Computer Science Journal of Moldova, vol.9, no.3(27), 2001

The Multiobjective Bottleneck Transportation Problem

Alexandra Tkachenko Artiom Alhazov

Abstract

In this paper we give the solution methods for a multicriterial transportation problem of a nonlinear type. We would like to note that the problems of this type do not have any classical solution algorithms.

The article consists of two parts dealing with 2 and 3 objectives respectively, one being non-linear of "bottleneck" type, and the rest being linear ones. Definitions of efficient and extreme efficient solutions are introduced and a separate solution algorithms for these models are described. The correctness theorems for the algorithms are proved. Examples solved by the computer programs implementing the algorithms are included.

Key words: efficient solution, efficient plan, "bottleneck" transportation problem, extreme efficient solution, the bottleneck model.

1 Bicriteria Problem

In this part we are solving 2-objective bottleneck model. A formal definition of the problem is made. A solution algorithm of a bicriteria transport nonlinear problem is proposed. First, a possible solution is found and optimized by time. Then, the time is sequentially traded for the cost, thus obtaining the set of all efficient solutions. A theorem is proved stating the correctness of the algorithm finding the set of efficient solutions. An example with the results of a computer program for this new and very efficient algorithm is included.

1.1 Defining the Problem

We consider both the time and the cost objectives. The problem is defined by supplies a, demands b, the costs c and the times t. The solution consists of efficient (optimal by Pareto) transportation plans x. $a = a(m)$, $b = b(n)$, $c = c(m,n)$, $t = t(m,n)$. The problem is to find $x = x(m,n)$ so that:

$$x_{ij} \geq 0 \qquad \forall i = \overline{1,m} \qquad \forall j = \overline{1,n} \tag{1}$$

$$\sum_{j=1}^{n} x_{ij} = a_i \qquad \forall i = \overline{1,m} \tag{2}$$

$$\sum_{i=1}^{m} x_{ij} = b_j \qquad \forall j = \overline{1,n} \tag{3}$$

$$C(x) = \sum_{i=1}^{m} \sum_{j=1}^{n} c_{ij} x_{ij} \qquad \to min \tag{4}$$

$$T(x) = \max_{(i,j)} \{t_{ij} : x_{ij} > 0\} \qquad \to min \tag{5}$$

Def 1 *x that satisfies 1,2,3 is called a plan.*

Let x^1, x^2 be plans.

Def 2 *It is said that x^1 dominates x^2 (not. $x^1 \preceq x^2$), if $C(x^1) \leq C(x^2)$ and $T(x^1) \leq T(x^2)$*

Def 3 *It is said that x^1 is better than x^2 (not. $x^1 \prec x^2$), if $x^1 \preceq x^2$ and not($x^2 \preceq x^1$)*

Def 4 *It is said that x^1 is equivalent to x^2 (not. $x^1 \sim x^2$), if $x^1 \preceq x^2$ and $x^2 \preceq x^1$*

Def 5 *A plan x^0 is called efficient, if no better plan exists.*

The answer to the BTP is a set of efficient plans. To simplify the problem, we shall limit ourselves to finding one efficient plan among all the equivalent ones.

A plan exists if and only if

$$\sum_{i=1}^{m} a_i = \sum_{j=1}^{n} b_j \tag{6}$$

Otherwise an extra row or column is added.

Def 6 *A BTP is called regular, if its solution does not have any degenerate transportation plans.*

Remark 1 *If no such proper subsets $I \subset \{1..m\}$ and $J \subset \{1..n\}$ exist, that*

$$\sum_{i \in I} a_i = \sum_{j \in J} b_j, \tag{7}$$

then the BTP is regular.

In case of a regular BTP, any basic plan will only have non-zero components.

1.2 General Algorithm of Regular BTP Solution

1. Find any plan (for example with the fastest element method)
2. Optimize the plan by time (for example with the time-potentials method)
3. Optimize the plan by cost (applying the modified cost-potentials method) Among all possible transitions we select the one so that:
   (a) $\Delta_{ij} > 0$ (so that the cost is decreased)
   (b) Choosing minimal t_{ij}
   (c) Choosing maximal Δ_{ij} of those satisfying 3a and 3b.

Theorem 1 *A plan x in phase 3 of the solution is efficient if and only if no such transition exists that $\Delta_{ij} > 0$ and $t_{ij} \leq T(x)$.*

Proof: By definition, x is efficient if no better plan x^1 exists, i.e.

