Acta Cybernetica 14 (1999) 251-261.

On a Merging Reduction of the Process Network Synthesis Problem*

Cs. Holló †, Z. Blázsik ‡ Cs. Imreh †, Z. Kovács †

Abstract

Since the combinatorial version of the process network synthesis (PNS) problem is NP-complete, it is important to establish such methods which render possible the reduction of the size of model. In this work, a new method called merging reduction is introduced which is based on the merging of operating units. The mergeable operating units are determined by an equivalence relation on the set of the operating units, and all of the operating units included in an equivalence class are merged into one new operating unit. This reduction has the following property: an optimal solution of the original problem can be derived from an optimal solution of the reduced problem and conversely. Presentation of this reduction technique is equipped with an empirical analysis on randomly generated problems which shows the measure of the size decrease.

1 Preliminaries

The foundations of PNS and the background of the combinatorial model studied here can be found in [3], [4], [5], and [9]. Therefore, we shall confine ourselves only to the recall of the definitions here. The merging reduction is presented in Section 2, while Section 3 contains the results of our empirical analysis.

In the combinatorial approach, the structure of a process can be described by the process graph (see [4]) 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* or *P-graph* in short. The set of vertices of this directed graph is $M \cup O$, and the set

*This work has been supported by the Ministry of Culture and Education of Hungary, Grant FKFP 0008/1999 and by the Hungarian National Foundation for Scientific Research, Grant TO 30074.

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

‡Research Group on Artificial Intelligence, Hungarian Academy of Sciences, Aradi vértanúk tere 1, H-6720 Szeged, Hungary

of arcs is $A = A_1 \cup A_2$ where $A_1 = \{(X,Y) : Y = (\alpha,\beta) \in O$ and $X \in \alpha\}$ and $A_2 = \{(Y,X) : Y = (\alpha,\beta) \in O$ and $X \in \beta\}$. If there exist vertices $X_1, X_2, \ldots, X_n$, such that $(X_1,X_2), (X_2,X_3), \ldots, (X_{n-1},X_n)$ are arcs of process graph (M,O), then the path determined by these arcs is denoted by $[X_1, X_n]$.

Now, let $o \subseteq O$ be arbitrary. Let us define the following functions on set o

$$mat^{in}(o) = \bigcup_{(\alpha,\beta)\in o} \alpha, \qquad mat^{out}(o) = \bigcup_{(\alpha,\beta)\in o} \beta,$$

and

$$mat(o) = mat^{in}(o) \bigcup mat^{out}(o).$$

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 from structural point of view. For this reason, let M^* be an arbitrarily fixed possibly 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 $M^* \cap O = \emptyset$, furthermore, α and β are finite sets for every $(\alpha,\beta) = u \in O$.

Then, process graph (M,O), where $M = \cup\{\alpha \cup \beta : (\alpha,\beta) \in O\}$, presents the interconnections among the operating units of O. Furthermore, every feasible process network, 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 the feasible process networks. If we do not consider further constraints such as material balance, then the subgraphs of (M,O) which can be assigned to the feasible process networks have common combinatorial properties. They are studied in [4] 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 conditions are satisfied:

$(A1)$ $P \subseteq m$,

$(A2)$ $\forall X \in m$, $X \in R \Leftrightarrow$ no (Y,X) arc in the process graph (m,o),

$(A3)$ $\forall Y_0 \in o$, $\exists$ path $[Y_0, Y_n]$ with $Y_n \in P$,

$(A4)$ $\forall X \in m$, $\exists(\alpha,\beta) \in o$ such that $X \in \alpha \cup \beta$.

The set of solution-structures of $\mathbf{M} = (P,R,O)$ will be denoted by $S(P,R,O)$ or $S(\mathbf{M})$.

