Algorithmic Approaches to Process Synthesis: Logic and Global Optimization

Christodoulos A. Floudas, Ignacio E. Grossmann

EDRC 06-174-94

Algorithmic Approaches to Process Synthesis: Logic and Global Optimization

Oiristodoulos A. Floudasl and Ignacio E. Grossmann^
[1]Depertment of Chemical Engineering, PriiK^ton Univwsity, Princctoiu NJ. 08544
2 Dcpwtment of Chemical Engineering, OroegieMdlon University, Pittsburgh, PA 15213

"In memory of Professor David W. T. Rippin whose woit in Process Systems Engineering has been a source of inspiration fior us and many other researchers."

ABSTRACT

This paper presents an overview on two recent developments in optimization techniques that address previous limitations that have been experienced with algorithmic methods in process synthesis: combinatorics and local optima. The first part deals with the development of logic baaed models and techniques for discrete optimization^ich can facilitate the modelling of these problems as well as reducing the combinatorial search. It will be shewn that various levels can-1* considered for the imegnttion of logic in mixed-integer optimization techniques. The second part deals with the development of deterministic optimization methods that can rigorously determine the global optimum in nonconvex optimization models. It will be shown thai this can fee effectively accomplished with algorithms that exploit identifiable nonlinear structures. Examples are presented throughout the paper and future research directions are also briefly discussed

INTRODUCTION

Process synthesis continues to be a major area of research in process systems engineering. Significant advances have been achieved in terms of developing synthesis methods for subsystems (reactor networks, separation systems, heat exchanger networks) and for total flowsheets. Earlier reviews on general developments can be found in Hendry, Rudd and Seader (1973), Mavacek (1978) and in Nishida, Stephanopoulos and Westerberg (1981). A review on algorithmic methods based on MINLP was given by Grossmann (1990a) at the previous POCAPD meeting in Snowmass. A recent review and trends in MINLP based methods were recently presented by Grossmann and Daichendt (1994) at the PSE94 meeting in Korea. As for the synthesis of subsystems, reviews have been given by Gundersen and Naess (1988) on heat exchanger networks, and by Westerberg (1985) and Floquet, Pibouleau and Domenech (1988) on separation systems. From these reviews it is apparent that some of the major trends in the synthesis area include an increasing emphasis on the use of algorithmic methods that are based on MINLP optimization and their combination and integration with other design methodologies.

It is important to note that from a practical point of view a major motivation behind algorithmic techniques is the development of automated tools that can help design engineers to systematically explore a large number of design alternatives. From a theoretical point of view a major motivation is to develop unified representations and solution methods. Given the clear progress that has been made in the last decade in algorithmic techniques, and given the advances that have taken place in optimization and computer technology, the debate of heuristics or physical insights vs. mathematical programming has become largely irrelevant. It has generally become clear that a comprehensive approach to process synthesis will require a combination or integration of the different types of approaches. It has also become dear that significant work and progress are still required in the underlying methods that support each of these approaches. It is precisely this issue that is considered in this paper in the context of algorithmic methods.

This paper concentrates in two fundamental areas of optimization techniques that are used to support algorithmic methods in process synthesis. Specifically, we present an overview of two major advances that have recently taken place: (a) die incorporation of logic in mixed-integer optimization methods to reduce the combinatorial search and to facilitate problem formulation; (b) the development of rigorous global optimization techniques that can handle nonconvcxities in the model and avoid getting trapped in suboptimal solutions. These advances have been largely motivated by two major difficulties that have been encountered in the solution of MINLP models for process synthesis: combinatorics and local optima. The former are due to the potentially large number of structural alternatives that arise in process synthesis; the latter are due to the nonconvcxities that arise in nonlinear process models. The negative implication in the former is often the impossibility of solving large synthesis models; the negative implication of the latter is generating poor suboptimal designs.

While new developments are still under way, a review of the progress achieved up to date in logic based methods and in global optimization would seem to be timely as this might hopefully promote further research work. These algorithmic techniques are also significant in that they can be applied to other areas such as process scheduling and process analysis. The paper is organized as follows. We first discuss general aspects of process synthesis to see how the work described in this paper fits in the overall scheme. We next present a motivation section to illustrate difficulties in existing algorithmic methods with combinatorics and nonconvexities. The remaining part of the paper then concentrates in providing the overview of the new developments in logic and global optimization. Finally, we present the conclusions where we indicate future directions for research.

GENERAL COMPONENTS OF PROCESS SYNTHESIS

Algorithmic methods in process synthesis are rather general in scope and they involve the following four major components: (a) *Representation of space of alternatives;* (b) *General solution strategy;* (c) *Formulation of optimization model; (Si Application of solution method.*

The representations can range from rather high level abstractions such as is the case of targeting methods, to relatively detailed flowsheet descriptions such as is the case of superstructure representations. It is important to note that these representations are in fact commonly closely related as their difference lies in the level of abstraction that is used.

Having developed a representation, the next step to consider is the general solution strategy. The two common and extreme solution strategies are the simultaneous and the sequential approaches. The simultaneous strategies attempt to optimize simultaneously all the components in a synthesis problem in order to properly capture all the interactions and economic trade-offs. While conceptually superior, these strategies may give rise to larger problems. The sequential approach on the other hand has the advantage of dealing with smaller subproblcms since they sequentially decompose the problem, although often at the expense of sacrificing optimality.

The nature of die optimization models is of course heavily dependent on the type of representation as well as on the general solution strategy being used Target models often involve only continuous variables since they usually do not generate topologies nor do they consider capital cost as they deal with higher level objectives (minimize utility consumption, maximize yield). Therefore, these models commonly give rise to linear (LP) or nonlinear programming (NLP) problems. At the other extreme superstructure models determine topologies and operating conditions, and account for capita! costs, often requiring 0-1 and continuous variabte giving rise to nixed-intcgerUnear (MILP) or mixed-integer nonlinear (MINLP) optimization models. Wiftin each of the levels of itfxeseittationtte degree of rigorousness of the model can of course also range from the simpler short-cot models to detailed simulation models.

As for the solution methods a global optimum solution can be guaranteed if the problem can be posed as an LP or MILP problem. Furthermore, in the case of LP models efficient solution times can be expected since these problems are theoretically solvable in polynomial time. This is however not the case of the MILP problems which generally are NP-complete, and therefore may have exponential time requirements, at least in the worst case. If the problem is posed as an NLP or MINLP the first drawback is that a unique global solution can only be guaranteed if the NLP or the continuous relaxation of the MINLP are convex. This is of course only a sufficient condition. But nevertheless, nonconvexities are prevalent in synthesis problems, often giving rise to multiple local solutions, or in fact even preventing convergence to feasible solutions with conventional NLP techniques. In addition to the numerical and theoretical difficulties of handling nonconvex models, there is the added difficulty of potential combinatorial explosion for the MINLP case. In the context of process synthesis a good example of the dilemma between the use of MDLP and MINLP models are the approaches for superstructure optimization of flowsheets by Papoulias and Grossmann (1983) and by Kocis and Grossmarm (1989). The advantage of the former is that the global optimum can be guaranteed but at the expense of using a discretized and approximate process model. The advantage of the latter is that nonlinear process models can be explicitly handled, but with the disadvantage that the global optimum cannot be guaranteed.

Based on the above discussion, it is clear that in order to properly support the development of algorithmic techniques, whether for targeting or superstructure models, or for simultaneous or sequential approaches, it is imperative that limitations due to combinatorics and nonconvexities be addressed. It is in this context that the two motivating examples below are presented.

MOTIVATING EXAMPLES

MILP Model for Heat Integrated Distillation Sequences

In order to illustrate potential combinatorial difficulties with synthesis problems, consider the MILP model reported in Raman and Grossmann (1993a) in which hem integration is considered between different separation tasks in the synthesis of sharp distillation sequences (see also Andrecovich and Westerberg (1985) and Floudas and Paulcs (1988)). An example of a superstructure for 4 components is given in Fig. 1 . For the heat integration part, it is assumed that the pressures of the columns can be adjusted in such a way that the condenser of every column can potentiaUy supply heat to the reboilers of the other columns as shown in Fig. 2 (multieffect columns are not considered). The MILP model involves as 0-1 variables the potential existence of columns and the potential beat exchanges between columns and reboilers, and as continuous variables the flows, heat loads and temperatures of condensers and reboilers, with which pressure changes arc accounted for. The objective function consists of die minimization of the investment cost of the columns and the operating cost for the utilities. The constraints involve mass and heat balances, and logical constraints that enferce feasible temperatures if heat exchange take pla^ and zero flows and heat loads if the corresponding 0-1 variables are 9et to zero.

For a four component system such as the one in Fig. 1 the MILP model involves 100 0-1 variables, 191 continuous variables and 258 constraints. The 100 binary variables are split as follows -10 to model the existence of the distillation columns and 90 to model the existence of heat exchange matches between the reboilers and condensers of the various columns. The computer codes ZOOM, OSL and SCICONIC were tried for solving this problem. The three of them were not able to even find a feasible solution after enumeiating mem than 100,000 nodes and after running more than 1 CPU hour on an IBM RISC/6000! A major reason for this performance was that the relaxation gap is very large in this problem; the LP relaxation in which the binary variables are treated as continuous the optimum is Sl,117,000/yr. while the optimal MILP solution is $l,900,000/yr. As will be shown later in the paper, by using logic rigorous optimization of this problem can be achieved in only few seconds!

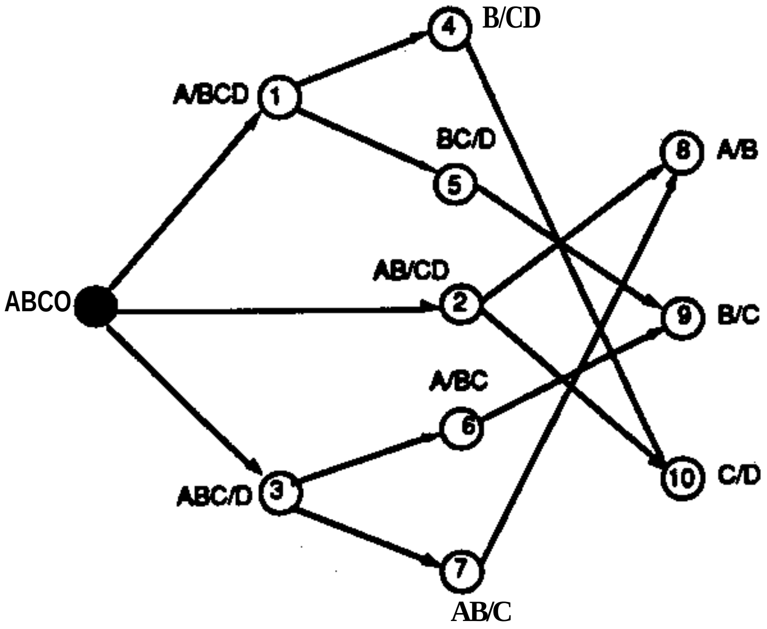


Fig. 1. Superstructure for 4-compoDent example.

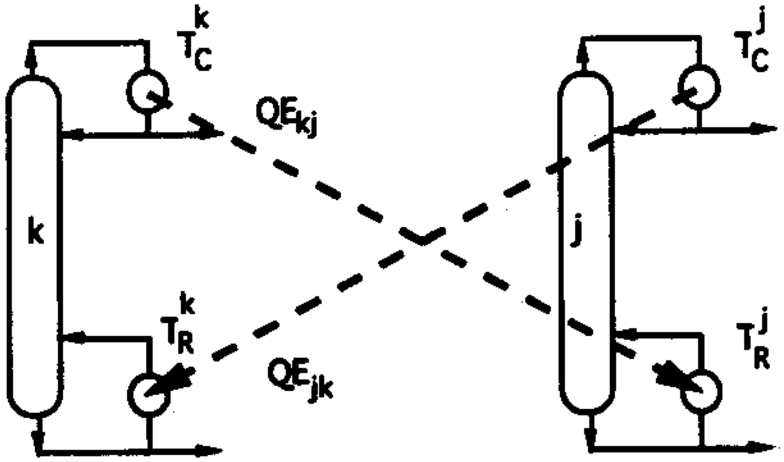


Fig. 2. **Heat integration between different separation tasks.**

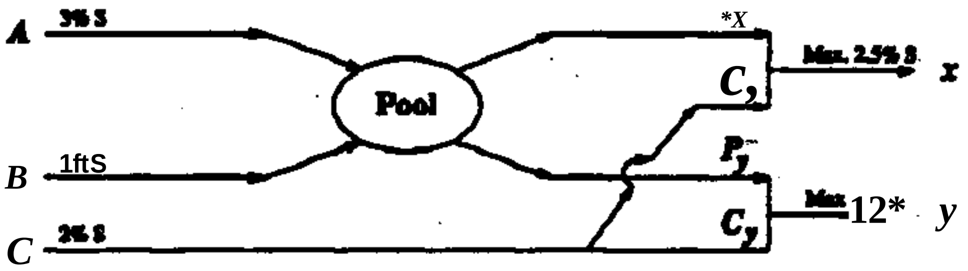
Nonconvex Model for Pooling/Blending Problems

To illustrate the potential difficulties associated with the existence of multiple solutions in nonlinear optimization NLP problems, we will consider as motivating example the pooling problem proposed by Haverly (1978) which is shown in Figure 3. Three crudes A, B, **and** G with different sulfur contents are to be combined to form two products x and y which have specifications on the maximum sulfur content Note that streams A and B are mixed in a pool and it is the existence of such a pool that introduces non-convexities in the mathematical model in the form of bilinear terms between the sulfur quality of the streams exiting the pool, denoted as p, and flowrates P_x, Py of the pool exiting streams. The objective in this pooling problem is to maximize the profit subject to (i) linear overall and component balances, (ii) bilinear pool quality and product quality constraints, and (iii) bounds on the products and sulfur quality. This problem has been studied using several local nonlinear optimization algorithms which have been reported to either obtain suboptimal solutions or fail to obtain even a feasible solution (see Floudas and Aggarwal, 1990 for a review of previous approaches and a decomposition strategy which alleviates but does not eliminate the multiplicity of local solutions problem). Table 1 presents results of local optimization algorithms (e.g. MINOS) for several starting points.

Table 1: Local Optimization for **the** Pooling Problem

		Solution Found	
No.	Sianins Quality	Objective value	Oualitv P
1	1.00	-750.0000	1.50
2	1.25	-750.0000	1.50
3	1.50	-750.0000	1.50
4	1.75	0.0000	1.75
5	2.00	0.0000	2.00
6	2.25	-125.0000	2.50
7	2.50	-125.0000	2.50
8	2.75	-125.0000	2.50
9	3.00	-125.0000	2.50

Figure 3: Motivating Example (Pooling Problem)



Formulation

rnln &4 + 13B + 1O(C. + *C,)* -9*x* - I5y

8.t.

$P_x + P_y - A - B = 0$ } pooffcofonce

z-F$_m$-P.-C. - 0. 1 *componentbalance*

V

p.(P> + P$_y$) - *SA* - *B m* 0 } ***pool quality***

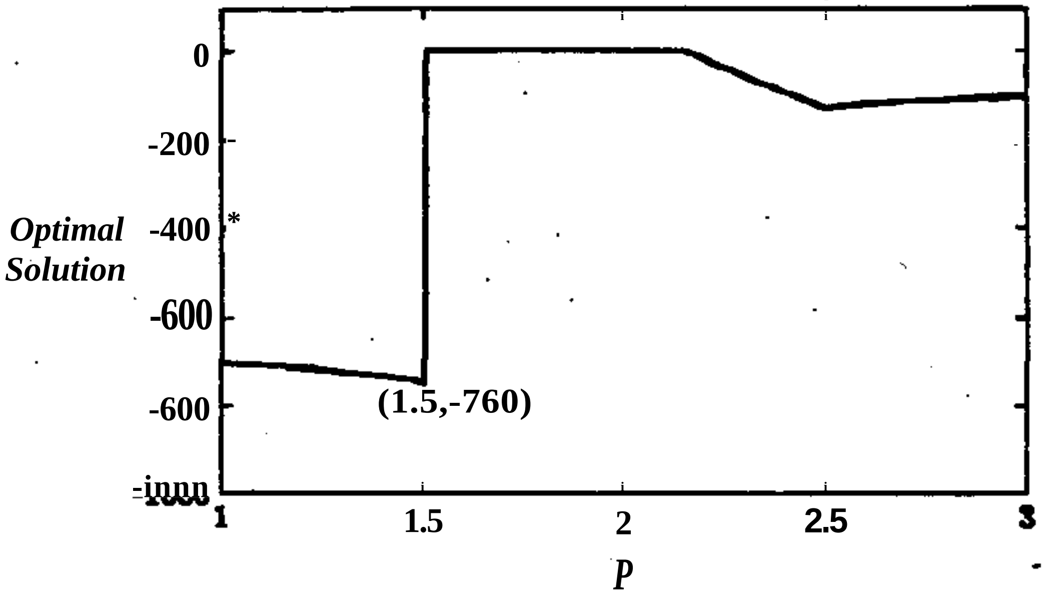
$p.P_x + 2.C_x - 2.5x \leq 0$
$p.P_y + 2.C_y - 1.5y \leq 0$ } *product quality con&traini**

m $\leq$ 100
y $\leq$ 200 } *Upper bounds an products*

1 $\leq$ P $\leq$ 3 I *Bound* on sulfur quality*

6

Figure 4; Optimal Solution in Projected Space



The non-convex nature of this pooling problem is better illustrated via Figure 4 where the optimal solution of the pooling model is shown for different values of the of the pool quality p. Note that the global optimum occurs at p « 1.5, while there exists a local optimum at p * 2.5 and between p • 1.S and p =2.2 (approximately) the optimal solutions are of the form of constant line. As a result, several starting points for p in the flat region or the region close to the local optimum terminate with the local solution or even fail to obtain a solution.

Floudas and Visweswaran (1990) applied the decomposition global optimization approach GOP, which is discussed in the global optimization section, to this pooling problem, as well as large instances of other pooling problems and multiperiod tankage problems (see also Visweswaran and Floudas, 1993) where the global optimum is obtained regardless of the starting point

INTEGRATION OF LOGIC IN MIXED-INTEGER PROGRAMMING

In this section we present a brief review of previous work on the modelling and solution techniques of logic based discrete optimization. We also review basic concepts for the representation of logic and inference problems. We then describe our recent work at Carnegie Mellon on the integration of logic in mixed-integer optimization which has been primarily motivated by process synthesis problems.

Review of Previous Work

A major issue in the application of mixed-integer programming is the efficient modelling of discrete decisions. Different representations are often possible for the same model, each of which may be solvable with varying degrees of difficulty. In some cases it is possible to even formulate an MH-P problem so that it is solvable as an LP, or else, so that its relaxation gap is greatly reduced. While some basic understanding has been achieved on how to properly formulaic special classes of mixed-integer programs (see Rardin and Choe, 1979; Nemhauser and Wolsey, 1988), the modelling of general purpose problems is largely performed *on an ad hoc* basis. The use of prepositional logic, however, offers an alternate framework for systematically developing mixed-integer optimization models as discussed by Jeroslow and Lowe (1984) and by Williams (1988).

The role of logic at the level of modelling of discrete optimization problems has also been studied by Balas (1974, 198S) who developed Disjunctive Programming (DP) as an alternate representation of mixed-integer programming problems. In this case, discrete optimization problems are formulated as linear programs in which a subset of constraints is expressed through disjunctions (sets of constraints of which at least one must be true). An interesting feature in the disjunctive formulation is that no 0-1 variables are explicitly included in the model, which is the more natural form to model some problems as, for instance, in the case of jobshop scheduling problems. Also, as noted by Balas (198S), every mixed-integer problem can be reformulated as a disjunctive program, and every bounded DP can be refonnulated as a mixed-integer program. The reason the disjunctive programming formulation has not been used more extensively is that very few methods have been proposed to explicitly solve the problem in that form. Most of die research has focused on characterizing the convex hull of disjunctive constraints and on the generation of strong cutting planes which are included in the corresponding mixed-integer problem to strengthen the LP relaxation (Balas, 1985; Jcroslow and Lowe, 1984). The only reported method, to our knowledge, that explicitly solves problem is the algorithm by Beaumont (1991) for the case where the functions are linear and there is only one constraint in each term of every disjunction. The method is similar to a branch and bound search except that the benching is done directly on the disjunctions. This requires the addition and deletion of the correspoiHiingdis^tive constraints in the LP subproblems. Although this may increase the overhead in the computations, Beaumom showed that the number of note required for the enumeration of the branch and bound tree can often be significantly reduced.

In terms of integrating logic explicitly for improving the solution efficiency of mixed-integer programs, aside from our own work which will be described in the next section (Raman and Grossmann, 1991,1992,1993a*, 1994), Lien and Whale (1991) considered the addition of a subset of unit resolution cuts for the branch and bound solution of MILP problems which produced large reductions of enumeration of nodes in the MILP formulation for heat integrated synthesis by Andrecovkh and Westerberg (1985). It should also be mentioned that logic has been considered earlier in process synthesis with the purpose of performing high level decisions in the structuring of process flowsheets (Mahalec and Motard, 1977).

Representations of the logic

Most of the work described above has been restricted to the form of logic calkd prepositional logic for developing modelling and solution techniques for discrete optimization problems (see Menddson, 1987, for general review on logic). The basic imit of a propositkmallogte express*^ which can correspond to a state or to an action, is called a literal which is a single variable that can assume either of two values, true or false. Associated with each literal J\ its negation NOT *Y* (-. *Y)* is such that *[Y* OR -ill is always true. A disjunctive clause is a set of literals separated by OR operators [v], and is also called a disjunction. A proposition is any logical expression and consists of a set of clauses />;, i»l,../ that are related by the logical operators OR [v], AND [A], IMPLICATION *[=>].*

In synthesis logic propositions usually refer to relations of existence of units in a superstructure. These are commonly expressed by a set of conjunctions of clauses,

Am{LjAL2A...... ALg} (1)

where Li is a logical proposition expressed with boolean variables Kj in terms of implications, OR, EXCLUSIVE OR and AND operators. In synthesis problems *Y* is a boolean variable representing the existence of unit i and -IYJ its nonexistence. There are two ways of transforming the propositions in A. In the simplest case, the logic propositions are converted into the conjunctive normal fonn [CNF] by removing the implications through contrapositions in each of the clauses Lj in (1) and applying De Morgans

Theorem. In this way each clause in the CNF from consists of only OR operators with non-negated and negated boolean variables as follows:

Ω_C = [$\underset{i \in P_1}{\vee}$ 0!) $\underset{i \in}{\vee}$ W] A [v ft) v <-r,)]A...*r v (R) A <-r,)] (2)

where Pi and $\overline{\text{P}}$ i are subsets of the boolean variables that correspond to some of the 0-1 variables, and s is the number of clauses.

In the second representation, the logic propositions in the CNF form are convened into the disjunctive normal form [DNF] (see Clocksin and Nfellish, 1984) by moving the AN^ operators inwards and the OR operators outwards by applying elementary boolean operations. The DNF form is as follows:

^{U}D [i*Ql left \ [$^{i€}$Ql ieGl J l'«ft> '«5 J W

where Qj and $\overline{\text{Qj}}$ are the index sets of the boolean variables which correspond to a partition of all the 0-1 variables y,\ i=7,../> in non-negated and negated terms. Each clause separated by a disjunction represents the assignment of units in a feasible configuration* where it is assumed that each boolean variable has a one-to-one correspondence with the 0-1 binary variables of the MEP model. Therefore, r represents the number of alternatives in the superstructure. While the DNF form is more convenient to manipulate, the drawback is that the transformation from CNF to DNF has exponential complexity in the worst case.

To illustrate the CNF and DNF representations in (2) and (3), consider the small example problem shown in Fig.5. The following prepositional logic expressions apply:

I4: *Y* v *Y2* => 1'3 (process l or process 2 imply process 3)

^{L}2: 1^3 =* ^{y}l ^{v} ^{Y}*2* (process 3 implies process 1 or process 2)

L3: -*tY* v -.^2 (do not select process l or do not select process 2)

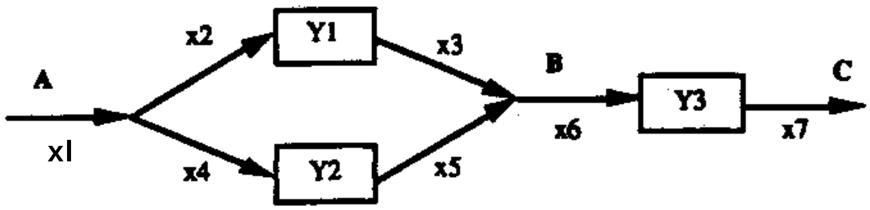


Fig 5. Superstructure for small example.

Applying the contrapositive to Li and L2, and using De Morgan's theorem, the corresponding CNF representation1 for the logic is:

Ω_C = *Hi* v *Y3)* A (-.72 v K3) A (-^3 v *Y* v *YD* A (-.ri v ¬*Y*2) (4)

Distributing the OR over the AND operators, the corresponding DNF representation is given by:

OD * <*T* A -^2 A *Y^)* V (72 A -iFl A *Y3)* V *(-JTl* A -n72 A -1X3) (5)

Note that the disjunctions in (5) represent the three alternatives in Fig 5.

In order to obtain an equivalent mathematical representation for any prepositional logic expression, this can be easily performed using the CNF form as a basis. We must first consider basic logical operators to determine how each can be transformed into an equivalent representation in the form of an equation or inequality. These transformations are then used to convert general logical expressions into an equivalent mathematical representation (Cavalier and Soyster, 1987; Cavalier***al*, 1990).

Table 2. Representation of logical relations with linear inequalities

Logical Relation	Comments	Boolean Expression	Representation as Linear Inequalities
Logical OR		Pi v P2 v .. v Pr	yi + y2 + •• • yr * 1
Logical AND		Pi A P2 A .. A Pr	$y_1 \geq 1$ $y_2 \geq 1$ .. $y_r \geq 1$
Implication	$P_1 \Rightarrow P_2$	-.P1VP2	1 - yl + y2 * 1
Equivalence	Pi if and only if P2 (Pl=*P2)A(P2=>Pl)	(-Pi VP2)A(-,P2v Pl)	yi«y2
Exclusive OR	exactly one of the variables is true	Pi ⊻ P2 ⊻ .. ⊻ Pr	$y_1 + y_2 + .. + y_r = 1$

To each literal P_v a Unary variable *y* is assigned. Then the negation or complement of/^ (-.Pj) is given by 1 - y_x. The logical value of true corresponds to the binary value of 1 and false corresponds to the binary value of 0. The basic operators used in prepositional logic and the representation of their relationships are shown in Table 2. With the basic equivalent relations given in Table 2 (e.g. see William's, 198S), one can systematically model an arbitrary prepositional logic expression that is given in terms of OR, AND, IMPLICATION operators, as a set of linear equality and inequality constraints. One approach is to systematically convert the logical expression into its equtvalent *conjunctive normal form* representation which involves the application of pure logical operations. The conjunctive normal form is a conjunction of clauses, *Q* A C2 A ... A g_s. Hence, for the conjunctive normal form to be true, each clause *Q* must be true independent of the others. Also since a clause *Q* is just a disjunction of literals, Pjv />2 v ... v P_r, it can be expressed in the linear mathematical form as the inequality.

