Control and Cybernetics

vol. **28** (1999) No. 1

Target assignment problem for air raid

by

Marian Chudy

Faculty of Cybernetics, Military University of Technology,
Warsaw, Poland

Abstract: The article deals with two formulations of the target assignment problem. The first one concerns a homogeneous collection of air raid means (different types of aircrafts and missiles). We propose a method for solving a subclass of the problem. The approach consists of two parts. First, an equivalent assignment-type problem is constructed, then a modified branch-and-bound method is used to solve the problem. The other formulation concerns a heterogeneous collection of means. To describe this problem a new algebra is introduced.

Keywords: air raid planning, assignment problem, branch-and-bound method

1. Introduction

The target assignment problem differs from the classical assignment problem, Hung and Rom (1980). It contains constraints associated with the air defense activity and routes for air raid means. Since the results of the air defense activity (a number of destroyed air means) are random the problem is stochastic. We consider only the underlying deterministic problem, Prékopa (1995), which provides only some estimates of features. There are many varieties of the target assignment problem depending on the information available for planning an air raid. Hence, we should not apply the algorithms like the ones by Bertsekas (1981), Glover, Glover, and Klingman (1986), Goldfarb (1985), Hung, and Rom (1980). After Ferland and Hertz, Lavoie (1996), we can call it an assignment-type problem (ATP). It remains an ATP even for the simplest considered case.

We make an attempt to solve the ATP that deals with the case of homogeneous collection of air raid means. An optimal solution of this simplest form of the problems considered estimates the lower bound of the required number of air raid means that can execute the air raid.

In order to obtain the estimate, we transform the master problem into an

- a general assignment problem (GAP),
- a system of linear inequalities.

Special properties of the obtained GAP (zero-one coefficients on the left-hand side of the constraints) suggest that we should not use the general approach to integer programming but rather some modification of the branch-and-bound method.

The relaxed problem that we propose can be solved analytically immediately. We also explain that the transformation we used is efficient even if the GAMS solvers are applied.

Let us introduce the following notation:

D the set of selected points in 3-D space of the considered activity,
i the index of location of the air raid means, $i = \overline{1, I}$, i.e. $i = 1, \ldots, I$,
j the target index, $j \in J$,
n the type index, $n = \overline{1, N}$.

Vector $d^m = (d_1^m, d_2^m, \ldots, d_h^m, \ldots, d_{H(m)}^m)$, where $d_h^m \in D$, $d_{h'}^m \in D$, $d_h^m \neq d_{h'}^m$ if $h \neq h'$, will denote m-th route.

Here, $m \in M_{ij}^n$, where

M_{ij}^n the set of indices of the route from place i to target j for the n-th type of means,
$H(m)$ the m-th route length.

The symbols introduced above will be used in the subsequent sections of the paper.

2. Targets assignment problem for homogeneous air raid means

Now, we assume that each target can be only destroyed by one type of air raid means (ARMS).

If we denote

x_{ij}^{nm} the number of the n-th type of ARMS from location i on the m-th route assigned to destroy the j-th target,

$$y_{jn} = \begin{cases} 1 & \text{if the } j\text{-th target can be destroyed by ARMS of type } n \\ 0 & \text{otherwise} \end{cases}$$

the set of feasible solutions $x = (x_{ij}^{nm})$ can be described by the following constraints:

$$\sum_{j \in J} \sum_{m \in M_{ij}^n} x_{ij}^{nm} \leq a_{in}, \; i = \overline{1, I}, \; n = \overline{1, N} \tag{1}$$

$$\sum^{I} \sum U_{ij}^{nm}(x_{ij}^{nm}) \geq c_{jn} y_{jn}, \; j \in J, \; n = \overline{1, N} \tag{2}$$

$$\sum_{n=1}^{N} y_{jn} = 1 \quad j \in J \tag{3}$$

$$x_{ij}^{nm} \geq 0 \text{ and integer}, \ y_{jn} \in \{0, 1\} \tag{4}$$

where

a_{in} the total number of n-th type of ARMS stationed in the i-th location,

c_{jn} the number of the n-th type of ARMS required to destroy the j-th target (to achieve an assumed level of destruction),

