

# Effective Modeling for Process Synthesis

Ferenc Friedler<sup>[1]</sup>

Department of Computer Science, University of Pannonia  
Veszprem, Egyetem u. 10. H-8200, Hungary

**Abstract**—Most frequently, process synthesis is executed by resorting to mathematical programming. Nevertheless, its most crucial step, *i.e.*, the generation of the mathematical model, has been largely ignored. The present contribution demonstrates that the mathematical model generated in algorithmic process synthesis based on mathematical programming affects profoundly the quality of the solution and the necessary computational time. It has been illustrated that the P-graph framework gives rise to a consistent methodology for generating the mathematical model and its solution.

**Key Words** : P-graph, Process synthesis, Model generation, Separation network

## INTRODUCTION

Process synthesis is one of the most, if not the most, important steps in designing any production process including chemical processes; it immensely affects the quality of solution. It is common to resort to a mathematical-programming (MILP or MINLP) method in executing process synthesis. For such a method, the mathematical model of any process synthesis problem needs to be couched in the parlance of MILP or MINLP (see Fig. 1). In practice, however, a process synthesis problem is seldom presented as such. Hence, its mathematical model must be derived from the original definition of process synthesis problems. For example, a separation-network synthesis (SNS) problem is given as illustrated in Fig. 2. In this SNS problem, the optimal network of sharp separators, dividers, and mixers must be determined for two three-component feed streams and three pure product streams where the cost of a network is the sum of costs of the separators. Thus, at the outset, the mathematical model needs to be generated followed by its solution, thereby indicating that the model generation is the key to process synthesis; the quality of the solution depends on the resultant model. It appears that the available literature on process synthesis is void of model generation; the model generation for process synthesis, *i.e.*, the generation of the appropriate MILP or MINLP model, is treated only in a limited number of papers (Kovacs *et al.*, 2000). The main emphasis of the current contribution is on model generation (see Fig. 3).

Let us now suppose that a process synthesis problem is framed by specifying a set of potential feed streams (raw materials) and product streams together with the mathematical models of plausible operating units. The problem aims at the generation

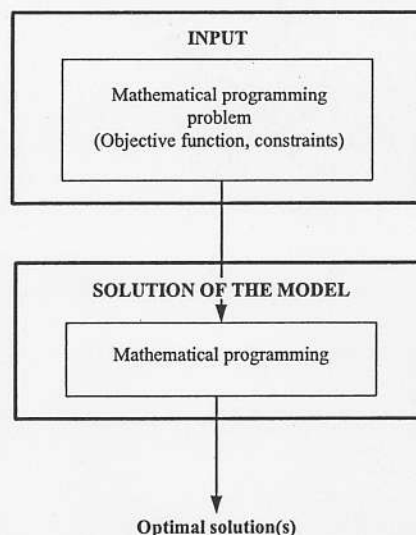


Fig. 1. Conventional mathematical programming.

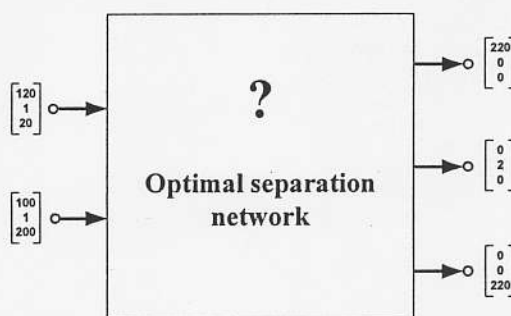


Fig. 2. Definition of a simple separation-network synthesis problem.

or identification of the optimal network of operating units through optimization. It has been repeatedly demonstrated that various mathematical models of a given process synthesis problem may result in solu-

<sup>[1]</sup> E-mail: friedler@dcs.vein.hu

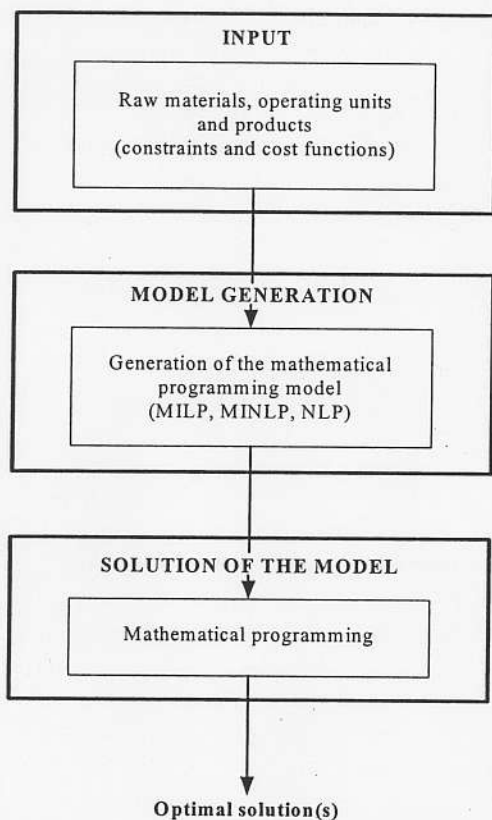


Fig. 3. Main steps of algorithmic process synthesis.

tions with significantly different values of the cost function. Thus, it is of the utmost importance that the mathematical model generated indeed yields the optimal solution of the process synthesis problem as originally framed.

Process synthesis is initiated most commonly by constructing the so-called super-structure, which in turn, gives rise to the mathematical model necessary for identifying the optimal solution (see Fig. 4). It is, therefore, essential that the structure or network of the optimal solution be contained in the super-structure; otherwise, the optimality of the resultant solution cannot be assured. A parametric study of a simple class of process synthesis problems will illustrate that such a super-structure cannot be generated readily.

