Proceedings of the 2007 IEEE Symposium on Computational Intelligence in Security and Defense Applications (CISDA 2007)

The Automation of Combat Decision Processes in the Simulation Based Operational Training Support System

Andrzej Najgebauer, Ryszard Antkiewicz, Zbigniew Tarapata, Jarosław Rulka,
Wojciech Kulas, Dariusz Pierzchała, Roman Wantoch-Rekowski
Military University of Technology
Cybernetics Faculty
Kaliskiego 2
00-908 Warsaw, Poland
e-mail: anajgebauer@wat.edu.pl, rantkiewicz@wat.edu.pl, ztarapata@wat.edu.pl, jrulka@wat.edu.pl,
wkulas@wat.edu.pl, dpierzchala@wat.edu.pl, rwantoch-rekowski@wat.edu.pl

***Abstract*- The interactive simulation environment for training (and/or analysis) of military operations is presented as the Simulation Based Operational Training Support System (SBOTSS). The system was constructed in order to provide cost-effective approach of Computer Assisted Exercises and it is an integrated, interactive, many-sided land, analysis and training support model based on particular components equipped with combat, logistics, engineering, electronic warfare and intelligence functions. The idea and model of command and control process applied for the decision automata at the tactical level are presented. The automata executes the two main processes: decision planning process and direct combat control. The decision planning process relating to the automata contains three stages: the identification of a decision situation, the generation of decision variants (course of actions) the variants evaluation and nomination the best variant of these, which satisfy the proposed criteria. The particular approach to identification of decision situation and variants of action are presented. The procedure of variants generation based on some kind of pre-simulation process contains the evaluation module, which allows us the best choice of action plan according to specified criteria. The direct combat control process contains such phases, like command, reporting and reaction to fault situations. Some results of the simulation process including the decisions made by automata is considered. The calibration process on the basis of battle scenarios is described and presented.**

I. INTRODUCTION

The idea of military unit structure used in SBOTSS [12] assumes, that physically simulated objects in the structure (for example brigade) are command posts, automatic commanders and basic units (in the SBOTSS basic units means company, battery or logistics platoon). The source of effectiveness is in the limitation of staff personnel, required in the CAXs and replacement the staff by "automatic commanders" [3].

There are papers which deal with automation of decision processes [13], [14]. In our proposition the decision situation is classified according to the following factors: own task, expected actions of opposite forces, environmental conditions – terrain, weather, the day and year season, current state of own and opposite forces in a sense of personnel, weapon systems and military materiel. For each class of decision situation there are generated the set of action plan templates for subordinate and support forces. For example the proposed action plan contains: forces redeployment, regions of attack or defence, or manoeuvre routes, intensity of fire for different weapon systems, terms of supply of military materiel combat forces by logistics units. In order to generate and evaluate possible variants we use the pre-simulation process based on some procedures – forces attrition procedure, slowing down rate of attack procedure, utilization of munitions and petrol procedure.

We consider the following criteria in the evaluation process: time and degree of task realization, own losses, utilization of munitions and petrol. The idea of decision generation using 3-stage algorithm was presented in [1], [3]. The presented paper is the continuation of the approach and its extension in the area of pattern matching, course of action and the calibration of the model.

II. MODELLING OF DECISION SITUATION

The model of decision situation concerns the first two steps (elliptical line) in the Fig. 1. We define decision situations space as follows:

$$DSS = \{SD:\ SD = (\ SD_r\ )_{r=1,..,8}\} \qquad (1)$$

The vector SD represents decision situation which is described by the following eight elements: SD_1 - commanding level of opposite forces, SD_2 - type of task of opposite forces (e.g. attack, defence), SD_3 - commanding level of own forces, SD_4 - type of task of own forces (e.g. attack, defence), SD_5 - net of squares as a model of activities (interest) area $SD_5 = \left[SD^5_{ij} \right]_{\substack{i=1,..,SD_7 \\ j=1,..,SD_8}}$, $SD^5_{ij} = (SD^{5,k}_{ij})_{k=1,..,7}$. The terrain square with the indices (i,j) each of elements denotes: $SD^{5,1}_{ij}$ - the degree of the terrain passability, $SD^{5,2}_{ij}$ - the degree of topographic terrain configuration, $SD^{5,3}_{ij}$ - the degree of terrain

1-4244-0700-1/07/$20.00 ©2007 IEEE

growth, $SD_{ij}^{5,4}$ - armoured power (potential) of opposite units deployed in the square, $SD_{ij}^{5,5}$ - infantry power (potential) of opposite units deployed in the square, $SD_{ij}^{5,6}$ - artillery power (potential) of opposite units deployed in the square, $SD_{ij}^{5,7}$ - coordinates of the square (i,j), SD_6 - the description of own forces: $SD_6 = \left(SD_i^6\right)_{i=1,..,4}$, SD_1^6 - total armoured power (potential) of own units, SD_2^6 - total infantry power (potential) of own units, SD_3^6 - total artillery power (potential) of own units, SD_4^6 - total air fire support power (potential); SD_7 - the width of an activities (interest) area (number of squares), SD_8- the depth of an activities (interest) area (number of squares).

