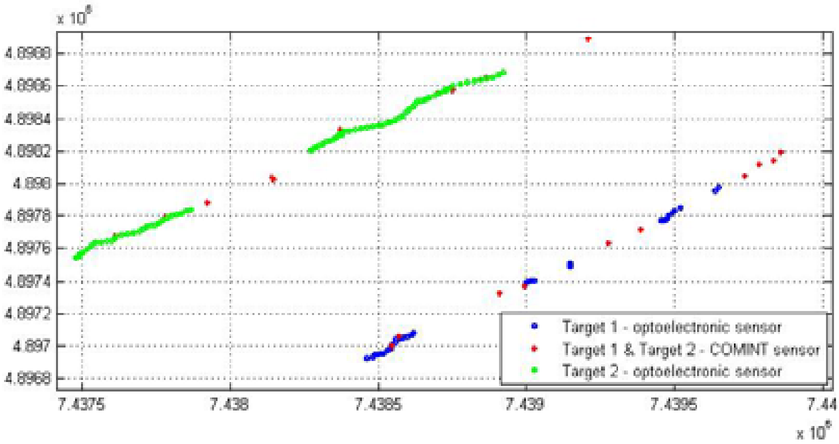

Figure 9. Visualization of the COMINT sensors and the optoelectronics sensors in the case when there are two targets

Fig.9 presents the results of the COMINT sensors and the optoelectronics sensors in the case when there are two targets.

Conclusion

In the defense community, it is very important to form a reliable common operational picture of the battlefield. In this paper, we presented the establishment of a fully interactive, graphical user interface of the battlefield simulator and the COMINT system simulator, as a basic part for decentralized data fusion. The use of these simulators enables the testing of different techniques and methods for a decision-level fusion approach.

Data from the COMINT system simulator and data from other sensors should provide multisensor ground target tracking. The fusion of the obtained results from different results and target tracking will be the subject of further research.

The modular concept of simulators provides a possibility to upgrade these simulators with some features. Developed simulators provide realistic data for a decentralized data fusion simulator intended to be used by the Network Centric Warfare application for developing a Common Operational Picture.

References

[1] HALL,D.L., LIGGINS,M.E., LLINAS,J.: *Handbook of multisensor data fusion : theory and practice* (2nd ed). CRC Press, Boca Raton, FL, 2009.

[2] KLEIN,L.A.: *Sensor and data fusion: a tool for information assessment and decision making*., Bellingham^ eWA WA: SPIE press, 2004, Vol.324.

[3] CHANDRESH,M.: GOVINDARAJAN,S., THENKURUSSI,K.: *An approach to design and development of decentralized data fusion simulator*, Virtual Reality Lab 809 Furnas Hall SUNY at Buffalo Buffalo, NY 14260, U.S.A.

[4] EISERLOH,P.P.: *An Introduction to Kalman Filters and Applications,* Data Systems Branch, Code 525300D Electronic Combat Range Naval Air Warfare Center China Lake, CA 93555

[5] POKRAJAC,I., OKILJEVIC,P., DESIMIR,V.: *Fusion of Multiple Estimation of Emitter Positions*, Scientific Technical Review, ISSN 1820-0206, 2012, Vol.62, No.3-4, pp.55-61.

[6] WELCH,G., BISHOP,G.: *An Introduction to the Kalman Filter*, Department of Computer Science University of North Carolina at Chapell Hill, Chapel Hill, NC 27599-3175.

Apendix A

Kalman filter

The Kalman filter is a method of recursively updating an estimate of the state of a system by processing successive measurements. It takes account of target motion uncertainties. It uses state vectors, and covariance matrices to maintain bookkeeping [4,6].

The five basic steps of Kalman filters are:
- extrapolate target positions and rates
- extrapolates error estimates
- calculate gain
- update state vector estimates
- update the error estimates for new data.

This forms a cycle which just keeps repeating for every new measurement.