37 • >2 +. . . . .• *yr* * *1* (6)

Symbolic **and Mathematical** Methods for Logic Inference

The most common logic inference problem is the satisfiability problem where, given the validity of a set of propositions, one has to prove the truth or validity of a conclusion which may be either a literal or a proposition. This inference problem is one of the basic issues in artificial intelligence and data bases. However, the general satisfiability problem for prepositional logic is NP-complete (Cook, 1971; Karp, 1972). Therefore, research has focused on identifying classes of problems within the general satisfiability problem that can be solved efficiently. Knowledge based systems commonly require the use of Horn clause systems which have at most one non-negated literal in each clause. The inference problem for this class of prepositional logic problems can be solved in linear time using unit resolution (Dowling and Gallier, 1984). The unit resolution technique (e.g. see Clocksin and Mellish, 1981) is one of the most common inference techniques, and in simple terms, it consists of solving sequentially each logic clause one at a time. Chandra and Hooker (1988) have extended the class of problems that can be solved in linear time to include extended Horn clause systems. One of the most effective logic-based methods for solving the general satisfiability problem is the algorithm of Davis and Putnam (1960) as treated by Loveland (1978). This approach is closely related to the branch and bound method for mixed-integer programming. Jereslow and Wang (1990) have developed branching heuristics to improve the performance of the Davis-Putnam procedure. It must be noted that although the previous work has been restricted to prepositional logic, the techniques used for this class are essential to higher order representations like predicate logic which involve additional logic operators like for all [V] and it exists **[3].**

Since the logical propositions can be systematically converted into a set of linear inequalities, instead of using symbolic inference techniques, the inference problem can be formulated as an integer linear programming problem. In particular, given a problem in which all the logical propositions have been converted to a set of linear inequalities, the inference problem that consists of proving a given clause,

$$\begin{array}{lll} \text{Prove} & P_u & \qquad \text{(UP1)} \\ \text{st} & B(P0 \quad i = U,..,q & \end{array}$$

can be formulated as the following MILP (Cavalier and Soyster, 1987):

$$\begin{array}{lll} \textbf{Min} & Z« \underset{i \in I(u)}{X} \alpha_i y_i & \\ & & \\ \text{st} & A y \;\; 2 \;\; a & \qquad \text{(UP2)} \\ & y \;\; € \;\; \{0J\}^n & \end{array}$$

where $Ay \;\; k$ a is die set of inequalities obtained by translating B $(P\backslash.P2*$ •• fq) into their linear mathematical form, and the objective function is obtained by also converting the clause P_u that is to be proved into its equivalent mathematical form. Here, /(u) corresponds to the index set of the binary variables associated with the clause P_u. This clause is always true if Z * 1 on minimizing the objective function as an integer programming problem. If Z = 0 for the optimal integer solution, this establishes an instance where the clause is false. Therefore, in this case, the clause is not always true. In many instances, the optimal integer solution to problem (LIP2) will be obtained by solving its linear programming relaxation (Hooker, 1988). Even if no integer solution is obtained, it may be possible to fetch conclusions from the relaxed UP problem (Cavalier and Soyster, 1987).

The qualitative knowledge available about the design of a system can be classified as one of the following two types - hard logical bets or uncertain heuristics. Hard, logical facts are never violated - for example, the reaction NaOH + HCl -~> N a C U H ^ holds from basic (Conical principles. Qualitative knowledge in the form of heuristics on the other hand are just rules of thumb which may not always hold. Therefore all the knowledge for synthesizing a design may not be consistent since the heuristics may contradict one another; for example, a rule that suggests to use higher temperatures to increase yield may conflict with a rule that suggests to use lower temperature to increase selectivity. Resolution of conflicts is an important pan of reasoning. In general one must violate a weaker (more uncertain) set of rules in order to satisfy stronger ones. Therefore, it becomes necessary to model the violation of heuristics, which is done as follows (Post, 1987),

$$\textit{Clause} \quad \textit{or} \quad V \qquad (7)$$

where either the clause is true or it is being violated (V). In order to discriminate between weak and strong rules, penalties are associated with the violation v| of each heuristic rule, i * U.jn. The penalty wj is a non-negative number which reflects the uncertainty of the corresponding logical expression. The more uncertain the rule, the lower the penalty for its violation. In this way, the logical inference problem with uncertain knowledge can be formulated as an MELP problem where the objective is to obtain a solution that satisfies all the logical relationships (i.e. Z * 0), and if that is not possible, to obtain a solution with the least total penalty for violation of the heuristics:

$$\begin{array}{llll} Min & Z \ll w^T v & & \\ st & Ay \quad Z \quad a & : \text{Logical facts} & \text{(UP3)} \\ & By \bullet v \; 2 \; b & : \text{Heuristics} & \\ & y \in \{0,l\}^n. \; v \pounds 0 & & \end{array}$$

Note that no violations are assigned to the inequalities Ay 2B since these correspond to hard logical facts that always have to be satisfied. The solution to (UP3) will then determine a design that best satisfies the possibly conflicting qualitative knowledge about the system.

Logic-based Formulations for Discrete Optimization

Given a superstructure of alternatives for a given design problem, the general form of the mixed-integer optimization model is (Grossmann, 1990a),

$$\begin{array}{lll} Min & ZmJy+jlx) & \\ \text{st} & k(x) \quad Z \; 0 & \\ & g(x) + My \; \pounds \; 0 & \text{(DPI)} \\ & xe \quad X.ye \quad Y & \end{array}$$

where *x* is the vector of continuous variables involved in design like pressure, temperature and flow rates, while *y* is the vector of binary decision variables like existence of a particular stream or unit Integer variables might also be involved but these are often expressed in terms of 0-1 variables. Also, model (DPI) may contain among the inequalities pure integer constraints for logical specifications (e.g. select only one reactor type). If all the functions and constraints are linear (PI) correspoiKis to afrMIlP problem; otherwise it is an MINLP. For the sake of simplicity, we assume that/frj, *g(x)* and *h(x)* are convex, differentiate functions. The case of nonconvexities will be addressed later in the paper.

The mixed-integer program (DPI), is not the only way of modelling the discrete optimization problem in a superstructure. As has been shown by Raman and Grossmann (1994) that problem can be formulated as the generalized disjunctive program:

$$Min \quad Z = \sum_{i} \sum_{k} c_{ik} + f(x)$$

$$st \quad h(x) \text{ £ } 0 \qquad\qquad (DP2)$$

$$\bigvee_{i \in D_k} \begin{bmatrix} Y_{ik} \\ g_{ik}(x) \le 0 \\ c_{ik} = \gamma_{ik} \end{bmatrix} \qquad ke\ SD$$

$$\Omega(Y) = True$$

$$x€R^n{}_9ceR^m,Y€ \quad \{true,false\}"^1$$

in which *Yfr* are the boolean variables that establish whether a given term in a disjunction is true *lgUc(x)* ≤ *0]* or false *[gUcfx)* > 0], while *Cl(Y)* are logical relations assumed to be in the form of prepositional logic involving only the boolean variables. *Yfc* are auxiliary variables that control the part of the feasible space in which the continuous variables, x, lie, and he variables c;* represent fixed charges which are activated to a value fik if the corresponding term of the disjunction is true. Finally, the logical conditions, *£l(Y),* express relationships between the disjunctive sets. In the context of synthesis problems the disjunctions in (DP2) typically arise for each unit i in the following form:

$$\begin{bmatrix} Y_i \\ g_i(x) \le 0 \\ c_i = \gamma_i \end{bmatrix} \vee \begin{bmatrix} \neg Y_i \\ B^i x = 0 \\ c_i = 0 \end{bmatrix} \qquad (8)$$

in which the inequalities gi apply and a fixed cost n is incurred if the unit is selected (?;£ otherwise *(~*Y0* there is no fixed cost and a subset of the x variables is set to zero with the matrix B*. An important advantage of the above modelling framework is that there is no need to introduce artificial parameters for the "big-M" constraints that are normally used to model disjunctions.

An interesting question that arises with problem (DP2) is whether it always pays to convert the general disjunctive program into mixed-integer form. To answer this question for die case of linear functions and constraints, Raman and Grossmann (1994) have developed the concept of w-MIP represcntability which is defined asfollows:

Definition: The disjunction $\underset{ieD_k}{\mathrm{V}} \quad \backslash Aik^{x}\wedge^{b}id$ is ***w-MIP representable*** iff the following conditions hold:

(i) There exists an i€ Dk for which the convex bull of the disjunction is reducible to the constraint:

$$A_{ik}x \;\; 2\,btkytk \qquad\qquad 0\ £\ ytk \leq 1$$

(ii) Every feasible solution

$$x'€F\text{-}\{xl \quad \underset{ieD_k}{\mathrm{V}} \quad [A_{ik}x \geq b_i\}$$

for which $Afkx^{*}\ 2$fcflfc,$Afcx'$ < &#, i * i implies that yft « 7 *andyik = 0 Vi *i*

Thus, in general, we can consider a partly transformed form of problem (DP2) where mixed-integer equations are used for the w-MIP constraints pan of the problem, while the rest is kept in disjunctive form, as this pan is "poorly-behaved" in equation form. In general, this partially reformulated problem has the form,

$$\begin{aligned}
&Min \quad Z \ll \sum_{k\in SD^1}\sum_{i\in D_k}\gamma_{ik}y_{ik} + \sum_{k\in SD^2}\sum_{i\in D_k}c_{ik} + f(x)\\
&st \quad h(x) \leq 0 \qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad \text{(DP3)}\\
&\qquad H^{*}) + By\ SO\\
&\qquad Ay\ k\ \ a\\
&\qquad \underset{ieD_k}{\mathrm{V}} \begin{bmatrix} Y_{ik} \\ s_{ik}(x) \leq 0 \\ c_{ik} = \gamma_{ik} \end{bmatrix} \qquad\qquad ke\ SD^2\\
&\qquad A(Y) \ll True\\
&\qquad xelP.ye \ \ (Ojf.\ \ Y \in \ \{true, false\}''^{1}
\end{aligned}$$

in which the subset of disjunctions SD^1 c SD, which are w-MIP representable, have all been convened into mixed-integer form. The inequalities r(x) + By £ 0 correspond to these constraints and to subsets of the inequalities gikfc cik) ^ 0, i € Dk, k € SD^2, which have also been converted into mixed-integer form. Finally, s& (x, cik) are the remaining inequalities which appear explicitly in the disjunctions k € SD^2.

Note also that a subset of the logical constraints in *Q(Y)* • True, which are required for the definition of the discrete optimization problem, have been translated to the form of linear inequality constraints Ay £ a. The simplest option is to convert the propositions into CNF which can then be translated readily into inequalities as was discussed in the previous section. In cases where the number of these constraints become large, the generation of a smaller number of tighter constraints through the application of cutting plane techniques may be useful. The rest of the logic constraints, A(K) « True, which are redundant and correspond to logic cuts that do not alter the optimal solution (Hooker et al, 1993), have been left in symbolic form in order to improve the enumeration in a branch and bound search.

It should be noted that a particular case of (DP3) of interest is when the entire problem is converted into mixed-integer form, but the logic cuts *A(Y)« True* are included as part of the formulation:

$$
\begin{array}{lll}
Min & Z= \overset{m}{\underset{i=1}{\wedge}} YiVi \ +/(*) & \\
st & h(x)\leq 0 & (DP4) \\
 & Hx) + By * 0 & \\
 & Ay \ 2 \ a & \\
 & A(Y) = True & \\
 & x€R^{n}{}_{9}ye \quad \{0.1\}^{m}. \ \ Y \ \ e \quad \{true, \ false\}^{9}" &
\end{array}
$$

Solution methods

As was mentioned in the review section there are still few methods for solving mixed-integer optimization problems that incorporate prepositional logic. As shown below, methods have been developed for addressing linear and nonlinear problems. Obviously some of the methods are equally applicable to both cases. However, for the sake of clarity, and to also emphasize the more useful methods in each case, we will distinguish between methods for linear and nonlinear problems.

For linear problems the simplest case is when logic cuts A(K) • True are added to an MDLP problem as in (DP4). These cuts, which represent redundant constraints in high level form, can be systematically generated for process networks as discussed in Raman and Grossmann (1993a). As an example, the logic cuts for the network in Fig. 1 in terms of the potential existence of the 10 columns are given by the propositions:

Yl = Y4v Y5*

Y2=>Y8AY10

Y3 =>Y6v Y7

Y4 =>Y1 A Y10

Y5=>Y1AY9

Y6=>Y3A Y9

Y7 = Y3 A Y8*

Y8=>Y2 vY7

*Y9=*Y5 v Y6*

*Y10=*Y2VY4*

There arc two bask ways of handling these cuts. One is to convert them into inequalities and add them to the MDLP (Raman and Grossmann, 1992). White this will increase the number of constraints, it generally reduces the relaxation gap. The other extreme is to process the logic symbolically as part of the branch and bound search for the MOP. In this case the logic is used to select the branching variables and to determine by inference whether additional Unary variables can be fixed at each node (Raman and Grossmann, 1993a,b). This can be accomplished by treating the logic either in CNF form as in (2) or in DNF form as in (3). The former requires unit resolution for the inference, whik the latter involves the solution of Boolean equations. Although the DNF form is generally more expensive to obtain, its nice theoretical property is that one con *guarantee that in the worst case the number of enumerated nodes does not exceed twice the number of clauses in (3) minus one* (see Raman and Grossmann (1993a) for proof). A third alternative is to use a hybrid approach in which only violated inequalities at the root node are included to strengthen the LP relaxation, but the remaining enumeration is performed by solving the logic symbolically.

For the case that the discrete optimization problem is formulated as in (DP3) by involving both disjunctions and mixed-integer constraints, Raman and Grossmann(1994) proposed an extension of the hybrid branch and bound method for (DP4) in which the disjunctions are converted for convenience into mixed-integer form, but the branching rule is altered to recognize the fact that no branching be performed on disjunctions that are logically satisfied, even if the corresponding 0-1 variables are non-integer. Note that such an algorithm can also be applied to problem (DP2). Finally, it is worth to mention that Beaumont (1991) has proposed an algorithm that applies to (DP2) in the case that only one equation is involved in each disjunction. In this algorithm constraints are successively added or deleted as needed in the branch and bound search.

Similarly as in the linear case, the simplest way to integrate logic in nonlinear discrete models is to add the logic cuts to an MINLP as in problem (DP4) (see Raman and Grossmann, 1992). If these are convened to inequalities this has the effect of reducing the relaxation gap. This has the important effect of strengthening the lower bound that is predicted by the master problem in the Generalized Benders decomposition method by Geoffrion (1972). As has been shown by Sahinidis and Grossmann (1991) the "optimal" formulation for the GBD method is when there is no gap between the relaxed and the integer optimum solution. In the case of the outer-approximation method by Duran and Grossmann (1986) the quantitative or symbolic integration has the effect of potentially reducing the branch and bound enumeration at the level of the MILP master problem. An interesting variation of the above idea is to integrate the logic inference problem with heuristics (UP3) in the MILP master problem as was proposed by Raman and Grossmann (1992). First assume that given the solution of K NLP subproUems the MILP master problem is represented by:

$$
\begin{array}{ll}
Min & a \\
st\ a\ £\ 4fa\ j) & \\
xj\ \wedge\ \wedge & *=\ 1\text{-}JT \qquad (Ml) \\
x€jr,y€y &
\end{array}
$$

in which ^*x,y)* represents either the Lagrangian in the GBD method or an objective linearization in the OA method, £2* is the linear approximation to the continuous feasible space and INTfc represents integer cuts to exclude configurations that were previously analyzed. The integer programming model (LIP3) can be integrated in the above master problem(Ml) by minimizing the weighted violation (plus an extra term to reflect the cost) and subject to constraining the lower bound to the current upper bound; that is,

$$
\begin{array}{ll}
Min\ [w^T v + \bar{w}(a-LB)/(UB^*-LB)] & \\
st\ \alpha \geq \phi_k(x,y) & k= 1,...,K \\
x,y \in \Omega_k & \\
y\ €\ INT_k & \\
Ay \qquad 2\ a & \\
By\ +\ v \qquad Zb & \qquad (M2) \\
a\ £\ UB^k & \\
xeX.yeY & \\
ae\ *\backslash\ \ v\ €\{0,1\} &
\end{array}
$$

in which $\bar{w}$ is a penalty chosen such that $\bar{w}$ « min* *(w%)* f LB is a valid lower bound to the solution of the MINLP (e.g.. solution to the relaxed NLP problem or some reasonable but valid bound) and UB^K is the current upper bound of the objective at iteration K. The interesting feature with the master problem (M2) is that optimality can still be guaranteed (within convexity assumptions) even though heuristics are used as part of the search. The master problem (M2) is especially appropriate for the GBD method because of the loose approximation that is obtained with that method. It is also important to note that the master problem (M2) can be used when applying Benders decomposition (Benders, 1962) in the solution of MILP problems.

For the case that the nonlinear discrete optimization problem is formulated as the generalized disjunctive program in (DP2) one can develop corresponding logic-based OA and GBD algorithms as described in Turkay and Grossmaim (1994). First, for fixed values of the boolean variables, Yfk * true and Y_{ik} = false, the corresponding NLP subproblem is as follows:

$$\text{Min} \quad Z = \sum_{i=1}^{m} c_{ik} + f(x)$$

$$\text{st} \quad h(x) \le 0 \qquad \text{(SP)}$$

$$\begin{Bmatrix} c_{ik}=\gamma_{ik} \\ g_{ik}(x)\le 0 \end{Bmatrix} \text{\forYv*true} \qquad k \in SD$$

$$c \; = 0 \;\; for\, Y \; = false \;\; i \ne \hat{i}$$

$$\text{x e } R^{n}, \; c_{ik} \in R^{m},$$

htote that oiu^ooosOTiirtsconespoiKiMg to true boolean variables are imposed. Also fixed charges-ft* are only applied to these terms. Assuming that K subproblems (SP) are solved in which sets of linearizations M...JC are generated for subsets of disjunction terms *L(ik) -{11 Y*ik ~ true }. ont can* define the following disjunctive OA master problem:

$$\text{Min} \quad Z = \sum_{i}\sum_{k} c_{ik} + \alpha$$

$$\text{st} \quad \alpha \ge f(x^{l}) + \nabla f(x^{l})^{T}(x-x^{l})$$

$$\text{«*>} + Vhfxtftx\text{-}x^{1})\text{£}0 \qquad I \;\; \text{-}1..X \qquad \text{(MDP2)}$$

$$\underset{ieD_k}{\text{V}} \begin{bmatrix} Y_{ik} \\ g_{ik}(x^{l}) + \nabla g_{ik}(x^{l})^{T}(x - x^{l}) \le 0 \;\; leHik) \\ cu_{t} = \gamma_{ik} \end{bmatrix} \quad ke \;\; SD$$

$$am\text{-} \;\; True$$

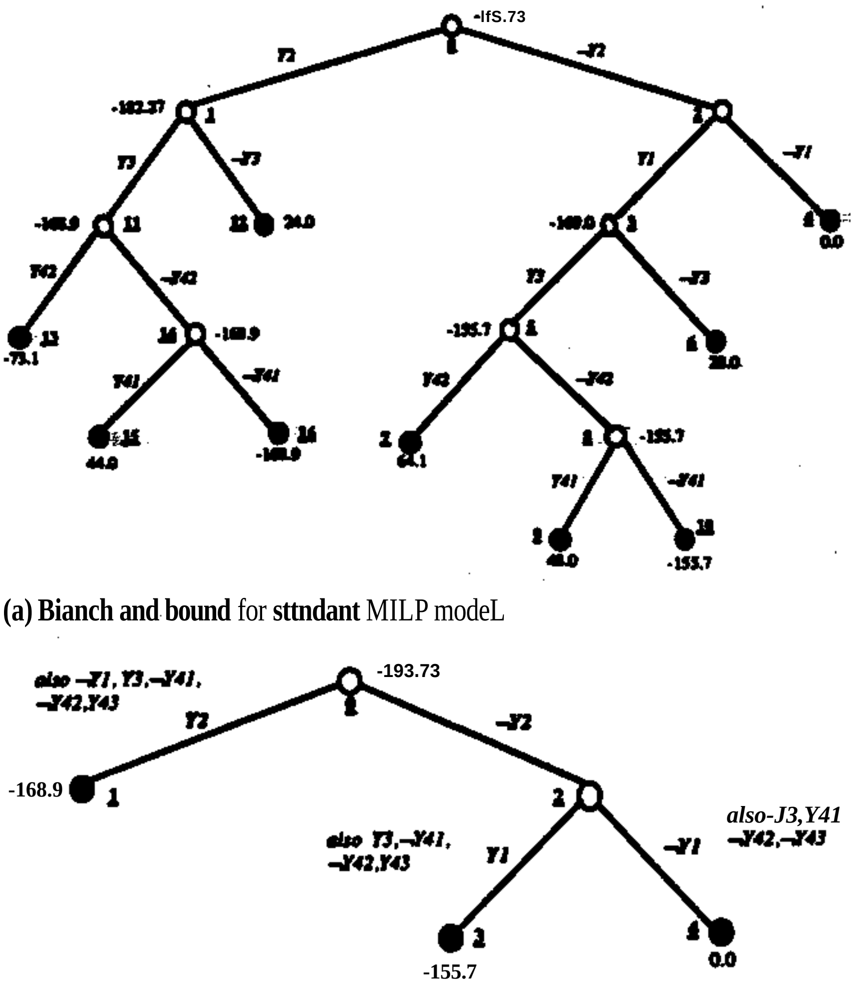
$$ae \;\; R, \; xe \;\; R^{H}.t{:}\text{€} \quad R^{m}.Ye \;\; \{true, \;\; false\}"$$

It should be noted that before applying the above master problem it is necessary to solve various subproblems so as to produce at least one linear approximation of each of the terms in the disjunctions. As shown by Turkay and Grossmann (1994) selecting the smallest number of subproblems amounts to the solution of a set covering problem. The above problem (MDP2) can be solved by any of the methods described for the linear case. It is also interesting to note that for the case of flowsheet synthesis problems Turkay and Grossmann (1994) have shown that the above solution method becomes equivalent to the modelling/decomposition strategy by Kocis and Grossmann (1988) if the master problem (MDP2) is converted into MEJ> form using a convex hull representation. Also, these authors have shown that while a logic-based GBD method cannot be derived as in the case of the OA algorithm, one can nevertheless determine for the MILP version of the master problem (MDP2) one Benders iteration which then yields a sequence similar to the GBD method for the algebraic case.

Computational Experience

From the methods described in the previous section the symbolic integration of logic both in DNF and CNF form have been automated in a special version of OSL, the MILP solver from IBM (Raman and Grossmann, 1993a). Also systematic methods have been developed to automate the generation of logic cuts in process networks (Raman and Grossmann, 1993a; Hooker et al$_f$ 1994). Work is also currently under way to automate the logic version of the OA and GBD algorithms.

In order to appreciate the potential impact of integrating logic in discrete optimization problems numerical results on selected examples are given in Table 3. Example (a) deals with an MILP for the synthesis of separation sequences involving 6 components (see Raman and Grossmann, 1992). Applying the standard version of Benders decomposition convci^eix^ is IKH achieved aftcrscvcralhoureaiKlnKrc than one hundred iterations on an older Vax-computer. In constrast, adding inequalities for the logic cuts in (DP4) convergence is achieved in only 13 iterations, and this despite the fact that the number of constraints is doubled. Note that the integrated master with heuristics is not as effective in this case. Example (b) deals with a small MINLP planning problem in which similar trends are observed when adding the logic cuts. The examples in (c) deal with the symbolic and hybrid integration of logic using branch and bound (see Raman and Grossmann, 1993). Note that for the MILP for the separation of 6 components the reduction in number of nodes enumerated is significant The more impressive results, however, are with the heat integrated model which corresponds to the motivating example. Adding the inequalities for the logic cuts the problem is solved to optimality in only 8 sec! And this is accomplished by almost doubling the number of constraints. With the symbolic integration of logic with DNF the time is even further reduced to less than 3 sec! The reason for the reduction is that in the symbolic integration there is no need to handle the inequalities for the logic cuts. It should be noted that the DNF logic involved 194 disjunctive terms. Therefore, theoretically it is possible to guarantee that the number of nodes in this type of enumeration will not exceed 387 nodes. In actual fact only 20 were needed. Finally, the example in (d) illustrates a problem in which a process network was initially formulated as the generalized disjunctive program (DP2) (see Raman and Grossmann, 1994). Converting it all into MILP form requires more than 1 hour of solution time with OSL. If instead the problem is formulated as in (DP3) in which disjunctions are identified that are not w-MIP representable the modified branch and bound method requires less than 10 minutes of CPU time. Fig. 6 presents the tree searches for a very small version of this problem. Note that even in this case the logic-based branch and bound for the disjunctive model (DP3) requires only 4 nodes as opposed to the 16 that are needed when the model is posed entirely as an MILP and solved with standard branch and bound methods.



(a) **Bianch and bound** for **sttndant** MILP modeL

(b) Logic based branch and bound for disjunctive model (DP3)

Fig. **6** . **Comparison of tree searches with standard and logic based branch and bound.**

Table 3. Computational Results on Selected Example Problems

(a) NflLP model 6 component separation. *Benders decomposition*

	Original Model (DPI)	Model with Logic (DP4)	Integrated Master (M2)
Constraints:			
Heuristic			187
Logic constraints		70	70
Other	86	86	86
Iterations	>100	**13**	**43**
Cpu-time*	>1000	**11.99**	**338.7**

'•min Micro-VaxD (SCICON1C)

(b) MINLP model planning **problem *GeneralizedBenders Decomposition***

	Original Model (DPI)	Model with logic **(DP4)**	Integrated Master (M2)
Heuristic constraints			**5**
Logic constraints	**1**	**8**	**8**
Other constraints	**9**	**9**	**9**
Number iterations	**7**	**3**	**4**
CPU time*	**28.20**	**11.7**	**18.8**

'•sec Micro-Vax D (SCICONIC/MINOS)

(c) MILP models. *Branch and bound*

	Original Model (DPI)	Model with logic (DP4)	**DNF based** approach	Hybrid DNF approach
Six components				
Logic constraints	0	70	0	11
no. of nodes	141	8	18	5
CPU time*	3.46	1.18	1.06	0.7
Heat Integrated Distillation				
Logic constraints	0	215	0	4
nodes	> 100,000	74	20	17
CPUtime*	> 5,000	**8.37**	2.76	2.62

'•secBM-RS6000(OSL)

(d) MILP Process Network with semi-continuous demands

	MILP model (DPI)	**Disjunctive Model (DP3)**
Constraints	**1382**	**1382**
Variables	**1326**	**1326**
Binary	73	**73**
Nodes	**16,532**	**1,771**
CPU time*	**76.2**	**8.3**

'•minlBM-RS6000(OSL)

GLOBAL OPTIMIZATION

Background

A significant effort has been expended in the last five decades toward theoretical and algorithmic studies of local optimization algorithms and their computational testing in applications that arise in Process Synthesis Design and CootroL Relative to such an extensive effort that has been devoted to local nonlinear optimization approaches, there has been much less work on the theoretical and algorithmic development of global optimization methods. In the last decade the area of global optimization has attracted a lot of interest form the Operations Research and Applied Matheniatics community, while in the last five yean we have experienced a resurgence of interest in Chemical Engineering for new methods of global optimization as well as the application of available global optimization algorithms to important chemical engineering problems. This recent surge of interest is attributed to three main reasons. First, a large number of process synthesis, design and, control problems are indeed global optimization problems. Second, the existing local nonlinear optimization approaches (e.g. generalized reduced gradient and successive quadratic programming methods) may either fail to obtain even a feasible solution or are trapped to a local optimum solution which may differ in value significantly form the global solution. Third, the global optimum solution may have a very different physical interpretation when it is compared to local solutions (eg. in phase equilibrium a local solution may provide incorrect prediction of types of phases at equilibrium, as well as the components' composition in each phase).

The existing approaches for global optimization are classified as deterministic or probabilistic. The deterministic approaches include: (a) Lipschitzian methods (e.g. Hansen et al. 1992 a, b), (b) Branch and Bound methods (e.g. Al-Khayyal and Falkl983; Horst and Tuy, 1987; Al-Khayyal 1990), (c) Cutting Plane methods (e.g. Tuy et al. 1985), (d) Difference of Convex (D.C.) and Reverse Convex methods (e.g. Tuy 1987 a,b), (e) Outer Approximation methods (e.g. Horst et al. 1992), (0 Primal-Dual methods (e.g. Shor 1990; Floudas and Visweswaran 1990,1993; Ben-Tal et al 1994), (g) Reformulation-Linearization methods (e.g. Sherali and Alameddine, 1992; Sherali and Tuncbilek 1992), and (h) Interval methods (e.g. Hansen 1979). The probabilistic methods include (i) random search approaches (e.g. Kirkpatrick et a. 1983), and (ii) clustering methods (e. g. Rinnoy Kan and Timmer 1987). Recent books for global optimization that discuss the above classes are available by Pardalos and Rosen (1987), Torn and Zilinskas (1989), Ratschek and Rokne (1988), Horst and Tuy (1990) and Floudas and Pardalos (1992).