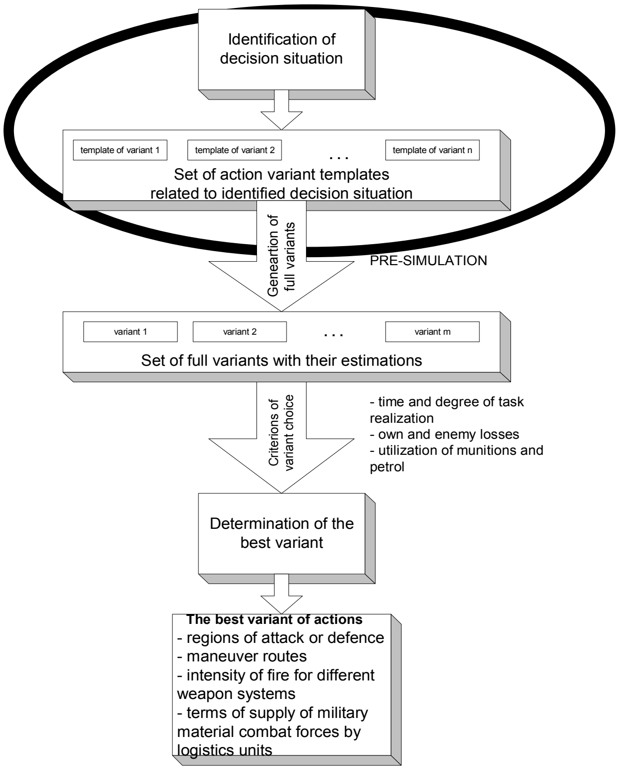

Fig. 1. Algorithm for selecting the best variant of action [1]

III. DECISION SITUATION PATTERN MATCHING PROBLEM

We have the set of decision situations patterns: $PDSS = \{PS : PS \in DSS\}$. For current decision situation we have to find the most similar situation from the set of patterns. Using the similarity measure function (4) we can evaluate distances between two different decision situations especially the current and the pattern. There are several methods of finding the most matched pattern situation to the current, which can be used. We propose two main approaches deal with following measures: distance vectors measure, weighted graphs similarity measure.

We determine the subset of decision situation patterns $PDSS_{CS}$ which are generally similar to the current situation considering such elements like: task type, command level of own and opposite units and own units potential:

$$PDSS_{CS} = \{PS = (PS_i)_{i=1,..,6} \in PDSS : PS_i = CS_i,\ i = 1,..,4, dist_{potwl}(CS, PS) \le \Delta Pot\} \tag{2}$$

where

$$dist_{potwl}(CS, PS) = \max\{\left|CS_k^6 - PS_k^6\right|, k = 1,..4\}$$

ΔPot - the maximum difference of own forces potential.

Distance vectors approach

Then we formulate and solve the multicriteria optimization problem which allow us to determine the most matched pattern situation from the point of view of terrain and military power characteristics:

$$Z = \left(PDSS_{CS}, F_{CS}, R_D\right) \tag{3}$$

$$F_{CS} : PDSS_{CS} \to R^2 \tag{4}$$

$$F_{CS}\left(PS\right) = (dist_{ter}(CS, PS), dist_{pot}(CS, PS))$$

$$dist_{ter}(CS, PS) = \sum_{k=1}^{3} \lambda_k \cdot \left(\sum_{i=1}^{I} \sum_{j=1}^{J} \left(CS_{ij}^{5,k} - PS_{ij}^{5,k} \right)^p \right)^{\frac{1}{p}}$$

$$\sum_{k=1}^{3} \lambda_k = 1, \lambda_k > 0, k = 1,..,3$$

$$dist_{pot}(CS, PS) = \sum_{k=4}^{6} \mu_k \cdot \left(\sum_{i=1}^{I} \sum_{j=1}^{J} \left(CS_{ij}^{5,k} - PS_{ij}^{5,k} \right)^p \right)^{\frac{1}{p}}$$

$$\sum_{k=4}^{6} \mu_k = 1, \mu_k > 0, k = 4,..,6$$

$$I = \min\{CS_7, PS_7\},\ J = \min\{CS_8, PS_8\}$$

$$R_D = \left\{ \begin{array}{l} (Y,Z) \in PDSS_{CS} \times PDSS_{CS} : \\ dist_{ter}(CS,Y) \le dist_{ter}(CS,Z) \wedge \\ dist_{pot}(CS,Y) \le dist_{pot}(CS,Z) \end{array} \right\} \tag{5}$$

For the hypothetical decision situations (*CS*-current, *PS*-pattern) presented in the Fig.2 the most matched pattern decision situation to current situation *CS* using method presented above is PS_2.

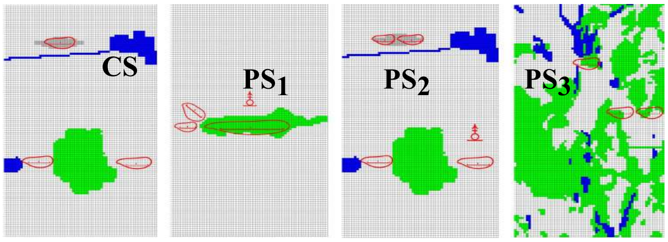

Fig.2. Hypothetical current situation *CS* and pattern situations (PS_1, PS_2, PS_3)

Weighted graph similarity approach

In the literature there are several methods for determining graphs similarity (based on: graphs isomorphism and homeomorphism [4], adjacency matrices similarity [5]). In our proposition the graphs similarity approach for identification of decision situation was presented in details in [15] and it consists of three stages:

1. Building weighted graphs $WGT(CS)$, $WGD(CS)$ and $WGT(PS)$, $WGD(PS)$ representing decision situations: current (CS) and pattern (PS) for topographical conditions (WGT) and units (potential) deploying (WGD);
2. Calculation of similarity measures between pairs: $WGT(CS)$, $WGT(PS)$ and $WGD(CS)$, $WGD(PS)$ for each $PS \in PDSS$;
3. Selecting the most similar PS to CS using calculated similarity measures.

