METHOD OF FINDING HAMILTON ROUTES IN TRANSPORT NETWORK

Irina Garashchenko, Anatoliy Panishev

Abstract: This article discusses a solution method for Hamilton Problem, which either finds the task's solution, or indicates that the task is unsolvable. Offered method has significantly smaller requirements for computing resources than known algorithms.

Keywords: Hamilton traveling-salesman problem, minimal cost route, assignment problem, cyclic expansion.

ACM Classification Keywords: Algorithms

Introduction

The paper presents the method of Hamilton's Problem (HP) solution with significantly lesser requirements for computing resources, than for known methods.

Let's formulate the HP.

Let *H* = (*V*, *U*) is a connected graph without closed loops and multiple edges, *V* – set of vertexes, $|V| = n$, *U* – set of edges. An edge $\{i, j\} \in U$ has the weight (cost) $d_{ij} \in Z_0^+$, $i, j = \overline{1, n}$, Z_0^+ is set of non-negative numbers. The symmetric matrix of weights $\left[d_{ij}\right]_n$ completely defines weighed graph *H* = (*V*, *U*), and at this matrix $d_{ij} \in Z_0^+$ if $\{i, j\} \in U$ else $d_{ij} = \infty$, $i \neq j$, $d_{ii} = \infty$, $i = \overline{1, n}$.

Graph *H* = (*V*, *U*) is Hamiltonian if it contains a Hamilton cycle, i.e. the simple cycle which is passing through all vertices of *V* exactly once. The cost $D(z)$ of the Hamilton cycle is equal to the sum of weights *z* edges – it is put in correspondence to this cycle.

The essence of HP is in finding a Hamilton cycle z^* of the minimum cost $D(z^*)$.

The task is *NP* - difficult [1]. Known methods of its solution are presented by schemes of the organization of exhaustive search of all cycles in graph *H* [2-5]. Practical implementation of these methods is problematic even with application of the most powerful computing systems.

The HP is also not always solvable. Therefore a unique method of finding a minimum cost Hamilton cycle z^* is the method which builds z^* if a HP it is solvable, or correctly indicates that graph *H* = (*V*, *U*) is not Hamiltonian.

Let's setup the solution search in two stages. At the first stage sufficient conditions of non-Hamiltonian graph *H* = = (*V*, *U*) are checked. Complexity of checking each of them is estimated by a polynomial of a degree not above 3 from a size of task input data. A HP it is not solvable, if at least one of sufficient conditions of the non-Hamiltonian graph is fulfilled.

It is obvious, that if graph *N* contains final vertices the HP has no solutions. The condition of graph containing concatenation points being non-Hamiltonian is less obvious. Concatenation point is a vertex deleting which together with incidental edges, results in a disconnect graph [6]. Recognition of concatenation points in connected graph *H* = (*V*, *U*) is fulfilled with complexity $O(|V| + |U|)$ [6]. It is not complicated to show, that graph is not Hamiltonian if it contains the bridge defined as such edge deleting which increases the number of connectivity components [6]. It is possible to fulfill recognition of bridges in time $O(|U| + |V|)$ in graph *H* = (*V*, *U*).

At the second stage let's search the solution a HP z^* for the graph which is not containing final vertexes, points of a concatenation and bridges.

The search scheme of the minimum cost Hamilton path on a transport network.

The z^* searching algorithm is constructed according to the main scheme of the branch and bound algorithm. It calls procedure of solving an assignment problem (AP) for an evaluation of lower bounds of magnitude $D(z^*)$ [4]. At the same time, it has features inherent only in it which in the course of branching allow to define, what HP subtasks have no solution.

Let's consider a matrix of weights of HP $\left[d_{ij}\right]_n$. To calculate a lower bound of required magnitude $D(z^*)$, it is required to solve AP for this matrix. But AP with a matrix of weights $\left[d_{ij}\right]_n$ contains a part of elements $d_{ij}=\infty$ and may not have a solution. Therefore, for an evaluation of the lower bounds in the course of finding z^* the algorithm is required , which correctly discovers AP solution or strictly indicates, that AP has no solutions. Modification of Caen-Munkres algorithm works exactly in such way [4].