1. $C(x) = min\{C(x^1) : x^1$ - possible,$T(x^1) \geq T(x)\}$
2. $T(x) = min\{T(x_1) : x^1$ - possible,$C(x^1) \geq C(x)\}$.

After phase 2, the fastest plan x^0 is obtained. The cheapest of all fastest plans is the first efficient plan; it is obtained after zero or more transitions with $\Delta_{ij} > 0$ and $t_{ij} \leq T(x^0)$. Starting here, condition 2 is verified. Let us verify that condition 1 holds for an arbitrary plan x by considering the following linear programming problem:

$$y_{ij} \geq 0 \quad \forall i = \overline{1,m} \quad \forall j = \overline{1,n} \tag{8}$$

$$\sum_{j=1}^{m} y_{ij} = a_i \quad \forall i = \overline{1,m} \tag{9}$$

$$\sum_{i=1}^{n} y_{ij} = b_j \quad \forall j = \overline{1,n} \tag{10}$$

$$C(y) = \sum_{i=1}^{m} \sum_{j=1}^{n} c_{ij} y_{ij} \quad \rightarrow min \tag{11}$$

$$t_{ij} \geq T(x) \Rightarrow \quad y_{ij} = 0 \quad \forall i = \overline{1,m}\ \forall j = \overline{1,n} \tag{12}$$

It is obvious that y=x satisfies 8,9,10,12. y=x is a locally optimal solution since no transitions $x \rightarrow y$ exist so that $C(y) < C(x)$. It suffices to notice the set of possible solutions is convex to state that y is a globally optimal solution, i.e. condition 1 holds.
End of proof.

1.3 Irregular BTP

Solving an irregular BTP with the method described above we actually use a slightly different time criterion

$$T_1(x) = \max_{(i,j)}\{t_{ij} : x_{ij} - basic\} \rightarrow min \tag{13}$$

instead of

$$T(x) = \max_{(i,j)}\{t_{ij} : x_{ij} > 0\} \rightarrow min \tag{14}$$

It can happen that $T_1(x) > T(x)$ in case $x_{ij} = 0$ for a basic component (i,j), which is a point of maximum time.

Two approaches are listed below:

1. Enlarging the class of BTP where $T_1 = T$.
2. Reducing irregular BTP to a few regular BTPs.

We now proceed to approach 2.
Suppose there exist such subsets $I_1 \subset \{1..m\}$ and $J_1 \subset \{1..n\}$, that

$$\sum_{i \in I_1} a_i = \sum_{j \in J_1} b_j \tag{15}$$

We define I_2 and J_2 to be complementary to I_1 and J_1:
$I_2 = \{1..m\} \backslash I_1, J_2 = \{1..n\} \backslash J_1$. Two choices are possible:

1. $x_{ij} > 0$, the regular method can be applied, giving a set X of solutions.
2. $x_{ij} = 0$, deleting (i,j) from the basis-tree $G(\{1..m\}, \{1..n\})$ decomposes it into two trees: $G_1(I_1, J_1)$ and $G_2(I_2, J_2)$.

Both problems are solved separately, yielding sets X_1 and X_2 of efficient solutions. (Notice that these sub-problems can also be irregular.) The pairs of efficient solutions x^1 and x^2 are combined into x' with $C(x') = C(x^1) + C(x^2)$, $T(x') = max(T(x^1) + T(x^2))$. Thus a set X' is obtained. Finally, the incomparable plans are selected from $X \cup X'$ by excluding plans that are worse than others, giving the set X_e, which is indeed the set of efficient solutions.

1.4 Enlarging the Class of Regular BTPs

Suppose there exist such subsets $I \subset \{1..m\}$ and $J \subset \{1..n\}$, that

$$\sum_{i \in I_1} a_i = \sum_{j \in J_1} b_j \tag{16}$$

A decomposition is possible by a basic zero at $(p, q) \in I_1 \times J_2 \cup I_2 \times J_1$. Decomposition apriori does not have to be considered if T_1 does not change when we exclude t_{pq}:

$$T_1(x) = \max_{(i,j)} \{t_{ij} : x_{ij} - basic\} \quad -> min \tag{17}$$

$$T_2(x) = \max_{(i,j)} \{t_{ij} : x_{ij} - basic, (i,j) \neq (p,q)\} \quad -> min \tag{18}$$

