Logic-based Modeling and Solution for Discrete/Continuous Problems in Process Systems Engineering

Ignacio E. Grossmann
Carnegie Mellon University
Pittsburgh, PA 15213

grossmann@cmu.edu
Outline

1. Introduction Mathematical Programming
2. Mixed-integer Linear Programming (MILP)
3. Propositional Logic and Disjunctions
4. Mixed-integer Nonlinear Programming (MINLP)
5. Generalized Disjunctive Programming (GDP)
6. Constraint Programming (CP)

Examples in Process Synthesis, Planning and Scheduling
Motivation

• Discrete/Continuous Optimization
  - Nonlinear models
  - 0-1 and continuous decisions

• Optimization Models
  - Mixed-Integer Linear Programming (MILP)
  - Mixed-Integer Nonlinear Programming (MINLP)

Alternative approaches:
  - Logic-based: Generalized Disjunctive Programming (GDP)
  - Constraint Programming (CP)

• Challenges
  - How to develop “best” model?
  - How to improve relaxation?
  - How to solve nonconvex GDP problems to global optimality?
  - How to overcome computational complexity?
Mathematical Programming

\[
\text{min } f(x, y) \quad \text{Cost}
\]
\[
s.t. \quad h(x, y) = 0 \quad \text{Process equations}
\]
\[
g(x, y) \leq 0 \quad \text{Specifications}
\]
\[
x \in X \quad \text{Continuous variables}
\]
\[
y \in \{0,1\} \quad \text{Discrete variables}
\]

Continuous optimization

Linear programming: \textit{LP}

Nonlinear programming: \textit{NLP}

Discrete optimization

Mixed-integer linear programming: \textit{MILP}

Mixed-integer nonlinear programming: \textit{MINLP}
Modeling systems

Mathematical Programming

GAMS (Meeraus et al, 1997)

AMPL (Fourer et al., 1995)

AIMSS (Bisschop et al. 2000)

1. Algebraic modeling systems => pure equation models
2. Indexing capability => large-scale problems
3. Automatic differentiation => no derivatives by user
4. Automatic interface with
   LP/MILP/NLP/MINLP solvers

Constraint Programming

OPL (ILOG), CHIP (Cosytech), Eclipse
$TITLE Test Problem
* Assignment problem for heat exchangers from pp.409-410 in
* "Optimization of Chemical Processes" by Edgar and Himmelblau

SETS
    I   streams       / A, B, C, D /       J   exchangers    / 1*4 / ;

TABLE  C(I,J)   Cost of assigning stream i to exchanger j

    1  2  3  4
  A  94 1 54 68
  B  74 10 88 82
  C  73 88  8 76
  D  11 74 81 21 ;

VARIABLES  X(I,J), Z ;
BINARY VARIABLES X(I,J);

EQUATIONS  ASSI(J), ASSJ(I), OBJ;

OBJ..      Z =E= SUM( (I,J), C(I,J)*X(I,J) ) ;
ASSI(J)..   SUM( I, X(I,J) ) =E= 1;
ASSJ(I)..   SUM( J, X(I,J) ) =E= 1;

MODEL HEAT / ALL / ;
SOLVE HEAT USING MIP MINIMIZING Z;
Linear Programming

**LP:** Algorithms:
- **Simplex** (Dantzig, 1949; Kantorovich, 1938)
- **Interior Point** (Karmarkar, 1988; Marsten et al., 1990)

**Major codes:**
- **CPLEX** (ILOG) (Bixby)
- **XPRESS** (Dash Optimization) (Beale, Daniel)
- **OSL** (IBM) (Forrest, Tomlin)

*Simplex:* up to 50,000 rows (constraints), 1,000,000 vars
*Interior Point:*
  - up to 500,000 rows (constraints), 500,000 vars
  - typically 20-40 Newton iterations regardless size
*Only limitation very large problems >500,000 constr*
MILP

\[
\begin{align*}
\min \ Z &= a^T y + b^T x \\
\text{st} \quad Ay + Bx &\leq d \\
y &\in \{0,1\}^m, \ x \geq 0
\end{align*}
\]

Objective function

Constraints

Theory for Convexification

Lovacz & Schrijver (1989), Sherali & Adams (1990),
Balas, Ceria, Cornuejols (1993)

Branch and Bound

Beale (1958), Balas (1962), Dakin (1965)

Cutting planes

Gomory (1959), Balas et al (1993)

Branch and cut

Johnson, Nemhauser & Savelsbergh (2000)

"Good" formulation crucial!  \Rightarrow Small LP relaxation gap

Drawback: exponential complexity
1. **Multiple choice**
   - *At least one*
     \[ \sum_{i \in I} y_i \geq 1 \]
   - *Exactly one*
     \[ \sum_{i \in I} y_i = 1 \]
   - *At most one*
     \[ \sum_{i \in I} y_i \leq 1 \]

2. **Implication**
   - *If select i then select k*
     \[ y_i - y_k \leq 0 \]
   - *Select i if and only if select k*
     \[ y_i - y_k = 0 \]

3. **Integer numbers**
   - \[ n = \sum_{k=1}^{N} ky_k, \quad \sum_{k=1}^{N} y_k = 1 \]
   - Also \[ n = \sum_{k=1}^{M} 2^k y_k \]
     - Fewer 0-1 variables
     - Weaker relaxation
Discontinuous Functions/Domains

a) **Domain**

\[ x = \begin{cases} 
0 & \text{IF } y = 0 \\
L \leq x \leq U & \text{IF } y = 1 
\end{cases} \]

\[ y = 0 \quad \Rightarrow \quad \begin{array}{c}
0 \\
L \\
y = 1 \\
U \\
y = 0
\end{array} \]

b) **Function**

\[ C = \begin{cases} 
0 & \text{IF } y = 0 \\
\alpha + \beta x & \text{IF } y = 1 
\end{cases} \]

\[ y = 1 \quad \Rightarrow \quad C = \alpha y + \beta x \]

\[ 0 \leq x \leq U y \]

\[ y = 0, 1 \]
Simple Minded Approaches

**Exhaustive Enumeration**

SOLVE LP’S FOR ALL 0-1 COMBINATIONS \((2^m)\)

<table>
<thead>
<tr>
<th>(m)</th>
<th># Combinations</th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
<td>32</td>
</tr>
<tr>
<td>100</td>
<td>(10^{30})</td>
</tr>
<tr>
<td>10,000</td>
<td>(10^{3000})</td>
</tr>
</tbody>
</table>

**Relaxation and Rounding**

SOLVE MILP WITH \(0 \leq y \leq 1\)

If solution not integer round closest

RELAXATION

Only special cases yield integer optimum (*Assignment Problem*)

Relaxed LP provides **LOWER BOUND** to MILP solution

Difference: Relaxation gap
ROUNDING
May yield **infeasible** or **suboptimal** solution

Integer Optimum

Relaxed optimum

INFEASIBLE !

Rounded Infeasible

SUBOPTIMAL !

Relaxed optimum

Rounded feasible
Convert MIP into a Continuous NLP

Example: \( \text{Min } Z = y_1 + 2y_2 \)

\[
\begin{align*}
\text{s.t. } & \quad 2y_1 + y_2 \geq 1 \\
& \quad y_1 = 0, 1 \quad y_2 = 0, 1
\end{align*}
\]

replace 0 – 1 conditions by

\[
\begin{align*}
0 \leq y_1 \leq 1, \quad y_1 (1-y_1) & \leq 0 \\
0 \leq y_2 \leq 1, \quad y_1 (1-y_2) & \leq 0
\end{align*}
\]

=> Nonlinear
Nonconvex!

only feasible pts.

(1,0)
(0,1)
(1,1)

Using CONOPT2

st. point \( y_1 = 0, \ y_2 = 0 \) => infeasible

st. pt. \( y_1 = 0.5, \ y_2 = 0.5 \) => \( y_1 = 0 \ y_2 = 1 \)
\[
Z = 2 \quad \text{suboptimal}
\]

\[
\begin{align*}
\text{correct solution} \quad y_1 = 1, \ y_2 = 0 \quad Z = 1
\end{align*}
\]
Branch and Bound

Tree Enumeration
Solve LP At Each Node

\[ \begin{align*}
\text{Min} & \quad y_1 + 2y_2 \\
\text{s.t.} & \quad 2y_1 + y_2 \geq 1 \quad (P) \\
& \quad y_1 = 0, 1, \quad y_2 = 0, 1
\end{align*} \]

Solve MILP with

\[ \begin{align*}
0 \leq y_1 \leq 1 & \quad y_1 = 0.5 \\
0 \leq y_2 \leq 1 & \quad y_2 = 0
\end{align*} \]

\[ Z = 0.5 \text{ Lower Bound} \]

Fix \( y_1 = 0 \)

Fix \( y_1 = 1 \)

\[ y_2 = 1, \quad Z = 2 \quad \text{OPTIMUM} \]

\[ y_2 = 0 \quad Z = 1 \]
Major Solution Approaches MILP

I. Enumeration

**Branch and bound**

*Land, Doig (1960) Dakin (1965)*

Basic idea: partition successively integer space to determine whether subregions can be eliminated by solving relaxed LP problems.

II. Convexification

**Cutting planes**

*Gomory (1958) Crowder, Johnson, Padberg (1983), Balas, Ceria, Cornjuelos (1993)*

Basic idea: solve sequence relaxed LP subproblems by adding valid inequalities that cut-off previous solutions.

**Remark**

- Branch and bound most widely used
- Recent trend to integrate it with cutting planes

BRANCH-AND-CUT
Branch and Bound
Partitioning Integer Space Performed with Binary Tree

Note: 15 nodes for $2^3=8$ 0-1 combinations

Node $k$ descendent node $\ell$
NODE k: LP

\[
\begin{align*}
\min Z &= c^T x + b^T y \\
\text{s.t.} & \quad Ax + By \leq d \\
& \quad x \geq 0 \quad 0 \leq y \leq 1 \\
& \quad y_i = 0 \text{ or } 1 \quad i \in I_k
\end{align*}
\]

Since node k descendent of node ℓ

1. IF LP\(\ell\) INFEASIBLE THEN LP\(k\) INFEASIBLE

2. IF LP\(k\) FEASIBLE \(Z^\ell \leq Z^k\)
   monotone increase objective
   \(Z^\ell\) : LOWER BOUND

3. IF LP\(k\) INTEGER \(Z^k \leq Z^*\)
   \(Z^k\) : UPPER BOUND

FATHOMING RULES: If node is infeasible
   If Lower Bound exceeds Upper Bound
Mixed-integer Linear Programming

- Well known & widely applied
- Efficient algorithms for moderately sized problems
- Search is based on solution of relaxed problems

**BRANCH AND BOUND**

\[
\begin{align*}
\text{max} & \quad 3x_1 + 4x_2 \\
4x_1 + 3x_2 & \leq 10 \\
x_1 & \in \{0, 1, 2, 3\}, \quad x_2 \in \{0, 1\}
\end{align*}
\]

\[(x_1, x_2) = (1.75, 1), \quad Z^L_P = 9.25\]

\[(1, 1)\text{-INT}, \quad Z^L_P = 7.0\]

\[(2, 0)\text{-INT}, \quad Z^L_P = 6\]

\[(2, 0.3)\]

\[Z^L_P = 7.2\]

INFEASIBLE
Mixed-integer Linear Programming

- Well known & widely applied
- Efficient algorithms for moderately sized problems
- Search is based on solution of relaxed problems

**CUTTING PLANES**

\[
\begin{align*}
\text{max } & \quad 3x_1 + 4x_2 \\
4x_1 + 3x_2 & \leq 10 \\
x_1 & \in \{0,1,2,3\}, \quad x_2 \in \{0,1\}
\end{align*}
\]

1. Add cut (C1) and resolve
2. Add cut (C2) and resolve
3. \(z_{LP}(3): \text{OPTIMAL}\)

- Solve entire problem at each node
- Exploit optimization information at each node
Example of MILP model for 4 component mixture

Separate mixture of A(lightest), B, C, D (heaviest) into pure components using sharp separators.

F_i flows, y_i existence columns

Network superstructure for 4 component example.
Data for example problem

a) Initial field

\[ F_{\text{TOT}} = 1000 \text{ kgmol/hr} \]

Composition (mole fraction)

<table>
<thead>
<tr>
<th></th>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>0.15</td>
<td>0.3</td>
<td>0.35</td>
<td>0.2</td>
</tr>
</tbody>
</table>

b) Economic data and heat duty coefficients

<table>
<thead>
<tr>
<th></th>
<th>Investment cost</th>
<th>Heat duty coefficients, ( \beta_k ), variable</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Separator, ( \alpha_k ), fixed,</td>
<td></td>
</tr>
<tr>
<td>K_k</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>A/BCD</td>
<td>145</td>
</tr>
<tr>
<td>2</td>
<td>AB/CD</td>
<td>52</td>
</tr>
<tr>
<td>3</td>
<td>ABC/D</td>
<td>76</td>
</tr>
<tr>
<td>4</td>
<td>A/BC</td>
<td>25</td>
</tr>
<tr>
<td>5</td>
<td>AB/C</td>
<td>44</td>
</tr>
<tr>
<td>6</td>
<td>B/CD</td>
<td>38</td>
</tr>
<tr>
<td>7</td>
<td>BC/D</td>
<td>66</td>
</tr>
<tr>
<td>8</td>
<td>A/B</td>
<td>112</td>
</tr>
<tr>
<td>9</td>
<td>B/C</td>
<td>37</td>
</tr>
<tr>
<td>10</td>
<td>C/D</td>
<td>58</td>
</tr>
</tbody>
</table>

Cost of utilities:

Cooling water \( C_C = 1.3 \times 10^3 \$\text{hr}/10^6 \text{kJyr} \)

Steam \( C_H = 34 \times 10^3 \$\text{hr}/10^6 \text{kJyr} \)
Split fractions in superstructure (using initial compositions)

\[
\begin{align*}
\xi_1^A &= 0.15 \\
\xi_1^{BCD} &= 0.85 \\
\xi_2^{AB} &= 0.45 \\
\xi_2^{CD} &= 0.55 \\
\xi_3^{ABC} &= 0.8 \\
\xi_3^D &= 0.2 \\
\xi_4^B &= 0.353 \\
\xi_4^{CD} &= 0.647 \\
\xi_5^{BC} &= 0.765 \\
\xi_5^D &= 0.235 \\
\xi_6^A &= 0.188 \\
\xi_6^{BC} &= 0.812 \\
\xi_7^{AB} &= 0.5625 \\
\xi_7^C &= 0.44 \\
\xi_8^C &= 0.636 \\
\xi_8^D &= 0.364 \\
\xi_9^B &= 0.462 \\
\xi_9^C &= 0.538 \\
\xi_{10}^A &= 0.333 \\
\xi_{10}^B &= 0.667
\end{align*}
\]
MILP model

Initial node in network

\[ F_1 + F_2 + F_3 = 1000 \quad (1) \]

For the remaining nodes in the network, mass balances for each intermediate product. Based on the recovery fractions, the mass balance for each intermediate product is as follows:

a) Intermediate (BCD) which is produced in column 1, and directed to columns 4 and 5,

\[ F_4 + F_5 - 0.85 F_1 = 0 \quad (2) \]

b) Intermediate (ABC) which is produced in column 3, and directed to columns 6 and 7,

\[ F_6 + F_7 - 0.8 F_3 = 0 \quad (3) \]

c) Intermediate (AB) which is produced in columns 2 and 7, and directed to column 10,

\[ F_{10} - 0.45 F_2 - 0.563 F_7 = 0 \quad (4) \]

d) Intermediate (BC) which is produced in columns 5 and 6, and directed to column 9,

\[ F_9 - 0.765 F_5 - 0.812 F_6 = 0 \quad (5) \]

e) Intermediate (CD) which is produced in columns 2 and 4, and directed to column 8,

\[ F_8 - 0.55 F_2 - 0.647 F_4 = 0 \quad (6) \]
Relating flows to the binary variables $y$:

$$F_k - 1000 \, y_k \leq 0,$$

$$F_k \geq 0, \quad y_k = 0,1, \quad k = 1, \ldots, 10 \quad (7)$$

Heat duties of condensers and reboliers, continuous variables $Q_k$, $k = 1, \ldots, 10$,

$$Q_k = K_k \, F_k, \quad k = 1, \ldots, 10 \quad (8)$$

where the parameters $K_k$ are given in Table.

