

On the reformulation of some classes of PNS-problems as set covering problems

J. Fülöp *

B. Imreh †

F. Friedler ‡

Abstract

Process network synthesis (PNS) has enormous practical impact; however, its solution is difficult in general. This experience has been recently reasoned by Blázsik and Imreh who pointed out that PNS-problems are NP-hard. They proved that a simple subclass of PNS-problems is equivalent to the class of set covering problems. In the present paper, it is shown that more general classes of PNS-problems can also be reformulated as set covering problems. This enables the sophisticated techniques developed for solving set covering problems also to be applied for solving some PNS-problems.

1 Introduction

The importance of process network synthesis (PNS) and the background of the combinatorial model studied here can be found in [5], [6], [7], [8], [9], and in the work [2] of this journal. Therefore, we shall confine ourselves only to the recall of the definitions. The combinatorial approach makes possible to show that the search of an optimal solution is difficult in general. This experience has been recently reasoned by Blázsik and Imreh [2] who pointed out that PNS-problems with weights are NP-hard. They proved that a simple subclass of PNS-problems with weights, to be discussed in Section 4, is equivalent to the class of set covering problems. Also in [2], it was raised as an open problem if there exist equivalent known optimization problems for more general classes of PNS.

In this paper, it is shown that the optimal solutions for a larger subclass of PNS-problems than the subclass presented in [2] as well as the optimal solutions of PNS-problems with nonnegative weights can be obtained by solving suitably constructed set covering problems. This enables the sophisticated techniques developed for solving set covering problems (see, e.g., [1, 4, 10] and the references therein) also to be applied for solving these special classes of PNS-problems with weights. To

*Laboratory of Operations Research and Decision Systems, Computer and Automation Institute, Hungarian Academy of Sciences, H-1518 Budapest, P.O.Box 63, Hungary

†Department of Informatics, József Attila University, Árpád tér 2, H-6720 Szeged, Hungary

‡Department of Computer Science, University of Veszprém, Egyetem u. 10, H-8200 Veszprém, Hungary

present our results, first we discuss the conjunctive normal form (CNF) proposed in [3] for describing the solution-structures of PNS-problems in Section 3. Some special classes of PNS-problems with weights, and the connection between the optimal solutions of these PNS and CNF-problems with weights are detailed in Section 4. The reformulation of a CNF with weights as a set covering problem is presented in Section 5.

2 Notions and notations

In the combinatorial approach, the structure of a process can be described by the process-graph (see [7] and [8]) defined as follows.

Let M be a finite nonempty set, the set of the materials. Furthermore, let $\emptyset \neq O \subseteq \wp'(M) \times \wp'(M)$ with $M \cap O = \emptyset$ where $\wp'(M)$ denotes the set of all nonempty subsets of M . The elements of O are called *operating units* and for an operating unit $(\alpha, \beta) \in O$, α and β are called the *input-set* and *output-set* of the operating unit, respectively. Pair (M, O) is defined to be a *process graph*. The set of vertices of this directed graph is $M \cup O$, and the set of arcs is $A = A_1 \cup A_2$ where $A_1 = \{(X, Y) : Y = (\alpha, \beta) \in O \text{ and } X \in \alpha\}$ and $A_2 = \{(Y, X) : Y = (\alpha, \beta) \in O \text{ and } X \in \beta\}$. If there exist vertices X_1, X_2, \dots, X_n , such that $(X_1, X_2), (X_2, X_3), \dots, (X_{n-1}, X_n)$ are arcs of process graph (M, O) , then the path determined by these arcs is denoted by $[X_1, X_n]$.

Let process graphs (m, o) and (M, O) be given. (m, o) is defined to be a *subgraph* of (M, O) , if $m \subseteq M$ and $o \subseteq O$.

Now, we can define the structural model of PNS for studying the problem in structural point of view. For this reason, let M^* be an arbitrarily fixed infinite set, the set of the available materials. By *structural model* of PNS, we mean a triplet (P, R, O) where P, R, O are finite sets, $\emptyset \neq P \subseteq M^*$ is the set of the *desired products*, $R \subseteq M^*$ is the set of the *raw materials*, and $O \subseteq \wp'(M^*) \times \wp'(M^*)$ is the set of the available operating units. It is assumed that $P \cap R = \emptyset$ and $M^* \cap O = \emptyset$.