#### PARAMETRIC STUDY OF A SIMPLE CLASS OF PROCESS SYNTHESIS PROBLEMS

Kovacs *et al.* (1998) have analyzed the set of potentially optimal networks for separation-network synthesis problems with two three-component feed streams and three pure product streams with simple sharp separators, dividers, and mixers (see Fig. 5). The sum of the costs of its separators is regarded as the cost of the network. By following the usual convention, the cost of a separator for separation between

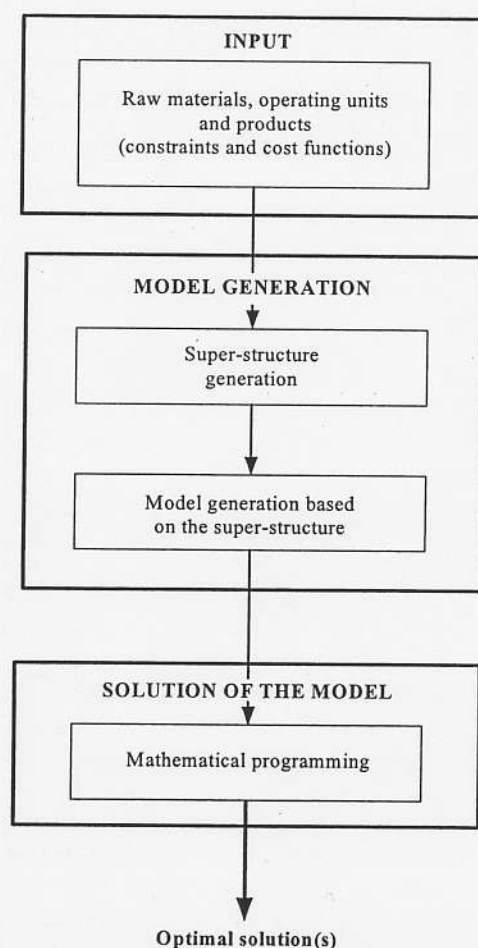


Fig. 4. Algorithmic process synthesis based on a super-structure.

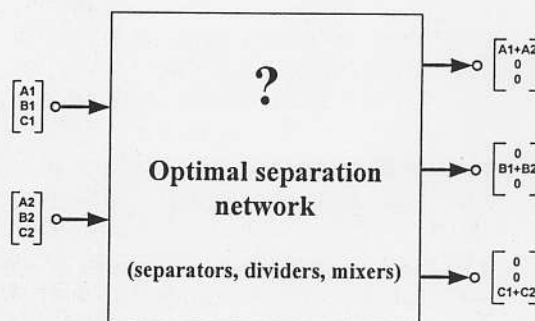


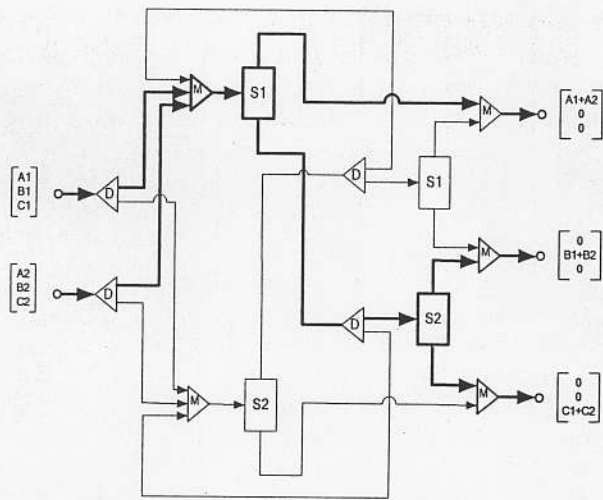
Fig. 5. Class 1 SNS problems.

components  $i$  and  $(i+1)$  is computed by the formula,

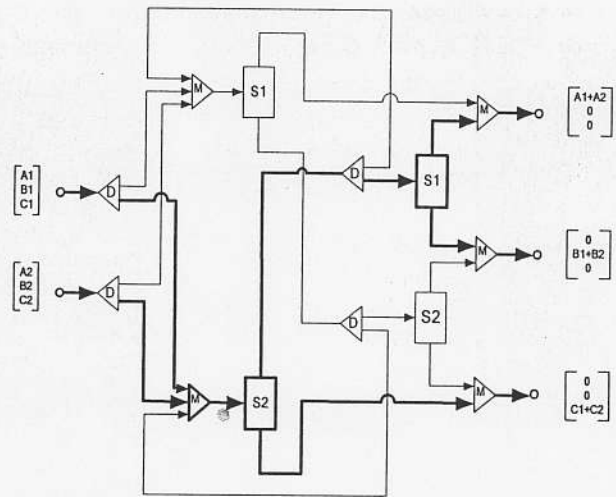
$$(fD_i)^b, \quad (1)$$

where  $f$  is the mass load through the separator;  $D_i$ , the degree of difficulty of the separation between components  $i$  and  $(i+1)$ ; and  $b$ , a constant between 0 and 1. (For simplicity,  $b$  is taken to be 0.6). This class of problems will be called the class 1 SNS problems. Note that the process synthesis problem as defined in Fig. 2 is one instance of the class 1 SNS problems.

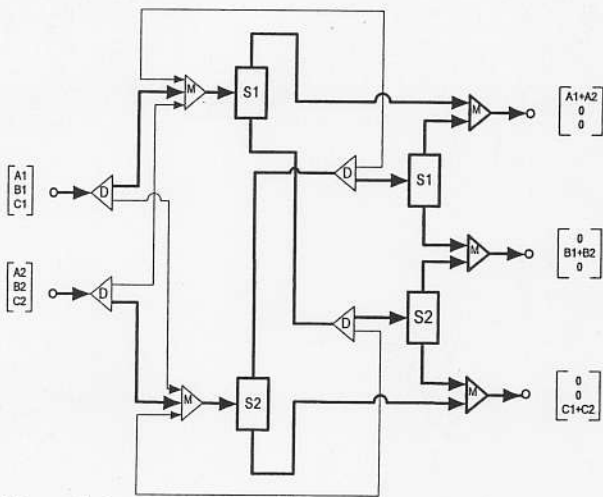
As illustrated in Fig. 6, fourteen possible