Stage 1

The first stage is to build weighted graphs WGT and WGD as follows:

$$WGT = \left\langle \begin{matrix} GT = \langle N_{GT}, A_{GT} \rangle, \\ \{f_k^T(n)\}_{\substack{k \in \{1,\dots,5\} \\ n \in N_{GT}}} \end{matrix} \right\rangle, WGD = \left\langle \begin{matrix} GD = \langle N_{GD}, A_{GD} \rangle, \\ \{f_k^D(n)\}_{\substack{k \in \{1,\dots,4\} \\ n \in N_{GD}}} \end{matrix} \right\rangle \quad \text{where}$$

G (GT or GD) – Berge's graphs, $G = \langle N_G, A_G \rangle$, N_G, A_G – sets of graph's nodes and arcs, $A_G = \{\langle n, n' \rangle : n, n' \in N_G\}$. Weighted graphs WGT and WGD describe decision situations (current CS and pattern PS). Each node n of GT and GD describes terrain cells $(i,j)=n$ with non-zero values of characteristics defined as components of SD_{ij} from SD and $\underset{k \in \{1,\dots,3\}}{\forall} f_k^T(n) = SD_{ij}^{5,k}$, $f_4^T(n) = SD_{ij}^{7}$, $\underset{k \in \{1,\dots,3\}}{\forall} f_k^D(n) = SD_{ij}^{5,3+k}$, $f_4^D(n) = SD_{ij}^{5,7}$. Two nodes $x, y \in N_{GD}$ (for $x, y \in N_{GT}$ by analogy) are linked using an arc when the cells represented by x and y are adjacent (the more precisely: are adjacent taking into account direction of action, see Fig.3). For example the terrain can be divided into 15 cells (3 rows and 5 columns, left-hand side, see Fig.3). The units are located in some cells (denoted by circles and sharps). Structural representation of units deploying is defined by the graph GD. Let's note that similar representation we can use for topographic conditions (single graph for one of the topographic information layer: water, forests, passability or single graph GT for all of these information, see Fig.3, right-hand side).

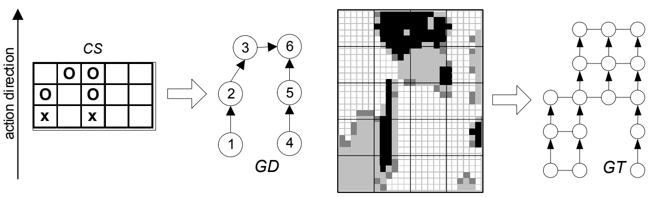

Fig. 3. Units deploying and their structural (graph GD) representation (left-hand side) and terrain covering (growth) and its structural (GT) representation (right-hand side). Circle (O) and sharp (X) describe two types of units.

The pattern generation from the terrain point of view is based on the specific classification. The terrain classification method is based on some model of the terrain, which is used in the SBOTSS. This model is closely integrated with a geographic information system (digital map) and a simulation system and it is defined as regular grid of terrain squares. Regular grid of squares divides terrain space into the squares with the same size. Each square can be analyzed considering the terrain characteristics (degree of velocity weakness, ability to camouflage, degree of visibility, etc.). This model is used to plan off-roads (cross-country) movement e.g. during attack planning. In the simulation system, the second terrain model (as road-railroad network model) is also defined but this is not used in the terrain classification method.

The terrain classification depends on the following characteristics:

- Terrain Topography = (surface, vegetation, soil);
- Weather = (temperature, wind, precipitation, transparency):
    - Temperature – high, medium, low,
    - Wind – strong, medium, weak,
    - Precipitation – strong, medium, lack,
    - Transparency – good, weak, bad;
- Season of the day – night, day (morning, afternoon, evening);
- Season of the year – spring, summer, autumn, winter.

The idea of the terrain classification method is to estimate terrain region in which own and opposite units will operate to obtain one of the four types of the terrain: go, slow go, no go, no move. The first type of the terrain (go) is excellent for movement (e.g. plain terrain), the second one (slow go) is good for movement (e.g. soft-hilly terrain), the third kind of the terrain (no go) is poor for movement (e.g. hard-hilly terrain or mountainous terrain) and the last kind of the terrain (no move) describes impassable terrain (e.g. lakes, seas, high mountains).

Stage 2

Having weighted graphs $WGD(CS)$ and $WGD(PS)$ ($WGT(CS)$ and $WGT(PS)$) representing current CS and pattern PS decision situations (for units deploying) we can modify graphs similarity approach [4], [5] to find the most similar decision situation pattern to current situation (for pair of graphs $WGT(CS)$ and $WGT(PS)$ by analogy). In general, having two weighted graphs G_A (e.g. $WGT(CS)$) and G_B (e.g. $WGT(PS)$) we have proposed to calculate two types of similarities of the G_A and G_B: structural and non-structural (quantitative) [15].

To calculate structural similarity between G_A and G_B it is proposed to use approach defined by Blondel, van Dooren et al. in [5]. Let A and B be transition matrices of G_A and G_B. We calculate following sequence of matrices:

$$Z_{k+1} = \frac{BZ_kA^T + A^TZ_kB}{\left\| BZ_kA^T + A^TZ_kB \right\|_F}, \quad k \geq 0 \qquad (6)$$

where $Z_0=\mathbf{1}$ (matrix with all elements equal 1); x^T – matrix x transposition; $\|x\|_F$ - Frobenius (Euclidian) norm for matrix x,

$\|x\|_F = \sqrt{\sum_{i=1}^{n_B}\sum_{j=1}^{n_A} x_{ij}^2}$, n_B – number of matrix rows (number of nodes of G_B), n_A – number of matrix columns (number of nodes of G_A). Element z_{ij} of the matrix Z describes similarity score between the i-th node of the G_B and the j-th node of the G_A. The greater value of z_{ij} the greater similarity between the i-th node of the G_B and the j-th node of the G_A. The essence of graph nodes similarity is fact that two graphs' nodes are similar if their neighborhoods are similar. We obtain structural similarity matrix $S(G_A,G_B)$ between nodes of graphs G_A and G_B as follows [5]:

$$S(G_A,G_B) = [s_{ij}]_{n_B\times n_A} = \lim_{k\to+\infty} Z_{2k} \tag{7}$$

Having matrix $S(G_A,G_B)$, we can formulate and solve optimal assignment problem (using e.g. Hungarian algorithm) to find the best allocation matrix $X = [x_{ij}]_{n_B\times n_A}$ of nodes from graph describing G_A, G_B:

$$d_S(G_A,G_B) = \sum_{i=1}^{n_B}\sum_{j=1}^{n_A} s_{ij}\cdot x_{ij} \to \max \tag{8}$$

with constraints:

$$\begin{aligned}
&\sum_{i=1}^{n_B} x_{ij} \le 1, \quad j = \overline{1,n_A}\\
&\sum_{j=1}^{n_A} x_{ij} \le 1, \quad i = \overline{1,n_B}\\
&\underset{i\in\{1,\dots,n_B\}}{\forall}\ \underset{j\in\{1,\dots,n_A\}}{\forall} x_{ij} \in \{0,1\}
\end{aligned} \tag{9}$$

The $d_S(G_A,G_B)$ describes value of *structural similarity measure* between G_A and G_B.

