Zbigniew TARAPATA*

A COMPUTER TOOL FOR SUPPORTING AND EVALUATING CONVOYS REDEPLOYMENT PLANNING

A computer environment being utilized to examine redeployment planning for a few convoys in real terrain is presented. This tool enables supporting and evaluating movement planning (and its algorithms) any object columns both for deterministic and stochastic structure of a network being model of the terrain. A mathematical model of K convoys redeployment is described and optimization problems related to this both in reliable and unreliable network are formulated. To examine redeployment and its algorithms during destructive influencing any opponents on the moved objects and network a simulation model of redeployment is discussed. A detailed description of presented application is given and numerical example being solved by utilization of considered tool is shown. Moreover, possibilities of civilian applications of presented tool are indicated.

1. Introduction

This paper deals with the computer environment being used to support redeployment planning K convoys. The tool contains options which enable to evaluate some algorithms constructed to solve the optimization problem related to redeployment planning and shortly described in section 2 of the paper. It is obvious, that during columns redeployment potential opponent will try to destroy both structure elements of the network (for example, crossings (network's node) or parts of the road (network's arcs)) and moved objects to make impossible achievement of the column destinations. In this connection discussed tool gives possibilities to find K-dimensional vector of the best paths (so-called K-path) in the network being destroyed using a few original algorithms. Moreover, a simulation module to examine and evaluate these algorithms during redeployment for various network destruction plans is

* Institute of Mathematics and Operations Research, Cybernetics Faculty, Military University of Technology, Kaliskiego Str. 2, 01-908 Warsaw, e-mail: ztarap@isi.wat.waw.pl.

contained. This module takes into consideration, among other things, frequency of information, received by columns being moved, from own reconaissance means.

This paper is organizend as follows.

In section 2 a mathematical model of the K convoys redeployment both in reliable and unreliable network is presented. Section 3 contains short description of the simulation model of redeployment. In section 4 detailed description of considered computer environment and numerical example being solved by utilization of discussed tool are given. In the last section a few conclusions concerning utilization of presented tool are discussed.

2. A mathematical model of redeployment

Redeployment process corresponds to one of types of military actions to be known as manoeuvre of detachments. In military practice, the moved vehicles are grouped in columns (convoys). In this connection each k-th objects column has some length α_k and velocity v_k. We say that each column has so-called time length $\delta_k = \frac{\alpha_k}{v_k}$. This value effects the length of occupied time interval by the k-th column of some route part. Moreover, to increase of redeployment safety, it is required paths for moved columns to be independent (disjoint). In other words, no common nodes for K column's paths are permitted. In case of military objects redeployment we can use so-called trafficable routes network (TRN) as a model of redeployment environment. For example, in [7] two kinds of TRN, rough and precise, were proposed. The first type of the network was based on the real roads in the terrain only. Depending on the detail level, we can consider only the roads of: the first categories, the first and the second ones, the first and the second and the third, etc. The precise network was based on partitioning of the terrain into squares of topographical homogeneous areas. This information was coded by utilization of quadtrees.

We define the routes network **S** as follows:

$$\mathbf{S} = \left\langle G, \left\{ \left\langle x_w, y_w \right\rangle \right\}_{w \in W_G} \right\rangle \tag{1}$$

in which G denotes Berge's graph describing structure of **S**, $G = \left\langle W_G, A_G \right\rangle$; A_G - set of the graph arcs; W_G - set of the graph nodes; x_w, y_w - respectively, x and y coordinates of centre of the node $w \in W_G$; $W = |W_G|$.

Let the maximum speeds of K moved objects be $v^1, v^2, \ldots, v^k, \ldots, v^K$, respectively. Because these speeds may be various for each of the objects, so the vector $t_{w,w'}$ of minimum crossing times for (w,w') arc assumes the following form:

$$t_{w,w'} = \left\langle t^1_{w,w'}, t^2_{w,w'}, \ldots, t^k_{w,w'}, \ldots, t^K_{w,w'} \right\rangle, \quad w, w' \in W_G \tag{2}$$

where the k-th component of $t_{w,w'}$ is as follows:

$$t^k_{w,w'} = \begin{cases} \dfrac{d_{w,w'}}{v^k \cdot \zeta((w,w'))} & , \quad (w,w') \in A_G \\ 0 & , \quad w = w' \\ \infty & , \quad (w,w') \notin A_G \end{cases}, \tag{3}$$

in which $d_{w,w'} = \sqrt{(x_w - x_{w'})^2 + (y_w - y_{w'})^2}$ - element of matrix $D = [d_{w,w'}]_{W \times W}$ of terrain distances between graph nodes. The $\zeta(\cdot)$ function values belong to the interval (0,1] and they describe coefficient of speed decreasing in topographical or weather conditions, etc., for a part of the road represented by arc being the function argument.

K-path **I** in the **S** network, from $\mathbf{i}^s$ to $\mathbf{i}^d$, is K-element set of simple paths, that is

$$\mathbf{I}(\mathbf{i}^s, \mathbf{i}^d) = (I_1, I_2, \ldots, I_k, \ldots, I_K) \tag{4}$$

in which vectors: $\mathbf{i}^s = \left\langle i_1^s, i_2^s, ..., i_k^s, ..., i_K^s \right\rangle$ and $\mathbf{i}^d = \left\langle i_1^d, i_2^d, ..., i_k^d, ..., i_K^d \right\rangle$ are called as *source K-node* and *destination* one, respectively.

