Journal of Theoretical and Applied Computer Science Vol. 7, No. 3, 2013, pp. 21-28
ISSN 2299-2634 (printed), 2300-5653 (online) http://www.jtacs.org

Reconfiguration of logical communication channels

Marek Bolanowski, Andrzej Paszkiewicz

Department of Distributed Systems Rzeszow University of Technology

marekb@prz.edu.pl, andrzejp@prz.edu.pl

Abstract: *In this paper a new reconfiguration of interconnection networks algorithm was proposed for the computational systems and the logical network connection. The proposed solution is based on the presentation of the network and base requirements in the form of linear Diophantine constraints solution in area {0,1}. A set of topologies that meet initial assumptions is obtained as the solution of the system of constrains. In the next step, the final topology is selected, which can be used to reconfigure the interconnection network. It is done using the proposed method of selecting the optimal solution from the set of acceptable solutions, taking into account a certain level of indeterminacy.*

Keywords: *Multibus network, graph, linear Diophantine constraints, reconfiguration, indeterminacy*

1. Introduction

During normal operation of the network system, some of its resources are released and there is a need to re-use them to perform new communication tasks. This is particularly important in systems with a multi-layer topological structure. For example, clusters of network nodes, operating within a logical topology, require the addition/subtraction of additional elements (nodes), adding new logical path for communication, etc. There are a number of reconfiguration tasks that need a flexible method of finding solutions. So far, methods used to find the appropriate structure for the specified constraints have high time complexity, the input data cannot be defined in detail, solutions are not accurate or can be used only in the case of territorial close elements (reconfiguration possible at the local level, not for the entire system) [1][2][3]. Based on the analysis presented in [4] can be concluded that in many applications still dominate the static connection structure in case of interconnection networks, characterized by a weak vulnerability to reconfiguration. In [5] is also presented a method of topology reconfiguration based on the routing function modification. It seems more appropriate, however, to introduce an additional stage of reconfiguration responsible for the formation of clusters connected by logical channels. Figure 1 shows a simple network topology in which the nodes are connected by a physical topology. During the system operation, reconfiguration request is submitted within which the nodes 1, 2, 3, 6, 8 have to be connected using logical links (LL) B_1-B_4. The red numbers refer to the number of connection in nodes which can be used, and red numbers by LL means the number of inputs (bandwidth) in communication channels.

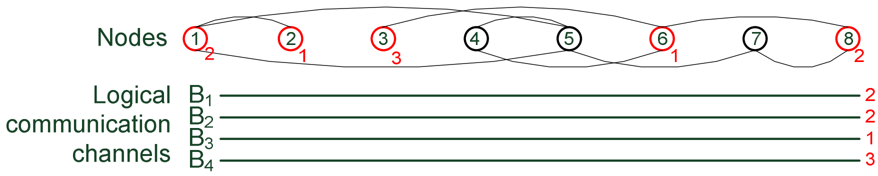

Figure 1. An example of the reconfiguration

After selecting a specific set of possible topologies final selection will be made using the method described in Section 3. Topology will then be implemented using dedicated routing methods available for multi-layer networks [6]. Therefore, it is necessary to develop a reconfiguration method for interconnections in the logical topology layer of the communication system which will be dynamically respond to the reconfiguration requests. Let us introduce the following assumptions:

- proposed method is dedicated to the search a logical topology for transmission system and for interconnection networks used in computational systems,
- the input data are: input degrees of communication buses (optical or logical), and degrees of vertices (logical end nodes or routers),
- calculations are performed: in the case of the computational systems on all nodes for which the topology are searching for, in the case of transmission systems by a dedicated management node,
- all logical channels are accessible from any node of the system.

Let's assume that the logical channels of designed communication system will be using the unidirectional communication channels, therefore a directed graph will be used to represent it.