Then, process graph (M, O) , where $M = \bigcup\{\alpha \cup \beta : (\alpha, \beta) \in O\}$, presents the interconnections among the operating units of O . Furthermore, every feasible process, producing the given set P of products from the given set R of raw materials using operating units from O , corresponds to a subgraph of (M, O) . Examining the corresponding subgraphs of (M, O) , therefore, we can determine an optimal process in principle. If we do not consider further constraints such as material balance, then the subgraphs of (M, O) which can be assigned to a feasible process have common combinatorial properties. They are studied in [7] and their description is given by the following definition.

Subgraph (m, o) of (M, O) is called a *solution-structure* of (P, R, O) if the following properties are satisfied:

$$(S1) \quad P \subseteq m,$$

$$(S2) \quad \forall X \in m, X \in R \Leftrightarrow \text{no } (Y, X) \text{ arc in the process graph } (m, o),$$

$$(S3) \forall Y_0 \in o, \exists \text{ path } [Y_0, Y_n] \text{ with } Y_n \in P,$$

$$(S4) \forall X \in m, \exists (\alpha, \beta) \in o \text{ such that } X \in \alpha \cup \beta.$$

Let us denote the set of solution-structures of (P, R, O) by $S(P, R, O)$. In the sequel, we shall assume that $S(P, R, O) \neq \emptyset$. This can be checked in polynomial time by using the algorithm presented in [9] for generating the maximal structure of (P, R, O) .

Let the set of the operating units be given by $O = \{(\alpha_1, \beta_1), \dots, (\alpha_l, \beta_l)\}$, and let $I = \{1, \dots, l\}$. Then, for any subgraph (m, o) of (M, O) , an l -vector of logical values $u_i, i \in I$, can be associated with such that u_i is true if and only if $(\alpha_i, \beta_i) \in o$. It is easy to see that this is a one-to-one mapping between the subgraphs of (M, O) fulfilling (S4) and the l -vectors of logical values. For logical l -vector u , subgraph (m, o) associated with u is determined by $m = \bigcup_{i \in T(u)} \alpha_i \cup \beta_i$ and $o = \{(\alpha_i, \beta_i) : i \in T(u)\}$ where $T(u) = \{i \in I : u_i \text{ is true}\}$.

3 CNF related to PNS

In [3], a logical expression given in CNF (A1)-(A4) below was used to describe some structures of (M, O) .

$$(A1) \bigwedge_{X \in P} \bigvee_{\substack{i \in I \\ X \in \beta_i}} u_i,$$

$$(A2) \bigwedge_{\substack{i \in I \\ R \cap \beta_i \neq \emptyset}} \neg u_i,$$

$$(A3) \bigwedge_{\substack{i \in I \\ X \in \alpha_i \setminus R}} (\neg u_i \vee \bigvee_{\substack{h \in I \\ X \in \beta_h}} u_h),$$

$$(A4) \bigwedge_{\substack{i \in I \\ P \cap \beta_i = \emptyset}} (\neg u_i \vee \bigvee_{\substack{h \in I \\ \beta_i \cap \alpha_h \neq \emptyset}} u_h).$$

In this section, the relationship between (S1)-(S4) and (A1)-(A4) will be discussed.

Proposition 1. *For any solution-structure (m, o) , the logical vector, u , associated with (m, o) fulfills (A1)-(A4).*

Proof. Let u be the logical vector associated with solution-structure (m, o) . From (S1)-(S2) and $P \cap R = \emptyset$, we obtain that any $X \in P$ is in the output-set of an operating unit of (m, o) . This gives (A1). (A2) follows directly from (S2).

Concerning (A3), we have to show that if u_i is true for some $i \in I$ and $X \in \alpha_i \setminus R$, then there exists an $h \in I$ such that u_h is true and $X \in \beta_h$, i.e., X is in the output-set of an operating unit of (m, o) . This follows however immediately from (S2).

To prove (A4), it is sufficient to consider the case when u_i is true and $P \cap \beta_i = \emptyset$. From (S3) we get that there exists a path in (m, o) from (α_i, β_i) to an element of P .