To compute non-structural (quantitative) similarity between G_A and G_B we should consider similarity between values of node and arc functions (*nodes and arcs quantitative similarity*) [15]. To compute nodes quantitative similarity we propose to create vector $\boldsymbol{v}(G_A,G_B) = \langle V_1,\dots,V_{LF}\rangle$ of matrices, where $V_k = \left[v_{ij}(k)\right]_{n_B\times n_A}$, k=1,…,LF, describing similarity matrix between nodes of G_A and G_B from the point of view of the k-th node's function ($f_k^A : N_{G_A} \to R^n$ for G_A and $f_k^B : N_{G_B} \to R^n$ for G_B) and $v_{ij}(k) = \left\| f_k^B(i) - f_k^A(j)\right\|$ describes "distance" between the i-th node of G_B and the j-th node of G_A from the point of view of f_k^B and f_k^A, respectively. We can apply norm with parameter $p \ge 1$ as distance measure:

$$\left\| f_k^B(i) - f_k^A(j)\right\|_p = \left(\sum_{r=1}^{n}\left| f_{k,r}^B(i) - f_{k,r}^A(j)\right|^p\right)^{1/p} \tag{10}$$

where $f_{k,r}^A(\cdot)$, $f_{k,r}^B(\cdot)$ describe the r-th component of n-component vector being value of f_k^A and f_k^B, respectively. Next, we compute for each k=1,…,LF normalized matrix $V_k^* = \left[v_{ij}^*(k)\right]_{n_B\times n_A}$, where $v_{ij}^*(k) = v_{ij}(k)\big/\|V_k\|_F$. This procedure guarantees that each $v_{ij}^*(k)\in[0,1]$. Finally, we compute total quantitative similarity between the i-th node of G_B and the j-th node of G_A as follows:

$$\bar{v}_{ij} = \sum_{k=1}^{LF}\lambda_k \cdot v_{ij}^*(k), \quad \sum_{k=1}^{LF}\lambda_k = 1, \quad \underset{k=1,\dots,LF}{\forall} \lambda_k \in [0,1] \tag{11}$$

We determine the $d_{QN}(G_A,G_B)$ *nodes quantitative similarity measure* of G_A and G_B, solving assignment problem (8)-(9) where $-\bar{v}_{ij}$ substitutes for s_{ij} (because of the smaller value of $\bar{v}_{ij}$ the better) and $d_{QN}(G_A,G_B)$ for $d_S(G_A,G_B)$ in (8).

Then, the weighted graphs $WGD(CS)$ and $WGD(PS)$ ($WGT(CS)$ and $WGT(PS)$) represent current CS and pattern PS decision situations (for units deploying) and we can use procedure described above to calculate structural and quantitative similarity measures for both graphs. So, we obtain for WGD:

$$d_S(WGD(CS),WGD(PS)) = d_S^D(CS,PS),$$

$$d_{QN}(WGD(CS),WGD(PS)) = d_{QN}^D(CS,PS)$$

and for WGT:

$$d_S(WGT(CS),WGT(PS)) = d_S^T(CS,PS),$$

$$d_{QN}(WGT(CS),WGT(PS)) = d_{QN}^T(CS,PS).$$

Stage 3

Let $SG = PDSS$ be a set of weighted graphs defining decision situation patterns and P ($WGD(CS)$ or $WGT(CS)$) defines a pattern. The problem is to find such a graph G^o from SG that is the most similar to P. We define this problem as multicriteria weighted graphs similarity problem (*MWGSP*), that is multicriteria optimization problem in the space SG with relation R_D:

$$MWGSP = \left(SG, F, R_D\right) \tag{12}$$

where $F: SG \to R^2$, $F(G) = \left(d_S(P,G), d_{QN}(P,G)\right)$ and

$$R_D = \left\{\begin{aligned}(Y,Z)\in SG\times SG:\ d_S(P,Y) \ge d_S(P,Z) \wedge\\ d_{QN}(P,Y) \le d_{QN}(P,Z)\end{aligned}\right\} \tag{13}$$

Domination relation R_D (Pareto relation between elements of SG) gives possibilities to compare graphs from SG. Weighted graph Z is more similar to P than Y if structural similarity between P and Y is not smaller than between P and Z and, simultaneously, quantitative similarity between P and Y is not greater than between P and Z. We propose to use scalar function $H(G): SG \to R$ as weighted sum of objectives:

$$\begin{aligned}&H(G) = \alpha_1\cdot d_S(P,G) + \alpha_2\cdot\left(-d_{QN}(P,G)\right)\\ &\alpha_1,\alpha_2 \ge 0, \quad \alpha_1+\alpha_2 = 1\end{aligned} \tag{14}$$

Taking into account (14) the problem of finding the most matched G^o to pattern P can be formulated as follows: to determine such a $G^o \in SG$, that $H(G^o) = \max_{G\in SG} H(G)$.