A) Extrapolating the Target and Uncertainties

The Kalman filter maintains the information about the target. An example could be estimates about its position, and rates of change (velocity, acceleration, etc.). This target data is stored in a state vector. In addition, Kalman filters remember how ”good” these estimates were. These uncertainties are stored in a covariance matrix.

As time progresses, a target moves and its rates can also change. These changes in a target's state over time are handled by extrapolating the target estimate by using the rates, and the amount of time since the filter was last updated. The covariance matrix is also extrapolated, because our uncertainty about the target increases with time and without any new measurements. A propagation matrix is used to extrapolate the estimate and covariance matrix.

The estimates of the target position and rates are kept in state vectors. The state vector usually contains the first derivative of all the components from the measurement vector. This is usually one more component that the sensor can measure. For example, if the sensor measures only x then the state vector should contain both x and v (the velocity, or rate of x 's change). The standard notation for the state vector is X .

To predict (or extrapolate) the state of a target, the state vector is propagated.

$$\begin{bmatrix} x \\ v \end{bmatrix} \rightarrow \begin{bmatrix} x + v\Delta t \\ v \end{bmatrix} \tag{1}$$

This can be treated as a matrix multiplication:

$$\begin{bmatrix} 1 & \Delta t \\ 0 & 1 \end{bmatrix}\begin{bmatrix} x \\ v \end{bmatrix} = \begin{bmatrix} x + v\Delta t \\ v \end{bmatrix} \tag{2}$$

The time propagator Φ will need to be tailored to the specific state vector. For our example, here Φ is:

$$\Phi(\Delta t) = \begin{bmatrix} 1 & \Delta t \\ 0 & 1 \end{bmatrix} \tag{3}$$

Giving the extrapolated state vector:

$$X^- = \Phi(\Delta t) X(t) \tag{4}$$

The propagation operation must be linear in state components! The components of the state vector appearing within the time propagation matrix must be only a simple multiplier, no trig functions, or even powers of the components. Some examples of propagation matrices for some of more complex state vectors will be presented here.

Here is a state vector with the measurement x , the velocity v , and the acceleration a :

$$\begin{bmatrix} x \\ v \\ a \end{bmatrix} \rightarrow \begin{bmatrix} x + v\Delta t + \frac{1}{2}a\Delta t^2 \\ v + a\Delta t \\ a \end{bmatrix} \tag{5}$$

It gives a propagation matrix:

$$\Phi(\Delta t) \rightarrow \begin{bmatrix} 1 & \Delta t & \frac{1}{2}\Delta t^2 \\ 0 & 1 & \Delta t \\ 0 & 0 & 1 \end{bmatrix} \tag{6}$$

The error estimates (uncertainties) are kept in a covariance matrix, which in a standard notation is P . These error estimates contain the history of the amount each measurement differs from the estimate in the state vector.

The uncertainties in the form of the covariance matrix are extrapolated by using the time propagation matrix developed earlier. The covariance matrix is extrapolated using:

$$P^{-} = \Phi P_{old}^{+} \Phi^{T} + Q \tag{7}$$

P^{-} is the extrapolated, but not yet updated covariance matrix and P_{old}^{+} is the updated covariance matrix from the previous filter cycle.

Q is the maneuver noise matrix (filter tuning parameter):

$$Q = \Phi Q_0 \Phi^{T} \tag{8}$$

$$Q_0 = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & \delta_a^2 \end{bmatrix} \tag{9}$$

where δ_a is a standard deviation during the extrapolation period.

B) Measurement error

The measurements from the sensor usually contain less information than the state vector. For example, a sensor may measure the position of an object, whereas the state vector has both the filtered position and its velocity. So we need to store the measurement in a smaller vector than the state vector. This measurement vector is named Z. Each measurement has some random error. Since the number of components being measured is usually lower than is kept in the state vector, it is not possible to simply use a matrix of the same size as the covariance matrix. The measurement error matrix R contains the variances and the covariances of the errors due to each measurement. This is a square matrix with the same dimension as the number of rows of the measurement Z. Determining R requires prior knowledge of the system being filtered by the filter designer. The filter designer usually hard codes R with the values specific to the sensor being filtered.