Let nodes, belonging to the paths of the K-path, satisfy following condition :

$$\underset{k,k'=\overline{1,K}}{\forall} \; \underset{\substack{r=\overline{0,R_k} \\ r'=\overline{0,R_{k'}} \\ r \neq r' : k = k'}}{\forall} \; i^r(k) \neq i^{r'}(k') \tag{5}$$

where $i^r(k)$ denotes the r-th node on the path for the k-th column; R_k+1 – count of nodes belonging to the path for the k-th column.

Independent (disjoint) K-path is such K-path in **S**, which satisfies (5). We denote by $\tau_k^d = \tau^{R_k}(k) + \delta_k$ ending moment of redeployment for the k-th column,

$$\tau_k^d = \tau_k^s + \sum_{r=1}^{R_k} t^k_{i^{r-1}(k), i^r(k)} + \delta_k \qquad , \; k = \overline{1,K} \tag{6}$$

where τ_k^s - starting moment of the k-th column from its source node; $t^k_{i^{r-1}(k), i^r(k)}$ - crossing time of arc between $i^{r-1}(k)$ and $i^r(k)$ nodes belonging to the path for the k-th column; δ_k - time length of the k-th column,

$$\delta_k = \left(\sum_{r=1}^{\tilde{r} \in \mathbf{R}_k : s_r \geq 0} \frac{s_r}{v^k \cdot \xi\left(\left(i^{r-1}(k), i^r(k)\right)\right)} \right) + \frac{\alpha_k - s_{\tilde{r}}}{v^k \cdot \xi\left(\left(i^{r-1}(k), i^r(k)\right)\right)} \tag{7}$$

where $\mathbf{R}_k = \{0,1,2,...,R_k\}$ denotes set of indices of nodes belonging to the path of the k-th column; $s_r = \alpha_k - \sum_{m=1}^{r} d_{i^{m-1}(k), i^m(k)}$; α_k - length of the k-th column (see at the beginning of this section); $d_{(\cdot),(\cdot)}$- element of D matrix described in (3).

Moreover, it assumes that elements of K-path from (4) must satisfy the following condition:

$$\underset{k=\overline{2,K}}{\forall} \; d_{min} \leq d^*(I_k, I_{k-1}) \leq d_{max} \tag{8}$$

where $d^*(\cdot,\cdot)$ – distance between paths for neighbouring columns; d_{min}, d_{max} - minimal and maximal distances between paths for neighbouring columns, respectively. The (8) constraint may result, for example, from necessities of safety ensuring for moved columns to minimize losses in case of eventual opponent attack (lower constraint) and when the width of an action strip for the columns is set (upper constraint).

The basic problem is to evaluate such $\mathbf{I}^*$ which satisfies the following condition:

$$\tau^*(\mathbf{I}^*) = \min_{\mathbf{I}(\mathbf{i}^s,\mathbf{i}^d)\in M(\mathbf{i}^s,\mathbf{i}^d)} \max_{k=\overline{1,K}} \tau_k^d(\mathbf{i}^s,\mathbf{i}^d) \tag{9}$$

where $\mathbf{i}^s,\mathbf{i}^d$ - source and destination K-nodes; $\mathbf{I}(\cdot,\cdot)$ – independent K-path in the $\mathbf{S}$ network; $M(\cdot,\cdot)$ - set of acceptable, independent K-paths in the $\mathbf{S}$ network (it is obvious, that we must add the condition (8) as an additional constraint on $M(\mathbf{i}^s,\mathbf{i}^d)$); $\tau_k{}^d(\cdot,\cdot)$ - described by (6).

In [6] an approximation algorithm (because the min-max optimization problem from (9) is NP-hard) for finding Shortest Independent K-Path (*SIKP*) and its complexity were given. The *SIKP* algorithm generalizes the well-known Dijkstra's shortest path algorithm on the case when we find a few (exactly K) disjoint shortest paths for the K moved objects with objective function (9).

In the problem being described the structure of the network may be dynamically changed during time elapsing (e.g. in consequence of searching and shooting by any opponent). In connection with this fact the unreliable network looks like below :

$$\mathbf{S}_m = \left\langle G, \left\{\overline{H}_w\right\}_{w\in W_G}, \left\{\{t_{w,w'}\right\}_{w,w'\in W_G}, \left\{\overline{F}_{w,w'}\right\}_{w,w'\in W_G}\right\} \right\rangle \tag{10}$$

where $\overline{H}_w(t)$ - probability of possibility of driving by node $w\in W_G$ in interval $(0,t]$; $\overline{F}_{w,w'}(t)$-probability of possibility of driving by arc $(w,w')\in A_G$ in interval $(0,t]$; $t_{w,w'}$ – described by (2).

Let's notice that functions $\overline{H}_w$ and $\overline{F}_{w,w'}$ describe probability distributions of random variables representing times-life of network's nodes and arcs, respectively. It is also assumed that all random variables are independent.