$T_1 = T_2$ if $t_{pq} \le max(T(I_1, J_1), T(I_2, J_2))$.
However, $T(I_1, J_1) \le max\{\max_{i\in I_1}(\min_{j\in J_1} tij); \max_{j\in J_1}(\min_{i\in I_1} t_{ij})\}$,
and $T(I_2, J_2) \le max\{\max_{i\in I_2}(\min_{j\in J_2} t_{ij}); \max_{j\in J_2}(\min_{i\in I_2} t_{ij})\}$.

Conclusion: if $t_{pq} \le max\{\max_{i\in I_1}(\min_{j\in J_1} t_{ij}); \max_{j\in J_1}(\min_{i\in I_1} t_{ij});$
$\max_{i\in I_2}(\min_{j\in J_2} t_{ij}); \max_{j\in J_2}(\min_{i\in I_2} t_{ij})\}$, then the regular method can be applied.

1.5 Example

Consider the following cost-time problem: Dimensions=6,7.

Supply, Demand, Cost=

9	10	20	2	6	4	3	15
8	8	15	2	8	2	6	7
4	5	4	9	7	6	2	45
12	15	6	3	10	7	2	30
3	6	10	4	11	8	4	12
7	9	7	5	8	2	5	16
20	13	11	27	9	5	40	$b_j \backslash a_i$

Time=

12	13	34	7	8	29	19
7	18	36	40	38	6	10
11	20	30	21	21	29	31
17	12	39	31	5	36	12
17	17	32	36	22	16	14
15	38	16	33	23	30	29

Let us research the regularity question.
There are 35 ways to split the problem into just 2 sub-problems, i.e. there exists 70 subsets $I_1 \subset \{1..m\}$ and $J_1 \subset \{1..n\}$ so that

$$\sum_{i\in I_1} a_i = \sum_{j\in J_1} b_j \tag{19}$$

However, for every such splitting, $\exists(p, q) \in I_1 \times J_2 \cup I_2 \times J_1$:
$t_{pq} \le max\{\max_{i\in I_1}(\min_{j\in J_1} t_{ij}); \max_{j\in J_1}(\min_{i\in I_1} t_{ij}); \max_{i\in I_2}(\min_{j\in J_2} t_{ij}); \max_{j\in J_2}(\min_{i\in I_2} t_{ij})\}$, so the regular method suffices.

Solving the problem with the regular method we obtain 8 efficient solutions.

1. $T = 21; C = 548$

0	0	0	15	0	0	0
0	0	0	0	0	5	2
11	13	0	12	9	0	0
0	0	0	0	0	0	30
4	0	0	0	0	0	8
5	0	11	0	0	0	0

2. $T = 23; C = 538$

0	0	0	15	0	0	0
0	0	0	0	0	5	2
16	13	0	12	4	0	0
0	0	0	0	0	0	30
4	0	0	0	0	0	8
0	0	11	0	5	0	0

3. $T = 29; C = 533$

0	0	0	15	0	0	0
0	0	0	0	0	5	2
11	13	0	12	9	0	0
0	0	0	0	0	0	30
9	0	0	0	0	0	3
0	0	11	0	0	0	5

4. $T = 30; C = 508$

0	0	0	15	0	0	0
0	0	0	0	0	5	2
8	13	11	12	1	0	0
0	0	0	0	0	0	30
12	0	0	0	0	0	0
0	0	0	0	8	0	8

5. $T = 31; C = 432$

0	0	0	15	0	0	0
0	2	0	0	0	5	0
8	11	11	0	0	0	15
0	0	0	12	0	0	18
12	0	0	0	0	0	0
0	0	0	0	9	0	7

6. $T = 33; C = 425$

0	0	0	15	0	0	0
0	2	0	0	0	5	0
8	11	11	0	0	0	15
0	0	0	5	0	0	25
12	0	0	0	0	0	0
0	0	0	7	9	0	0

7. $T = 38; C = 423$

0	0	0	15	0	0	0
0	0	0	0	2	5	0
8	13	11	0	0	0	13
0	0	0	3	0	0	27
12	0	0	0	0	0	0
0	0	0	9	7	0	0

8. $T = 40; C = 402$

0	0	0	15	0	0	0
0	0	0	7	0	0	0
8	13	11	0	0	0	13
0	0	0	3	0	0	27
12	0	0	0	0	0	0
0	0	0	2	9	5	0

2 Tricriteria Problem

This part is dedicated to the transportation model with three objectives, a non-linear one of “bottleneck” type and two linear criteria. The algorithm consists of reducing the main problem to a problem with one bottleneck objective and a sequence of problems with 2 linear criteria.