The measurement error (or noise) matrix R is related to the covariance matrix. R is in the measurement system, and contains the estimate of the errors for each measurement in Z. The covariance matrix P is in the state vector system, and contains the errors for the filter estimates in the state vector X. When a new measurement Z arrives, the covariance will be updated using the Kalman gains K, and the measurement error R.

C) Calculating the gain

In order to update the state vector, it is important to apply a weight (or gain) to the innovation.

The innovation gives the difference between the estimate and the new measurement. Each filter cycle, a new innovation N_i, is calculated. When measurements (the vector Z) do not contain the same components as those contained in the state vector X, an auxiliary measurement matrix H is required to transform one to the other. This transforms the state vector X into the measurement system Z:

$$Z = HX \tag{10}$$

Now, the inovation is:

$$N = Z - HX^{-} \tag{11}$$

The gain indicates how much of the innovation should be applied to the estimate. The gain is stored in the matrix K with the same number of elements as the state vector.

The gain components usually range from zero to one. In a system with incomplete state measurements, the components which are not measured may be larger than one during the first few filter cycles. This is dependent upon the initial values of the state vector.

The gain is calculated for each non-extrapolated filter cycle. The closer the new measurement agrees with the current estimate, the more confident we are about this measurement. This shows up in the covariance matrix P.

The optimal gain is:

$$K = P^{-} H^{T} (HP^{-} H^{T} + R)^{-1} \tag{12}$$

where H is the auxiliary measurement matrix.

D) Updating the State Vector and the Covariance Matrix with the New Measurement

When a new measurement arrives, and passes the optional residual check, then the next step is to update the filter's estimate with the new data. The current estimate is taken and converted into the measurement system. This is subtracted from the new measurement to form the innovation (or residual).

The state vector is updated by applying the gains to the innovation, and combining it to the extrapolated state vector:

$$X^{+} = X^{-} + K(Z - HX^{-}) \tag{13}$$

When the state vector is updated with a new measurement, it is also required to update the covariance matrix. This is only performed when the new measurement passes the residual check if one was used. The simplest form for the updated covariance matrix is:

$$P^{+} = (I - KH)P^{-} \tag{14}$$

where I is the identity matrix.

Received:22.07.2014.

Projektovanje i razvoj simulatora za decentralizovanu fuziju podataka – Simulator bojišta i simulator COMINT sistema

Fuzija podataka u više-senzorskim sistemima je veoma važna u odbrambenim sistemima. Fuzija podataka se koristi za objedinjavanje podataka dobijenih od pojedinačnih senzora u združenu operativnu sliku bojišta. Pouzdano formiranje združene operativne slike bojišta zahteva detekciju, lociranje, identifikaciju, klasifikaciju i praćenje dinamičkih objekata kao što su radio-predajnici, različite platforme, oružje i vojne jedinice. U ovom radu, predstavljamo pristup dizajnu i razvoju dva simulatora za simulator za decentralizovanu fuziju podataka – simulator bojišta i simulator COMINT sistema (samo radio-goniometri). Pre svega je opisan simulator bojišta kao osnovni deo fuzije podataka. Razvili smo potpuno interaktivan grafički

korisnički interfejs baziran na generisanju scenarija za formiranje scenarija bojišta. U simulatoru bojišta je simulirano ponašanje ciljeva. Predloženi simulator COMINT sistema generiše pozicije na osnovu detekcije elektromagnetnog zračenja ciljeva. Prilikom generisanja pozicije u obzir je uzeto nekoliko faktora kao što su osetljivost, greška posmatranja, aktivnost cilja, metode za određivanje pozicije. U simulatoru su implementirane tri metode za određivanje pozicije cilja. Ove metode predstavljaju tehnike određivanja pozicije u dva koraka, zasnovane na proceni posebnog parametra kao što je ugao dolaska radio-talasa (angle of arrival - AOA).