Objective function, minimization of the sum of the costs in the 10 columns.

$$\min \ C = \sum_{k=1}^{10} (\alpha_k y_k + \beta_k F_k) + (34 + 1.3) \sum_{k=1}^{10} Q_k$$

cost coefficients $\alpha_k$, $\beta_k$, are given in Table.
Optimal separation sequence

Second best solution. Make $y_2 = y_8 = y_{10} = 1$ infeasible

$y_2 + y_8 + y_{10} \leq 2$

Second best sequence
General MILP model for distillation sequences

Sets

a) \( IP = \{ m \mid m \text{ is an intermediate product} \} \)
   e.g. \( IP = \{(ABC), (BCD), (AB), (BC), (CD)\} \)

b) \( COL = \{ k \mid k \text{ is a column in the superstructure} \} \)
   e.g. \( COL = \{1,2,...,9,10\} \)

c) \( FS_F = \{ \text{columns } k \text{ that have as feed the initial mixture} \} \)
   e.g. \( FS_F = \{1,2,3\} \)

d) \( FS_m = \{ \text{columns } k \text{ that have as feed intermediate } m \} \)
   e.g. for \( m = (BCD) \), \( FS_m = \{4,5\} \)

e) \( PS_m = \{ \text{columns } k \text{ that produce intermediate } m \} \)
   e.g. for \( m = (CD) \), \( PS_m = \{2,4\} \)
\[
\text{min} \quad C = \sum_{k \in \text{COL}} \left[ \alpha_k y_k + \beta_k F_k \right] + (C_H + C_C) Q_k \\
\text{s.t.} \quad \sum_{k \in \text{FS}_F} F_k = F_{\text{TOT}} \\
(16) \quad \sum_{k \in \text{FS}_m} F_k - \sum_{k \in \text{PS}_m} \xi_k F_k = 0 \quad m \in \text{IP} \\
Q_k - K_k F_k = 0 \quad k \in \text{COL} \\
F_k - U y_k \leq 0 \quad k \in \text{COL} \\
F_k, Q_k \geq 0, y_k = 0, 1 \quad k \in \text{COL}
\]

where \( F_{\text{TOT}} \) is the flowrate of the initial mixture, \( \xi_k^m \) are the recoveries of intermediate \( m \) in column \( k \), and \( U \) is an upper bound for the flowrates which for simplicity we can select as \( F_{\text{TOT}} \).
“If sufficient care is exercised, it is now possible to solve MILP models of size approaching ‘large’ LP’s. Note, however, that ‘sufficient care’ is the operative phrase”.

JOHN TOMLIN (1983)

HOW TO MODEL INTEGER CONSTRAINTS?

Propositional Logic
Disjunctions
LITERAL IN PROPOSITIONAL LOGIC \( P_i \) TRUE
NEGATION \( \neg P_i \) FALSE

Example \( P_i \): select unit I, execute task j

PROPOSITION: set of literals \( P_i \) separated by OR, AND IMPLICATION

**Representation Linear 0-1 Inequalities**

ASSIGN binary \( y_i \) to \( P_i \) \((1 - y_i)\) to \( \neg P_i \)

**OR** \( P_1 \lor P_2 \lor \ldots \lor P_r \) \( y_1 + y_2 + \ldots + y_r \geq 1 \)

**AND** \( P_1 \land P_2 \land \ldots \land P_r \) \( y_1 \geq 1, y_2 \geq 1, \ldots, y_r \geq 1 \)

**IMPLICATION** \( P_1 \Rightarrow P_2 \)
**EQUIVALENT TO** \( \neg P_1 \lor P_2 \) \( 1 - y_1 + y_2 \geq 1 \)

**OR** \( y_2 \geq y_1 \)
EQUIVALENCE $P_1 \iff P_2$

$(P_1 \implies P_2) \land (P_2 \implies P_1)$

EQUIVALENT TO $(\neg P_1 \lor P_2) \land (\neg P_2 \land P_1)$

$1 - y_1 + y_2 \geq 1 \quad 1 - y_2 + y_1 \geq 1$

OR $y_1 = y_2$
Systematic Procedure to Derive Linear Inequalities for Logic Propositions

Goal is to Convert Logical Expression into Conjunctive Normal Form (CNF)

\[ Q_1 \land Q_2 \land \ldots \land Q_s \]

where clause \( Q_i : P_1 \lor P_2 \lor \ldots \lor P_r \)  \( (Note: \ all \ OR) \)

**BASIC STEPS**

1. REPLACE IMPLICATION BY DISJUNCTION

   \[ P_1 \Rightarrow P_2 \iff \neg P_1 \lor P_2 \]

2. MOVE NEGATION INWARD APPLYING DE MORGAN'S THEOREM

   \[ \neg (P_1 \land P_2) \iff \neg P_1 \lor \neg P_2 \]
   \[ \neg (P_1 \lor P_2) \iff \neg P_1 \land \neg P_2 \]

3. RECURSIVELY DISTRIBUTE OR OVER AND

   \[ (P_1 \land P_2) \lor P_3 \iff (P_1 \lor P_3) \land (P_2 \lor P_3) \]
EXAMPLE

\[ \text{flash} \Rightarrow \text{dist} \lor \text{abs} \]
\[ \text{memb} \Rightarrow \neg \text{abs} \land \text{comp} \]

\[ \text{PF} \Rightarrow \text{PD} \lor \text{PA} \quad (1) \]
\[ \text{PM} \Rightarrow \neg \text{PA} \land \text{PC} \quad (2) \]

(1) \[ \neg \text{PF} \lor \text{PD} \lor \text{PA} \quad \text{remove implication} \]

\[ 1 - y_F + y_D + y_A \geq 1 \]
\[ y_D + y_A \geq y_F \]

(2) \[ \neg \text{PM} \lor (\neg \text{PA} \land \text{PC}) \quad \text{remove implication} \]

\[ (\neg \text{PM} \lor \neg \text{PA}) \land (\neg \text{PM} \lor \text{PC}) \quad \text{distribute OR over AND} \quad \Rightarrow \text{CNF!} \]

\[ 1 - y_M + 1 - y_A \geq 1 \]
\[ 1 - y_M + y_C \geq 1 \]
\[ y_M + y_A \leq 1 \]
\[ y_C \geq y_M \]

\[ \begin{align*}
  y_D + y_A & \geq y_F \\
  y_M + y_A & \leq 1 \\
  y_C & \geq y_M
\end{align*} \]

Verify:
\[ y_F = 1 \quad y_D + y_A \geq 1 \quad y_F = 0 \quad y_D + y_A \geq 0 \]
\[ y_M = 1 \quad \Rightarrow \quad y_A = 0 \quad y_C = 1 \]
EXAMPLE

if $x = 1$ and $y = 1$ then $z = 1$

if $x = 0$ and $y = 0$ then $z = 1$

if $x = 1$ and $y = 0$ then $z = 0$

if $x = 0$ and $y = 1$ then $z = 0$

1. $x \land y \rightarrow z$

2. $\neg x \land \neg y \rightarrow z$

3. $x \land \neg y \rightarrow \neg z$

4. $\neg x \land y \rightarrow \neg z$

1) $x \land y \rightarrow z \iff \neg (x \land y) \lor z \iff \neg x \lor \neg y \lor z$

$1 - x + 1 - y + z \geq 1$
2) \( \neg x \land \neg y \rightarrow z \iff \neg (\neg x \land \neg y) \lor z \iff x \lor y \lor z \quad x + y + z \geq 1 \)

3) \( x \land \neg y \rightarrow \neg z \iff \neg (x \land \neg y) \lor \neg z \iff \neg x \lor y \lor \neg z \quad 1 - x + y + 1 - z \geq 1 \)

4) \( \neg x \land y \rightarrow \neg z \iff \neg (\neg x \land y) \lor \neg z \iff x \lor \neg y \lor \neg z \quad x + 1 - y + 1 - z \geq 1 \)

\[ \begin{align*}
    z \geq & x + y - 1 \\
    z \geq & 1 - x - y \\
    z \leq & 1 - x + y \\
    z \leq & 1 + x - y
\end{align*} \]
EXAMPLE

Integer Cut

Constraint that is infeasible for integer point
\[ y_i = 1 \quad i \in B \]
and feasible for all other integer points

\[ y_i = 0 \quad i \in N \]

*Balas and Jeroslow (1968)*
Example: Multiperiod Problems

“If Task $y_i$ is performed in any time period $i = 1, ..n$ select Unit $z$”

**Intuitive Approach**

\[ y_1 + y_2 + \ldots + y_n \leq n*z \]  \hspace{1cm} (1)

**Logic Based Approach**

\[ y_1 \lor y_2 \lor \ldots \lor y_n \Rightarrow z \]

\[ \neg(y_1 \lor y_2 \lor \ldots \lor y_n) \lor z \]

\[ (\neg y_1 \land \neg y_2 \land \ldots \land \neg y_n) \lor z \]

\[ (\neg y_1 \lor z) \land (\neg y_2 \lor z) \land \ldots \land (\neg y_n \lor z) \]

\[ 1 - y_1 + z \geq 1 \quad 1 - y_2 + z \geq 1 \quad 1 - y_n + z \geq 1 \]

\[
\begin{align*}
y_1 & \leq z \\
y_2 & \leq z \\
\vdots \\
y_n & \leq z
\end{align*}
\]

*Inequalities in (2) are stronger than inequalities in (1)*
Geometrical interpretation

\[ y_1 \leq z \]
\[ y_2 \leq z \]

All extreme points in hypercube are integer!
Geometrical interpretation

\[ y_1 + y_2 \leq 2z \]

Non-integer extreme points

Weaker relaxation!
Modeling of Disjunctions

\[ \bigvee_{i \in D} \left[ A_i x \leq b_i \right] \quad \text{one inequality must hold} \]

Example: A before B OR B before A

\[ \left[ TS_A + pt_A \leq TS_B \right] \lor \left[ TS_B + pt_B \leq TS_A \right] \]

**Big M Formulation**

\[ A_i x \leq b_i + M_i (1 - y_i) \quad i \in D \]

\[ \sum_{i \in D} y_i = 1 \]

**Difficulty: Parameter M_i**

Must be sufficiently large to render inequality redundant

Large value yields poor relaxation
Convex-hull Formulation  (Balas, 1985)

\[
x = \sum_{i \in D} z_i \quad \text{disaggregation vars.}
\]

\[
A_i z_i \leq b_i y_i \quad i \in D
\]

\[
\sum_{i \in D} y_i = 1
\]

\[
0 \leq z_i \leq Uy_i \quad i \in D \quad \text{(may be removed)}
\]

\[
y_i = 0, 1
\]

**Derivation**

\[
A_i x y_i \leq b_i y_i \quad i \in D \quad \text{(B) nonlinear disj. equiv.}
\]

\[
\sum_{i \in D} y_i = 1
\]
Let $z_i = x \cdot y_i$ be the disaggregated variable.

$$\sum_{i \in D} z_i = \sum_{i \in D} xy_i = x \sum_{i \in D} y_i$$

since $\sum_{i \in D} y_i = 1 \implies \sum_{i \in D} z_i = x$ \hspace{1cm} (A)

To ensure $z_i = 0$ if $y_i = 0$

$$0 \leq z_i \leq Uy_i$$ \hspace{1cm} (C)

(A) $\implies x = \sum_{i \in D} z_i$

Substitute (B) $A_i z_i \leq b_i y_i \quad i \in D$

$$\sum_{i \in D} y_i = 1$$

(C) $\implies 0 \leq z_i \leq Uy_i \quad i \in D$
Example

\[
[x_1 - x_2 \leq -1] \lor \left[ -x_1 + x_2 \leq 1 \right] \\
0 \leq x_1, x_2 \leq 4
\]

big M

\[
x_1 - x_2 \leq -1 + M (1 - y_1) \\
-x_1 + x_2 \leq -1 + M (1 - y_2) \\
y_1 + y_2 = 1 \quad M = 10 \text{ possible choice}
\]

Convex hull

\[
x_1 = z_1^1 + z_1^2 \\
x_2 = z_2^1 + z_2^2 \\
z_1^1 - z_2^1 \leq -y_1 \\
-z_1^2 + z_2^2 \leq -y_2 \\
y_1 + y_2 = 1 \\
0 \leq z_1^1 \leq 4 \quad y_1 \\
0 \leq z_1^2 \leq 4 \quad y_2 \\
0 \leq z_2^1 \leq 4 \quad y_1 \\
0 \leq z_2^2 \leq 4 \quad y_2
\]
Nonlinear Programming

NLP: Algorithms (variants of Newton's method)
    Successive quadratic programming (SQP) (Han 1976; Powell)
    Reduced gradient
    Interior Point Methods

Major codes:
    MINOS (Murtagh, Saunders, 1978, 1982)
    CONOPT (Drud, 1994)
    SQP: SNOPT (Murray, 1996) OPT (Biegler, 1998)
    IP: IPOPT (Wachter, Biegler, 2002)  www.coin-or.org

Typical sizes: 50,000 vars, 50,000 constr. (unstructured)
    500,000 vars (few degrees freedom)