Finally, we join their results to construct the set of extreme efficient solutions. A theorem proving the validity of this algorithm finding the set of extreme efficient solutions is stated and proved. An example solved by the computer program implementing the algorithm is included.

2.1 Defining the Problem

We shall now consider two linear (cost-type) objectives and one bottleneck objective. The problem is defined by supplies a, demands b, the costs c and d and the times t. The solution consists of efficient (optimal by Pareto) transportation plans x. Since the set of solutions usually has a power of continuum, we shall only list the extreme efficient solutions. All the efficient solutions are among the linear combinations of extreme efficient ones.

$a = a(m)$, $b = b(n)$, $c = c(m,n)$, $d = d(m,n)$, $t = t(m,n)$. The problem is to find $x = x(m,n)$ so that:

$$x_{ij} \geq 0 \qquad \forall i = \overline{1,m} \qquad \forall j = \overline{1,n} \tag{20}$$

$$\sum_{j=1}^{n} x_{ij} = a_i \qquad \forall i = \overline{1,m} \tag{21}$$

$$\sum_{i=1}^{m} x_{ij} = b_j \qquad \forall j = \overline{1,n} \tag{22}$$

$$C(x) = \sum_{i=1}^{m} \sum_{j=1}^{n} c_{ij} x_{ij} \quad \rightarrow min \ - \ cost1 \ criterion \tag{23}$$

$$D(x) = \sum_{i=1}^{m} \sum_{j=1}^{n} d_{ij} x_{ij} \quad \rightarrow min \ - \ cost2 \ criterion \tag{24}$$

$$T(x) = \max_{(i,j)} \{t_{ij} : x_{ij} > 0\} \quad \rightarrow min \ - \ time \ criterion \tag{25}$$

2.2 Solving the 2-Linear-Criteria Problem

In this paragraph we shall consider the problem defined by 21-25 . A solution of a two criteria problem is a set of extreme efficient solutions.

Finding the first and last extreme efficient plans: Construct any plan and optimize it by C-potentials method. Optimize it further by modified D-potentials method, considering only cells (i, j) with $\Delta_{ij} = 0$.

A plan x^c will be constructed in that way. Then,

1. $C_0 := C(x^c) = min\ C(x)$
2. $D_1 := D(x^c) = min\{D(x') : C(x') = min\ C(x)\}$.

Similarly, find the last extreme efficient plan x^d that satisfies

1. $D_0 := D(x^d) = min\ D(x)$
2. $C_1 := C(x^c) = min\{C(x') : D(x') = min\ D(x)\}$.

If $C(x^c) = C(x^d)$ and $D(x^c) = D(x^d)$, then x^c and x^d are equivalent (or the same) and $E = \{x^c\}$ is the answer. Otherwise continue. The set E of the found extreme efficient plans now contains $\{x^c, x^d\}$. The set L of the untried segments in the $C \times D$ space is currently $\{(p_c, p_d)\}$ where $p_c = (C_0, D_1), p_d = (C_1, D_0)$. The set F of the optimized segments is currently empty.

Finding intermediate extreme efficient plans: Pick $(p_1, p_2) \in L$, $p_1 = (c_1, d_1)$, $p_2 = (c_2, d_2)$, exclude it from L. Solve a transport problem with a single criterion:

$$C'(x) = |d_1 - d_2|C(x) + |c_1 - c_2|D(x)\ or \tag{26}$$

$$C'(x) = \sum_{i=1}^{m}\sum_{j=1}^{n}(|d_1 - d_2|c_{ij} + |c_1 - c_2|d_{ij})x_{ij} \rightarrow min \tag{27}$$

A solution y will be obtained. $p_3 = (C(y), D(y))$. If $p_3 = p_1$ or $p_3 = p_2$, then the new plan is equivalent to another one already found, include (p_1, p_2) in F. Otherwise include (p_1, p_3) and (p_3, p_2) in L and include y in E. Repeat the procedure for the next element in L until L is empty. The set E will consist of all extreme efficient plans (up to equivalence).