$U_{ij}^{nm}(x_{ij}^{nm})$ a function of decrease of the number of n-th type of ARMS on the route m

satisfying the following condition

$$0 \leq U_{ij}^{nm}(x_{ij}^{nm}) \leq x_{ij}^{nm} \text{ for } x_{ij}^{nm} \geq 0 \tag{5}$$

$U_{ij}^{nm}(x_{ij}^{nm})$ can be interpreted as the number of the n-th type of ARMS that remain after the task on the route m has been completed. Here we assume that the initial assignment number of ARMS is equal X_{ij}^{nm}.

Sometimes it is necessary to take other constraints into account, as well.

In order to construct the optimization problem, we should describe an objective function.

The commonly used objective functions are:

1. $$\sum_{i=1}^{I} \sum_{j \in J} \sum_{n=1}^{N} \sum_{m \in M_{ij}^{n}} d_{ij}^{nm} x_{ij}^{nm} \tag{6}$$

   which is the total distance to be covered by all ARMS from their locations to all objects, here,

   d_{ij}^{nm} the distance from location i to the object j for the n-th type of ARMS on the m-th route,

2. $$\sum_{i=1}^{I} \sum_{j \in J} \sum_{n=1}^{N} \sum_{m \in M_{ij}^{n}} X_{ij}^{nm} \tag{7}$$

   which is the total number of ARMS participating in an air raid at the initial time,

3. $$\max_{i,j,n,m} t_{ij}^{nm} g(x_{ij}^{nm}) \tag{8}$$

   which is the execution time, here,

   $$g(x_{ij}^{nm} = \begin{cases} 1 & \text{if } x_{ij}^{nm} > 0 \\ 0 & \text{otherwise} \end{cases}$$

   and t_{ij}^{nm} is time required by the n-th type of ARMS to cover the distance d_{ij}^{nm}.

In this case we assume that ARMS start at the same time. By combining functions (6)-(8) with constraints (1)-(4) we obtain different target assignment

3. Target assignment problem for multiple types

In this model we assume that a target can be destroyed by different types of ARMS at the same time.

In this case we use b_{jn} instead of c_{jn}:

b_{jn} the level of destruction of object j by the n-th type of ARMS.

Due to the introduction of b_{jn} we define an algebra on the set $Z = \{x \in R : x \geq 0\}$. Here, Z denotes all possible levels of destruction.

We define two operations on elements of the set Z:

1. the sum of elements from Z, denoted by $\oplus$ or $\Sigma\circ$ in general case:

   $x \oplus y = z, \ \forall x, y \in Z$

   which has the properties

   (a) $x \oplus y = y \oplus x, \ x, y \in Z$

   (b) $(x \oplus y) \oplus z = x \oplus (y \oplus z), \ x, y, z \in Z$

   (c) $0 \oplus x = x, \ z \in Z$

   (d) $x \oplus y \leq x + y, \ x, y \in Z.$

   For example

   $$x \oplus y = \sqrt{x^2 + y^2} \tag{9}$$

2. the multiplication defined as follows

   $$\alpha x = \overbrace{x \oplus x \oplus \ldots \oplus x}^{\alpha},$$

   where $x \in Z$ and $\alpha \in N$ the set of natural numbers, $\alpha z = 0$, if $x \in Z$, $\alpha = 0$.

