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

Carnegie Mellon

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

$$\begin{aligned} &\min f(x, y) && \textit{Cost} \\ \text{s.t.}\quad & h(x, y) = 0 && \textit{Process equations} \\ & g(x, y) \le 0 && \textit{Specifications} \\ & x \in X && \textit{Continuous variables} \\ & y \in \{0,1\} && \textit{Discrete variables} \end{aligned}$$

Continuous optimization

Linear programming: ***LP***

Nonlinear programming: ***NLP***

Discrete optimization

Mixed-integer linear programming: ***MILP***

Mixed-integer nonlinear programming: ***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

Input file GAMS

```
$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 at 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

$$\min Z = a^T y + b^T x \qquad \textit{Objective function}$$

$$st \qquad Ay + Bx \le d \qquad \textit{Constraints}$$

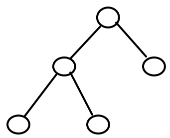
$$y \in \{0,1\}^m,\ x \ge 0$$

Theory for Convexification

Lovacz & Schrijver (1989), Sherali & Adams (1990),

Balas, Ceria, Cornuejols (1993)

Branch and Bound ***Beale (1958), Balas (1962), Dakin (1965)***



LP (simplex) based

Cutting planes ***Gomory (1959), Balas et al (1993)***

Branch and cut ***Johnson, Nemhauser & Savelsbergh (2000)***

"Good" formulation crucial! => *Small LP relaxation gap*

Drawback: exponential complexity

Modeling with MILP
Note: linear constraints

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

$$n = \sum_{k=1}^{N} k y_k, \quad \sum_{k=1}^{N} y_k = 1$$

also

$$n = \sum_{k=1}^{M} 2^k y_k$$

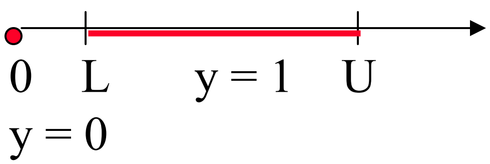
Fewer 0-1 variables

Weaker relaxation

Discontinuous Functions/Domains

a) Domain

$$x = \begin{cases} 0 & \text{IF } y = 0 \\ L \le x \le U & \text{IF } y = 1 \end{cases}$$



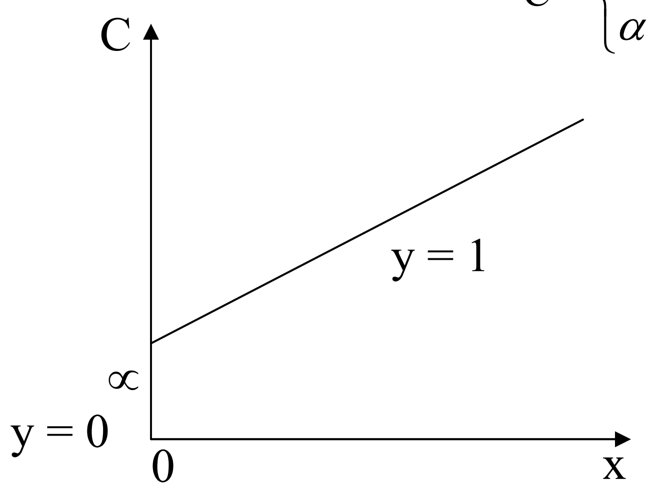
$\Rightarrow$

MIXED-INTEGER MODEL

$$Ly \le x \le Uy$$
$$y = 0, 1$$

b) Function

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



$\Rightarrow$

MIXED-INTEGER MODEL

$$C = \alpha y + \beta x$$
$$0 \le x \le U y$$
$$y = 0, 1$$

Simple Minded Approaches

Exhaustive Enumeration

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

IF $m = 5$	32	COMBINATIONS
IF $m = 100$	10^{30}	COMBINATIONS
IF $m = 10{,}000$	10^{3000}	COMBINATIONS

Relaxation and Rounding

SOLVE MILP WITH $0 \leq y \leq 1$

If solution not integer round closest

RELAXATION

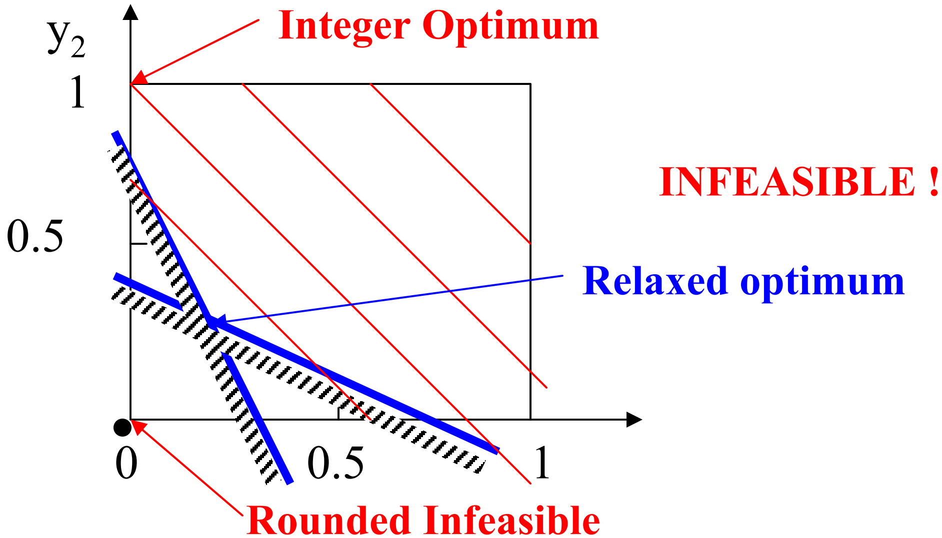
Only special cases yield integer optimum (*Assignment Problem*)

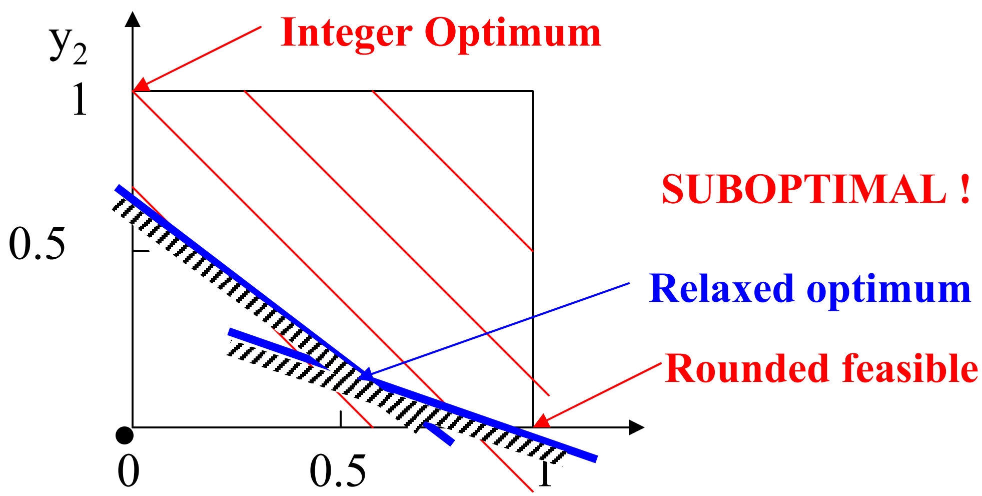
Relaxed LP provides LOWER BOUND to MILP solution

Difference: Relaxation gap

ROUNDING

May yield infeasible or suboptimal solution





Convert MIP into a Continuous NLP

Example: $\text{Min } Z = y1 + 2y2$

$\text{s.t. } 2y1 + y2 \geq 1$

$y1 = 0, 1 \quad y2 = 0, 1$

replace 0 – 1 conditions by

$0 \leq y1 \leq 1, \quad y1\,(1-y1) \leq 0$

$0 \leq y2 \leq 1 \quad y1\,(1-y2) \leq 0$

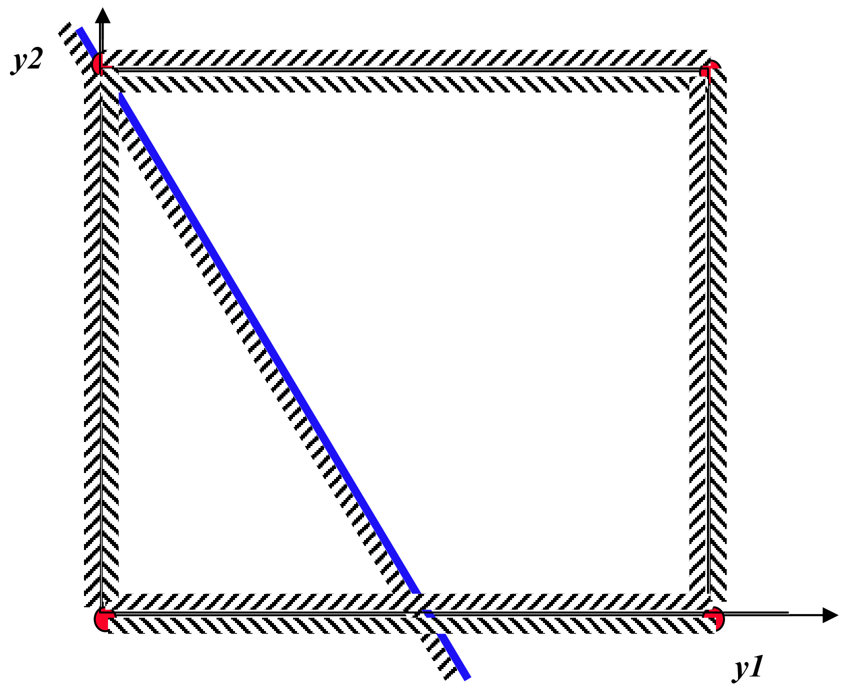
=> **Nonlinear Nonconvex!**

only feasible pts.