Now, we can formulate problem (12), separately for WGT and WGD, where: SG:=$PDSS$, $F(G)$:=$F_D(PS)$, $d_S(P,G) := d_S^D(CS,PS)$, $d_{QN}(P,G) := d_{QN}^D(CS,PS)$ for WGD and $F(G)$:=$F_T(PS)$, $d_S(P,G) := d_S^T(CS,PS)$, $d_{QN}(P,G) := d_{QN}^T(CS,PS)$ for WGT (where ":=" means substitution). Next, we define scalar functions (14) to solve the problem (12) for WGD and WGT: $H_D(\cdot) = \alpha_1\cdot d_S^D(\cdot,\cdot) + \alpha_2\cdot(-d_{QN}^D(\cdot,\cdot))$ and

$H_T(\cdot) = \gamma_1 \cdot d_S^T(\cdot,\cdot) + \gamma_2 \cdot (-d_{QN}^T(\cdot,\cdot))$.

Having $H_D(PS)$ and $H_T(PS)$ we can combine these criteria (like in (14)) or to set some threshold values and to select the most matched pattern situation to current one.

Example of using presented approach to find the most matched pattern decision situation to current one is presented in the Fig.4 and in the Table 1. We present results of calculations $H_D(PS)$ for each $PS \in PDSS = \{PS_1,...,PS_6\}$. We use only function $f_4^{D(CS)}(n) = SD_{ij}^8$ ($f_4^{D(PS)}(n)$ for pattern PS) from WGD to compute nodes quantitative similarity because all units have the same type. Thus, vector $\boldsymbol{v}(WGD(CS),WGD(PS))$ of matrices contains one component $V_1 = [v_{ij}(1)]_{|N_{GD(PS)}|\times|N_{GD(CS)}|}$. Function $f_4^{D(CS)}(n)$ describes coordinates of node n (left-lower cell has coordinates (1,1)). The norm from (10) has the form: $\left\| f_4^D(i) - f_4^D(j) \right\|_{p=2} = \left(\sum_{r=1}^{2} \left| f_{4,r}^D(i) - f_{4,r}^D(j) \right|^2 \right)^{1/2}$ and it describes geometric distance between nodes $i \in N_{GD(PS)}$ and $j \in N_{GD(CS)}$. Let's note that for weights $\alpha_1 = 0,\ \alpha_2 = 1$ value in the Table 1 (for the row PS_i) describes $d_{QN}^D(CS,PS_i)$ and for $\alpha_1 = 1,\ \alpha_2 = 0$ describes $d_S^D(CS,PS_i)$. The best matched PS to CS is PS_2 (taking into account d_S^D and d_{QN}^D for non-zero values of α_1, α_2), Table 1.

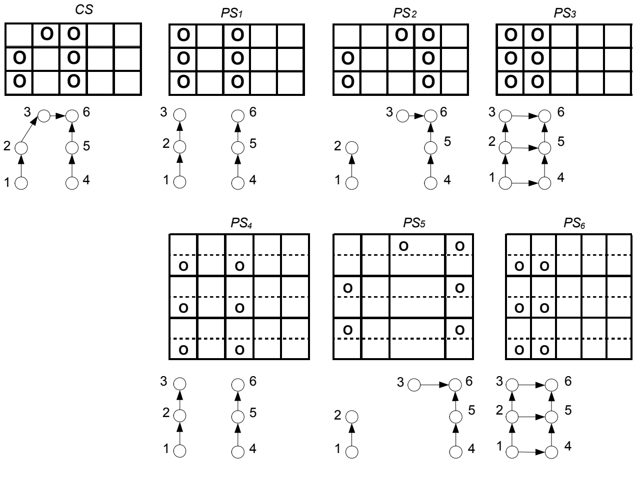

Fig.4. Current situation CS with graph $GD(CS)$ and six pattern situations PS_i (i=1,…,6) with graphs $GD(PS_i)$ describing structure of units deploying. Patterns 1-4, 2-5, 3-6 have the same structure but cells for patterns 4,..,6 have greater size than for patterns 1,…,3.

The process of optimal selection of weights can be organized as follows: we build learning set $\{CS_i, PDSS_i\}_{i=1,\ldots,LS}$ and for different values of weights experts estimate whether, in their subjective opinion, CS_i is similar to the $PS^* \in PDSS_i$ determined from the procedure. Combination of weights values which are indicated by majority of experts is optimal combination.

TABLE 1

Value of scalar function $H_D(PS_i)$ combining structural (weight α_1) and quantitative (weight α_2) similarity measures between $GD(CS)$ and $GD(PS_i)$ from Fig.4. With bold type we denote the best (maximum) value in the column

Pattern	Weights (α_1 ; α_2)				
	(0; 1)	(0.33; 0.67)	(0.5; 0.5)	(0.67; 0.33)	(1; 0)
PS_1	**-0.094**	**0.283**	0.463	0.800	**1.527**
PS_2	-0.370	**0.283**	**0.593**	**0.870**	1.504
PS_3	-0.478	0.157	0.360	0.726	1.254
PS_4	-0.474	0.120	0.461	0.824	**1.527**
PS_5	-0.706	0.032	0.378	0.761	1.504
PS_6	-0.63	0.070	0.279	0.631	1.254

IV. COURSE OF ACTION GENERATION

If we have pattern decision situation most similar to current one, we could obtain set of action plan templates from tactical knowledge base. Action plan template contains such elements as: type of formation, tasks of units in each echelon of formation, type of manoeuvre. In order to generate full operation plan, we should determine deployment of our forces, manoeuvre routes, plan of fire, tasks for support units and for air support, plan of supply of military materiel by logistic units.

The next steps, after generation of set of operation plans, are evaluation of all variants of operation plan and choice the best one. For variants evaluation we use the pre-simulation process based on some procedures: forces attrition procedure, slowing down rate of attack procedure, utilization of munitions and petrol procedure.