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

PNS problem with weights

Let a structural model of PNS problem $\mathbf{M} = (P, R, O)$ be given. Moreover, let w be a positive real-valued function defined on O, the weight function. The basic model is then

$$(1) \qquad \min\{\sum_{u \in o} w(u) : (m, o) \in S(P, R, O)\}.$$

It is known (see [1],[2], and [10]) that this problem is NP-complete. In what follows, for the sake of simplicity, we call the elements of $S(\mathbf{M})$ *feasible solutions* and by PNS problem we mean a PNS problem with weights.

It is a basic observation that if (m, o) and (m', o') are solution-structures of $\mathbf{M}$, then $(m, o) \cup (m', o')$ is also a solution-structure of $\mathbf{M}$. This yields that $S(\mathbf{M})$ has a greatest element called *maximal structure* provided that $S(\mathbf{M}) \neq \emptyset$. Indeed, the maximal structure is the union of all the solution-structures of $\mathbf{M}$. Obviously, the P-graph of an arbitrary PNS problem can contain unnecessary operating units and materials. On the basis of the maximal structure, we can disregard from these unnecessary operating units and materials as follows. Let $(\bar{M}, \bar{O})$ denote the P-graph of the maximal structure. Then, the P-graph of structural model $\bar{\mathbf{M}} = (P, R \cap \bar{M}, \bar{O})$ is $(\bar{O}, \bar{M})$, and since each solution-structure of $\mathbf{M}$ is a subgraph of $(\bar{M}, \bar{O})$, it is a solution-structure of $\bar{\mathbf{M}}$, and conversely. Consequently, $S(\mathbf{M}) = S(\bar{\mathbf{M}})$. On the other hand, $\bar{\mathbf{M}}$ does not contain any unnecessary operating unit and material. Structural model $\bar{\mathbf{M}}$ is called *reduced structural model* of PNS.

To determine the reduced structural model for a PNS problem, an effective procedure is presented in [6], [7]; it can decide if $S(\mathbf{M})$ is empty; if $S(\mathbf{M})$ is not empty, the algorithm provides the corresponding maximal structure. Regarding the significance of this reduction, an empirical analysis is presented in [11], where the reduction procedure is executed on randomly generated PNS problems. It turned out that the decrease of size is about 47%.

Now, we recall this algorithm. This procedure consists of two major parts. The first part is intended to reduce the set of available operating units by eliminating some or all inappropriate operating units. Even if one desired product cannot be generated by any of the remaining operating units, no solution-structure exists for the structural model of PNS under consideration; consequently, there is no maximal structure. If it is still possible to have the maximal structure, then the second part of the algorithm constructs a P-graph from a subset of the operating units left after the first part, which is exactly the maximal structure. To elucidate this procedure, let a structural model of PNS be given by $\mathbf{M} = (P, R, O)$.

Algorithm for Maximal Structure Generation

1. Reduction

Initialization

- Let $O_0 = O \setminus \{(\alpha,\beta) : (\alpha,\beta) \in O \;\&\; \beta \cap R \neq \emptyset\}$ and $M_0 = mat(O_0)$. If $P \not\subseteq M_0$, then terminate since there is no maximal structure for **M**. If not, then let $T_0 = \{X : X \in M_0 \setminus R \;\&\; ((\alpha,\beta) \in O_0 \longrightarrow X \notin \beta)\}$. Finally, set $r := 0$.

Iteration

- *Step 1.1.* If $T_r = \emptyset$, then proceed to the initialization for building. If not, then choose a material X from T_r and set $O_X = \{(\alpha,\beta) : (\alpha,\beta) \in O_r \;\&\; X \in \alpha\}$. Let $O_{r+1} = O_r \setminus O_X$; moreover, $M_{r+1} = mat(O_{r+1})$. If $P \not\subseteq M_{r+1}$, then terminate since there is no maximal structure for **M**. If not, then construct set T'_r by $T'_r = \{Y : Y \in mat^{out}(O_X) \;\&\; Y \notin mat^{out}(O_{r+1}) \;\&\; Y \in mat^{in}(O_{r+1})\}$. Let $T_{r+1} = (T_r \cap M_{r+1}) \cup T'_r$. Set $r := r + 1$ and proceed to the following iteration for reduction.