Contributions from the chemical engineering community to the area of global optimization can be traced to the early work of Stcphanopoulos and Westerberg (1975), Westcrberg and Shah (1978), and Wang and Luus (1978). Renewed interest in seeking global solution was motivated form the work of Floudas et al (1989). The first exact primal-dual global optimization approach was proposed by Floudas and Visweswaran (1990), (1993) and its features were explored for quadratically constrained and polynomial problems in the work of Visweswaran and Floudas (1992), (1993). At the same time Swaney (1990) proposed a branch and bound global optimization approach and more recently Quesada and Grossmann (1993) combined convex undeiestimators in a branch and bound framework for fractional programs. Manousiouthakis and Sourlas (1992) proposed a reformulation to a series of reverse convex problems, and Tsirukis and Reklaitis (1993 a,b) proposed a feature extraction algorithm for constrained global optimization. Maranas and Floudas (1992,1993,1994 a,b) proposed a novel branch and bound method combined with a difference of convex functions transformation for the global optimization of atomic clusters and molecular conformation problems that arise in computational diemistry. Vaidyanathan and El-Halwagi (1994) proposed an interval analysis based method and Ryoo and Sahinidis (1994) proposed reduction tests for branch and bound based methods.

In this review paper, we will focus, on deterministic global optimization methods since they provide a rigorous framework for exploiting the inherent structure of process synthesis models . In particular, we will discuss decomposition based primal-dual methods and branch and bound with difference of convex functions global optimization approaches developed in the Computer-Aided Systems Laboratory, CASL, of the Department of Chemical Engineering of Princeton University.

Decomposition Methods

Floudas and Visweswaran (1990, 1993) proposed a deterministic primal-relaxed dual global optimization approach, GOP, for solving several classes of non-convex optimization problems for their global optimum solutions. These classes are defined as:

Determine a globally e-optimal solution of the following problem:

$$\min_{x,y} \; f(x.y)$$

$$\begin{array}{rrcl} \text{subject to} & g\,(x,y) & \text{£} & 0 \\ & h(x,y) & = & 0 \\ & X & \text{€} & X \\ & y & \text{€} & Y \end{array} \qquad \text{(PI)}$$

where X and Y are non-empty, compact, convex sets, $g(x,y)$ is an m-vector of inequality constraints and $h(x.y)$ is a p-vector of equality constraints. It is assumed that the functions $ftx.y)$, $g(x,y)$ and $h(x,y)$ are continuous, piecewise differentiable and given in analytical form over X x Y. The variables y are defined in such a way that:

(a) $f(x,y)$ is convex in x for every fixed y, and convex in y for every fixed r,

(b) $g(x,y)$ is convex in x for every fixed y, and convex in y for every fixed x and

(c) A(xj)isaffii^inxforevei7ftted>,aiulaffinein>tocveiy fixedx.

Examples of process synthesis problems with this structure are superstructures for separation systems, and heat exchanger networks in which balance equations involve bilinear terms, as well as phase equilibrium problems that can be transformed so as to exhibit the bi-convex characteristics of the above conditions.

Making use of duality theory along with several new theoretical properties, a global optimization algorithm, GOP, has been proposed for the solution of the problem through a series of **primal** and relaxed **dul** problems that provide valid upper and lower bounds on tbeglobal solution. The GOP algorithm decomposes the original problem into **primal** and **relaxed dual** subproblems. The primal problem is solved by projecting on the *y* variables, and takes the form:

$$v(y^k) = MB^* \, f(x, y^k)$$

x

subject to *g (x,/)* £ 0 (P2)

***h (x^)* * 0**

X€ X

A feasible solution x^k of the primal problem (P2) with objective ***viy*)*** represents an upper bound on the global optimum (i.e. Upper BouMfevfy*;) solution of (PI), and at the same time it provides the Lagrange multipliers X*. |JL* for the equality and inequality constraints respectively.

The Lagrange multipliers (X*. $\backslash i^k$) **are** subsequently used to formulate the Lagrange function *L* $(x, y_t$ X*. *\fy* which is used in the dual problem. Invoking the dual of (PI) and making use of several properties of the problem structure, the GOP algorithm solves a relaxation of the dual problem through a series of relaxed dual subproblems. The y-space is partitioned into subdomains and each relaxed dual subproblem represents a valid underestimation ef^d (PI) for a particular subdomain. Each relaxed dual is awociatftd with a combination of bounds B_p of the x variables which appear in bilinear x-y products in the Lagrange function, and takes the forms:

MIN *HB*

S.I.

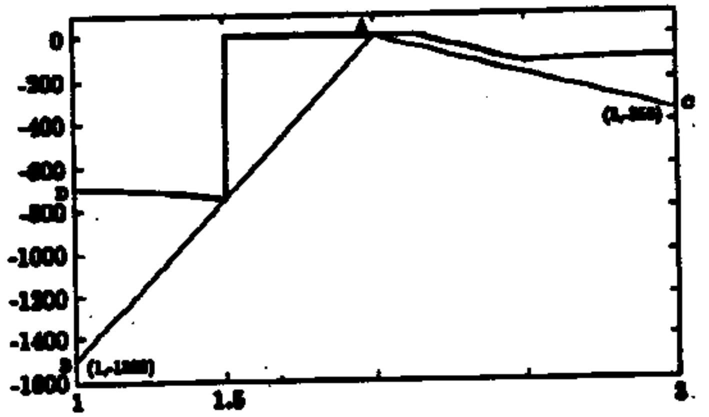
$$\left.\begin{array}{l} \mu_B \geq L\left(x^{Bj}, y, \lambda^k, \mu^k\right)\Big|_{x^k}^{lin} \\ \nabla_{x_i} L\left(x, y, \lambda^k, \mu^k\right)\Big|_{x^k} \leq 0 \text{ if } x_i^{Bj} = x_i^U \\ \nabla_{x_i} L\left(x, y, \lambda^k, \mu^k\right)\Big|_{x^k} \geq 0 \text{ if } x_i^{Bj} = x_i^L \end{array}\right\} k = 1, 2 \ldots, (K-1)$$

$$\left.\begin{array}{l} \mu_B \geq L\left(x^{Bp}, y, \lambda^K, \mu^K\right)\Big|_{x^K}^{lin} \\ \mathrm{V}_{x_i} \; L(x.y,X^K.n^K)\backslash_x K \;\; * \;\; 0 \;\; «\mathrm{f} \;\; xf? \; = \; x_i^U \\ \mathrm{V}_{x_i} \; L\left(x, y, \lambda^K, \mu^K\right)\Big|_{x^K} \;\; \geq 0 \;\; if \;\; x_i^{Bp} = x_i^L \end{array}\right\} \begin{array}{c} \textit{current} \\ \textit{iteration} \\ K \end{array} \qquad \textbf{(P3)}$$

The first three sets of constraints of (P3) correspond to the previous (K-l) iterations with the first one denoting the i_^gcpngf. undfifpstitPftting cute and the second tnd third defining the partitioning of the y-space. In the current iteration K the bounds Bj of the previous iterations are fixed while the current combinations of bounds Bp need to be considered. The last three sets of constraints, which change as B_p change, are the underestimating cuts for the partitioned subdomain under consideration. Hence, the relaxed dual problems at the current iteration K are equivalent to setting the x-variables to a combination of their bounds Bp,and solving for a corresponding domain of the y-variables. After solving (P3) for all combinations of bounds Bp, we select the minimum of these solutions and the solutions of the previous iterations. This will provide the new y to be considered in the primal problem (P2) and its corresponding solution is guaranteed to be a valid lower bound on (PI). Solving the primal problem (P2) and updating the upper bound as the minimum solution found, a monotonically non-increasing sequence of updated upper bounds is generated. Solving the relaxed dual problems (P3), a monotonically non-decreasing sequence of valid lower bounds is generated due to the accumulation of previous constraints. As a result, the GOP algorithm attains finite convergence to an e-global solution of (PI) through successive iteration between the primal and relaxed dual problems.

The GOP algorithm along with its primal problem (P2) and its relaxed dual problems (P3) have an interesting geometrical interpretation. Figures 7a, 7b and 7c illustrate graphically the GOP applied to the motivating pooling/blending problem discussed earlier. For a starting point of p * 2, the primal problem corresponds to point A of Figure 7a. Note that for p * 2 the primal problem is a linear programming problem with objective equal to zero. The y-space, which is 1 £ p £ 3, is pardoned into 2 sub-domains, one for 1 £ p £ 2 and the other for 2 ^ p £ 3, and one relaxed dual problem is solved for each sub-domain. Figure 7a shows the linear underestimator AB for 1 £ p S 2, and the underestimator AC for 2 £ p £ 3. Note that the underestimators are linear since the relaxed dual problems are linear in p and the points B and C correspond to the solutions of the corresponding relaxed dual problems. Also note that the underestimator AB passes through the global optimum (p * 1.3, -750). At the end of the first iteration we have an upper bound of zero and a lower bound of -1500. Since -1500 < -350, the next point under consideration for p is p = 1. For p « 1 the primal problem has as solution point D with objective value of -700. Since point D is in the boundary of the range of p, there is only one relaxed dual problem and hence one underestimator, shown as DE in Figure 7b, where point E is the solution of the relaxed dual problem.

Figure 7(a)t Iteration I



F

Figure 7(b): Iteration H

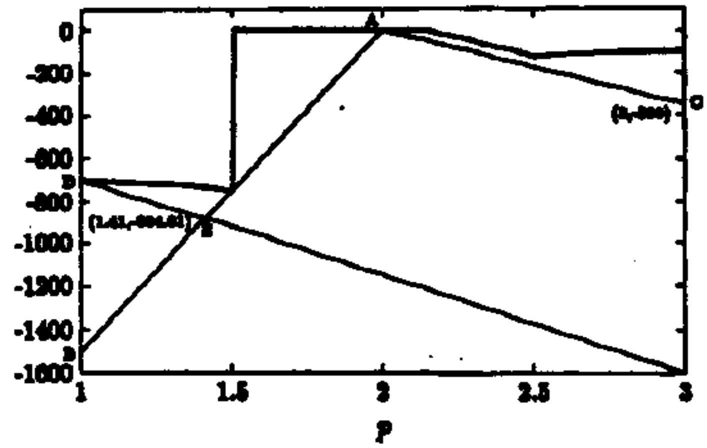
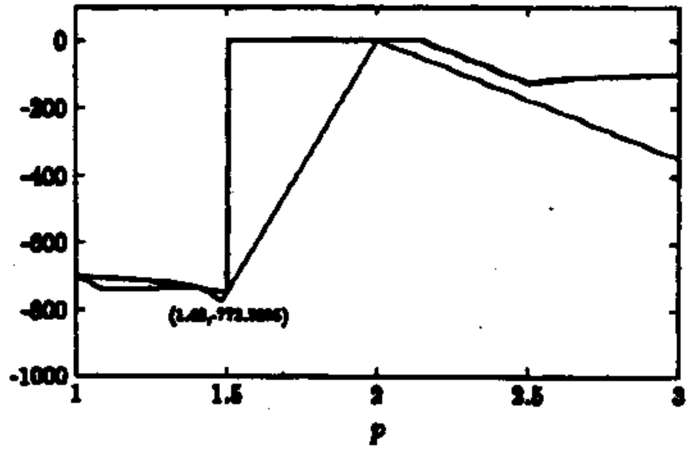


Figure 7(c)t Undereitimator after Iteration III



At the end of the second iteration, we have an upper bound of -700 and a lower bound of -884.61. Since -884.61 < -350, the next p under consideration is p « 1.41. Figure 7c shows the underestimating function after three iterations of the GOP algorithm. Note that we have a piece-wise linear underestimating function. Also note that since the primal problem for p« 1.41 has lower value than -350 we can eliminate the domain 2 £ p £ 3. The GOP algorithm has quickly identified the region of the global optimum by providing tight upper and lower bounds, and converges to the global solution in 6-7 iterations.

Visweswaran and Floudas (1990) demonstrated that the *Global Optimization Algorithm. GOP, cm* address several classes of BOB-COOvex mathematical problems that include:

(i) Bilinear, negative definite and indefinite quadratic programming problems.

(ii) Quadratic programming problems with quadratic constraints.

(iii) Unconstrained optimization of polynomial functions.

(iv) Optimization problems with polynomial constraints.

(v) Constrained optimization of ratios of polynomials.

Analysis of the results, obtained via the computational experience of the GOP algorithm on the above mentioned classes of nonconvex optimization problems, verified that a global optimum solution can be obtained from any starting point

Visweswaran and Floudas (1992) studied the class of polynomial functions of one variable in the objective and constraints of problem (PI) and showed that the primal problem reduces to a single function evaluation while the relaxed dual problem is equivalent to the simultaneous solution of two linear equations in two variables. The resulting global optimization approach was demonstrated to perform favorably compared to other algorithms.

Visweswaran and Floudas (1993) proposed new theoretical properties that enhance significantly the computational performance of the GOP algorithm. These properties exploit further (i) the structure of the linearized Lagrange function around x^k, which contains bilinear terms in x and y, linear terms in x, and either linear or convex terms in y, and (ii) the gradients of linearized Lagrange function around x^k, which are linear functions of only the y variables. The first property identifies the combinations of bounds that need not be considered if the gradients of the linearized Lagrange function maintain the same sign. The second property shows that if the gradient of the linearized Lagrange function with respect to xi is zero, then we can set x\ to either its lower or upper bound. The third property allows for updates of the bounds on the x variables at each iteration. Properties 1 and 2 reduce significantly the number of combinations of bounds of the x variables, and hence reduce the number of relaxed dual problems that needed to be solved at each iteration. Property 3 results in tighter underestimators for each of the partitioned subdomains, which in turn results in faster convergence of the upper and lower bounding sequences. The effect of the new properties is illustrated through application of the GOP algorithm to a difficult indefinite quadratic problem, a multiperiod tankage quality problem that occurs frequently in the modeling of refinery processes, and a set of pooling/blending problems from the literature. In addition, extensive computational experience is reported for randomly generated concave and indefinite quadratic programming problems of different sizes. The results show that the properties help to make the algorithm computationally efficient for fairly large problems. Visweswaran and Floudas (1994) presented a (MILP) formulation for aU relaxed duak ai each ben^on of the GOP algroithm. This is baaed on a branch and bound framework for the GOP and allows for implicit enumeration of the partitioned subdomains.

A very important advance on the GOP approach has been recendy made by Liu and F k > ^ (1993). It is shown thai the GOP approach can be applied to very general dasses of nonlinear problems defined as:

MIN F(x)
x

ST. Gi(x)£O i«l,2,...,m (P4)
X€ X

where X is a non empty, compact, convex set in R^n, and the functions F(x), Gi(x) are C^2 continuous on X. This result, even though it is an existence theorem, is very significant because it extends the classes of mathematical problems that the GOP can be applied to firom polynomials or ratios of polynomials to arbitrary nonlinear objective function and constraints that may include exponential terms and trigonometrk: terms with the only requirement that these functions have continuous first and second order derivatives. Based on this result, it is clear the GOP approach is applicable to very broad mathematical problems.

Branch and Bound Methods with (D. C.) transformatlon

A novel branch and bound global optimization approach which combines a special type of difference of convex functions' transformation with lower bounding underestimating functlons was recently proposed by Maranas and Floudas (1994 a,b). This approach is applicable to the broad class of optimization problems stated in (P4), and this special type of (IXC.) transformation is the basis of the result reported in Liu and Floudas (1993). In the sequel, we will discuss the essential elements of this approach by considering the problem of:

MIN F(x)
x

<P5)

ST. x€X * {xj|xf £ xj £ x_i£xj\i = l,2,...,n}

where X is a nonempty, compact, convex set in R^n, and the objective function F(x) is C^2 continuous on X.

Adding a separable quadratic term to F(x), introducing new variables $x\{ * x_h$ and subtracting the same term firom F(x) we have:

$$\text{MIN} \qquad F(x) + \alpha \sum_{i=1}^{n} [x_i^2 - x_i \cdot x_i']$$

xf i Xj £ x_i''

$$(x_i')^{\ell} \le x_i' \le (x_i')^{u}$$

(P6)

S.T. xj-x{ **=0** **f«1.2**. . . .n

The key idea is to employ eigenvalue analysis and define the nonnegative parameter a in such a **way that the following tern:**

$$+(x)\text{-}F(x) + a \sum_{i=1} x?$$

becomes convex. Then, (P6) takes the form

$$\text{MIN} \qquad \phi(x) - \alpha \sum_{i=1}^{n} x_i x_i'$$

xf £ *x_i* £ x_i^u

$$(x_i')^{\ell} \le x_i' \le (x_i')^{u}$$

(FT)

S.T. Xj - x[= 0

which has as objective a difference of two convex functions out of which the one that is substracted is separable quadratic. Formulating the dual of (F7) and applying die KKT conditions, Maranas and Floudas (1994***JBL)*** showed that the dual of (P7) is equivalent to (P8) (see Appendix A3 of that paper):

$$\text{MIN } L(x) = \left[F(x) + \alpha \sum_{i=1}^{n} x(\,\text{-}f x_i \; X \; r\text{-}x_i\,)^{7} \right]$$

xf ^ Xi ≤*xf*

(P8)

where **a** is a nonnegative parameter which is greater or equal to the negative one half of the minimum eigenvalue of the Hessian of F(x) over the box xf £ x_k £ x"$_i$ i ** 1, 2, ...,n $\left(\text{i.e} \quad \mathbf{a} \text{ £ MAXj} \left\{ 0_y \text{—} \frac{1}{2} \lambda_{min} > \right\} \right)$ Note that the term added to F(x) has the effect of overpowering the

nonconvexity characteristics of F(x) with the addition of the term (2 a) to all of the eigenvalues of its

Hessian. The smaller the value of a, the tighter the uiulefestimator Ux) is fcr F(x) which may imply less total manber erf iterations for convergence. Hence, one would ideally desire the non negative parameter a to be exactly equal to $\left(-\frac{1}{2}\right.$^ min $\left.\right)$ **** •*>***« th* to the branch and bound widi difference of convex function transformation it suffices to find an upper bound on $\left(\frac{1}{2}\right.$-T^awi 1$\left.\right)$ »* ^{9ckci} ○ •* *$& t$^{\circ this}$ *upper* bound. In this case we add more convex terms than needed and do not produce the tightest underestimator, bat we satisfy dierequiredconditions for convergence.

Given F(x), the aekction erf the nonnegative parameter a may involve (i) die derivation of analytical expressions for the eigenvalues erf its Hessian, or (ü) the devetoproemcrf bounds on the eigenvalues of the Hessian of F(x). Mannas and Floudas (1992), (1993) studied alternative (i) for a variety of atomic/molecular clusters. They derived analytical expressions for the eigenvalues for any potential function which is a function of only the distance between atoms (e. g. Lennard-Jones, Coulomb, Mie, Morse, Gaussian, Bom-Mayer, Buckingham). Mannas and Floudas (1994. a, b) proposed a number of ways of obtaining bounds on the eigenvalues of the Hessian of F(x). One general way is via the use of the measure of a matrix, a concept recently applied to the stability erf reactor networks at die process synthesis level (see Kokossis and Floudas, 1994). If a denotes die Hessian of F(x),thcn the measure of the matrix (-A), denoted as \i (-A), provides an upper bound on (-Xmin)- $^{\text{Ils}}$ formulation is a convex problem, and we can use either the 1 or ©o norm. Appendix A.2 of Mannas and Floudas (1994*), describes such a formulation.

It should be pointed out however that if Xmin goes to (- «>), then this represents a case in which we cannot create <J>(x) convex. A sufficient condition which excludes such a possibility is when the dements of the Hessian matrix have finite values. This can be seen easily using the measure erf a matrix concept One instance of Xmin tending to (- °°) is reported in the Weber's facility location problem (see Mannas and Flouds, 1994.C)

The function L(x) is a lower bounding function of F(x), and exhibits the following important properties:

Property 1: L(x) is always a valid underestimatorofF(x) inside the box $[xf_f \ x_i^*]$, that is

$$L(x) \leq IF(x).$$

Property 2: L(x) matches F(x) at all corner points erf the box.

Property 3: L(x) is convex in the box $[xf, \ x_i'']$.

Property 4: The maximum separation between L(x) and F(x) is bounded and is proportional to a and to the square of the diagonal of the box $[xf, \ x_i'']$, that is

$$\max_{x_i^\ell \le x_i \le x_i^u} (F(x)\text{-}Ux)) = \text{iai}(x?\text{-}xf)$$

Property 5: The undercstiiiuttorL(x) constructed over tsub-tox of the current box is always lighter than the underestimator of the current box.

In summary, the properties show that L(x) is a convex, lower bounding function of F(x), L(x) matches F(x) at all comer points erf the box constraints inside which it has been defined. The values of L(x) at any point, if L(x) is constructed over a tighter box of constraints each time, define a nondecreasing sequence. Also note that Property 4 answers the question of how small the sub-boxes must become before the upper and lower bounds of F(x) are within £. If 5 is the diagonal of the sub-box, and E is the convergence tolerance, we have have:

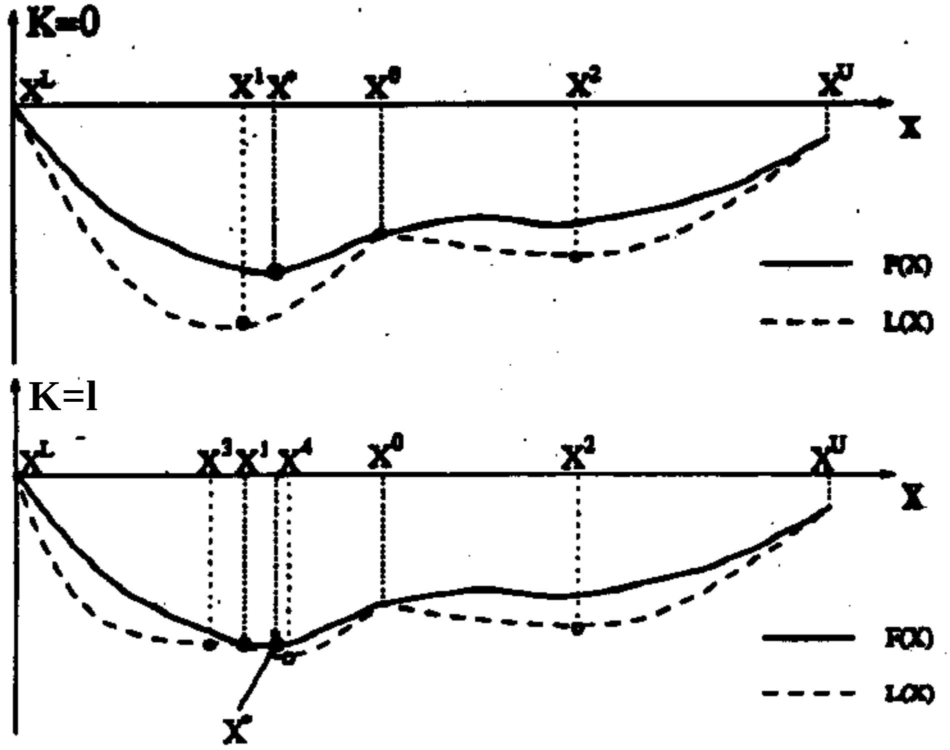
$$\delta < \sqrt{\frac{4\varepsilon}{\alpha}}$$

Note that 8 is proportional to the square root of e and inversely proportional to the square root of a. As a result, the smaller value of a the faster the convergence rate becomes.

These properties of the lower bounding function, L(x), coupled with an efficient partitioning scheme resulted in a branch and bound global optimization approach that is guaranteed to converge to an e-global solution in a finite number of iterations. Maranas and Floudas (1994*) analyzed the structure of the branch and bound tree resulting from the subdivision process and developed formulas for finite upper and lower bounds on the total number of iterations required for £ - convergence. The maximum number of iterations is exponential in the total number of variables while the minimum number of iterations depends linearly on the total number of variables. Computational experience with molecular conformation problems indicated that the total number of iterations is much close to the minimum one.

Figure 8 provides the geometrical interpretation of the lower bounding scheme for a function F(x) of one variable x in a box$[x^L, x^u]$. Starting at a point x° we partition the original box into two intervals $[x^L, x°]$ and $[x°, x^u]$, while F(x°) is the current upper bound. For each interval we solve the corresponding convex lower bounding problem and obtain their respective minima at x^1, L(x*) and x^2, $L(x^2)$ respectively. Note at this point the underestimating functions shown with non-solid lines.

Figure 8: Geometric Interpretation of Branch and Bound with (D.C.)



> mlnF(JV), single variable problem in X

- $L(X) = F(X) + a(X^{LBD} \quad -XXX^{UBD} \quad - \quad X)$

Since $L(x^*) < L(x^2)$ we focus on the $[x^L, x^\circ]$ for the second iteration, evaluate the function $F(x^1)$ andpaitition the interval $[x^L, x^\circ]$ into the intervals $[x^L, x^1]$ and $[x^1, x^0]$. For each of these intervals we obtain the undcrcstimators and their minima which arc at x^3 and x^4 respectively. Since $L(x^3) > L(x^4)$ we focus on the interval $[x^1, x^4]$ forthe next intention and evaluate $F(x^4)$. Note that we are very close to the global solution in just two iterations.

The branch and bound with (D. C) transformation was applied to (a) clusters of atoms/molecules in which only non-bonded interactions take place, (b) molecular structure detenninatm of small molecules in which bonded and non-bonded interactions are taken into account, and (c) financial planning models for multiperiod operation. Application (a) resulted in ratios of polynomials and exponential terms in the distances between atoms. Application (b) involved very complex expressions not only in the distances but also in the dihedral angles and had ratios of polynomials, exponentials, and trigonometric terms. Application (c) employed multiperod models for stochastic programming using the mean-variance model over all possible scenarios, and resulted in generalized polynomials and square root terms. All computational results highlight the power of the (D. C.) transformation within a branch and bound framework.

Global Optimization Tools and Computational Experience

Global optimization tools have been recently developed in the Computer Aided Systems Laboratory, CASL, of the Department of Chemical Engineering at Princeton University for the primal-relaxed dual algorithm, GOP, and the branch and bound approach that combines (D.C.) transformation and a special type of lower bounding function. These tools are denoted as cGOP and OtBB for the decomposition and branch and bound global optimization algorithms respectively. Both cGOP and a B B are written entirely in C and make use of MINOS, NPSOL, CPLEX for linear subproblems; MINOS, NPSOL for nonlinear programming subproblems. They have been implemented as a library of subroutines with emphasis on modularity and expandability, the subroutines for the same task have the same interfaces, and modifications in the problem data are allowed at any stage. Both cGOP and a B B have a user specified function capability which allows for connection to any external subroutine that can be treated as a black box. The current versions of cGOP and a B B can be either standalone or can be called as subroutines.

Computational experience with cGOP and a B B is shown in Table 4 and Table 5 for a wide variety of applications, that include: pooling/blending problems, heat exchanger network synthesis problems, nonsharp separation synthesis, problems with quadratic objective and box constraints, concave programming problems, bilevel linear optimization problems, minimization of the Gibbs free energy with NR1L and UNIQUAC in phase and chemical reaction equilibrium, tangent plane stability criterion in phase equilibrium, clusters of atoms and molecules, molecular structure determination problems, and financial planning problems. The first three and the last pooling problems correspond to the Haverly problem and the multiperiod tankage problem and are described in Floudas and Visweswaran (1990) and Visweswaran and

Floudas (1993). The fourth and fifth pooling problems are described inBen-TaletaL(1994). The first two heat exchMigcr problems are taken from Floudas and Ciric (1989) while the last three are descibed in Ben-Taletal.(1994). The first two heat exchanger problems are taken from Floudas and Ciric (1989) whik the last three are described in Quesada and Gfossmann (1993). The separations problem is described in Aggarwal and Floudas (1990). The minimization of Gibbsfirce energy problems arc discussed in McDonald and Floudas (1994a). The tangent pine stability criterion problems are presented in McDonald and Floudas (1994b). The quadratic objective with box constraints, concave objective with linear constraints, and indefinite quadratic problems are discussed in Visweswaran and Floudas (1993). The Lennard Jones clusters of atoms problems are discussed in Mannas and Floudas (1993). The molecular structure determination problems are presented in Maranas and Floudas (1994jub.). The molecular structure detenninaoon problems arc presetted inM aranas and Floudas (1994 a,b). The financial planning problems are described in Maranas et aL (1994). As Tables 4, 5 illustrate, small medium, and in certain cases large global optimization problems can be solved within a modest computational effort.