(1,0)

(0,1)

(1,1)



Using CONOPT2

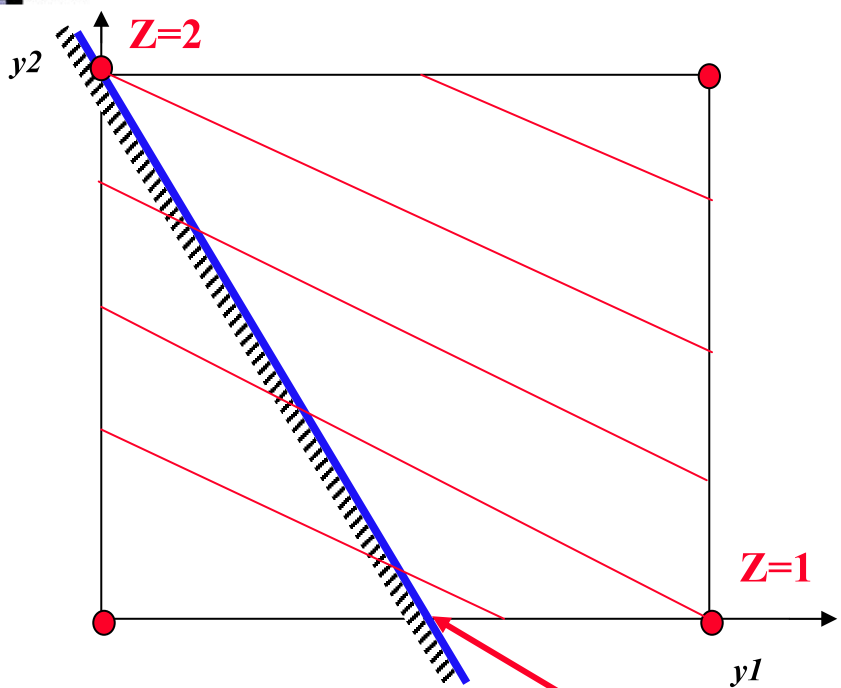
st. point $y_1 = 0, \; y_2 = 0$ => **infeasible**

st. pt. $y_1 = 0.5, \; y_2 = 0.5$ => $y_1 = 0 \; y_2 = 1$

$Z = 2$ **suboptimal**

correct solution $y_1 = 1, y_2 = 0 \quad Z = 1$

Branch and Bound



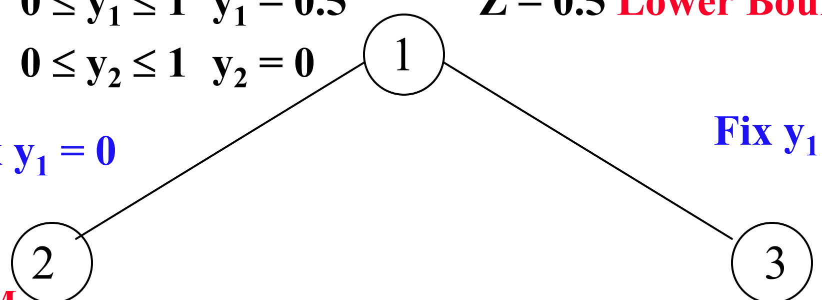
Tree Enumeration
Solve LP At Each Node

$$Min \quad y_1 + 2y_2$$
$$s.t. \quad 2y_1 + y_2 \geq 1 \qquad (P)$$
$$y1 = 0, 1 \qquad y2 = 0, 1$$

Solve MILP with

$0 \leq y_1 \leq 1$ $y_1 = 0.5$ **Z = 0.5 Lower Bound**

$0 \leq y_2 \leq 1$ $y_2 = 0$



Fix $y_1 = 0$

Fix $y_1 = 1$

$y_2 = 1$ Z = 2
OPTIMUM

$y_2 = 0$ 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

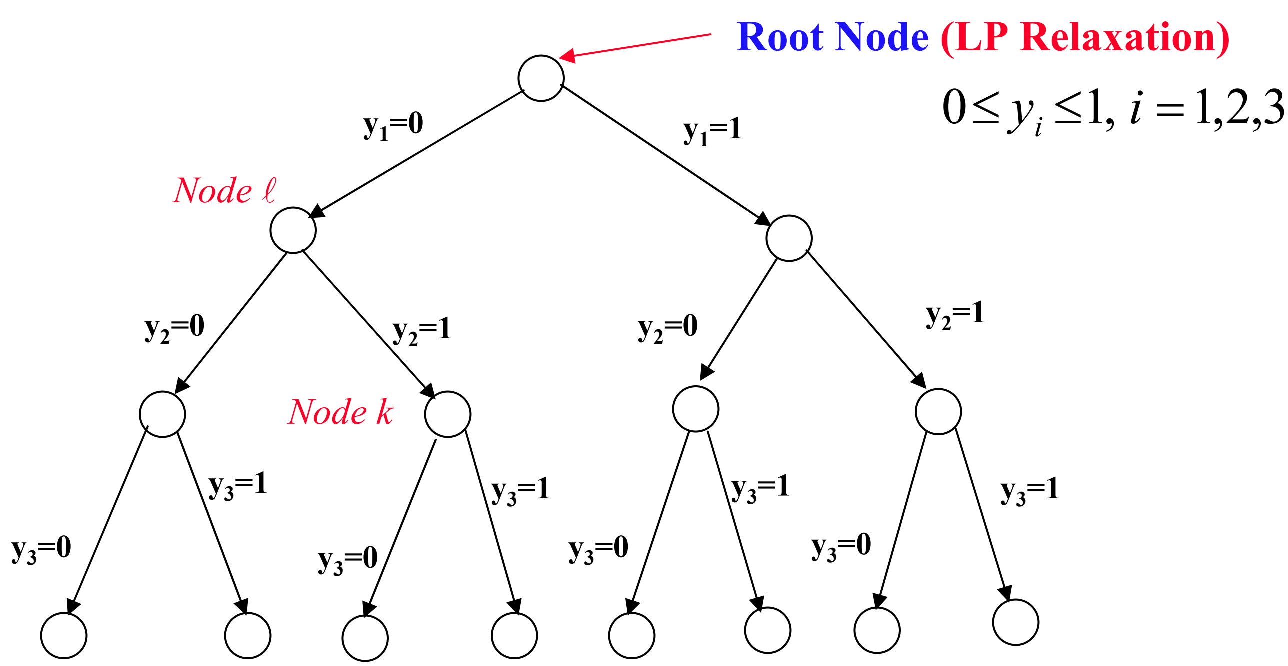
Remark

- 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 *ℓ*

NODE k: LP $\min Z = c^T x + b^T y$

$$\begin{aligned} \text{s.t.} \quad & Ax + By \le d \\ & x \ge 0 \qquad 0 \le y \le 1 \\ & y_i = 0 \text{ or } 1 \quad i \in I_k \end{aligned}$$

Since node k descendent of node ℓ

1. IF LP$^{\ell}$ INFEASIBLE THEN LP^k INFEASIBLE

2. IF LP^k FEASIBLE $Z^{\ell} \le Z^k$

   monotone increase objective

   Z^{ℓ} : LOWER BOUND

3. IF LP^k INTEGER $Z^k \le Z^*$

   Z^k : UPPER BOUND

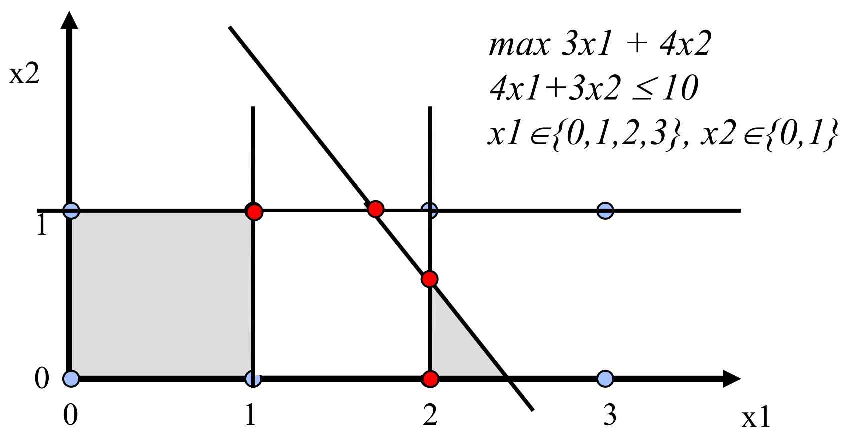
FATHOMING RULES: If node is infeasible
If Lower Bound exceeds Upper Bound

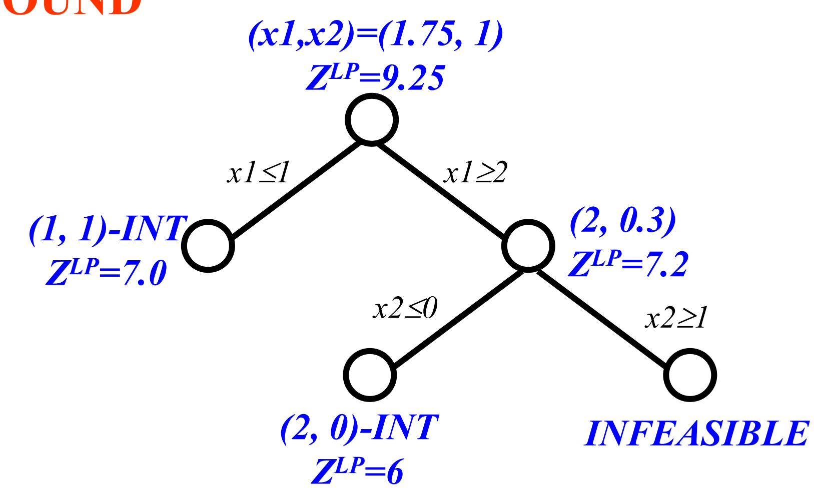
Carnegie Mellon

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



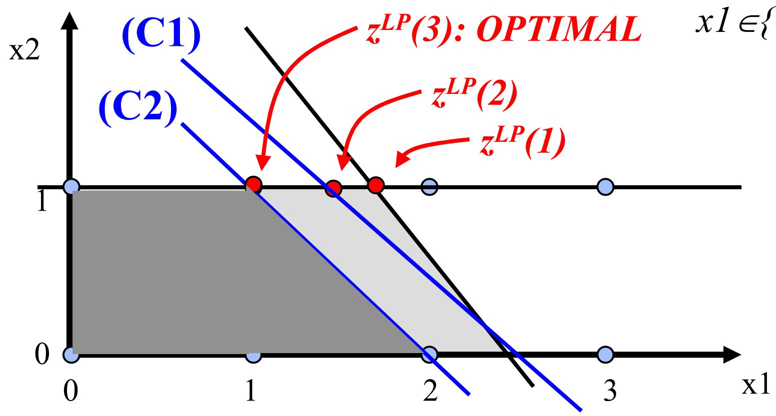


Mixed-integer Linear Programming

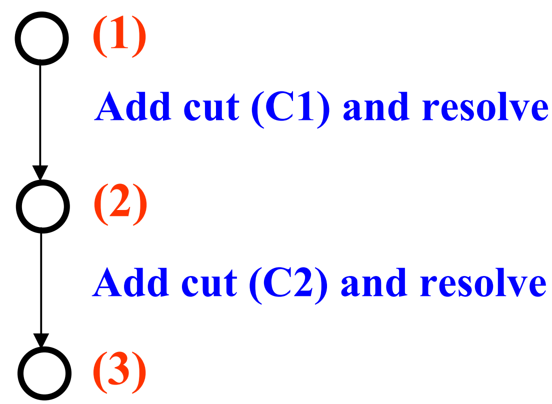
- Well known & widely applied
- Efficient algorithms for moderately sized problems
- Search is based on solution of relaxed problems

CUTTING PLANES

max $3x1 + 4x2$
$4x1+3x2 \le 10$
$x1 \in \{0,1,2,3\}$, $x2 \in \{0,1\}$



$x1 + x2 \le 2$



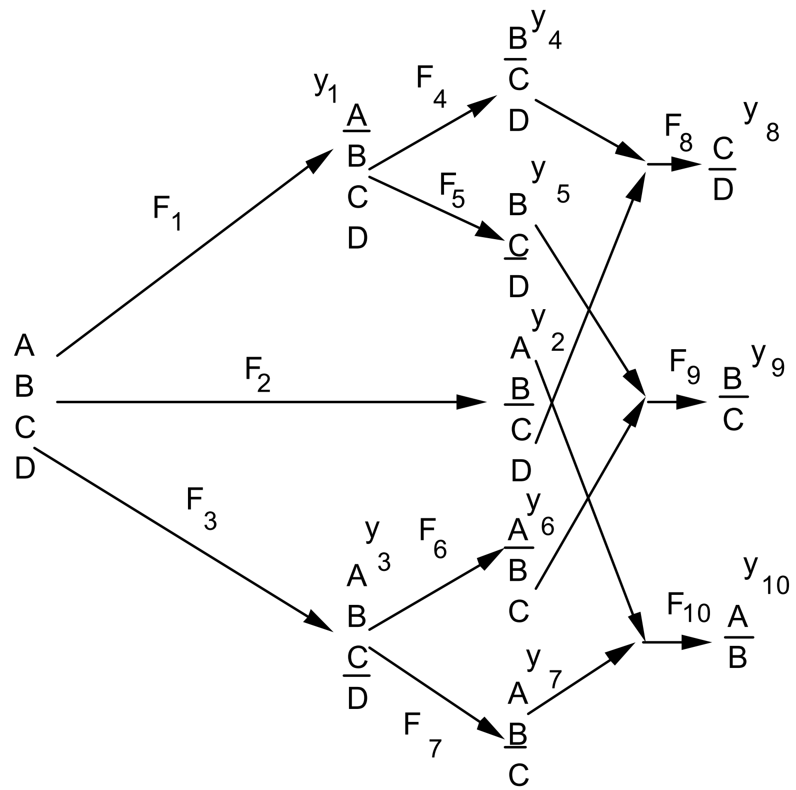
✓ **Solve entire problem at each node**

✓ **Exploit optimization information at each node**

Carnegie Mellon

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_{TOT} = 1000 kgmol/hr

Composition (mole fraction)

A	0.15
B	0.3
C	0.35
D	0.2

b) Economic data and heat duty coefficients

k	Separator	Investment cost α_k, fixed,	β_k, variable	Heat duty coefficients, K_k
1	A/BCD	145	0.42	0.028
2	AB/CD	52	0.12	0.042
3	ABC/D	76	0.25	0.054
4	A/BC	25	0.78	0.024
5	AB/C	44	0.11	0.039
6	B/CD	38	0.14	0.040
7	BC/D	66	0.21	0.047
8	A/B	112	0.39	0.022
9	B/C	37	0.08	0.036
10	C/D	58	0.19	0.044

Cost of utilities:

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

Steam C_H = 34 (10^3\$hr/10^6kJyr)

Split fractions in superstructure (using initial compositions)

$\xi_1^{A} = 0.15$

$\xi_1^{BCD} = 0.85$

$\xi_2^{AB} = 0.45$

$\xi_2^{CD} = 0.55$

$\xi_3^{ABC} = 0.8$

$\xi_3^{D} = 0.2$

$\xi_4^{B} = 0.353$

$\xi_4^{CD} = 0.647$

$\xi_5^{BC} = 0.765$

$\xi_5^{D} = 0.235$

$\xi_6^{A} = 0.188$

$\xi_6^{BC} = 0.812$

$\xi_7^{AB} = 0.5625$

$\xi_7^{C} = 0.44$

$\xi_8^{C} = 0.636$

$\xi_8^{D} = 0.364$

$\xi_9^{B} = 0.462$

$\xi_9^{C} = 0.538$

$\xi_{10}^{A} = 0.333$

$\xi_{10}^{B} = 0.667$

MILP model

Initial node in network

$$F_1 + F_2 + F_3 = 1000 \qquad (1)$$

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,

$$F_4 + F_5 - 0.85\,F_1 = 0 \qquad (2)$$

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

$$F_6 + F_7 - 0.8\,F_3 = 0 \qquad (3)$$

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

$$F_{10} - 0.45\,F_2 - 0.563\,F_7 = 0 \qquad (4)$$

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

$$F_9 - 0.765\,F_5 - 0.812\,F_6 = 0 \qquad (5)$$

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

$$F_8 - 0.55\,F_2 - 0.647\,F_4 = 0 \qquad (6)$$

Relating flows to the binary variables y:

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

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

Heat duties of condensers and reboliers, continuous variables Q_k , k = 1, ...10,

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

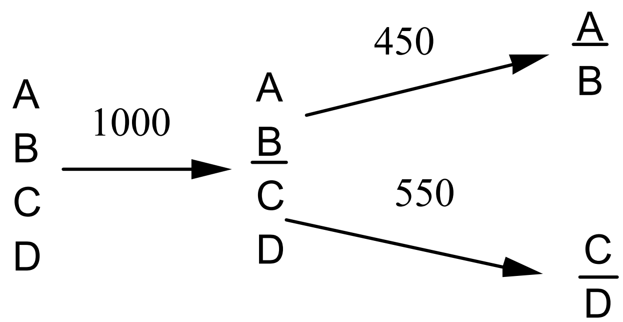
where the parameters K_k are given in Table.