2. Building

Initialization

- Let $W_0 = P$, $m_0 = \emptyset$ and $o_0 = \emptyset$; moreover, set $s := 0$.

Iteration

- *Step 2.1.* If $W_s = \emptyset$, then terminate. There exists at least one solution-structure for **M**. In particular, $(\bar{m}, o_s)$ is the maximal structure of **M** where $\bar{m} = mat(o_s)$. If $W_s \neq \emptyset$, then proceed to Step 2.2.

- *Step 2.2.* Choose one material from W_s; denote this material by X, and let $m_{s+1} = m_s \cup \{X\}$. Then, form set $O^*_X = \{(\alpha,\beta) : (\alpha,\beta) \in O_r \;\&\; X \in \beta\}$. Also, let $o_{s+1} = o_s \cup O^*_X$ and $W_{s+1} = (W_s \cup mat^{in}(O^*_X)) \setminus (R \cup m_{s+1})$. Then, set $s := s + 1$, and proceed to the succeeding iteration for building.

2 Merging reduction

While the general reduction presented above renders possible to exclude the unnecessary operating units and materials from the investigation, the merging reduction compresses the P-graph by merging some of its operating units. If $u_1 = (\alpha_1, \beta_1)$ and $u_2 = (\alpha_2, \beta_2)$, then one can merge these two operating units into a new operating unit defined by $u = (\alpha_1 \cup \alpha_2, \beta_1 \cup \beta_2)$. It is worth noting that after the merging of two or more operating units, we obtain a new structural model of PNS.

If we want to use this new structural model for solving the original problem, then a strong relationship must be established between the feasible solutions of the two problems. To establish this relationship, it is a basic question that which operating units are mergeable.

For this purpose, let $\mathbf{M} = (P, R, O)$ be a reduced structural model of PNS. Then, operating units $u_1, u_2 \in O$ are called *mergeable* if for any feasible solution, either both of them are contained in it or both of them are excluded from it. Formally stated, u_1 and u_2 are mergeable if $u_1 \in o$ implies $u_2 \in o$, and conversely, for every feasible solution $(m, o) \in S(\mathbf{M})$.

It can be readily seen that this relation is reflexive, symmetric, and transitive, and thus, it is an equivalence relation on set O which is denoted by $\equiv$. Let us define structural model $\mathbf{M}/\equiv = (P, R, O^*)$ by

$$O^* = \{(\cup\{\alpha_t : u_t = (\alpha_t, \beta_t) \in C(u)\}, \cup\{\beta_t : u_t = (\alpha_t, \beta_t) \in C(u)\}) : u \in O\}$$

where $C(u)$ denotes the equivalence class containing u. The visual meaning of $\mathbf{M}/\equiv$ can be given as follows. For each equivalence class, we merge all of the operating units belonging to this class into a new operating unit. This new operating unit will substitute the original ones in $\mathbf{M}/\equiv$. Obviously, $\mathbf{M}/\equiv$ is a structural model of PNS and its maximal structure is (M, O^*). Now, we define a mapping φ of $M \cup O$ onto $M \cup O^*$. For every $X \in M$, let $\varphi(X) = X$, furthermore, for every $u_s \in C(u)$, let $\varphi(u_s) = (\cup\{\alpha_t : u_t \in C(u)\}, \cup\{\beta_t : u_t \in C(u)\})$. As it is usual, we shall use the notation $\varphi(o) = \{\varphi(u) : u \in o\}$ and $\varphi(m) = \{\varphi(X) : X \in m\}$ for a subset o of O and for a subset m of M, respectively. Using this extension, we can take the image of an arbitrary P-graph (m, o) of (M, O) under φ as $(\varphi(m), \varphi(o))$. This mapping is denoted also by φ.

The following statement establishes a strong relationship between the two sets $S(\mathbf{M})$ and $S(\mathbf{M}/\equiv)$ of feasible solutions.