Ključne reči: **simulator, fuzija podataka, nivo odlučivanja, senzor, radio-goniometar, određivanje položaja, simulacija bojišta.**

Проектирование и разработка тренажёров для децентрализованного объединения данных - тренажёр поля битвы и тренажёр COMINT для системы радиоразведки

Фьюжн (слияние) данных в многодатчиковых системах очень важно в системах обороны. Фьюжн данных используется для сбора и объединения данных, полученных от отдельных датчиков, в совместную оперативную картину поля битвы. Надёжное формирование совместной оперативной картины поля битвы требует обнаружение, размещение, идентификацию, классификацию и отслеживание динамических объектов, таких как радиопередатчики, различные платформы, оружия и воинские части. В этой статье мы представляем подход к проектированию и разработке двух тренажёров в роли тренажёра для децентрализованного объединения данных – тренажёр поля битвы и тренажёр COMINT для системы радиоразведки (только радиопеленгаторы). Прежде всего здесь описывается симулятор поля битвы как фундаментальная часть слияния данных. Мы разработали полностью интерактивный графический пользовательский интерфейс, основанный на генерирующих сценариях для формирования сценариев поля битвы. В симуляторе поля битвы моделируются поведения целей. Предлагаемый тренажёр COMINT системы генерирует позиции на основании обнаружения электромагнитного излучения целей. При создании положения во внимание было принято несколько факторов, таких как чувствительность, погрешность наблюдения, деятельность цели, методы определения положения. Симулятор реализует три метода для определения положения цели. Эти методы являются методами для определения положения в двух этапах, на основе оценки определённых параметров, таких как угол прихода радиоволн (angle of arrival - AOA).

Ключевые слова: **тренажёр, фьюжн (слияние) данных, уровень принятия решений, датчик, радиопеленгатор, позиционирование положения, моделирование поля битвы.**

Conception et développement du simulateur pour la fusion décentralisée des données – Simulateur du champ de bataille et simulateur du système COMINT

La fusion des données dans les systèmes multi capteurs est très importante dans les systèmes de défense. La fusion des données est utilisée pour intégrer les données obtenues par les capteurs individuels dans l'image commune opérationnelle du champ de bataille. La formation fiable de l'image commune opérationnelle du champ de bataille exige détection, location, identification, classification et suivi des objets dynamiques tels que émetteurs radio, différentes plateformes, armes et unités militaires. Dans ce papier on présente une approche du dessin et du développement de deux simulateurs pour le simulateur destiné à la fusion décentralisée des données – simulateur de champ de bataille et simulateur du système COMINT (goniomètres radio seulement) . On a décrit d'abord le simulateur de champ de bataille comme la partie basique de la fusion des données. On a développé un interface graphiquement complètement interactif d'utilisateur et il est basé sur la création du scénario pour la formation du scénario de champ de bataille. Le comportement des cibles a été simulé par le simulateur de champ de bataille. Le simulateur proposé du système COMINT génère les positions à partir de la détection de la radiation électromagnétique des cibles. Pendant la création des positions on a considéré plusieurs facteurs tels que : sensitivité, erreur d'observation, activité de cible, méthodes pour la détermination des positions. Dans le simulateur trois méthodes ont été installées pour la détermination de positions de cible. Ces méthodes représentent les techniques de détermination à deux pas, basées sur l'estimation du paramètre spécifié tel que l'angle d'arrivée de l'onde radio (angle of arrival – AOA).

Mots clés: **simulateur, fusion des données, niveau de décision, capteur, goniomètre radio, détermination de position, simulation de champ de bataille,**