Plane	TWA3G	<$	3	2	85	0.94
Stability	PBW3T1	6	3	2	53	0.62
Criterion**	PBW3G1	6	3	2	213	2.37
	PBW3T6	6	3	2	549	4.98
	PBW3G6	6	3	2	757	7.09
Quadratic	QBR1	10	300	«	2	6.45
Objective,	QBR2	20	300	-	2	46.01
Box	QBR3	30	160	-	2	345.83
CrOnstFaints	QBR4	30	300	-	2	411.016
	CLR1	50	SO	SO	2	1.62
Concave	CLR2	100	100	100	2	22.95
Objective	CLR3	SO	ISO	100	2	0.65
Linear	CLR4	SO	200	100	2	2.73
Constraints	CLR5	SO	250	100	2	10.47
	CLR6	100	250	100	2	47.5
	IND1	100	100	100	2	11.53
	IND2	SO	50	50	2	0.71
Indefinite	IND3	100	50	50	2	4.35
Objective,	IND4	SO	100	50	2	1.28
Linear	IND5	50	200	50	3	15.17
Constraints	IND6	50	200	100	2	6.76
	IND7	75	200	100	2	17.72
	IND8	50	250	100	2	22.27
	BL1	2	3	6	3	0.47
	BL2	2	2	5	3	0.28
	BL3	1	1	6	2	0.11
Bilevel	BL4	1	1	5	3	0.23
Linear	BLS	6	3	10	3	0.75
	BL6	1	1	5	3	0.29
	BL7	1	2	4	2	0.16
	BL8	1	1	4	3	0.23
	BL9	1	1	4	3	0.22
	BL10	1	2	4	2	0.16
	BL11	2	3	6	5	0.82

N_x	:	number of x-• variables
Ny	:	number of y-• variables
Nc	:	number of constraints
Nl	:	number of iterations
CPU	:	see's in HP-730
•*	:	using GLOPEQ (McDonald and Floodas, 1994)

Table Si Computational Rg«ult« with (XBB

I. Clusters of Atoms/Molecules

Problem Name	TV	NCV	RT	Ni
U8	18	3	1%	12
U13	33	3	1.5%	15
LJ18	48	3	1.5%	20
LJ22	60	3	1.5%	16
U24	66	3	13%	19

n. Mnlwulor ^trurtUK 'Determination

Problem Name	iy.	NCV	RI	Ni
PRO	21	2	**0.01%**	400
APRO	27	2	0.01%	200
ABUT	51	3	0.01%	1000
BUT	54	3	0.01%	100
NPEN	90	4	0.01%	1000

m. Financial Planning

Problem Name	TV	NCV	Nc	Ni
FM100	8	8	11	2
FM300	8	8	1I	2
FM500	8	8	1I	3
FM1000	8	8	11	6
FM10000	8	8	11	6
FMC100	8	8	11	2
FMCTX100	8	8	1I	7

TV: total number of variables

NCV : nonconvex variables

RT: relative tolerance

CONCLUDING REMARKS

This *paper* has attempted to present an overview of two major emerging areas in algorithmic synthesis: logic and global optimization. As indicated at the beginning of the paper these areas have been motivated by the need to improve the modelling in discrete optimization techniques, reduce the combinatorial search and avoid getting trapped into poor suboptimal scdutions. In the next two subsections we briefly discuss some future directions for research.

Current and Future Directions for Logic Based Optimization

Comparing the review on MINLP given by Grossmann (1990a) at the previous Snowmass meeting, it is apparent that the work on logic based optimization has provided a new direction to address the need of integrating qualitative knowledge into mixed-integer optimization models for synthesis (see also Rippin, 1989). As has been shown by developing new models and branch and bound methods that effectively incorporate logic, order of magnitude reductions can be achieved in the combinatorial search involved in these problems. Furthermore, another very important aspect has been to achieve a better understanding of some fundamental issues related to the modelling of discrete optimization problems. In particular, the concept of w-MIP representability has proved to be a useful theoretical concept few characterizing the nature of discrete constraints. While significant progress has been made, it is clear that a number of major issues and challenges still remain for future research. These include the following:

1. The handling of temporal and modal logic is challenging and should prove to be very useful for a wide range of problems in process scheduling.

2. Other kinds of logic cuts should be investigated apart from the logic relating units in a superstructure. The cuts affect the solution efficiency considerably and also allow one to better understand the modelling of discrete programming problems. One possibility for logic cuts are constraints that prevent multiple mathematical representations for the same design configuration within a superstructure.

3. Most of the work on integration of logic has been directed to discrete linear problems. Still much work remains in the integration of logic for nonlinear problems. In addition, there is the issue of integration with new cutting plane methods such as the one by Balas et al. (1993).

4. The problem of developing techniques to efficiently model and solve superstructures of large scale process flowsheet problems is another major issue. The use of disjunctions should serve to reduce the level of nonlinearity present in a mixed-integer representation, as well as allow for a systematic scheme for generating efficient models for these problems.

5. Further study is required on the representability of disjunctive constraints as mixed-integer constraints. Our work on w-MIP representability can only be regarded as preliminary work in the area and has just demonstrated the potential for research in this problem. A better understanding of representability issues could lead to the development of modelling languages for generating efficient discrete optimization models.

6. The ctevetopment of computer software th^ efficiently automates the various approaches based on logic and their more extensive testing on large scale problems is still required.

7. The integration with other design methodologies should be exploit in which logic information can be generated from a preliminary screening. Example of this are the work by FriedleretaL (1991) and the work by Daichcndtand Grossmann (1994aJ>).

8. The ultimate objective is ID provide aso^ foundation to new danes of hybrid optiinization models which are expressed in terms of equations and logic relatwns. This shouW also provide a clefflkik with dynamic simulation models which yi#A tend to exhibit this structure.

Progress and better understanding in the above problem will undoubtabiy lead to a new generation of discrete optimization models and solution methods. Furthermore, it is clear that these efforts can complement advances in global optimization.

Current and Future Directions in Global Optimization

In the global optimization section we have attempted to present an overview of global optimization methods which are based on the concepts of decomposition and branch and bound coupled with a (DC) transformation. From this review, it is apparent that we have experienced a significant progress in the area of global optiinization and its applications in Chemical Engineering over the last five years. New theoretical results and algorithms have emerged and their application to a number of Process Synthesis, Design, and Control problems has already resulted in encouraging results. At.the same time applications in the area of computational chemistry, facility location, and financial planning demonstrate clearly the potential impact of global optimization in the design of new materials and biological systems, the design of process layout, and the design of financial management systems. It is also worth noting that it is the first time that the progress in the area of global optimization is reviewed in a FOCAPD meeting, which is indicative of the recent advances, the potential usefulness, and the growth of this area in Chemical Engineering Design and Control. Global optimization, as a new area, however has a number of important challenges and several open problems which will be the subject of current and future research woik. These challenges include:

(1) new global optimization approaches for non-convex (MINLP) models arising in Process Synthesis;

(2) global optimization methods for generalized geometric programming problems (e.g. signomials) which arise in many design and robust control applications;

(3) new global optimization methods for nonconvex models with trigonometric and exponential functions that arise in Computational Chemistry, Biology and fMwu^ml reaction engineering applications;

(4) global optimization methods which can determine all solutions of nonlinear systems of equations that arise in phase equilibrium, azcotropic distillation, and reaction engineering;

(5) global optimization methods for bilcvcl and muUilevel linear and nonlinear models that appear in planning problems, flexibility analysis, and optimal control approaches in batch distillation;

(6) new global optimization approaches whkA can address impliciUy define functions; and

(7) distributed computing methods for global optimization with the aim at addressing medium to large scale optimization problems.

Even though the above challenges represent undoubtedly formidable tasks, we should see exciting developments over the next dfcatfp.

References

Aggarwal A. and C. A. Floudas, "Synthesis of General Distillation Sequences-Nonsharp Separations", Computers and Chemical Engineering, 14,6, pp. 631-653,1990.

Al-Khayyal F. A. and J. E. Falk, "Jointly Constrained Bioconvex Programming", *Math Opers. Res.*, 8, (1983).

Al-Khayyal, F. A., "Jointly Constrained Bilinear Programs and Related Problems: An Overview", *Computers in Mathematical Applications,* Vol. 19, pp. 53-62,1990.

Andrecovich, MJ. and A.W. Westerberg, "An MILP Formulation for Heat-Integrated Distillation Sequence Synthesis," *AIChEJ.* 31,1461 (1985b)

Bagajewicz, M. and V. Manousiouthakis, "On the Generalized Benders Decomposition," *Comptu. chem.* £ngng.9, 1(, 6 1 91) 1

Balas, E. /Disjunctive Programming: Properties of the convexhull of feasible points," MSRR #348, Carnegie Mellon University, Pittsburgh, PA.(1974).

Balas, E. /Disjunctive Programming and a hierarchy of relaxations for discreteoptimization problems, *SIAM J. Alg. Disc. Meth., 6,* 466-486 (1985).

Balas, E., Ceria, S. And Cornuejols, G. A Lift-and-Project Cutting Plane Algorithm for Mixed 0-1 Programs, *Mathematical Programming,* S8,295-324 (1993).

Beaumont, N. "An Algorithm for Disjunctive Programs," *European Journal of Operations Research,* 48, 362-371(1991).

Benders, J. R, "Partitioning Procedures for Solving Mixed-variables Programming Problems", *Numerische Mathematik 4,*238-252 (1962).

Ben-Tal A., G. Eiger, and V. Gershovitz, "Global Minimization by Reducing the Duality Gap", *Mathematical Programming,* in press (1994).

Cavalier, T. M. And Soyster, A. L.,"Logkal Deduction via Linear Programming/ IMSE Working Paper 87-147, Department of Industrial and Management Systems Engineering, Pennsylvania State University (1987).

Cavalier, T. M., Pardalos, P. M. And Soyster, A. L., "Modeling and Integer Programming techniques applied to Prepositional Calculus," *Computers and Operations Research,* 17(6), 561-570 (1990).

Chandru, V. And Hooker, J. N., "Extended Horn Sets in Prepositional Logic," Working Paper 88-89-39, Graduate School of Industrial Administration, Carnegie Mellon University, Pittsburgh (1989).

Clocksin, W. R And Mcllish, C S., *"Programming in Prolog,"* Springer-Veriag, New York, NY (1981).
Cook, S. A. t "The complexity of theorem proving procedures/ Proceedings of the 3rd ACM Symposium oft the Theory of Computing, pp. 151.158(1971).

Dakhendt, MM. and IJB. Grossmann, "Preliminary Screening for the MINLP Synthesis of Process Systems 1: . Ag|"ft^»andDecomposition*Tecbniqpt$,"Com/Ha.chem.Engng.*1 18,663(1994a)

Dafchendt, M.M. and LE. Grossmann, "Preliminary Screening for the MINLP Synthesis of Process Systems n: Heal Exchanger Networks,"Compn/. *chem. Engng.*, 18,679 (1994b)

Davis, M. And Putnam, H., "A computing procedure for quantification theory/ /. *ACM*, 8, 201-215 (1960).

Denenberg, L. And Lewis, H. R., "Logical Syntax and Computational Complexity," Proceedings of the Logic Q)lk)quium at Aadieiu Springer Uctuie Notes in Mathematics 1104, pp. 109-115 (1983).

Douglas, *JM.*, "A Hierarchical Decision Procedure for Process Synthesis," *AIChEJ.* 31,353 (1985).

Dowling, W J. And Gallier, J. H., "Linear-time algorithms for testing the satifiability of prepositional Horn formulae," *LogicProgramming*, 3,267-284 (1984).

Duran, M.A. and IJL Grossmann, "An Outer-Approximation Algorithm for a Class of Mixed-integer Nonlinear Programs," *Math Programming* 36,307 (1986).

Floudas, C. A. and A. Aggarwal, "A decomposition strategy for global optimum search in the pooling problem", *Opers Res. J. Compute* 2 (3) (1990).

Floudas, C. A., A. Aggarwal and A. R. Ciric, "Global optimum search for nonconvex NLP and MINLP problems", *Computers Chem. Engng.*, 13,1117 (1989).

Floudas, C. A. and A. R. Ciric, "Strategies for Overcoming Uncertainties In Heat Exchanger Network Synthesis", Computers and Chemical Engineering, 13,10, pp. 1133-1152,1989.

Floudas, C.A. and G.E. Paules IV, "A Mixed-Integer Nonlinear Programming Formulation for the Synthesis of Heat-Integrated Distillation Sequences," *Comput. chem. Engng.* 12,531 (1988)

Floudas, C. A., and Pardalos, P. M., "A Collection of Test Problems for Constrained Global Optimization Algorithms", *Lecture Notes in Computer Science,* Springer-Verlag, Berlin, Germany, Vol. 455,1990.

Floudas, C. A., and Pardalos, P. M., "Recent Advances in Global Optimization", Princeton University Press, Princeton, New Jersey, 1992.

Floudas, C. A. and V. Visweswaran, "A Primal-Relaxed Dual Global Optimization Approach", *Journal of Optimization, Theory, and its Applications,* 78,2, pp. 87-225 (1993).

Floudas, C. A. and V. Visweswaran, "A Global Optimization Algorithm (GOP) for Certain Classes of Nonconvex NLPs -1. Theory", Computers and Chemical Engineering, 14,12, pp. 1397-1417,1990.

Friedlcr, F., K. Taijan, Y.W. Huang and L.T. Fan, "An Accelerated Branch and Bound Method for Process Synthesis," Presented at the 4th Worid Congress of Chemical Engineering, Karlsruhe (1991).

Grossmann, LE., "Mixed-Integer Programming Approach for the Synthesis of Integrated Process Flowsheets," *Comput. chem. Engng.* 9,463 (1985)

Grossmann, LE.,t "MINLP Optimization Strategies and Algorithms for Process Synthesis," in *Foundations of Computer-Aided Design,* JJ. SiirolaJ.E. Grossmann and G. Stephanopoulos (Eds.), Cache-Elsevier, Amsterdam (1990a)

Grossmann, I.E., "Mixed-Integer Nonlinear Programming Techniques for the Synthesis of Engineering Systems/ *Res. Eng. Des. I,*205 (1990b)

Grossmann, I.E. and M.M. Daichendt, "New Trends in Optimization-based Approaches for Process Synthesis", to appear in Proceedings of Process Systems Engineering, Korea (1994).

Geoffrion, A. M, "Generalized Benders Decomposition," *Journal of Optimization Theory and Applications,* 10(4X237-260(1972).

Gundcrscn, T. and L. Naess, The Synthesis of Cost Optimal Heat Exchanger Networks. An Industrial Review of the State of the Art," *Comput. chem. Engng.* 12,503 (1988)

Hendry, JJE., D.F. Rudd and J.D. Seader, "Synthesis in the Design of Chemical Processes/ *AIChEJ.* 19,1 (1973)

Hansen, P., B. Jaumard and S. Lu, "Global Optimization of Univariate Lipschitz Functions: I. Surrey and Properties". *Mathematical Programming,* 55,2S1-272 (1992*).

Hansen, P., B. Jaumard and S. Lu, "Global Optimization of Univariate Lipschitz Functions: New Algorithms and Computational Comparison", *Mathematical Programming,* 55,273-292 (1992.b).

Hansen, E. R., "Global Optimization Using Interval Analysis: TheMulti-dimensional Case", Numerische Mathematik, Vol. 34, pp. 247-270, 1980.

Havcrly, C. A., "Studies of the behaviour of recursion for the pooling problem", *SIGMAP Bull,* 25,19 (1978).

Hlavacek, V., "Synthesis in the Design of Chemical Processes,", *Computera and Chemical Engineering,* 2, 67-75(1978).

Hooker, J. N., "Resolution vs Cutting plane solution of inference problems : some computational experience," *Operations Research Utters,* 7(1), 1 (1988).

Hooker, J.N., H. Yan, I.E. Grossmann, and R. Raman "Logic Cuts for Processing Networks with Fixed Charges," *Computers and Operations Research* 21,265-279 (1994).

Horst, R. and H. Tuy, "On the convergence of global methods in multiextrcmal optimization", *J. Optimization Theory Applic,* 54,253 (1987).

Horst, R., Thoai, N. V., and De Vries, J., "A New Simplicial Cover Technique in Constrained Global Optimization", *Journal of Global Optimization,* VoL 2, pp. 1-19,1992.

IBM, "OSL User Reference Manual," IBM Corp, Kingston, New York (1991).

Jcroslow, R. G. And Lowe, J. K., "Modelling with Integer Variables," *Mathematical Programming Study,* 22, 167-184 (1984).

Jeroslow, R. E. and Wang, J. /Solving prepositional satisfiability problems". *Annals of Mathematics and AI,* 1,167-187 (1990).

Karp, R. M., "Reducibility among combinatorial problems", *Complexity of Computer Calculations* (Miller, R. E. and Thatcher, J. W., eds) Plenum, New York, pp. 85-104 (1972).

Kocis, G.R. and I.E. Grossmann, "Relaxation Strategy for the Structural Optimization of Process Flow Sheets," *Ind. Eng. Chem. Res.* 26,1869 (1987)

Kocis, G.R. and I.E. Grossmann, "A Modeling and Decomposition Strategy for the MINLP Optimization of Process Flowsheets," *Comput. chem. Engng.* 13, 797 (1989b)

Kokossis, A. C. and C. A. Floudas, "Stability in Optimal Design: Synthesis of Complex Reactor Networks", AIChEJ., 40(5), pp. 849-861 (1994).

Lien. K. M. And WahL P. E. "If you cant beat them, join them. Combine Artificial Intelligence and Operations Research Technique* in Chemical Process Systems Design." Proceedings of PSE91, VoLIV. pp.1.1-15, MoMBbeUo. Canada (1991).

Liu. W. B. and C A. Floudas. "A Remark on the OOP Algorithm for Global *Optimiuaaon", Journal of Global Optimisation,* VoL 3. No. 4. pp. 519-522 (1993).

Lovdand, D.W., "Automated Theorem Proving: A U>gkal Baas." North Hollaod, Amsterdam (1978).

Mahalec. V. aid Motard, R. L.. "Procedures for the initial design of chemical processing systems." *Computers and Chemical Engineering, 1,51-6** (1977).

MaMurioatlialrivM. and D. Soi»1s«, "A Gtobal O p ^ization Approach to Rationally Constrained Rational Programming". Chem. Eng. Comm., 115.127-147 (1992).

Mannas, C D. and C. A. Floudas, "A Global Optimization Approach for Lennard-Jones Microclusters", J. Chem. Phys.. 97(10). pp. 7667-7678 (1992).

Mannas. C. D. and C. A. Floudas, "Global Optimization for Molecular Conformation Problems". *Annals of Operations Research,* VoL 42, pp. 85-117

Mannas, C. D. and C. A. Floudas, "Global Minimum Potential Energy Conformations of Small Molecules". *Journal of Global Optimization,* VoL 4. No. 2,135-170 (1994a).

Mannas, C. D. and C. A. Floudas, "A Deterministic Global Optimization Approach for Molecular Structure Determination". *Journal of Chemical Physics,* 100.2. January 15. pp. 1247-1261 (1994.b).

Mannas, C. D. and C. A. Floudas. "A Global Optimization Method for Weber's Problem with Attraction and Repulsion". *Large Scale Optimisation: Slate of the Art* (W. W. Hager. D. W. Heam, and P. M. Pwdalos, Editors), Kluwer Academic Publishers, pp. 265-300 (1994.c).

Mannas, C. D., I. P. Androulakis, C. A. Fkwdas, A. J. Berger, and J. M. Mulvery, "Solving Stochastic Control Problems in Finance via Global Optimization", submitted for publication, (1994).

McDonald, C. M. and C. A. Floudas, "Global Optimization for the Phase and Chemical Equilibrium Problem: Application to the NRTL Equation", Computers and Chemical Engineering, accepted for publication (1994a).

McDonald, C. M. and C. A. Floudas, "Global Optimization for the Phase Stability Problem", AIChE J., accepted for publication (1994b).

Mendelson, E., "Introduction to Mathematical Logic", Van Nestrand, New York. 1987.

Nemhauser, G. L. And Wolsey, L. A.,"Integer and Combinatorial Optimization," Wiley-Inierscience, New York (1988).

Nithida, N., G. Stephanopoutos and A.W. Westerberg, "A Review of Process Synthesis," *AlChEJ.* 27,321 (1981)

Papoulias, SA. and LE. Grossmann, "A Structural Optimization Approach in Process Synthesis. Pan I: Utility Systems," *Comput. chem. Engng.* 7,695 (1983a)

Papoulias, S.A. and I.E. Grossmann, "A Structural Optimization Approach in Process Synthesis. Part II: Heat Recovery Networks," *Comput. chem. Engng.* 7,707 (1983b)

Papoulias, S.A. and LE. Grossmann, "A Structural Optimization Approach in Process Synthesis. Part III: Total Processing Systems," *Comput. chem. Engng.* 7,723 (1983c)

Pardalos, P. M. and Rosen, J. B., "Constrained Global Optimization: Algorithms and Applications", *Lecture Notes in Computer Science,* Springer-Veriag, Berlin, Germany, VoL 268,1987.

Post, S. "Reasoning with Incomplete and Uncertain Knowledge as an Integer Linear Program," *Proceedings OfAvignon 87: Expert Systems and their Applications.* Avignon, Prance (1987).

Quesada, I., and LE Grossman, "Global Optimization Algorithm focHeat Exchanger Networks", *Ind. Eng. Chem. Res.,* 32, 487-499 (1993).

Quesada, I. and UL Grossmann, "A Global OptimizationAlgorithm for Linear Fractional and Bilinear Programs", Presented at HMS/ORSAMeeting, Chicago (1993a).

Quesada, I. and LE. Giossmann, "Global Optimization Algorithm for Heal Exchanger Networks," *Ind. Eng. Chem. Research,* 32.487 (1993b).

Quesada, L and LE. Grossmann, "Global Optimization Algorithm of Process Networks with Multicomponent Flows," presented at AIChE Meeting, St. Louis (1993c).

Raman, R. and LE. Giossmann, "Relation Between MILP Modelling and Logical Inference for Chemical Process Synthesis," *Computers andChemical Engineering* 15,73 (1991).

Raman, R. and LE. Grossmann, "Integration of Logic and Heuristic Knowledge in the MINLP Optimization for Process Synthesis," *Computers and Chemical Engineering* 16,155-171 (1992).

Raman, R. and LE. Grossmann, "Symbolic Integration of Logic in Mixed Integer Linear Programming Techniques for Process Synthesis," *Computers and Chemical Engineering,* 17,909 (1993*).

Raman, R. and IJE. Giossmann, "Symbolic Integration of Logic in MILP Branch and Bound Methods for the Synthesis of Process Networks," *Annals of Operations Research,* 42,169-191 (1993.b).

Raman, R. and IJE. Grossmann, "Modeling and Computational Techniques for Logic Based Integer Programming," Computers and Chemical Engineering, 18,563 (1994).

Rardin, R. L. And Choe, U. "Tighter Relaxations of FixedCharge Network Flow Problems," Georgia Institute of Technology, Industrial and Systems Engineering Report Series, #J-79-18, Atlanta (1979).

Ratschek, H., and Rokne, J., "New Computer Methods for Global Optimization", Halsted Press, Chichester, Great Britain, 1988.

Rinnoy Kan A. H.G.andG.T.Timmer, "Stochastic global optimization methods. Parti: clustering methods", *Math. Program,* 39,27 (1987).

Rippin, D.W.T., "Introduction: Approaches to Chemical Process Synthesis," in *Foundations of Computer-Aided Design,* JJ. Siirola, I.E. Grossmann and G. Stephanopoulos (Eds.), Cache-Elsevier, Amsterdam (1990).

Ryoo H.S. and N.V. Sahinidis, "Global Optimization of Nonconvex NLPs and MINLPs with Applications in Process Design", *Computers and Chemical Engineering,* submitted for publication (1994).

Sahinidis, N.V. and *IE.* Grossmann, "Convergence Properties of Generalized Benders Decomposition," *Computers and Chemical Engineering,* 15,481 (1991).

SCICON, "SCICONIC / VM User Guide," SCICON Ltd., London (1986).

Shendi, HL, and Ttacbikk, C. H., "A Global Optimizatk* Algorithm for 1 ^ ^omial Programming Problems Using a RcfonnuUtion-Lincarizauon Technique", *Journal of Global Optimization,* VoL 2, pp. 101-112,1992.

Shendi, R D. *mA* A. Alameddine," A New Refonnulaticm Linearization T e ^ ^ue for Bilinear Programming Problems", *J. Of Global Optimization,* VoL 2, No. 4, pp. 379-410 (1992).

Star, N. Z., "Dual Quadratic Estimates in Polyromial aral Boolean Progran^ing", *Annals of Operations Research,* VoL 25, pp. 163-168,1990.

Stephanopouk*, O. and A. W. Westoberg, "The UseofHegtenes* Method of Multipliers to Resolve Dual Gaps in Engineering System Optimization", *Journal of Optimization Theory and Applications,* 15(3), 285-309(1975).

Swaney, R. E.,$_t$ "Global Solution of Algebraic Nonlinear *Prognms^m,AIChE Annual Meeting, Chicago, 11* (1990).

Torn, A., aod A. Zilinskas, "Global Optimization", *Lecture Notes in Computer Science,* 350, Springer-Veriag, Berlin (1989).

Torn, A., and Zilinskas, A., "Global Optimization", *Lecture Notes in Computer Science,* Springer-Veriag, Berlin, Germany, VoL 350, 1989.

Tsirukis, A. G. and G. V. Reklaitis, "Feature Extraction Algorithms for Constrained Global Optimization I. Mathematical Inundation", A ^ ^ *of Operations Research,* 42, 229, (1993).

Tsimkis, A. G. and G. V. Reklaitis, "Feature Extraction Algorithms for Constrained Optimization II. Batch Process Scheduling Application**, *Annals of Operations Research,* 42,275 (1993).

Turkay, M. and I.E. Grossmann, "A Logic Based Outer-Approximation Algorithm for MINLP Optimization of Process Flowsheets", to bepresented AlChE Annual Meeting, San Francisco (1994).

Tuy, H., Thieu, T. V., and Thai, N. Q., "A Conical Algorithm for Globally Minimizing a Concave Function over a Closed Convex Set", *Mathematics of Operations Research,* VoL 10, pp. 498-514,198S.

Tuy, H., "Global Minimum of a Difference of Two Convex Functions", *Mathematical Programming Study,* Vol. 30, pp. 150-182, 1987.

Vaidyanathan R. And M. El-Halwagi, "Global Optimization of Nonconvex Nonlinear Programs via Interval Analysis", Computers and Chemical Engineering, in press (1994).

Viswanathan, J. and IJE. Grossmann, "A Combined Penalty Function and Outer-Approximation Method for MINLP Optimization," *Comput. chem. Engng.* 14,769 (1990)

Visweswaran, G, and C A. Floudas, "A Global Optimization Algorithm (GOP) for Certain Classes of Nonconvex NLPs - II. Application of Theory and Test Problems", *Computers and Chemical Engineering,* 14 (12), 1419-1434 (1990).

Visweswaran, V., and Floudas, C. A., "Unconstrained and Constrained Global Optimization of Polynomial Functions in One Variable", *Journal of Global Optimization,* VoL 2, pp. 73-100,1992.

Visweswaran, V. and C. A. Floudas, '[New Properties and Computational Improvement of the GOP Algorithm for Problems with Quadratic Objective Function and Constraints", *Journal of Global Optimization,* Vol. 3, No. 4, pp. 439-462 (1993).

Visweswaran V. and C. A. Floudas, "An MILP Reformulation of the Relaxed Duals in the GOP Algorithm", manuscriopt in preparation (1994).