The directed graph is a pair $G = (V, E)$, for which $V = \{v_1, v_2, \ldots, v_n\}$ is the set of elements called vertices, and set $E \subseteq V \times V$ is the set of elements called arcs where $V \times V$ is Cartesian square of set V. A graph will be called labeled (weighted) when to presentation of the elements from sets $V$and/or Ethere is used not only the identifier, but it is necessary to use the additional variables (edge and/or vertices weight) to store some characteristic or attributes. If the weights of vertices or edges are not specified then such a graph will be called a unlabeled graph.

The adjacency matrix $X = [x_{i,j}]$ will be called square matrix with dimension $n \times n$ elements, where n- number of the graph vertices, which are described as follows: $x_{i,j} = 1$ if in graph G there is an edge connecting vertices v_i and v_j; $x_{i,j} = 0$, if there is no edge connecting vertices v_i and v_j [7][8][3][9][10].

The sum R_i of the elements of the matrix Xin relation to rows i.e.: $R_i = \sum_{\alpha=1}^{n} X_{i\alpha}$, describes the semi-degree output of vertex i, and the same sum in relation to columns, i.e.: $C_i = \sum_{\alpha=1}^{n} X_{\alpha i}$, describes the semi-degree input of vertex i. In this manner, each non negative integer matrix X has a corresponding directed pseudograph with n vertices, described by the set of two dimensional, integer and non-negative vectors of semi-degrees $\overline{d_1} = (R_1, C_1), \ldots, \overline{d_n} = (R_n, C_n)$, for which: $\sum_{i=1}^{n} R_i = \sum_{i=1}^{n} C_i$. First two relation between elements of pseudograph adjacency matrix and semi-degrees its vertices can be considered as a system of constraints with unknown elements and known vectors of the vertices semi-degrees:

$$\begin{array}{ccccccccccc} x_{11} & + & x_{12} & + & x_{13} & + & \cdots & + & x_{1n} & = & R_1 \\ + & & + & & + & & & & + & & \\ x_{21} & + & x_{22} & + & x_{23} & + & \cdots & + & x_{2n} & = & R_3 \\ + & & + & & + & & & & + & & \\ \vdots & & \vdots & & \vdots & & & & \vdots & & \vdots \\ + & & + & & + & & & & + & & \\ x_{n1} & + & x_{n2} & + & x_{n3} & + & \cdots & + & x_{nn} & = & R_n \\ \| & & \| & & \| & & & & \| & & \\ C_1 & & C_2 & & C_3 & & \cdots & & C_n & & \end{array} \tag{1}$$

The system of constraints (1) for given $d_j = (R_j, C_j)$ and $n > 1$ has many solutions in the general case. Truly, the difference between the number of unknown variables and the number of independent constraints is equal $(n-1)^2$. The set $P(\overline{d_1}, \dots, \overline{d_n})$ of non-negative integer solution is limited $0 \leq X_{ij} \leq \min(R_i, C_j)$, for $1 \leq (i,j) \leq n$ and can be described by $(n-1)^2$independent integer parameters. Each solution can be represented by the labeled graph.

From the theory of system of linear constraints, we know that the solution of heterogeneous system of linear constraints can be presented as a sum of some specific solution of heterogeneous system and the solution of homogeneous system corresponded with them.

In order to find the connection networks which has been described by (1) we have to resolve the system of constrains N. We can transform the N to homogeneous system J:

$$N = \begin{cases} x_{11} + x_{12} + x_{13} + \cdots + x_{1n} = R_1 \\ \vdots \quad \vdots \quad \vdots \quad \quad \vdots \quad \vdots \\ x_{n1} + x_{n2} + x_{n3} + \cdots + x_{nn} = R_n \\ x_{11} + x_{21} + x_{31} + \cdots + x_{n1} = C_1 \\ \vdots \quad \vdots \quad \vdots \quad \quad \vdots \\ x_{1n} + x_{2n} + x_{3n} + \cdots + x_{nn} = C_n \end{cases} \Rightarrow J = \begin{cases} x_{11} + x_{12} + x_{13} + \cdots + x_{1n} - (R_1) = 0 \\ \vdots \quad \vdots \quad \vdots \quad \quad \vdots \quad \vdots \quad \quad \vdots \\ x_{n1} + x_{n2} + x_{n3} + \cdots + x_{nn} - (R_n) = 0 \\ x_{11} + x_{21} + x_{31} + \cdots + x_{n1} - (C_1) = 0 \\ \vdots \quad \vdots \quad \vdots \quad \quad \vdots \quad \vdots \quad \quad \vdots \\ x_{1n} + x_{2n} + x_{3n} + \cdots + x_{nn} - (C_n) = 0 \end{cases}$$

For multilayer communication network described in [4] and [5] which have n vertices and represented by bipartition graph (node creates first partition, and logical communication channel creates second partitions) we can reduce the number of equations in the resolved systems of constrains. In order to that the following tree conditions has to fulfilled:

1) there is no loops in the network i.e.: $x_{ij} = 0$ for $i = j$ and $i, j = 1, \dots, n$,
2) there is no connections between nodes of the same type in the bipartition graph which represent the multibus system. If vertices $v_1, v_2, \dots, v_m$ represent the nodes, where $m < n$, and vertices $v_{m+1}, v_{m+2}, \dots, v_n$ represent the communication channels then above condition for the matrix X elements can be written if following form:

$$\begin{matrix} x_{1\ 1} & \cdots & x_{1\ m} \\ \vdots & & \vdots \\ x_{m\ 1} & \cdots & x_{m\ m} \end{matrix} = \begin{matrix} x_{m+1\ m+1} & \cdots & x_{m+1\ n} \\ \vdots & & \vdots \\ x_{n\ m+1} & \cdots & x_{n\ n} \end{matrix} = 0,$$

3) connections in the graph which represent the communications system are undirected i.e.:$x_{1\ m+1} = x_{m+1\ 1},\ x_{1\ n} = x_{n\ 1},\ \dots, x_{m\ n} = x_{n\ m}$.

Taking into consideration the points 1, 2, 3 the system of constrains can be reduced to following equivalent forms

$$\begin{cases} x_{1\ m+1} + \dots + x_{1\ n} = R_1 \\ \vdots \\ x_{m\ m+1} + \dots + x_{m\ n} = R_m \\ x_{1\ m+1} + \dots + x_{m\ m+1} = C_1 \\ \vdots \\ x_{1\ n} + \dots + x_{m\ n} = C_m \end{cases} \Leftrightarrow \begin{cases} x_{m+1\ 1} + \dots + x_{m+1\ m} = R_{m+1} \\ \vdots \\ x_{n\ 1} + \dots + x_{n\ m} = R_n \\ x_{m+1\ 1} + \dots + x_{n\ 1} = C_{m+1} \\ \vdots \\ x_{m+1\ m} + \dots + x_{n\ m} = C_n \end{cases}$$

where $R_1 = R_{m+1}, \dots, R_m = R_n$and $C_1 = C_{m+1}, \dots, C_m = C_n$.

2. Algorithm MTSS{0,1}

The idea of the proposed algorithm is to reduce the set of possible solutions of the system$J$. Classical algorithms [11,12] generate a set consisting of all possible combinations of vectors with coordinates $0-1$, length q, where q is the number of variables in the system of equations J (e.g. $q = 20$, the set consists of 2^{20} vectors) Note that for a given system J the factors matrix has size:$(A+B) \times (A \times B + 1)$:

$$\begin{bmatrix} 11..1 & 00..0 & \dots & 00..0 & p_1 \\ 00..0 & 11..1 & \dots & 00..0 & p_2 \\ & & \dots & & \\ 00..0 & 00..0 & \dots & 11..1 & p_A \\ 10..0 & 10..0 & \dots & 10..0 & m_1 \\ 01..0 & 01..0 & \dots & 01..0 & m_2 \\ & & \dots & & \\ 00..1 & 00..1 & \cdots & 00..1 & m_B \end{bmatrix}.$$