We denote by **T**(**I**) and **P**(**I**), ending moment of redeployment of K columns on the K-path **I** and probability of existing this K-path during redeployment, respectively. We obtain :

$$\mathbf{T}(\mathbf{I}) = \max_{k=1,K} \tau_k^d(\mathbf{I}) \tag{11}$$

$$\mathbf{P}(\mathbf{I}) = \left(\prod_{k=1}^{K} \prod_{r=1}^{R_k} \overline{F}_{i^{r-1}(k), i^r(k)} \Big(\sum_{m=1}^{r} t^k_{i^{m-1}(k), i^m(k)} + \delta_k\Big) \cdot \overline{H}_{i^{r-1}(k)} \Big(\sum_{m=1}^{r-1} t^k_{i^{m-1}(k), i^m(k)} + \delta_k\Big) \right) \cdot$$

$$\cdot \overline{H}_{i^{R_k}(k)} \Big(\sum_{m=1}^{R_k} t^k_{i^{m-1}(k), i^m(k)} + \delta_k\Big) \tag{12}$$

We describe, on the set M($\mathbf{i}^s$,$\mathbf{i}^d$) of all independent K-path from $\mathbf{i}^{\mathbf{s}}$ to $\mathbf{i}^{\mathbf{d}}$ in $\mathbf{S}_m$ network, a vectoral objective function:

$$y(\mathbf{I}) = \langle \ \mathbf{T}(\mathbf{I}) \ , \mathbf{P}(\mathbf{I}) \ \rangle, \qquad \mathbf{I} \in M(\mathbf{i}^s, \mathbf{i}^d) \tag{13}$$

The redeployment problem in unreliable network has been solved as two-criteria optimization problem [7, sect. 4]:

$$\langle \ M(\mathbf{i}^s, \mathbf{i}^d), \ y(\mathbf{I}) \ , R^D \ \rangle \tag{14}$$

where $R^D \subset Y^D(\mathbf{i}^s,\mathbf{i}^d) \times Y^D(\mathbf{i}^s,\mathbf{i}^d)$ is domination relation in the criteria space

$$Y^D(\mathbf{i}^s,\mathbf{i}^d) = \{ y(\mathbf{I}) = \langle \ \mathbf{T}(\mathbf{I}) \ , \mathbf{P}(\mathbf{I}) \ \rangle : \mathbf{I} \in \ M(\mathbf{i}^s,\mathbf{i}^d) \} \tag{15}$$

where :

$$R^D = \left\{ \langle y(\mathbf{I}_m), y(\mathbf{I}_n) \rangle \in Y^D(\cdot,\cdot) \times Y^D(\cdot,\cdot) : \Psi(y(\mathbf{I}_m), y(\mathbf{I}_n)) \right\} \tag{16}$$

$$\Psi(y(\mathbf{I}_m), y(\mathbf{I}_n)) = \begin{cases} 1 & \text{when } (\mathbf{T}(\mathbf{I}_m) \le \mathbf{T}(\mathbf{I}_n)) \wedge (\mathbf{P}(\mathbf{I}_m) \ge \mathbf{P}(\mathbf{I}_n)) \\ 0 & \text{otherwise} \end{cases} \tag{17}$$

We can solve the problem (14) by evaluating *nondominated* or *compromise* K-path. Set of nondominated results is equals to :

$$Y^{ND}(\mathbf{i}^s, \mathbf{i}^d) = \left\{ y(\mathbf{I}) \in Y^D(\cdot,\cdot) : \sim \underset{\substack{z \in Y^D(\cdot,\cdot) \\ z \ne y}}{\exists} \langle z(\mathbf{I}), y(\mathbf{I}) \rangle \in R^D \right\} \tag{18}$$

Set of *nondominated K-paths* is created as inverse image of Y^{ND} set, that is

$$M^{ND}(\mathbf{i}^s,\mathbf{i}^d) = \left\{\mathbf{I} \in M(i^s,i^d) : y(\mathbf{I}) \in Y^{ND}(\cdot,\cdot)\right\} \tag{19}$$

To find compromise K-path we must first evaluate

$$\mathbf{T^*} = \min_{\mathbf{I}\in M(\cdot,\cdot)} \mathbf{T}(\mathbf{I}) \tag{20}$$

$$\mathbf{P^*} = \max_{\mathbf{I}\in M(\cdot,\cdot)} \mathbf{P}(\mathbf{I}) \tag{21}$$

Having **P*** and **T*** values we can write

$$\overline{\mathbf{P}}(\mathbf{I}) = \frac{\mathbf{P}(\mathbf{I})}{\mathbf{P}^*}, \qquad \overline{\mathbf{T}}(\mathbf{I}) = \frac{\mathbf{T}(\mathbf{I})}{\mathbf{T}^*} \tag{22}$$

obtaining normalized vectoral objective function

$$h(\mathbf{I}) = \left\langle \frac{\mathbf{T}(\mathbf{I})}{\mathbf{T}^*}, \frac{\mathbf{P}(\mathbf{I})}{\mathbf{P}^*} \right\rangle \tag{23}$$

Taking into consideration that $\overline{\mathbf{T}}(\mathbf{I}) \geq 1$ and $\overline{\mathbf{P}}(\mathbf{I}) \leq 1$ for $\mathbf{I} \in M(\cdot,\cdot)$ we will obtain normalized ideal result $h^* = \langle 1,1 \rangle$.