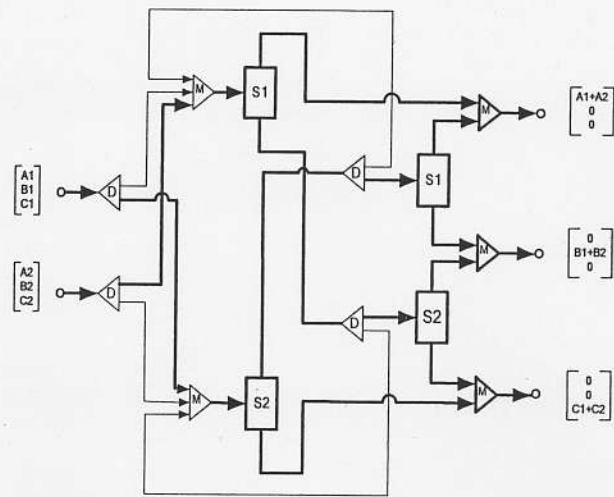
Network 1



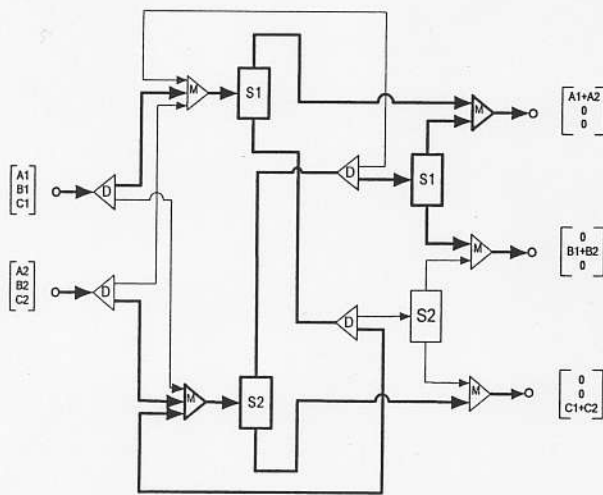
Network 2



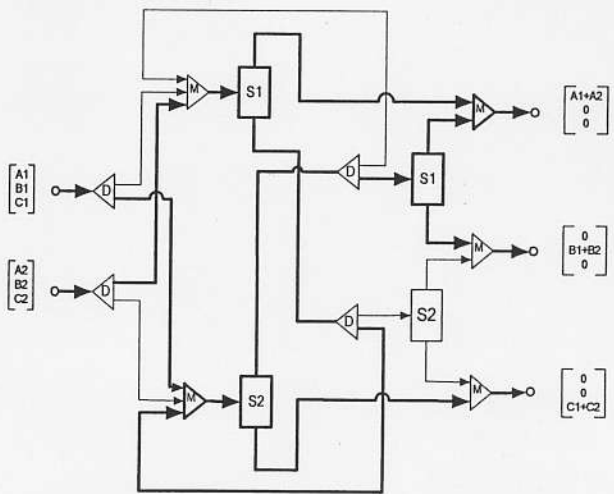
Network 3



Network 4

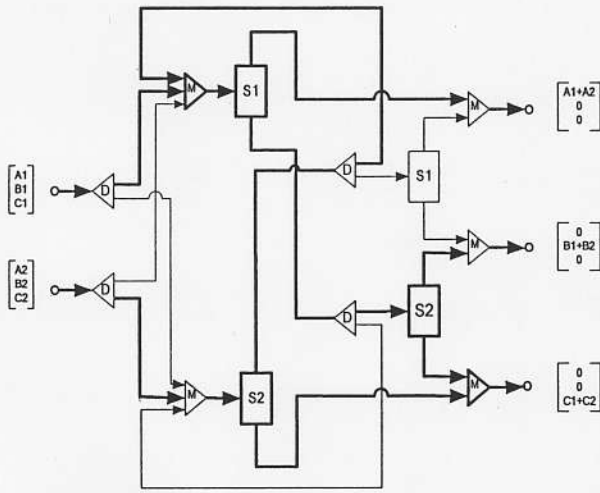


Network 5

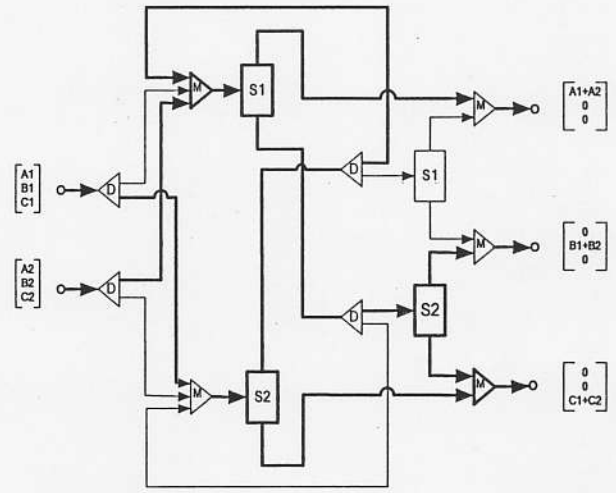


Network 6

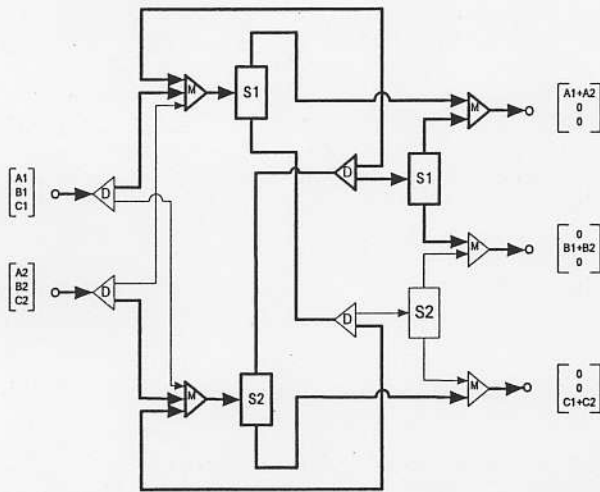
Fig. 6. Feasible networks of the class 1 SNS problems.