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

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

cost coefficients α_k , β_k , are given in Table.

Optimal separation sequence

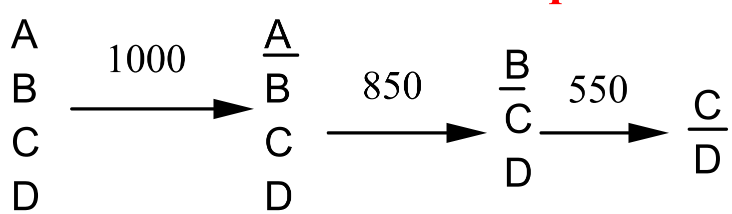


$3,308,000/yr

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

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

Second best sequence



$3,927,000/yr

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 = {columns k that have as feed the initial mixture}

e.g. $FS_F = \{1,2,3\}$

d) FS_m = {columns k that have as feed intermediate m}

e.g. for m = (BCD), $FS_m = \{4,5\}$

e) PS_m = {columns k that produce intermediate m}

e.g. for m = (CD), $PS_m = \{2,4\}$

$$\min \quad C = \sum_{k \in COL} \left[\alpha_k y_k + \beta_k F_k) + (C_H + C_C)\, Q_k \right]$$

$$\text{s.t.} \quad \sum_{k \in FS_F} F_k = F_{TOT}$$

(16)

$$\sum_{k \in FS_m} F_k - \sum_{k \in PS_m} \xi_k F_k = 0 \qquad m \in IP$$

$$Q_k - K_k F_k = 0 \qquad k \in COL$$

$$F_k - U y_k \leq 0 \qquad k \in COL$$

$$F_k, Q_k \geq 0, y_k = 0,1 \qquad k \in COL$$

where F_{TOT} is the flowrate of the initial mixture, ξ_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_{TOT} .

Modeling of Integer Programs

"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

Mathematical Modeling of Boolean Expressions

Williams (1988)

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 $\neg$ Pi

OR $P_1 \vee P_2 \vee \ldots \vee P_r$ $y_1 + y_2 + .. + y_r \geq 1$

AND $P_1 \wedge P_2 \wedge \ldots \wedge P_r$ $y_1 \geq 1, y_2 \geq 1, \ldots y_r \geq 1$

IMPLICATION $P_1 \Rightarrow P_2$

EQUIVALENT TO $\neg P_1 \vee P_2$ $1 - y_1 + y_2 \geq 1$

OR $y_2 \geq y_1$

Carnegie Mellon

EQUIVALENCE $P_1 \Leftrightarrow P_2$

$$(P_1 \Rightarrow P_2) \wedge (P_2 \Rightarrow P_1)$$

EQUIVALENT TO $(\neg\, P_1 \vee P_2) \wedge (\neg\, P_2 \wedge P_1)$

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

OR $\quad y_1 = y_2$

Systematic Procedure to Derive Linear Inequalities for Logic Propositions

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

$$Q_1 \wedge Q_2 \wedge \ldots \wedge Q_s$$

where clause Q_i : $P_1 \vee P_2 \vee \ldots \vee P_r$ *(Note: all OR)*

BASIC STEPS

1. REPLACE IMPLICATION BY DISJUNCTION

$$P1 \Rightarrow P2 \Leftrightarrow \neg P1 \vee P2$$

2. MOVE NEGATION INWARD APPLYING DE MORGAN'S THEOREM

$$\neg (P1 \wedge P2) \Leftrightarrow \neg P1 \vee \neg P2$$

$$\neg (P1 \vee P2) \Leftrightarrow \neg P1 \wedge \neg P2$$

3. RECURSIVELY DISTRIBUTE OR OVER AND

$$(P1 \wedge P2) \vee P3 \Leftrightarrow (P1 \vee P3) \wedge (P2 \vee P3)$$

EXAMPLE

flash $\Rightarrow$ dist $\vee$ abs

memb $\Rightarrow$ not abs $\wedge$ comp

$$P_F \Rightarrow P_D \vee P_A \qquad (1)$$
$$P_M \Rightarrow \neg P_A \wedge P_C \qquad (2)$$

(1) $\neg P_F \vee P_D \vee P_A$ *remove implication*

$$1 - y_F + y_D + y_A \geq 1$$
$$y_D + y_A \geq y_F$$

(2) $\neg P_M \vee (\neg P_A \wedge P_c)$ *remove implication*

$(\neg P_M \vee \neg P_A) \wedge (\neg P_M \vee P_c)$ *distribute OR over AND* **=> CNF!**

$$1 - y_M + 1 - y_A \geq 1 \qquad 1 - y_M + y_C \geq 1$$
$$y_M + y_A \leq 1 \qquad y_c \geq y_M$$

$$\boxed{\begin{array}{l} y_D + y_A \geq y_F \\ y_M + y_A \leq 1 \\ y_C \geq y_M \end{array}}$$

Verify: $y_F = 1 \quad y_D + y_A \geq 1 \qquad y_F = 0 \quad y_D + y_A \geq 0$

$y_M = 1 \Rightarrow y_A = 0 \qquad y_C = 1$

Carnegie Mellon

Chemical ENGINEERING

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. $x \wedge y \rightarrow z$
2. $\neg x \wedge \neg y \rightarrow z$
3. $x \wedge \neg y \rightarrow \neg z$
4. $\neg x \wedge y \rightarrow \neg z$

1) $x \wedge y \rightarrow z \Leftrightarrow \neg (x \wedge y) \vee z \Leftrightarrow \neg x \vee \neg y \vee z$

$$1 - x + 1 - y + z \geq 1$$

2) $\neg x \wedge \neg y \rightarrow z \Leftrightarrow \neg(\neg x \wedge \neg y) \vee z \Leftrightarrow x \vee y \vee z$

$$x + y + z \geq 1$$

3) $x \wedge \neg y \rightarrow \neg z \Leftrightarrow \neg(x \wedge \neg y) \vee \neg z \Leftrightarrow \neg x \vee y \vee \neg z$

$$1 - x + y + 1 - z \geq 1$$

4) $\neg x \wedge y \rightarrow \neg z \Leftrightarrow \neg(\neg x \wedge y) \vee \neg z \Leftrightarrow x \vee \neg y \vee \neg z$

$$x + 1 - y + 1 - z \geq 1$$

$$\boxed{\begin{aligned} z &\geq x + y - 1 \\ z &\geq 1 - x - y \\ z &\leq 1 - x + y \\ z &\leq 1 + x - y \end{aligned}}$$

EXAMPLE

Integer Cut

Constraint that is infeasible for integer point

$$y_i = 1 \quad i \in B \qquad\qquad y_i = 0 \quad i \in N$$

and feasible for all other integer points

Balas andJeroslow (1968)

Example: Multiperiod Problems

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

Intuitive Approach

$$y_1 + y_2 + \ldots + y_n \leq n^*z \qquad (1)$$

Logic Based Approach

$$y_1 \vee y_2 \vee \ldots \vee y_n \Rightarrow z$$

$$\neg(y_1 \vee y_2 \vee \ldots \vee y_n) \vee z$$

$$(\neg y_1 \wedge \neg y_2 \wedge \ldots \wedge \neg y_n) \vee z$$

$$(\neg y_1 \vee z) \wedge (\neg y_2 \vee z) \wedge \ldots \wedge (\neg y_n \vee z)$$

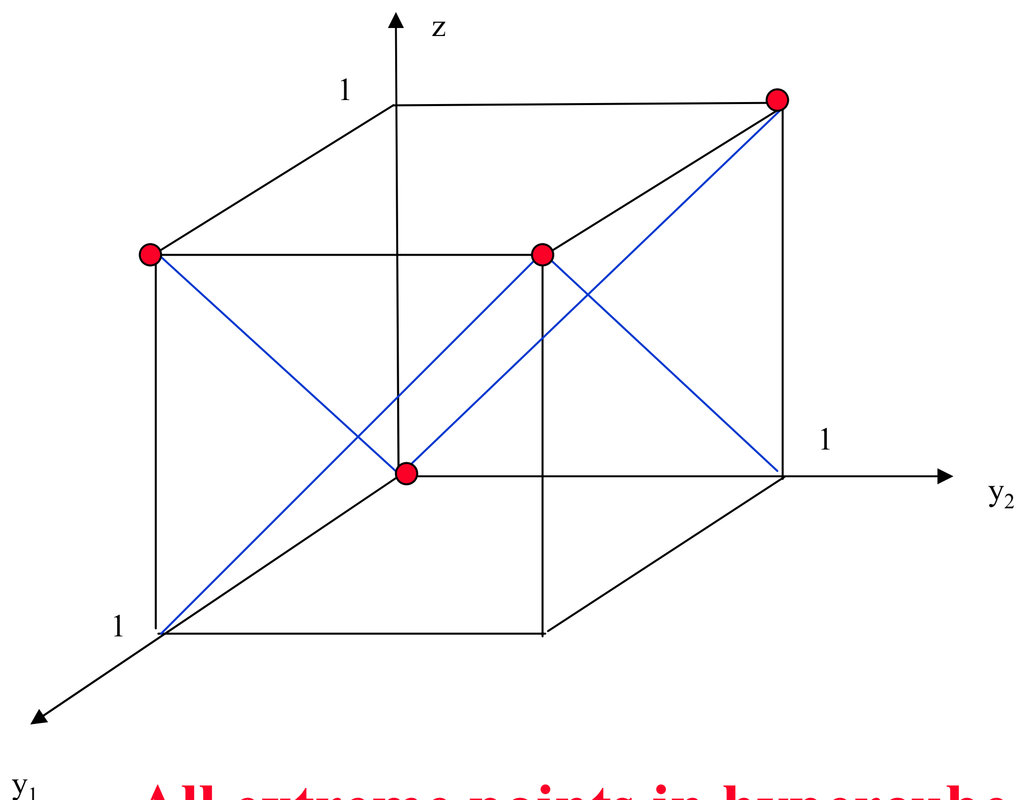
$$1 - y_1 + z \geq 1 \quad 1 - y_2 + z \geq 1 \quad 1 - y_n + z \geq 1$$

$$\boxed{\begin{array}{ll} y_1 \leq z & \\ y_2 \leq z & (2) \\ \vdots & \\ y_n \leq z & \end{array}}$$

Inequalities in (2) are stronger than inequalities in (1)

Geometrical interpretation

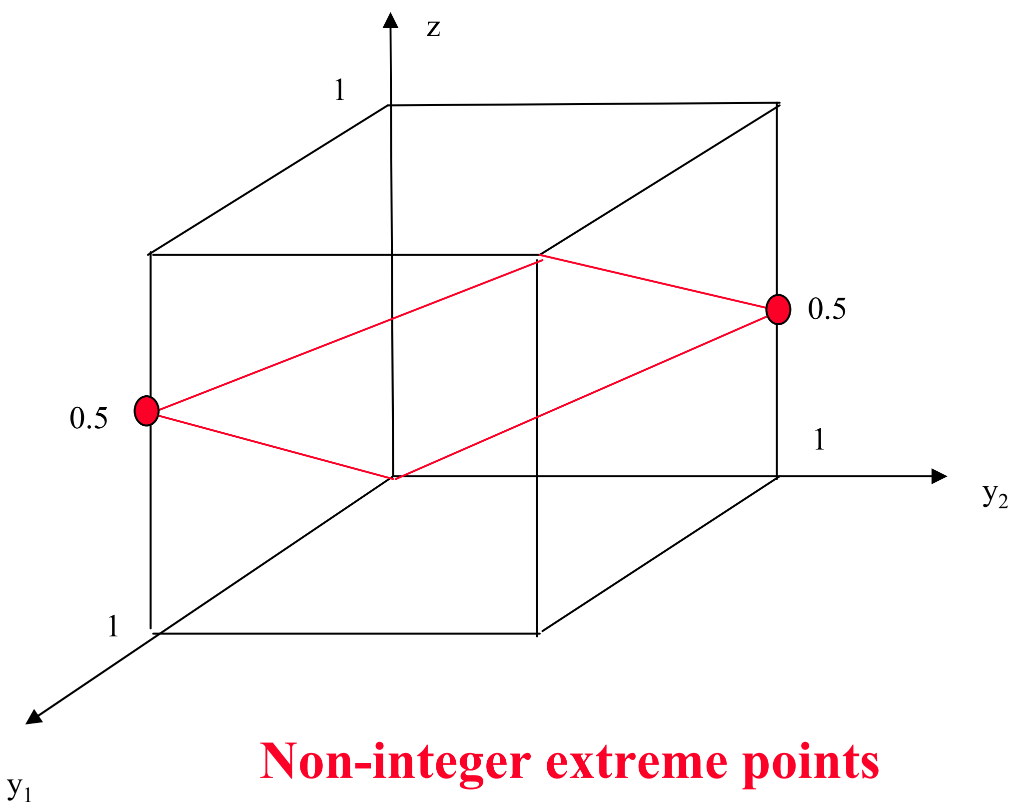
$$y_1 \leq z$$
$$y_2 \leq z$$



All extreme points in hypercube are integer!