Convergence: Good initial guess essential (Newton's)
    Nonconvexities: Local optima, non-convergence
MINLP

- Mixed-Integer Nonlinear Programming

\[
\begin{align*}
\text{min } Z &= f(x, y) \\
\text{s.t. } g(x, y) &\leq 0 \\
x &\in X, \ y \in Y
\end{align*}
\]

\[
X = \{x \mid x \in \mathbb{R}^n, x^L \leq x \leq x^U, Bx \leq b\}
\]

\[
Y = \{y \mid y \in \{0, 1\}^m, Ay \leq a\}
\]

- \(f(x,y)\) and \(g(x,y)\) - assumed to be convex and bounded over \(X\).
- \(f(x,y)\) and \(g(x,y)\) commonly \textbf{linear} in \(y\)
Solution Algorithms

- **Branch and Bound method (BB)**
  Branch and cut: Stubbs and Mehrotra (1999)

- **Generalized Benders Decomposition (GBD)**
  Geoffrion (1972)

- **Outer-Approximation (OA)**

- **LP/NLP based Branch and Bound**
  Quesada and Grossmann (1992)

- **Extended Cutting Plane (ECP)**
  Westerlund and Pettersson (1995)
Basic NLP subproblems

a) NLP Relaxation *Lower bound*
\[
\begin{align*}
\min Z_{LB}^k &= f(x, y) \\
\text{s.t.} & \quad g_j(x, y) \leq 0 \quad j \in J \\
& \quad x \in X, \ y \in Y_R \\
& \quad y_i \leq \alpha_i^k \quad i \in I_{FL}^k \\
& \quad y_i \geq \beta_i^k \quad i \in I_{FU}^k
\end{align*}
\]  
(NLP1)

b) NLP Fixed \( y^k \) *Upper bound*
\[
\begin{align*}
\min Z_U^k &= f(x, y^k) \\
\text{s.t.} & \quad g_j(x, y^k) \leq 0 \quad j \in J \\
& \quad x \in X
\end{align*}
\]  
(NLP2)

c) Feasibility subproblem for fixed \( y^k \).
\[
\begin{align*}
\min u \\
\text{s.t.} & \quad g_j(x, y^k) \leq u \quad j \in J \\
& \quad x \in X, \ u \in R^l
\end{align*}
\]  
(NLPF) 

Infinity-norm
Cutting plane MILP master

(Duran and Grossmann, 1986)

Based on solution of K subproblems \((x^k, y^k)\) \(k=1,...K\)

**Lower Bound**

M-MIP

\[
\begin{align*}
\min \ Z^k_L &= \alpha \\
\text{st} \quad \alpha &\geq f(x^k, y^k) + \nabla f(x^k, y^k)^T \begin{bmatrix} x - x^k \\ y - y^k \end{bmatrix} \\
&\quad + \nabla g_j(x^k, y^k)^T \begin{bmatrix} x - x^k \\ y - y^k \end{bmatrix} \leq 0 \quad j \in J \\
x \in X, \ y \in Y
\end{align*}
\]

**Notes:**

a) Point \((x^k, y^k)\) \(k=1,...K\) normally from NLP2

b) Linearizations *accumulated* as iterations K increase

c) Non-decreasing sequence lower bounds
Linearizations and Cutting Planes

Underestimate Objective Function

Overestimate Feasible Region
Branch and Bound

\[
\text{NLP1:} \quad \min Z_{LB}^k = f(x, y) \\
\text{s.t.} \quad g_j(x, y) \leq 0 \quad j \in J
\]

Tree Enumeration

\[ x \in X, \quad y \in Y_R \]
\[ y_i \leq \alpha_i^k \quad i \in I_{FL}^k \]
\[ y_i \geq \beta_i^k \quad i \in I_{FU}^k \]

Successive solution of NLP1 subproblems

Advantage:
Tight formulation may require one NLP1 \((I_{FL} = I_{FU} = \emptyset)\)

Disadvantage:
Potentially many NLP subproblems

Convergence global optimum:
Uniqueness solution NLP1 \((\text{sufficient condition})\)

\text{Less stringent than other methods}
**Outer-Approximation**

Alternate solution of NLP and MIP problems:

NLP2: \[ \min Z_U = f(x, y^k) \]
\[ s.t. \quad g_j(x, y^k) \leq 0 \quad j \in J \]
\[ x \in X \]

M-MIP: \[ \min Z_L^k = \alpha \]
\[ s.t. \quad \alpha \geq f(x^k, y^k) + \nabla f(x^k, y^k)^T \left[ \begin{array}{c} x - x^k \\ y - y^k \end{array} \right] \]
\[ g_j(x^k, y^k) + \nabla g_j(x^k, y^k)^T \left[ \begin{array}{c} x - x^k \\ y - y^k \end{array} \right] \leq 0 \quad j \in J^k \]
\[ x \in X, \; y \in Y \]

*Property.* Trivially converges in one iteration if \( f(x, y) \) and \( g(x, y) \) are linear

- If infeasible NLP solution of feasibility NLP-F required to guarantee convergence.
MIP Master problem need not be solved to optimality

Find new $y^{k+1}$ such that predicted objective lies below current upper bound $UB^K$:

(M-MIPF)

$$\begin{align*}
\min & \quad Z^K_L = 0\alpha \\
\text{s.t.} & \quad \alpha \leq UB^K - \varepsilon \\
& \quad \alpha \geq f(x^k, y^k) + \nabla f(x^k, y^k)^T \begin{bmatrix} x - x^k \\ y - y^k \end{bmatrix} \\
& \quad g_j(x^k, y^k) + \nabla g_j(x^k, y^k)^T \begin{bmatrix} x - x^k \\ y - y^k \end{bmatrix} \leq 0 \quad j \in J \\
& \quad x \in X, \ y \in Y
\end{align*}$$

$k = 1, \ldots, K$

Remark.

M-MIPF will tend to increase number of iterations
Generalized Benders Decomposition

Benders (1962), Geoffrion (1972)

Particular case of Outer-Approximation as applied to (P1)

1. Consider Outer-Approximation at \((x^k, y^k)\)

\[
\alpha \geq f(x^k, y^k) + \nabla f(x^k, y^k)^T \begin{bmatrix} x - x^k \\ y - y^k \end{bmatrix}
\]

\[
g(x^k, y^k) + \nabla g(x^k, y^k)^T \begin{bmatrix} x - x^k \\ y - y^k \end{bmatrix} \leq 0 \quad j \in J^k
\]  

(1)

2. Obtain linear combination of (1) using Karush-Kuhn-Tucker multipliers \(\mu^k\) and eliminating \(x\) variables

\[
\alpha \geq f(x^k, y^k) + \nabla f(x^k, y^k)^T (y - y^k)
\]

\[
+ \mu^k \left[ g(x^k, y^k) + \nabla g(x^k, y^k)^T (y - y^k) \right]
\]

\[\text{Lagrangian cut}\]

Remark. Cut for infeasible subproblems can be derived in a similar way.

\[
\lambda^k \left[ g(x^k, y^k) + \nabla g(x^k, y^k)^T (y - y^k) \right] \leq 0
\]
**Generalized Benders Decomposition**

Alternate solution of NLP and MIP problems:

NLP2: \[ \min Z_U = f(x, y^k) \]
\[ s.t. \quad g_j(x, y^k) \leq 0 \quad j \in J \]
\[ x \in X \]

M-GBD: \[ \min Z_L^k = \alpha \]
\[ s.t. \quad \alpha \geq f(x^k, y^k) + \nabla_y f(x^k, y^k)^T (y - y^k) \]
\[ + (\mu^k)^T \left[ g(x^k, y^k) + \nabla_y g(x^k, y^k)^T (y - y^k) \right] \quad k \in KFS \]
\[ \left( \lambda^k \right)^T \left[ g(x^k, y^k) + \nabla_y g(x^k, y^k)^T (y - y^k) \right] \leq 0 \quad k \in KIS \]
\[ y \in Y, \quad \alpha \in \mathbb{R}^l \]

**Property 1.** If problem (P1) has zero integrality gap, Generalized Benders Decomposition converges in one iteration when optimal \((x^k, y^k)\) are found.

\[ \Rightarrow \text{Also applies to Outer-Approximation} \]
Extended Cutting Plane

Westerlund and Pettersson (1995)

Add linearization most violated constraint to M-MIP

\[ J^k = \{ \hat{j} \in \arg \max_{j \in J} g_j (x^k, y^k) \} \]

Remarks.
- Can also add full set of linearizations for M-MIP
- Successive M-MIP's produce non-decreasing sequence lower bounds
- Simultaneously optimize \( x^k, y^k \) with M-MIP

= > Convergence may be slow
LP/NLP Based Branch and Bound (*Branch & Cut*)

Quesada and Grossmann (1992)

Integrate NLP and M-MIP problems

\[ \text{LP1} \rightarrow \text{LP2} \rightarrow \text{LP3} \rightarrow \text{LP4} \rightarrow \text{LP5} = \text{Integer} \]

- Solve NLP and update bounds open nodes

Remark.

Fewer number branch and bound nodes for LP subproblems

May increase number of NLP subproblems
Numerical Example

\[
\begin{align*}
\min\ Z &= y_1 + 1.5y_2 + 0.5y_3 + x_1^2 + x_2^2 \\
\text{s.t.} & \quad (x_1 - 2)^2 - x_2 \leq 0 \\
& \quad x_1 - 2y_1 \geq 0 \\
& \quad x_1 - x_2 - 4(1-y_2) \leq 0 \\
& \quad x_1 - (1 - y_1) \geq 0 \\
& \quad x_2 - y_2 \geq 0 \\
& \quad x_1 + x_2 \geq 3y_3 \\
& \quad y_1 + y_2 + y_3 \geq 1 \\
& \quad 0 \leq x_1 \leq 4, \quad 0 \leq x_2 \leq 4 \\
& \quad y_1, y_2, y_3 = 0, 1
\end{align*}
\]

(MIP-EX)

Optimum solution: \( y_1=0, y_2 = 1, y_3 = 0, x_1 = 1, x_2 = 1, Z = 3.5. \)
Starting point $y_1 = y_2 = y_3 = 1$.

**Summary of Computational Results**

<table>
<thead>
<tr>
<th>Method</th>
<th>Subproblems</th>
<th>Master problems (LP's solved)</th>
</tr>
</thead>
<tbody>
<tr>
<td>BB</td>
<td>5 (NLP1)</td>
<td></td>
</tr>
<tr>
<td>OA</td>
<td>3 (NLP2)</td>
<td>3 (M-MIP) (19 LP's)</td>
</tr>
<tr>
<td>GBD</td>
<td>4 (NLP2)</td>
<td>4 (M-GBD) (10 LP's)</td>
</tr>
<tr>
<td>ECP</td>
<td>-</td>
<td>5 (M-MIP) (18 LP's)</td>
</tr>
</tbody>
</table>
Example: Process Network with Fixed Charges

- Duran and Grossmann (1986)
  - Network superstructure

Diagram: A process network with fixed charges, labeled with variables such as $x_1$ to $x_{25}$. The diagram shows the flow from input to output through various units labeled 1 to 8, with flow rates indicated by arrows and variable names.
### Example  (Duran and Grossmann, 1986)

**Algebraic MINLP:**  *linear in y, convex in x*

8 0-1 variables, 25 continuous, 31 constraints (5 nonlinear)

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>NLP</th>
<th>MIP</th>
</tr>
</thead>
<tbody>
<tr>
<td>Branch and Bound <em>(F-L)</em></td>
<td>20</td>
<td>-</td>
</tr>
<tr>
<td>Outer-Approximation:</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>Generalized-Benders</td>
<td>10</td>
<td>10</td>
</tr>
<tr>
<td>Extended Cutting Plane</td>
<td>-</td>
<td>15</td>
</tr>
<tr>
<td>LP/NLP based</td>
<td>3</td>
<td>7 LP's vs 13 LP's OA</td>
</tr>
</tbody>
</table>
Effect of nonconvexities

1. NLP subproblems may not have unique local optimum

2. MIP-master problem may not predict rigorous lower bounds

Handling of nonconvexities

I. Rigorous approach
   a) Consider structured NLP
   b) Develop convex underestimator => Convex MINLP
   c) Solve convex MINLP within global search for continuous variables

II. Heuristic strategies for unstructured NLP
   1. Redefine MIP master introducing slacks to allow violation of linearizations (augmented penalty)
   2. Drop linearizations that produce violations in previous search points
Handling nonlinear equations

\[ h(x,y) = 0 \]

1. In GBD no special provision needed


\[
T^k \nabla h \left( x^k, y^k \right) T \begin{bmatrix} x-x^k \\ y-y^k \end{bmatrix} \leq 0
\]

\[ T^k = \{ t^k_i \}, \quad t^k_{ii} = \text{sign}(\lambda^k_i), \quad \lambda^k_i \text{ multiplier equation } h_f(x, y) = 0 \]

Remark.
Rigorous if equations relax as \( h(x,y) \leq 0 \), \( h(x,y) \) convex.
Otherwise may cut-off optimum
MIP-Master Augmented Penalty
Viswanathan and Grossmann, 1990

Slacks: \( p_k, q_k \) with weights \( w_k \)

\[
\min Z^K = \alpha + \sum_{k=1}^{K} \left[ w_k p_k^k + w_k q_k^k \right] \tag{M-APER}
\]

\[
s.t. \quad \alpha \geq f(x^k, y^k) + \nabla f(x^k, y^k)^T \begin{bmatrix} x-x^k \\ y-y^k \end{bmatrix}
\]

\[
\quad T^k \nabla h(x^k, y^k)^T \begin{bmatrix} x-x^k \\ y-y^k \end{bmatrix} \leq p^k \quad k=1,\ldots,K
\]

\[
\quad g(x^k, y^k) + \nabla g(x^k, y^k)^T \begin{bmatrix} x-x^k \\ y-y^k \end{bmatrix} \leq q^k
\]

\[
\quad \sum_{i \in B^k} y_i - \sum_{i \in N^k} y_i \leq |B^k| - 1 \quad k=1,\ldots,K
\]

\[
\quad x \in X, \quad y \in Y, \quad \alpha \in \mathbb{R}^1, \quad p_k^k, q_k^k \geq 0
\]

If convex MINLP then slacks take value of zero
\( \Rightarrow \) reduces to OA/ER

Basis DICOPT (nonconvex version)

1. Solve relaxed MINLP
2. Iterate between MIP-APER and NLP subproblem
   until no improvement in NLP
Mixed-integer Nonlinear Programming