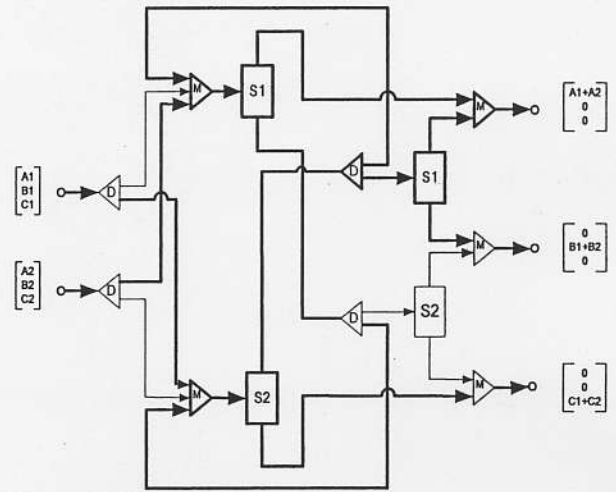
Network 7



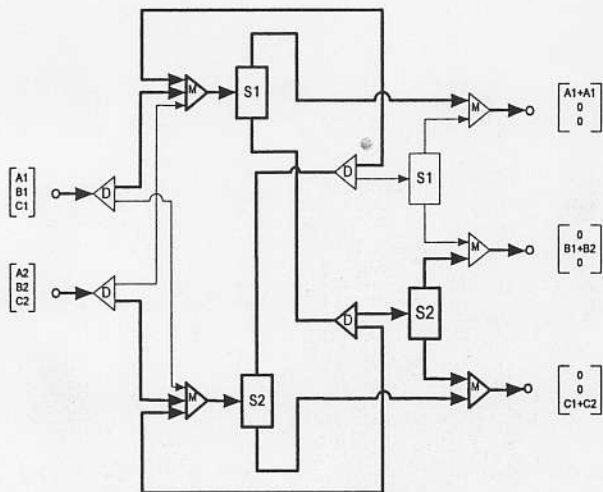
Network 8



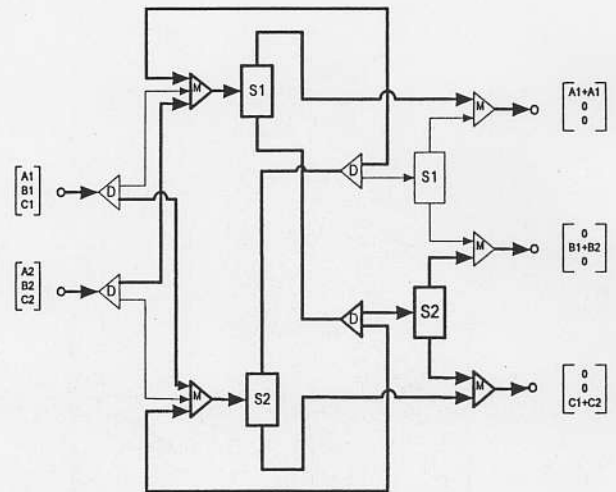
Network 9



Network 10



Network 11



Network 12

Fig. 6 (cont'd).

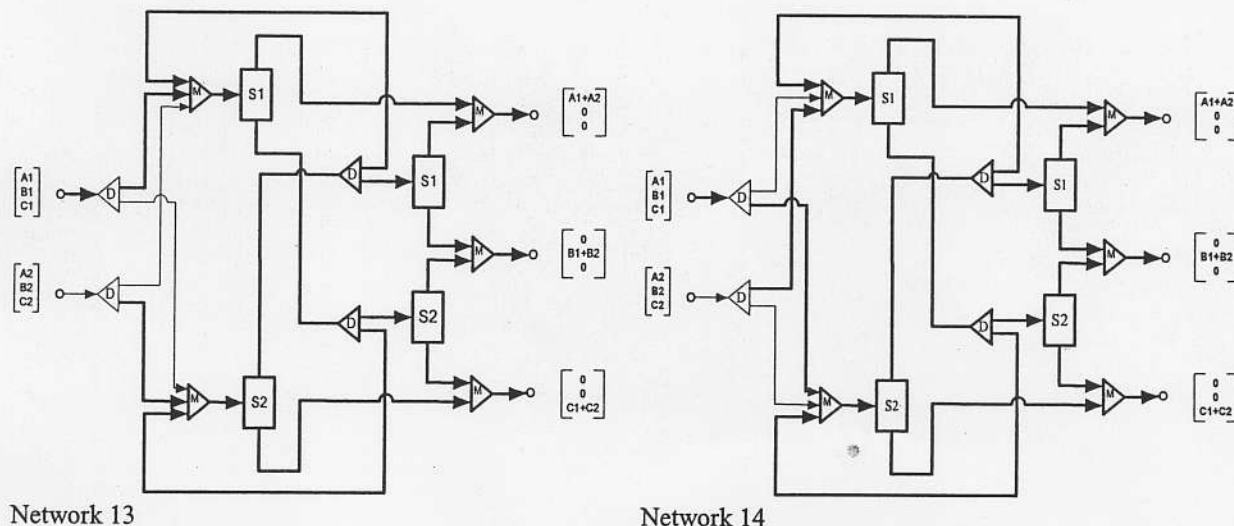


Fig. 6 (cont'd).

Table 1. Feed streams to ten of the fourteen networks of Fig. 6 that are the optimal solutions for the respective SNS problems.

F1	F2	Optimal Network
[130,1,100]	[100,50,30]	network 1
[1,50,100]	[100,1,200]	network 2
[200,1,100]	[5,10,100]	network 5
[5,10,100]	[200,1,100]	network 6
[100,10,5]	[100,1,200]	network 7
[100,1,200]	[100,10,5]	network 8
[200,1,100]	[1,1,100]	network 9
[1,1,100]	[200,1,100]	network 10
[100,2,2]	[100,1,200]	network 11
[100,1,200]	[100,2,2]	network 12

structures emerge for this class of SNS problems (Kovacs *et al.*, 1998). A parametric study of it has indicated that ten of these fourteen networks can be optimal depending on the values of the problem's parameters (see Table 1). As can be seen on Fig. 6, the only difference between networks 3 and 4 is that feed F1 of the former is the same as feed F2 for the latter and vice versa. Similarly, such is the case between networks 5 and 6, between networks 7 and 8, between networks 9 and 10, between networks 11 and 12 and between networks 13 and 14.