The algorithm of Caen-Munkres solves AP on a maximum with the assumption that all units $d_{ij}\neq\infty$, $i\neq j$. The input of the updated algorithm is the matrix $\left[d'_{ij}\right]_n$ where $d_{ij}=d-d_{ij}$ if $d_{ij}\neq\infty$ and $d'_{ij}=-\infty$ else, $d\neq\infty$ is a maximum element of matrix $\left[d'_{ij}\right]_n$. Then, if there is solution of AP on a maximum for a matrix $\left[d'_{ij}\right]_n$, it is the solution AP on a minimum for a matrix $\left[d_{ij}\right]_n$. Weights of solutions $C(\pi)$ and $C'(\pi)$ accordingly for matrixes $\left[d_{ij}\right]_n$ and $\left[d'_{ij}\right]_n$ are linked by equality

$C(\pi)=nd-C'(\pi)$.

The updated algorithm of Caen-Munkres searches for an AP solution in a bipartite graph $K(X,Y,E)$, $|X|=|Y|=n$, $|E|=2|U|$ corresponding to a matrix $\left[d'_{ij}\right]_n$ where the vertex $x_i\in X$ is connected to vertex $y_j\in Y$ by an edge (x_i,y_j) with weight $d(x_i,y_j)=d_{ij}\neq\infty$. The AP is solvable if the perfect matching π with the maximum sum of weights of edges is constructed in the graph $K(X,Y,E)$. The AP is unsolvable, if the graph $K(X,Y,E)$ does not contain perfect matchings.

The detailed description of Caen-Munkres algorithm modification is presented in [4]. Its main part is a known procedure of searching of a perfect matching in the bipartite non-weighed graph $H(X,Y,E)$, $|X|=|Y|$, $|E|=2|U|$ with additional means of determining amount of units $d_{ij}=-\infty$ and their disposition in the matrix $\left[d'_{ij}\right]_n$ when AP is insoluble [4, 7]. Algorithm of Caen-Munkres and its modification both are characterized by the labor expenditures estimated by magnitude $O(n^4)$ [7].

The algorithm of finding z^* is fulfilled under the scheme of a branch and bound algorithm offered in [2] for symmetric TSP solution. A combination of this scheme with modification of algorithm of Caen-Munkres is applicable for a solution a HP as well.

Let's assume the perfect matching π is constructed. It supplies the target AP functional with a minimum $C(\pi)$, which is accepted as the lower estimate of weights of a required cycle z^*. Considering the matching π as permutation of columns of the weight matrix, we will present its cyclic expansion, i.e. as a set of non-intersecting cycles. Cyclic expansion of permutation π and the estimate $C(\pi)$ form the rout of a tree of branching. The permutation π presented by a unique cycle, is a z^* – HP solution. It's weight is $C(\pi)$.

Generally the AP solution contains some non-intersecting cycles. Let's select from them a cycle $\sigma=(v_1\ v_2,\ \ldots, v_k\ ,\ v_1)$ with the minimum number of edges. Let's delete all solutions of AP, which contain a cycle σ, without losing any admissible solution z in an initial matrix. It is possible to present set of all solutions of an AP as a partition on k subsets because at least one of k edges (v_1,v_2), $(v_2,v_3)\ldots$, (v_k,v_1) should not be included in z. Let's designate AP with an initial matrix $\left[d_{ij}\right]_n$ as P_0. Then P_0 is divided on k subtasks P_1, P_2, …, P_k. Each of those subtasks correspond in a one-to-one relationship to edges of cycle σ. Weights of edges of σ are set to ∞ in $\left[d_{ij}\right]_n$ matrix , all of the remaining weights are not modified. In the matrix $\left[d'_{ij}\right]_n$ of AP on maximum this edge's weight is assigned $-\infty$. Then if there exists a HP solution it belongs to a set of solutions of any of the subtasks P_1, P_2, …, P_k , which are presented by vertices of a branching tree, emerging from vertex P_0.

In each subtask P_i , $i=\overline{1,\ k}$, it is possible to eliminate not only those solutions which containing the cycle σ, but also the solutions including cycles with vertices from set $S=\{v_1,v_2,\ldots,v_k\}$. To achieve that, let's take the weight matrix of subtask P_i received from P_0 by assigning the element $d_{v_i v_{i+1}}$, $i=\overline{1,\ k}$, $v_{k+1}=v_1$, of weight equal to ∞. In this matrix let's set $d_{v_i v_j}=\infty$ for all $v_i\in S\setminus\{v_i\}$. In the corresponding weight matrix of AP subtask on maximum each unit $d_{v_i v_j}$ gets weight $-\infty$.