Geometrical interpretation

$$y_1 + y_2 \leq 2z$$



Non-integer extreme points

Weaker relaxation!

Modeling of Disjunctions

$$\bigvee_{i \in D} \left[A_i x \le b_i \right]$$ one inequality must hold

Example: A before B *OR* B before A

$$\left[TS_A + pt_A \le TS_B \right] \vee \left[TS_B + pt_B \le TS_A \right]$$

Big M Formulation

$$A_i\, x \le b_i + M_i\,(1 - y_i) \qquad i \in D$$

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

Difficulty: Parameter M_i

Must be sufficiently large to render inequality redundant

Large value yields poor relaxation

Carnegie Mellon

Convex-hull Formulation (Balas, 1985)

$$x = \sum_{i \in D} z_i$$ disaggregation vars.

$$A_i z_i \le b_i y_i \qquad i \in D$$

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

$$0 \le z_i \le U y_i \qquad i \in D$$ (may be removed)

$$y_i = 0,\ 1$$

Derivation

$$A_i \ x \, y_i \le b_i \, y_i \qquad i \in D \qquad \text{(B)}$$ nonlinear disj. equiv.

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

Carnegie Mellon

Let $z_i = x\, y_i$ disaggregated variable

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

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

to ensure $z_i = 0$ if $y_i = 0$ (C)

$$0 \le z_i \le U y_i$$

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

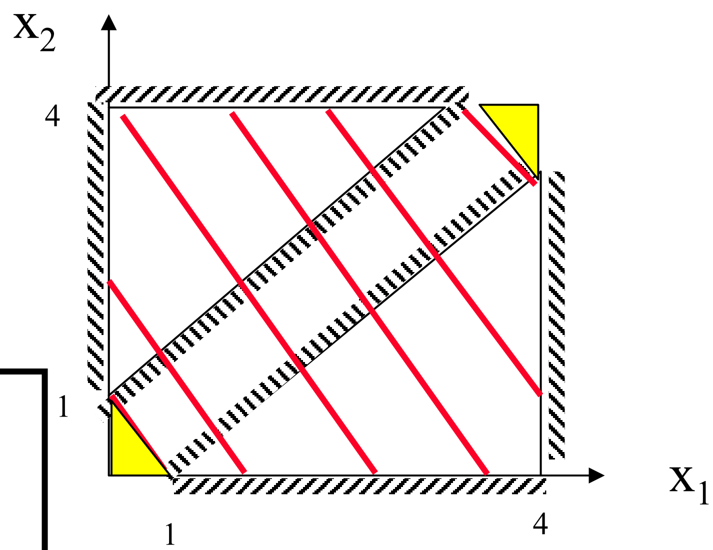
subst. (B) $A_i z_i \le b_i y_i \qquad i \in D$

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

(C) $\Rightarrow$ $0 \le z_i \le U y_i \qquad i \in D$

Example

$$[x_1 - x_2 \le -1] \vee [-x_1 + x_2 \le 1]$$
$$0 \le x_1, x_2 \le 4$$



big M

$$x_1 - x_2 \le -1 + M(1 - y_1)$$
$$-x_1 + x_2 \le -1 + M(1 - y_2)$$
$$y_1 + y_2 = 1 \quad M = 10 \text{ possible choice}$$

Convex hull

$$x_1 = z_1^1 + z_1^2$$
$$x_2 = z_2^1 + z_2^2$$
$$z_1^1 - z_2^1 \le -y_1 \qquad -z_1^2 + z_2^2 \le -y_2$$
$$y_1 + y_2 = 1$$
$$0 \le z_1^1 \le 4y_1$$
$$0 \le z_1^2 \le 4y_2$$
$$0 \le z_2^1 \le 4y_1$$
$$0 \le z_2^2 \le 4y_2$$

Nonlinear Programming

NLP: Algorithms *(variants of Newton's method)*

- Sucessive 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

Carnegie Mellon

MINLP

- **Mixed-Integer Nonlinear Programming**

$$\min Z = f(x,y)$$ *Objective Function*

$$s.t. \quad g(x,y) \le 0$$ *Inequality Constraints*

$$x \in X,\ y \in Y$$

$$X = \{x \mid x \in R^n, x^L \le x \le x^U, Bx \le b\}$$

$$Y = \{y \mid y \in \{0,1\}^m, Ay \le a\}$$

- ***f(x,y)* and *g(x,y)* - assumed to be convex and bounded over X.**
- ***f(x,y)* and *g(x,y)* commonly linear in y**

Carnegie Mellon

Solution Algorithms

- **Branch and Bound method (BB)**

Ravindran and Gupta (1985) Leyffer and Fletcher (2001)
Branch and cut: *Stubbs and Mehrotra (1999)*

- **Generalized Benders Decomposition (GBD)**

Geoffrion (1972)

- **Outer-Approximation (OA)**

Duran & Grossmann (1986), Yuan et al. (1988), Fletcher & Leyffer (1994)

- **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*

$$\begin{aligned} &\min Z_{LB}^k = f(x,y) \\ s.t.\ \ & g_j(x,y) \le 0 \quad j \in J \\ & x \in X,\ y \in Y_R \\ & y_i \le \alpha_i^k \ \ i \in I_{FL}^k \\ & y_i \ge \beta_i^k \ \ i \in I_{FU}^k \end{aligned} \qquad \text{(NLP1)}$$

b) NLP Fixed y^k *Upper bound*

$$\begin{aligned} &\min Z_U^k = f(x,y^k) \\ s.t.\ \ & g_j(x,y^k) \le 0 \quad j \in J \\ & x \in X \end{aligned} \qquad \text{(NLP2)}$$

c) Feasibility subproblem for fixed y^k.

$$\begin{aligned} &\min u \\ s.t.\ \ & g_j(x,y^k) \le u \quad j \in J \\ & x \in X,\ u \in R^1 \end{aligned} \qquad \text{(NLPF)}$$

Infinity-norm

Cutting plane MILP master

(Duran and Grossmann, 1986)

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

Lower Bound

M-MIP

$$\min Z_L^K = \alpha$$

$$\left.\begin{aligned} st\ \ \alpha &\geq f(x^k, y^k) + \nabla f(x^k, y^k)^T \begin{bmatrix} x - x^k \\ y - y^k \end{bmatrix} \\ & g_j(x^k, y^k) + \nabla g_j(x^k, y^k)^T \begin{bmatrix} x - x^k \\ y - y^k \end{bmatrix} \leq 0 \quad j \in J \end{aligned}\right\} k = 1, \dots K$$

$$x \in X,\ y \in Y$$

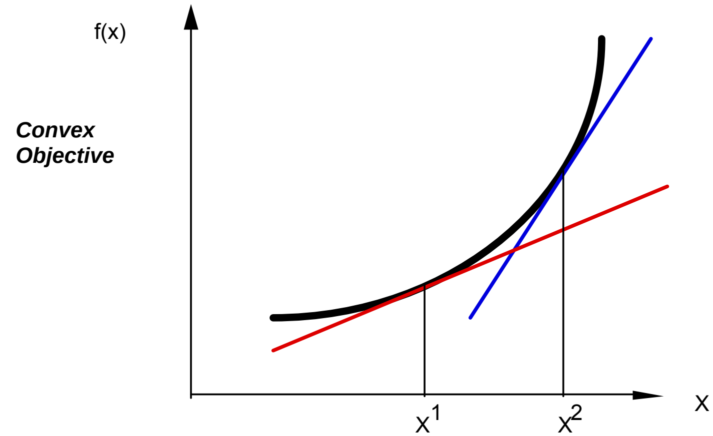
Notes:

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

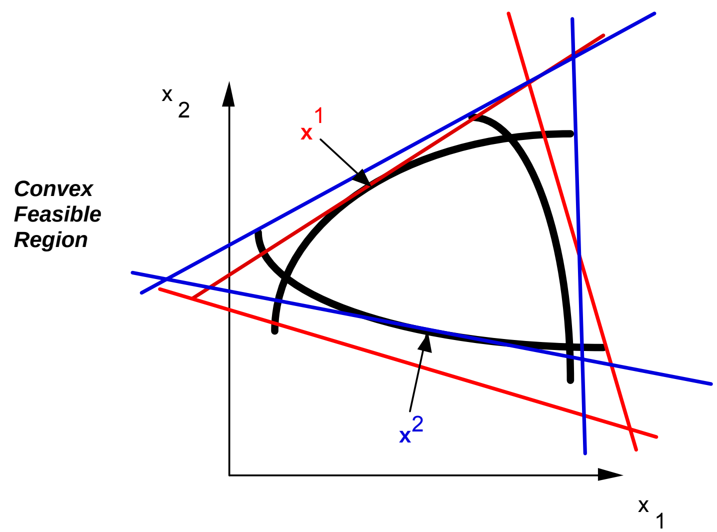
b) Linearizations *accumulated* as iterations K increase

c) Non-decreasing sequence lower bounds

Linearizations and Cutting Planes



Underestimate Objective Function



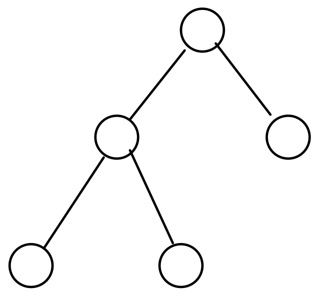
Overestimate Feasible Region

Branch and Bound

NLP1:

$$\min \; Z_{LB}^{k} = f(x,y)$$
$$s.t. \quad g_j(x,y) \le 0 \qquad j \in J$$
$$x \in X, \quad y \in Y_R$$
$$y_i \le \alpha_i^k \qquad i \in I_{FL}^k$$
$$y_i \ge \beta_i^k \qquad i \in I_{FU}^k$$

Tree Enumeration



Successive solution of NLP1 subproblems

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

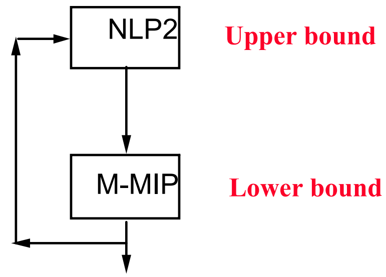
Disadvantage:
Potentially many NLP subproblems

Convergence global optimum:
Uniqueness solution NLP1 *(sufficient condition)*

Less stringent than other methods

Outer-Approximation

Alternate solution of NLP and MIP problems:



NLP2:

$$\begin{aligned} &\min Z_U^k = f(x, y^k) \\ s.t.\ \ & g_j(x, y^k) \le 0 \quad j \in J \\ & x \in X \end{aligned}$$

M-MIP:

$$\begin{aligned} &\min \quad Z_L^K = \alpha \\ st \quad & \left.\begin{aligned} \alpha &\ge f(x^k, y^k) + \nabla f(x^k, y^k)^T \begin{bmatrix} x - x^k \\ y - y^k \end{bmatrix} \\ & g_j(x^k, y^k) + \nabla g_j(x^k, y^k)^T \begin{bmatrix} x - x^k \\ y - y^k \end{bmatrix} \le 0 \quad j \in J^k \end{aligned}\right\} \; k = 1, \ldots K \\ & x \in X, \quad y \in Y \end{aligned}$$

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 objetive lies below current upper bound UB^K:

(M-MIPF)

$$
\begin{aligned}
& \min \quad Z_L^K = 0\alpha \\
s.t. \quad & \boxed{\alpha \le UB^K - \varepsilon} \\
& \left.\begin{aligned}
& \alpha \ge f(x^k, y^k) + \nabla f(x^k, y^k)^T \begin{bmatrix} x - x^k \\ y - y^k \end{bmatrix} \\
& g_j(x^k, y^k) + \nabla g_j(x^k, y^k)^T \begin{bmatrix} x - x^k \\ y - y^k \end{bmatrix} \le 0 \quad j \in J
\end{aligned}\right\} \quad k = 1, .. K \\
& x \in X, \;\; y \in Y
\end{aligned}
$$

Remark.