Having defined the above operations we can define the set of feasible plans:

$$\sum_{j \in J} \sum_{m \in M_{ij}^{n}} x_{ij}^{nm} \leq a_{in}, \ i = \overline{1, I}, \ n = \overline{1, N} \tag{10}$$

$$\sum_{i=1}^{I} \circ \sum_{n=1}^{N} \circ \sum_{m \in M_{ij}^{nm}} \circ b_{jn} W_{ij}^{m}(x_{ij}^{nm}) \geq \beta_j, \ j \in J \tag{11}$$

$$x_{ij} \geq 0, \ x_{ij}^{nm} \in C - \ \text{the set of integer numbers} \tag{12}$$

Here

$W_{ij}^{m}(x_{ij}^{nm})$ an integer function of decrease of the number of n-th type of ARMS on the route m satisfying the following conditions

$$0 \leq W_{ij}^{m}(x_{ij}^{nm}) \leq x_{ij}^{nm} \ \text{for} \ x_{ij}^{nm} \geq 0 \tag{13}$$

$$W_{ij}^{m}(x_{ij}^{nm}) \in C \tag{14}$$

where β_j required level of destruction of the target j.

As we said before, the form of function $W_{ij}^{m}(x_{ij}^{nm})$ strongly depends on the model of air defense and the air combat escort. Some examples of this function are

The other proposals can be based on linear or non-linear birth-and-death processes, Barucha-Reid (1960).

The constraints (10)-(12) and chosen objective functions define assignment problems for different ARMS. We do not know a method for solving the assignment problems formulated in this way. However, an adequate method for solving a particular class of these problems can be proposed.

4. Special assignment problem for homogeneous arms

We will consider the assignment problem defined by constrains (1)-(4) and the objective function (7). The latest experience in local wars allows us to assume that

$$U_{ij}^{nm}(x_{ij}^{nm}) = x_{ij}^{nm} \tag{15}$$

Assuming that an air defense is weak but an air combat escort is very strong, the master problem can be written as follows

$$F(x^*) = \min_x \sum_{i=1}^{I} \sum_{j\in J} \sum_{n=1}^{N} \sum_{m\in M_{ij}^n} x_{ij}^{nm} \tag{16}$$

subject to

$$\sum_{j\in J} \sum_{m\in M_{ij}^n} x_{ij}^{nm} \le a_{in},\ i = \overline{1,I},\ n = \overline{1,N} \tag{17}$$

$$\sum_{i=1}^{I} \sum_{m\in M_{ij}^n} x_{ij}^{nm} \ge c_{jn} y_{jn},\ j \in J,\ n = \overline{1,N} \tag{18}$$

$$\sum_{n=1}^{N} y_{jn} = 1,\ j \in J \tag{19}$$

$$x_{ij}^{nm} \ge 0 \text{ and integer},\ y_{jn} \in \{0,1\} \tag{20}$$

In this problem we could cancel index m since the optimal solution does not depend on the routes. We keep it to have the uniform notation.

Let us formulate the following GAP problem (the subproblem 1)

$$\min \sum_{j\in J} \sum_{n=1}^{N} c_{jn} y_{jn} \tag{21}$$

subject to

$$\sum_{n=1}^{N} y_{jn} = 1,\ j \in J \tag{22}$$

$$\sum_{j \in J} c_{jn} y_{jn} \le \sum_{i=1}^{I} a_{in},\ n = \overline{1,N} \tag{23}$$

$$y_{jn} \in \{0,1\} \tag{24}$$

We denote by Y the set of feasible solutions of this problem i.e.

$$Y = \{y = (j_{jn} j \in J,\ n = \overline{1,N} : y \text{ satisfies (22)-(24)}\} \tag{25}$$

We can notice some interesting links between the main problem (16)-(20) and the binary problem (21)-(24).

PROPOSITION 4.1 *If a pair (x, y) satisfies the constraints (17)-(20) then $y \in Y$.*

Proof. (22) follows directly from (19). From (17) we have

$$\sum_{i=1}^{I} \sum_{j \in J} \sum_{m \in M_{ij}^{n}} x_{ij}^{nm} \le \sum_{i=1}^{I} a_{in},\ n = \overline{1,N} \tag{26}$$

and from (18):

$$\sum_{i=1}^{I} \sum_{j \in J} \sum_{m \in M_{ij}^{n}} x_{ij}^{nm} \ge \sum_{i=1}^{I} c_{jn} y_{jn},\ n = \overline{1,N} \tag{27}$$

(26) together with (27) give (23).