For AP on maximum corresponding to subtask P_i , $i=\overline{1,\ k}$, let's apply modification of algorithm of Caen-Munkres to build a permutation π_i if P_i is solvable or find out that it has no solutions. If the subtask P_i is unsolvable the vertex corresponding to it in a tree of branching has no admissible prolongation. Let's suppose that from k subtasks P_i there are k_1 solvable subtasks P_{i_s}, $i\in\{1,\ 2,\ \ldots,\ k\}$, $s=\overline{1,\ k}$, i.e. there were built optimal permutations π_{i_s} and values supplied by them were calculated $C\left(\pi_{i_s}\right)$. Obviously, it is possible to limit the cost of the required Hamilton cycle z^* from below to magnitude

$$C=\min\left\{C\left(\pi_{i_s}\right)\mid i\in\{1,\ 2,\ldots,\ k\},\ s=\overline{1,\ k_1}\right\}.$$

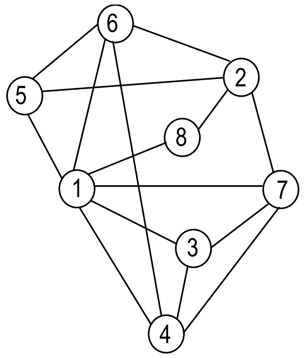
Let's consider a subtask P_q for which $C\left(\pi_q\right)=C$. If the solution π_q is a Hamilton cycle it will be a solution of HP also. Otherwise permutation π_q produces several non-intersecting cycles. Then the node P_q of the solution tree is declared as top of branching [2]. The task P_q is divided into the subtasks which solutions do not contain a minimum length cycle σ from permutation π_q expansion. In solutions of subtasks all cycles generated on set of vertexes of σ are eliminated also. Having fulfilled modification of algorithm of Caen-Munkres for each received subtask P_{q_s}, we will define k_2 solvable subtasks, $q\in\{1,\ 2,\ \ldots, k_1\}$, $s=\overline{1,\ k_2}$. Now a current lower bound of cost of an optimal Hamilton cycle z^* is magnitude

$$C=\min\left\{\min\left\{C\left(\pi_{q_s}\right)\mid q\in\{1,\ 2,\ldots,\ k_1\},\ s=\overline{1,\ k_2}\right\},\ \min\left\{C\left(\pi_{i_s}\right)\mid i\in\{1,\ 2,\ldots,\ k\},\ s\in\{1,\ 2,\ldots,\ k_1\},\ i_s\neq q\right\}\right\},$$

corresponding to the task P_p. If all subtasks received from the task P_q are unsolvable the vertex of branching of a solution tree corresponds to the task P_p with a current estimation $C=\min\left\{C\left(\pi_{i_s}\right)\mid i\in\{1,\ 2,\ldots,\ k\},\ s\in\{1,\ 2,\ldots,\ k_1\},\ i_s\neq q\right\}$.

Further branching for vertex P_p is carried out in the same way as for vertex P_q. Finding the HP solution is completed in one of two cases. In the first case the algorithm finds a Hamilton cycle z^* for HP, which is a solution of a AP with current value of *C.* In the second case algorithm determines that all finite vertexes of the solution tree are not subjects to further branching, and therefore the HP is unsolvable.

Graph *H* has no trailing vertexes and concatenation points. As a result of vertex-edges reorganization (VER) its only chain (1, 8, 2) is replaced by an edge connecting vertexes 1 and 2. Therefore it is impossible to assert that graph *H* not Hamiltonian.



$$\left[d_{ij}\right]_8 = \begin{array}{c} \\ 1\\2\\3\\4\\5\\6\\7\\8 \end{array} \begin{array}{c} \begin{array}{cccccccc} 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 \end{array} \\ \left[\begin{array}{cccccccc} \infty & \infty & 9 & 5 & 10 & 6 & 11 & 4 \\ \infty & \infty & \infty & \infty & 11 & 4 & 9 & 4 \\ 9 & \infty & \infty & 4 & \infty & \infty & 2 & \infty \\ 5 & \infty & 4 & \infty & \infty & 11 & 9 & \infty \\ 10 & 11 & \infty & \infty & \infty & 12 & \infty & \infty \\ 6 & 4 & \infty & 11 & 12 & \infty & \infty & \infty \\ 11 & 9 & 2 & 9 & \infty & \infty & \infty & \infty \\ 4 & 4 & \infty & \infty & \infty & \infty & \infty & \infty \end{array}\right] \end{array}.$$

Fig. 1. Graph *H* = (*V*, *U*) and a matrix of weights of its edges.