Using the method of determining compromise solutions with parameter p≥1 we will assume a metrics ε_p in the space $Y^D(\cdot,\cdot)$ [1] :

$$\varepsilon_p(h^*, h(\mathbf{I})) = \|h^*, h(\mathbf{I})\|_p = \left(\sum_{n=1}^{2} \left| h_n^* - h_n(\mathbf{I}) \right|^p \right)^{\frac{1}{p}} \tag{24}$$

in which n describes count of criterions (for considered case n=2).

For p=1 we obtain:

$$\varepsilon_1(h^*, h(\mathbf{I})) = \left| 1 - \frac{\mathbf{T}(\mathbf{I})}{\mathbf{T}^*} \right| + \left| 1 - \frac{\mathbf{P}(\mathbf{I})}{\mathbf{P}^*} \right| \tag{25}$$

From the reason that $1 - \frac{\mathbf{T}(\mathbf{I})}{\mathbf{T}^*} \leq 0$ and $1 - \frac{\mathbf{P}(\mathbf{I})}{\mathbf{P}^*} \geq 0$ we will obtain that

$$\varepsilon_1(h^*, h(\mathbf{I})) = \frac{\mathbf{T}(\mathbf{I})}{\mathbf{T}^*} - 1 + 1 - \frac{\mathbf{P}(\mathbf{I})}{\mathbf{P}^*} = \frac{\mathbf{T}(\mathbf{I})}{\mathbf{T}^*} - \frac{\mathbf{P}(\mathbf{I})}{\mathbf{P}^*} \tag{26}$$

As compromise result h^0 will be such one, that

$$\varepsilon_1(h^*, h^0(\mathbf{I})) = \min_{\mathbf{I}\in M} \varepsilon_1(h^*, h(\mathbf{I})) = \min_{\mathbf{I}\in M} \left[\frac{\mathbf{T}(\mathbf{I})}{\mathbf{T}^*} - \frac{\mathbf{P}(\mathbf{I})}{\mathbf{P}^*} \right] \tag{27}$$

Compromise K-path $\mathbf{I}^c$ (with p=1) is such K-path for which the formula (27) is satisfied.

Approximate algorithms for solving the problem (14) have been presented in [7, sect.4]. These algorithms look for nondominated and compromise K-path and base on the *SIKP* algorithm described in [6].

3. A simulation model of redeployment

The algorithms indicated in the previous section evaluate solution (K-path **I**) with two characteristics: **T**(**I**) and **P**(**I**). Moreover, we obtain from these algorithms only one K-path and we don't know what to do when this K-path will be destroyed during redeployment. It is also problem, how to evaluate these algorithms (and K-paths in consequence) between themselves. To this end, we define the *examination environment* **B** to examine redeployment process and its algorithms for fixed: count K of columns, time length δ_k of the k-th column, source $\mathbf{i}^s$ and destination $\mathbf{i}^d$ K-nodes :

$$\mathbf{B} = \left\langle \overline{\mathbf{S}}_{\mathbf{m}}, K, \{\delta_k\}_{k=\overline{1,K}}, \mathbf{i}^s, \mathbf{i}^d, A, C \right\rangle \tag{28}$$

where $\overline{\mathbf{S}}_m = \left\langle G, \varnothing, \left\{ t_u, Z = \left\{ \overline{F}_{w,w'} \right\}_{w,w' \in W_G} \right\} \right\rangle$ - unreliable network with destruction of arcs only (because each network with functions described both on the arcs and the nodes may be replaced by network in which functions are described on the arcs only, see [2, p. 188]), $Z \in \mathbf{Z}$ – network destruction plan; A - algorithm for finding the best K-path on each of the experiment stage (experiment stage ≡ each moment when we look for a new K-path from the stage K-node to the fixed destination one $\mathbf{i}^d$), $A \in \mathbf{A}$, where **A** describes set of algorithms for finding the best K-path in the network with stochastic structure (e.g., described in the previous section or others); C – frequency of information receiving about destruction of the network elements (this parameter simulate reconnaissance information), $C\in\mathbf{C}=\{C_1, C_2, C_3\}$, C_1 - information about each change of network elements condition; C_2 - periodic information for fixed time interval; C_3 - time interval between each information is a random variable with fixed probability distribution function and its parameters.

Scheme of one simulation experiment repetition looks like below:

WHILE (~Column_Destroyed) **AND** (~ Exit_Current_Experiment_Repetition) **DO**
  FOR *each moment when an arc of the network was destroyed* **DO**
    IF *there was any column on the destroyed arc* **THEN**

      Column_Destroyed:=TRUE;
      Exit_Current_Experiment_Repetition:=TRUE;

    ELSIF *destroyed arc belongs to the path for any column* **AND** *the arc was destroyed before moment t_c of lately information receiving* **THEN**

      for each column we must withdraw to the lately achieved node and from such new source K-node we find a new optimal K-path I^* to the destination one using A algorithm;

      IF *we can't withdraw to the lately achieved node for any column* **OR** *we can't find a new K-path* **THEN**

        Exit_Current_Experiment_Repetition:=TRUE;

      END IF;
    END IF;
  END FOR;
END WHILE;

It is important to say that we approach to the examination environment **B** as to „black box" where we „throw in" the algorithm A and test it for various network destruction plans Z. After simulations we obtain values of some characteristics for each of the pair (A,Z) (algorithm, network destruction plan) :

- mean value of ending moment of redeployment of K columns (the less the better);

- probability of possibility of achieving the destination K-node $\mathbf{i}^d$ (the greater the better);
- total estimate which relate to both characteristics presented above;

The solution estimation concerns choice of algorithm for which we will obtain the most profitable characteristics (f.e., probability of possibility of achieving the destination K-node, mean value of ending moment of redeployment, etc.). This information will give possibilities, during real redeployment, of using the best, in sense of chosen characteristic, stage algorithm, which we must use to look for a new K-path after receiving information about change of network elements condition belonging to the part of the K-path which was not yet achieved by any column.