Since $P \cap \beta_i = \emptyset$, the vertex second to (α_i, β_i) in the path is an $(\alpha_{h'}, \beta_{h'})$, $h' \in I$, such that $\beta_i \cap \alpha_{h'} \neq \emptyset$. This implies (A4) immediately. \square

Proposition 2. *For any logical vector u fulfilling (A1)-(A4), the subgraph, (m, o) , associated with u satisfies (S1), (S2), and (S4).*

Proof. (A1) states that for every $X \in P$, there exists an $i \in I$ such that u_i is true and $X \in \beta_i$. This gives $X \in m$, and thus, (S1) holds.

To prove (S2), consider an $X \in m \cap R$. From (A2) we get that u_i is false for every $i \in I$ with $X \in \beta_i$. The way of construction of (m, o) from u implies that there exists no (Y, X) arc in (m, o) .

Conversely, consider an $X \in m \setminus R$. We show that there exists an arc (Y, X) in (m, o) , i.e., X is in the output set of an operating unit associated with a true component of u . Since $X \in m$, there exists an $i \in I$ such that u_i is true and $X \in \alpha_i \cup \beta_i$. If $X \in \beta_i$, we are done. Otherwise, $X \in \alpha_i \setminus R$ and (A3) implies that there exists an $h \in I$ such that u_h is true and $X \in \beta_h$.

Finally, (S4) follows from the way of construction of (m, o) from u . \square

It is worth noting that (A1)-(A4) does not imply (S3). Namely, considering a general process graph, (M, O) , there may exist an operating unit $Y_0 \in o$ in subgraph (m, o) constructed from u fulfilling (A1)-(A4) such that there is no path from Y_0 to any element of P . However, for special PNS-problems, (S3) is also implied by (A1)-(A4), thus, (S1)-(S4) and (A1)-(A4) are equivalent.

Proposition 3. *If process graph (M, O) does not contain circuit, then (S1)-(S4) and (A1)-(A4) are equivalent.*

Proof. By Propositions 1 and 2, it is sufficient to show that (A4) implies (S3) in this case. Consider a $Y_{i_0} = (\alpha_{i_0}, \beta_{i_0}) \in o$. If $P \cap \beta_{i_0} \neq \emptyset$, we can construct a path from Y_{i_0} to an element of $P \cap \beta_{i_0}$. Otherwise, by (A4), there exists another operating unit $Y_{i_1} = (\alpha_{i_1}, \beta_{i_1})$ such that $Y_{i_1} \in o$ and $\beta_{i_0} \cap \alpha_{i_1} \neq \emptyset$. We have now path $[Y_{i_0}, Y_{i_1}]$ in (m, o) , and we can repeat the investigation above now for Y_{i_1} instead of Y_{i_0} .

In a general step, we have operating unit $Y_{i_k} = (\alpha_{i_k}, \beta_{i_k})$ and path $[Y_{i_0}, Y_{i_k}]$ in (m, o) . If $P \cap \beta_{i_k} \neq \emptyset$, we are ready. Otherwise, we can extend the path from Y_{i_k} . Since (M, O) contains no circuit, every vertex of the path is different. However, (M, O) is finite, thus, after constructing a finite number of arcs, the path has to terminate in an element of P . \square

Assume that in process graph (M, O) of a PNS- problem, with a suitable positive integer k , we have $M = M_1 \cup \dots \cup M_{k+1}$ where the sets, M_1, \dots, M_{k+1} , are pairwise disjoint nonempty sets. Furthermore, let $O = O_1 \cup \dots \cup O_k$ with $O_i \subseteq \wp'(M_1 \cup \dots \cup M_i) \times \wp'(M_{i+1})$, $i = 1, \dots, k$. Let us call such a PNS-problem a PNS_k -problem. Then, it is easy to see that for any PNS_k -problem, there exists no circuit in its process graph, and consequently, we have the following corollary.

Corollary 1. *(S1)-(S4) and (A1)-(A4) are equivalent for PNS_k -problems.*

4 PNS-problems with weights

Let us consider PNS-problems in which each operating unit has a weight. We are to find a feasible process with the minimal weight where by weight of a process we mean the sum of the weights of the operating units belonging to the process under consideration. Every feasible process in such a class of PNS-problems is determined uniquely from the corresponding solution-structure and vice versa. Therefore, the above problem can be formalized in the following way.