Forces attrition procedure is based on the following relations:

$$\begin{aligned} Pog_B(id',t+\Delta t) = Pog_B(id',t) - \\ \sum_{id \in JW_B^{-1}(id',t)} f\_\mathrm{int}(id,t) \cdot \Lambda_{ref}(id,t_0,id',dist(id,t,id')) \cdot \\ \cdot \frac{Pog_A(id,t)/dist(id',t,id)}{\sum_{id'' \in JW_A(id.t)} (Pog_B(id'',t)/dist(id,t,id''))} \cdot \frac{Pog_A(id,t)}{Pog_A(id,t_0)} \cdot \Delta t \end{aligned} \qquad (15)$$

for $id' \in B$ (for $id \in A$ by analogy), where: A, B – sides of combat;

$Pog_A(id,t), Pog_B(id,t)$ - combat potential of two sides units;

$\Lambda_{ref}(id',t_0,id,dist(id',t,id))$ - intensity of id' unit fire against the unit id, under distance condition *dist(id',t,id)* and fully supplied units,

$f\_\mathrm{int}(id',t)$ - the part of full potential fire of unit *id'* used at time *t*.

The slowing down rate of attack procedure uses the following functions:

$$v_{akt}(id,t) = \min\{v_{dec},\ v_{\max}^{op}(id,t)\} \qquad (16)$$

where: $v_{\max}^{op}(id,t)$ - real maximum velocity of unit *id*;

$v_{\max}^{op}(id,t) = v_{\max}(id,t) \cdot StOsl\,\mathrm{Pr}\,edk(Cond\_env(id,t), StSp_A(id,t),$
$in\_kill\_ratio(id, JW_A^{-1}(id,t),t))$

$v_{\max}(id,t)$ - maximum velocity of unit *id* depends on technical possibilities of armaments;

$StOsl\Pr edk$ - slowing down velocity function depends on:

a) terrain conditions - $Cond\_env(id,t)$

b) unit percent dismounted - $StSp_A(id,t)$

c) kill ratio index - $in\_kill\_ratio(id, JW_A^{-1}(id,t),t)$ - depends on attrition rates of combat potential;

The utilization of munitions and petrol procedures are described in details in [1], [3].

Manoeuvre routes and units velocity are determined using procedures, which contain two main parts:

- the determination of shortest path for subordinate units under attack condition with maximum possible velocity,
- the modification of velocity values due to coordination of subordinate units during their actions in the battlefield.

During pre-simulation process, we obtain values of such combat characteristics as: time and degree of task realization, own and enemy losses, utilization of munitions and petrol. We can formulate problem of finding the best operational plan as a multiple criteria optimization problem with lexicographical relation. The next phase of automata activity there is direct combat control, which is connected with realization of decision made in previous phase. On the basis of observed actions of subordinate units the automata reacts to possible deviation of real trajectories in comparison to determined in planning phase.

The automata was implemented in the environment of distributed interactive simulation system (Polish Simulation Based Operational Training Support System for CAX [12]) in ADA language and it was tested with some scenarios of land combat exercises at the brigade and division levels. The environment proposed is constructed as distributed interactive simulator with respect to HLA (High Level Architecture). HLA was developed by the DMSO of the US DoD to meet the needs of reusability and interoperability in virtual, constructive and live simulations. Due to HLA features there is easy way to include new models, unit structures and tactical rules. The synchronization and communication mechanisms rely on conservative algorithms and implement assumptions of a constructive discrete-event simulation. Special extensions of ADA language were constructed to manage a set of simulation events, activities and simulation time. Time management services concern the chronological order of events (local and delivered to federates via messages), and the mechanisms for advancing simulation time.

V. CALIBRATION OF COMBAT MODELS

One of the most important feature of simulation model is its adequateness. The good simulation model should represent real system as accurately as possible. It should include some internal mechanisms which give possibilities to flexible modification of parameters for system's model tuning. Battlefield processes are very complex and relatively poorly recognized because of small set of results of real conflicts. The calibration of simulation models is considered in [6], [8]. The target searching, firing and movement in context of decision process evaluation can be used as example of such simulated battlefield process.

The importance of model calibration for practical work is highlighted in different publications [2], [7]. It relays on adjusting an already implemented model to a reference system (or, if system data is not available, to a trusted reference model). This is usually done by adjusting some internal parameters of the model and then verifying obtained outputs. The adjustments done by parameter calibration are involved in the models abstractions, idealization, and many disputable assumptions. Truly reliable data for the V&V process is not available and even the reference system is not clearly defined. Therefore the calibration of most military models is described in more humble, approximate, and selective manner. One of the important ways of adjusting of combat simulation model there is collecting of tactical parameters value during real conflict or battlefield exercises and using as the simulation model parameters. Another sufficient way there is military expert participation in calibration process and attuning of the parameters taking into account the behaviour of simulated units during the experiment.

Calibrating tool

The calibrator in SBOTSS is a tool which enables (before running simulation scenario) fixing values of calibration parameters which have influence on many simulation models. It includes 126 parameters which are grouped into 8 categories and their values are being written (read) in (from) calibration database. For each of these parameters the default value has been proposed (after pre-calibration) but it is possible to change values of these parameters separately for each of the simulation scenarios. The most important categories of the calibration parameters are as follows:

- parameters of detection category (16 parameters) – coefficients which have influence on intensity of target detections depending on: time of the day (night), velocity of target, velocity of observer, terrain roofing (buildings, forests) and relief;
- slowing down rate of actions category (14 parameters) – coefficients which describe slowing down rate of actions depending on kind of vehicle, kind of terrain soil and time range of precipitation;
- rate of battalion advance category (43 parameters) – coefficients which describes rate of battalion advance depending on: type of the terrain, degree of engineering preparation of defence, strength of opponent (military potential);
- action capabilities category (18 parameters) – these coefficients describe levels o units military potential below which units lose its action capabilities or are broken.

Analysis of selected calibration results

We present selected results of calibration for three scenarios: k_base (day scenario), k_time (night scenario), k_meteo (meteo scenario). The k_base scenario has been used for majority of calibrations. Military actions in this scenario are conducted during the summer day with excellent weather. Difference between k_time and k_base scenarios consists in start time for units actions (in the k_base units starts their actions about 6.00 and in the k_time – about 21.00). Difference between k_meteo and k_base scenarios consists in weather (in the k_base scenario weather is very good and in the k_meteo is pouring rain).