Let's apply modified algorithm of Caen-Munkres [4] to solve AP P_0 with an input weight matrix $\left[d_{ij}\right]_8$. The algorithm searches for the AP solution on maximum for a matrix

$$\left[d'_{ij}\right]_8 = \begin{array}{c} \\ 1\\2\\3\\4\\5\\6\\7\\8 \end{array} \begin{array}{c} \begin{array}{cccccccc} 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 \end{array} \\ \left[\begin{array}{cccccccc} -\infty & -\infty & 3 & 7 & 2 & 6 & 1 & 8 \\ -\infty & -\infty & -\infty & -\infty & 1 & 8 & 3 & 8 \\ 3 & -\infty & -\infty & 8 & -\infty & -\infty & 10 & -\infty \\ 7 & -\infty & 8 & -\infty & -\infty & 1 & 3 & -\infty \\ 2 & 1 & -\infty & -\infty & -\infty & 0 & -\infty & -\infty \\ 6 & 8 & -\infty & 1 & 0 & -\infty & -\infty & -\infty \\ 1 & 3 & 10 & 3 & -\infty & -\infty & -\infty & -\infty \\ 8 & 8 & -\infty & -\infty & -\infty & -\infty & -\infty & -\infty \end{array}\right] \end{array}.$$

Optimal solution of the AP P_0 both on maximum and minimum is permutation π = (4, 8, 7, 1, 6, 5, 3, 2), $C'(\pi)$= = 50. The minimum of a target functional of AP P_0 is $C(\pi)$ = 46. It limits from below weight of the required HP solution z^*. Cyclic expansion of permutation π looks like the following: σ_1 = (1, 4, 1), σ_2 = (2, 8, 2), σ_3 = (3, 7, 3), σ_4 = (5, 6, 5). Each cycle of expansion contains two edges. Thus for branching we will select any of four, for example σ_3 = (3, 7, 3). On fig. 2 the branching tree of finding the Hamilton cycle z^* $C(\pi)$ is presented. All calculation results, which form the branching tree, are given in table 1.

The AP P_0 as a result of branching generates two tasks P_1 and P_2 on maximum with weight matrixes received from $\left[d'_{ij}\right]_8$. In matrix $\left[d'_{ij}\right]_8$ for task P_1 $-\infty$ is assigned to element d'_{37} =10, and for task P_2 – to element d_{73} =10. The modified algorithm of Caen-Munkres finds solutions π_1 and π_2 for these tasks and defines weights of received solutions $C(\pi_1)=C(\pi_2)$ = 49. For branching it is possible to select any of subtasks with identical estimates. Let's select task P_1. Permutation π_1 is exhausted with two cycles, from which the cycle (3, 4, 7, 3) has the minimum length. Elimination of AP solutions containing this cycle and all cycles with vertices from set {3, 4, 7} produces three tasks P_{1_1}, P_{1_2}, P_{1_3} on maximum with corresponding weight matrices. In the first matrix $d'_{37} = d'_{34} = -\infty$, in the second $d'_{37} = d'_{43} = d'_{47} = -\infty$, in the third $d'_{37} = d'_{73} = d'_{74} = -\infty$. All three tasks are solvable. Weights of their solutions are $C\left(\pi_{1_1}\right)$ = 57, $C\left(\pi_{1_2}\right)$ = 52, $C\left(\pi_{1_3}\right)$ = 58. By the current moment all final vertices of branching tree P_2, P_{1_1}, P_{1_2} are active P_{1_3}. As, $C = \min\left\{C(\pi_2)\, C\left(\pi_{1_1}\right)\, C\left(\pi_{1_2}\right)\, C\left(\pi_{1_3}\right)\right\}$ = min{49, 57, 52, 58} = 49, the vertex P_2 appears as branching vertex.