PROPOSITION 4.2 *If at least one pair $(x, \bar{y})$ exists, satisfying, (17)-(20), then for each $y \in Y$ the problem (18)-(20) has a feasible solution.*

Proof. Assume that problem (16)-(20) has at least one feasible solution.
For $y \in Y$ we construct the following sets

$$J_n(y) = \{j \in J :\ y_{jn} = 1\},\ n = \overline{1,N} \tag{28}$$

From (23) we have

$$\sum_{j \in J_n(y)} c_{jn} \le \sum_{i=1}^{I} a_{in},\ n = \overline{1,N} \tag{29}$$

Thus, for the fixed route m (it may be different for different n) and fixed n we can get the values that satisfy the constraints

$$\sum x_{ij}^{nm} \le a_{in},\ i = \overline{1,I} \tag{30}$$

$$\sum_{i=1}^{I} x_{ij}^{nm} \geq c_{jn},\ j \in J_n(y) \tag{31}$$

by using, for instance, the method for finding an initial solution in the cost transportation problem.

We should solve N systems of inequalities (30), (31) to find all values of x_{ij}^{nm}.

The system of inequalities corresponding to (30), (31) can be solved in parallel.

PROPOSITION 4.3 *The optimal value of objective function (16) is equal to the optimal value of objective function (21).*

Proof. Let y^* be the optimal solution of the problem (21)-(24). The value

$$G(y^*) = \sum_{j \in J} \sum_{n=1}^{N} c_{jn} y_{jn}^* \tag{32}$$

can be interpreted as the minimum number of ARMS needed to destroy all targets in required level of destruction. This implies that $F(x^*)$ should equal $G(y^*)$.

To obtain $x^* = (x_{jn}^{*nm})$ we should do as in Proposition 2. For each n we define the set $J_n(y^*)$ and reduce constraints (31) to equalities

$$\sum_{i=1}^{I} x_{ij}^{*nm} = c_{jn},\ j \in J_n(y^*) \tag{33}$$

From the above propositions, we can conclude that in order to solve problem (16)-(20) it is necessary to solve the problem (21)-(24). We can use a branch-and-bound method to solve the subproblem 2 which we will also call PLB.

5. A branch-and-bound method

According to the general idea of our method, we must construct a set of PLB subproblems and a suitable set of their relaxation. Successive partitions of Y can be constructed step-by-step, forming a tree. A current partition of Y is created by the nodes of the current tree. These nodes do not have successors.

By Y_k, $k = 0, 1, \ldots$, $(Y_0 = Y)$ we denote not only a subset of Y but also a node of the corresponding tree. The PLB problem at node Y_0 is given by

$$\min_y G_0(y) = \min_y \sum_{j \in J} \sum c_{jn} y_{jn} \tag{34}$$

subject to

$$\sum^{N} y_{jn} = 1,\ j \in J \tag{35}$$

$$\sum_{j\in J} c_{jn}y_{jn} \le \sum_{i=1}^{I} a_{in},\ n=\overline{1,N} \tag{36}$$

$$y_{jn}\in\{0,1\},\ j\in J,\ n=\overline{1,N} \tag{37}$$

We consider the following relaxation of the problem (34)-(37)

$$\min \sum_{j\in J}\sum_{n=1}^{N} c_{jn}y_{jn} \tag{38}$$

subject to

$$\sum_{n=1}^{N} y_{jn}=1,\ j\in J \tag{39}$$

$$y_{jn}\in\{0,1\},\ j\in J,\ n=\overline{1,N} \tag{40}$$

The set

$$T_0=\{y=(y_{jn},\ j\in J,\ n=\overline{1,N} : y \text{ satisfies (39), (40)}\} \tag{41}$$

contains the set Y_0 ($Y_0\subset T_0$).