Algorithmic Approaches to Process Synthesis: Logic and Global Optimization

Christodoulos A. Floudas, Ignacio E. Grossmann

EDRC 06-174-94

Algorithmic Approaches to Process Synthesis: Logic and Global Optimization

Oiristodoulos A. Floudasl and Ignacio E. Grossmann^
[1]Depertment of Chemical Engineering, PriiK^ton Univwsity, Princctoiu NJ. 08544
2 Dcpwtment of Chemical Engineering, OroegieMdlon University, Pittsburgh, PA 15213

"In memory of Professor David W. T. Rippin whose woit in Process Systems Engineering has been a source of inspiration fior us and many other researchers."

ABSTRACT

This paper presents an overview on two recent developments in optimization techniques that address previous limitations that have been experienced with algorithmic methods in process synthesis: combinatorics and local optima. The first part deals with the development of logic baaed models and techniques for discrete optimization^ich can facilitate the modelling of these problems as well as reducing the combinatorial search. It will be shewn that various levels can-1* considered for the imegnttion of logic in mixed-integer optimization techniques. The second part deals with the development of deterministic optimization methods that can rigorously determine the global optimum in nonconvex optimization models. It will be shown thai this can fee effectively accomplished with algorithms that exploit identifiable nonlinear structures. Examples are presented throughout the paper and future research directions are also briefly discussed

INTRODUCTION

Process synthesis continues to be a major area of research in process systems engineering. Significant advances have been achieved in terms of developing synthesis methods for subsystems (reactor networks, separation systems, heat exchanger networks) and for total flowsheets. Earlier reviews on general developments can be found in Hendry, Rudd and Seader (1973), Mavacek (1978) and in Nishida, Stephanopoulos and Westerberg (1981). A review on algorithmic methods based on MINLP was given by Grossmann (1990a) at the previous POCAPD meeting in Snowmass. A recent review and trends in MINLP based methods were recently presented by Grossmann and Daichendt (1994) at the PSE94 meeting in Korea. As for the synthesis of subsystems, reviews have been given by Gundersen and Naess (1988) on heat exchanger networks, and by Westerberg (1985) and Floquet, Pibouleau and Domenech (1988) on separation systems. From these reviews it is apparent that some of the major trends in the synthesis area include an increasing emphasis on the use of algorithmic methods that are based on MINLP optimization and their combination and integration with other design methodologies.

It is important to note that from a practical point of view a major motivation behind algorithmic techniques is the development of automated tools that can help design engineers to systematically explore a large number of design alternatives. From a theoretical point of view a major motivation is to develop unified representations and solution methods. Given the clear progress that has been made in the last decade in algorithmic techniques, and given the advances that have taken place in optimization and computer technology, the debate of heuristics or physical insights vs. mathematical programming has become largely irrelevant. It has generally become clear that a comprehensive approach to process synthesis will require a combination or integration of the different types of approaches. It has also become dear that significant work and progress are still required in the underlying methods that support each of these approaches. It is precisely this issue that is considered in this paper in the context of algorithmic methods.

This paper concentrates in two fundamental areas of optimization techniques that are used to support algorithmic methods in process synthesis. Specifically, we present an overview of two major advances that have recently taken place: (a) die incorporation of logic in mixed-integer optimization methods to reduce the combinatorial search and to facilitate problem formulation; (b) the development of rigorous global optimization techniques that can handle nonconvcxities in the model and avoid getting trapped in suboptimal solutions. These advances have been largely motivated by two major difficulties that have been encountered in the solution of MINLP models for process synthesis: combinatorics and local optima. The former are due to the potentially large number of structural alternatives that arise in process synthesis; the latter are due to the nonconvcxities that arise in nonlinear process models. The negative implication in the former is often the impossibility of solving large synthesis models; the negative implication of the latter is generating poor suboptimal designs.

While new developments are still under way, a review of the progress achieved up to date in logic based methods and in global optimization would seem to be timely as this might hopefully promote further research work. These algorithmic techniques are also significant in that they can be applied to other areas such as process scheduling and process analysis. The paper is organized as follows. We first discuss general aspects of process synthesis to see how the work described in this paper fits in the overall scheme. We next present a motivation section to illustrate difficulties in existing algorithmic methods with combinatorics and nonconvexities. The remaining part of the paper then concentrates in providing the overview of the new developments in logic and global optimization. Finally, we present the conclusions where we indicate future directions for research.

GENERAL COMPONENTS OF PROCESS SYNTHESIS

Algorithmic methods in process synthesis are rather general in scope and they involve the following four major components: (a) *Representation of space of alternatives;* (b) *General solution strategy;* (c) *Formulation of optimization model; (Si Application of solution method.*

The representations can range from rather high level abstractions such as is the case of targeting methods, to relatively detailed flowsheet descriptions such as is the case of superstructure representations. It is important to note that these representations are in fact commonly closely related as their difference lies in the level of abstraction that is used.

Having developed a representation, the next step to consider is the general solution strategy. The two common and extreme solution strategies are the simultaneous and the sequential approaches. The simultaneous strategies attempt to optimize simultaneously all the components in a synthesis problem in order to properly capture all the interactions and economic trade-offs. While conceptually superior, these strategies may give rise to larger problems. The sequential approach on the other hand has the advantage of dealing with smaller subproblcms since they sequentially decompose the problem, although often at the expense of sacrificing optimality.

The nature of die optimization models is of course heavily dependent on the type of representation as well as on the general solution strategy being used Target models often involve only continuous variables since they usually do not generate topologies nor do they consider capital cost as they deal with higher level objectives (minimize utility consumption, maximize yield). Therefore, these models commonly give rise to linear (LP) or nonlinear programming (NLP) problems. At the other extreme superstructure models determine topologies and operating conditions, and account for capita! costs, often requiring 0-1 and continuous variabte giving rise to nuxed-intcgerUnear (MILP) or mixed-integer nonlinear (MINLP) optimization models. Wiftin each of the levels of itfxeseittationtte degree of rigorousness of the model can of course also range from the simpler short-cot models to detailed simulation models.

As for the solution methods a global optimum solution can be guaranteed if the problem can be posed as an LP or MILP problem. Furthermore, in the case of LP models efficient solution times can be expected since these problems are theoretically solvable in polynomial time. This is however not the case of the MILP problems which generally are NP-complete, and therefore may have exponential time requirements, at least in the worst case. If the problem is posed as an NLP or MINLP the first drawback is that a unique global solution can only be guaranteed if the NLP or the continuous relaxation of the MINLP are convex. This is of course only a sufficient condition. But nevertheless, nonconvexities are prevalent in synthesis problems, often giving rise to multiple local solutions, or in fact even preventing convergence to feasible solutions with conventional NLP techniques. In addition to the numerical and theoretical difficulties of handling nonconvex models, there is the added difficulty of potential combinatorial explosion for the MINLP case. In the context of process synthesis a good example of the dilemma between the use of MDLP and MINLP models are the approaches for superstructure optimization of flowsheets by Papoulias and Grossmann (1983) and by Kocis and Grossmarm (1989). The advantage of the former is that the global optimum can be guaranteed but at the expense of using a discretized and approximate process model. The advantage of the latter is that nonlinear process models can be explicitly handled, but with the disadvantage that the global optimum cannot be guaranteed.

Based on the above discussion, it is clear that in order to properly support the development of algorithmic techniques, whether for targeting or superstructure models, or for simultaneous or sequential approaches, it is imperative that limitations due to combinatorics and nonconvexities be addressed. It is in this context that the two motivating examples below are presented.

MOTIVATING EXAMPLES

MILP Model for Heat Integrated Distillation Sequences

In order to illustrate potential combinatorial difficulties with synthesis problems, consider the MILP model reported in Raman and Grossmann (1993a) in which hem integration is considered between different separation tasks in the synthesis of sharp distillation sequences (see also Andrecovich and Westerberg (1985) and Floudas and Paulcs (1988)). An example of a superstructure for 4 components is given in Fig. 1 . For the heat integration part, it is assumed that the pressures of the columns can be adjusted in such a way that the condenser of every column can potentiaUy supply heat to the reboilers of the other columns as shown in Fig. 2 (multieffect columns are not considered). The MILP model involves as 0-1 variables the potential existence of columns and the potential beat exchanges between columns and reboilers, and as continuous variables the flows, heat loads and temperatures of condensers and reboilers, with which pressure changes arc accounted for. The objective function consists of die minimization of the investment cost of the columns and the operating cost for the utilities. The constraints involve mass and heat balances, and logical constraints that enferce feasible temperatures if heat exchange take pla^ and zero flows and heat loads if the corresponding 0-1 variables are 9et to zero.

For a four component system such as the one in Fig. 1 the MILP model involves 100 0-1 variables, 191 continuous variables and 258 constraints. The 100 binary variables are split as follows - 10 to model the existence of the distillation columns and 90 to model the existence of heat exchange matches between the reboilers and condensers of the various columns. The computer codes ZOOM, OSL and SCICONIC were tried for solving this problem. The three of them were not able to even find a feasible solution after enumeiating mem than 100,000 nodes and after running more than 1 CPU hour on an IBM RISC/6000! A major reason for this performance was that the relaxation gap is very large in this problem; the LP relaxation in which the binary variables are treated as continuous the optimum is Sl,117,000/yr. while the optimal MILP solution is $l,900,000/yr. As will be shown later in the paper, by using logic rigorous optimization of this problem can be achieved in only few seconds!

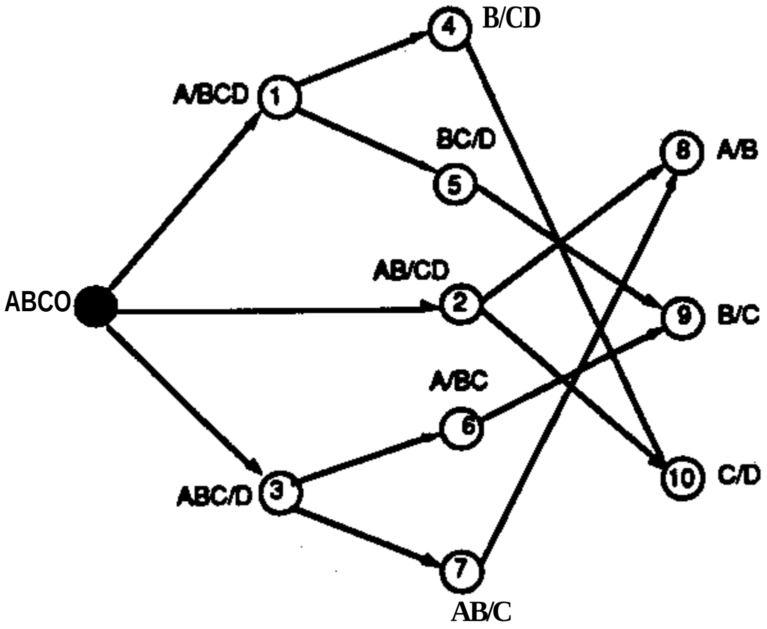


Fig. 1. Superstructure for 4-compoDent example.

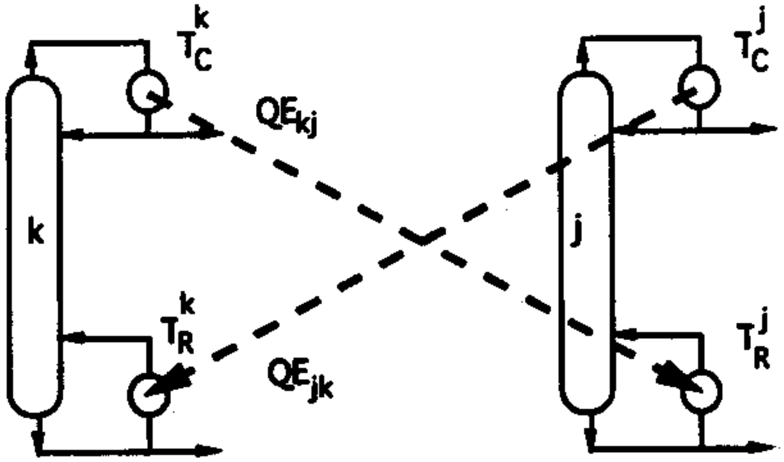


Fig. 2. **Heat integration between different separation tasks.**

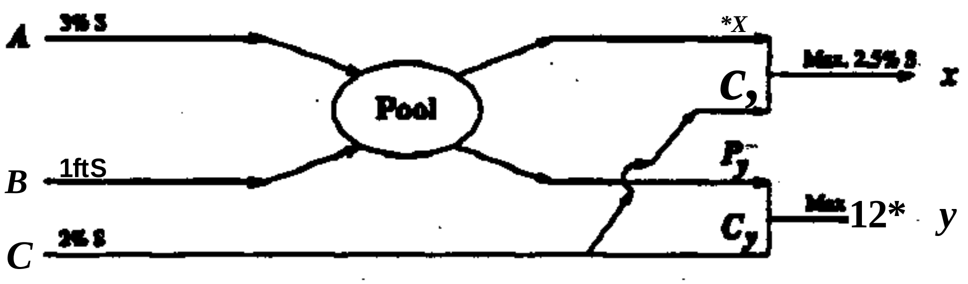
Nonconvex Model for Pooling/Blending Problems

To illustrate the potential difficulties associated with the existence of multiple solutions in nonlinear optimization NLP problems, we will consider as motivating example the pooling problem proposed by Haverly (1978) which is shown in Figure 3. Three crudes A, B, **and** G with different sulfur contents are to be combined to form two products x and y which have specifications on the maximum sulfur content Note that streams A and B are mixed in a pool and it is the existence of such a pool that introduces non-convexities in the mathematical model in the form of bilinear terms between the sulfur quality of the streams exiting the pool, denoted as p, and flowrates P_x, Py of the pool exiting streams. The objective in this pooling problem is to maximize the profit subject to (i) linear overall and component balances, (ii) bilinear pool quality and product quality constraints, and (iii) bounds on the products and sulfur quality. This problem has been studied using several local nonlinear optimization algorithms which have been reported to either obtain suboptimal solutions or fail to obtain even a feasible solution (see Floudas and Aggarwal, 1990 for a review of previous approaches and a decomposition strategy which alleviates but does not eliminate the multiplicity of local solutions problem). Table 1 presents results of local optimization algorithms (e.g. MINOS) for several starting points.

Table 1: Local Optimization for **the** Pooling Problem

		Solution Found	
No.	Sianins Quality	Objective value	Quality P
1	1.00	-750.0000	1.50
2	1.25	-750.0000	1.50
3	1.50	-750.0000	1.50
4	1.75	0.0000	1.75
5	2.00	0.0000	2.00
6	2.25	-125.0000	2.50
7	2.50	-125.0000	2.50
8	2.75	-125.0000	2.50
9	3.00	-125.0000	2.50

Figure 3: Motivating Example (Pooling Problem)



Formulation

rnln &4 + 13B + 1O(C. + *C,)* -9*x* - I5y

8.t.

$P_x + P_y - A - B = 0$ } pooffcofonce

z-F$_m$-P. -C. - 0. 1 *componentbalance*

V

p.(P> + P_y) - *SA* - *B m* 0 } *pool quality*

$p.P_x + 2.C_x - 2.5x \leq 0$ } *product quality*

$p.P_y + 2.C_y - 1.5y \leq 0$ *con&traini**

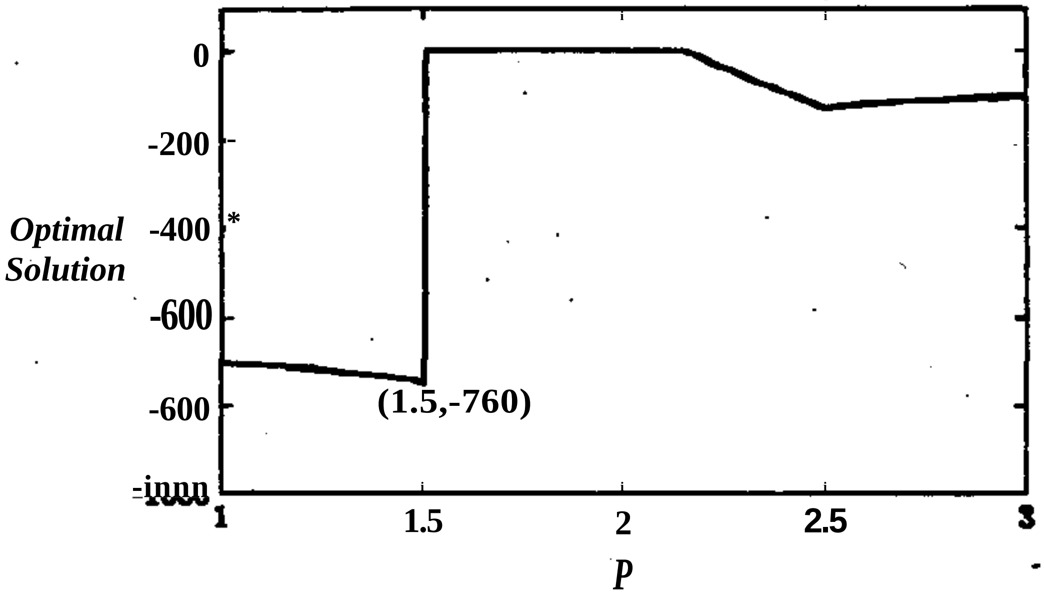
m $\leq$ 100 } *Upper bounds an*

y $\leq$ 200 *products*

1 $\leq$ P $\leq$ 3 I *Bound* on sulfur quality*

6

Figure 4; Optimal Solution in Projected Space



The non-convex nature of this pooling problem is better illustrated via Figure 4 where the optimal solution of the pooling model is shown for different values of the of the pool quality p. Note that the global optimum occurs at p « 1.5, while there exists a local optimum at p * 2.5 and between p • 1.S and p =2.2 (approximately) the optimal solutions are of the form of constant line. As a result, several starting points for p in the flat region or the region close to the local optimum terminate with the local solution or even fail to obtain a solution.

Floudas and Visweswaran (1990) applied the decomposition global optimization approach GOP, which is discussed in the global optimization section, to this pooling problem, as well as large instances of other pooling problems and multiperiod tankage problems (see also Visweswaran and Floudas, 1993) where the global optimum is obtained regardless of the starting point

INTEGRATION OF LOGIC IN MIXED-INTEGER PROGRAMMING

In this section we present a brief review of previous work on the modelling and solution techniques of logic based discrete optimization. We also review basic concepts for the representation of logic and inference problems. We then describe our recent work at Carnegie Mellon on the integration of logic in mixed-integer optimization which has been primarily motivated by process synthesis problems.

Review of Previous Work

A major issue in the application of mixed-integer programming is the efficient modelling of discrete decisions. Different representations are often possible for the same model, each of which may be solvable with varying degrees of difficulty. In some cases it is possible to even formulate an MH-P problem so that it is solvable as an LP, or else, so that its relaxation gap is greatly reduced. While some basic understanding has been achieved on how to properly formulaic special classes of mixed-integer programs (see Rardin and Choe, 1979; Nemhauser and Wolsey, 1988), the modelling of general purpose problems is largely performed *on an ad hoc* basis. The use of prepositional logic, however, offers an alternate framework for systematically developing mixed-integer optimization models as discussed by Jeroslow and Lowe (1984) and by Williams (1988).

The role of logic at the level of modelling of discrete optimization problems has also been studied by Balas (1974, 198S) who developed Disjunctive Programming (DP) as an alternate representation of mixed-integer programming problems. In this case, discrete optimization problems are formulated as linear programs in which a subset of constraints is expressed through disjunctions (sets of constraints of which at least one must be true). An interesting feature in the disjunctive formulation is that no 0-1 variables are explicitly included in the model, which is the more natural form to model some problems as, for instance, in the case of jobshop scheduling problems. Also, as noted by Balas (198S), every mixed-integer problem can be reformulated as a disjunctive program, and every bounded DP can be refonnulated as a mixed-integer program. The reason the disjunctive programming formulation has not been used more extensively is that very few methods have been proposed to explicitly solve the problem in that form. Most of die research has focused on characterizing the convex hull of disjunctive constraints and on the generation of strong cutting planes which are included in the corresponding mixed-integer problem to strengthen the LP relaxation (Balas, 1985; Jcroslow and Lowe, 1984). The only reported method, to our knowledge, that explicitly solves problem is the algorithm by Beaumont (1991) for the case where the functions are linear and there is only one constraint in each term of every disjunction. The method is similar to a branch and bound search except that the benching is done directly on the disjunctions. This requires the addition and deletion of the correspoiHiingdis^tive constraints in the LP subproblems. Although this may increase the overhead in the computations, Beaumom showed that the number of note required for the enumeration of the branch and bound tree can often be significantly reduced.

In terms of integrating logic explicitly for improving the solution efficiency of mixed-integer programs, aside from our own work which will be described in the next section (Raman and Grossmann, 1991,1992,1993a*, 1994), Lien and Whale (1991) considered the addition of a subset of unit resolution cuts for the branch and bound solution of MILP problems which produced large reductions of enumeration of nodes in the MILP formulation for heat integrated synthesis by Andrecovkh and Westerberg (1985). It should also be mentioned that logic has been considered earlier in process synthesis with the purpose of performing high level decisions in the structuring of process flowsheets (Mahalec and Motard, 1977).

Representations of the logic

Most of the work described above has been restricted to the form of logic calkd prepositional logic for developing modelling and solution techniques for discrete optimization problems (see Menddson, 1987, for general review on logic). The basic imit of a propositkmallogte express*^ which can correspond to a state or to an action, is called a literal which is a single variable that can assume either of two values, true or false. Associated with each literal J\ its negation NOT *Y* (-. *Y)* is such that *[Y* OR -ill is always true. A disjunctive clause is a set of literals separated by OR operators [v], and is also called a disjunction. A proposition is any logical expression and consists of a set of clauses />;, i»l,../ that are related by the logical operators OR [v], AND [A], IMPLICATION *[=>].*

In synthesis logic propositions usually refer to relations of existence of units in a superstructure. These are commonly expressed by a set of conjunctions of clauses,

Am{LjAL2A. . . . . . ALg} (1)

where Li is a logical proposition expressed with boolean variables Kj in terms of implications, OR, EXCLUSIVE OR and AND operators. In synthesis problems *Y* is a boolean variable representing the existence of unit i and -IYJ its nonexistence. There are two ways of transforming the propositions in A. In the simplest case, the logic propositions are converted into the conjunctive normal fonn [CNF] by removing the implications through contrapositions in each of the clauses Lj in (1) and applying De Morgans

Theorem. In this way each clause in the CNF from consists of only OR operators with non-negated and negated boolean variables as follows:

Ω_C=[$_{i\in P_1}^{\vee}$ 0!)$_{i\in}$ v $_{\bullet}$ W] A [v ft) v <-r,)]A...*r v (R) A <-r,)] (2)

where Pi and $\overline{\text{P}}$ i are subsets of the boolean variables that correspond to some of the 0-1 variables, and s is the number of clauses.

In the second representation, the logic propositions in the CNF form are convened into the disjunctive normal form [DNF] (see Clocksin and Nfellish, 1984) by moving the AN^ operators inwards and the OR operators outwards by applying elementary boolean operations. The DNF form is as follows:

^{U}D [i*Ql left \ [$^{i€}$Ql ieGl J l'«ft> '«5 J W

where Qj and $\overline{\text{Qj}}$ are the index sets of the boolean variables which correspond to a partition of all the 0-1 variables y,\ i=7,../> in non-negated and negated terms. Each clause separated by a disjunction represents the assignment of units in a feasible configuration* where it is assumed that each boolean variable has a one-to-one correspondence with the 0-1 binary variables of the MEP model. Therefore, r represents the number of alternatives in the superstructure. While the DNF form is more convenient to manipulate, the drawback is that the transformation from CNF to DNF has exponential complexity in the worst case.

To illustrate the CNF and DNF representations in (2) and (3), consider the small example problem shown in Fig.5. The following prepositional logic expressions apply:

I4: *Y* v *Y2* => 1'3 (process l or process 2 imply process 3)

^{L}2: 1^3 =* ^{y}1 ^{v} ^{Y}*2* (process 3 implies process 1 or process 2)

L3: -*tY* v -.^2 (do not select process l or do not select process 2)

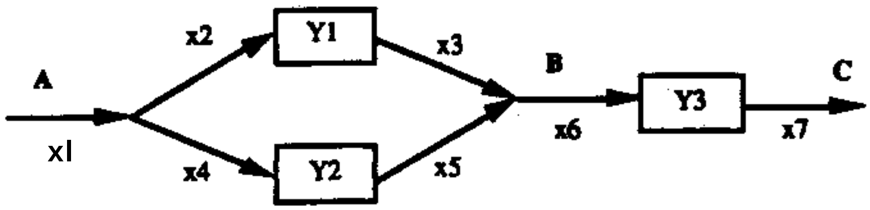


Fig 5. Superstructure for small example.

Applying the contrapositive to Li and L2, and using De Morgan's theorem, the corresponding CNF representation1 for the logic is:

Ω_C = *Hi* v *Y3)* A (-.72 v K3) A (-^3 v *Y* v *YD* A (-.ri v ¬*Y*2) (4)

Distributing the OR over the AND operators, the corresponding DNF representation is given by:

OD * <*T* A -^2 A *Y^)* V (72 A -iFl A *Y3)* V (*-JTl* A -n72 A -1X3) (5)

Note that the disjunctions in (5) represent the three alternatives in Fig 5.

In order to obtain an equivalent mathematical representation for any prepositional logic expression, this can be easily performed using the CNF form as a basis. We must first consider basic logical operators to determine how each can be transformed into an equivalent representation in the form of an equation or inequality. These transformations are then used to convert general logical expressions into an equivalent mathematical representation (Cavalier and Soyster, 1987; Cavalier***al*, 1990).

Table 2. Representation of logical relations with linear inequalities

Logical Relation	Comments	Boolean Expression	Representation as Linear Inequalities
Logical OR		Pi v P2 v .. v Pr	yi + y2 + •• • yr * 1
Logical AND		Pi A P2 A .. A Pr	$y_1 \geq 1$ $y_2 \geq 1$ .. $y_r \geq 1$
Implication	$P_1 \Rightarrow P_2$	-.P1VP2	1 - yl + y2 * 1
Equivalence	Pi if and only if P2 (Pl=*P2)A(P2=>Pl)	(-Pi VP2)A(-,P2v Pl)	y i « y 2
Exclusive OR	exactly one of the variables is true	Pi ⊻ P_2 ⊻ .. ⊻ Pr	$y_1 + y_2 + .. + y_r = 1$

To each literal P_v a Unary variable $y\backslash$ is assigned. Then the negation or complement of/^ (-.Pj) is given by 1 - y_x. The logical value of true corresponds to the binary value of 1 and false corresponds to the binary value of 0. The basic operators used in prepositional logic and the representation of their relationships are shown in Table 2. With the basic equivalent relations given in Table 2 (e.g. see William's, 198S), one can systematically model an arbitrary prepositional logic expression that is given in terms of OR, AND, IMPLICATION operators, as a set of linear equality and inequality constraints. One approach is to systematically convert the logical expression into its eqtvalent *conjunctive normal form* representation which involves the application of pure logical operations. The conjunctive normal form is a conjunction of clauses, *Q* A C2 A ... A g_s. Hence, for the conjunctive normal form to be true, each clause *Q* must be true independent of the others. Also since a clause *Q* is just a disjunction of literals, P j v />2 v ... v P_r, it can be expressed in the linear mathematical form as the inequality.