M-MIPF will tend to increase number of iterations

Carnegie Mellon

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)

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

$$g(x^k,y^k) + \nabla g_j(x^k,y^k)^T \begin{bmatrix} x-x^k \\ y-y^k \end{bmatrix} \leq 0 \quad j \in J^k \qquad (1)$$

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

$$\left.\begin{aligned} \alpha \geq f(x^k,y^k) + \nabla_y f(x^k,y^k)^T (y-y^k) \\ + (\mu^k)^T \left[g(x^k,y^k) + \nabla_y g(x^k,y^k)^T (y-y^k) \right] \end{aligned}\right\} \qquad (2)$$

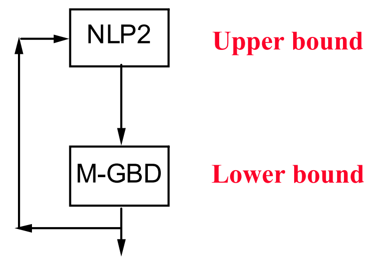
Lagrangian cut

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

$$(\lambda^k)^T \left[g(x^k,y^k) + \nabla_y g(x^k,y^k)^T (y-y^k) \right] \leq 0$$

Generalized Benders Decomposition

Alternate solution of NLP and MIP problems:



NLP2:

$$
\begin{aligned}
&\min Z_U^k = f(x, y^k) \\
&s.t. \;\; g_j(x, y^k) \le 0 \quad j \in J \\
&x \in X
\end{aligned}
$$

M-GBD:

$$
\begin{aligned}
&\min \;\; Z_L^K = \alpha \\
&st \;\; \alpha \ge f(x^k, y^k) + \nabla_y f(x^k, y^k)^T \left(y - y^k\right) \\
&\quad + \left(\mu^k\right)^T \left[g(x^k, y^k) + \nabla_y g\,(x^k, y^k)^T \left(y - y^k\right) \right] \quad k \in KFS \\
&\left(\lambda^k\right)^T \left[g(x^k, y^k) + \nabla_y g\,(x^k, y^k)^T \left(y - y^k\right) \right] \le 0 \quad k \in KIS \\
&y \in Y, \;\; \alpha \in R^1
\end{aligned}
$$

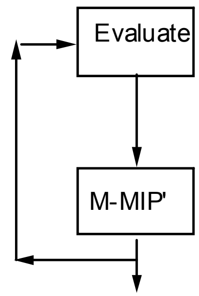
Property 1. If problem (P1) has zero integrality gap, Generalized Benders Decomposition converges in one iteration when optimal *(x^k, y^k)* are found.

Sahinidis, Grossmann (1991)

=> Also applies to Outer-Approximation

Extended Cutting Plane

Westerlund and Pettersson (1995)



No NLP !

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

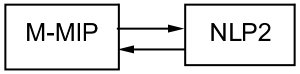
- Simultaneously optimize x^k, y^k with M-MIP

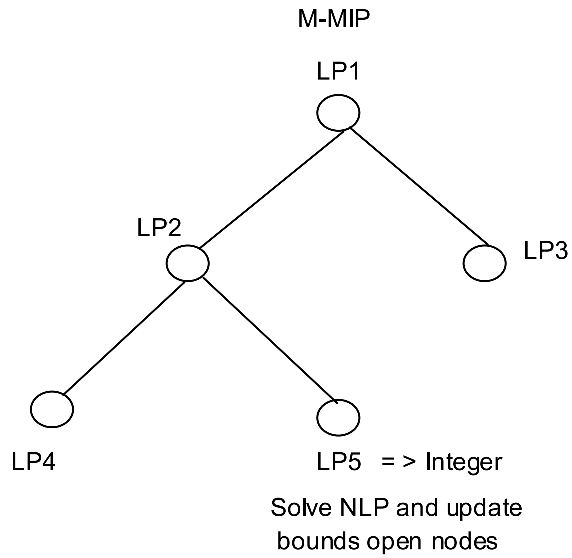
= > Convergence may be slow

LP/NLP Based Branch and Bound *(Branch & Cut)*

Quesada and Grossmann (1992)

Integrate NLP and M-MIP problems





Remark.

Fewer number branch and bound nodes for LP subproblems

May increase number of NLP subproblems

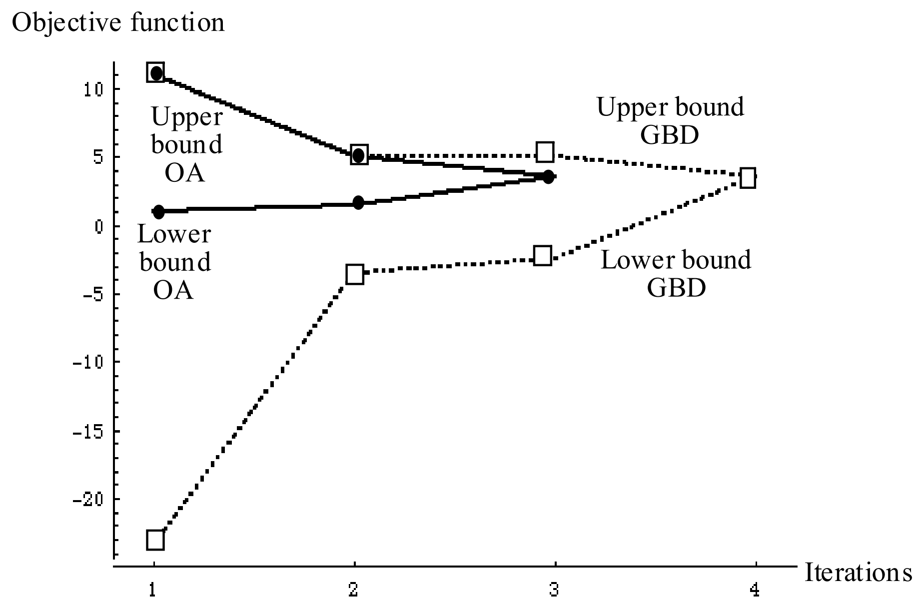
Numerical Example

$$\min Z = y_1 + 1.5y_2 + 0.5y_3 + x_1^2 + x_2^2$$

$$\begin{aligned}
\text{s.t.}\quad & (x_1 - 2)^2 - x_2 \le 0 \\
& x_1 - 2y_1 \ge 0 \\
& x_1 - x_2 - 4(1-y_2) \le 0 \\
& x_1 - (1 - y_1) \ge 0 \\
& x_2 - y_2 \ge 0 \qquad\qquad\qquad\qquad \text{(MIP-EX)} \\
& x_1 + x_2 \ge 3y_3 \\
& y_1 + y_2 + y_3 \ge 1 \\
& 0 \le x_1 \le 4, \quad 0 \le x_2 \le 4 \\
& y_1, y_2, y_3 = 0, 1
\end{aligned}$$

Optimum solution: $y_1=0$, $y_2 = 1$, $y_3 = 0$, $x_1 = 1$, $x_2 = 1$, $Z = 3.5$.

Starting point $y_1 = y_2 = y_3 = 1.$

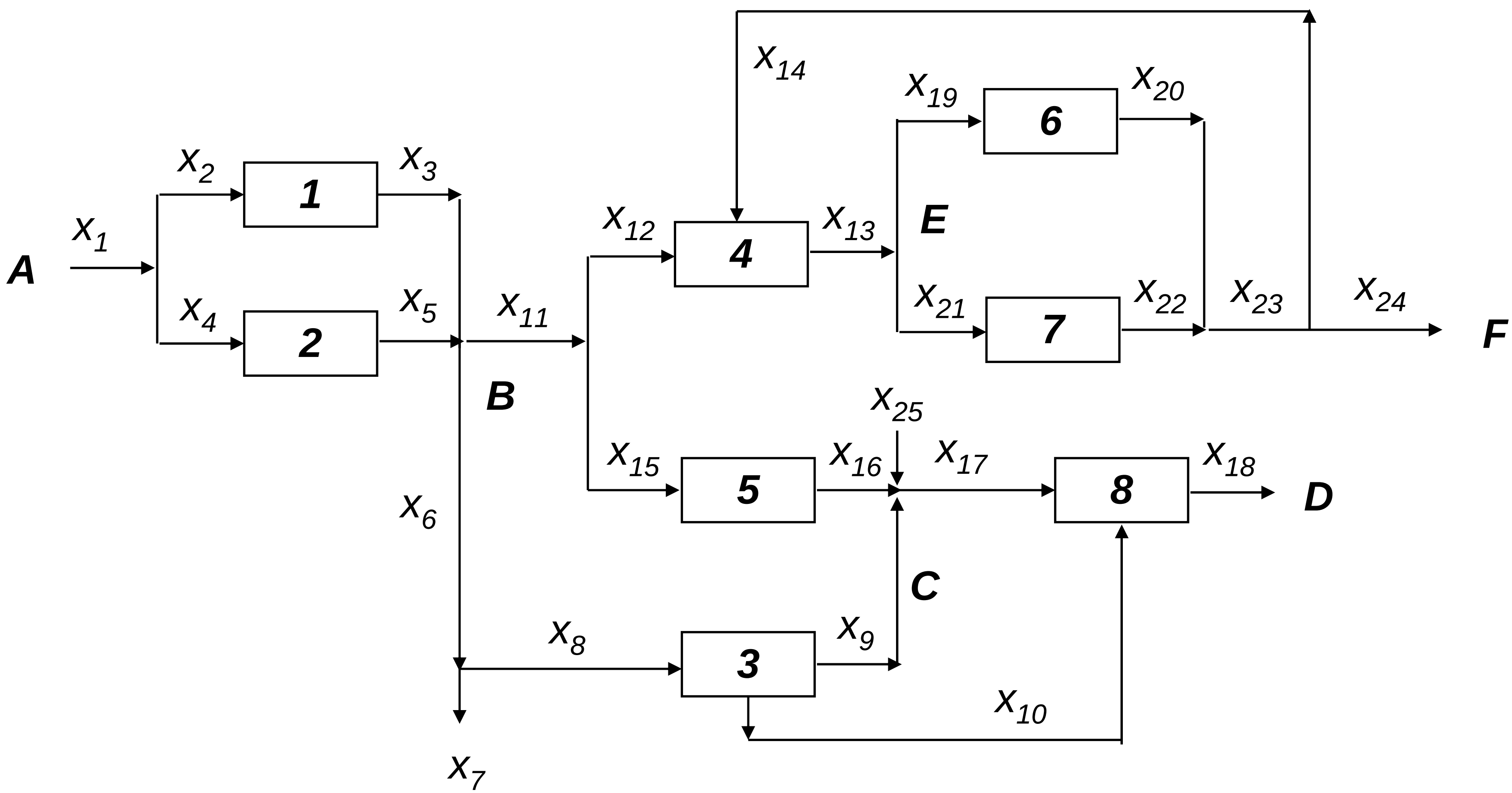


Summary of Computational Results

Method	Subproblems	Master problems (LP's solved)
BB	5 (NLP1)	
OA	3 (NLP2)	3 (M-MIP) (19 LP's)
GBD	4 (NLP2)	4 (M-GBD) (10 LP's)
ECP	-	5 (M-MIP) (18 LP's)

Example: Process Network with Fixed Charges

- *Duran and Grossmann (1986)*
  - **Network superstructure**



Example *(Duran and Grossmann, 1986)*

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

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

	NLP	MIP
Branch and Bound *(F-L)*	20	-
Outer-Approximation:	3	3
Generalized-Benders	10	10
Extended Cutting Plane	-	15
LP/NLP based	3	7 LP's vs 13 LP's OA

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 understimator => *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

2. Equality relaxation in OA *Kocis and Grosmann (1987)*

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

$T^k = \{t_{ii}^k\}$, $t_{ii}^k = sign(\lambda_i^k)$, λ_i^k multiplier equation $h_i(x, y) = 0$

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

$$\min \quad Z^K = \alpha + \sum_{k=1}^{K} \left[w_p^k p^k + w_q^k q^k \right] \qquad \text{(M-APER)}$$

$$
\left.