Structure of units for both conflict sides is presented in the Fig.5 (the same for all scenarios). The first conflict side (blue, 1) realizes attack and the second one (red, 2) defends. The first side (mechanized brigade) consists of 2 mechanized battalions (with 3 or 4 mechanized companies on BMP-1 mechanized combat vehicles) and 1 armoured battalion (with 3 armoured companies on T-72m tanks). The second side consists of 1 mechanized battalion with 3 companies on BMP-1.

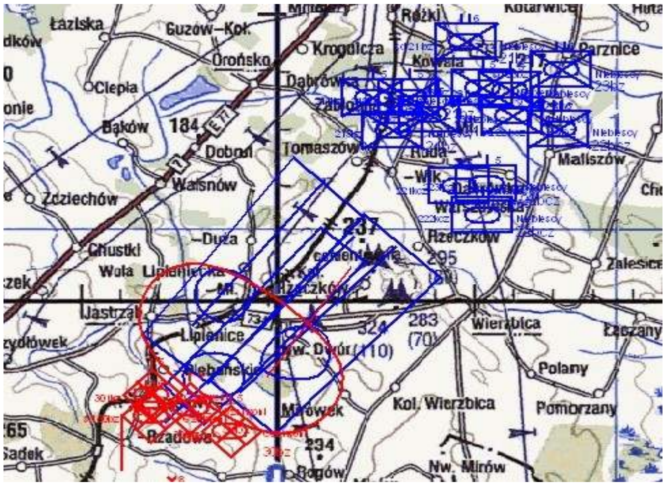

Fig. 5. Tactical situation concerning scenarios k_base, k_time, k_meteo

We would like to present calibration results for the nominal visibility range at day and their influence on losses and target detections of both conflict sides (Fig.6¸7). From the Fig.6 results the smaller nominal detection distance (visibility range) the greater suffers losses the attacking side (1) because of the greater detection intensity by the 2nd side which defends. From comparison Fig.7 and Fig.6 results that intensity of target detection by the 1st side is greater for the greater distances because in such a case the 1st side suffers smaller losses that is more combat means of the 1st side realize detection process and take advantage of effective range of fire (because the first side has tanks with greater effective range of fire than defending side). The last conclusion is the more suffers losses of the 2nd side (Fig.7, graph for 6000m is “over” the graphs for other values of parameter) the less means losses the 1st side (Fig.6, graph for 6000m is “under” the graphs for other values of parameter) and vice versa.

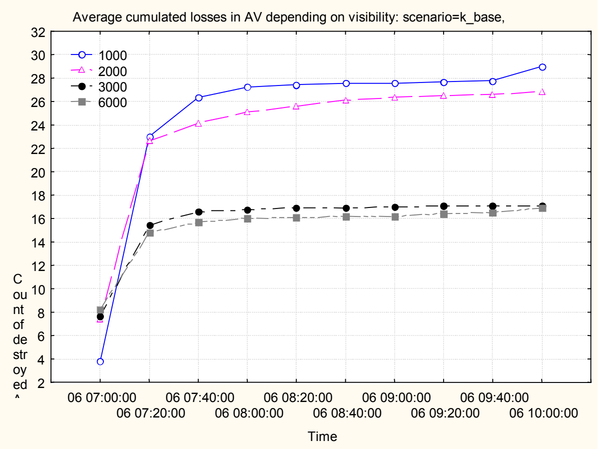

Fig. 6. Average cumulated losses in armoured vehicles (AV) depending on visibility range at day: scenario=k_base, side=1

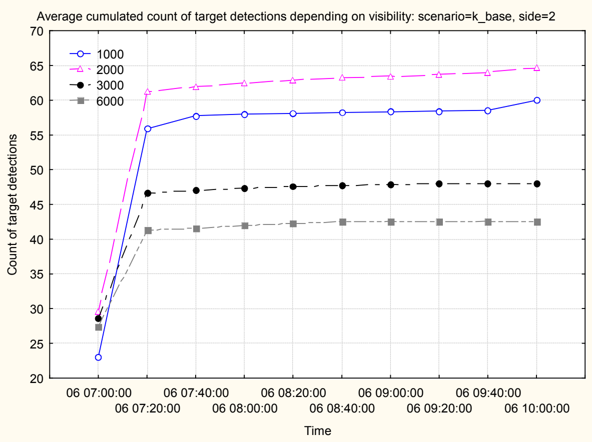

Fig. 7. Average cumulated count of target detections depending on visibility range at day: scenario=k_base, side=2

Let’s note that graphs for target detection of one side has similar shape as graphs for losses for another side (for example Fig.6 and Fig.7).

Velocity of observer and target and their influence on simulation results for calibration scenarios are presented in the Fig.8÷9. The 2nd side defends hence it is static and the greatest losses for attacking (Fig.8) are for the greater value of coefficient static_observer-moving_target because of the greater value of the coefficient the greater intensity of target detection of moving target (1st side) by static observer (2nd side). From the point of view of the 2nd side losses (Fig.8) the most profitable is the greatest value because of greater intensity of target detection, faster target destroying and, in consequence, less number of combat means of attacking side may destroy the 2nd side.

The 1st side attacks hence it is moving and the greatest losses for defending (Fig.9) are for the greater value of coefficient moving_observer-static_target because of the greater value of the coefficient the greater intensity of target detection of static targets (2nd side) by moving observer (1st side). From the point of view of 1st side losses (Fig.9) the most profitable is the greatest value of coefficient because of greater intensity of target detection, faster target destroying and, in consequence, less number of combat means of defending side may destroy the 1st side.