Naturally, two separators are sufficient to solve any instance of the class 1 SNS problems: Three pure components are to be generated from two three-component feed streams. The cost function adopted is concave for any separator. It is, therefore, expected that three or more separators, *i.e.*, redundancy, cannot give rise to the optimality. Among the ten networks that are optimal under various circumstances, however, two networks contain two separators each; and eight networks, three separators each. Hence, eight networks contain redundant separators;

these are networks 5 through 12. Moreover, in each of the four networks among them, including networks 5 through 8, the redundant separators are on a path between a feed and a product. In other words, for some instances of the class 1 SNS problems, the optimal solutions based on the mathematical model, which excludes redundancy, are not optimal solutions of the synthesis problems as originally framed (Kovacs *et al.*, 1998). Moreover, the inclusion of a loop in an optimal separation network of the class 1 SNS problems is unexpected (Floudas, 1987). Apparently, it is conventional wisdom that a loop in this class of separation networks leads to inconsequential or purposeless transport of a stream around the network. This is obviously detrimental to the network's performance and increases its cost, thereby preventing the network to be optimal. On the contrary, four of the ten optimal separation networks involve looping. This implies that the optimal solution obtained from the mathematical model excluding loops for the class 1 SNS problems is not the optimal solutions for some of the SNS problems as originally framed (Kovacs *et al.*, 1993). Consequently, the optimal solution of any instance of the class 1 SNS problems can be attained only if the super-structure on which the mathematical model is based includes all potentially optimal networks. It is the union of the ten networks (see Fig. 7).

In an SNS problem for generating multicomponent product streams from multicomponent feed streams, it is often possible to bypass certain amounts of feed streams to some product streams. If the cost of bypassing is negligibly small, it is expected that the extent of bypassing would be always maximal in an optimal solution. Nevertheless, an example is given by Kovacs *et al.* (1995) to illustrate that this is not always the case: Even when the cost

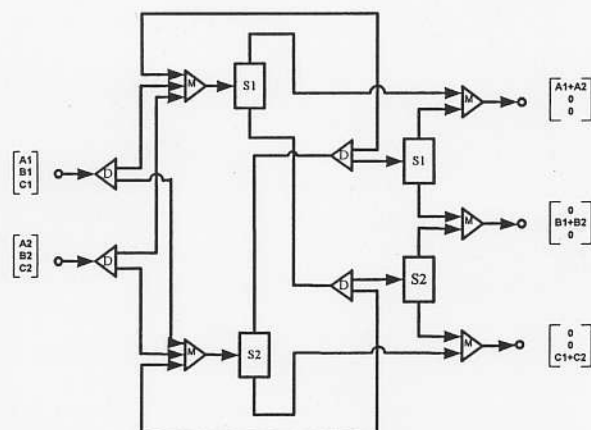


Fig. 7. Rigorous super-structure for the class 1 SNS problems.

of bypassing is zero, it may affect the network structure, thus may resulting in an increase in the network's cost, which offsets any advantage gained from bypassing. Table 2 summarizes the results of analysis by Kovacs *et al.* (1998) in terms of the structural properties of optimal separation networks. Consequently, these properties have to be taken into account in generating a super-structure.

### RIGOROUS SUPER-STRUCTURE

If a super-structure is incomplete, the resultant mathematical model is also incomplete. As such, the attainment of optimality cannot be assured, and most often it is the case. If it is assumed that no more than the ten networks listed in Table 1 can be optimal for any instances of the class 1 SNS problems, the union of these networks (see Fig. 7) always include the network of the optimal solution. Hence, if the mathematical model is based on this super-structure, the optimality of the solution is assured.

**Definition:** Suppose that a systematic procedure is available so that a valid mathematical-programming model can be generated for a network of the given operating units. A network of these operating units is defined to be a rigorous super-structure of a class of process synthesis problem if the optimality of the resultant solution cannot be improved for any instance of this class of problems by any other procedure for network and model generation.

Hence, the network depicted in Fig. 7 is a rigorous super-structure of the class 1 SNS problems, provided that a valid mathematical model can be generated from it with an available procedure. Different types of mathematical models may give rise to the same optimality; nevertheless, the effort required for their solution can vary. To achieve the maximum efficiency for synthesis, it is mandatory that the methods for generating the mathematical models and those for their solution be determined collectively.

### P-GRAPH FRAMEWORK

The P-graph framework has been established for the effective integration of model-generation and solution in process-network synthesis (PNS). The framework includes a specific network representation and algorithms.

Conventional graphs are suitable for analyzing a process structure; however, such graphs are incapable of uniquely representing process structures in process synthesis (Friedler *et al.*, 1992a). Thus, a special directed bipartite graph, P-graph, has been introduced to circumvent this difficulty. It is bipartite since its vertices are partitioned into two sets, and no two vertices in the same set are adjacent in the graph. Vertices in one of the partitions are for representing operating units, and those in the other is for representing materials. Stated formally, let finite sets  $m$  and  $o$  be given with

$$o \subseteq \wp(m) \times \wp(m). \quad (2)$$

A P-graph is defined to be pair  $(m, o)$  where the vertices of the graph are the elements of  $m \cup o$ . The arcs of the graph are the elements of set  $A1 \cup A2$  where

$$A1 = \{(x, y) : y = (\alpha, \beta) \in o \text{ and } x \in \alpha\} \text{ and} \quad (3)$$

$$A2 = \{(y, x) : y = (\alpha, \beta) \in o \text{ and } x \in \beta\}. \quad (4)$$

A simple P-graph is shown on Fig. 8.

### Combinatorial properties of process networks in process synthesis

Let  $M$  be a given finite set of all material species, or materials in short, which are to be involved

Table 2. Summary of structural properties of optimal separation networks.