Theorem 1. *Mapping φ is a bijective mapping of $S(\mathbf{M})$ onto $S(\mathbf{M}/\equiv)$.*

Proof. Let $(m, o) \in S(\mathbf{M})$ be an arbitrary feasible solution. First, it is shown that $(\varphi(m), \varphi(o))$ is a feasible solution of $\mathbf{M}/\equiv$. Obviously, $(\varphi(m), \varphi(o))$ is such a P-graph which is a subgraph of (M, O^*). Consequently, it is enough to prove that $(\varphi(m), \varphi(o))$ satisfies conditions $(A1)$ through $(A4)$. Condition $(A1)$ is clearly valid, since $P \subseteq m = \varphi(m)$. To prove condition $(A2)$, let us observe that mapping φ preserves the sources. Regarding condition $(A3)$, let $u \in \varphi(o)$ be an arbitrary operating unit. Then, there is at least one $u_j \in o$ such that $\varphi(u_j) = u$. On the other hand, $(m, o) \in S(\mathbf{M})$, and thus, on the base of $(A3)$, there is a $[u_j, Y_n]$ path in (m, o) with $Y_n \in P$. Now taking the images under φ of the vertices of this path, we obtain a path $[u, Y_n']$ in $(\varphi(m), \varphi(o))$ where $Y_n' \in P$ which implies the validity of $(A3)$. Finally, to prove $(A4)$, let $X \in \varphi(m)$ be arbitrary. Then, $X \in m$, and by property $(A4)$, there exists an operating unit $u_j = (\alpha_j, \beta_j)$ such that $X \in \alpha_j \cup \beta_j$. Let $\varphi(u_j) = (\alpha, \beta)$. Then, by the definition of $\varphi(u_j)$, $X \in \alpha \cup \beta$, thereby validating $(A4)$.

Now, it is proven that φ is an injective mapping. For this purpose, let $(m,o) \neq (m',o') \in S(\mathbf{M})$. If $m \neq m'$, then $\varphi(m) \neq \varphi(m')$, and thus, the images are different. Otherwise, $o \neq o'$. To prove this case by contradiction, let us suppose that $(\varphi(m), \varphi(o)) = (\varphi(m'), \varphi(o'))$. Since $o \neq o'$, without loss of generality, we may assume that there exists a $u' \in o'$ with $u' \notin o$. Let $\varphi(u') = u$. Since $(\varphi(m), \varphi(o)) = (\varphi(m'), \varphi(o'))$, there exists a $\bar{u} \in o$ with $\varphi(\bar{u}) = u$. Then, by the definition of φ, $\bar{u} \equiv u'$, and thus, by the definition of the equivalence relation, $u' \in o$ which is a contradiction. Consequently, φ is a one-to-one mapping.

Finally, we show that φ is a mapping of $S(\mathbf{M})$ onto $S(\mathbf{M}/\equiv)$. For this purpose, let us consider an arbitrary feasible solution denoted by (m^*, o^*) of $S(\mathbf{M}/\equiv)$. Let $m = m^*$ and $o = \{u_j : u_j \in O \;\&\; \varphi(u_j) \in o^*\}$. Obviously, $\varphi(m,o) = (\varphi(m), \varphi(o)) = (m^*, o^*)$. Therefore, we have to prove that (m,o) is a feasible solution of $\mathbf{M}$. It can be easily seen that (m,o) is such a P-graph which is a subgraph of (M,O). Thus, we have to prove that (m,o) satisfies conditions $(A1)$ through $(A4)$.

Since $(m^*, o^*) \in S(\mathbf{M}/\equiv)$, condition $(A1)$ implies $P \subseteq m^*$. On the other hand, $m = m^*$, thereby indicating the validity of $(A1)$ for (m,o).

Since the ancestor of a source in (m^*, o^*) is a source in (m,o) under φ, and (m^*, o^*) satisfies $(A2)$, (m,o) satisfies condition $(A2)$ as well.