2.3 Validity and Finiteness

Theorem 2 *A point $z \in E \Leftrightarrow z$ is an extreme efficient point.*

Sufficiency The algorithm may terminate either upon establishing the equivalence of x^c and x^d, or upon exhaustion of the set L. In the former case, the only point recorded is obviously an extreme efficient point. In the latter case, x^c and x^d both are clearly extreme efficient points. Therefore, the element $\{x^c, x^d\}$ of L at the first iteration denotes two extreme efficient points. Suppose at current iteration each element $(r, s) \in L$ corresponds to extreme efficient points r and s. Therefore $a_1 = |d^{(r)} - d^{(s)}| \geq 0$ and $a_2 = |c^{(r)} - c^{(s)}| \geq 0$ and hence if the new point is found not equivalent to r or s, then it is a result of minimizing a positively weighted average of the objective functions. So, the new point recorded is an extreme efficient point. Thus at each iteration each element corresponds to two extreme efficient points, ensuring that all points included in E, are extreme efficient.

Necessity In case $x^c \sim x^d$ there is no other extreme efficient point. We now proceed to the other case. Let r and s be corresponding to $(r, s) \in L \cup F$. Define the sets

$$A(r,s) = \{z : z \geq \lambda r + (1-\lambda)r \quad , \lambda \in R^1\} \tag{28}$$
$$B(r,s) = \{z : z \leq \lambda r + (1-\lambda)r \quad , 0 \leq \lambda \leq 1\} \tag{29}$$

It will be now proved by induction that if k is an extreme efficient point, then at each iteration,

$$k \in \cup_{(r,s)\in L\cup F} B(r,s) \tag{30}$$

It is clearly true at iteration 1 when $L \cup F = \{x^c, x^d\}$. Suppose at some iteration m, $(r, s) \in L$ is chosen. Then the algorithm either find a new extreme efficient point or moves (r, s) from L to F. In the former case $L \cup F$ is changed because of the deletion of (r, s) from L and inclusion of (r, k) and (k, s) in the set L. Since the feasible set in the objective space is convex, there is no efficient point in the interior of the convex hull generated by r,k and s. Thus if $k \in B(r,s)$, then indeed, $k \in B(r,k) \cup B(k,s)$. Hence at iteration $m+1$ also

$$k \in \cup_{(r,s)\in L\cup F} B(r,s) \tag{31}$$

The algorithm enters (r,s) in the set F only if the minimization process, where the linear function to be minimized has the same slope as that of the line passing through r and s, gives the minimum to be r and s. Therefore, the entire set $E \subset A(r,s)$ for each $(r,s) \in F$ Thus if k is an extreme efficient point $k \in \cap_{(r,s)\in L\cup F} A(r,s)$ and also $k \in \cup_{(r,s)\in L\cup F} b(r,s)$. Since $L \cup F = F$ at the termination of the algorithm, $\exists (r,s) \in F : k \in A(r,s) \cap B(r,s)$. In other words for some $\lambda_k (0 \leq \lambda_k \leq 1), k \in \lambda_k r + (1-\lambda_k)s$ and since k is an extreme point $k = r$ or s. Hence k is indeed found by the algorithm.

End of proof. The finiteness of the algorithm results from the finiteness of the extreme points.

2.4 Solving the Tricriteria Problem

Solve the 2-linear-criteria problem. Each plan in E is evaluated by 3 criteria: C, D and T. If E is empty, stop. (It does not happen at the first time) Determine $T* = max\{T(x) : x \in E\}$. Redefine $c_{ij} = \inf$ and $d_{ij} = \inf$ for each cell with $t_{ij} \geq T*$. Repeat the procedure until E is empty. The set of non-dominated points among all the plans found forms the solution of the problem.

2.5 Practical Modification

Instead of dealing with infinite costs, it is useful to initially calculate the best time, and start solving each 2-linear-criteria problem from the best-time solution, simply blocking the cells with $t_{ij} \geq T*$.

2.6 Example

Consider the following 3-criteria problem: Dimensions= 3, 4.