Let $y^0(0)$ denote the optimal solution of the problem (38)-(40).

This matrix can be obtained as follows:

$$\text{for each } j\in J \quad y^0_{jn}(0)=\begin{cases}1 & \text{for } n=n_j\\ 0 & \text{otherwise}\end{cases} \tag{42}$$

where

$$n_j=\min\{\bar{n}_0 : c_{j\bar{n}_0}=\min_n c_{jn}\} \tag{43}$$

since for each $j\in J$ only one variable y_{jn} can equal 1.

Denote by

D_k the path (in the tree) from the node Y_0 to the node Y_k,

W_k the set of pairs of indices (j,n), whose variables y_{jn} have fixed values on the path D_k,

$$\begin{aligned} W_k^+ &= \{(j,n)\in W_k : y_{jn}=1\}\\ W_k^- &= \{(j,n)\in W_k : y_{jn}=0\}\\ F_k &= (J\times N)\setminus W_k \end{aligned}$$

The PLB problem associated with the node Y_k is as follows

$$\min G_k(y) = \min \sum c_{jn}y_{jn} + \sum c_{jn} \tag{44}$$

subject to

$$\sum_{n \in N_{jk}} y_{jn} = 1, \; j \in J \setminus J_k^+ \tag{45}$$

$$\sum_{j \in J \setminus J_k^+} c_{jn} y_{jn} \le \sum_{i=1}^{I} a_{in} - \sum_{j \in J_k^+} c_{jn} = r_k, \; n = \overline{1, N} \tag{46}$$

$$y_{jn} \in \{0, 1\}, \; (j, n) \in F_k \tag{47}$$

Here,

$$\begin{aligned} N_{jk} &= \{n : (j, n) \in F_k\} \\ J_k^+ &= \{j \in J : (j, n) \in W_k^+\} \\ J_k^- &= \{j \in J : (j, n) \in W_k^-\} \end{aligned}$$

The set of feasible solution of the problem (44)-(47), denoted by we Y_k, can be described as follows:

$$Y_k = \left\{ y : y \text{ satisfies (45)-(47)}, \; y_{jn} = \left\{ \begin{array}{ll} 1 & \text{for } (j, n) \in W_k^+ \\ 0 & \text{for } (j, n) \in W_k^- \end{array} \right\} \right\}$$

We denote by $y^*(k)$ the optimal solution of the problem (44)-(47).
We propose the relaxation of the problem (44)-(47) in the form of

$$\min_y G_k(y) = \min_y \sum_{(j,n) \in F_k} c_{jn} y_{jn} + \sum_{(j,n) \in W_k^+} c_{jn} \tag{48}$$

subject to

$$\sum_{n \in N_{jk}} y_{jn} = 1, \; j \in J \setminus J_k^+ \tag{49}$$

$$y_{jn} \in \{0, 1\}, \; j \in J \setminus J_k^+, \; n = \overline{1, N} \tag{50}$$

The set $T_k (Y_k \subset T_k)$ is defined by the constraints (49), (50).
The optimal solution of this problem can be easily obtained:

$$\text{for each } j \in J \setminus J_k \; y_{jn}^0(k) = \left\{ \begin{array}{ll} 1 & \text{for } n = n_j \\ 0 & \text{otherwise} \end{array} \right. \tag{51}$$

where n_j is given by (43) and $J_k = J_k^+ \cup J_k^-$, where

$$\bar{n}_j = \min\{\bar{n} : c_{j\bar{n}} = \min_{n \in \bar{N} \setminus N_{jk}^-} c_{jn}\}, \tag{53}$$

$$\bar{N} = \{1, 2, \ldots, N\},\ N_{jk}^- = \{n : (j, n) \in W_k^-\}.$$

Now, we establish the elimination rules of nodes. As usual, the node is eliminated if it does lead to the improvement of a solution. The node that has not been eliminated is called the active node.