\begin{aligned}
s.t. \quad & \alpha \geq f(x^k, y^k) + \nabla f(x^k, y^k)^T \begin{bmatrix} x - x^k \\ y - y^k \end{bmatrix} \\
& T^k \, \nabla h(x^k, y^k)^T \begin{bmatrix} x - x^k \\ y - y^k \end{bmatrix} \leq p^k \\
& g(x^k, y^k) + \nabla g(x^k, y^k)^T \begin{bmatrix} x - x^k \\ y - y^k \end{bmatrix} \leq q^k
\end{aligned}
\right\} \; k = 1, \ldots K
$$

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

$$x \in X, \; y \in Y, \; \alpha \in \boldsymbol{R}^1, \; p^k, q^k \geq 0$$

If convex MINLP then slacks take value of zero
=> 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 Peterssson (1996)*
MINOPT *Schweiger and Floudas (1998)*
BARON *Sahinidis et al. (1998)*
SBB *GAMS simple B&B*

Test Problems:

GAMSWorld: http://www.gamsworld.org/minlp
Leyffer-AMPL: http://www-unix.mcs.anl.gov/~leyffer/MacMINLP/
Floudas MINOPT: http://titan.princeton.edu/MINOPT/library-tests.html

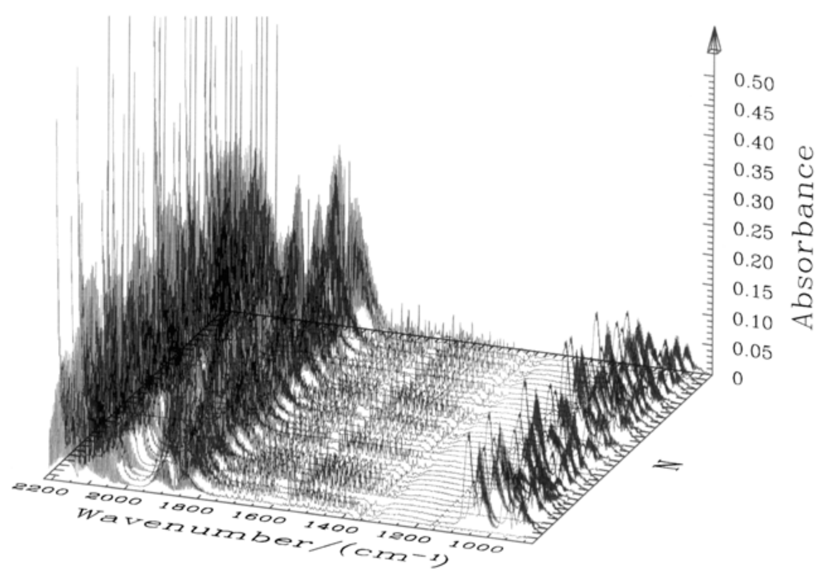
Parameter Estimation FTIR-Spectroscopy

Brink and Westerlund (1995)

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

c_i^k concentration component i, run k

a_j^k absorbence of mixture wave number j, run k



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

$$c_i = \sum_{j=1}^{q} p_{ij} a_j \qquad i = 1,..n$$

Carnegie Mellon

Derivation of Model

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

Akaike information criterion

$$\min AIC = -2\,\ell n L + 2P$$

P number of parameters p_{ij}

L likelihood function

$$\ell n L = -\frac{N}{2} n \ell n(2\pi) + \frac{N}{2}\ell n\left(\left|R^{-1}\right|\right) - \frac{1}{2}\sum_{k=1}^{N} e_k^{\mathrm{T}} R^{-1} e_k$$

R covariance matrix

Let $y_{ij} = \begin{cases} 1 & if \;\; p_{ij}^L \le p_{ij} \le p_{ij}^U \\ 0 & if \; p_{ij} = 0 \end{cases}$

Chemical ENGINEERING

MINLP Model *(MIQP)*

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

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

$$p_{ij}^L y_{ij} \le p_{ij} \le p_{ij}^U y_{ij} \quad i = 1,..n \quad j = 1,..q$$

$$y_{ij} = 0,1 \qquad p_{ij} \ge 0$$

Carnegie Mellon

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)

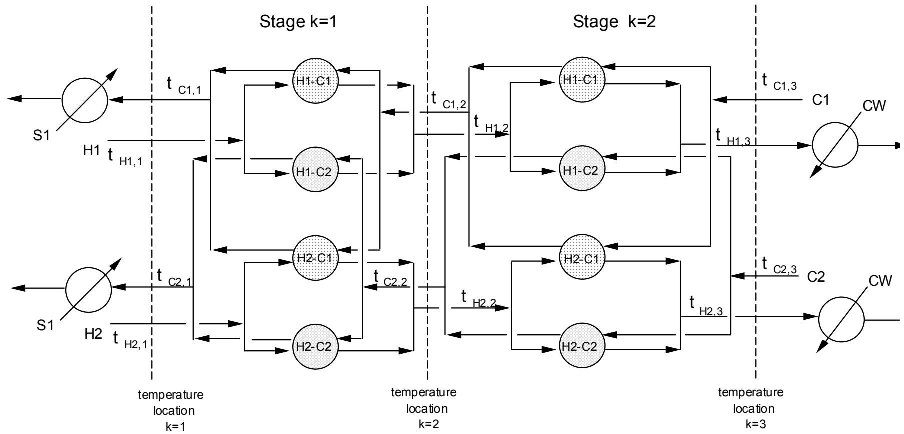
5 selected parameters

$p_{CO,87} = 714.78_{\,2016.2-2030.3}$ $p_{CO,97} = 65.07_{\,2157.6-2171.7}$

$p_{NO,83} = 315.57_{\,1959.6-1973.7}$

$p_{CO_2,10} = 22.47_{\,927.3-941.4}$ $p_{CO_2,20} = 25.01_{\,1068.7-1082.8}$

MINLP Model for Superstructure Optimization of Heat Exchanger Network Synthesis Yee and Grossmann (1990)



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

dt_{ijk} = temperature approach for match (i,j) at temperature location k

$dtcu_i$ = temperature approach for the match of hot stream i and cold utility

$dthu_j$ = temperature approach for the match of cold stream j and hot utility

q_{ijk} = heat exchanged between hot process stream i and cold process stream j in stage k

qcu_i = heat exchanged between hot stream i and cold utility

qhu_j = heat exchanged between hot utility and cold stream j

$t_{i,k}$ = temperature of hot stream i at hot end of stage k

$t_{j,k}$ = temperature of cold stream j at hot end of stage k

z_{ijk} = binary variable to denote existence of match (i,j) in stage k