where: A is the number of vertices (CPU, nodes in logical topology,) in the system, B – number of links (buses, interconnections). Algorithm (Fig. 2) consists in determining the solutions to the “upper” part, that is, equations from 1 to$A$, and checking that among these solutions are solutions of the “bottom” part. The solutions are formed in the following way: for the first equation solution vectors $x_1 + x_2 + \dots + x_B = p_1$ are formed, which have a length B, and they have a set of components from set $\{0,1\}$; solutions vectors for the second equation $x_{B+1} + x_{B+2} + \dots + x_{2B} = p_2$ (…) until the last of the equations of the “upper” part $x_{(A-1)B+1} + x_{(A-1)B+2} + \dots + x_{AB} = p_A$. All the solutions vectors from “upper” part have length B and are vectors 0-1. The next step is to create all combinations of solutions one another, which will give us vectors of length $A \cdot B$. To these vectors we must add the last component – the 1 – it corresponds to the factors $[p_1, p_2, p_3, .., p_A]$. In this manner all the possible solutions of the “upper” part have been achieved. *It should be checked* which of these solutions are solution for equation from the “lower” part of the system.

In the case of this algorithm the number of generated vectors is limited and the number of checks from $2^{A \cdot B}$, to $\prod_1^A C_B^{p_i}$.