Product Streams	Pure		Multicomponent	
	Single	Multiple	Single	Multiple
Feed streams				
Recycling	impossible	possible	possible	possible
Redundancy	impossible	possible	possible	possible
Premixing	impossible	possible	impossible	possible
Bypassing	impossible	impossible	possible <sup>a</sup>	possible <sup>a</sup>

<sup>a</sup> Maximal bypass is not necessarily optimal.

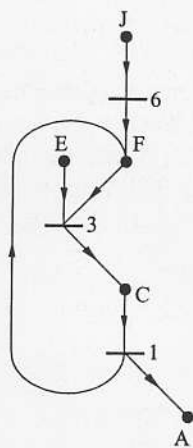


Fig. 8. Simple P-graph.

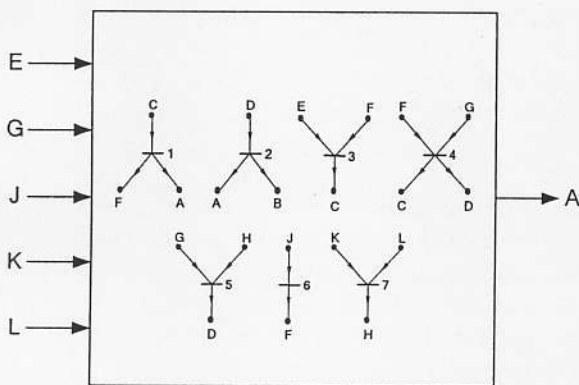


Fig. 9. Structural parts of a simple PNS problem.

in the synthesis of a process system. Suppose that a process network synthesis problem is specified by triplet  $(P,R,O)$  where  $P (\subset M)$  is the set of products to be produced;  $R (\subset M)$ , the set of available raw materials; and  $O$ , the set of operating units. If P-graph  $(m,o)$  is the network or structure of a feasible solution of PNS problem  $(P,R,O)$ , it must satisfy certain properties. These properties are stated as axioms of combinatorially feasible process networks (Friedler *et al.*, 1992a, 1998).

**Axioms of combinatorially feasible process structures**

- (S1) Every final product is represented in the structure.
- (S2) A material represented in the structure is a raw material if and only if it is not an output of any operating unit represented in the structure.
- (S3) Every operating unit represented in the structure is defined in the synthesis problem.
- (S4) Any operating unit represented in the structure has at least one path leading to a product.
- (S5) If a material belongs to the structure, it must be an input to or output from at least one operating unit represented in the structure.

**Axioms and their relations to the MINLP model**

It is usually assumed in the available literature that a process synthesis problem can appropriately be formulated as a MINLP problem. Now suppose that the following MINLP model describes rigorously a process synthesis problem.

$$\min f(x,y) \tag{5}$$

$$\text{s.t. } h(x,y) = 0, \tag{6}$$

$$g(x,y) \leq 0, \tag{7}$$

$$x \in \mathbf{R}^n, y \in \text{integer}. \tag{8}$$

Naturally, if P-graph  $(m,o)$  is the structure of a feasible solution of process network synthesis problem  $(P,R,O)$ , then the synthesized process must produce each product involved in set  $P$ . In the structure of this process, therefore, every final product must be represented; otherwise, the process is infeasible. The requirement for producing every product appears in the MINLP model as a constraint on the required amount to be produced. Consequently, every product appears in the process network, and therefore, axiom (S1) must be satisfied. Axiom (S1), therefore, is embedded implicitly in the MINLP model. This statement is valid for all the remaining axioms. Consequently, we have a collection of networks' properties of feasible solutions of a PNS problem. For example, for the PNS problem specified on Fig. 9, nineteen networks of the operating units satisfy the five axioms, *i.e.*, only these nineteen networks must be taken into account in solving the PNS problem (see Fig. 10). Another example appeared in Friedler *et al.* (1993) for synthesizing a process network from 35 plausible operating units. For this example, the number of possible networks is  $2^{35} - 1 \approx 34$  billion. The five axioms reduce this search space to 3465, *i.e.*, it is sufficient to take into account only these 3465 networks in synthesizing the process. Naturally, a question arises as to the possibility of reducing the search space further by resorting to an additional combinatorial axiom or axioms. The answer is negative: any of the 3465 networks can be optimal under certain parameters and constraints of the problem.

**Rigorous super-structure: maximal structure**

Suppose that  $S(P,R,O)$  denotes the set of all combinatorially feasible networks of PNS problem  $(P,R,O)$ . The union of all combinatorially feasible networks, *i.e.*, network

$$\mu(P,R,O) = \bigcup_{\sigma \in S(P,R,O)} \sigma \tag{9}$$

is a rigorous super-structure for PNS problem  $(P,R,O)$ ; it is called as maximal structure. It has been proved in Friedler *et al.* (1992a) that the union of