Time, Supply, Demand=

10	95	73	52	8
68	66	30	21	19
37	63	19	23	17
11	3	14	16	$b_j \backslash a_i$

Cost1, 2=

1	2	7	7
4	4	3	4
1	9	3	4
5	8	9	10
8	9	4	6
6	2	5	1

Regularity question: the problem can be split in
$I_1 = \{1,3\};\ J_1 = \{1,3\};\ I_2 = \{2\};\ J_2 = \{2,4\}$ or
$I_1 = \{1,2\};\ J_1 = \{1,4\};\ I_2 = \{3\};\ J_2 = \{2,3\}$.
In case 1 apriori $T(X_2) \geq 66 \geq 23 = t_{34}$
In case 2 apriori $T(X_2) \geq 63 \geq 23 = t_{34}$.
We then proceed with the regular method.

The problem solution gives 9 extreme efficient points listed below. Iteration1: solutions 1,2,3,4,6; Iteration2: solutions 7,8,9,5,6; Iteration3: solutions 7,8,9.

Transportation plans and corresponding criteria=

1. (143,265,95)				2. (156,200,95)				3. (176,175,95)			
5	3	0	0	5	3	0	0	0	3	5	0
6	0	0	13	6	0	13	0	11	0	8	0
0	0	14	3	0	0	1	16	0	0	1	16
4. (186,171,95)				5. (202,173,73)				6. (208,167,73)			
0	2	6	0	0	0	6	2	0	0	8	0
11	0	8	0	11	0	8	0	11	2	6	0
0	1	0	16	0	3	0	14	0	1	0	16
7. (158,283,68)				8. (172,213,68)				9. (178,203,68)			
8	0	0	0	8	0	0	0	6	0	0	2
3	0	0	16	3	0	14	2	5	0	14	0
0	3	14	0	0	3	0	14	0	3	0	14

References

[1] Chandra S., Saxena P.K. *Time minimizing transportation problem with impurities.* Asia-Pacific Jour. Op. Res., 4, 1987

pp. 19-27

[2] Current J. *Multiple objectives in transportation network design and routing.* Eur. Jour. Op. Res. 65, 1993 pp. 1-3

[3] Diaz Joze A. *Solving multiobjective transportation problems.* Econ-math overview, 15, 1979, pp. 62-73

[4] Diaz Joze A. *Finding a Complete Description of All Efficient Solutios to Multiobjective Transportation Problem.* Econ-math overview, 14, 1978, pp. 267-274

[5] Yu. P.L., Zeleny M. *The Set of All Nondominated Solutions in Linear Cases and Multicriteria Simplex Method.* Reprinted from Jour. of Math. Anal. And Appl., V.49, N2, New York, 1975, pp. 430-468

[6] Alexandra Tcacenco, I.M. Stancu-Minasian *Metoda de rezolvare a problemei de transport de tip "bottleneck" cu doua criterii.* Studii si cercetari de calcul economic si cibernetica economica, 4, Academia de studii economice, Bucuresti

[7] Alexandra I. Tcacenco *Modele si metode pentru conducerea optimala a unor procese economice* : Teza de doctorant. Bucuresti, 1994

[8] Aneja Y.P., Nair K.P.K. *Bicriteria transportation problem.* Management Sci., V. 25, N.1 1979 pp.73-79

[9] Garfinkel R.S., Rao N.R. *The Bottleneck Transportation Problem.* N.R.L.Q., 18, 1971, pp. 465-472

[10] Gupta B., Gupta R. *Multi-criteria simplex method for a linear multi-objective transportation problem.* Indian Jour. Pure. Appl. Math. 14(2), 1983, pp. 222-232

[11] Gupta R. *Time-Cost transportation problem.* Econ-math overview,4, 1977, pp. 431-443

[12] Hammer P.Z. *Time Minimizing Transportation Problems.* N.R.L.Q., 16, 1969, p. 345-357

[13] Ringuest J., Rinks D. *Interactive solution for the linear multiobjective transportation problem.* Eur. Jour. Of Op. Res., 32, 1987, p. 96-106

[14] Zeleny M. *Linear Multiobjective Programming.* Lecture Notes on Economics and Mathematical Systems, Op. Res., 1995, Springer-Verlag, 1974

A. Tkachenko, A. Alhazov,
State University of Moldova
60 Mateevici Str.
Chisinau, Moldova MD-2009
e–mail: *tkacenco@mdl.net*

Received September 10, 2001