Let a structural model of PNS-problem (P, R, O) be given. Moreover, let w be a real-valued function defined on O , the weight function. The basic model is then

$$\min \left\{ \sum_{U \in o} w(U) : (m, o) \in S(P, R, O) \right\}. \tag{1}$$

We refer (1) as a PNS_w -problem; we denote the class of such problems by PNS_w . PNS_k -problems with weights are referred as PNS_{wk} -problems, their subclass is denoted by PNS_{wk} . These latter problems were introduced, and the connection between PNS_{w1} -problems and set covering problems was also discussed in [2].

The feasible set of the optimization problem (1) is the set of the subgraphs (m, o) fulfilling (S1)-(S4). According to the discussion of the relation between (S1)-(S4) and (A1)-(A4), another optimization problem based on the CNF (A1)-(A4) can also be considered:

$$\min \left\{ \sum_{i \in T(u)} w_i : u \text{ fulfills (A1)-(A4)} \right\} \tag{2}$$

where $w_i = w((\alpha_i, \beta_i)), i \in I$. We refer (2) as a CNF_w -problem associated with PNS_w -problem (1), and denote the class of such problems by CNF_w .

By Propositions 1 and 2, CNF_w can be considered as a relaxation of PNS_w . This gives rise to the following statements.

Proposition 4. *Both (1) and (2) have finite optimal value. The optimal value of (1) is greater than or equal to that of (2). Furthermore, if (S3) holds in the subgraph (m^*, o^*) associated with an optimal solution of (2), then (m^*, o^*) is an optimal solution of (1).*

In the case of PNS_k -problems, the equivalence between (S1)-(S4) and (A1)-(A4) implies a similar equivalence between the relating problems of PNS_{wk} and CNF_{wk} .

Corollary 2. *Consider problems (1) and (2) generated by a PNS_k -problem. Then, the subgraph, (m^*, o^*) , associated with an optimal solution of (2) is optimal to (1), and conversely, the l -vector of logical values associated with an optimal solution of (1) is an optimal solution to (2).*

The following statements relate to special subclasses of PNS_w .

Proposition 5. *If the weights, w_i , $i \in I$, are positive, then subgraph (m^*, o^*) associated with an optimal solution of (2) is optimal to (1), and conversely, the l -vector of logical values associated with an optimal solution of (1) is an optimal solution to (2).*

Proof. Let u^* be an optimal solution of (2), and let (m^*, o^*) be the subgraph associated with u^* . By Proposition 4, it is sufficient to show that (S3) holds for (m^*, o^*) . Let

$$\bar{o} = \{U \in o^* : \exists \text{ path in } (m^*, o^*) \text{ from } U \text{ to a } Y \in P\}, \quad (3)$$

$$\bar{m} = \sum_{\substack{(\alpha, \beta) \in o^* \\ i \in I}} \alpha_i \cup \beta_i. \quad (4)$$

Clearly, (\bar{m}, \bar{o}) is a subgraph of (m^*, o^*) . If $\bar{o} = o^*$, we are done. Otherwise, we shall show below that (\bar{m}, \bar{o}) is a solution-structure of (P, R, O) . Then, the logical vector, \bar{u} , associated with (\bar{m}, \bar{o}) is feasible to (2). However, since $w(U) > 0$ for every $U \in o^* \setminus \bar{o}$, the objective function value of \bar{u} is less than that of u^* , and this contradicts the optimality of u^* in (2). Consequently, $o^* = \bar{o}$ must hold.

We show now that (S1)-(S4) holds for (\bar{m}, \bar{o}) . By Proposition 2, (m^*, o^*) fulfills (S1), (S2), and (S4). Thus, from (3)-(4), we get immediately that (S1), (S3), and (S4) hold for (\bar{m}, \bar{o}) .

To prove (S2) for (\bar{m}, \bar{o}) , consider an $X \in \bar{m} \cap R$. Since there exists no (Y, X) arc in (m^*, o^*) , and (\bar{m}, \bar{o}) is a subgraph of (m^*, o^*) , there exists (Y, X) arc neither in (\bar{m}, \bar{o}) . Conversely, consider an $X \in \bar{m} \setminus R$. In (m^*, o^*) , there exists a (Y, X) arc. In addition, since $X \in \alpha \cup \beta$ for an $(\alpha, \beta) \in \bar{o}$, there exists a path in (\bar{m}, \bar{o}) from (α, β) to an element of P , thus, also from Y to the same element of P . Therefore, $Y \in o$ and (Y, X) is an arc in (\bar{m}, \bar{o}) .