**MINLP: Algorithms**
- Branch and Bound (BB) *Leyffer (2001), Bussieck, Drud (2003)*
- Generalized Benders Decomposition (GBD) *Geoffrion (1972)*
- Outer-Approximation (OA) *Duran and Grossmann (1986)*
- Extended Cutting Plane (ECP) *Westerlund and Pettersson (1992)*

**Codes:**
- DICOPT ++ *(GAMS)* *Viswanathan and Grossman (1990)*
- AOA *(AIMSS)*
- MINLP *(AMPL)* *Fletcher and Leyffer (1999)*
- α-ECP *Westerlund and Pettersson (1996)*
- MINOPT *Schweiger and Floudas (1998)*
- BARON *Sahinidis et al. (1998)*
- SBB *GAMS simple B&B*

**Test Problems:**
- GAMSWorld: [http://www.gamsworld.org/minlp](http://www.gamsworld.org/minlp)
- Floudas MINOPT: [http://titan.princeton.edu/MINOPT/library-tests.html](http://titan.princeton.edu/MINOPT/library-tests.html)
Parameter Estimation FTIR-Spectroscopy

Brink and Westerlund (1995)

Given multicomponent mixture n components \((i=1,..n)\) which N data are given \((k=1,..N)\) at different wave number \(j=1,..q\):

\[ c_i^k \quad \text{concentration component } i, \text{ run } k \]
\[ a_j^k \quad \text{absorbence of mixture wave number } j, \text{ run } k \]

Find number and values of parameters \(p_{ij}\) for linear correlation:

\[ c_i = \sum_{j=1}^{q} p_{ij} a_j \quad i = 1,..n \]
Derivation of Model

Error:
\[ e_i^k = c_i^k - \sum_{j=1}^{q} p_{ij} a_j^k \quad i = 1,..n \quad k = 1,.. N \]

\textit{Akaike information criterion}

\[ \min AIC = -2 \ln L + 2P \]

P number of parameters \( p_{ij} \)

L likelihood function

\[ \ln L = -\frac{N}{2} n \ln(2\pi) + \frac{N}{2} \ln(|R^{-1}|) - \frac{1}{2} \sum_{k=1}^{N} e_k^T R^{-1} e_k \]

R covariance matrix

Let \( y_{ij} = \begin{cases} 1 & \text{if } p_{ij}^L \leq p_{ij} \leq p_{ij}^U \\ 0 & \text{if } p_{ij} = 0 \end{cases} \)
MINLP Model (MIQP)

\[
\min Z = \sum_{k=1}^{N} \sum_{i=1}^{n} e_i^k R^{-1} e_i^k + 2 \sum_{i=1}^{n} \sum_{j=1}^{q} y_{ij}
\]

s.t. \( e_i^k = c_i^k - \sum_{j=1}^{q} p_{ij} a_j^k \) \( i = 1,..n \) \( k = 1,..N \)

\( p_{ij}^L y_{ij} \leq p_{ij} \leq p_{ij}^U y_{ij} \) \( i = 1,..n \) \( j = 1,..q \)

\( y_{ij} = 0,1 \) \( p_{ij} \geq 0 \)
Results

35 spectra for CO, NO, CO$_2$ mixture

Range spectra: 800-2200 cm$^{-1}$

Resolution 28 cm$^{-1}$ (100 subintervals)

MINLP: 300 0-1 vars.  3800 cont vars.
4100 constraints

Outer-Approximation: 60 CPU-sec (DEC-3000)

5 selected parameters

$P_{CO,87} = 714.78_{2016.2-2030.3}$
$P_{CO,97} = 65.07_{2157.6-2171.7}$
$P_{NO,83} = 315.57_{1959.6-1973.7}$
$P_{CO_2,10} = 22.47_{927.3-941.4}$
$P_{CO_2,20} = 25.01_{1068.7-1082.8}$

Two-stage superstructure

**Parameters**
- TIN = inlet temperature of stream
- TOUT = outlet temperature of stream
- F = heat capacity flow rate
- U = overall heat transfer coefficient
- CCU = unit cost for cold utility
- CHU = unit cost of hot utility
- CF = fixed charge for exchangers
- C = area cost coefficient
- B = exponent for area cost
- NOK = total number of stages
- Ω = upper bound for heat exchange
- Γ = upper bound for temperature difference

**Variables**
- δi,j,k = temperature approach for match (i,j) at temperature location k
- δtcui = temperature approach for the match of hot stream i and cold utility
- δthuj = temperature approach for the match of cold stream j and hot utility
- qijk = heat exchanged between hot process stream i and cold process stream j in stage k
- qcu = heat exchanged between hot stream i and cold utility
- qhu = heat exchanged between hot utility and cold stream j
- ti,k = temperature of hot stream i at hot end of stage k
- tj,k = temperature of cold stream j at hot end of stage k
- zijk = binary variable to denote existence of match (i,j) in stage k
- zcu = binary variable to denote that cold utility exchanges heat with stream i
- zhu = binary variable to denote that hot utility exchanges heat with stream j
Assumption

Isothermal mixing $\Rightarrow$ linear constraints

\[ q_{ijk} \quad t_{ik} \quad z_{ijk=0,1} \quad t_{ik+1} \]

Rigorous if no stream splits

Procedure

1. Solve MINLP assuming isothermal mixing
2. If splitting streams, solve NLP on reduced final configuration

No. stages: max \{no. hot, no. cold\}
Overall heat balance for each stream

\[(TOUT_j - TIN_j) \sum_{k \in ST} \sum_{i \in HP} q_{ijk} + qhu_j \quad j \in CP\]

\[(TIN_i - TOUT_i) \sum_{k \in ST} \sum_{j \in CP} q_{ijk} + qcu_i \quad i \in HP\]

Heat balance at each stage

\[(t_{i,k} - t_{i,k+1}) \sum_{j \in CP} q_{ijk} \quad i \in HP, k \in ST\]

\[(t_{j,k} - t_{j,k+1}) \sum_{j \in HP} q_{ijk} \quad j \in CP, k \in ST\]

Feasibility of temperatures

\[t_{i,k} \geq t_{i,k+1} \quad k \in ST, \; i \in HP\]

\[t_{j,k} \geq t_{j,k+1} \quad k \in ST, \; j \in CP\]

\[TOUT_i \geq t_{i,NOK+1} \quad i \in HP\]

\[TOUT_j \geq t_{j,1} \quad j \in CP\]

Hot and cold utility load

\[(t_{i,NOK+1} - TOUT_i) q_{ci} \quad i \in HP\]

\[(TOUT_j - t_{j,i}) q_{hu} \quad j \in CP\]
**Logical constraints**

\[ q_{ijk} - \Omega z_{ijk} \leq 0 \quad i \in HP, j \in CP, \ k \in ST \]

\[ q_{cui} - \Omega z_{cui} \leq 0 \quad i \in HP \]

\[ q_{huj} - \Omega z_{huj} \leq 0 \quad j \in CP \]

\[ z_{ijk}, z_{cui}, z_{huj} = 0,1 \]

**Calculation of approach temperatures**

\[ dt_{ijk} \leq t_{i,k} - t_{j,k} + \Gamma (1 - z_{ijk}) \quad k \in ST, i \in HP, j \in CP \]

\[ dt_{ijk+1} \leq t_{i,k+1} - t_{j,k+1} + \Gamma (1 - z_{ijk}) \quad k \in ST, i \in HP, j \in CP \]

\[ dt_{cui} \leq t_{i,NOK+1} - T_{OUTCU} + \Gamma (1 - z_{cui}) \quad i \in HP \]

\[ dt_{hui} \leq T_{OUTHU} - t_{j,1} + \Gamma (1 - z_{huj}) \quad j \in CP \]

\[ dt_{ijk} \geq EMAT \]

**Objective function**

Chen approximation (1987)

\[ LMTD \sim [(dt_1 * dt_2) * (dt_1 + dt_2) / 2]^{1/3} \]

\[ \text{min} \quad Z = \sum_{i \in HP} CCUq_{cui} + \sum_{j \in CP} CHUq_{huj} \]

\[ + \sum_{i \in HP} \sum_{j \in CP} \sum_{k \in ST} CF_{ij} z_{ijk} + \sum_{i \in HP} CF_{i,cui} z_{cui} + \sum_{j \in CP} CF_{i,huj} z_{huj} \]

\[ + \sum_{i \in HP} \sum_{j \in CP} \sum_{k \in ST} U_{ijk} \left[ (dt_{ijk})(dt_{ijk+1})(dt_{ijk} + dt_{ijk+1}) / 2 \right]^{1/3} + \ldots \text{etc.} \]
*PROCESS STREAMS
*

*HOT:
TIIN.FX('1') = 480.00;
TIOUT.FX('1') = 340.00;
FCI('1') = 1.50;
CFI('1') = 1.00;

TIIN.FX('2') = 420.00;
TIOUT.FX('2') = 330.00;
FCI('2') = 2.00;
CFI('2') = 1.00;

*COLD:
TJIN.FX('1') = 320.00;
TJOUT.FX('1') = 410.00;
FCJ('1') = 1.00;
CFJ('1') = 1.00;

TJIN.FX('2') = 350.00;
TJOUT.FX('2') = 460.00;
FCJ('2') = 2.00;
CFJ('2') = 1.00;

TMAPP = 10.00;

*UTILITIES
*

CFHU = 1.00;
THUIN = 500.00;
THUOUT = 500.00;
CFCU = 1.00;
TCUIN = 300.00;
TCUOUT = 300.00;

*COSTS
*

*UTILITIES
HUCOST = 80.00;
CUCOST = 20.00;

UNITC = 1000.00;
ACOEFF = 20.00;
HUOCEFF = 20.00;
CUCOEFF = 20.00;
MODEL STATISTICS

<table>
<thead>
<tr>
<th>Blocks of Equations</th>
<th>27</th>
<th>Single Equations</th>
<th>73</th>
</tr>
</thead>
<tbody>
<tr>
<td>Blocks of Variables</td>
<td>16</td>
<td>Single Variables</td>
<td>61</td>
</tr>
<tr>
<td>Non Zero Elements</td>
<td>261</td>
<td>Non Linear N-Z</td>
<td>32</td>
</tr>
<tr>
<td>Derivative Pool</td>
<td>36</td>
<td>Constant Pool</td>
<td>17</td>
</tr>
<tr>
<td>Code Length</td>
<td>1586</td>
<td>Discrete Variables</td>
<td>12</td>
</tr>
</tbody>
</table>

DICOPT Log File

<table>
<thead>
<tr>
<th>Step</th>
<th>Iter</th>
<th>Function</th>
<th>CPU time</th>
<th>Iterations</th>
<th>Evaluation</th>
<th>Solver</th>
</tr>
</thead>
<tbody>
<tr>
<td>NLP 1</td>
<td>4272.27511</td>
<td>0.02</td>
<td>135</td>
<td>0</td>
<td>minos5</td>
<td></td>
</tr>
<tr>
<td>MIP 1</td>
<td>32732.40039</td>
<td>0.01</td>
<td>101</td>
<td>0</td>
<td>xa</td>
<td></td>
</tr>
<tr>
<td>NLP 2</td>
<td>8962.59738&lt;</td>
<td>0.02</td>
<td>57</td>
<td>0</td>
<td>minos5</td>
<td></td>
</tr>
<tr>
<td>MIP 2</td>
<td>33732.40039</td>
<td>0.02</td>
<td>102</td>
<td>0</td>
<td>xa</td>
<td></td>
</tr>
<tr>
<td>NLP 3</td>
<td>9962.59738</td>
<td>0.01</td>
<td>14</td>
<td>0</td>
<td>minos5</td>
<td></td>
</tr>
</tbody>
</table>

Total solver times: NLP = 0.05  MIP = 0.03  Perc. of total: NLP = 60.98  MIP = 39.02
Total Network Cost ($/yr) = 8962.60
SYNHEAT: T-Q CURVE
Table: Problem Data Example 2

<table>
<thead>
<tr>
<th>Stream</th>
<th>$T_{in}$ (°C)</th>
<th>$T_{out}$ (°C)</th>
<th>$F$ (kW C$^{-1}$)</th>
<th>$h$ (kW m$^{-2}$ C$^{-1}$)</th>
<th>Cost ($\text{kW}^{-1} \text{yr}^{-1}$)</th>
</tr>
</thead>
<tbody>
<tr>
<td>H1</td>
<td>159</td>
<td>77</td>
<td>2.285</td>
<td>0.10</td>
<td>-</td>
</tr>
<tr>
<td>H2</td>
<td>267</td>
<td>80</td>
<td>0.204</td>
<td>0.04</td>
<td>-</td>
</tr>
<tr>
<td>H3</td>
<td>343</td>
<td>90</td>
<td>0.538</td>
<td>0.50</td>
<td>-</td>
</tr>
<tr>
<td>C1</td>
<td>26</td>
<td>127</td>
<td>0.933</td>
<td>0.01</td>
<td>-</td>
</tr>
<tr>
<td>C2</td>
<td>118</td>
<td>265</td>
<td>1.961</td>
<td>0.50</td>
<td>-</td>
</tr>
<tr>
<td>S1</td>
<td>300</td>
<td>300</td>
<td>-</td>
<td>0.05</td>
<td>110</td>
</tr>
<tr>
<td>W1</td>
<td>20</td>
<td>60</td>
<td>-</td>
<td>0.20</td>
<td>10</td>
</tr>
</tbody>
</table>

Cost of heat exchangers ($\text{yr}^{-1}) = 7400 + 80 \ [\text{Area(m}^2)]$
### Example 2

- **Exchanger Heat Load (kW)**
- **AMTD Area (m²)**
- **LMTD Area (m²)**

<table>
<thead>
<tr>
<th>Exchanger</th>
<th>Heat Load (kW)</th>
<th>AMTD Area (m²)</th>
<th>LMTD Area (m²)</th>
</tr>
</thead>
<tbody>
<tr>
<td>3-1-2</td>
<td>94.23</td>
<td>95.09</td>
<td>99.74</td>
</tr>
<tr>
<td>3-2-1</td>
<td>41.88</td>
<td>0.96</td>
<td>0.96</td>
</tr>
<tr>
<td>CU-1</td>
<td>187.37</td>
<td>36.03</td>
<td>36.94</td>
</tr>
<tr>
<td>CU-2</td>
<td>38.15</td>
<td>8.57</td>
<td>9.64</td>
</tr>
<tr>
<td>HU-2</td>
<td>246.39</td>
<td>55.41</td>
<td>65.47</td>
</tr>
</tbody>
</table>
Olefin Separation System \((BP)\)

\(\text{(Lee, Foral, Logsdon, Grossmann, 2003)}\)

Goal: Synthesize optimal separation system

**Input**
- **Feed**
  - Ethane C\(_2\)H\(_6\)
  - Propane C\(_3\)H\(_8\)
  - Butane C\(_4\)H\(_10\)
  - Naphtha C\(_8\)–C\(_{12}\)