Detailed description of simulation model of redeployment was presented in [9].

4. A characterization of computer tool for redeployment planning

The considered tool was created in object-oriented simulation language MODSIM II [3]. This simulation software is being used recent years in Cybernetics Faculty of Military Univesity of Technology to create simulators both for civilian and military applications. To graphical visualization of data, results and convoys movement simulation the object packages SIMGRAPHICS II [4] and SIMOBJECT [5] were used.

The computer tool for redeployment planning fulfils two functions:

- as an environment to carry out research of redeployment planning algorithms (described in section 2) both in deterministic and stochastic structure of the network;

♦ as a tool for computer aiding and evaluating columns movement planning under the condition that potential opponent can affect the network structure (see section 3).

In the Fig.1 appearance of main window of application is presented. The application consists of four fundamental modules:

1. input-output („Plik” (File) menu);
2. network edition („Siec” (Network) menu);
3. analysis of redeployment algorithms for reliable and unreliable network („Algorytm” (Algorithm) menu);
4. simulation analysis of redeployment and its algorithms („Symulacja” (Simulation) menu).

Fig.1. Apperance of main window with map background during realization of simulation experiment repetition

The first module give possibilities of saving actually visible network model (that is, network structure, weights matrix, probabilistic distribution functions described on the network arcs, map background, etc.) or opening selected one.

The second module consists of a few parts. The first one enables to construct network model manually by clicking a left mouse button (nodes and arc adding) or right mouse button (editing node and arc properties, see Fig.2 and Fig.3).

Fig.2 Node properties

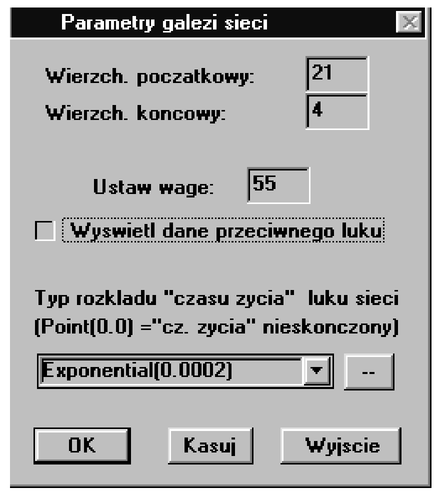

Fig.3 Arc properties

Coordinates of each node are described with respect to left bottom corner of main application window. It is possible to add appropriate map background as Windows "*.bmp" file using options from „Mapa" (Map) menu (see Fig.1).

The second part of this module is a network generator (see Fig.4). This option is very usefull to generate a big networks with demanded characteristics. The option allows to generate a grid and non-grid network. To generate a grid network it is required to set the following characteristics of network:

- number of network nodes;
- geometric average of count of direct node successors in network;
- maximal of count of direct node successors in network;

- probability distribution function of nodes "time-life". It is possible to choose following types of distribution: exponential, erlang, normal, uniform, triangular, weibull, point;
- number of node rows in grid network.

To generate a non-grid network it is required to set properties described above expect the last one.

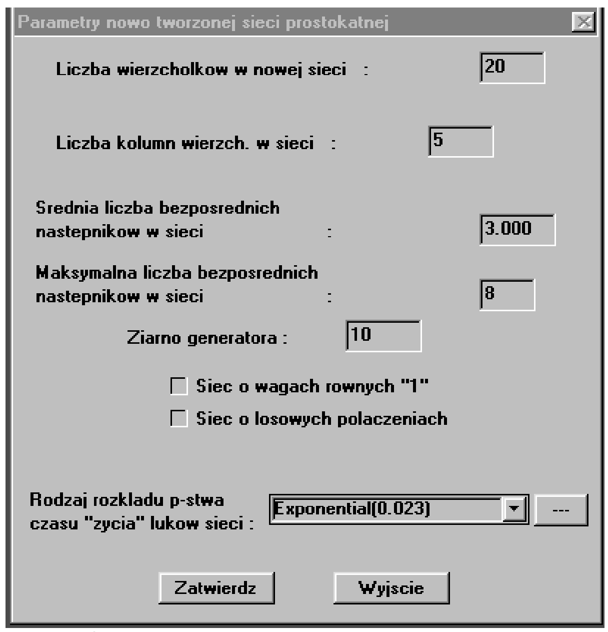

Fig.4 Properties of generated network

After generating a network it is also possible to move network nodes and fit it into real terrain information from the map, obtaining the network which looks like, for example, in Fig.5.