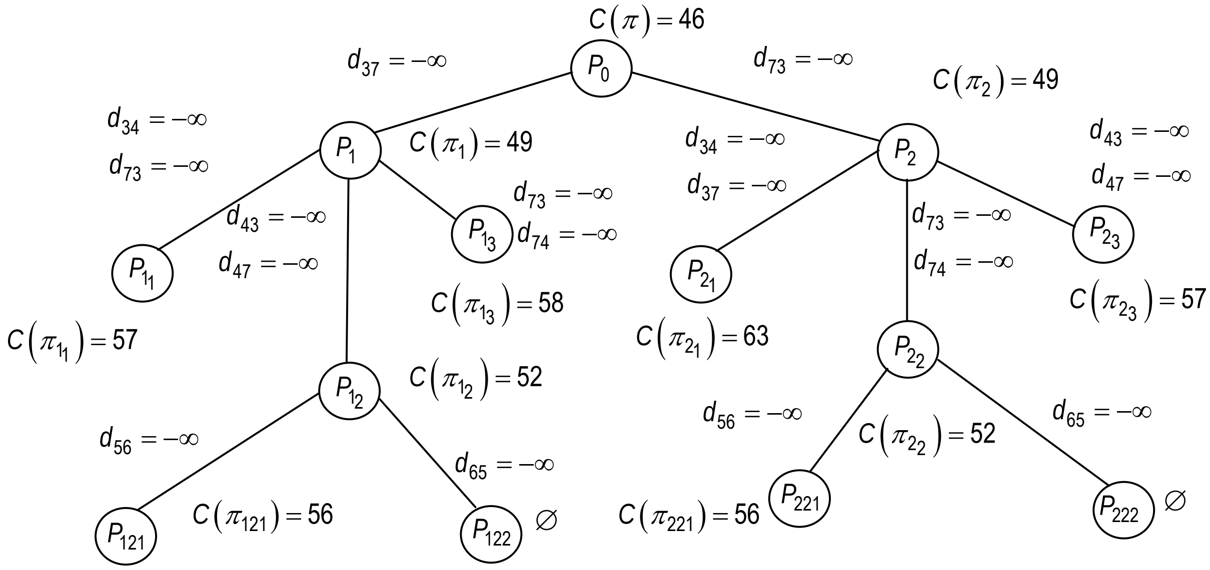


Fig. 2. The branching tree for HP with numerical data of an example 1.

Task P_2 is divided into three subtasks P_{2_1}, P_{2_2}, P_{2_3} with weights of solutions $C\left(\pi_{2_1}\right) = 63$, $C\left(\pi_{2_2}\right) = 52$, $C\left(\pi_{2_3}\right) = 57$. From current set of active vertexes $\left\{P_{1_1}, P_{1_2}, P_{1_3}, P_{2_1}, P_{2_2}, P_{2_3}\right\}$ of the branching tree we will select vertex with the minimum estimate $C = \min$ {57, 52, 58, 63, 52, 57} = 52. Vertices P_{1_2}, P_{2_2} have equal rights to become the branching vertex. For further branching we will select subtask P_{1_2}. In cyclical expansion of permutation π_{1_2} the cycle (5, 6, 5) has the minimum length. Therefore the task P_{1_2} generates two subtasks P_{121} and P_{122}. The modified algorithm of Caen-Munkres builds AP solution of P_{121} in the form of a Hamilton cycle (1, 8, 2, 7, 3, 4, 6, 5, 1) with weight $C(\pi_{121}) = 56$ and determines that task P_{122} has no solutions.

Having built Hamilton cycle with weight 56 we no longer need to branch all active vertices with equal or greater estimates. There remains a single AP P_{2_2} whose weight of solution is equal to 52. Cyclical expansion of permutation π_{2_2} includes a cycle (5, 6, 5), calling two new subtasks P_{121} and P_{222}. AP P_{222} has no solution, and a solution of AP P_{221} is a Hamilton cycle (1, 5, 6, 4, 3, 7, 2, 8, 1) for which the total weight of edges $C(\pi_{221})$ is equal to 56.

The cost of the constructed Hamilton cycles is less, than a lower bound in any final vertex of the branching tree. Thus, the optimal solution of the HP are cycles $z_1^* = \pi_{121}$ = (1, 8, 2, 7, 3, 4, 6, 5, 1), $z_2^* = \pi_{221}$ = (1, 5, 6, 4, 3, 7, 2, 8, 1). □

Table 1.