**RXN System**
- Pyrolysis Furnaces

**Components**
- **Mixture**
  - Hydrogen H\(_2\)
  - Fuel gas CH\(_4\)
  - Acetylene C\(_2\)H\(_2\)
  - Ethylene C\(_2\)H\(_4\)
  - Ethane C\(_2\)H\(_6\)
  - MAPD C\(_3\)H\(_4\)
  - Propylene C\(_3\)H\(_6\)
  - Propane C\(_3\)H\(_8\)
  - C\(_4\) Mixture
  - C\(_5\) Mixture
  - C\(_6+\) Mixture

**Separation**
- **Separation Tasks**

**Output**
- Hydrogen H\(_2\)
- Fuel gas CH\(_4\)
- Ethylene C\(_2\)H\(_4\)
- Propylene C\(_3\)H\(_6\)
- C\(_4\) Mixture
- C\(_5\) Mixture
- C\(_6+\) Mixture

**Recycle**
- Ethane C\(_2\)H\(_6\)
- Propane C\(_3\)H\(_8\)
25 states
53 separation task

Process Superstructure
MINLP Model

- GDP reformulated as a MINLP

- Problem Size
  - No. of 0-1 variables = 5,800
  - No. of variables = 24,500
  - No. of constraints = 52,700

- GAMS/DICOPT
  - NLP solver: CONOPT2/ MIP solver: CPLEX
  - CPU time ~ 3 hrs on Pentium III PC

- Verification: ASPENPLUS model
  - Fixed process configuration is simulated/optimized
MINLP optimal solution

Dephlegmator first process
7 separation units
20 M$/yr cost saving

Total cost: 110.82 M$/yr

1 dephlegmator
1 absorber
4 distillation columns
1 cold box
1 heat exchange
Logic-based Optimization

Motivation

1. Facilitate modeling of discrete/continuous optimization problems through use symbolic expressions
2. Reduce combinatorial search effort
3. Improve handling nonlinearities

Emerging techniques

2. Generalized Disjunctive Programming \quad Raman and Grossmann (1994)
3. Mixed-Logic Linear Programming \quad Hooker and Osorio (1999)
Generalized Disjunctive Programming (GDP)

- Raman and Grossmann (1994)  *(Extension Balas, 1979)*

\[
\min \ Z = \sum_k c_k + f(x) \\
\text{s.t.} \quad r(x) \leq 0 \\
\bullet \quad Y_{jk} \geq 0, k \in K \\
\bullet \quad c_k = \gamma_{jk} \\
\bullet \quad \Omega(Y) = \text{true} \\
\bullet \quad x \in R^n, c_k \in R^1 \\
\bullet \quad Y_{jk} \in \{\text{true, false}\}
\]

- **Objective Function**
- **Common Constraints**
- **Disjunction**
- **Constraints**
- **Fixed Charges**
- **Logic Propositions**
- **Continuous Variables**
- **Boolean Variables**

*Multiple Terms / Disjunctions*
Modeling Example

Process: A → B → C

MINLP

\[
\begin{align*}
\text{min} \quad & Z = 3.5y_1 + y_2 + 1.5y_3 + x_4 \\
& + 7x_6 + 1.2x_5 + 1.8x_1 \\
& -11x_8 \\
\text{s.t} \quad & x_1 - x_2 - x_3 = 0 \\
& x_7 - x_4 - x_5 - x_6 = 0 \\
& x_5 \leq 5 \\
& x_8 \leq 1 \\
& x_8 - 0.9x_7 = 0 \\
& x_4 = \ln(1+x_2) \\
& x_5 = 1.2 \ln(1+x_3) \\
& x_7 - 5y_1 \leq 0 \\
& x_2 - 5y_2 \leq 0 \\
& x_3 - 5y_3 \leq 0 \\
& x_i \in \mathbb{R} \quad i = 1, \ldots, 8 \\
& y_j \in \{0,1\} \quad j = 1,2,3
\end{align*}
\]
Generalized Disjunctive Programming (GDP)

- Raman and Grossmann (1994)

\[
\begin{align*}
\text{min} & \quad Z = \sum_k c_k + f(x) \\
\text{s.t.} & \quad r(x) \leq 0 \\
& \quad g_{jk}(x) \leq 0, k \in K \\
& \quad c_k = \gamma_{jk} \\
& \quad \Omega(Y) = \text{true} \\
& \quad x \in \mathbb{R}^n, c_k \in \mathbb{R}^1 \\
& \quad Y_{jk} \in \{\text{true, false}\} \\
\end{align*}
\]

Objective Function
Common Constraints
Disjunction
Constraints
Fixed Charges
Logic Propositions
Continuous Variables
Boolean Variables

Relaxation?

Carnegie Mellon
Big-M MINLP (BM)

- **MINLP reformulation of GDP**

\[
\begin{align*}
\min Z &= \sum_{k \in K} \sum_{j \in J_k} \gamma_{jk} y_{jk} + f(x) \\
\text{s.t.} \quad r(x) &\leq 0 \\
g_{jk}(x) &\leq M_{jk} (1 - y_{jk}) \quad j \in J_k, k \in K \\
\sum_{j \in J_k} y_{jk} &= 1, \quad k \in K \\
A y &\leq a \\
x &\geq 0, \quad y_{jk} \in \{0, 1\}
\end{align*}
\]

**NLP Relaxation**
\[
0 \leq y_{jk} \leq 1
\]
Nonlinear Convex Hull Relaxation

- **Disjunction** \((x \in F1) \lor (x \in F2)\)

\[ \text{Feasible region of GDP} \]

\[ \text{Convex Hull} \]

- **Assumption:** Convexity of constraints
- **Convex Hull:** Set of points given by all linear combination of points in F1 and F2.
Convex Hull Formulation

- Consider Disjunction $k \in K$

\[
\bigvee_{j \in J_k} \begin{bmatrix}
Y_{jk} \\
g_{jk}(x) \\
c = \gamma_{jk}
\end{bmatrix}
\]

- Theorem: Convex Hull of Disjunction $k$ \textit{(Lee, Grossmann, 2000)}
  - Disaggregated variables $v^j$
    \[
    \{(x, c) \mid x = \sum_{j \in J_k} v^j, \quad c = \sum_{j \in J_k} \gamma_{jk}, \quad 0 \leq v^j \leq \lambda_{jk} U_{jk}, \quad j \in J_k \Rightarrow \text{Convex Constraints}
    \]
    \[
    \sum_{j \in J_k} \lambda_{jk} = 1, \quad 0 < \lambda_{jk} \leq 1,
    \]
    \[
    \lambda_{jk} g_{jk} \left( \frac{v^j}{\lambda_{jk}} \right) \leq 0, \quad j \in J_k
    \]
  - $\lambda_j$ - weights for linear combination
  - Generalization of Stubbs and Mehrotra (1999)
Remarks

1. \[ h(\nu, \lambda) = \lambda \cdot g(\nu / \lambda) \]

If \( g(x) \) is a bounded convex function, \( h(\nu, \lambda) \) is a bounded convex function \( \text{Hiriart-Urruty and Lemaréchal (1993)} \)

2. \( h(\nu, 0) = 0 \) for bounded \( g(x) \)

3. Implementation

\[ \lambda_{jk} g_{jk}(\nu^{jk} / (\lambda_{jk} + \varepsilon)) \leq 0 \]

\( \varepsilon \) tolerance (e.g. 0.0001)

should be smaller than integer tolerance

4. For linear constraints convex hull reduces to result by \textit{Balas (1985)}
Note. For linear disjunctions

\[ \bigvee_{i \in D} \left[ \sum_{j} a_{ij} x_j \leq b_i \right] \]

above reduces to

\[ x_j = \sum_{i} z^i_j \quad j \in N \]

\[ \sum_{j \in N} a_{ij} z^i_j \leq b_i y_i \quad i \in D \]

\[ \sum_{i} y_i = 1 \quad \text{Balas (1985)} \]

\[ y_i \geq 0 \quad i \in D \]
Convex Relaxation Problem (CRP)

CRP:

\[
\begin{align*}
\min \ Z &= \sum_{k \in K} \sum_{j \in J_k} \gamma_{jk} \lambda_{jk} + f(x) \\
\text{s.t.} \quad r(x) &\leq 0 \\
\quad x &= \sum_{j \in J_k} \nu_{jk}, k \in K \\
\quad 0 &\leq \nu_{jk} \leq \lambda_{jk} U_{jk}, j \in J_k, k \in K \\
\quad \sum_{j \in J_k} \lambda_{jk} &= 1, k \in K \\
\quad \lambda_{jk} g_{jk}(\nu_{jk} / \lambda_{jk}) &\leq 0, j \in J_k, k \in K \\
\quad A\lambda &\leq a \\
\quad x, \nu_{jk} &\geq 0, 0 \leq \lambda_{jk} \leq 1, j \in J_k, k \in K
\end{align*}
\]

*Property:* The NLP (CRP) yields a lower bound to optimum of (GDP).
Big-M MINLP (BM)

**Theorem:** The relaxation of (CRP) yields a lower bound that is greater than or equal to the lower bound that is obtained from the relaxation of problem (BM):

\[
\text{RBM: } \min \sum_{k \in K} \sum_{j \in J_k} \gamma_{jk} y_{jk} + f(x)
\]

\[s.t. \quad r(x) \leq 0\]

\[g_{jk}(x) \leq M_{jk} (1 - y_{jk}) \quad j \in J_k, k \in K\]

\[\sum_{j \in J_k} y_{jk} = 1, \quad k \in K\]

\[A y \leq a\]

\[x \geq 0, \quad 0 \leq y_{jk} \leq 1\]
Methods Generalized Disjunctive Programming

GDP

Logic based methods

Branch and bound
(Lee & Grossmann, 2000)

Decomposition
Outer-Approximation
Generalized Benders
(Turkay & Grossmann, 1997)

Reformulation MINLP

Outer-Approximation
Generalized Benders
Extended Cutting Plane

Convex-hull
Cutting plane
(Sawaya & Grossmann, 2004)

Big-M

Key: relaxation
MINLP Reformulation

Specify in CRP $\lambda$ as 0-1 variables

$$\min \ Z = \sum_{k \in K} \sum_{j \in J_k} \gamma_{jk} \lambda_{jk} + f(x)$$

s.t. \quad r(x) \leq 0

$$x = \sum_{j \in J_k} \nu^{jk}, \ k \in K$$

$$0 \leq \nu^{jk} \leq \lambda_{jk} U_{jk}, \ j \in J, \ k \in K$$

$$\sum_{j \in J_k} \lambda_{jk} = 1, \ k \in K$$

$$\lambda_{jk} g_{jk} (\nu^{jk} / \lambda_{jk}) \leq 0, \ j \in J_k, \ k \in K$$

$$A\lambda \leq a$$

$$x, \ nu^{jk} \geq 0, \ \lambda_{jk} = 0,1 \ j \in J_k, \ k \in K$$
Cutting Planes for Linear Generalized Disjunctive Programming

**GDP Model:**

\[
\text{Min } Z = \sum_{k \in K} c_k + h^T x \\
\text{s.t. } Bx \leq b \\
\Omega(Y) = \text{True} \\
x \in \mathbb{R}^n, \ Y_{jk} \in \{\text{True, False}\}, \ c_k \in \mathbb{R} \\
j \in J_k, \ k \in K
\]

Sawaya, Grossmann (2004)

**Objective Function**

**Common Constraints**

**Disjunctive Constraints**

**Logic Constraints**

**Boolean Variables**
Reformulations as MILP

**Big-M**

\[
\begin{align*}
\text{Min } Z &= \sum_{k \in K} \sum_{j \in J_k} y_{jk} \lambda_{jk} + h^T x \\
\text{s.t. } & \quad Bx \leq b \\
& \quad A_{jk} x - a_{jk} \leq M_{jk} (1 - \lambda_{jk}) \quad j \in J_k, k \in K \\
& \quad \sum_{j \in J_k} \lambda_{jk} = 1 \quad k \in K \\
& \quad D\lambda \leq d \\
& \quad x \in \mathbb{R}^n, \lambda_{jk} \in \{0, 1\} \quad j \in J_k, k \in K
\end{align*}
\]

(BM)

**Convex Hull**

\[
\begin{align*}
\text{Min } Z &= \sum_{k \in K} \sum_{j \in J_k} y_{jk} \lambda_{jk} + h^T x \\
\text{s.t. } & \quad Bx \leq b \\
& \quad A_{jk} \nu_{jk} - a_{jk} \lambda_{jk} \leq 0 \quad j \in J_k, k \in K \\
& \quad x = \sum_{j \in J_k} \nu_{jk} \quad k \in K \\
& \quad 0 \leq \nu_{jk} \leq \lambda_{jk} U_{jk} \quad j \in J_k, k \in K \\
& \quad \sum_{j \in J_k} \lambda_{jk} = 1 \quad k \in K \\
& \quad D\lambda \leq d \\
& \quad x \in \mathbb{R}^n, \nu_{jk} \in \mathbb{R}_+, \lambda_{jk} \in \{0, 1\} \quad j \in J_k, k \in K
\end{align*}
\]

(CH)
Motivation for Cutting Plane Method

**Proposition:** The projected relaxation of (CH) onto the space of (BM) is always as tight or tighter than that of (BM)  
(\textit{Grossmann I.E. , S. Lee, 2003})

**Trade-off:** Big-M fewer vars/weaker relaxation vs Convex-Hull tighter relaxation/more vars
Cutting Plane Method

1. Solve relaxed Big-M MILP $x^{R_{BM}}$.

2. Solve separation problem: find point $x^{SEP}$ closest to $x^{R_{BM}}$.
   Feasible region corresponds to relaxed Convex Hull.

3. Cutting plane is generated and added to relaxed big-M MILP.

4. Solve strengthened relaxed Big-M MILP. Go to 2.
Cutting Plane Method: Different Cuts

Proposition: There exists a vector $\xi$ such that
$$\xi^T(z^{SEP} - z^{BM}) \geq 0$$
is a valid linear inequality, where $\xi$ is a subgradient of $\Phi(z)$ at $z^{SEP}$.