The third module allows to find redeployment plan for K convoys both for reliable and unreliable network using some algorithms included there (see Fig.6). It is required to set following parameters to find redeployment plan for convoys :

- number K of moved convoys;
- K-nodes: source $\mathbf{i}^s$ and destination $\mathbf{i}^d$;
- time length δ_k for each convoy;
- type of algorithm;

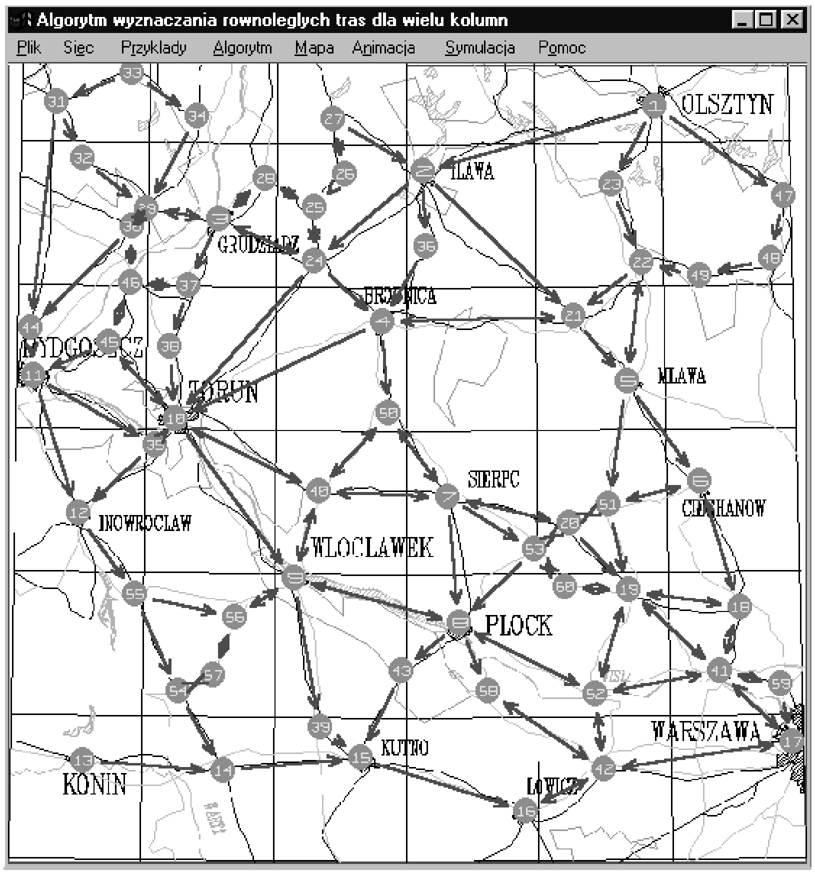

Fig.5 Network for example

The “Metoda Monte Carlo” option in the Fig.6 enables to estimate probability of receiving of optimal solution from basic *SIKP* algorithm for fixed class of the network. Detailed method of carrying out of it with statistical aspects is presented in [7, sect. 3.3.4].

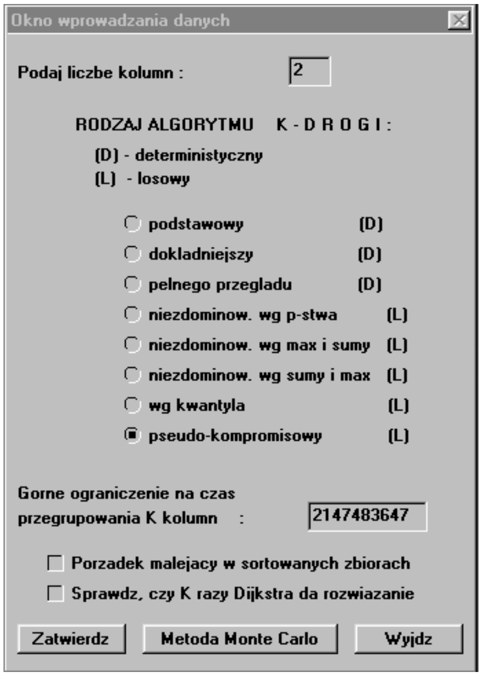

Fig.6 Parameters for K-path determining

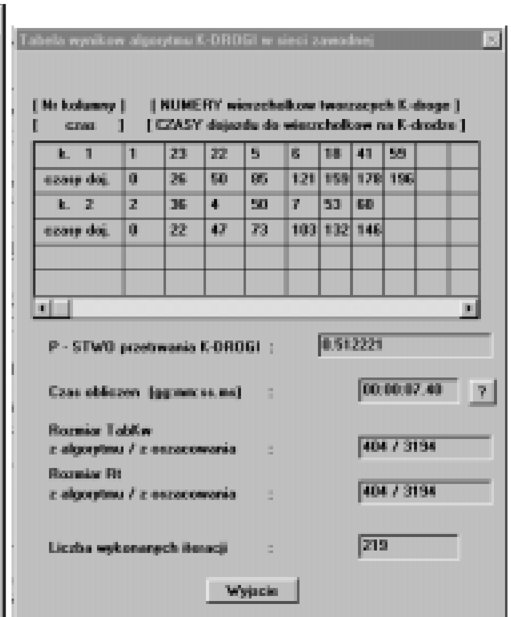

Fig.7 Results of K-path determining for the network from Fig.5

The results of K-path determining are presented in the form shown in the Fig.7. In this figure result of 2-path determining in the network from Fig.5 for $\mathbf{i}^s = \langle 1,2 \rangle$ and $\mathbf{i}^d = \langle 59,60 \rangle$ is given, that is, redeployment paths for two convoys are given : for the first one - from neighbourhood of Olsztyn to neighbourhood of Warszawa, for the second one - from neighbourhood of Iława to neighbourhood of Płock. Additionally, the running time of algorithm and number of algorithm iterations to find solution are shown in the Fig.7. These values give possibilities to check computational complexity of algorithm being run.