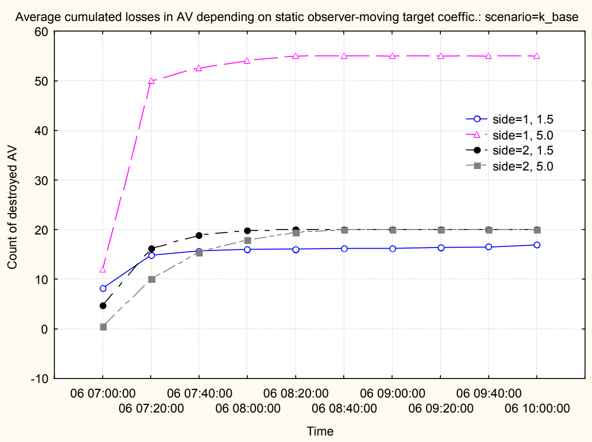

Fig. 8. Average cumulated losses in AV depending on static_observer – moving_target coefficient: scenario=k_base

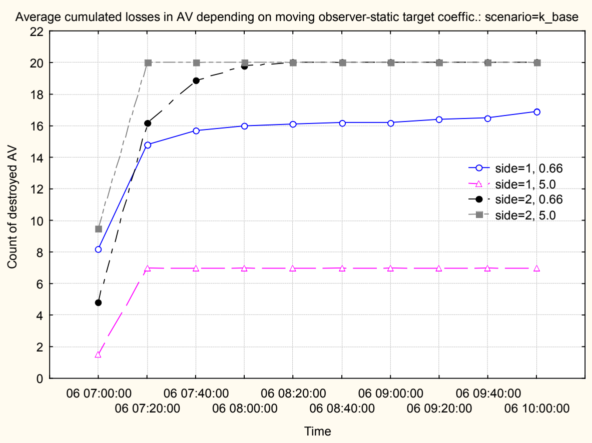

Fig. 9. Average cumulated losses in AV depending on moving_observer – static_target coefficient: scenario=k_base

VI. CONCLUSIONS

The environment proposed is built as an open system and can be developed and improved - that means there is easy way to include new combat models, unit structures, tactical rules and more monitored characteristics. The characteristics of battle process are being monitored during the simulation process and their statistical analysis allows combat actions predicting for different conflict situations. It should be stressed that approach proposed here requires good knowledge of conflict processes and careful preparation of a conflict scenario. The validation process is very difficult but it is possible to use such tools like calibrator and expert knowledge. The construction of the simulation model enables the testing different course of actions including ideas in the area of Network Enabled Capabilities [10].

REFERENCES

[1]. Antkiewicz R., Kulas W., Najgebauer A., Pierzchała D., Rulka J., Tarapata Z., Wantoch-Rekowski R.: Case-Based C2 Modelling and Effective Development, Implementation and Experimentation for Simulation Based Operational Training Support System. Proceedings of the NATO RTO Modelling and Simulation Conference, ISBN 92-837-0047-3, 13-14.10.2005, Warsaw.

[2]. Antkiewicz R., Najgebauer A., Rulka J., Tarapata Z.: Calibration of simulation models of selected battlefield processes, Proceedings of the 1st Military Communication and Information Systems Conference, ISBN 83-920120-1-1, 18-19.09.2006, Gdynia (Poland).

[3]. Antkiewicz R., Kulas W., Najgebauer A., Pierzchała D., Rulka J., Tarapata Z., Wantoch-Rekowski R.: Modelling and simulation of C2 processes based on cases in the operational simulation system for CAXes, Proceedings of the 1st Military Communication and Information Systems Conference, 18-19.09.2006, Gdynia, ISBN 83-920120-1-1.

[4]. Bunke H.: Graph matching: Theoretical foundations, algorithms, and applications, in Proceedings Vision Interface 2000, Montreal, 2000, pp.82 – 88.

[5]. Blondel V., Gajardo A., Heymans M., Senellart P., Van Dooren P.: A measure of similarity between graph vertices: applications to synonym extraction and web searching. SIAM Review, v.46(4), 647-666, 2004.

[6]. Dockery J., Woodcock A.E.R.: The Military Landscape, Mathematical Models of Combat, Published by Woodhead Publishing Ltd., Cambridge, England, 1993.

[7]. Hoffman M.: On the Complexity of Parameter Calibration in Simulation Models, JDMS, The Society for Modeling and Simulation International, Volume 2, Issue 4, pp.217–226, October 2005.

[8]. Hofmann H., Hofmann M.: On the Development of Command & Control Modules for Combat Simulation Models on Battalion down to Single Item Level, Defense Technical Information Center, 2001.

[9]. Melnik S., Garcia-Molina H., Rahm E.: Similarity Flooding: A Versatile Graph Matching Algorithm And Its Application To Schema Matching, Proceedings of the 18th International Conference on Data Engineering, San Jose, California, (2002), 117–128.

[10]. Moffat J.: Complexity Theory and Network Centric Warfare, CCRP Publication Series, ISBN 1-893723-11-9, Washington 2003.

[11]. Najgebauer A., Decision Support Systems for Conflict Situations. Models, Methods and the Interactive Simulation Environments (in Polish). Ed. Biuletyn WAT. Warsaw 1999, Poland. (294 p.). ISBN 83-908620-6-9.

[12]. Najgebauer A.: Polish Initiatives in M&S and Training. Simulation Based Operational Training Support System (SBOTSS) Zlocien, Proceedings of the ITEC'2004, London, UK, April 20-22 2004.

[13]. Ross K., Klein G., Thunholm P., Schmitt J., Baxter H.: The Recognition-Primed Decision Model, Military Review, July-August 2004, pp.6-10.

[14]. Sokolowski, J.A.: Can a Composite Agent be Used to Implement a Recognition-Primed Decision Model?, In: Proceedings of the Eleventh Conference on Computer Generated Forces and Behavioral Representation, Orlando, FL., May 7-9 2002, pp.473-478.

[15]. Tarapata Z.: Multicriteria weighted graphs similarity and its application for pattern recognition of decision situations, in: Technical Report from the Research Project N^oo PBW/571/2006, Military University of Technology, Faculty of Cybernetics, Warsaw 2006.