To prove $(A3)$ by contradiction, let us suppose that $(A3)$ is not valid for (m,o). Let us denote by o_1 the set of operating units in o from which there is no path in (m,o) into some required product, *i.e.*, let

$$o_1 = \{u_j : u_j \in o \;\&\; \text{no } [u_j, Y] \text{ path exists with } Y \in P \text{ in } (m,o)\}.$$

By our assumption, $o_1 \neq \emptyset$. Now, let us consider P-graph (m',o') where $o' = o \setminus o_1$ and $m' = mat(o')$. We shall prove that (m',o') is a feasible solution of $\mathbf{M}$.

Since $(m^*, o^*) \in S(\mathbf{M}/\equiv)$, $(A1)$ implies that for any $X \in P$, there exists an operating unit u producing X directly. Taking an ancestor of u, we obtain that there is an operating unit denoted by u' in o producing X directly, and thus, u' is not contained in o_1. Consequently, $u' \in o'$, thereby resulting in $P \subseteq m'$, *i.e.*, (m',o') satisfies condition $(A1)$.

To prove $(A2)$, let $X \in m'$ be arbitrary. If $X \in R$, then X is a source in (m^*, o^*), and since the ancestor of X is a source in (m,o) under φ, X is a source of (m,o). But $(m',o') \subseteq (m,o)$, and thus, X is a source in (m',o'). Conversely, let us suppose that X is a source in (m',o'). Then, X is a source in (m,o). Indeed, in the opposite case, X would be an output material of at least one operating unit from o_1. Let u_1 denote such an operating unit. Then, there is a $[u_1, Y]$ path in (m,o) since X is a source in (m',o'), and thus, X is an input material for some operating unit in o'. This fact contradicts the definition of o_1. Hence, X is a source in (m,o). In this case, X is a source in (m^*, o^*), and since $(A2)$ is valid for (m^*, o^*), $X \in R$. Consequently, (m',o') satisfies $(A2)$.

The validity of conditions $(A3)$ and $(A4)$ follows from the definitions of o_1 and (m',o'), and thus, we obtain that (m',o') is a feasible solution of $\mathbf{M}$.

Now, let us observe that $\varphi(m', o') = (m^*, o^*) = \varphi(m, o)$, which implies $(m', o') = (m, o)$ since φ is injective. Hence, $o_1 = \emptyset$ which is a contradiction. Consequently, $(A3)$ is valid for (m, o).

In proceeding to prove the validity of condition $(A4)$, let $X \in m$ be an arbitrary material. Then $X \in m^*$, and since (m^*, o^*) satisfies $(A4)$, there exists an operating unit $u = (\alpha, \beta) \in o^*$ such that $X \in \alpha \cup \beta$. This implies that there exists an operating unit $u_j = (\alpha_j, \beta_j) \in o$ such that $\varphi(u_j) = u$ and $X \in \alpha_j \cup \beta_j$. Indeed, in the opposite case, we would have that $X \notin \alpha \cup \beta$ which is a contradiction.

This completes the proof of Theorem 1.

Let us equip structural model $\mathbf{M}/\equiv$ with the weight function $\bar{w}$ defined as follows. For every $u \in O^*$, let $\bar{w}(u) = \sum_{u_t \in C(u')} w(u_t)$ where $\varphi(u') = u$. Since the equivalent operating units have an identical image, function $\bar{w}$ is well-defined. The constructed new model is then

$$\min\{\sum_{u \in o} \bar{w}(u) : (m, o) \in S(\mathbf{M}/\equiv)\}. \tag{2}$$

Extend the weight functions for the feasible solutions in the following way. For any $(m, o) \in S(\mathbf{M})$ and $(m^*, o^*) \in S(\mathbf{M}/\equiv)$, let $w(m, o) = \sum\{w(u) : u \in o\}$ and $\bar{w}(m^*, o^*) = \sum\{\bar{w}(u) : u \in o^*\}$. Then, $w(m, o) = \bar{w}(\varphi(m, o))$ is valid, for all feasible solutions $(m, o) \in S(\mathbf{M})$. On the basis of this observation and Theorem 1, the validity of the following statement is obvious.