37 • >2 +. . . . .• *yr* * *1* (6)

Symbolic and Mathematical Methods for Logic Inference

The most common logic inference problem is the satisfiability problem where, given the validity of a set of propositions, one has to prove the truth or validity of a conclusion which may be either a literal or a proposition. This inference problem is one of the basic issues in artificial intelligence and data bases. However, the general satisfiability problem for prepositional logic is NP-complete (Cook, 1971; Karp, 1972). Therefore, research has focused on identifying classes of problems within the general satisfiability problem that can be solved efficiently. Knowledge based systems commonly require the use of Horn clause systems which have at most one non-negated literal in each clause. The inference problem for this class of prepositional logic problems can be solved in linear time using unit resolution (Dowling and Gallier, 1984). The unit resolution technique (e.g. see Clocksin and Mellish, 1981) is one of the most common inference techniques, and in simple terms, it consists of solving sequentially each logic clause one at a time. Chandra and Hooker (1988) have extended the class of problems that can be solved in linear time to include extended Horn clause systems. One of the most effective logic-based methods for solving the general satisfiability problem is the algorithm of Davis and Putnam (1960) as treated by Loveland (1978). This approach is closely related to the branch and bound method for mixed-integer programming. Jereslow and Wang (1990) have developed branching heuristics to improve the performance of the Davis-Putnam procedure. It must be noted that although the previous work has been restricted to prepositional logic, the techniques used for this class are essential to higher order representations like predicate logic which involve additional logic operators like for all [V] and it exists **[3].**

Since the logical propositions can be systematically converted into a set of linear inequalities, instead of using symbolic inference techniques, the inference problem can be formulated as an integer linear programming problem. In particular, given a problem in which all the logical propositions have been converted to a set of linear inequalities, the inference problem that consists of proving a given clause,

$$\begin{array}{lll} \text{Prove} & P_u & \qquad\qquad \text{(UP1)} \\ \text{st} & B(P0 & i = U,..,q \end{array}$$

can be formulated as the following MILP (Cavalier and Soyster, 1987):

$$\begin{array}{lll} \textbf{Min} & Z« \underset{i \in I(u)}{X} \alpha_i y_i & \\ \text{st} & A y \;\; 2 \;\; a & \qquad\qquad \text{(UP2)} \\ & y \;\; € \;\; \{0J\}^n & \end{array}$$

where *Ay* *k* a is die set of inequalities obtained by translating *B* (*P\.P2** •• *f*q) into their linear mathematical form, and the objective function is obtained by also converting the clause P_u that is to be proved into its equivalent mathematical form. Here, /(u) corresponds to the index set of the binary variables associated with the clause P_u. This clause is always true if Z * 1 on minimizing the objective function as an integer programming problem. If Z = 0 for the optimal integer solution, this establishes an instance where the clause is false. Therefore, in this case, the clause is not always true. In many instances, the optimal integer solution to problem (LIP2) will be obtained by solving its linear programming relaxation (Hooker, 1988). Even if no integer solution is obtained, it may be possible to fetch conclusions from the relaxed UP problem (Cavalier and Soyster, 1987).

The qualitative knowledge available about the design of a system can be classified as one of the following two types - hard logical bets or uncertain heuristics. Hard, logical facts are never violated - for example, the reaction NaOH + HCl -~> N a C U H ^ holds from basic (Conical principles. Qualitative knowledge in the form of heuristics on the other hand are just rules of thumb which may not always hold. Therefore all the knowledge for synthesizing a design may not be consistent since the heuristics may contradict one another; for example, a rule that suggests to use higher temperatures to increase yield may conflict with a rule that suggests to use lower temperature to increase selectivity. Resolution of conflicts is an important pan of reasoning. In general one must violate a weaker (more uncertain) set of rules in order to satisfy stronger ones. Therefore, it becomes necessary to model the violation of heuristics, which is done as follows (Post, 1987),

$$Clause \quad or \quad V \qquad (7)$$

where either the clause is true or it is being violated (V). In order to discriminate between weak and strong rules, penalties are associated with the violation v| of each heuristic rule, i * U.jn. The penalty wj is a non-negative number which reflects the uncertainty of the corresponding logical expression. The more uncertain the rule, the lower the penalty for its violation. In this way, the logical inference problem with uncertain knowledge can be formulated as an MELP problem where the objective is to obtain a solution that satisfies all the logical relationships (i.e. Z * 0), and if that is not possible, to obtain a solution with the least total penalty for violation of the heuristics:

$$\begin{array}{lllll} Min & Z \ « \ w^T v & & & \\ st & Ay \quad Z \ a & : & \text{Logical facts} & \text{(UP3)} \\ & By \ \bullet \ v \ 2 \ b & : & \text{Heuristics} & \\ & y \ € \ \{0,l\}^n . \ v \ £ \ 0 & & & \end{array}$$

Note that no violations are assigned to the inequalities Ay 2B since these correspond to hard logical facts that always have to be satisfied. The solution to (UP3) will then determine a design that best satisfies the possibly conflicting qualitative knowledge about the system.

Logic-based Formulations for Discrete Optimization

Given a superstructure of alternatives for a given design problem, the general form of the mixed-integer optimization model is (Grossmann, 1990a),

$$\begin{array}{lll} Min & ZmJy+jlx) & \\ st & k(x) \quad Z \ 0 & \\ & g(x) + My \ £ \ 0 & \text{(DPI)} \\ & xe \quad X.ye \quad Y & \end{array}$$

where *x* is the vector of continuous variables involved in design like pressure, temperature and flow rates, while *y* is the vector of binary decision variables like existence of a particular stream or unit Integer variables might also be involved but these are often expressed in terms of 0-1 variables. Also, model (DPI) may contain among the inequalities pure integer constraints for logical specifications (e.g. select only one reactor type). If all the functions and constraints are linear (PI) correspoiKis to afrMIlP problem; otherwise it is an MINLP. For the sake of simplicity, we assume that/frj, *g(x)* and *h(x)* are convex, differentiate functions. The case of nonconvexities will be addressed later in the paper.

The mixed-integer program (DPI), is not the only way of modelling the discrete optimization problem in a superstructure. As has been shown by Raman and Grossmann (1994) that problem can be formulated as the generalized disjunctive program:

$$\mathit{Min} \quad Z = \sum_{i} \sum_{k} c_{ik} + f(x)$$

$$\mathit{st} \quad h(x) \text{ £ } 0 \qquad\qquad \text{(DP2)}$$

$$\bigvee_{i \in D_k} \begin{bmatrix} Y_{ik} \\ g_{ik}(x) \leq 0 \\ c_{ik} = \gamma_{ik} \end{bmatrix} \qquad ke\ SD$$

$$\Omega(Y) = True$$

$$x€R^n{}_9ceR^m,Y€ \quad \{true,false\}"^1$$

in which *Yfr* are the boolean variables that establish whether a given term in a disjunction is true *lgUc(x)* ≤ *0]* or false *[gUcfx)* > 0], while *Cl(Y)* are logical relations assumed to be in the form of prepositional logic involving only the boolean variables. *Yfc* are auxiliary variables that control the part of the feasible space in which the continuous variables, x, lie, and he variables c;* represent fixed charges which are activated to a value fik if the corresponding term of the disjunction is true. Finally, the logical conditions, *£l(Y),* express relationships between the disjunctive sets. In the context of synthesis problems the disjunctions in (DP2) typically arise for each unit i in the following form:

$$\begin{bmatrix} Y_i \\ g_i(x) \leq 0 \\ c_i = \gamma_i \end{bmatrix} \vee \begin{bmatrix} \neg Y_i \\ B^i x = 0 \\ c_i = 0 \end{bmatrix} \qquad\qquad (8)$$

in which the inequalities gi apply and a fixed cost n is incurred if the unit is selected (?;£ otherwise *(~*Y0* there is no fixed cost and a subset of the x variables is set to zero with the matrix B*. An important advantage of the above modelling framework is that there is no need to introduce artificial parameters for the "big-M" constraints that are normally used to model disjunctions.

An interesting question that arises with problem (DP2) is whether it always pays to convert the general disjunctive program into mixed-integer form. To answer this question for die case of linear functions and constraints, Raman and Grossmann (1994) have developed the concept of w-MIP represcntability which is defined asfollows:

Definition: The disjunction $\bigvee_{ieD_k} \backslash Aik^{x}\wedge^{b}id$ is ***w-MIP representable*** iff the following conditions hold:

(i) There exists an i€ Dk for which the convex bull of the disjunction is reducible to the constraint:

$$A_{ik}\, x \;\; 2\, btkytk \qquad\qquad 0 \;£\; ytk \;\leq 1$$

(ii) Every feasible solution

$$x'€F\text{-}\{xl \quad \bigvee_{ieD_k} [A_{ik}x \geq b_{i}\}$$

for which $Afkx^{*} \;\; 2$fcflfc,$Afcx' <$ &#, i * i implies that yft « 7 *andyik = 0 Vi *i*

Thus, in general, we can consider a partly transformed form of problem (DP2) where mixed-integer equations are used for the w-MIP constraints pan of the problem, while the rest is kept in disjunctive form, as this pan is "poorly-behaved" in equation form. In general, this partially reformulated problem has the form,

$$Min \quad Z « \sum_{k\in SD^1}\sum_{i\in D_k} \gamma_{ik} y_{ik} + \sum_{k\in SD^2}\sum_{i\in D_k} c_{ik} + f(x)$$

$$st \quad h(x) \leq 0 \qquad\qquad \text{(DP3)}$$

$$H^{*}) + By\; SO$$

$$Ay\; k \;\; a$$

$$\bigvee_{ieD_k} \begin{bmatrix} Y_{ik} \\ s_{ik}(x) \leq 0 \\ c_{ik} = Y_{ik} \end{bmatrix} \qquad\qquad ke\; SD^{2}$$

$$A(Y) « True$$

$$xelP.ye \;\; (Ojf. \;\; Y € \;\; \{true, false\}"^{1}$$

in which the subset of disjunctions SD^1 c SD, which are w-MIP representable, have all been convened into mixed-integer form. The inequalities r(x) + By £ 0 correspond to these constraints and to subsets of the inequalities gikfc cik) ^ 0, i € Dk, k € SD^2, which have also been converted into mixed-integer form. Finally, s& (x, cik) are the remaining inequalities which appear explicitly in the disjunctions k € SD^2.

Note also that a subset of the logical constraints in *Q(Y)* • True, which are required for the definition of the discrete optimization problem, have been translated to the form of linear inequality constraints Ay £ a. The simplest option is to convert the propositions into CNF which can then be translated readily into inequalities as was discussed in the previous section. In cases where the number of these constraints become large, the generation of a smaller number of tighter constraints through the application of cutting plane techniques may be useful. The rest of the logic constraints, A(K) « True, which are redundant and correspond to logic cuts that do not alter the optimal solution (Hooker et al, 1993), have been left in symbolic form in order to improve the enumeration in a branch and bound search.

It should be noted that a particular case of (DP3) of interest is when the entire problem is converted into mixed-integer form, but the logic cuts *A(Y)« True* are included as part of the formulation:

$$
\begin{array}{lll}
Min & Z = \hat{}\,\sum_{i=1}^{m} Y_i V_i + /(*) & \\
st & h(x) \le 0 & \text{(DP4)} \\
 & Hx) + By * 0 & \\
 & Ay\ 2\ a & \\
 & A(Y) = True & \\
 & x \text{€} R^{n}{}_{9} ye\ \ \{0.1\}^{m}.\ \ Y\ e\ \ \{true,\ false\}^{9}{}'' &
\end{array}
$$

Solution methods

As was mentioned in the review section there are still few methods for solving mixed-integer optimization problems that incorporate prepositional logic. As shown below, methods have been developed for addressing linear and nonlinear problems. Obviously some of the methods are equally applicable to both cases. However, for the sake of clarity, and to also emphasize the more useful methods in each case, we will distinguish between methods for linear and nonlinear problems.

For linear problems the simplest case is when logic cuts A(K) • True are added to an MDLP problem as in (DP4). These cuts, which represent redundant constraints in high level form, can be systematically generated for process networks as discussed in Raman and Grossmann (1993a). As an example, the logic cuts for the network in Fig. 1 in terms of the potential existence of the 10 columns are given by the propositions:

Yl = Y4v Y5*

Y2=>Y8AY10

Y3 =>Y6v Y7

Y4 =>Y1 A Y10

Y5=>Y1AY9

Y6=>Y3A Y9

Y7 = Y3 A Y8*

Y8=>Y2 vY7

*Y9=*Y5 v Y6*

*Y10=*Y2VY4*

There arc two bask ways of handling these cuts. One is to convert them into inequalities and add them to the MDLP (Raman and Grossmann, 1992). White this will increase the number of constraints, it generally reduces the relaxation gap. The other extreme is to process the logic symbolically as part of the branch and bound search for the MOP. In this case the logic is used to select the branching variables and to determine by inference whether additional Unary variables can be fixed at each node (Raman and Grossmann, 1993a,b). This can be accomplished by treating the logic either in CNF form as in (2) or in DNF form as in (3). The former requires unit resolution for the inference, whik the latter involves the solution of Boolean equations. Although the DNF form is generally more expensive to obtain, its nice theoretical property is that one con *guarantee that in the worst case the number of enumerated nodes does not exceed twice the number of clauses in (3) minus one* (see Raman and Grossmann (1993a) for proof). A third alternative is to use a hybrid approach in which only violated inequalities at the root node are included to strengthen the LP relaxation, but the remaining enumeration is performed by solving the logic symbolically.

For the case that the discrete optimization problem is formulated as in (DP3) by involving both disjunctions and mixed-integer constraints, Raman and Grossmann(1994) proposed an extension of the hybrid branch and bound method for (DP4) in which the disjunctions are converted for convenience into mixed-integer form, but the branching rule is altered to recognize the fact that no branching be performed on disjunctions that are logically satisfied, even if the corresponding 0-1 variables are non-integer. Note that such an algorithm can also be applied to problem (DP2). Finally, it is worth to mention that Beaumont (1991) has proposed an algorithm that applies to (DP2) in the case that only one equation is involved in each disjunction. In this algorithm constraints are successively added or deleted as needed in the branch and bound search.

Similarly as in the linear case, the simplest way to integrate logic in nonlinear discrete models is to add the logic cuts to an MINLP as in problem (DP4) (see Raman and Grossmann, 1992). If these are convened to inequalities this has the effect of reducing the relaxation gap. This has the important effect of strengthening the lower bound that is predicted by the master problem in the Generalized Benders decomposition method by Geoffrion (1972). As has been shown by Sahinidis and Grossmann (1991) the "optimal" formulation for the GBD method is when there is no gap between the relaxed and the integer optimum solution. In the case of the outer-approximation method by Duran and Grossmann (1986) the quantitative or symbolic integration has the effect of potentially reducing the branch and bound enumeration at the level of the MILP master problem. An interesting variation of the above idea is to integrate the logic inference problem with heuristics (UP3) in the MILP master problem as was proposed by Raman and Grossmann (1992). First assume that given the solution of K NLP subproUems the MILP master problem is represented by:

Min *a*
st *a* £ *4fa* j)
xj ^ ^ *= 1-JT (Ml)

x€jr,y€y

in which ^*x,y)* represents either the Lagrangian in the GBD method or an objective linearization in the OA method, £2* is the linear approximation to the continuous feasible space and INTfc represents integer cuts to exclude configurations that were previously analyzed. The integer programming model (LIP3) can be integrated in the above master problem(Ml) by minimizing the weighted violation (plus an extra term to reflect the cost) and subject to constraining the lower bound to the current upper bound; that is,

$$Min\ [w^T v + \bar{w}(a\text{-}LB)/(UB^*\text{-}LB)]$$

$$\begin{array}{ll} st\ \alpha \geq \phi_k(x,y) & \\ x,y \in \Omega_k & k = 1,\ldots,K \end{array}$$

y € *INT k*
Ay *2* *a*
By + v *Zb* (M2)
a £ *UB*^{k}
xeX.yeY
ae *\ v €{0,1}

in which $\bar{w}$ is a penalty chosen such that $\bar{w}$ « min* *(w%)* f LB is a valid lower bound to the solution of the MINLP (e.g.. solution to the relaxed NLP problem or some reasonable but valid bound) and UB^K is the current upper bound of the objective at iteration K. The interesting feature with the master problem (M2) is that optimality can still be guaranteed (within convexity assumptions) even though heuristics are used as part of the search. The master problem (M2) is especially appropriate for the GBD method because of the loose approximation that is obtained with that method. It is also important to note that the master problem (M2) can be used when applying Benders decomposition (Benders, 1962) in the solution of MILP problems.

For the case that the nonlinear discrete optimization problem is formulated as the generalized disjunctive program in (DP2) one can develop corresponding logic-based OA and GBD algorithms as described in Turkay and Grossmaim (1994). First, for fixed values of the boolean variables, Yfk * true and Y_{ik} = false, the corresponding NLP subproblem is as follows:

$$\begin{array}{ll} Min & Z = \sum_{i=1}^{m} c_{ik} + f(x) \\ st & h(x) \le 0 \end{array} \qquad \text{(SP)}$$

$$\left\{ \begin{array}{c} c_{ik} = \gamma_{ik} \\ g_{ik}(x) \le 0 \end{array} \right\} \text{\forYv*true} \qquad k \in SD$$

$$c_{ik} = 0 \;\; for\, Y_{ik} = false \;\; i \ne \hat{i}$$

$$x \in R^{n}, \; c_{ik} \in R^{m},$$

htote that oiu^ooosOTiirtsconespoiKiMg to true boolean variables are imposed. Also fixed charges-ft* are only applied to these terms. Assuming that K subproblems (SP) are solved in which sets of linearizations M...JC are generated for subsets of disjunction terms *L(ik) -{11 Y*ik ~ true }. ont can* define the following disjunctive OA master problem:

$$\begin{array}{ll} Min & Z = \sum_{i}\sum_{k} c_{ik} + \alpha \\ st & \alpha \ge f(x^{l}) + \nabla f(x^{l})^{T}(x-x^{l}) \qquad I\ -1..X \\ & «*> + Vhfxtftx-x^{1})£0 \end{array} \qquad \text{(MDP2)}$$

$$\bigvee_{i \in D_k} \left[\begin{array}{c} Y_{ik} \\ g_{ik}(x^{l}) + \nabla g_{ik}(x^{l})^{T}(x - x^{l}) \le 0 \;\; l \in L(ik) \\ c_{ik} = \gamma_{ik} \end{array} \right] \quad k \in SD$$

$$am- \;True$$

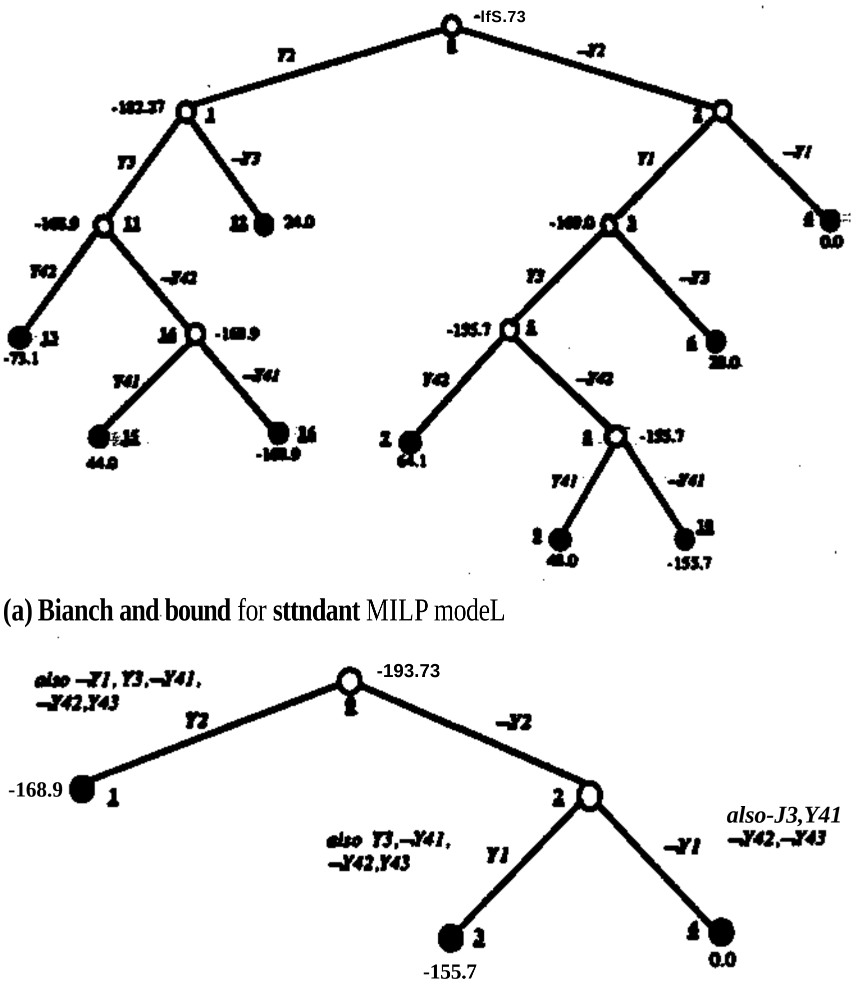
$$\alpha \in R,\; x \in R^{H}.t:€ \;\; R^{m}.Y \in \{true,\; false\}"$$

It should be noted that before applying the above master problem it is necessary to solve various subproblems so as to produce at least one linear approximation of each of the terms in the disjunctions. As shown by Turkay and Grossmann (1994) selecting the smallest number of subproblems amounts to the solution of a set covering problem. The above problem (MDP2) can be solved by any of the methods described for the linear case. It is also interesting to note that for the case of flowsheet synthesis problems Turkay and Grossmann (1994) have shown that the above solution method becomes equivalent to the modelling/decomposition strategy by Kocis and Grossmann (1988) if the master problem (MDP2) is converted into MEJ> form using a convex hull representation. Also, these authors have shown that while a logic-based GBD method cannot be derived as in the case of the OA algorithm, one can nevertheless determine for the MILP version of the master problem (MDP2) one Benders iteration which then yields a sequence similar to the GBD method for the algebraic case.

Computational Experience

From the methods described in the previous section the symbolic integration of logic both in DNF and CNF form have been automated in a special version of OSL, the MILP solver from IBM (Raman and Grossmann, 1993a). Also systematic methods have been developed to automate the generation of logic cuts in process networks (Raman and Grossmann, 1993a; Hooker et al$_f$ 1994). Work is also currently under way to automate the logic version of the OA and GBD algorithms.

In order to appreciate the potential impact of integrating logic in discrete optimization problems numerical results on selected examples are given in Table 3. Example (a) deals with an MILP for the synthesis of separation sequences involving 6 components (see Raman and Grossmann, 1992). Applying the standard version of Benders decomposition convci^eix^ is IKH achieved aftcrscvcralhoureaiKlnKrc than one hundred iterations on an older Vax-computer. In constrast, adding inequalities for the logic cuts in (DP4) convergence is achieved in only 13 iterations, and this despite the fact that the number of constraints is doubled. Note that the integrated master with heuristics is not as effective in this case. Example (b) deals with a small MINLP planning problem in which similar trends are observed when adding the logic cuts. The examples in (c) deal with the symbolic and hybrid integration of logic using branch and bound (see Raman and Grossmann, 1993). Note that for the MILP for the separation of 6 components the reduction in number of nodes enumerated is significant The more impressive results, however, are with the heat integrated model which corresponds to the motivating example. Adding the inequalities for the logic cuts the problem is solved to optimality in only 8 sec! And this is accomplished by almost doubling the number of constraints. With the symbolic integration of logic with DNF the time is even further reduced to less than 3 sec! The reason for the reduction is that in the symbolic integration there is no need to handle the inequalities for the logic cuts. It should be noted that the DNF logic involved 194 disjunctive terms. Therefore, theoretically it is possible to guarantee that the number of nodes in this type of enumeration will not exceed 387 nodes. In actual fact only 20 were needed. Finally, the example in (d) illustrates a problem in which a process network was initially formulated as the generalized disjunctive program (DP2) (see Raman and Grossmann, 1994). Converting it all into MILP form requires more than 1 hour of solution time with OSL. If instead the problem is formulated as in (DP3) in which disjunctions are identified that are not w-MIP representable the modified branch and bound method requires less than 10 minutes of CPU time. Fig. 6 presents the tree searches for a very small version of this problem. Note that even in this case the logic-based branch and bound for the disjunctive model (DP3) requires only 4 nodes as opposed to the 16 that are needed when the model is posed entirely as an MILP and solved with standard branch and bound methods.



(a) **Bianch and bound** for **sttndant** MILP modeL

(b) Logic based branch and bound for disjunctive model (DP3)

Fig. **6** . **Comparison of tree searches with standard and logic based branch and bound.**

Table 3. Computational Results on Selected Example Problems

(a) NflLP model 6 component separation. *Benders decomposition*

	Original Model (DPI)	Model with Logic (DP4)	Integrated Master (M2)
Constraints:			
Heuristic			187
Logic constraints		70	70
Other	86	86	86
Iterations	>100	**13**	**43**
Cpu-time*	>1000	**11.99**	**338.7**

'•min Micro-VaxD (SCICON1C)

(b) MINLP model planning **problem** ***Generalized Benders Decomposition***

	Original Model (DPI)	Model with logic **(DP4)**	Integrated Master (M2)
Heuristic constraints			**5**
Logic constraints	**1**	**8**	**8**
Other constraints	**9**	**9**	**9**
Number iterations	**7**	**3**	**4**
CPU time*	**28.20**	**11.7**	**18.8**

'•sec Micro-Vax D (SCICONIC/MINOS)

(c) MILP models. *Branch and bound*

	Original Model (DPI)	Model with logic (DP4)	**DNF based** approach	Hybrid DNF approach
Six components				
Logic constraints	0	70	0	11
no. of nodes	141	8	18	5
CPU time*	3.46	1.18	1.06	0.7
Heat Integrated Distillation				
Logic constraints	0	215	0	4
nodes	> 100,000	74	20	17
CPUtime*	> 5,000	**8.37**	2.76	2.62

'•secBM-RS6000(OSL)

(d) MILP Process Network with semi-continuous demands

	MILP model (DPI)	**Disjunctive Model (DP3)**
Constraints	**1382**	**1382**
Variables	**1326**	**1326**
Binary	73	**73**
Nodes	**16,532**	**1,771**
CPU time*	**76.2**	**8.3**

'•minlBM-RS6000(OSL)

GLOBAL OPTIMIZATION

Background

A significant effort has been expended in the last five decades toward theoretical and algorithmic studies of local optimization algorithms and their computational testing in applications that arise in Process Synthesis Design and CootroL Relative to such an extensive effort that has been devoted to local nonlinear optimization approaches, there has been much less work on the theoretical and algorithmic development of global optimization methods. In the last decade the area of global optimization has attracted a lot of interest form the Operations Research and Applied Matheniatics community, while in the last five yean we have experienced a resurgence of interest in Chemical Engineering for new methods of global optimization as well as the application of available global optimization algorithms to important chemical engineering problems. This recent surge of interest is attributed to three main reasons. First, a large number of process synthesis, design and, control problems are indeed global optimization problems. Second, the existing local nonlinear optimization approaches (e.g. generalized reduced gradient and successive quadratic programming methods) may either fail to obtain even a feasible solution or are trapped to a local optimum solution which may differ in value significantly form the global solution. Third, the global optimum solution may have a very different physical interpretation when it is compared to local solutions (eg. in phase equilibrium a local solution may provide incorrect prediction of types of phases at equilibrium, as well as the components' composition in each phase).

The existing approaches for global optimization are classified as deterministic or probabilistic. The deterministic approaches include: (a) Lipschitzian methods (e.g. Hansen et al. 1992 a, b), (b) Branch and Bound methods (e.g. Al-Khayyal and Falkl983; Horst and Tuy, 1987; Al-Khayyal 1990), (c) Cutting Plane methods (e.g. Tuy et al. 1985), (d) Difference of Convex (D.C.) and Reverse Convex methods (e.g. Tuy 1987 a,b), (e) Outer Approximation methods (e.g. Horst et al. 1992), (0 Primal-Dual methods (e.g. Shor 1990; Floudas and Visweswaran 1990,1993; Ben-Tal et al 1994), (g) Reformulation-Linearization methods (e.g. Sherali and Alameddine, 1992; Sherali and Tuncbilek 1992), and (h) Interval methods (e.g. Hansen 1979). The probabilistic methods include (i) random search approaches (e.g. Kirkpatrick et a. 1983), and (ii) clustering methods (e. g. Rinnoy Kan and Timmer 1987). Recent books for global optimization that discuss the above classes are available by Pardalos and Rosen (1987), Torn and Zilinskas (1989), Ratschek and Rokne (1988), Horst and Tuy (1990) and Floudas and Pardalos (1992).