AP	AP solution	Cycle expansion of AP solution
P_0	π = (4, 8, 7, 1, 6, 5, 3, 2), $C(\pi)=46$	(1, 4, 1) (2, 8, 2) (3, 7, 3) (5, 6, 5)
P_1	π_1 = (5, 8, 4, 7, 6, 2, 3, 1), $C(\pi_1)=49$	(1, 5, 6, 2, 8, 1) (3, 4, 7, 3)
P_2	π_2 = (5, 8, 7, 3, 6, 2, 4, 1), $C(\pi_2)$ = 49	(1, 5, 6, 2, 8, 1) (3, 7, 4, 3)
P_{1_1}	π_{1_1} = (4, 8, 1, 7, 6, 5, 3, 2), $C(\pi_{1_1})=57$	(1, 4, 7, 3, 1) (2, 8, 2) (5, 6, 5)
P_{1_2}	π_{1_2} = (8, 7, 4, 1, 6, 5, 3, 2), $C(\pi_{1_2})=52$	(1, 8, 2, 7, 3, 4, 1) (5, 6, 5)
P_{1_3}	π_{1_3} = (3, 7, 4, 3, 6, 5, 2, 1), $C(\pi_{1_3})=58$	(1, 8, 1) (2, 7, 2) (3, 4, 3) (5, 6, 5)
P_{2_1}	π_{2_1} = (8, 7, 1, 3, 6, 5, 4, 2), $C(\pi_{2_1})=63$	(1, 8, 2, 7, 4, 3, 1) (5, 6, 5)
P_{2_2}	π_{2_2} = (4, 8, 7, 3, 6, 5, 2, 1), $C(\pi_{2_2})=52$	(1, 4, 3, 7, 2, 8, 1) (5, 6, 5)
P_{2_3}	π_{2_3} = (3, 8, 7, 1, 6, 5, 4, 2), $C(\pi_{2_3})=57$	(1, 3, 7, 4, 1) (2, 8, 2) (5, 6, 5)
P_{121}	π_{121} = (8, 7, 4, 6, 1, 5, 3, 2), $C(\pi_{121})=56$	(1, 8, 2, 7, 3, 4, 6, 5, 1)
P_{122}	AP is unsolvable	–
P_{221}	π_{222} = (5, 8, 7, 3, 6, 4, 2, 1), $C(\pi_{5_2})=56$	(1, 5, 6, 4, 3, 7, 2, 8, 1)
P_{222}	AP is unsolvable	–

Conclusion

The method has been implemented in C# programming language. For perfomance tests we have used Celeron 1.8GHz PC. Solution time for HP with cost matrix size around 60 in the worst case does not exceed 3 minutes.

Bidliography

[1] Гэри М., Джонсон Д. Вычислительные машины и труднорешаемые задачи. – М.: Мир, 1982. – 416с.

[2] Кристофидес Н. Теория графов. Алгоритмический подход. – М.: Мир, 1978. – 432 с.

[3] Бондаренко М.Ф., Белоус Н.В., Руткас А.Г. Компьютерная дискретная математика. – Харьков: «Компания СМИТ», 2004. – 480с.

[4] Панишев А.В., Плечистый Д.Д. Модели и методы оптимизации в проблеме коммивояжера. – Житомир: ЖГТУ, 2006. – 300с.

[5] Вьялицин А. А. О перечислении гамильтоновых циклов // Дискретная математика. – 1991. – Т.3, вып. 3. – С. 46 – 49.

[6] Рейнгольд Э., Нивергельт Ю., Део Н. Комбинаторные алгоритмы. Теория и практика. – М.: Мир, 1980. – 476 с.

[7] Свами М., Тхуласираман К. Графы, сети, алгоритмы. – М.: Мир, 1984. – 454 с.

Authors' Information

Irina Vladimirovna Garashchenko *– teacher-assistant of the chair of informatics and computer modelling , aspirant, Zhitomir State Technological University, Chernyahovskogo 103, Zhitomir, Ukraine, 10005; e-mail: irinagar@rambler.ru*

Anatoliy Vasilyovich Panishev *– head of the chair of informatics and computer modelling, professor, doctor of technical sciences, Zhitomir State Technological University, Chernyahovskogo 103, Zhitomir, Ukraine, 10005*