Theorem 2. *The image of an optimal solution of problem* (1) *under* φ *is an optimal solution of problem* (2), *and conversely, the image of an optimal solution of problem* (2) *under* φ^{-1} *is an optimal solution of problem* (1).

To execute the merging reduction on an instance, we need to determine the equivalence relation introduced. For this reason, a further notation is introduced. Let $\mathbf{M} = (P, R, O)$ be a reduced structural model of PNS with $S(\mathbf{M}) \neq \emptyset$. Furthermore, let $u_j \in O$ be arbitrary. Then, we can construct a new structural model of PNS, $\mathbf{M}(u_j) = (P, R, O \setminus \{u_j\})$. Let us denote the maximal structure of $\mathbf{M}(u_j)$ by (M_j, O_j) provided that it exists. If it does not exist, then let $M_j = O_j = \emptyset$. Then, we have the following statement.

Theorem 3. *For every* $u_i, u_j \in O$, $u_i \equiv u_j$ *if and only if* $u_i \in O \setminus O_j$ *and* $u_j \in O \setminus O_i$ *are simultaneously valid.*

Proof. Let us suppose that $u_i \in O \setminus O_j$ and $u_j \in O \setminus O_i$ for some $u_i \neq u_j \in O$. Let us consider an arbitrary feasible solution (m, o). We have to distinguish three cases.

Case 1. (m, o) does not contain u_i. Then, (m, o) is a subset of (M_i, O_i), and thus, by our assumption, (m, o) does not contain u_j.

Case 2. (m, o) does not contain u_j. In this case, (m, o) is a subset of (M_j, O_j), and hence, by our assumption, (m, o) does not contain u_i.

Case 3. (m, o) contains both u_i and u_j.

Since there is no further case, we have proved that $u_i \equiv u_j$.

In order to prove the necessity of the condition, let us suppose that $u_i \equiv u_j$ for some $u_i \neq u_j \in O$. Let us consider the structural models $\mathbf{M}(u_i)$ and $\mathbf{M}(u_j)$. Then, (M_j, O_j) is the union of those feasible solutions which do not contain u_j provided that there exists such a feasible solution. Since $u_i \equiv u_j$ none of these feasible solutions contains u_i. Consequently, their union does not contain u_i, *i.e.*, $u_i \in O \setminus O_j$. We can obtain by a similar argument that $u_j \in O \setminus O_i$. If every feasible solution contains u_j, *i.e.*, $O_j = \emptyset$, then from $u_i \equiv u_j$, it follows that every feasible solution contains u_i as well, and thus $O_i = \emptyset$, and the corresponding inclusions are obviously valid.

From Theorem 3, we get immediately the following corollary.

Corollary. *If $O_j = O \setminus \{u_j\}$, then u_j is not mergeable with any other operating unit.*

Now, by Theorem 3 and the Maximal Structure Generation algorithm, we obtain the following procedure to determine the required equivalence relation where it is assumed that $O = \{u_1, \ldots, u_n\}$.

Procedure

Initialization

- *Step 1.* Set $i := 1$, $k := 1$, $N = \{1, \ldots, n\}$.
- *Step 2.* Determine the maximal structure of $\mathbf{M}(u_i)$ by the maximal structure generation algorithm. If $O_i = O \setminus \{u_i\}$, then let $V_k = \{u_i\}$, $N = N \setminus \{i\}$, $k = k + 1$. Proceed to Step 3.
- *Step 3.* If $i = n$, then proceed to Step 4. Otherwise, let $i = i + 1$, and proceed to Step 2.
- *Step 4.* Terminate if $N = \emptyset$. Otherwise, let i denote the smallest element of N. Let $J = \{t : t \in N \,\&\, u_t \in O \setminus O_i\}$. Let $V = \emptyset$, and proceed to Step 5.
- *Step 5.* If $J = \emptyset$, then let $N = N \setminus \{i\}$, $V_k = V \cup \{u_i\}$, $k = k + 1$, and proceed to Step 4. Otherwise, proceed to Step 6.
- *Step 6.* Choose an element j from J. Let $J = J \setminus \{j\}$. If $u_i \in O \setminus O_j$, then let $V = V \cup \{u_j\}$, $N = N \setminus \{j\}$. Proceed to Step 5.