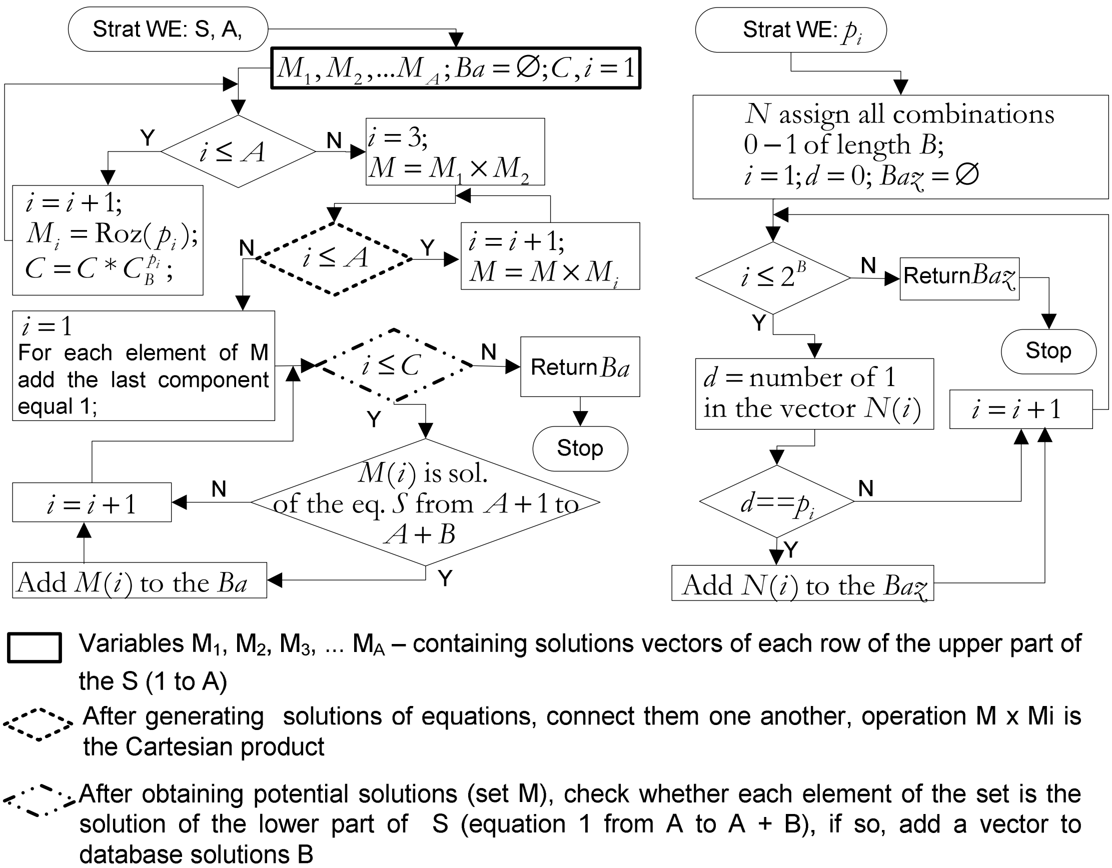

Figure 2. Diagram of proposed algorithm

The problem of obtaining solutions for each equation of the "upper" part can be solved by a single generation of all combinations of vectors of the component $\{0,1\}$ and the length B. Then the vectors are stored in a matrix, and the number of ones in these vector is compared to the absolute value of factor p_i for each equations. The algorithms to generate $B -$ bitwords can also be used, which contain a certain number of ones. The presented algorithm can be implemented in parallel mode (SIMD). To each of the compute nodes (processor cores) the initial values of word B are send (for each node different), and each of the nodes using the algorithm MTSS {0,1}, checks the if the word is the solution. Speedup of the calculation is proportional to the number of computing nodes. Diagram of the algorithm is shown in Fig 2.

3. Selection of the optimal solution under conditions of indeterminacy

In this chapter, proposed method is used to selection of a single solution from a set of possible, acceptable solutions. This method allows you to select one definitive solution taking into account indeterminacy [13].

Let *W* denote the set of topology in vector form that was obtained from the algorithm MTSS. We also assume that $W = \{w_i,\ i = 1 \cdots N\}$, $W \neq \emptyset$. The aim is to obtain only one solution $y \in W$. To determine the set *W* it was necessary to define the topological constraints in the form of R_i and C_i. Thanks to them, was able to weed out the topological structures that were not in our interest. All obtained solutions fulfill these constraints, as a result of the algorithm MTSS. Therefore, to obtain the final solution should be introduced additional constrains (parameters) relating to the possibility of communication of individual solutions.

These constrains will be described in the form of vector $F(w) = \left(f_1, \dots, f_j\right), j \geq 1$. The number and type of parameters depends on the system of communication. Typically these are the values derived on the basis of technical and economic criteria (functions). Here are some examples of evaluation criteria. The first example of the criterion function is *maximization of the traffic capacity*:

$$\max \sum_{sd,r} (r \cdot S_{sd}^r),$$

where: S_{sd}^r – the number of flows type *r* for a route from the node *s* to the node *d*, which can be effective transmitted.

Another example would be *minimization of a maximum light path load*. Let's assume, $L_{\max}$ is the maximum level of interconnection channel congestion. Minimization of the level $L_{\max}$ leads to load distribute between all interconnection channels.

$$\min L_{\max},$$

where: $L_{\max} = \max \sum_{sd,r} \left(r \cdot \lambda_{sd,ij}^r\right) \forall ij \lambda_{sd,ij}^x$ – the number of base transmission flows required between the node s and d which will be routed on the light path ij.

Whereas the primary measure of determining the quality of the links in the context of performed tasks (services) is the quality of the transmission channel or link. In this case, this measure can be represented as a coefficient representing the error rate:

$$P_e = \lim_{e_n \to \infty} \left(\frac{e_o}{e_n}\right),$$

where: e_o is the number of bits received correctly; e_n – total number of bits sent.

On the other hand, the basic and commonly used coefficient of the technical systems efficiency is a life cycle costs call LCC (Life Cycle Cost):

$$LCC = K_C + K_E + K_U$$

where: K_C is the costs of construction (design, implementation, testing, etc.); K_E is operating expenses; K_U describes loss on disposal.