The second part of the statement can be easily proved by using Proposition 1 and the first part of the statement. \square

Proposition 6. *If the weights, w_i , $i \in I$, are nonnegative, then subgraph (\bar{m}, \bar{o}) defined by (3)-(4) for (m^*, o^*) associated with an optimal solution of (2) is optimal to (1).*

Proof. According to the proof of Proposition 5, (\bar{m}, \bar{o}) is feasible to (1). It may happen now that $o^* \setminus \bar{o} \neq \emptyset$ but from the nonnegativity of the weights and using the same reasoning as in the proof of Proposition 5, we obtain that $w(U) = 0$ for every $U \in o^* \setminus \bar{o}$. The objective function values of (\bar{m}, \bar{o}) and (m^*, o^*) coincide in (1). Therefore, (\bar{m}, \bar{o}) is optimal to (1). \square

The set \bar{o} defined in (3) can easily be generated by using the classical labeling technique of graph theory [13]. A similar technique is used also in [9] for generating the maximal structure of a process graph. It can be shown that (\bar{m}, \bar{o}) is the union of all solution-structure subgraphs of (m^*, o^*) . See [9] for more details.

5 Reformulation of a CNF with weights as a set covering problem

By the results presented in the previous section, the optimal solution of some important classes of PNS_w -problems, such as problems with nonnegative weights and PNS_{wk} -problems, can be obtained by solving the appropriate CNF_w -problems of form (2). However, it has not been discussed yet how to solve (2). In this section, we show that (2) can be transcribed into the form of an equivalent set covering problem. This can also be considered as an extension of the results presented in [2] for PNS_{w1} -problems.

For every $u_i, i \in I$, we introduce two 0-1 variables, z_i^+ and z_i^- , such that $z_i^+ = 1$ if and only if u_i is true, and $z_i^- = 1 - z_i^+$. Then, at the expense of doubling the number of variables and introducing some appropriate new constraints, (2) can be written into the equivalent form

$$\min \sum_{i \in I} w_i z_i^+, \tag{5}$$

$$\sum_{\substack{i \in I \\ X \in \beta_i}} z_i^+ \geq 1 \quad \text{for all } X \in P, \tag{6}$$

$$z_i^- = 1 \quad \text{for all } i \in I, R \cap \beta_i \neq \emptyset, \tag{7}$$

$$z_i^- + \sum_{\substack{h \in I \\ X \in \beta_h}} z_h^+ \geq 1 \quad \text{for all } i \in I, X \in \alpha_i \setminus R, \tag{8}$$

$$z_i^- + \sum_{\substack{h \in I \\ \beta_i \cap \alpha_h \neq \emptyset}} z_h^+ \geq 1 \quad \text{for all } i \in I, P \cap \beta_i = \emptyset, \tag{9}$$

$$z_i^+ + z_i^- = 1 \quad \text{for all } i \in I, \tag{10}$$

$$z_i^+, z_i^- \in \{0, 1\} \quad \text{for all } i \in I. \tag{11}$$

In (5)-(11), (5) and (6)-(9) are the direct transcription of the objective function in (2) and the constraints (A1)-(A4), respectively. Constraints (10)-(11) describe the relation among u_i, z_i^+ and z_i^- . Since we have assumed that $S(P, R, O) \neq \emptyset$, problems (1), (2), hence (5)-(11), too, have feasible solution and finite optimal value.

Problem (5)-(11) is a set covering/partitioning problem for which efficient solution methods have been developed, see [4] and the references therein. Constraint (7) means to fix $z_i^- = 1$ and $z_i^+ = 0$ for all $i \in I, R \cap \beta_i \neq \emptyset$, and these can entail the possible fixation of further variables and the deletion of some constraints [1, 4, 10].

By using the well-known trick of converting set partitioning constraints into set covering ones (cf. e.g. [10]), we obtain the following statement.