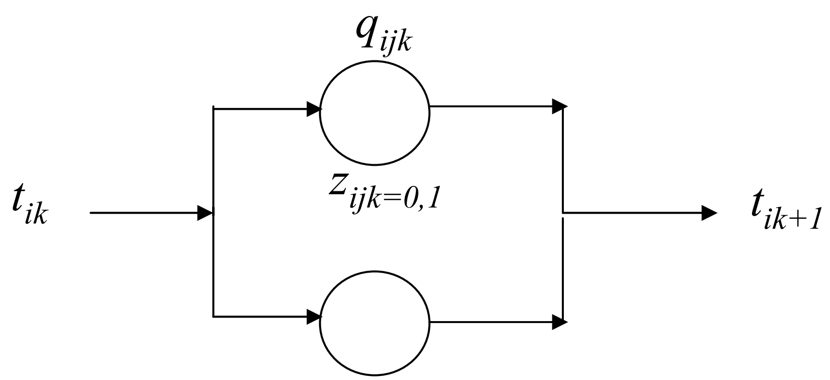
zcu_i = binary variable to denote that cold utility exchanges heat with stream i

zhu_j = binary variable to denote that hot utility exchanges heat with stream j

Carnegie Mellon

Assumption

Isothermal mixing => linear constraints



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}

Carnegie Mellon

Overall heat balance for each stream

$$(TOUT_j - TIN_j)\, F_j = \sum_{k \in ST} \sum_{i \in HP} q_{ijk} + qhu_j \quad j \in CP$$

$$(TIN_i - TOUT_i)\, F_i = \sum_{k \in ST} \sum_{j \in CP} q_{ijk} + qcu_i \quad i \in HP$$

Heat balance at each stage

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

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

Feasibility of temperatures

$t_{i,k} \geq t_{i,k}+1 \qquad k \in ST,\ i \in HP$

$t_{j,k} \geq t_{ji,k}+1 \qquad k \in ST,\ j \in CP$

$TOUTi \geq ti_{,NOK+1} \qquad i \in HP$

$TOUTj \geq t_{j,1} \qquad j \in CP$

Hot and cold utility load

$(ti_{,NOK+1} - TOUT_i)\, F_i = qcu_i \quad i \in HP$

$(TOUTj - t_{j,1})\, Fj = qhu_j \qquad j \in CP$

Logical constraints

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

$qcu_i - \Omega zcu_i \leq 0 \quad i \in HP$

$qhu_j - \Omega zhu_j \leq 0 \quad j \in CP$

$z_{ijk}, zcu_i, zhuj = 0,1$

Calculation of approach temperatures

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

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

$dtcu_i \leq t_{i,NOK+1} - TOUTCU + \Gamma(1 - zcu_i) \quad i \in HP$

$dthu_i \leq TOUTHU - t_{j,1} + \Gamma(1 - zhu_j) \quad j \in CP$

$dt_{ijk} \geq EMAT$

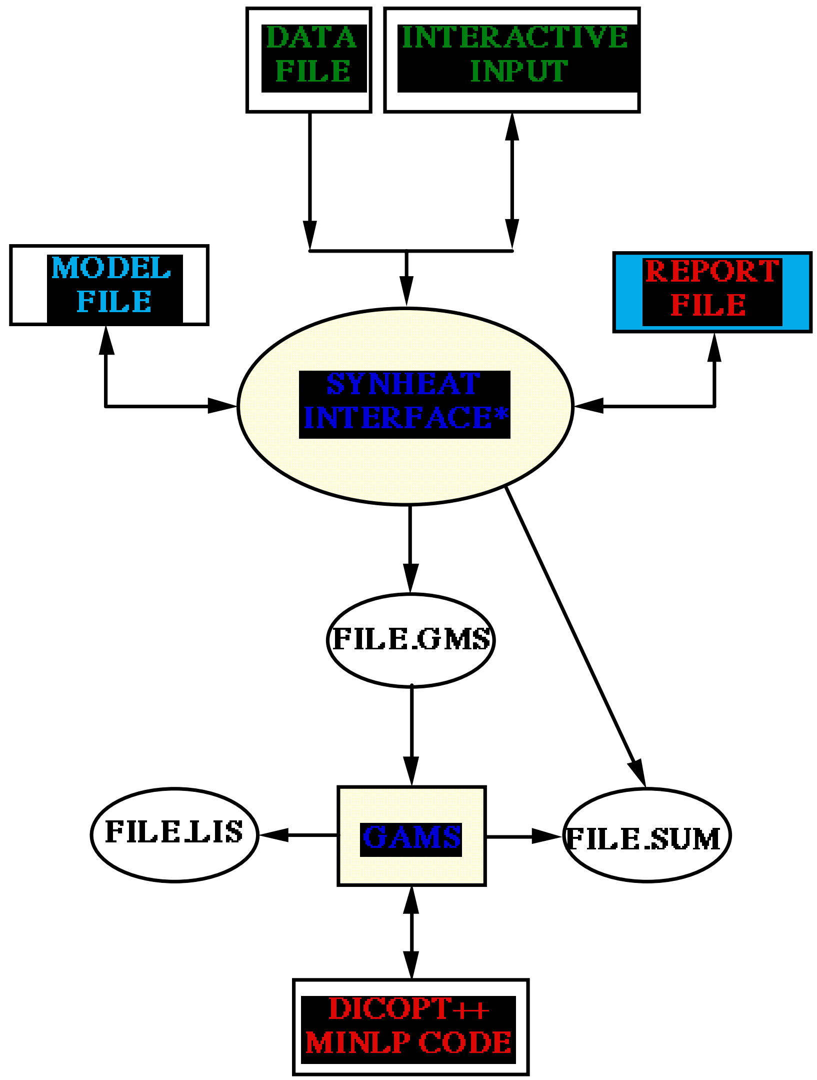
Objective function

Chen approximation (1987)

*LMTD ~ [(dtl*dt2)*(dtl+dt2)/2]1/3*

$$\min \; Z = \sum_{i \in HP} CCUqcu_i + \sum_{j \in CP} CHUqhu_j$$
$$+ \sum_{i \in HP} \sum_{j \in CP} \sum_{k \in ST} CF_{ij} z_{ijk} + \sum_{i \in HP} CF_{i,CU} zcu_i + \sum_{j \in CP} CF_{i,HU} zhu_j$$
$$+ \sum_{i \in HP} \sum_{j \in CP} \sum_{k \in ST} \frac{C_{ij} q_{ijk}}{U_{ijk}[(dt_{ijk})(dt_{ijk+1})(dt_{ijk} + dt_{ijk+1})/2)]^{1/3}} + ...etc.$$

STRUCTURE OF SYNHEAT



```
*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;
HUCOEFF = 20.00;
CUCOEFF = 20.00;
```

Carnegie Mellon

MODEL STATISTICS

BLOCKS OF EQUATIONS	27	SINGLE EQUATIONS	73
BLOCKS OF VARIABLES	16	SINGLE VARIABLES	61
NON ZERO ELEMENTS	261	NON LINEAR N-Z	32
DERIVATIVE POOL	36	CONSTANT POOL	17
CODE LENGTH	1586	DISCRETE VARIABLES	12

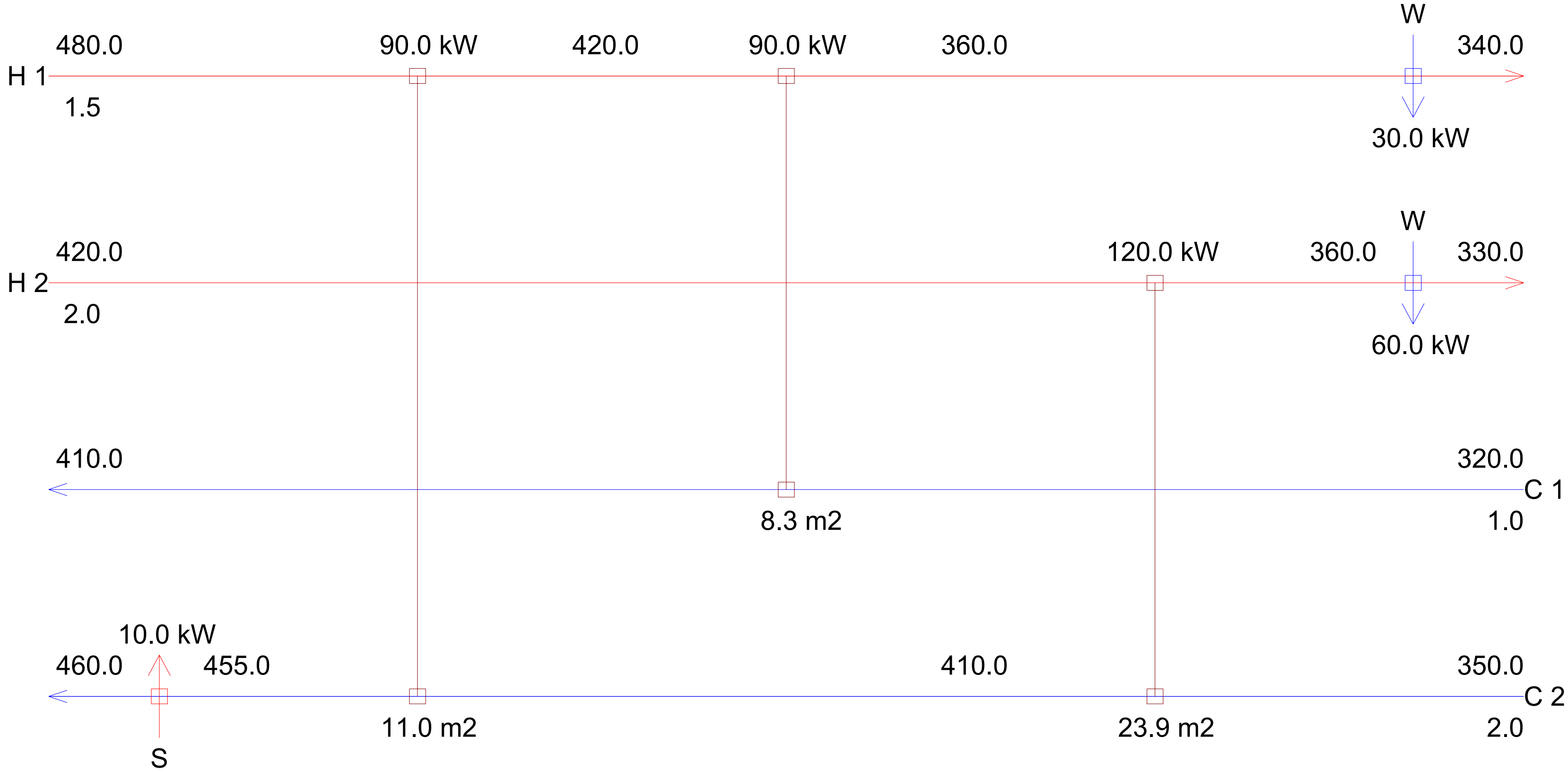
DICOPT Log File

Major Step	Major Iter	Objective Function	CPU time (Sec)	Itera- tions	Evaluation Errors	Solver
NLP	1	4272.27511	0.02	135	0	minos5
MIP	1	32732.40039	0.01	101	0	xa
NLP	2	8962.59738<	0.02	57	0	minos5
MIP	2	33732.40039	0.02	102	0	xa
NLP	3	9962.59738	0.01	14	0	minos5

Total solver times : NLP = 0.05 MIP = 0.03

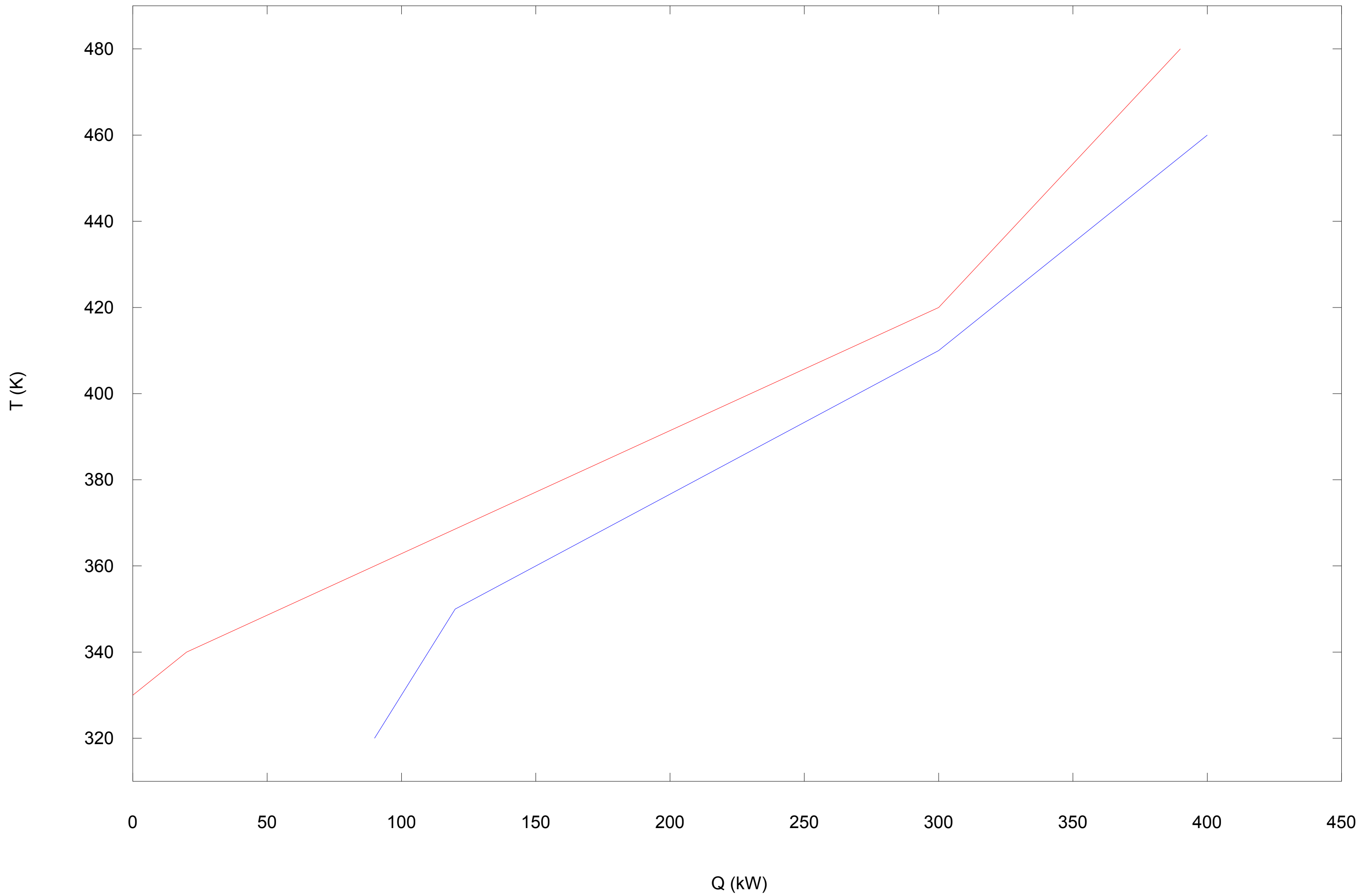
Perc. of total : NLP = 60.98 MIP = 39.02

Total Network Cost ($/yr) = 8962.60



Carnegie Mellon

SYNHEAT: T-Q CURVE



Carnegie Mellon

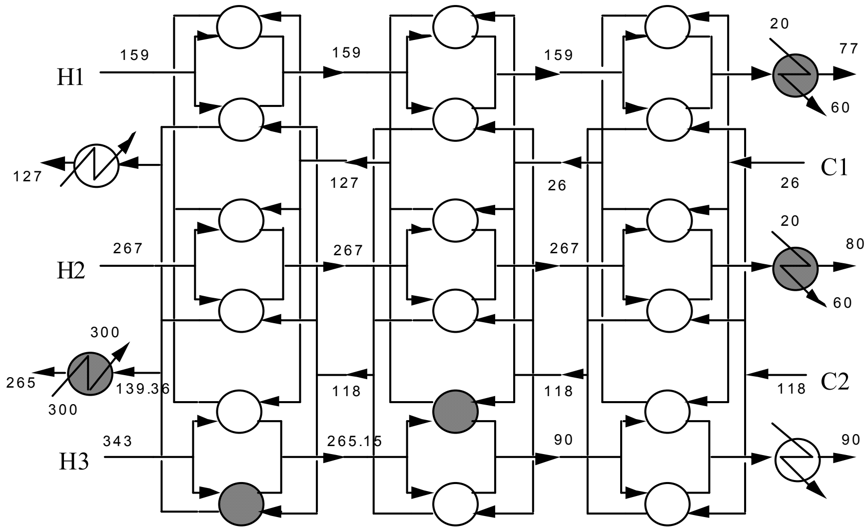
Table: Problem Data Example 2

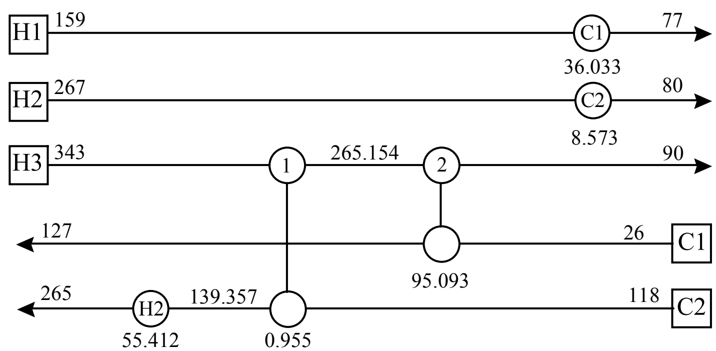
Stream	T_{in} (°C)	T_{out} (°C)	F ($kW\ C^{-1}$)	h ($kW\ m^{-2}\ C^{-1}$)	Cost ($\$\ kW^{-1}\ yr^{-1}$)