Contributions from the chemical engineering community to the area of global optimization can be traced to the early work of Stcphanopoulos and Westerberg (1975), Westcrberg and Shah (1978), and Wang and Luus (1978). Renewed interest in seeking global solution was motivated form the work of Floudas et al (1989). The first exact primal-dual global optimization approach was proposed by Floudas and Visweswaran (1990), (1993) and its features were explored for quadratically constrained and polynomial problems in the work of Visweswaran and Floudas (1992), (1993). At the same time Swaney (1990) proposed a branch and bound global optimization approach and more recently Quesada and Grossmann (1993) combined convex undeiestimators in a branch and bound framework for fractional programs. Manousiouthakis and Sourlas (1992) proposed a reformulation to a series of reverse convex problems, and Tsirukis and Reklaitis (1993 a,b) proposed a feature extraction algorithm for constrained global optimization. Maranas and Floudas (1992,1993,1994 a,b) proposed a novel branch and bound method combined with a difference of convex functions transformation for the global optimization of atomic clusters and molecular conformation problems that arise in computational diemistry. Vaidyanathan and El-Halwagi (1994) proposed an interval analysis based method and Ryoo and Sahinidis (1994) proposed reduction tests for branch and bound based methods.

In this review paper, we will focus, on deterministic global optimization methods since they provide a rigorous framework for exploiting the inherent structure of process synthesis models . In particular, we will discuss decomposition based primal-dual methods and branch and bound with difference of convex functions global optimization approaches developed in the Computer-Aided Systems Laboratory, CASL, of the Department of Chemical Engineering of Princeton University.

Decomposition Methods

Floudas and Visweswaran (1990, 1993) proposed a deterministic primal-relaxed dual global optimization approach, GOP, for solving several classes of non-convex optimization problems for their global optimum solutions. These classes are defined as:

Determine a globally e-optimal solution of the following problem:

$$\min_{x,y} \; f(x.y)$$

$$\begin{array}{rrcl} \text{subject to} & g\;(x,y) & \text{£} & 0 \\ & h(x,y) & = & 0 \\ & X & \text{€} & X \\ & y & \text{€} & Y \end{array} \qquad \text{(PI)}$$

where X and Y are non-empty, compact, convex sets, *g(x,y)* is an m-vector of inequality constraints and *h(x.y)* is a p-vector of equality constraints. It is assumed that the functions *ftx.y), g(x,y)* and *h(x,y)* are continuous, piecewise differentiable and given in analytical form over X x Y. The variables *y* are defined in such a way that:

(a) *f(x,y)* is convex in x for every fixed y, and convex in *y* for every fixed r,

(b) *g(x,y)* is convex in x for every fixed y, and convex in *y* for every fixed x and

(c) A(xj)isaffii^inxforevei7ftted>,aiulaffinein>tocveiy fixedx.

Examples of process synthesis problems with this structure are superstructures for separation systems, and heat exchanger networks in which balance equations involve bilinear terms, as well as phase equilibrium problems that can be transformed so as to exhibit the bi-convex characteristics of the above conditions.

Making use of duality theory along with several new theoretical properties, a global optimization algorithm, GOP, has been proposed for the solution of the problem through a series of **primal** and relaxed **dul** problems that provide valid upper and lower bounds on tbeglobal solution. The GOP algorithm decomposes the original problem into **primal** and **relaxed dual** subproblems. The primal problem is solved by projecting on the *y* variables, and takes the form:

$$v(y^k) = MB^* \underset{x}{} f(x,y^k)$$

x

subject to *g (x,/)* £ 0 (P2)

h (x^) *** 0**

X€ X

A feasible solution x^k of the primal problem (P2) with objective ***viy*)*** represents an upper bound on the global optimum (i.e. Upper BouMfevfy*;) solution of (PI), and at the same time it provides the Lagrange multipliers X*. |JL* for the equality and inequality constraints respectively.

The Lagrange multipliers (X*. \i^k) **are** subsequently used to formulate the Lagrange function *L (x,y$_t$* X*. \f*y* which is used in the dual problem. Invoking the dual of (PI) and making use of several properties of the problem structure, the GOP algorithm solves a relaxation of the dual problem through a series of relaxed dual subproblems. The y-space is partitioned into subdomains and each relaxed dual subproblem represents a valid underestimation of^d (PI) for a particular subdomain. Each relaxed dual is awociatftd with a combination of bounds B_p of the x variables which appear in bilinear x-y products in the Lagrange function, and takes the forms:

MIN *HB*

S.I.

$$\left.\begin{array}{l} \mu_B \geq L\left(x^{Bj}, y, \lambda^k, \mu^k\right)\Big|_{x^k}^{lin} \\ \nabla_{x_i} L\left(x, y, \lambda^k, \mu^k\right)\Big|_{x^k} \leq 0 \text{ if } x_i^{Bj} = x_i^U \\ \nabla_{x_i} L\left(x, y, \lambda^k, \mu^k\right)\Big|_{x^k} \geq 0 \text{ if } x_i^{Bj} = x_i^L \end{array}\right\} k = 1, 2 \ldots, (K-1)$$

$$\left.\begin{array}{l} \mu_B \geq L\left(x^{Bp}, y, \lambda^K, \mu^K\right)\Big|_{x^K}^{lin} \\ V_{x_i} \ L(x.y,X^K.n^K)\backslash_x K \ \ * \ \ 0 \ \ \text{«f} \ \ xf? = x_i^U \\ V_{x_i} \ L\left(x, y, \lambda^K, \mu^K\right)\Big|_{x^K} \ \geq 0 \text{ if } x_i^{Bp} = x_i^L \end{array}\right\} \begin{array}{c} \textit{current} \\ \textit{iteration} \\ K \end{array} \qquad \text{(P3)}$$

The first three sets of constraints of (P3) correspond to the previous (K-l) iterations with the first one denoting the i ^grpngf. undfifpstitPftting cute and the second tnd third defining the partitioning of the y-space. In the current iteration K the bounds Bj of the previous iterations are fixed while the current combinations of bounds Bp need to be considered. The last three sets of constraints, which change as B_p change, are the underestimating cuts for the partitioned subdomain under consideration. Hence, the relaxed dual problems at the current iteration K are equivalent to setting the x-variables to a combination of their bounds Bp,and solving for a corresponding domain of the y-variables. After solving (P3) for all combinations of bounds Bp, we select the minimum of these solutions and the solutions of the previous iterations. This will provide the new y to be considered in the primal problem (P2) and its corresponding solution is guaranteed to be a valid lower bound on (PI). Solving the primal problem (P2) and updating the upper bound as the minimum solution found, a monotonically non-increasing sequence of updated upper bounds is generated. Solving the relaxed dual problems (P3), a monotonically non-decreasing sequence of valid lower bounds is generated due to the accumulation of previous constraints. As a result, the GOP algorithm attains finite convergence to an e-global solution of (PI) through successive iteration between the primal and relaxed dual problems.

The GOP algorithm along with its primal problem (P2) and its relaxed dual problems (P3) have an interesting geometrical interpretation. Figures 7a, 7b and 7c illustrate graphically the GOP applied to the motivating pooling/blending problem discussed earlier. For a starting point of p * 2, the primal problem corresponds to point A of Figure 7a. Note that for p * 2 the primal problem is a linear programming problem with objective equal to zero. The y-space, which is 1 £ p £ 3, is pardoned into 2 sub-domains, one for 1 £ p £ 2 and the other for 2 ^ p £ 3, and one relaxed dual problem is solved for each sub-domain. Figure 7a shows the linear underestimator AB for 1 £ p S 2, and the underestimator AC for 2 £ p £ 3. Note that the underestimators are linear since the relaxed dual problems are linear in p and the points B and C correspond to the solutions of the corresponding relaxed dual problems. Also note that the underestimator AB passes through the global optimum (p * 1.3, -750). At the end of the first iteration we have an upper bound of zero and a lower bound of -1500. Since -1500 < -350, the next point under consideration for p is p = 1. For p « 1 the primal problem has as solution point D with objective value of -700. Since point D is in the boundary of the range of p, there is only one relaxed dual problem and hence one underestimator, shown as DE in Figure 7b, where point E is the solution of the relaxed dual problem.

Figure 7(a)t Iteration I

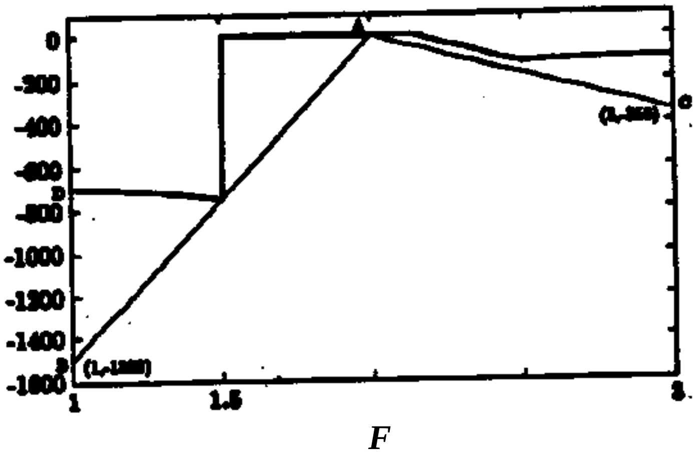


Figure 7(b): Iteration H

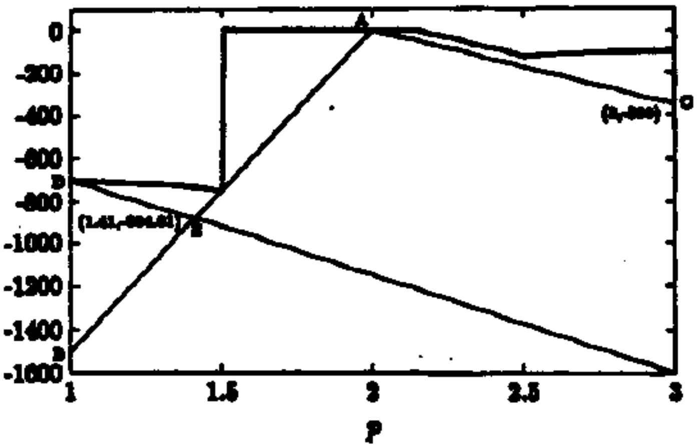
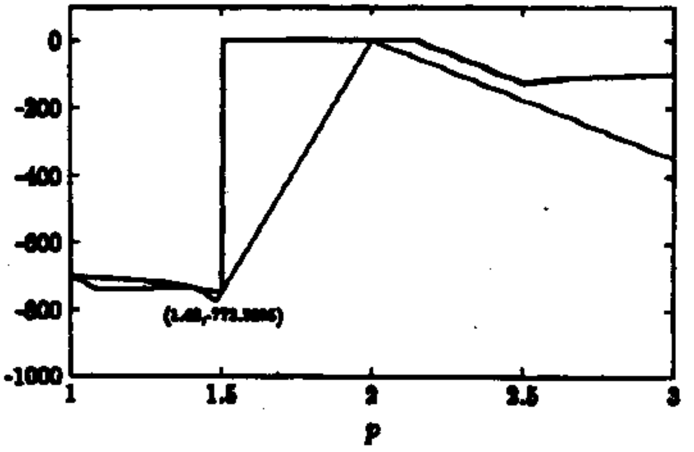


Figure 7(c)t Undereitimator after Iteration III



27

At the end of the second iteration, we have an upper bound of -700 and a lower bound of -884.61. Since -884.61 < -350, the next p under consideration is p « 1.41. Figure 7c shows the underestimating function after three iterations of the GOP algorithm. Note that we have a piece-wise linear underestimating function. Also note that since the primal problem for p« 1.41 has lower value than -350 we can eliminate the domain 2 £ p £ 3. The GOP algorithm has quickly identified the region of the global optimum by providing tight upper and lower bounds, and converges to the global solution in 6-7 iterations.

Visweswaran and Floudas (1990) demonstrated that the *Global Optimization Algorithm. GOP, cm* address several classes of BOB-COOvex mathematical problems that include:

(i) Bilinear, negative definite and indefinite quadratic programming problems.

(ii) Quadratic programming problems with quadratic constraints.

(iii) Unconstrained optimization of polynomial functions.

(iv) Optimization problems with polynomial constraints.

(v) Constrained optimization of ratios of polynomials.

Analysis of the results, obtained via the computational experience of the GOP algorithm on the above mentioned classes of nonconvex optimization problems, verified that a global optimum solution can be obtained from any starting point

Visweswaran and Floudas (1992) studied the class of polynomial functions of one variable in the objective and constraints of problem (PI) and showed that the primal problem reduces to a single function evaluation while the relaxed dual problem is equivalent to the simultaneous solution of two linear equations in two variables. The resulting global optimization approach was demonstrated to perform favorably compared to other algorithms.

Visweswaran and Floudas (1993) proposed new theoretical properties that enhance significantly the computational performance of the GOP algorithm. These properties exploit further (i) the structure of the linearized Lagrange function around x^k, which contains bilinear terms in x and y, linear terms in x, and either linear or convex terms in y, and (ii) the gradients of linearized Lagrange function around x^k, which are linear functions of only the y variables. The first property identifies the combinations of bounds that need not be considered if the gradients of the linearized Lagrange function maintain the same sign. The second property shows that if the gradient of the linearized Lagrange function with respect to xi is zero, then we can set *x* to either its lower or upper bound. The third property allows for updates of the bounds on the x variables at each iteration. Properties 1 and 2 reduce significantly the number of combinations of bounds of the x variables, and hence reduce the number of relaxed dual problems that needed to be solved at each iteration. Property 3 results in tighter underestimators for each of the partitioned subdomains, which in turn results in faster convergence of the upper and lower bounding sequences. The effect of the new properties is illustrated through application of the GOP algorithm to a difficult indefinite quadratic problem, a multiperiod tankage quality problem that occurs frequently in the modeling of refinery processes, and a set of pooling/blending problems from the literature. In addition, extensive computational experience is reported for randomly generated concave and indefinite quadratic programming problems of different sizes. The results show that the properties help to make the algorithm computationally efficient for fairly large problems. Visweswaran and Floudas (1994) presented a (MILP) formulation for aU relaxed duak ai each ben^on of the GOP algroithm. This is baaed on a branch and bound framework for the GOP and allows for implicit enumeration of the partitioned subdomains.

A very important advance on the GOP approach has been recendy made by Liu and F k > ^ (1993). It is shown thai the GOP approach can be applied to very general dasses of nonlinear problems defined as:

MIN F(x)
x

ST. Gi(x)£O i«l,2,...,m (P4)
X€ X

where X is a non empty, compact, convex set in R^n, and the functions F(x), Gi(x) are C^2 continuous on X. This result, even though it is an existence theorem, is very significant because it extends the classes of mathematical problems that the GOP can be applied to firom polynomials or ratios of polynomials to arbitrary nonlinear objective function and constraints that may include exponential terms and trigonometrk: terms with the only requirement that these functions have continuous first and second order derivatives. Based on this result, it is clear the GOP approach is applicable to very broad mathematical problems.

Branch and Bound Methods with (D. C.) transformatlon

A novel branch and bound global optimization approach which combines a special type of difference of convex functions' transformation with lower bounding underestimating functlons was recently proposed by Maranas and Floudas (1994 a,b). This approach is applicable to the broad class of optimization problems stated in (P4), and this special type of (IXC.) transformation is the basis of the result reported in Liu and Floudas (1993). In the sequel, we will discuss the essential elements of this approach by considering the problem of:

MIN F(x)
x

<P5)

ST. x€X * {xj|xf £ xj £ x_i£xj\i = l,2,...,n}

where X is a nonempty, compact, convex set in R^n, and the objective function F(x) is C^2 continuous on X.

Adding a separable quadratic term to F(x), introducing new variables *x{* * *x_h* and subtracting the same term firom F(x) we have:

$$\text{MIN} \qquad F(\mathbf{x}) + \alpha \sum_{i=1}^{n}\left[x_i^2 - x_i \cdot x_i'\right]$$

$$\text{xf i Xj £ } x_i''$$

$$(x_i')^{\ell} \leq x_i' \leq (x_i')^{u}$$

(P6)

$$\textbf{S.T.} \qquad \text{xj-x\{ } \mathbf{=0} \qquad \mathbf{f«1.2}\ldots n$$

The key idea is to employ eigenvalue analysis and define the nonnegative parameter a in such a **way that the following tern:**

$$\mathbf{+(x)\text{-}F(x) + a \underset{i=1}{£}x?}$$

becomes convex. Then, (P6) takes the form

$$\text{MIN} \qquad \phi(x) - \alpha \sum_{i=1}^{n} x_i x_i'$$

$$\mathbf{xf £} \, x_i \, \mathbf{£ } x_i^{u}$$

$$(x_i')^{\ell} \leq x_i' \leq (x_i')^{u}$$

(FT)

$$\text{S.T.} \qquad \text{Xj} \; - \; \text{x[} \; = \; 0$$

which has as objective a difference of two convex functions out of which the one that is substracted is separable quadratic. Formulating the dual of (F7) and applying die KKT conditions, Maranas and Floudas (1994***JBL)*** showed that the dual of (P7) is equivalent to (P8) (see Appendix A3 of that paper):

$$\text{MIN} \quad L(x) = \left[F(x) + \alpha \sum_{i=1}^{n} x(-f x_i \; X \; r - x_i) \right] \qquad \textbf{(P8)}$$

$$\mathbf{xf \wedge Xi} \leq \boldsymbol{xf}$$

where **a** is a nonnegative parameter which is greater or equal to the negative one half of the minimum eigenvalue of the Hessian of F(x) over the box xf £ x_k £ x"$_i$ i • ** 1, 2, ...,n $\left\{ \text{i.e} \quad \mathbf{a} \text{ £ MAXj } \left(0_y \frac{1}{2} \lambda_{min} > \right\} \right)$ Note that the term added to F(x) has the effect of overpowering the nonconvexity characteristics of F(x) with the addition of the term (2 a) to all of the eigenvalues of its

Hessian. The smaller the value of a, the tighter the uiulefestimator Ux) is fcr F(x) which may imply less total manber erf iterations for convergence. Hence, one would ideally desire the non negative parameter a to be exactly equal to $\left(I - \frac{1}{2} \wedge \min I \right)$ **** •*>***« th* to the branch and bound widi difference of convex function transformation it suffices to find an upper bound on $\left(f - \frac{1}{2} T \wedge a w i \right)$ 1 »* ^{9ckci} ∘ •* *$& t$^{\circ this}$ *upper* bound. In this case we add more convex terms than needed and do not produce the tightest underestimator, bat we satisfy die required conditions for convergence.

Given F(x), the aekction erf the nonnegative parameter a may involve (i) die derivation of analytical expressions for the eigenvalues erf its Hessian, or (ü) the devetoproemcrf bounds on the eigenvalues of the Hessian of F(x). Mannas and Floudas (1992), (1993) studied alternative (i) for a variety of atomic/molecular clusters. They derived analytical expressions for the eigenvalues for any potential function which is a function of only the distance between atoms (e. g. Lennard-Jones, Coulomb, Mie, Morse, Gaussian, Bom-Mayer, Buckingham). Mannas and Floudas (1994. a, b) proposed a number of ways of obtaining bounds on the eigenvalues of the Hessian of F(x). One general way is via the use of the measure of a matrix, a concept recently applied to the stability erf reactor networks at die process synthesis level (see Kokossis and Floudas, 1994). If a denotes die Hessian of F(x),thcn the measure of the matrix (-A), denoted as \i (-A), provides an upper bound on (-Xmin)- $^{\text{Ils}}$ formulation is a convex problem, and we can use either the 1 or ©o norm. Appendix A.2 of Mannas and Floudas (1994*), describes such a formulation.

It should be pointed out however that if Xmin goes to (- «>), then this represents a case in which we cannot create <I>(x) convex. A sufficient condition which excludes such a possibility is when the dements of the Hessian matrix have finite values. This can be seen easily using the measure erf a matrix concept One instance of Xmin tending to (- °°) is reported in the Weber's facility location problem (see Mannas and Flouds, 1994.C)

The function L(x) is a lower bounding function of F(x), and exhibits the following important properties:

Property 1: L(x) is always a valid underestimatorofF(x) inside the box $[xf_f \;\; x_i^*]$, that is

$$L(x) \leq IF(x).$$

Property 2: L(x) matches F(x) at all corner points erf the box.

Property 3: L(x) is convex in the box $[xf, \;\; x_i'']$.

Property 4: The maximum separation between L(x) and F(x) is bounded and is proportional to a and to the square of the diagonal of the box $[xf, \;\; x_i'']$, that is

$$\max_{x_i^{\ell} \leq x_i \leq x_i^{u}} \quad (F(x)\text{-}Ux)) = \underset{i-i}{\text{iai}}(x?\text{-}xf)$$

Property 5: The undercstiiiuttorL(x) constructed over tsub-tox of the current box is always lighter than the underestimator of the current box.

In summary, the properties show that L(x) is a convex, lower bounding function of F(x), L(x) matches F(x) at all comer points erf the box constraints inside which it has been defined. The values of L(x) at any point, if L(x) is constructed over a tighter box of constraints each time, define a nondecreasing sequence. Also note that Property 4 answers the question of how small the sub-boxes must become before the upper and lower bounds of F(x) are within £. If 5 is the diagonal of the sub-box, and E is the convergence tolerance, we have have:

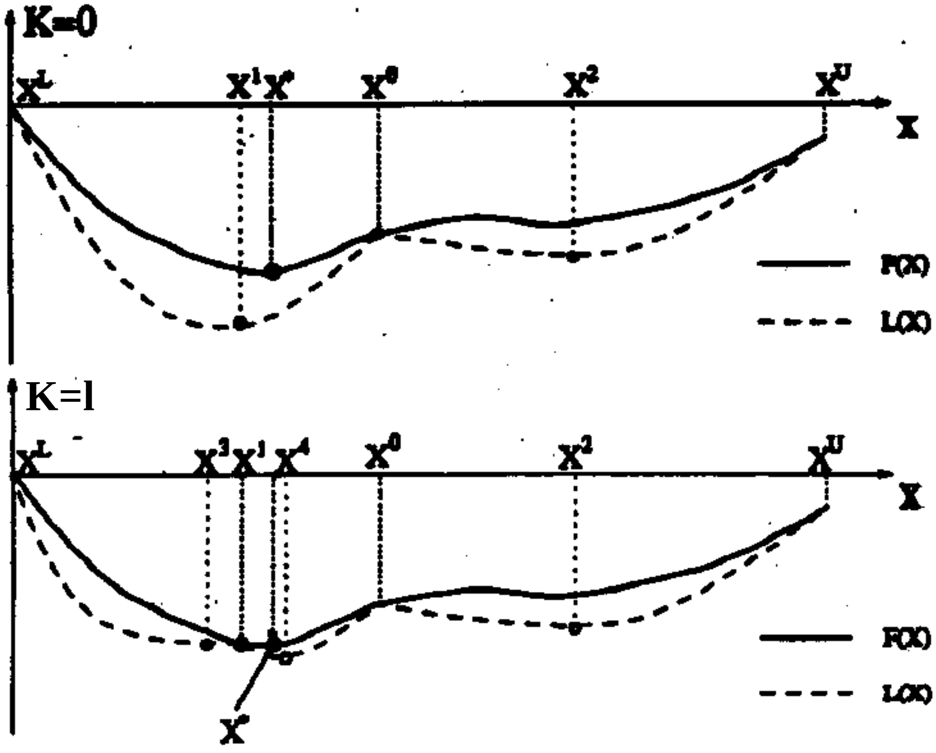
$$\delta < \sqrt{\frac{4\varepsilon}{\alpha}}$$

Note that 8 is proportional to the square root of e and inversely proportional to the square root of a. As a result, the smaller value of a the faster the convergence rate becomes.

These properties of the lower bounding function, L(x), coupled with an efficient partitioning scheme resulted in a branch and bound global optimization approach that is guaranteed to converge to an e-global solution in a finite number of iterations. Maranas and Floudas (1994*) analyzed the structure of the branch and bound tree resulting from the subdivision process and developed formulas for finite upper and lower bounds on the total number of iterations required for £ - convergence. The maximum number of iterations is exponential in the total number of variables while the minimum number of iterations depends linearly on the total number of variables. Computational experience with molecular conformation problems indicated that the total number of iterations is much close to the minimum one.

Figure 8 provides the geometrical interpretation of the lower bounding scheme for a function F(x) of one variable x in a box$[x^L, x^u]$. Starting at a point x° we partition the original box into two intervals $[x^L, x°]$ and $[x°, x^u]$, while F(x°) is the current upper bound. For each interval we solve the corresponding convex lower bounding problem and obtain their respective minima at x^1, L(x*) and x^2, $L(x^2)$ respectively. Note at this point the underestimating functions shown with non-solid lines.

Figure 8: Geometric Interpretation of Branch and Bound with (D.C.)



> mlnF(JV), single variable problem in X

- $L(X) = F(X) + a(X^{LBD} -XXX^{UBD} - X)$

Since $L(x^*) < L(x^2)$ we focus on the $[x^L, x^\circ]$ for the second iteration, evaluate the function $F(x^1)$ andpaitition the interval $[x^L, x^\circ]$ into the intervals $[x^L, x^1]$ and $[x^1, x^0]$. For each of these intervals we obtain the undcrcstimators and their minima which arc at x^3 and x^4 respectively. Since $L(x^3) > L(x^4)$ we focus on the interval $[x^1, x^4]$ forthe next intention and evaluate $F(x^4)$. Note that we are very close to the global solution in just two iterations.

The branch and bound with (D. C) transformation was applied to (a) clusters of atoms/molecules in which only non-bonded interactions take place, (b) molecular structure detenninatm **of small molecules in which** bonded and non-bonded interactions are taken into account, and (c) financial planning models for multiperiod operation. Application (a) resulted in ratios of polynomials and exponential terms in the distances between atoms. Application (b) involved very complex expressions not only in the distances but also in the dihedral angles and had ratios of polynomials, exponentials, and trigonometric terms. Application (c) employed multiperod models for stochastic programming using the mean-variance model over all possible scenarios, and resulted in generalized polynomials and square root terms. All computational results highlight the power of the (D. C.) transformation within a branch and bound framework.

Global Optimization Tools and Computational Experience

Global optimization tools have been recently developed in the Computer Aided Systems Laboratory, CASL, of the Department of Chemical Engineering at Princeton University for the primal-relaxed dual algorithm, GOP, and the branch and bound approach that combines (D.C.) transformation and a special type of lower bounding function. These tools are denoted as cGOP and OtBB for the decomposition and branch and bound global optimization algorithms respectively. Both cGOP and aBB are written entirely in C and make use of MINOS, NPSOL, CPLEX for linear subproblems; MINOS, NPSOL for nonlinear programming subproblems. They have been implemented as a library of subroutines with emphasis on modularity and expandability, the subroutines for the same task have the same interfaces, and modifications in the problem data are allowed at any stage. Both cGOP and aBB have a user specified function capability which allows for connection to any external subroutine that can be treated as a black box. The current versions of cGOP and aBB can be either standalone or can be called as subroutines.

Computational experience with cGOP and aBB is shown in Table 4 and Table 5 for a wide variety of applications, that include: pooling/blending problems, heat exchanger network synthesis problems, nonsharp separation synthesis, problems with quadratic objective and box constraints, concave programming problems, bilevel linear optimization problems, minimization of the Gibbs free energy with NR1L and UNIQUAC in phase and chemical reaction equilibrium, tangent plane stability criterion in phase equilibrium, clusters of atoms and molecules, molecular structure determination problems, and financial planning problems. The first three and the last pooling problems correspond to the Haverly problem and the multiperiod tankage problem and are described in Floudas and Visweswaran (1990) and Visweswaran and

Floudas (1993). The fourth and fifth pooling problems are described inBen-TaletaL(1994). The first two heat exchMigcr problems are taken from Floudas and Ciric (1989) while the last three are descibed in Ben-Taletal.(1994). The first two heat exchanger problems are taken from Floudas and Ciric (1989) whik the last three are described in Quesada and Gfossmann (1993). The separations problem is described in Aggarwal and Floudas (1990). The minimization of Gibbsfirce energy problems arc discussed in McDonald and Floudas (1994a). The tangent pine stability criterion problems are presented in McDonald and Floudas (1994b). The quadratic objective with box constraints, concave objective with linear constraints, and indefinite quadratic problems are discussed in Visweswaran and Floudas (1993). The Lennard Jones clusters of atoms problems are discussed in Mannas and Floudas (1993). The molecular structure determination problems are presented in Maranas and Floudas (1994jub.). The molecular structure detenninaoon problems arc presetted tnM aranas and Floudas (1994 a,b). The financial planning problems are described in Maranas et aL (1994). As Tables 4, 5 illustrate, small medium, and in certain cases large global optimization problems can be solved within a modest computational effort.