Proposition 7. *Choose any $L > \sum_{i \in I} w_i$, and consider the set covering problem*

$$\min \sum_{i \in I} [(w_i + L)z_i^+ + Lz_i^-], \tag{12}$$

$$\sum_{\substack{i \in I \\ X \in \beta_i}} z_i^+ \geq 1 \quad \text{for all } X \in P, \quad (13)$$

$$z_i^- \geq 1 \quad \text{for all } i \in I, R \cap \beta_i \neq \emptyset, \quad (14)$$

$$z_i^- + \sum_{\substack{h \in I \\ X \in \beta_h}} z_h^+ \geq 1 \quad \text{for all } i \in I, X \in \alpha_i \setminus R, \quad (15)$$

$$z_i^- + \sum_{\substack{h \in I \\ \beta_i \cap \alpha_h \neq \emptyset}} z_h^+ \geq 1 \quad \text{for all } i \in I, P \cap \beta_i = \emptyset, \quad (16)$$

$$z_i^+ + z_i^- \geq 1 \quad \text{for all } i \in I, \quad (17)$$

$$z_i^+, z_i^- \in \{0, 1\} \quad \text{for all } i \in I. \quad (18)$$

Then, problems (5)-(11) and (12)-(18) have the same set of optimal solutions.

Proof. It is easy to see that any feasible solution of (5)-(11) is feasible to (12)-(18) as well, and the difference of the two objective function values is the constant, lL . As a consequence, since the optimal value of (5)-(11) is less than L , the optimal value of (12)-(18) is less than $(l+1)L$.

Conversely, consider a feasible solution of (12)-(18) and assume that it is not feasible to (5)-(11). Then, its objective function value in (12) is greater than or equal to $(l+1)L$. Thus, any optimal solution of (12)-(18) is feasible to (5)-(11), and this implies the statement. \square

In set covering problem (12)-(18), as well, constraint (14) entails the possible reduction of the problem size. For further size reduction techniques and for recent sophisticated methods for solving set covering problems, see [1, 4] and the references therein.

References

- [1] E. Balas and M. C. Carrera, A Dynamic Subgradient- Based Branch-and-Bound Procedure for Set Covering, *Operations Research*, **44**(6), 1996, 875-890.
- [2] Z. Blázsik and B. Imreh, A Note on Connection between PNS and Set Covering Problems, *Acta Cybernetica*, **12**, 1996, 309-312.
- [3] M. H. Brendel, F. Friedler and L.T. Fan, Combinatorial Foundation for Logical Formulation in Total Flowsheet Synthesis, *Computers chem. Engng.* (submitted).
- [4] M. L. Fisher and P. Kedia, Optimal Solution of Set Covering/Partitioning Problems Using Dual Heuristics, *Management Science*, **36**, 1990, 674-688.
- [5] C. A. Floudas and I. E. Grossmann, Algorithmic Approaches to Process Synthesis: Logic and Global Optimization, International Symposium on Foundations of Computer Aided Process Design, Snowmass Village, CO, U.S.A., July 10-15, 1994 (in press).

- [6] F. Friedler, L. T. Fan and B. Imreh, Process Network Synthesis: Problem Definition, *Networks* (to appear).
- [7] F. Friedler, K. Tarjan, Y. W. Huang and L. T. Fan, Graph-Theoretic Approach to Process Synthesis: Axioms and Theorems, *Chem. Eng. Sci.*, **47**(8), 1992, 1973-1988.
- [8] F. Friedler, K. Tarjan, Y. W. Huang and L. T. Fan, Combinatorial Structure of Process Network Synthesis, Sixth SIAM Conference on Discrete Mathematics, Vancouver, Canada, 1992.
- [9] F. Friedler, K. Tarjan, Y. W. Huang and L. T. Fan, Graph-Theoretic Approach to Process Synthesis: Polynomial Algorithm for Maximal Structure Generation, *Computers chem. Engng.*, **17**(9), 1993, 929-942.
- [10] R. S. Garfinkel and G. L. Nemhauser, *Integer Programming*, Wiley, New York, 1972.
- [11] I. E. Grossmann, V. T. Voudouris and O. Ghattas, Mixed-Integer Linear Programming Reformulations for Some Nonlinear Discrete Design Optimization Problems, In: *Recent Advances in Global Optimization* (Eds: C. A. Floudas and P. M. Pardalos) Princeton University Press, New Jersey, 1992.
- [12] B. Imreh, F. Friedler and L. T. Fan, An Algorithm for Improving the Bounding Procedure in Solving Process Network Synthesis by a Branch-and-Bound Method, in: *Developments in Global Optimization* (Eds: I. M. Bonze, T. Csendes, R. Horst, P. M. Pardalos), Kluwer Academic Publishers, 1997, 315-348.
- [13] E. Lawler, *Combinatorial Optimization*, Holt, Rinehart and Winston, New York, 1976.

Received August, 1997