The above coefficients are only examples. When designing real connection structures should be considered much more various functions.

Taking into account above considerations the matrix *J* can be presented in the form of 3-dimensional, while each element j_{kl} will be represented in the form of a vector $j_{kl} = \{j'_{kl}, f_1, \dots, f_n\}$, where j'_{kl} is the value in the range {0,1} obtained by the algorithm MTSS.

Taking as an evaluation criteria $F(w)$ aims to obtain a solution based on a set (vector) of partial evaluation criteria. In this way, there is a problem of contradiction between the criteria. This means that the solution is not sought as minimum or maximum for all the criteria, but is a kind of compromise between them [13]. Thus, the problem mainly rely on the determination of this compromise. This compromise can be represented by the expression:

$$\varphi(f_1, \dots, f_n),$$

where: φ – the convolution principle of the partial functions of efficiency.

Therefore, a compromise introduces a certain level of indeterminacy. Control and reduce this indeterminacy can be achieved through the introduction of a coefficient for functions of efficiency. Then the relative value $\overline{f_i}$ of *i*-th function of efficiency is defined by the expression $\overline{f_i} = a_i f_i$. Weight coefficients have to comply the following requirement:

$$\sum_{i=1}^{n} a_i = 1, a_i \geq 0, \ i = 1, \dots, n. \quad (2)$$

Because objective graduation of the criterion validity does not exist, finding of the concrete value of weight coefficients is impossible. Therefore, weight coefficients can be formally given only in the form of a preferences set for decision-making system, where a decision-making factor is a designer. Therefore, we get:

$$A\{a_j\}|\begin{cases} f_i \succ f_j \Rightarrow a_i > a_j \\ f_i \prec f_j \Rightarrow a_i < a_j. \\ f_i \equiv f_j \Rightarrow a_i = a_j \end{cases} \tag{3}$$

The possibility of define of the permissible ranges of the coefficients value also exists in many design tasks, i.e.: $f_i \in [f_{i_{\min}}, f_{i_{\max}}]$, $i = 1, \dots, n$, and also describing their weight coefficients. If weight coefficients comply conditions (2), (3) and we can define the concrete ranges of acceptable values for them:

$$a_i \in [a_{i_{\min}}, a_{i_{\max}}], \quad i = 1, \dots, n, \tag{4}$$

Defining the degree of validity for the partial coefficients of the quality we can exclusively define the set of weight coefficients: $A = \{a| \sum_{i=1}^{n} a_i = 1, \quad a_i \geq 0, \quad i = 1, \dots, n\}$, which every element complies requirements (2) and (3) [13].

Based on above expressions, we can write that $F(w) = [\bar{f_1}(w), \dots, \bar{f_n}(w)]$, therefore $\varphi(f_1, \dots, f_1) = F(y)$. At this point, we can use the well-known *weighted objectives method* [14] and thus we get the expression:

$$F(w) = \sum_{j=1}^{s} a_j f_j(w) = \sum_{j=1}^{s} \bar{f}_j(w).$$

Based on set theory and taking into account the expression (2) and (3), indeterminacy can be represented as the geometrical presentation of the set of weight coefficients [15]. In this case, the coordinates of the *n*-dimensional plane are represented by the coefficients *a*. However, due to the expression (2), (3) and (4) we obtain the *n*-dimensional space, rather than a single point. Thus, it is necessary to use expressions in the form [15]:

$$F(w) = \frac{1}{\Omega} \int_A (f(w), a) da,$$

where: $\Omega = \int_A da$ – the measure of the area of weight coefficients $a = (a_i, \dots, a_n)$ for the integral form.