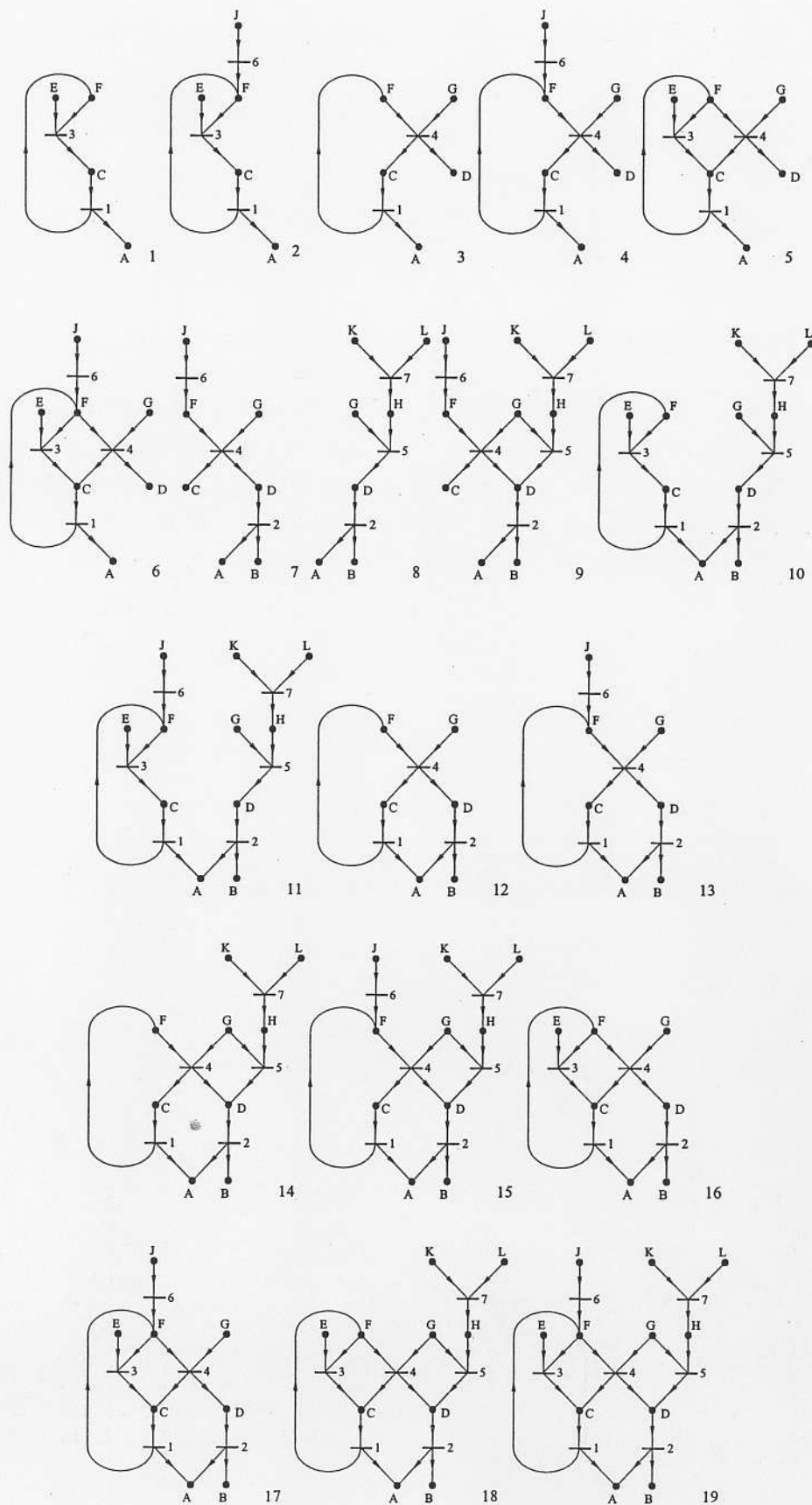


Fig. 10. Combinatorially feasible networks of the problem given in Fig. 9.



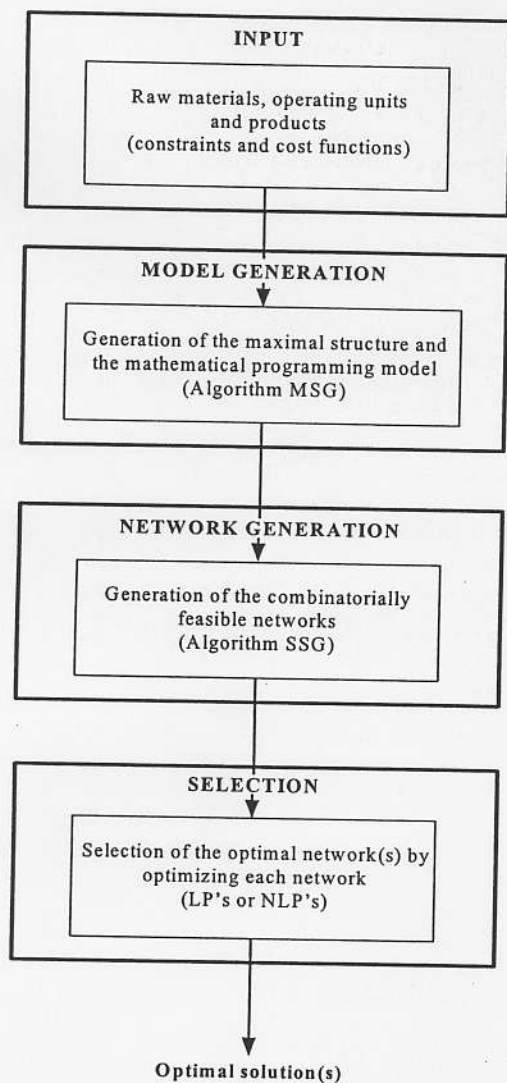


Fig. 11. Algorithmic process synthesis based on exhaustive search.

two combinatorially feasible networks is also combinatorially feasible, *i.e.*, if  $\sigma_1 \in S(P,R,O)$  and  $\sigma_2 \in S(P,R,O)$ , then  $\sigma_1 \cup \sigma_2 \in S(P,R,O)$  is also valid. Consequently, a maximal structure is also combinatorially feasible network of the problem, *i.e.*,  $\mu(P,R,O) \in S(P,R,O)$ . On the basis of this property, the maximal structure can be generated algorithmically in polynomial time (Friedler *et al.*, 1993). Algorithm SSG, has been introduced in Friedler *et al.* (1992b, 1995) for generating the elements of set  $S(P,R,O)$ , *i.e.*, the set of all combinatorially feasible networks. Algorithms MSG and SSG together is considered to be the first procedure for algorithmic process network synthesis (see Fig. 11). Even though this procedure is proved to be effective, it can be further improved by the accelerated branch-and-bound algorithm of PNS (Friedler *et al.*, 1996), *i.e.*, algorithm ABB (see Fig. 12). Further information related to P-graphs is available on home-page [www.p-graph.com](http://www.p-graph.com); also see Peters *et al.* (2003).