The fourth module realizes the simulation model of redeployment described in section 3. To carry out of simulation research for current network model we must set the following parameters (see Fig.8) :

- number K of moved convoys;

- K-nodes: source $\mathbf{i}^s$ and destination $\mathbf{i}^d$;
- time length δ_k for each convoy;
- type of stage algorithm;
- number of simulation repetitions;
- kind of information receiving frequency about network structure condition;

Let's notice that parameters presented above correspond with elements of (28). It is important to say that option "CZESTOSC kontroli stanu sieci" (frequency of network condition checking) in Fig.8 may be used to simulate information receiving from own reconnaissance means. This frequency affects the K-path determining (see scheme of simulation experiment described in section 3).

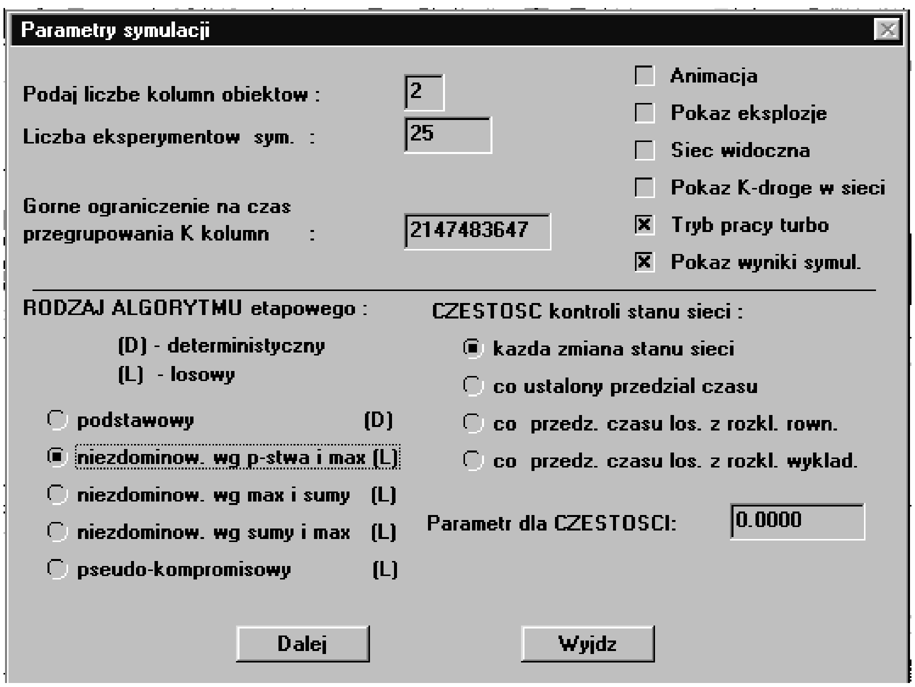

Fig.8 Parameters for simulation model of redeployment planning

Simulation results are being presented like in the Fig.9 and the Fig.10. These results was presented for network from Fig.5, $\mathbf{i}^s = \langle 1,2 \rangle$, $\mathbf{i}^d = \langle 59,60 \rangle$ and simulation parameters from Fig.8. In the Fig.9 the graph of probability distribution described on K-path numbers on which redeployment was possible is drawed. In the Fig.10 detailed information (such as: number of K-path, sequences of nodes belonging to each path, probability of redeployment possibility on K-path, length of K-path) concerning K-paths from the Fig.9 are given.