As a result of this procedure, we obtain the equivalence classes belonging to the required equivalence relation as $V_1, \ldots, V_k$.

Regarding the merging reduction one can raise the following questions.

(1) *Does the merging reduction decrease the measure of practical problems or is it only a theoretical aspect?*

(2) *Is the decrease of the measure able to balance the higher complexity of the operating units caused by the merging reduction with respect to the running times of the known procedures for solving PNS problems?*

Both questions were investigated empirically. The corresponding computational experiences and their results are presented in the following section.

3 Empirical analysis

The first empirical analysis is devoted to the estimation of the decrease of measure. More precisely, it was investigated that how large the decrease of the model size was in general. For this reason, we considered 1000 randomly generated PNS problems (for their generation *cf.* [11]), and for each problem, the maximal structure was determined, then the merging reduction was performed. Figure 1 shows the average numbers of the operating units in the initial problem, in the maximal structure, and in the problem after the merging reduction. Figure 2 presents the same information in percent.

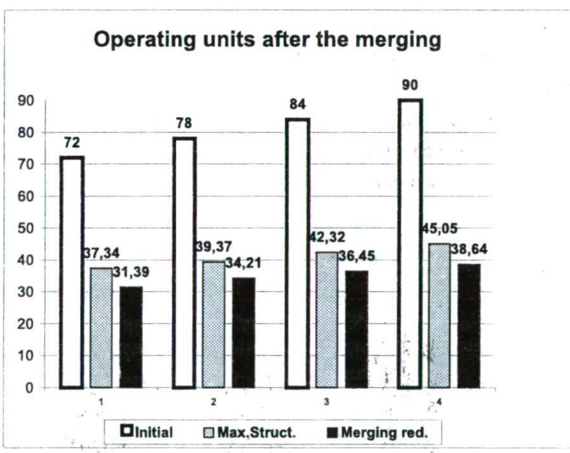

Figure 1: Average number of operating units.

As the results of the empirical analysis show, the merging reduction results in a decrease of 7% in general. It is obvious that the price of this decrease is that the new problem will be more complex than the initial one, namely, the operating units will have more input and output materials. Therefore, it is interesting to study the behaviours of the available procedures for solving PNS problems on the problems obtained by merging reduction. For this reason, we executed the following empirical investigation. Three procedures, the Accelerated Branch-and-Bound Algorithm, ABBA in short (see [8]), the Modified Accelerated Branch-and-Bound Procedure, in short MABBA (*cf.* [9]), and a version of the Refined Modified Accelerated Branch-and-Bound Procedure, in short RMABBA [11] were involved in

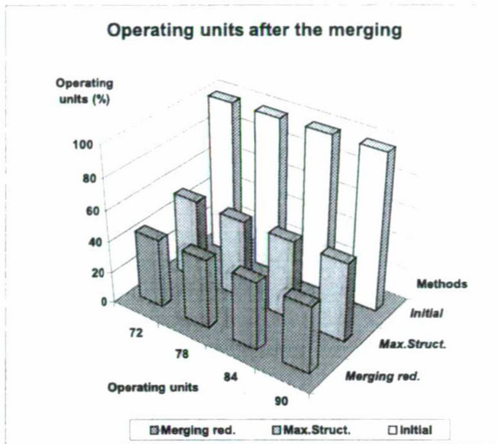

Figure 2: Average number of operating units in percent.

the empirical analysis. 1000 PNS problems with 100 materials were generated randomly, and for each of them the maximal structure was determined and the merging reduction was performed as well. Then, the two problems (problem belonging to the maximal structure and problem obtained by the merging reduction) were solved by the three procedures considered. Figure 3 shows the averages of the running times in percent for the different procedures.