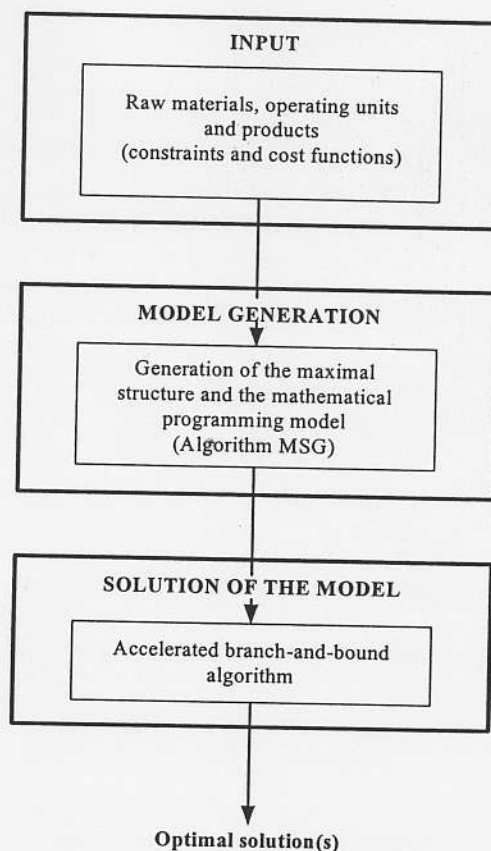


Fig. 12. Algorithmic process synthesis with the accelerated branch-and-bound algorithm.

## CONCLUSION

The difficulty of algorithmic process synthesis has been illustrated by analyzing one of the simplest classes of separation-network synthesis problems. It has been shown that the selection of the superstructure affects the quality of the solution based on the mathematical model generated from this superstructure. A brief discourse is given to indicate that the P-graph framework has been proven, in general, to offer a consistent methodology for the algorithmic process synthesis.

## ACKNOWLEDGEMENT

This contribution summarizes the unanticipated but fundamentally important results from my long-lasting collaborative work with Dr. Fan. The finding of such results is largely attributable to his unceasing commitment to the development of modern paradigms for various fields of Chemical Engineering, in general, and for process systems engineering, in particular.

## NOMENCLATURE

$A$	set of arcs of a P-graph
$b$	constant between 0 and 1
$D_i$	degree of difficulty of the separation between components $i$ and $(i+1)$
$f$	mass load through a separator
$(m,o)$	P-graph
$m, M$	set of materials
MILP	mixed-integer linear programming
MINLP	mixed-integer nonlinear programming
$o, O$	set of operating units
$P$	set of products
$(P,R,O)$	synthesis problem defined by the specific sets of products ( $P$ ), raw materials ( $R$ ), and operating units ( $O$ )
$R$	set of raw materials
$\mathfrak{R}^n$	real $n$ -dimensional vector space
SNS	separation network synthesis
$S(P,R,O)$	set of combinatorially feasible process structures for synthesis problem $(P,R,O)$
(S1), (S2), ..., (S5)	axioms of combinatorially feasible structures

## Greek symbols

$\mu(P,R,O)$	maximal structure for synthesis problem $(P,R,O)$
$\sigma$	solution-structure

## Mathematical symbols

$\emptyset$	power-set
$\times$	Cartesian product
$\{ \}$	set
$\subset$	subset or subgraph
$\cup$	union of sets or graphs

## REFERENCES

- Floudas, C. A., "Separation Synthesis of Multicomponent Feed Streams into Multicomponent Product Streams", *AIChE J.*, **33**, 540 (1987).
- Friedler, F., K. Tarjan, Y. W. Huang, and L. T. Fan, "Graph-Theoretic Approach to Process Synthesis: Axioms and Theorems," *Chem. Eng. Sci.*, **47**, 1973 (1992a).
- Friedler, F., K. Tarjan, Y. W. Huang, and L. T. Fan, "Combinatorial Algorithms for Process Synthesis," *Comput. Chem. Eng.*, **16**, S313 (1992b).
- Friedler, F., K. Tarjan, Y. W. Huang, and L. T. Fan, "Graph-Theoretic Approach to Process Synthesis: Polynomial Algorithm for Maximal Structure Generation," *Comput. Chem. Eng.*, **17**, 929 (1993).
- Friedler, F., J. B. Varga, and L. T. Fan, "Decision-Mapping: A Tool for Consistent and Complete Decisions in Process Synthesis," *Chem. Eng. Sci.*, **50**, 1755 (1995).
- Friedler, F., J. B. Varga, E. Feher, and L. T. Fan, "Combinatorially Accelerated Branch-and-Bound Method for Solving the MIP Model of Process Network Synthesis, Nonconvex Optimization and Its Applications," *State of the Art in Global Optimization, Computational Methods and Applications*, C. A. Floudas and P. M. Pardalos, Eds., p. 609, Kluwer Academic Publishers, Dordrecht, U.S.A. (1996).
- Friedler, F., L. T. Fan, and B. Imreh, "Process Network Synthesis: Problem Definition," *Networks*, **28**, 119 (1998).
- Kovacs, Z., F. Friedler, and L. T. Fan, "Recycling in a Separation Process Structure," *AIChE J.*, **39**, 1087 (1993).
- Kovacs, Z., F. Friedler, and L. T. Fan, "Parametric Study of Separation Network Synthesis: Extreme Properties of Optimal Structures," *Comput. Chem. Eng.*, **19**, S107 (1995).
- Kovacs, Z., Z. Ercsey, F. Friedler, and L. T. Fan, "Redundancy in a Separation-Network," *Hung. J. Ind. Chem.*, **26**, 213 (1998).
- Kovacs, Z., Z. Ercsey, F. Friedler, and L. T. Fan, "Separation-Network Synthesis: Global Optimum through Rigorous Super-Structure," *Comput. Chem. Eng.*, **24**, 1881 (2000).
- Peters, M. S., K. D. Timmerhaus, and R. E. West, *Plant Design and Economics for Chemical Engineers*, McGraw-Hill Chemical Engineering Series, The McGraw Hill Companies, Inc., New York, U.S.A. (2003).