Let $\bar{G}_k$ and of $\underline{G}_k$ be the upper and lower bounds of $G_k(y^*)$ i. e. the optimal value of the objective function of the problem PBL at the node Y_k.

A node Y_k is eliminated if one of the following conditions holds:
a) $\overline{G}_k = \underline{G}_k$,
b) $\underline{G}_k \geq \overline{G}_0$,
c) $Y_k = \emptyset$.
These conditions can be treated as the elimination rules of nodes.

It can be observed that the optimal solution $y^0(k)$ gives us the lower bound of $G_k(y^*)$.

If $y^0(k) \in Y_k$, then $G_k(y^0(k)) = \overline{G}_k$ is also the upper bound of $G_k(y^*)$. This allows us to eliminate the node Y_k.

If $y^0(k) \notin Y_k$ then we should select a node for partitioning and a variable which helps us to obtain the partition of the selected node (branching variable).

We establish it as a principle that we will choose this node to partition, which was just obtained by assigning some variable y_{jn} the value 1.

We will construct a procedure to obtain the index of this branching variable as follows.

Let Y_k be the selected node. Then the index (j_k, n_k) of branching variable $y_{j_k n_k}$ to partition Y_k into subsets is given by the formula:

$$c_{j_k n_k} = \min\left\{\min_{j \in J \setminus j_k} c_{jn},\ \min_{(j,n) \in \overline{W}_k} c_{jn}\right\} \tag{54}$$

where

$$\overline{W}_k = \{(j, n) : (J_k^- \times \overline{N}) \setminus (W_k^- \cup (j_k^- \cap J_k^+) \times \overline{N})\}$$

The expression (54) comes from the form of the objective function of the PLB problem and the form of constraint (35).

Y_k is partitioned into two subsets

$$Y_k \cap \{y : y_{j_k n_k} = 1\} \text{ and } Y_k \cap \{y : y_{j_k n_k} = 0\} \tag{55}$$

which become the successors of Y_k.

Now, we can propose the procedure for solving the problem (34)-(37).

1. Set $F_0 = J \times N$, $\overline{G}_0 = \infty$, $\underline{G}_0 = -\infty$, $k = 0$.

2. If there are no active nodes, go to 5.
   Otherwise, we consider the previously obtained node (if $k = 0$, this will be Y_0, if $k \geq 1$, this will be the node obtained from Y_k by assigning 1 to $y_{j_k n_k}$).
   Define the set $A_k = \{i : r_i \leq 0\}$. If $A_k \neq \emptyset$ or $Y_k = \emptyset$, eliminate node Y_k and go to 4.
   Otherwise, determine the optimal solution $y^0(k)$ to relaxation (48)-(50) by using (51)-(52) and the lower bound $\underline{G}_k = G_k(y^0(k))$.
3. Verify $y^0(k)$.
   If $y^0(k) \in Y_k$, determine $\overline{G}_k = \underline{G}_k = G_k(y^0(k))$ and eliminate node Y_k. Set $\overline{G}_0 = \min\{\overline{G}_0, \overline{G}_k\}$. Remove any mode for which $\underline{G}_k \geq \overline{G}_0$. Go to 4.
   If $y^0(k) \notin Y_k$, determine index (j_k, n_k) by using (54). Construct the successors of Y_k according to (55) and go to 2.
4. If there is no active node, go to 5. Otherwise, set the index of the lately calculated node among the currently active nodes at k.
   Determine $A_k = \{i : r_i \leq 0\}$. If $A_k \neq \emptyset$ or $Y_k = \emptyset$, remove Y_k and go to 4. If $A_k = \emptyset$, determine the optimal solution $y^0(k)$ of the relaxation problem (48)-(50) according to (51), (52) and $\underline{G}_k = G_k(y^0(k))$. Go to 3.
5. If $\overline{G} = \infty$, then the problem (34)-(37) does not have any feasible solution ($Y_0 = \emptyset$) and STOP. If $\overline{G} < \infty$, then the matrix y where $G(y) = \overline{G}_0$ is the optimal solution of the problem (34)-(37) i.e. $y = y^*$. STOP.