Note: $z=(x,\lambda)$

Proposition: (1) Let $\Phi(z) \equiv \|z - z^{BM}\|_2 \equiv (z - z^{BM})^T(z - z^{BM})$. Then,
$$\xi \equiv \nabla \Phi = (z - z^{BM})$$

(2) Let $\Phi(z) \equiv \|z - z^{BM}\|_\infty \equiv \max_i |z_i - z_i^{BM}|$. Then,
$$\xi \equiv (\mu^+ - \mu^-)$$

\[
\begin{align*}
\text{Min } u \\
\text{s.t. } & u \geq z_i - z_i^{BM} \quad i \in I \\
& u \geq -z_i + z_i^{BM} \quad i \in I
\end{align*}
\]

Lagrange Multipliers
$$\mu^+ \quad \mu^-$$

Feasible region of (SEP)

(3) Let $\Phi(z) \equiv \|z - z^{BM}\|_1 \equiv \sum |z_i - z_i^{BM}|$. Then,
$$\xi \equiv (\mu^+ - \mu^-)$$

\[
\begin{align*}
\text{Min } \sum u_i \\
\text{s.t. } & u_i \geq z_i - z_i^{BM} \quad i \in I \\
& u_i \geq -z_i + z_i^{BM} \quad i \in I
\end{align*}
\]

Lagrange Multipliers
$$\mu^+ \quad \mu^-$$

Feasible region of (SEP)

- We need to fit a set of small rectangles with width $w_i$ and length $l_i$ onto a large rectangular strip of fixed width $W$ and unknown length $L$. The objective is to fit all small rectangles onto the strip without overlap and rotation while minimizing length $L$ of the strip.
GDP Model For Strip-packing Problem

\[
\begin{align*}
\text{Min } Z &= L \\
\text{s.t. } & L \geq x_i + l_i & i \in N \\
& Y_{ij}^1 \lor Y_{ij}^2 \lor Y_{ij}^3 \lor Y_{ij}^4 & i,j \in N, \ i < j \\
& x_i + l_i \leq x_j & \forall i \in N \\
& x_j + l_j \leq x_i & \forall i \in N \\
& y_i - h_i \geq y_j & \forall i \in N \\
& y_j - h_j \geq y_i & \forall i \in N \\
& 0 \leq x_i \leq U_i - l_i & i \in N \\
& h_i \leq y_i \leq W & i \in N \\
& x_i, y_i \in R & i \in N \\
& Y_{ij}^1, Y_{ij}^2, Y_{ij}^3, Y_{ij}^4 \in \{\text{True, False}\} & i,j \in N, \ i < j
\end{align*}
\]
## 21-rectangle Strip-packing Problem

### Problem Size

<table>
<thead>
<tr>
<th></th>
<th>Total number of constraints</th>
<th>Total number of variables</th>
<th>Number of discrete variables</th>
</tr>
</thead>
<tbody>
<tr>
<td>Convex Hull</td>
<td>5272</td>
<td>4244</td>
<td>840</td>
</tr>
<tr>
<td>Big-M</td>
<td>1072</td>
<td>884</td>
<td>840</td>
</tr>
</tbody>
</table>

### Solution

![Solution Diagram](image)

Optimal Length: 24
## Numerical Results

(CPLEX v. 8.1, default MIP options turned on)

<table>
<thead>
<tr>
<th></th>
<th>Relaxation</th>
<th>Optimal Solution</th>
<th>Gap (%)</th>
<th>Total Nodes in MIP</th>
<th>Solution Time for Cut Generation (sec)</th>
<th>*Total Solution Time (sec)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Convex Hull</td>
<td>9.1786</td>
<td>---</td>
<td>---</td>
<td>968 652</td>
<td>0</td>
<td>&gt;10 800</td>
</tr>
<tr>
<td>Big-M</td>
<td>9</td>
<td>24</td>
<td>62.5</td>
<td>1 416 137</td>
<td>0</td>
<td>4 093.39</td>
</tr>
<tr>
<td>Big-M + 20 cuts</td>
<td>9.1786</td>
<td>24</td>
<td>61.75</td>
<td>306 029</td>
<td>3.74</td>
<td>917.79</td>
</tr>
<tr>
<td>Big-M + 40 cuts</td>
<td>9.1786</td>
<td>24</td>
<td>61.75</td>
<td>547 828</td>
<td>7.48</td>
<td>1 063.51</td>
</tr>
<tr>
<td>Big-M + 60 cuts</td>
<td>9.1786</td>
<td>24</td>
<td>61.75</td>
<td>28 611</td>
<td>11.22</td>
<td>79.44</td>
</tr>
<tr>
<td>Big-M + 62 cuts</td>
<td>9.1786</td>
<td>24</td>
<td>61.75</td>
<td>32 185</td>
<td>11.59</td>
<td>91.4</td>
</tr>
</tbody>
</table>

* Total solution time includes times for relaxed MIP(s) + LP(s) from separation problem + MIP

Results also for retrofit, scheduling problems
A Branch and Bound Algorithm for GDP

- **Tree Search**
  - NLP subproblem at each node
- **Solve CRP of GDP**
  - lower bound
- **Branching Rule**
  - Set the largest \( \lambda_j \) as 1 or 0
  - Dichotomy rule
- **Logic inference**
  - CNF unit resolution (Raman & Grosmann, 1993)
- **Depth first search**
  - When all the terms are fixed upper bound
- **Repeat Branching until** \( Z_L > Z_U \).
GDP Example

- Find $x \geq 0, (x \in S_1) \lor (x \in S_2) \lor (x \in S_3)$ to minimize $Z = (x_1 - 3)^2 + (x_2 - 2)^2 + c$

- Objective Function = continuous function + fixed charge (discontinuous).

Contour of $f(x)$

Global Optimum $(3.293, 1.707)$
$Z^* = 1.172$

Carnegie Mellon
Example: convex hull

Convex hull = $\text{conv}(\bigcup S_j)$
Example: CRP solution

Convex hull = \( \text{conv}(\mathbf{U}S_j) \)

Convex combination of \( z_j \)

\[ z_j = \psi / \lambda_j \]

Local solution point

\[ x^* \]

Convex hull optimum, \( Z^L = 1.154 \)

\[ x^L = (3.159, 1.797) \]

Infeasible to GDP

Weight

\[ \lambda_1 = 0.016 \]
\[ \lambda_2 = 0.955 \]
\[ \lambda_3 = 0.029 \]
Example: branch and bound

First Node: $S_2$
Optimal solution: $Z^U = 1.172$

Optimal Solution
(3.293, 1.707)
$Z^* = 1.172$
Example: branch and bound

Second Node: $\text{conv}(S_1 \cup S_3)$
Optimal solution: $Z^L = 3.327$

Lower Bound $Z^L = 3.327$

Upper Bound $Z^U = 1.172$
Example: Search Tree

- **Branching Rule:** $\lambda_j$ - the weight of disaggregated variable
  - Fix $Y_j$ as true: fix $\lambda_j$ as 1.
Process Network with Fixed Charges

- Türkay and Grossmann (1997)
  - Superstructure of the process
Optimal solution

- Minimum Cost: $68.01M/year
Proposed BB Method

Proposed BB

\[ Z^L = 62.48 \]
\[ \lambda = [0.31, 0.69, 0.03, 1.0, 1, 0, 1, 0, 1] \]

\[ Z^U = 68.01 = Z^* \]
\[ \lambda = [0, 1, 0.022, 1.0, 1, 0, 1, 0, 1] \]

Feasible Solution

Optimal Solution

\[ Z^U = 71.79 \]
\[ \lambda = [0, 1, 1, 1.0, 1, 0, 1] \]

\[ Z^L = 65.92 \]
\[ \lambda = [0, 1, 0.022, 1.0, 1, 0, 1] \]

Stop

\[ Z^L = 75.01 > Z^U \]

\[ Z^L = 15.08 \]

Standard BB

\[ Y_4 = 0 \]
\[ Y_4 = 1 \]
\[ Y_6 = 0 \]
\[ Y_6 = 1 \]
\[ Y_8 = 0 \]
\[ Y_8 = 1 \]

\[ Y_1 = 0 \]
\[ Y_1 = 1 \]
\[ Y_3 = 0 \]
\[ Y_3 = 1 \]

\[ \text{Fix } \lambda_2 = 1 \]
\[ \text{Fix } \lambda_2 = 0 \]
\[ \text{Fix } \lambda_3 = 1 \]
\[ \text{Fix } \lambda_3 = 0 \]

\[ \text{Fix } \lambda_2 = 0 \]

\[ \text{Fix } \lambda_3 = 1 \]

\[ \text{Fix } \lambda_3 = 0 \]

\[ \text{Stop} \]

\[ \text{CPU time: 2.578 vs. 3.383 of Standard BB (300MHz Pentium II PC)} \]

\[ \text{5 nodes vs. 17 nodes of Standard BB (lower bound = 15.08)} \]
Nonconvex GDP

\[
\min \quad Z = \sum_k c_k + f(x)
\]

\[
s.t. \quad r(x) \leq 0
\]

\[
\begin{bmatrix}
Y_{jk} \\
g_{jk}(x) \leq 0 \\
c_k = \gamma_{jk} \\
\Omega(Y) = true
\end{bmatrix}, k \in K
\]

\[
x \in R^n, c_k \in R^1
\]

\[
Y_{jk} \in \{true, false\}
\]

\[f, g \text{ and } r: \text{ nonconvex}\]
Convex Underestimator GDP (R)

- Introducing convex underestimators (see Sahinidis lecture)

\[
\min Z = \sum_k c_k + \bar{f}(x) \\
\text{s.t.} \quad \bar{r}(x) \leq 0 \\
\bigvee_{j \in J_k} \begin{bmatrix} Y_{jk} \\ \bar{g}_{jk}(x) \leq 0 \end{bmatrix}, k \in K \\
\gamma_{jk} = c_k \\
\Omega(Y) = \text{true} \\
x \in R^n, c_k \in R^1 \\
Y_{jk} \in \{\text{true, false}\} \\
\bar{f}, \bar{r} \text{ and } \bar{g} : \text{convex}
\]

Convex underestimators

- Bilinear: Linear
  McCormick (1976), Al-Khayyal (1992)

- Linear fractional: Convex nonlinear
  Quesada and Grossmann (1995)

- Concave separable: Linear secant

\*Problem (R) yields a valid lower bound to Problem (GDP)
Convex envelopes

Concave function

Secant $g(x)$

\[ g(x) = f(a) + \left( \frac{f(b) - f(a)}{b - a} \right)(x - a) \]
Bilinear

\[ w = xy \]

\[ x^L \leq x \leq x^U \quad y^L \leq y \leq y^U \]

\[ (x - x^L) \geq 0, \quad (y - y^L) \geq 0 \]

\[ (x - x^L)(y - y^L) \geq 0 \]

\[ w = xy \geq x^L y + y^L x - x^L y^L \quad \text{Valid underestimator} \]

\[ \text{Convex envelope} \]

McCormick convex envelopes

\[ w \geq x^L y + y^L x - x^L y^L \]

\[ w \geq x^U y + y^U x - x^U y^U \]

\[ w \leq x^L y + y^U x - x^L y^U \]

\[ w \leq x^U y + y^L x - x^U y^L \]
Basic Ideas

1. Branch and bound enumeration on disjunctions of convex GDP \((R)\)

2. When feasible discrete solution found, switch to spatial branch and bound (NLP subproblem)
Synthesis Multiproduct Batch Plant

(Birewar & Grossmann, 1990)

More than 100 alternatives: each requires nonlinear optimization
Synthesis Multiproduct Batch Plant

Nonconvex GDP Model

\[
\min \text{COST} = \sum_{j=1}^{M} N_j^{EQ} C_j + \sum_j CS_j
\]

s.t. \( V_i^T \geq B_i S_{it} \quad i = 1, \ldots, N_p ; t = 1, \ldots, T \)
\[
pt_{ij} = \sum_{i \in I_j} pty_{ij} \quad i = 1, \ldots, N_p ; j = 1, \ldots, M
\]
\[
n_i B_i \geq Q_i \quad i = 1, \ldots, N_p
\]
\[
\sum_{i=1}^{N_r} n_i T_{Li} \leq H
\]

Objective function

Sizing

Process time

Demand

Horizon time

\[
\bigvee_{j \in J_t} \begin{bmatrix}
Y_j \\
V_j \geq V_{r}^T \\
pty_{ij} = pt_{it}^T \\
pty_{ij'} = 0, j' \neq j
\end{bmatrix} \quad t \in T
\]

Disjunction for Task Assignments

Nonconvex functions
GDP model (continued)

\[
YEX_j = \gamma_j + \alpha_j V_j^{0.8}, \quad V_L^j \leq V_j^L \leq V_C^j
\]

\[
\begin{align*}
&[YC_{i,j}] \\
&N_j^{eq} = 1 \lor \left(2T_{i} \geq pt_\phi \right) \\
&T_{i} \geq pt_\phi
\end{align*}
\]

\[
\begin{align*}
&[YC_{2,j}] \\
&N_j^{eq} = 2 \lor \left(3T_{i} \geq pt_\phi \right) \\
&T_{i} \geq pt_\phi
\end{align*}
\]

\[
\begin{align*}
&[YC_{3,j}] \\
&N_j^{eq} = 3 \lor \left(4T_{i} \geq pt_\phi \right) \\
&T_{i} \geq pt_\phi
\end{align*}
\]

\[
\begin{align*}
&[YC_{4,j}] \\
&N_j^{eq} = 4 \lor \left(T_{i} \geq 0 \right)
\end{align*}
\]

\[
\begin{align*}
&[\neg YEX_j] \\
&C_j = 0 \\
&V_j = 0 \\
&N_j^{eq} = 0 \\
&pt_\phi = 0 \\
&T_{i} \geq 0
\end{align*}
\]

\[
\begin{align*}
&[\neg YS_j] \\
&B_j = B_j \\
&VST_j = 0 \\
&CS_j = 0
\end{align*}
\]

\[
CS_j = 5000 + 80VST_j^{eq}
\]

\[
YEX_1 \iff Y_{11}, YEX_2 \iff Y_{12} \lor Y_{22}, YEX_3 \iff Y_{33}
\]

\[
YEX_4 \iff Y_{14} \lor Y_{24} \lor Y_{34}, YEX_5 \iff Y_{45}
\]

\[
W_{14} \lor W_{24} \lor W_{34}
\]

\[
W_{14} \iff (Y_{14} \land \neg Y_{24} \land \neg Y_{34}) \lor \neg (Y_{14} \land Y_{24} \land Y_{34})
\]

\[
W_{24} \iff (Y_{14} \land Y_{24} \land \neg Y_{34}) \lor \neg (Y_{14} \land \neg Y_{24} \land Y_{34})
\]

\[
W_{34} \iff Y_{14} \land Y_{24} \land Y_{34}
\]

\[
0 \leq C_j, V_j, V_{i}^{eq}, n_j, B_j, T_{i}, pt_\phi, N_j^{eq}, pty_\phi; YEX_\phi, Y_j, YC_j, W_j \in \{true, false\}
\]
Proposed Algorithm for Nonconvex GDP

Step 0
Nonconvex MINLP

\[ Z^U \]

Step 1
Bound Contraction

New Bound

Step 2
BB with Y’s Update \( Z^L \)

When solution is Integral

Add Integer Cut

Stop when \( Z^L \geq Z^U \)

Step 3
Spatial BB Update \( Z^U \)

Fixed Y’s

OA (Viswanathan and Grossmann, 1990)

(Zamora and Grossmann, 1999)

(Lee and Grossmann, 2000)

(Quesada and Grossmann, 1995)
Step 2: Disjunctive Branch and Bound

- **Procedure** (Lee and Grossmann, 1999)
  - Relaxed NLP problem is solved.
  - Check Infeasibility of each term.