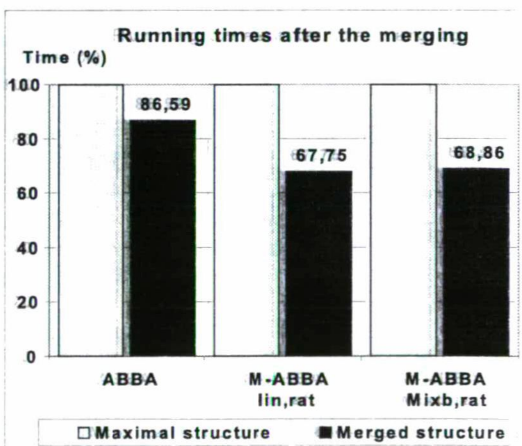

Figure 3: Behaviours of the procedures.

Conclusions. The empirical analysis shows that the merging reduction is appropriate to get further reduction of the model, moreover, the higher complexity of the operating units not necessarily implies longer running time for the procedures considered. The smaller measure of the PNS problem resulted in a smaller running time for the procedures investigated even if the complexity of the operating units became higher.

References

[1] Blázsik, Z. and B. Imreh, A note on connection between PNS and set covering problems, *Acta Cybernetica* **12** (1996), 309-312.

[2] Fülöp, J., B. Imreh, F. Friedler, On the reformulation of some classes of PNS problems as set covering problems, *Acta Cybernetica*, **13** (1998), 329-397.

[3] Friedler, F., L. T. Fan, B. Imreh, Process Network Synthesis: Problem Definition, *Networks* **28** (1998), 119-124.

[4] Friedler, F., K. Tarján, Y. W. Huang, and L. T. Fan, Graph-Theoretic Approach to Process Synthesis: Axioms and Theorems, *Chem. Eng. Sci.* **47**(8) (1992), 1973-1988.

[5] Friedler, F., K. Tarján, Y.W. Huang, and L.T. Fan, Combinatorial Algorithms for Process Synthesis, *Computers chem. Engng.* **16** (1992), S313-S320.

[6] Friedler, F., K. Tarján, Y. W. Huang, and L. T. Fan, Graph-Theoretic Approach to Process Synthesis: Polinomyal Algorithm for maximal structure generation, *Computer chem. Engng.* **17** (1993), 924-942.

[7] Friedler, F., K. Tarján, Y. W. Huang, and L. T. Fan, Combinatorial Algorithms for Process Synthesis, *Computer chem. Engng.* **16** (1992), 313-320.

[8] Friedler, F., J. B. Varga, E. Fehér, and L. T. Fan, Combinatorially Accelerated Branch-and-Bound Method for Solving the MIP Model of Process Network Synthesis, Nonconvex Optimization and its Applications, Kluwer Academic Publisher, Norwell, MA, U.S.A. (in press).

[9] Imreh, B., F. Friedler, L. T. Fan, An Algorithm for Improving the Bounding Procedure in Solving Process Network Synthesis by a Branch-and-Bound Method *Developments in Global Optimization*, editors: I. M. Bonze, T. Csendes, R. Horst, P. M. Pardalos, Kluwer Academic Publisher, Dordrecht, Boston, London, 1996, 301-348.

[10] Imreh, B., J. Fülöp, F. Friedler, On the Equivalence of the Set Covering and Process Network Synthesis Problems, *Networks*, submitted for publication.

[11] Imreh, B., G. Magyar, Empirical Analysis of Some Procedures for Solving Process Network Synthesis Problem, *Journal of Computing and Information Technology*, to appear.