6. Numerical results and comments

The experiments were done on a PENTIUM - Pro200. We used the GAMS (CPLEX) system to solve the master problem (16)-(20) and the subproblem 1 (21), (24). The coefficients and parameters of these problems were generated randomly.

The parameters I, J, N were close to reality.

We implemented the proposed method for solving subproblem 1 in MODULA. All the results are summarized in Table 1. In fact, we compare here not only the methods but the computer programs, as well.

We assume that the time devoted to solving subproblem 2 is negligible, so it is excluded from the comparison.

We conclude, from Table 1, that:

- the transformation of master problem into equivalent pair of two subproblems is advantageous,
- in most cases the proposed method runs faster than the professional solver (the sample contained one hundred problems).

			CPU time (sec.)		
I	J	N	Master problem GAMS (CPLEX)	Subproblem 1 GAMS (CPLEX)	Subproblem 1 Proposed methods MODULA
3	20	9	0.11	0.05	0.05
4	20	9	0.16	0.06	0.00
7	20	10	0.17	0.06	0.00
3	16	8	0.06	0.06	0.00
2	17	5	0.06	0.06	0.05
5	17	8	0.11	0.06	0.00
7	20	9	0.22	0.06	0.00
4	17	10	0.11	0.05	0.05
4	16	6	0.11	0.05	0.06
2	17	5	0.06	0.05	0.06
2	15	8	0.11	0.05	0.00
2	20	6	0.11	0.05	0.05
3	20	10	0.11	0.05	0.00
2	15	6	0.05	0.05	0.05
2	15	9	0.11	0.06	0.06
3	17	7	0.11	0.06	0.00
4	19	10	0.16	0.05	0.00
3	20	10	0.11	0.06	0.05
3	20	8	0.11	0.05	0.11
4	18	8	0.11	0.06	0.00

Table 1.

References

BALAS, E., SALTZMAN, M.J. (1991) An algorithm for the three-index assignment problem. *Operations Research*, **39**, 150-161.

BARUCHA-REID, A.T. (1960) *Elements of the Theory of Stochastic Processes and Their Applications*. Mc Graw-Hill Book Company, New York.

BERTSEKAS, D.P. (1981) A new algorithm for the assignment problem. *Mathematical Programming*, **21**, 152-171.

FERLAND, J.A., HERTZ, A., LAVOIE, A. (1996) An object-oriented methodology for solving assignment - type problems with neighborhood search techniques. *Operations Research*, **44**, 347-359.

GLOVER, F., GLOVER, R. and KLINGMAN, D. (1986) Threshold assignment algorithm. *Mathematical Programming Study*, **26**, 12-37.

GOLDFARB, D. (1985) Efficient dual simplex algorithms for the assignment problem. *Mathematical Programming*, **33**, 187-203.

HUBER, R.K. (1984) *Systems Analysis of Modelling in Defence. Trends and Issues*. New York and London.

HUNG, H.W. and ROM, W.O. (1980) Solving the assignment problem by relaxation. *Operations Research*, **35**, 969-982.

KENNINGTON, J., WANG, Z. (1992) A shortest augmenting path algorithm for the semi-assignment problem. *Operations Research*, **40**, 178-187.

MURTHY, I. (1993) Solving the multiperiod assignment problem with start-up costs using dual ascent. *Naval Research Logistics*, **40**, 325-344.

PRÉKOPA, A. (1995) *Stochastic Programming*. Kluwer Academic Publishers, Dordrecht, Boston, London.