- **Branching Rule**
  - Select a term with Minimum Infeasibility first.
  - Logic Inference – fixes additional Boolean Variables.

- **Lower Bound is updated**

- When all the Boolean variables are fixed and the relaxation gap is nonzero:
  - switch to Spatial Branch and Bound.
Step 3: Spatial Branch and Bound

- **Fixed Boolean Variables: Convex NLP** Quesada and Grossmann (1995)
  - Check the relaxation gap of each underestimator function.
  - Select the constraint with maximum gap.
  - Branch on a variable which causes the maximum gap.
  - Bisection Rule: Feasible region is divided into 2 subregions.
  - If the gap is small enough, an upper bound is updated.

- **Return to Step 2 with:**
  - Updated Global Upper Bound
  - Feasibility Cut, $Z < \text{GUB}$
  - Integer Cut for the chosen set of Y’s

![Nonconvex Objective Function $f$](image)
Upper Bound Solution

Cost = $277,928 (by GAMS/DICOPT++)

- Use 4 Stages (6 units) without Storage Tank

Mixing

Reaction

Crystallization

Drying

\[ j = 1 \]

\[ V_1 = 4,842 \text{ L} \]

\[ j = 2 \]

\[ V_2 = 2,881 \text{ L} \]

\[ j = 4 \]

\[ V_4 = 2,469 \text{ L} \]

\[ j = 5 \]

\[ V_5 = 8,071 \text{ L} \]

\begin{align*}
A & \rightarrow 243 \text{ batches, 4.5hrs} \\
B & \rightarrow 260 \text{ batches, 6hrs} \\
C & \rightarrow 372 \text{ batches, 9hrs} \\
\end{align*}

6000 hrs

\begin{align*}
A & \rightarrow 1093 \text{ hrs} \\
B & \rightarrow 1562 \text{ hrs} \\
C & \rightarrow 3345 \text{ hrs} \\
\end{align*}
Optimal Solution: Multiproduct Batch Plant

- Global optimal cost = $264,887 (5% improvement)
- 3 Stages + 1 storage tank (5 units)

Mixing Reaction Storage Crystallization Drying

Tank

A
B
C

$j = 2$

$V_2 = 4,309$ L  $VST_2 = 4,800$ L  $V_3 = 3,600$ L  $V_5 = 11,753$ L

A 250 batches, 5hrs
B 293 batches, 3hrs
C 418 batches, 5.5hrs

6000 hrs

Storage

1503 hrs

167 batches, 9hrs

184 batches, 12 hrs

255 batches, 9hrs
Logic-based Outer Approximation

Main point: avoids solving MINLP in full space

Turkay, Grossmann (1997)

**NLP Subproblem:**
(reduced)

\[
\begin{align*}
\min Z &= \sum_{k \in SD} c_k + f(x) \\
\text{s.t.} \quad g(x) &\leq 0 \\
h_{d_k}(x) &\leq 0 \quad \text{for } Y_{ik} = \text{true} \quad i \in D_k, k \in SD \\
c_k &= \gamma_{ik} \\
B^T x &= 0 \quad \text{for } Y_{ik} = \text{false} \quad i \in D_k, i \neq \hat{i}, k \in SD \\
x &\in R^n, \quad c_i \in R^m,
\end{align*}
\]

**Master Problem:**

\[
\begin{align*}
\text{Min} \quad Z &= \sum_i c_i + \alpha \\
\text{s.t.} \quad \alpha &\geq f(x^i) + \nabla f(x^i)^T (x - x^i) \\
g(x^i) + \nabla g(x^i)^T (x - x^i) &\leq 0 \quad \text{for } i = 1, \ldots, L
\end{align*}
\]

\[
\left[ \begin{array}{c}
Y_{ik} \\
h_{d_k}(x^i) + \nabla h_{d_k}(x^i)^T (x - x^i) \leq 0 \\
c_k = \gamma_{ik} \\
\end{array} \right] \quad k \in SD
\]

\[
\bigvee_{i \in D_k} Y_{ik} \quad \Omega(Y) = \text{True}
\]

\[
\alpha \in R, \quad x \in R^n, \quad c \in R^m, \quad Y \in \{\text{true, false}\}^m
\]

Master problem solved with disjunctive branch and bound or with MILP reformulation

Redundant constraints are eliminated with false values

Proceed as OA. Requires initialization several NLPs to cover all disjunctions
LogMIP

Part of GAMS Modeling System
- Disjunctions specified with IF Then ELSE statements

\[
\text{DISJUNCTION } D1(I,K,J);
\]
\[
D1(I,K,J)
\]
    \[\text{with } (L(I,K,J)) \text{ IS}\]
\[
\text{IF } Y(I,K,J) \text{ THEN}
\]
\[
\text{NOCLASH1}(I,K,J);
\]
\[
\text{ELSE}
\]
\[
\text{NOCLASH2}(I,K,J);
\]
\[
\text{ENDIF};
\]

- Logic can be specified in symbolic form (⇒, OR, AND, NOT )
  or special operators (ATMOST, ATLEAST, EXACTLY)
- Linear case: MILP reformulation big-M, convex hull
- Nonlinear: Logic-based OA

http://www.ceride.gov.ar/logmip/
SET I /1*3/;
SET J /1*2/;
BINARY VARIABLES Y(I);
POSITIVE VARIABLES X(J), C;
VARIABLE Z;
EQUATIONS EQUAT1, EQUAT2, EQUAT3,
EQUAT4, EQUAT5, EQUAT6,
    DUMMY, OBJECTIVE;

EQUAT1.. X('2')- X('1') + 2 =L= 0;
EQUAT2.. C =E= 5;
EQUAT3.. 2 - X('2') =L= 0;
EQUAT4.. C =E= 7;
EQUAT5.. X('1')-X('2') =L= 1;
EQUAT6.. X('1') =E= 0;
DUMMY.. SUM(I, Y(I)) =G= 0;

OBJECTIVE.. Z =E= C + 2*X('1') + X('2');
X.UP(J)=20;
C.UP=7;

$ONTEXT BEGIN LOGMIP

DISJUNCTION D1, D2;
D1 IS
IF Y('1') THEN
    EQUAT1;
    EQUAT2;
ELSIF Y('2') THEN
    EQUAT3;
    EQUAT4;
ENDIF;
D2 IS
IF Y('3') THEN
    EQUAT5;
ELSE
    EQUAT6;
ENDIF;
Y('1') and not Y('2') -> not Y('3');
Y('2') -> not Y('3');
Y('3') -> not Y('2');

$OFFTEXT END LOGMIP

OPTION MIP=LOGMIPM;
MODEL PEQUE /ALL/;
SOLVE PEQUE USING MIP MINIMIZING Z;
Structural Optimization of Vinyl Chloride Plant

**Major options:**
- Direct Chlorination vs. Oxychlorination
- Air vs. Oxygen
- Pressure Pyrolysis
- Separation sequence

**Optimization with discontinuous cost models:**
- Multiple size regions
- Pressure, temperature factors
Process Flowsheet Synthesis

Superstructure Vinyl Chloride Monomer

(Turkay & Grossmann, 1997)

Direct chlorination: \[ \text{C}_2\text{H}_4 + \text{Cl}_2 \rightarrow \text{C}_2\text{H}_4\text{Cl}_2 \]

Oxychlorination: \[ \text{C}_2\text{H}_4 + 2\text{HCl} + \frac{1}{2} \text{O}_2 \rightarrow \text{C}_2\text{H}_4\text{Cl}_2 + \text{H}_2\text{O} \]

Pyrolysis: \[ \text{H}_4\text{Cl}_2 \rightarrow \text{C}_2\text{H}_3\text{Cl} + \text{HCl} \]
Optimal Solution  (CPU-time: 3.8min)

<table>
<thead>
<tr>
<th>Item</th>
<th>Flowsheet 1</th>
<th>Flowsheet 2</th>
<th>Master Pr. 1</th>
<th>Flowsheet 3</th>
<th>Master Pr. 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Binary var.</td>
<td>162</td>
<td>168</td>
<td>259</td>
<td>168</td>
<td>259</td>
</tr>
<tr>
<td>Continuous var.</td>
<td>879</td>
<td>883</td>
<td>1715</td>
<td>883</td>
<td>1822</td>
</tr>
<tr>
<td>Constraints</td>
<td>699</td>
<td>703</td>
<td>1750</td>
<td>703</td>
<td>1858</td>
</tr>
<tr>
<td>Profit ($M/yr)</td>
<td>27.678</td>
<td>75.283</td>
<td>82.763</td>
<td>71.809</td>
<td>65.262</td>
</tr>
<tr>
<td>Major Iterations</td>
<td>3</td>
<td>3</td>
<td>N/A</td>
<td>3</td>
<td>N/A</td>
</tr>
<tr>
<td>CPU time (sec)</td>
<td>88.00</td>
<td>64.99</td>
<td>8.24</td>
<td>53.76</td>
<td>13.21</td>
</tr>
</tbody>
</table>
Optimal Feedtray Location

Sargent & Gaminibandara (1976)

NLP Formulation

Min cost

\[ \sum_{i \in \text{LOC}} f_i = F \]

NLP VMP: Variable-Metric Projection
Optimal Feedtray Location (Cont)

Viswanathan & Grossmann (1990)

MINLP Formulation

\[
\text{Min cost}
\]

\[
st \ MESH eqtns
\]

\[
\sum_{i \in \text{LOC}} z_i = 1
\]

\[
\sum_{i \in \text{LOC}} f_i = F
\]

\[
f_i - F z_i \leq 0 \ i \in \text{LOC}
\]

\[
z_i = 0,1 \ i \in \text{LOC}
\]

**MINLP DICOPT: AP-Outer Approximation-ER**

Remark: MINLP solves as relaxed NLP!

*Feed tray composition tends to match composition of feed*
Optimization of Number of Trays

Viswanathan & Grossmann (1993)

Discrete variables: Number of trays, feed tray location.
Continuous variables: reflux ratio, heat loads, exchanger areas, column diameter.

Zero flows- Discontinuities appear, convergence difficulties.
Redundant equations are solved- Increases CPU time.

MINLP  \Rightarrow  Number of trays

\begin{align}
zr_i &= 0.1 \\
zr_m &= 1 \\
zb_i &= 0.1 \\
zb_n &= 1
\end{align}
Disjunctive Programming Model

Yeomans & Grossmann (2000)

Permanent trays:
Feed, reboiler, condenser

Conditional trays:
Intermediate trays might be selected or not.

Trays not allowed to “disappear” from column:

- VLE mass transfer if selected.
- No VLE, trivial mass/energy balance if not selected

Disjunction

VLE

-OR-

NOT VLE (tray bypass)
Single Column GDP Model

• Permanent and conditional trays:
  - MESH equations for condenser, reboiler and feed trays
  - Mass & energy balances for rectification and stripping trays.

• Conditional trays only:
  - Apply VLE constraints ($Y_n=True$) or not ($Y_n=False$)
  - Use disjunctions as modeling tool.

Equilibrium Stage  \[\downarrow\]  Vapor Flow

Non-equilibrium Stage  \[\uparrow\]  Liquid Flow
Logic-based OA Algorithm

Turkay, Grossmann (1996)

OA Algorithm

Selected Equations

Subproblem (NLP)

Continuous variables for Initialization

Master Problem (MILP)

Discrete variables

Initial Subproblems (NLP)

Converge?

Solution

NO

YES

Implemented in GAMS CPLEX/CONOPT
### Example GDP

**GDP Formulation**
- Mixture: Methanol/Ethanol/water
- Feed Flow = 10 mol/sec
- Feed composition = 0.2/0.2/0.6
- P = 1.01 bar
- Product Specification: products composition reversible model
- Upper bound No. Trays: 60

---

### Methanol/ethanol/water - GDP: fixed tray location

**Preprocessing Phase: NLP tray-by-tray Models**

<table>
<thead>
<tr>
<th>Continuous Variables</th>
<th>1597</th>
</tr>
</thead>
<tbody>
<tr>
<td>Constraints</td>
<td>1544</td>
</tr>
<tr>
<td>Total CPU time (s)</td>
<td>1.12</td>
</tr>
</tbody>
</table>

**Model Description**

<table>
<thead>
<tr>
<th>Continuous Variables</th>
<th>2933</th>
</tr>
</thead>
<tbody>
<tr>
<td>Binary Variables</td>
<td>60</td>
</tr>
<tr>
<td>Constraints</td>
<td>2862</td>
</tr>
<tr>
<td>Nonlinear nonzero elements</td>
<td>5656</td>
</tr>
<tr>
<td>Number of iterations</td>
<td>10</td>
</tr>
<tr>
<td>NLP CPU time (s)</td>
<td>9.14</td>
</tr>
<tr>
<td>MILP CPU time (s)</td>
<td>16.97</td>
</tr>
<tr>
<td>Total CPU time (s)</td>
<td>401</td>
</tr>
</tbody>
</table>

**Optimal Solution**

| Total number of trays | 41 |
| Feed tray             | 20 |
| Column diameter (m)   | 0.51 |
| Condenser duty (KJ/s) | 387.4 |
| Reboiler duty (KJ/s)  | 386.5 |
| Objective value ($/yr)| 117,600 |

- **GAMS PIII, 667 MHz. with 256 MB of RAM.**
- **CONOPT2 NLP subproblems/ CPLEX MILP subproblems.**
Variables: Continuous, integer, boolean

Constraints:

- **Algebraic**: \( h(x) \leq 0 \)
- **Disjunctions**: \( [ A_1 x \leq b_1 ] \lor [ A_2 x \leq b_2 ] \)
- **Conditional**: If \( g(x) \leq 0 \) then \( r(x) \leq 0 \)

Unusual Operators:

- **All different**: \( \text{all different}(x_1, x_2, \ldots x_n) \)
Constraint Programming

1. Declarative language with high level operators (*OPL-ILOG*)

2. Tree search: implicit enumeration
   - Depth first search
   - Lower bound: partial solution
   - Upper bound: feasible solution