Plane Stability Criterion**	TWA3G	<$	3	2	85	0.94
	PBW3T1	6	3	2	53	0.62
	PBW3G1	6	3	2	213	2.37
	PBW3T6	6	3	2	549	4.98
	PBW3G6	6	3	2	757	7.09
Quadratic Objective, Box CrOnstFaints	QBR1	10	300	«	2	6.45
	QBR2	20	300	-	2	46.01
	QBR3	30	160	-	2	345.83
	QBR4	30	300	-	2	411.016
Concave Objective Linear Constraints	CLR1	50	SO	SO	2	1.62
	CLR2	100	100	100	2	22.95
	CLR3	SO	ISO	100	2	0.65
	CLR4	SO	200	100	2	2.73
	CLR5	SO	250	100	2	10.47
	CLR6	100	250	100	2	47.5
Indefinite Objective, Linear Constraints	IND1	100	100	100	2	11.53
	IND2	SO	50	50	2	0.71
	IND3	100	50	50	2	4.35
	IND4	SO	100	50	2	1.28
	IND5	50	200	50	3	15.17
	IND6	50	200	100	2	6.76
	IND7	75	200	100	2	17.72
	IND8	50	250	100	2	22.27
Bilevel Linear	BL1	2	3	6	3	0.47
	BL2	2	2	5	3	0.28
	BL3	1	1	6	2	0.11
	BL4	1	1	5	3	0.23
	BLS	6	3	10	3	0.75
	BL6	1	1	5	3	0.29
	BL7	1	2	4	2	0.16
	BL8	1	1	4	3	0.23
	BL9	1	1	4	3	0.22
	BL10	1	2	4	2	0.16
	BL11	2	3	6	5	0.82

N_x : number of x-• variables

Ny : number of y-• variables

Nc : number of constraints

Nl : number of iterations

CPU : see's in HP-730

•* : using GLOPEQ (McDonald and Floodas, 1994)

Table *Si* Computational **Rg«ult« with (XBB**

I. Clusters of Atoms/Molecules

Problem Name	TV	NCV	RT	NI
U8	18	3	1%	**12**
U13	33	3	1.5%	15
LJ18	48	**3**	1.5%	20
LJ22	60	3	1.5%	16
U24	66	**3**	13%	19

n: Mnlwulor ^trurtUK 'Determination

Problem Name	iy.	NCV	RI	Ni
PRO	21	2	**0.01%**	400
APRO	27	2	0.01%	200
ABUT	51	3	0.01%	1000
BUT	54	3	0.01%	100
NPEN	90	4	0.01%	1000

m. Financial Planning

Problem Name	TV	NCV	Nc	NI
FM100	8	8	11	2
FM300	8	8	1I	2
FM500	8	8	1I	3
FM1000	8	8	11	6
FM10000	8	8	11	6
FMC100	8	8	11	2
FMCTX100	8	8	1I	7

TV: total number of variables

NCV : nonconvex variables

RT: relative tolerance

CONCLUDING REMARKS

This *paper* has attempted to present an overview of two major emerging areas in algorithmic synthesis: logic and global optimization. As indicated at the beginning of the paper these areas have been motivated by the need to improve the modelling in discrete optimization techniques, reduce the combinatorial search and avoid getting trapped into poor suboptimal scdutions. In the next two subsections we briefly discuss some future directions for research.

Current and Future Directions for Logic Based Optimization

Comparing the review on MINLP given by Grossmann (1990a) at the previous Snowmass meeting, it is apparent that the work on logic based optimization has provided a new direction to address the need of integrating qualitative knowledge into mixed-integer optimization models for synthesis (see also Rippin, 1989). As has been shown by developing new models and branch and bound methods that effectively incorporate logic, order of magnitude reductions can be achieved in the combinatorial search involved in these problems. Furthermore, another very important aspect has been to achieve a better understanding of some fundamental issues related to the modelling of discrete optimization problems. In particular, the concept of w-MIP representability has proved to be a useful theoretical concept few characterizing the nature of discrete constraints. While significant progress has been made, it is clear that a number of major issues and challenges still remain for future research. These include the following:

1. The handling of temporal and modal logic is challenging and should prove to be very useful for a wide range of problems in process scheduling.

2. Other kinds of logic cuts should be investigated apart from the logic relating units in a superstructure. The cuts affect the solution efficiency considerably and also allow one to better understand the modelling of discrete programming problems. One possibility for logic cuts are constraints that prevent multiple mathematical representations for the same design configuration within a superstructure.

3. Most of the work on integration of logic has been directed to discrete linear problems. Still much work remains in the integration of logic for nonlinear problems. In addition, there is the issue of integration with new cutting plane methods such as the one by Balas et al. (1993).

4. The problem of developing techniques to efficiently model and solve superstructures of large scale process flowsheet problems is another major issue. The use of disjunctions should serve to reduce the level of nonlinearity present in a mixed-integer representation, as well as allow for a systematic scheme for generating efficient models for these problems.

5. Further study is required on the representability of disjunctive constraints as mixed-integer constraints. Our work on w-MIP representability can only be regarded as preliminary work in the area and has just demonstrated the potential for research in this problem. A better understanding of representability issues could lead to the development of modelling languages for generating efficient discrete optimization models.

6. The ctevetopment of computer software th^ **efficiently automates the various approaches based on** logic and their more extensive testing on large scale problems is still required.

7. The integration with other design methodologies should be exploit **in which logic information** can be generated from a preliminary screening. Example of this are the work by FriedleretaL (1991) and the work by Daichcndtand Grossmann (1994aJ>).

8. The ultimate objective is ID provide a so^ foundation to new danes of hybrid optiinization models which are expressed in terms of equations and logic relatfwns. This shouW also provide a clefflkik with dynamic simulation models which yi#A tend to exhibit this structure.

Progress and better understanding in the above problem will undoubtabiy lead to a new generation of discrete optimization models and solution methods. Furthermore, it is clear that these efforts can complement advances in global optimization.

Current and Future Directions in Global Optimization

In the global optimization section we have attempted to present an overview of global optimization methods which are based on the concepts of decomposition and branch and bound coupled with a (DC) transformation. From this review, it is apparent that we have experienced a significant progress in the area of global optiinization and its applications in Chemical Engineering over the last five years. New theoretical results and algorithms have emerged and their application to a number of Process Synthesis, Design, and Control problems has already resulted in encouraging results. At.the same time applications in the area of computational chemistry, facility location, and financial planning demonstrate clearly the potential impact of global optimization in the design of new materials and biological systems, the design of process layout, and the design of financial management systems. It is also worth noting that it is the first time that the progress in the area of global optimization is reviewed in a FOCAPD meeting, which is indicative of the recent advances, the potential usefulness, and the growth of this area in Chemical Engineering Design and Control. Global optimization, as a new area, however has a number of important challenges and several open problems which will be the subject of current and future research woik. These challenges include:

(1) new global optimization approaches for non-convex (MINLP) models arising in Process Synthesis;

(2) global optimization methods for generalized geometric programming problems (e.g. signomials) which arise in many design and robust control applications;

(3) new global optimization methods for nonconvex models with trigonometric and exponential functions that arise in Computational Chemistry, Biology and fMwu^ml reaction engineering applications;

(4) global optimization methods which can determine all solutions of nonlinear systems of equations that arise in phase equilibrium, azcotropic distillation, and reaction engineering;

(5) global optimization methods for bilcvcl and muUilevel linear and nonlinear models that appear in planning problems, flexibility analysis, and optimal control approaches in batch distillation;

(6) new global optimization approaches whkA can address impliciUy define functions; and

(7) distributed computing methods for global optimization with the aim at addressing medium to large scale optimization problems.

Even though the above challenges represent undoubtedly formidable tasks, we should see exciting developments over the next dfcatfp.

References

Aggarwal A. and C. A. Floudas, "Synthesis of General Distillation Sequences-Nonsharp Separations", Computers and Chemical Engineering, 14,6, pp. 631-653,1990.

Al-Khayyal F. A. and J. E. Falk, "Jointly Constrained Bioconvex Programming", *Math Opers. Res.*, 8, (1983).

Al-Khayyal, F. A., "Jointly Constrained Bilinear Programs and Related Problems: An Overview", *Computers in Mathematical Applications,* Vol. 19, pp. 53-62,1990.

Andrecovich, MJ. and A.W. Westerberg, "An MILP Formulation for Heat-Integrated Distillation Sequence Synthesis," *AIChEJ.* 31,1461 (1985b)

Bagajewicz, M. and V. Manousiouthakis, "On the Generalized Benders Decomposition," *Comptu. chem.* £ngng.9, 1(, 6) 191)1

Balas, E. /Disjunctive Programming: Properties of the convexhull of feasible points," MSRR #348, Carnegie Mellon University, Pittsburgh, PA.(1974).

Balas, E. /Disjunctive Programming and a hierarchy of relaxations for discreteoptimization problems, *SIAM J. Alg. Disc. Meth., 6,* 466-486 (1985).

Balas, E., Ceria, S. And Cornuejols, G. A Lift-and-Project Cutting Plane Algorithm for Mixed 0-1 Programs, *Mathematical Programming,* S8,295-324 (1993).

Beaumont, N. "An Algorithm for Disjunctive Programs," *European Journal of Operations Research,* 48, 362-371(1991).

Benders, J. R, "Partitioning Procedures for Solving Mixed-variables Programming Problems", *Numerische Mathematik 4,*238-252 (1962).

Ben-Tal A., G. Eiger, and V. Gershovitz, "Global Minimization by Reducing the Duality Gap", *Mathematical Programming,* in press (1994).

Cavalier, T. M. And Soyster, A. L.,"Logkal Deduction via Linear Programming/ IMSE Working Paper 87-147, Department of Industrial and Management Systems Engineering, Pennsylvania State University (1987).

Cavalier, T. M., Pardalos, P. M. And Soyster, A. L., "Modeling and Integer Programming techniques applied to Prepositional Calculus," *Computers and Operations Research,* 17(6), 561-570 (1990).

Chandru, V. And Hooker, J. N., "Extended Horn Sets in Prepositional Logic," Working Paper 88-89-39, Graduate School of Industrial Administration, Carnegie Mellon University, Pittsburgh (1989).

Clocksin, W. R And Mcllish, C S., *"Programming in Prolog,"* Springer-Veriag, New York, NY (1981).
Cook, S. A. t "The complexity of theorem proving procedures/ Proceedings of the 3rd ACM Symposium oft the Theory of Computing, pp. 151.158(1971).

Dakhendt, MM. and IJB. Grossmann, "Preliminary Screening for the MINLP Synthesis of Process Systems 1: . Ag|"ft^»andDecomposition*Tecbniqpt$"Com/Ha.chem.Engng.*1 18,663(1994a)

Dafchendt, M.M. and LE. Grossmann, "Preliminary Screening for the MINLP Synthesis of Process Systems n: Heal Exchanger Networks,"Compn/. *chem. Engng.*, 18,679 (1994b)

Davis, M. And Putnam, H., "A computing procedure for quantification theory/ /. *ACM*, 8, 201-215 (1960).

Denenberg, L. And Lewis, H. R., "Logical Syntax and Computational Complexity," Proceedings of the Logic Q)lk)quium at Aadieiu Springer Uctuie Notes in Mathematics 1104, pp. 109-115 (1983).

Douglas, *JM*., "A Hierarchical Decision Procedure for Process Synthesis," *AIChEJ.* 31,353 (1985).

Dowling, W J. And Gallier, J. H., "Linear-time algorithms for testing the satifiability of prepositional Horn formulae," *LogicProgramming*, 3,267-284 (1984).

Duran, M.A. and IJL Grossmann, "An Outer-Approximation Algorithm for a Class of Mixed-integer Nonlinear Programs," *Math Programming* 36,307 (1986).

Floudas, C. A. and A. Aggarwal, "A decomposition strategy for global optimum search in the pooling problem", *Opers Res. J. Compute* 2 (3) (1990).

Floudas, C. A., A. Aggarwal and A. R. Ciric, "Global optimum search for nonconvex NLP and MINLP problems", *Computers Chem. Engng.*, 13,1117 (1989).

Floudas, C. A. and A. R. Ciric, "Strategies for Overcoming Uncertainties In Heat Exchanger Network Synthesis", Computers and Chemical Engineering, 13,10, pp. 1133-1152,1989.

Floudas, C.A. and G.E. Paules IV, "A Mixed-Integer Nonlinear Programming Formulation for the Synthesis of Heat-Integrated Distillation Sequences," *Comput. chem. Engng.* 12,531 (1988)

Floudas, C. A., and Pardalos, P. M., "A Collection of Test Problems for Constrained Global Optimization Algorithms", *Lecture Notes in Computer Science,* Springer-Verlag, Berlin, Germany, Vol. 455,1990.

Floudas, C. A., and Pardalos, P. M., "Recent Advances in Global Optimization", Princeton University Press, Princeton, New Jersey, 1992.

Floudas, C. A. and V. Visweswaran, "A Primal-Relaxed Dual Global Optimization Approach", *Journal of Optimization, Theory, and its Applications,* 78,2, pp. 87-225 (1993).

Floudas, C. A. and V. Visweswaran, "A Global Optimization Algorithm (GOP) for Certain Classes of Nonconvex NLPs -1. Theory", Computers and Chemical Engineering, 14,12, pp. 1397-1417,1990.

Friedlcr, F., K. Taijan, Y.W. Huang and L.T. Fan, "An Accelerated Branch and Bound Method for Process Synthesis," Presented at the 4th Worid Congress of Chemical Engineering, Karlsruhe (1991).

Grossmann, LE., "Mixed-Integer Programming Approach for the Synthesis of Integrated Process Flowsheets," *Comput. chem. Engng.* 9,463 (1985)

Grossmann, LE.,t "MINLP Optimization Strategies and Algorithms for Process Synthesis," in *Foundations of Computer-Aided Design,* JJ. SiirolaJ.E. Grossmann and G. Stephanopoulos (Eds.), Cache-Elsevier, Amsterdam (1990a)

Grossmann, I.E., "Mixed-Integer Nonlinear Programming Techniques for the Synthesis of Engineering Systems/ *Res. Eng. Des. I,*205 (1990b)

Grossmann, I.E. and M.M. Daichendt, "New Trends in Optimization-based Approaches for Process Synthesis", to appear in Proceedings of Process Systems Engineering, Korea (1994).

Geoffrion, A. M, "Generalized Benders Decomposition," *Journal of Optimization Theory and Applications,* 10(4X237-260(1972).

Gundcrscn, T. and L. Naess, The Synthesis of Cost Optimal Heat Exchanger Networks. An Industrial Review of the State of the Art," *Comput. chem. Engng.* 12,503 (1988)

Hendry, JJE., D.F. Rudd and J.D. Seader, "Synthesis in the Design of Chemical Processes/ *AIChEJ.* 19,1 (1973)

Hansen, P., B. Jaumard and S. Lu, "Global Optimization of Univariate Lipschitz Functions: I. Surrey and Properties". *Mathematical Programming,* 55,2S1-272 (1992*).

Hansen, P., B. Jaumard and S. Lu, "Global Optimization of Univariate Lipschitz Functions: New Algorithms and Computational Comparison", *Mathematical Programming,* 55,273-292 (1992.b).

Hansen, E. R., "Global Optimization Using Interval Analysis: TheMulti-dimensional Case", Numerische Mathematik, Vol. 34, pp. 247-270, 1980.

Havcrly, C. A., "Studies of the behaviour of recursion for the pooling problem", *SIGMAP Bull,* 25,19 (1978).

Hlavacek, V., "Synthesis in the Design of Chemical Processes,", *Computera and Chemical Engineering,* 2, 67-75(1978).

Hooker, J. N., "Resolution vs Cutting plane solution of inference problems : some computational experience," *Operations Research Utters,* 7(1), 1 (1988).

Hooker, J.N., H. Yan, I.E. Grossmann, and R. Raman "Logic Cuts for Processing Networks with Fixed Charges," *Computers and Operations Research* 21,265-279 (1994).

Horst, R. and H. Tuy, "On the convergence of global methods in multiextrcmal optimization", *J. Optimization Theory Applic,* 54,253 (1987).

Horst, R., Thoai, N. V., and De Vries, J., "A New Simplicial Cover Technique in Constrained Global Optimization", *Journal of Global Optimization,* VoL 2, pp. 1-19,1992.

IBM, "OSL User Reference Manual," IBM Corp, Kingston, New York (1991).

Jcroslow, R. G. And Lowe, J. K., "Modelling with Integer Variables," *Mathematical Programming Study,* 22, 167-184 (1984).

Jeroslow, R. E. and Wang, J. /Solving prepositional satisfiability problems". *Annals of Mathematics and AI,* 1,167-187 (1990).

Karp, R. M., "Reducibility among combinatorial problems", *Complexity of Computer Calculations* (Miller, R. E. and Thatcher, J. W., eds) Plenum, New York, pp. 85-104 (1972).

Kocis, G.R. and I.E. Grossmann, "Relaxation Strategy for the Structural Optimization of Process Flow Sheets," *Ind. Eng. Chem. Res.* 26,1869 (1987)

Kocis, G.R. and I.E. Grossmann, "A Modeling and Decomposition Strategy for the MINLP Optimization of Process Flowsheets," *Comput. chem. Engng.* 13, 797 (1989b)

Kokossis, A. C. and C. A. Floudas, "Stability in Optimal Design: Synthesis of Complex Reactor Networks", AIChEJ., 40(5), pp. 849-861 (1994).

Lien. K. M. And WahL P. E. "If you cant beat them, join them. Combine Artificial Intelligence and Operations Research Technique* in Chemical Process Systems Design." Proceedings of PSE91, VoLIV. pp.1.1-15, MoMBbeUo. Canada (1991).

Liu. W. B. and C A. Floudas. "A Remark on the OOP Algorithm for Global *Optimiuaon", Journal of Global Optimisation,* VoL 3. No. 4. pp. 519-522 (1993).

Lovdand, D.W., "Automated Theorem Proving: A U>gkal Baas." North Hollaod, Amsterdam (1978).

Mahalec. V. aid Motard, R. L.. "Procedures for the initial design of chemical processing systems." *Computers and Chemical Engineering, 1,51-6** (1977).

MaMurioatlialrivM. and D. Soi»1s«, "A Gtobal O p ^ization Approach to Rationally Constrained Rational Programming". Chem. Eng. Comm., 115.127-147 (1992).

Mannas, C D. and C. A. Floudas, "A Global Optimization Approach for Lennard-Jones Microclusters", J. Chem. Phys.. 97(10). pp. 7667-7678 (1992).

Mannas. C. D. and C. A. Floudas, "Global Optimization for Molecular Conformation Problems". *Annals of Operations Research,* VoL 42, pp. 85-117

Mannas, C. D. and C. A. Floudas, "Global Minimum Potential Energy Conformations of Small Molecules". *Journal of Global Optimization,* VoL 4. No. 2,135-170 (1994a).

Mannas, C. D. and C. A. Floudas, "A Deterministic Global Optimization Approach for Molecular Structure Determination". *Journal of Chemical Physics,* 100.2. January 15. pp. 1247-1261 (1994.b).

Mannas, C. D. and C. A. Floudas. "A Global Optimization Method for Weber's Problem with Attraction and Repulsion". *Large Scale Optimisation: Slate of the Art* (W. W. Hager. D. W. Heam, and P. M. Pwdalos, Editors), Kluwer Academic Publishers, pp. 265-300 (1994.c).

Mannas, C. D., I. P. Androulakis, C. A. Fkwdas, A. J. Berger, and J. M. Mulvery, "Solving Stochastic Control Problems in Finance via Global Optimization", submitted for publication, (1994).

McDonald, C. M. and C. A. Floudas, "Global Optimization for the Phase and Chemical Equilibrium Problem: Application to the NRTL Equation", Computers and Chemical Engineering, accepted for publication (1994a).

McDonald, C. M. and C. A. Floudas, "Global Optimization for the Phase Stability Problem", AIChE J., accepted for publication (1994b).

Mendelson, E., "Introduction to Mathematical Logic", Van Nestrand, New York. 1987.

Nemhauser, G. L. And Wolsey, L. A.,"Integer and Combinatorial Optimization," Wiley-Inierscience, New York (1988).

Nithida, N., G. Stephanopoutos and A.W. Westerberg, "A Review of Process Synthesis," *AlChEJ.* 27,321 (1981)

Papoulias, SA. and LE. Grossmann, "A Structural Optimization Approach in Process Synthesis. Pan I: Utility Systems," *Comput. chem. Engng.* 7,695 (1983a)

Papoulias, S.A. and I.E. Grossmann, "A Structural Optimization Approach in Process Synthesis. Part II: Heat Recovery Networks," *Comput. chem. Engng.* 7,707 (1983b)

Papoulias, S.A. and LE. Grossmann, "A Structural Optimization Approach in Process Synthesis. Part III: Total Processing Systems," *Comput. chem. Engng.* 7,723 (1983c)

Pardalos, P. M. and Rosen, J. B., "Constrained Global Optimization: Algorithms and Applications", *Lecture Notes in Computer Science,* Springer-Veriag, Berlin, Germany, VoL 268,1987.

Post, S. "Reasoning with Incomplete and Uncertain Knowledge as an Integer Linear Program," *Proceedings OfAvignon 87: Expert Systems and their Applications.* Avignon, Prance (1987).

Quesada, I., and LE Grossman, "Global Optimization Algorithm focHeat Exchanger Networks", *Ind. Eng. Chem. Res.,* 32, 487-499 (1993).

Quesada, I. and UL Grossmann, "A Global OptimizationAlgorithm for Linear Fractional and Bilinear Programs", Presented at HMS/ORSAMeeting, Chicago (1993a).

Quesada, I. and LE. Giossmann, "Global Optimization Algorithm for Heal Exchanger Networks," *Ind. Eng. Chem. Research,* 32.487 (1993b).

Quesada, L and LE. Grossmann, "Global Optimization Algorithm of Process Networks with Multicomponent Flows," presented at AIChE Meeting, St. Louis (1993c).

Raman, R. and LE. Giossmann, "Relation Between MILP Modelling and Logical Inference for Chemical Process Synthesis," *Computers andChemical Engineering* 15,73 (1991).

Raman, R. and LE. Grossmann, "Integration of Logic and Heuristic Knowledge in the MINLP Optimization for Process Synthesis," *Computers and Chemical Engineering* 16,155-171 (1992).

Raman, R. and LE. Grossmann, "Symbolic Integration of Logic in Mixed Integer Linear Programming Techniques for Process Synthesis," *Computers and Chemical Engineering,* 17,909 (1993*).

Raman, R. and IJE. Giossmann, "Symbolic Integration of Logic in MILP Branch and Bound Methods for the Synthesis of Process Networks," *Annals of Operations Research,* 42,169-191 (1993.b).

Raman, R. and IJE. Grossmann, "Modeling and Computational Techniques for Logic Based Integer Programming," Computers and Chemical Engineering, 18,563 (1994).

Rardin, R. L. And Choe, U. "Tighter Relaxations of FixedCharge Network Flow Problems," Georgia Institute of Technology, Industrial and Systems Engineering Report Series, #J-79-18, Atlanta (1979).

Ratschek, H., and Rokne, J., "New Computer Methods for Global Optimization", Halsted Press, Chichester, Great Britain, 1988.

Rinnoy Kan A. H.G.andG.T.Timmer, "Stochastic global optimization methods. Parti: clustering methods", *Math. Program,* 39,27 (1987).

Rippin, D.W.T., "Introduction: Approaches to Chemical Process Synthesis," in *Foundations of Computer-Aided Design,* JJ. Siirola, I.E. Grossmann and G. Stephanopoulos (Eds.), Cache-Elsevier, Amsterdam (1990).

Ryoo H.S. and N.V. Sahinidis, "Global Optimization of Nonconvex NLPs and MINLPs with Applications in Process Design", *Computers and Chemical Engineering,* submitted for publication (1994).

Sahinidis, N.V. and *IE.* Grossmann, "Convergence Properties of Generalized Benders Decomposition," *Computers and Chemical Engineering,* 15,481 (1991).

SCICON, "SCICONIC / VM User Guide," SCICON Ltd., London (1986).

Shendi, HL, and Ttacbikk, C. H., "A Global Optimizatk* Algorithm for 1 ^ ^omial Programming Problems Using a RcfonnuUtion-Lincarizauon Technique", *Journal of Global Optimization,* VoL 2, pp. 101-112,1992.

Shendi, R D. *mA* A. Alameddine,"A New Refonnulaticm Linearization T e ^ ^ue for Bilinear Programming Problems", *J. Of Global Optimization,* VoL 2, No. 4, pp. 379-410 (1992).

Star, N. Z., "Dual Quadratic Estimates in Polyromial aral Boolean Progran^ing", *Annals of Operations Research,* VoL 25, pp. 163-168,1990.

Stephanopouk*, O. and A. W. Westoberg, "The UseofHegtenes* Method of Multipliers to Resolve Dual Gaps in Engineering System Optimization", *Journal of Optimization Theory and Applications,* 15(3), 285-309(1975).

Swaney, R. E.,t "Global Solution of Algebraic Nonlinear *Prognms^m,AIChE Annual Meeting, Chicago, 11* (1990).

Torn, A., aod A. Zilinskas, "Global Optimization", *Lecture Notes in Computer Science,* 350, Springer-Veriag, Berlin (1989).

Torn, A., and Zilinskas, A., "Global Optimization", *Lecture Notes in Computer Science,* Springer-Veriag, Berlin, Germany, VoL 350, 1989.

Tsirukis, A. G. and G. V. Reklaitis, "Feature Extraction Algorithms for Constrained Global Optimization I. Mathematical Inundation", A ^ ^*of Operations Research,* 42, 229, (1993).

Tsimkis, A. G. and G. V. Reklaitis, "Feature Extraction Algorithms for Constrained Optimization II. Batch Process Scheduling Application**, *Annals of Operations Research,* 42,275 (1993).

Turkay, M. and I.E. Grossmann, "A Logic Based Outer-Approximation Algorithm for MINLP Optimization of Process Flowsheets", to bepresented AlChE Annual Meeting, San Francisco (1994).

Tuy, H., Thieu, T. V., and Thai, N. Q., "A Conical Algorithm for Globally Minimizing a Concave Function over a Closed Convex Set", *Mathematics of Operations Research,* VoL 10, pp. 498-514,198S.

Tuy, H., "Global Minimum of a Difference of Two Convex Functions", *Mathematical Programming Study,* Vol. 30, pp. 150-182, 1987.

Vaidyanathan R. And M. El-Halwagi, "Global Optimization of Nonconvex Nonlinear Programs via Interval Analysis", Computers and Chemical Engineering, in press (1994).

Viswanathan, J. and IJE. Grossmann, "A Combined Penalty Function and Outer-Approximation Method for MINLP Optimization," *Comput. chem. Engng.* 14,769 (1990)

Visweswaran, G, and C A. Floudas, "A Global Optimization Algorithm (GOP) for Certain Classes of Nonconvex NLPs - II. Application of Theory and Test Problems", *Computers and Chemical Engineering,* 14 (12), 1419-1434 (1990).

Visweswaran, V., and Floudas, C. A., "Unconstrained and Constrained Global Optimization of Polynomial Functions in One Variable", *Journal of Global Optimization,* VoL 2, pp. 73-100,1992.

Visweswaran, V. and C. A. Floudas, '[New Properties and Computational Improvement of the GOP Algorithm for Problems with Quadratic Objective Function and Constraints", *Journal of Global Optimization,* Vol. 3, No. 4, pp. 439-462 (1993).

Visweswaran V. and C. A. Floudas, "An MILP Reformulation of the Relaxed Duals in the GOP Algorithm", manuscriopt in preparation (1994).

Westerberg A. W., and J. V. Shah, "Assuring a Global Mnumum by the Use of an Upper Bound on the Lower (Dual) Bound", *Computers Chan. Engng.*, 2,83 (1978).

Williams, H. P., "Model Building in Mathematical Programming," John Wiley, Chichester (1988).