On the other hand we can use the minimax form:

$$F(w) = \max_{a \in A} (a_1 f^1(w), \dots, a_n f^n(w)).$$

4. Conclusions

This paper presents a new approach for the automatic reconfiguration of logical communication channels in a network environment with reuse of the released system resources. The use of a two-stage approach to find a solution (the new system topology which contains of unused resources) can significantly accelerate the reconfiguration of the system while preserving the accuracy obtained in the multi-objective optimization. The algorithm is designed to be implemented on an arbiter that manages the complex system. In this way is possible to more efficient use of supercomputing resources, efficient allocation of logical paths in MPLS, or faster solution search for the RCA problem. The proposed solution is part of the new approach of the interconnection network dynamic reconfiguration in complex systems, which assumes that the state of permanent reconfiguration of the system is the normal state of its operation. In further work authors will try to focus on expanding the limited formula of the selection the final topology and implementation described algorithm in the MPLS environment.

References

[1] Gencata A.E., Mukherjee B. *Virtual-topology Adaptation for WDM Mesh Networks under Dynamic Traffic.* IEEE Press, IEEE/ACM Transactions on Networking, Vol. 11, Issue 2, pp. 236-247, Piscataway 2002.
[2] Prathombutr P. *Virtual Topology Reconfiguration in wavelength-routed optical networks*, Thesis presented in Kansas City University, Missouri 2003.
[3] Sreenath N., Gurucharan B.H., Mohan G., Siva Ram Murthy C. *A Two-stage Approach for Virtual Topology Reconfiguration using Path-add Heuristics*. Springer Netherlands, Optical Networks Magazine, Vol. 2, No. 3, Dordrecht 2001.
[4] Penaranda R., Gomez C., Gomez M. E., Lopez P., Duato J.:A New Family of Hybrid Topologies for Large–Scale Interconnection Networks; 2012 IEEE 11th International Symposium on Network Computing and Applications
[5] Sem-Jacobsen F., Lysne O.: *Topology Agnostic Dynamic Quick Reconfiguration for Large-Scale Interconnection Networks*; 2012 12th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing
[6] Oki E., Shiomoto K., Shimazaki D., Yamanaka N., Imajuku W., Takigawa Y: *Dynamic Multi-layer Routing Schemes in GMPLS-Based IP+Optical Networks*; IEEE Communications Magazine; January 2005.
[7] Wilson R.J.: *Wprowadzenie do teorii grafów*; Wydawnictwo Naukowe PWN, 2012.
[8] Hajder M., Loutskii H., Stręciwilk W.: *Informatyka. Wirtualna podróż w świat systemów i sieci komputerowych;*Wydawnictwo WSIZ 2005.
[9] Hajder M. Byczkowska-Lipinska L. Bolanowski M. *Analysis and synthesis of a computational system with reconfigurable multichannel connections.* X International PhD Workshop OWD'2008, 18--21 October 2008.
[10] Hajder M., Bolanowski M. *Optical communication multibus systems*. *Wydawnictwo UMCS*, Annales UMCS Informatica, Sectio Ai Informatica, Vol. 5, ss. 333-342, Lublin 2006.
[11] Kryvyi S. L.: *Algorithms for solving systems of linear diophantine equations in integer domains*; Cybernetics and Systems Analysis; March 2006, Volume 42, Issue 2, pp 163-175.
[12] Крывый С.Л. *Критерий совместности систем линейных диофаитовых уравнений над множеством натуральных чисел*; Допов. НАНУ, Но. 5, сс. 107-112, Киев 1999.
[13] Hajder M., Paszkiewicz A., Bolanowski M. *Intuicja projektanta jako nieokreśloność przy projektowaniu sieci komputerowych*. Współczesne problemy sieci komputerowych – nowe technologie, WNT, s.259-268, Zakopane 2004.
[14] Ehrgott M., Figueira J.R., Greco S. (Eds.), *Trends in Multicriteria Decision Analysis*, International Series in Operations Research & Management Science, Springer, 2010.
[15] Hajder M., Paszkiewicz A. *Selecting communication means in condition of incomplete information for distributed systems with hierarchy of topology*, Polish Journal of Environmental Studies, Vol. 17, No 2A, p.19-23, 2008.