3. Constraint propagation (in contrast to generate and test):
   - **Domain Reduction**
     - Reduction of bounds/discrete values:
       - a) Tighten bounds linear/monotonic functions
       - b) "edge-finding" for jobshop scheduling

\[ \text{Job } i \text{ before Job } j \] \text{ OR } \{\text{Job } j \text{ before Job } i\}
Problem
Find values for $x$, $y$ and $z$ in \{1,2,3\} that satisfy

\[ x - y = 1 \]
\[ x + y + z = 6 \]
Constraint Programming
Applying domain reduction

\[ x - y = 1 \quad \Rightarrow x \neq 1 \quad \quad x + y + z = 6 \]

\[ x \in \{1, 2, 3\} \quad \quad y \in \{1, 2, 3\} \quad \quad z \in \{1, 2, 3\} \]
Constraint Programming

Applying domain reduction

\[ x - y = 1 \implies x \neq 1 \]
\[ x=2 \implies y = 1 \]
\[ x=3 \implies y = 2 \]

\[ x + y + z = 6 \]

\[ x \in \{1,2,3\} \]
\[ y \in \{1,2,3\} \]
\[ z \in \{1,2,3\} \]
Constraint Programming
Applying domain reduction

\[ x - y = 1 \Rightarrow x \neq 1 \]
\[ x = 2 \Rightarrow y = 1 \]
\[ x = 3 \Rightarrow y = 2 \]

\[ x + y + z = 6 \]

\[ x \in \{1, 2, 3\} \]
\[ y \in \{1, 2, 3\} \]
\[ z \in \{1, 2, 3\} \]
\[ x - y = 1 \quad \Rightarrow x \neq 1 \]
\[ x = 2 \quad \Rightarrow \quad y = 1 \]
\[ x = 3 \quad \Rightarrow \quad y = 2 \]

\[ x + y + z = 6 \]

Constraint Programming
Applying domain reduction
Constraint Programming
Applying domain reduction

\[ x - y = 1 \quad \Rightarrow x \neq 1 \]
\[ x = 2 \quad \Rightarrow \quad y = 1 \]
\[ x = 3 \quad \Rightarrow \quad y = 2 \]

\[ x + y + z = 6 \]

\[ x \in \{1, 2, 3\} \]
\[ y \in \{1, 2, 3\} \]
\[ z \in \{1, 2, 3\} \]
Constraint Programming
Applying domain reduction

\[ x - y = 1 \implies x \neq 1 \]
\[ x=2 \implies y = 1 \]
\[ x=3 \implies y = 2 \]

\[ x + y + z = 6 \implies (x = 2) \land (y = 1) \implies z = 3 \]
\[ (x = 3) \land (y = 2) \implies z = 1 \]
Constraint Programming

\[ x - y = 1 \implies x \neq 1 \]
\[ x=2 \implies y = 1 \]
\[ x=3 \implies y = 2 \]

\[ x + y + z = 6 \implies (x = 2) \land (y = 1) \implies z = 3 \]
\[ (x = 3) \land (y = 2) \implies z = 1 \]

Solution 1: \(x = 2, y = 1, z = 3\)
Solution 2: \(x = 3, y = 2, z = 1\)
OPL  *(ILOG OPL Studio)*  Van Hentenryck (1999)

*Single language for linear and integer programming, constraint programming and scheduling*

**CLP.mod**

// Number of Machines (Packing + Manufacturing)
int nbMachines = ...;
range Machines 1..nbMachines;

// Number of Jobs
int nbJobs = ...;
range Jobs 1..nbJobs;
int+ duration[Jobs,Machines] = ...;
int+ cost[Jobs,Machines]=...;
int+ release[Jobs] = ...;
int+ due[Jobs] = ...;

scheduleOrigin = (min(j in Jobs) release[j]);
scheduleHorizon = (max(j in Jobs) due[j]);
Activity task[j in Jobs];
UnaryResource tool[Machines];
AlternativeResources s(tool);
var Machines assign[Jobs];
minimize
    sum(j in Jobs) cost[j, assign[j]]
subject to {
    forall(j in Jobs)
        task[j].start >= release[j] & task[j].start <= due[j] -
            duration[j, assign[j]] ;
    forall(j in Jobs)
        task[j] requires s;
    forall(j in Jobs)
        forall(m in Machines)
            activityHasSelectedResource(task[j], s, tool(m)) <=>
                assign[j] = m;
    forall(j in Jobs)
        task[j].duration = duration[j, assign[j]];
};

CLP.dat
nbMachines = 2;
nbJobs = 3;
duration = [
    [10, 14],
    [6, 8],
    [11, 16]
];
cost = [
    [10, 6],
    [8, 5],
    [12, 7]
];
release = [2, 3, 4];

due = [16, 13, 21];
Constraint vs. Mixed-integer Programming

Mixed-integer Programming

➢ *Intelligent search strategy for general purpose models*
  
  Computationally effective for optimization problems with many feasible solutions
  
  Not effective for feasibility problems and *sequencing* problems

Constraint Programming

➢ *Fast algorithms for special problems*
  
  Computationally effective for *highly constrained, feasibility and sequencing* problems
  
  Not effective for optimization problems with *complex structure and many feasible solutions*

Basic Idea

Decompose problem into two parts:

1. Use MILP for high-level optimization decisions *(assignment)*
2. Use CP for low-level decisions *(sequencing)*
MILP/CP Hybrid Models

*Jain, Grossmann (2001)*

**MILP:**

\[(M1) : \min \ c^T x\]

\[\text{s.t. } Ax + By + Cv \leq a\]

\[A'x + B'y + C'v \leq a'\]

\[x \in \{0,1\}^n, \ y \in \{0,1\}^m, \ v \in R^p\]

**Complicating rows**

**Complicating variables**

(not in objective function)

**CP:**

\[(M2) : \min \ f(\bar{x})\]

\[\text{s.t. } G(\bar{x}, \bar{y}, \bar{v}) \leq 0\]

\[\bar{x}, \bar{y}, \bar{v} \in D\]

**Hybrid:**

\[(M3) : \min \ c^T x\]

\[\text{s.t. } Ax + By + Cv \leq a\]

\[x \Leftrightarrow \bar{x}\]

\[G(\bar{x}, \bar{y}, \bar{v}) \leq 0\]

\[x \in \{0,1\}^n, \ y \in \{0,1\}^m, \ v \in R^p\]

\[\bar{x}, \bar{y}, \bar{v} \in D\]

\{MILP (optimality)\}

\{CP (feasibility)\}
**Decomposition Hybrid Model**

**MILP**

Solve iteration K relaxed MILP problem ($RM^K$) to optimality

\[
\begin{align*}
\min & \quad c^T x \\
\text{s.t.} & \quad Ax + By + Cv \leq a \\
& \quad Q^k x \leq q^k \quad \forall k \in \{1, \ldots, K-1\} \\
& \quad x \in \{0,1\}^n, \quad y \in \{0,1\}^m, \quad v \in \mathbb{R}^p 
\end{align*}
\]

Partial optimal solution $\bar{x}^K$

Fix equivalent $\bar{x}^K$ to solve feasibility CP subproblem

Find $\bar{y}, \bar{v}$

\[
\begin{align*}
\text{s.t.} & \quad G(\bar{x}^K, \bar{y}, \bar{v}) \leq 0 \\
& \quad \bar{y}, \bar{v} \in D 
\end{align*}
\]

**CP**

Generate cuts $Q^k x \leq q^k$

Set $K = K + 1$

No Solution $\implies$ Infeasible

**Infeasible**

\[
\sum_{i \in T^K} x_i - \sum_{i \in F^K} x_i \leq B^K - 1
\]

$T^K = \{i \mid x_i^k = 1\}$, $F^K = \{i \mid x_i^k = 0\}$

No-good Cuts (weak)

Balas & Jeroslow (1972)

Feasible $\implies$ Optimal Solution

Optimal Solution $\implies$ Feasible
Scheduling of parallel units (Single Stage)

(Jain and Grossmann, 2001)

Given: $n$ jobs/orders (release dates, processing times, due dates)

$m$ units (cost different for each unit/machine)

Find schedule that minimizes cost and meets all due dates

Diagram:
- Jobs (Job 1, Job 2, ..., Job $n$) with processing times.
- Units (Unit 1, Unit m) with processing time.
- Time axis with release dates and due dates.
MILP Optimization Model

\[ x_{im} = \begin{cases} 
1 & \text{if task } i \text{ to unit } m \\
0 & \text{otherwise} 
\end{cases} \quad t_{si} = \text{start time task } i \]

\[
\min \sum_{i \in I} \sum_{m \in M} C_{im} x_{im} \quad \text{Cost processing}
\]

\[
s.t. \quad t_{si} \geq r_i \quad \text{Earliest start}
\]

\[
t_{si} \leq d_i - \sum_{m \in M} p_{im} x_{im} \quad \forall i \in I \quad \text{Latest start}
\]

\[
\sum_{m \in M} x_{im} = 1 \quad \forall i \in I \quad \text{Assign to only one unit}
\]

\[
\sum_{i \in I} x_{im} p_{im} \leq \max_i \{d_i\} - \min_i \{r_i\} \quad \forall m \in M
\]

Assignment constraints
Sequencing tasks in each unit

Let \( y_{ii'} = 1 \) if task \( i \) before task \( i' \) on given unit

If \( x_{im} \) AND \( x_{i'm} \) true then \( y_{ii'} \) OR \( y_{i'i} \) are true

\[
y_{ii'} + y_{i'i} \geq x_{im} + x_{i'm} - 1 \quad \forall i, i' \in I, i > i', m \in M
\]

If \( y_{ii'} = 1 \) then \( ts_{i'} \geq ts_i + p_{im} \)

\[

ts_{i'} \geq ts_i + \sum_{m \in M} p_{im} x_{im} - M (1 - y_{ii'}) \quad \forall i, i' \in I, i \neq i'
\]

Big-M Constraint

\[
x_{im} = \{0,1\}, \quad y_{ii'} = \{0,1\}, \quad ts_i \geq 0
\]
Assignment orders to units: Mixed-integer linear programming

\[
x_{im} = \begin{cases} 
1 & \text{if job } i \text{ to unit } m \\
0 & \text{otherwise}
\end{cases}
\]

\[
\min \sum_{i \in I} \sum_{m \in M} C_{im} x_{im}
\]

s.t. \(ts_i \geq r_i\)

\[
\sum_{i \in I} \sum_{m \in M} C_{im} x_{im} \leq \sum_{m \in M} p_{im} x_{im} \quad \forall i \in I
\]

\[
\sum_{m \in M} x_{im} = 1 \quad \forall i \in I
\]

\[
\sum_{i \in I} x_{im} p_{im} \leq \max_i \{d_i\} - \min_i \{r_i\} \quad \forall m \in M
\]
Sequencing of orders in each unit

Constraint Programming

\[ \text{if } (x_{im} = 1) \text{ then } (z_i = m) \forall i \in I, m \in M \]

\[ i.\text{start} \geq r_i \ \forall i \in I \]

\[ i.\text{start} \leq d_i - p_{zi} \ \forall i \in I \]

\[ i.\text{duration} = p_{zi} \]

\[ i \text{ requires } t_{zi} \ \forall i \in I \]

\[ x_{im} \in \{0, 1\}, ts_i \geq 0 \ \forall i \in I, m \in M \]

\[ z_i \in M \ \forall i \in I; i.\text{start} \in D, i.\text{duration} \in D \ \forall i \in I \]

Global constraint (implicit)
Decomposition Strategy

MILP

Job 1
Job 2
Job 3
Job 4
Job 5
Job 6

Machine 1
Machine 2
Machine 3

Assignment

CP

Sequencing
Carnegie Mellon

Decomposition Strategy

Separate problem into assignment and sequencing problems

- Assign jobs
  - Find the minimum assignment (production) cost MILP
  - Fix assignments
    - For each unit find a feasible sequence for the chosen assignment CP
  - Analyze solution

- Sequence jobs
- Feasible?
  - Yes
  - Stop
  - No
    - Add cuts

- Stop
Cuts for infeasible assignments:

\[ \sum_{i \in I_m^k} x_{im} \leq B^m_k - 1 \quad \forall m \in M \]

\[ I_m^k = \left\{ i \mid x_{im}^k = 1 \right\}, \quad B^m_k = |I_m^k| \]

- No sequence can be found in machine 2 in the assignment of is infeasible and can be cut out
- The cuts also exclude large number of supersets
## Computational Results

### CPU times with integer data  
(Jain and Grossmann, 2001)

**CPLEX 6.5/Sun Ultra 60**

<table>
<thead>
<tr>
<th>Problem</th>
<th>Set 1</th>
<th>Set 2</th>
<th>Set 1</th>
<th>Set 2</th>
<th>Set 1</th>
<th>Set 2</th>
<th>Set 1</th>
<th>Set 2</th>
<th>Set 1</th>
<th>Set 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>MILP</td>
<td>0.04</td>
<td>0.04</td>
<td>0.31</td>
<td>0.27</td>
<td>926.3</td>
<td>199.9</td>
<td>1784.7</td>
<td>73.3</td>
<td>18142.7</td>
<td>102672.3</td>
</tr>
<tr>
<td>CLP</td>
<td>0</td>
<td>0.02</td>
<td>0.04</td>
<td>0.14</td>
<td>3.84</td>
<td>0.38</td>
<td>553.5</td>
<td>9.28</td>
<td>68853.5</td>
<td>2673.9</td>
</tr>
<tr>
<td>Hybrid</td>
<td>0.02</td>
<td>0.01</td>
<td>0.52</td>
<td>0.02</td>
<td>4.18</td>
<td>0.02</td>
<td>2.25</td>
<td>0.04</td>
<td>14.13</td>
<td>0.41</td>
</tr>
</tbody>
</table>

### CPU times with arbitrary rational numbers  
(Harjunkoski et al., 2002)

**CPLEX 6.5/Sun Ultra 60**

<table>
<thead>
<tr>
<th>Problem</th>
<th>Set 1</th>
<th>Set 2</th>
<th>Set 1</th>
<th>Set 2</th>
<th>Set 1</th>
<th>Set 2</th>
<th>Set 1</th>
<th>Set 2</th>
<th>Set 1</th>
<th>Set 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hybrid</td>
<td>0.00</td>
<td>0.01</td>
<td>0.51</td>
<td>0.02</td>
<td>5.36</td>
<td>0.03</td>
<td>0.64</td>
<td>0.92</td>
<td>36.63</td>
<td>4.79</td>
</tr>
</tbody>
</table>
Conclusions Discrete/Continuous Optimization

1. MINLP Optimization
   Significant progress has been made
   Modeling, efficiency and robustness still issues

2. Generalized Disjunctive Programming
   Facilitates modeling, improves solution complex MINLPs
   Cutting planes promising for *trade-off of size vs. tightness*

3. Hybrid MILP/Constrained Programming
   Synergistic effect *greatly reduces exponential times*
   in batch scheduling problems