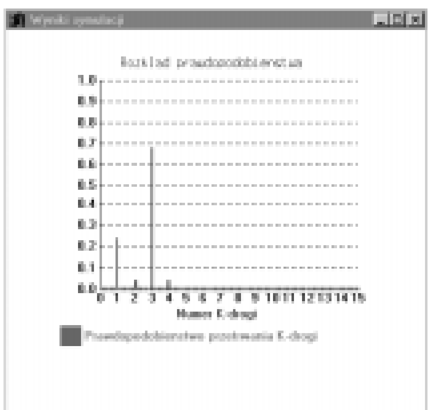

Fig.9 Probability distribution described on K-path numbers

```
   **********************************************
   *    PARAMETRY EKSPERYMENTU SYMULACYJNEGO    *
   **********************************************
Nazwa plik modelu sieci        : B1M_EXP1
Rodzaj badania sieci           : KazdaZmiana
Parametr badania sieci         : 0.00000
Rodzaj algorytmu etap.         : Alg. niezd. wg
                                 rel. p-stwo_max
Dlugosc czasowa kolumny nr 1   : 50
Dlugosc czasowa kolumny nr 2   : 51
Liczba eksperymentow           : 25

   **********************************************
   * CHARAKTERYSTYKI EKSPERYMENTU SYMULACYJNEGO *
   **********************************************
K-Droga nr 1
Droga dla obiektu nr 1     -
Droga dla obiektu nr 2     -
Prawdopodobienstwo K-Drogi = 0.240000
Dlugosc K-Drogi            = 2147483647

K-Droga nr 2
Droga dla obiektu nr 1     - 1 23 22 5 6 18 41 59
Droga dla obiektu nr 2     - 2 36 4 50 7 20 53 60
Prawdopodobienstwo K-Drogi = 0.040000
Dlugosc K-Drogi            = 196

K-Droga nr 3
Droga dla obiektu nr 1     - 1 23 22 5 6 18 41 59
Droga dla obiektu nr 2     - 2 36 4 50 7 53 60
Prawdopodobienstwo K-Drogi = 0.680000
Dlugosc K-Drogi            = 196

K-Droga nr 4
Droga dla obiektu nr 1     - 1 23 22 5 51 19 41 59
Droga dla obiektu nr 2     - 2 36 4 50 7 53 60
Prawdopodobienstwo K-Drogi = 0.040000
Dlugosc K-Drogi            = 199
```

Fig.10 Detailed information from simulation

Mean values of K-path length (or other redeployment characteristics) for each of the pair: algorithm – network destruction plan, enable formulation of non-cooperative two-person game. We may recognize a solution of this game as a decision : which strategy (algorithm) we must use and which strategy (destruction plan) the opponent must use to optimize own redeployment characteristics. In the case of examination only one selected algorithm we should select K-path which has the best redeployment characteristics. For example, if we want to have the most reliable K-path then we must choose K-path number 3 (Fig.9) which has the best reliability (p=0,68) among others. The simulation results are also written in files with "*.mdb" and "*.sym" extensions.

Analysis of obtained simulation results made possible [9]:

- making decision to choose stage algorithm for determining the best K-path in unreliable network to optimize values of selected transport characteristics;
- showing influence of frequency of the received information about network structure condition on values of selected transport characteristics;
- showing influence of columns time length on values of selected transport characteristics;
- evaluating an optimal time interval between received information about network structure condition to minimize gaining cost of this information and simultaneously to optimize values of selected transport characteristics.

5. Conclusions

Application discussed in the paper may be applied to the following problems:

- ♦ movement planning of transportation columns;
- ♦ finding independent paths of tasks from server of parallel (distributed) computations to computers-nodes of such a system;

- finding in computer network the optimal schedule for tasks;
- projecting paths on the printed-circuit boards;
- courier's problem;
- simulation of objects movement on the digital map in unreliable structure of the routes network.

Moreover, this application makes up, as module of manoeuvre planning, part of computer system for supporting and evaluating decisions in combat actions, which was built in Cybernetics Faculty within the confines of the research carried out recent years.

References

[1] AMELJAŃCZYK A., *Optymalizacja wielokryterialna w problemach sterowania i zarządzania*, Ossolineum, Kraków 1984.

[2] KORZAN B., *Elementy teorii grafów i sieci. Metody i zastosowania*, WNT, Warszawa 1978.

[3] *MODSIM II. The Language for Object-Oriented Programming. Reference Manual*, CACI Products Company, La Jolla, California 1994.

[4] *Simgraphics II. User's Manual for MODSIM II*, CACI Products Company, La Jolla, California 1995.

[5] *SIMOBJECT II. Programmer's Guide*, CACI Products Company, La Jolla, California 1995.

[6] TARAPATA Z., *Algorithm for simultaneous finding a few independent shortest paths*, Conference Proceedings of the 9th European Simulation Symposium, Passau 1997, pp. 89-93.

[7] TARAPATA Z., *Algorytmy komputerowego wspomagania planowania przemieszczania równoległego kolumn*, Rozprawa doktorska, Wojskowa Akademia Techniczna, Warszawa 1998.

[8] TARAPATA Z., *Modelling of terrain for necessities of military objects movement simulation*, Bulletin of Military University of Technology **4** (1999). (in the press)

[9] TARAPATA Z., *Simulation method of aiding and estimation of transportation columns movement planning in stochastic environment*, Proceedings of the 13th European Simulation Multiconference, Warsaw 1999. (in the press)

Komputerowe narzędzie wspomagania i oceny planowania przegrupowania konwojów

Zaprezentowano komputerowe środowisko wykorzystywane do oceny planowania przegrupowania konwojów w rzeczywistym terenie. Umożliwia ono wspomagania i ocenę planowania (i jego algorytmów) przemieszczania dowolnych kolumn obiektów w sieci (będącej modelem terenu) o strukturze zarówno deterministycznej, jak i stochastycznej. Opisano matematyczny model przegrupowania K konwojów oraz sformułowano zadania optymalizacyjne dotyczące problemu przegrupowania w sieci zawodnej i niezawodnej. Aby zbadać przegrupowanie i jego algorytmy podczas niszczącego oddziaływania przeciwnika, podano model symulacyjny przegrupowania. Opisano szczegółowo aplikację będącą przzdmiotem zainteresowania oraz przy użyciu opisywanej aplikacji rozwiązano przykład dotyczący rzeczywistego przegrupowania. Podano również możliwości innych, niż wojskowe, zastosowań prezentowanego oprogramowania.