H1	159	77	2.285	0.10	-
H2	267	80	0.204	0.04	-
H3	343	90	0.538	0.50	-
C1	26	127	0.933	0.01	-
C2	118	265	1.961	0.50	-
S1	300	300	-	0.05	110
W1	20	60	-	0.20	10

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

Carnegie Mellon

Example 2



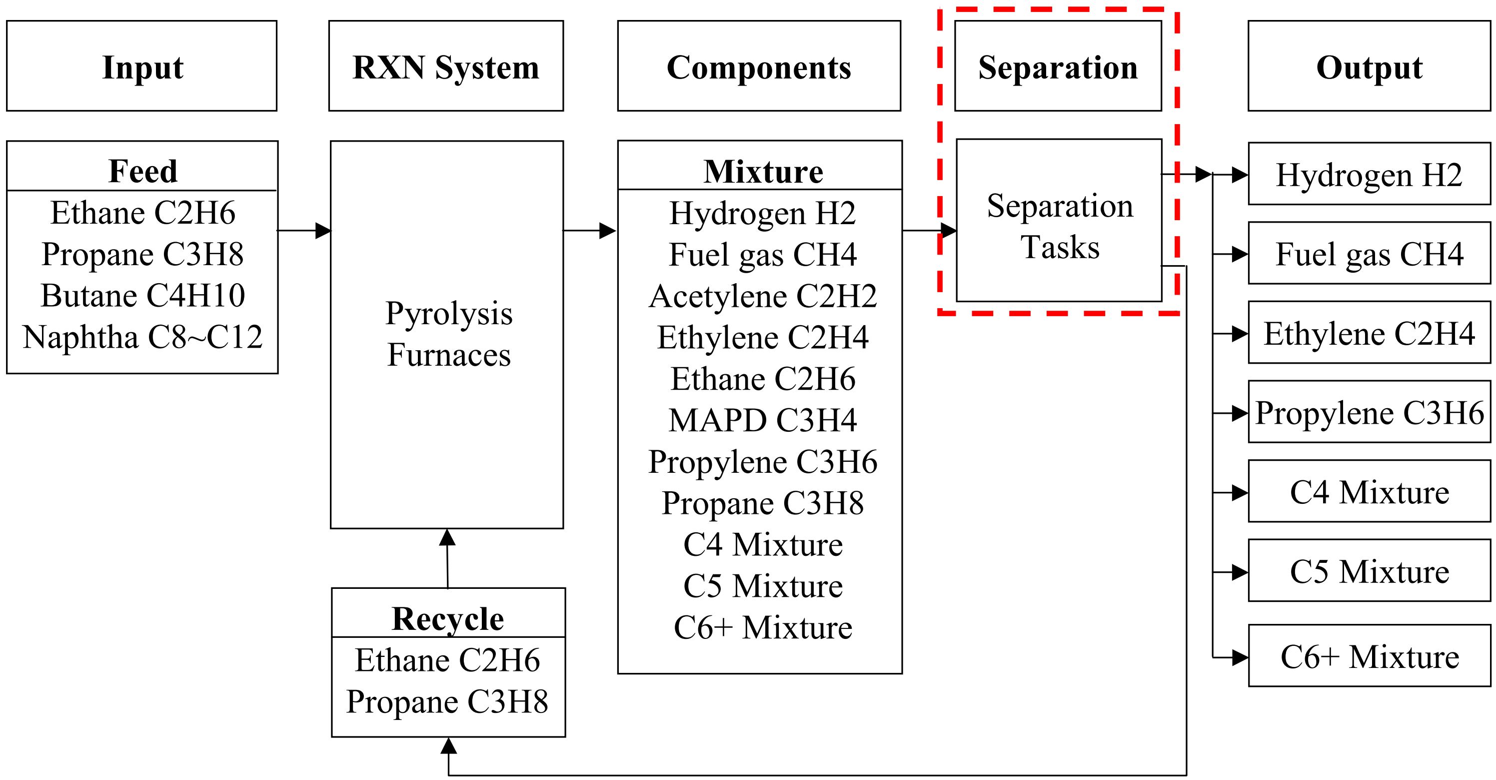


Exchanger	Heat Load (kW)	AMTD Area (m^2)	LMTD Area (m^2)
3-1-2	94.23	95.09	99.74
3-2-1	41.88	0.96	0.96
CU-1	187.37	36.03	36.94
CU-2	38.15	8.57	9.64
HU-2	246.39	55.41	65.47

Olefin Separation System *(BP)*

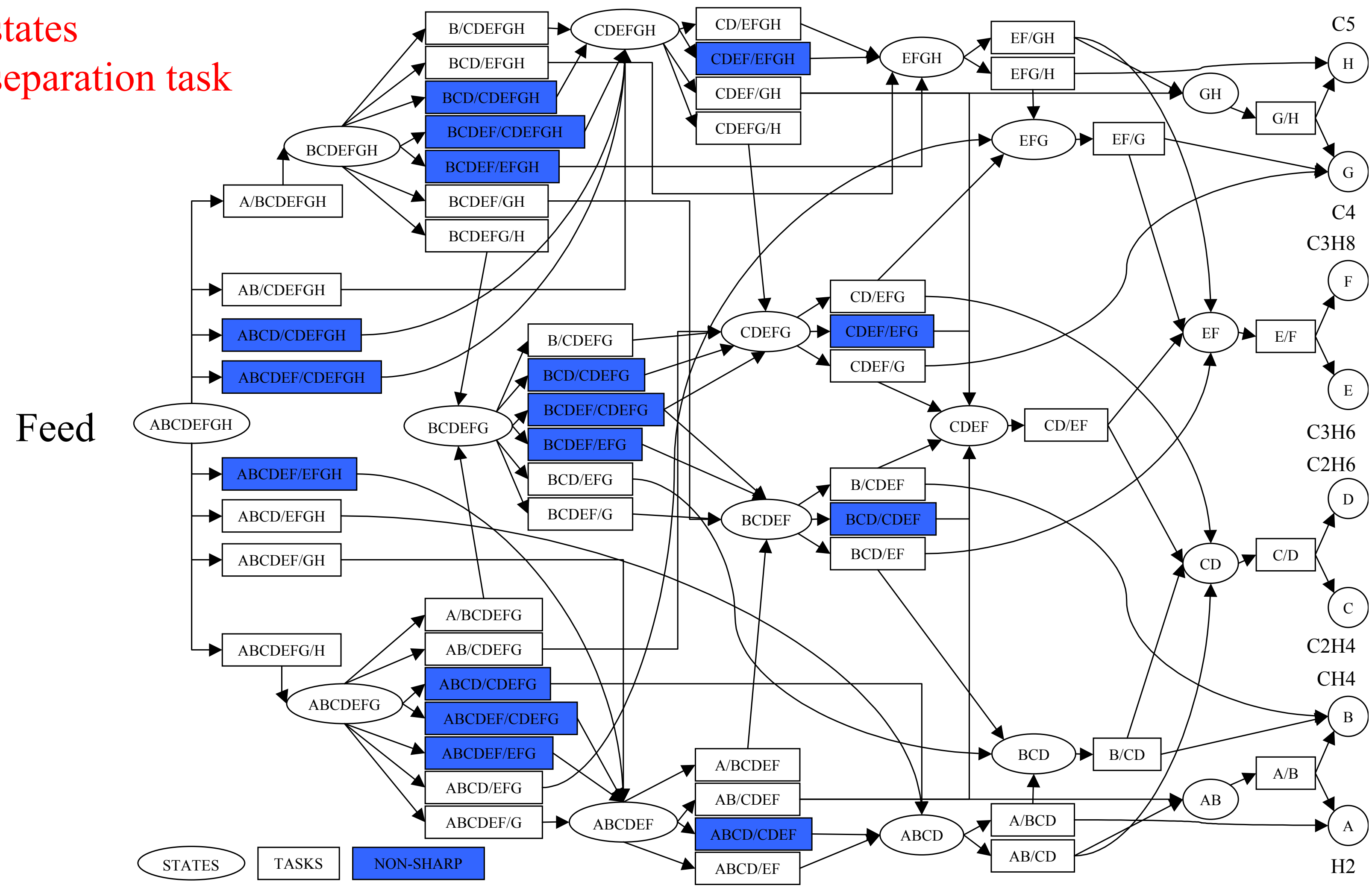
(Lee, Foral, Logsdon, Grossmann, 2003)

Goal: Synthesize optimal separation system



Process Superstructure

25 states
53 separation task



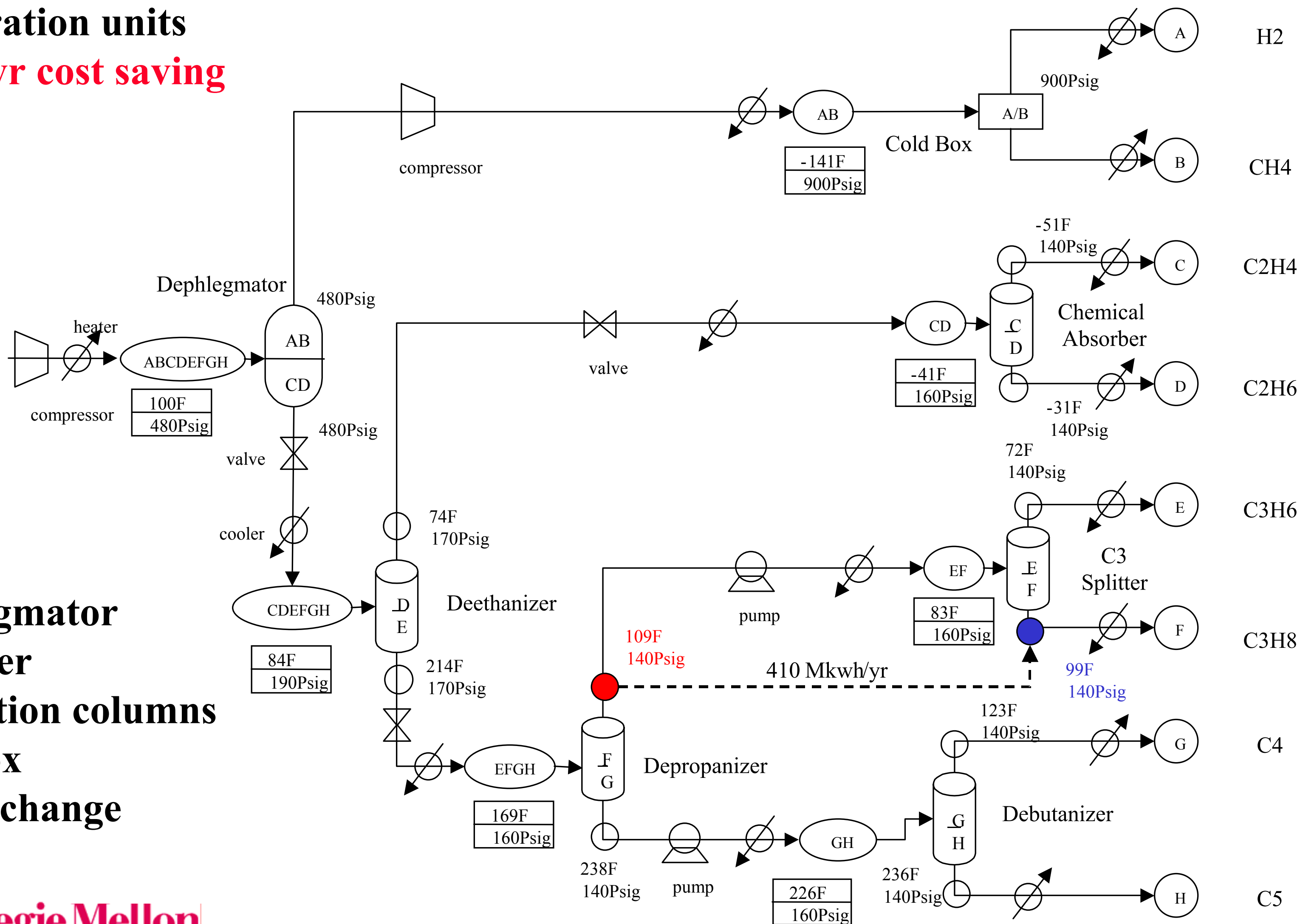
MINLP Model

- **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
20M$/yr cost saving

Total cost: 110.82 M$/yr



1 dephlegmator
1 absorber
4 distillation columns
1 cold box
1 heat exchange

Carnegie Mellon

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

1. Constraint Programming — *Van Hentenryck (1989)*
2. Generalized Disjunctive Programming — *Raman and Grossmann (1994)*
3. Mixed-Logic Linear Programming — *Hooker and Osorio (1999)*

Generalized Disjunctive Programming (GDP)

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

$$\min \; Z = \sum_k c_k + f(x) \qquad \textbf{Objective Function}$$

$$s.t. \quad r(x) \le 0 \qquad \textbf{Common Constraints}$$

$$\textbf{OR operator} \longrightarrow \bigvee_{j \in J_k} \begin{bmatrix} Y_{jk} \\ g_{jk}(x) \le 0 \\ c_k = \gamma_{jk} \end{bmatrix}, k \in K \qquad \begin{array}{c} \textbf{Disjunction} \\ \textbf{Constraints} \\ \textbf{Fixed Charges} \end{array}$$

$$\Omega(Y) = true \qquad \textbf{Logic Propositions}$$

$$x \in R^n, c_k \in R^1 \qquad \textbf{Continuous Variables}$$

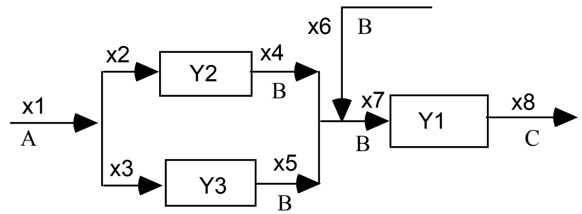
$$Y_{jk} \in \{true, \; false\} \qquad \textbf{Boolean Variables}$$

◆**Multiple Terms / Disjunctions**

Carnegie Mellon

Modeling Example

Process: A –> B –> C



MINLP

$$\min \quad Z = 3.5y_1+y_2+1.5y_3+x_4 +7x_6+1.2x_5+1.8x_1 -11x_8$$

s.t

$$x_1 - x_2 - x_3 = 0$$
$$x_7 - x_4 - x_5 - x_6 = 0$$
$$x_5 \leq 5$$
$$x_8 \leq 1$$

$$x_8 - 0.9x_7 = 0$$

$$x_4 = \ln(1+x_2)$$

$$x_5 = 1.2 \ln(1+x_3)$$

$$x_7 - 5y_1 \leq 0$$
$$x_2 - 5y_2 \leq 0$$
$$x_3 - 5y_3 \leq 0$$

$$x_i \in R \quad i = 1,\ldots,8$$
$$y_j \in \{0,1\} \quad j = 1,2,3$$

Generalized Disjunctive Programming (GDP)

- Raman and Grossmann (1994)

$$\min \; Z = \sum_k c_k + f(x)$$ Objective Function

$$s.t. \quad r(x) \le 0$$ Common Constraints

OR operator ⟶
$$\bigvee_{j \in J_k} \begin{bmatrix} Y_{jk} \\ g_{jk}(x) \le 0 \\ c_k = \gamma_{jk} \end{bmatrix}, k \in K$$
Disjunction
Constraints
Fixed Charges

$$\Omega(Y) = true$$ Logic Propositions

$$x \in R^n, c_k \in R^1$$ Continuous Variables

$$Y_{jk} \in \{ true, \; false \}$$ Boolean Variables

Relaxation?

Big-M MINLP (BM)

- **MINLP reformulation of GDP**

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

$$s.t. \quad r(x) \le 0$$

$$g_{jk}(x) \le M_{jk}(1 - y_{jk}) \ , j \in J_k, k \in K$$ **Big-M Parameter**

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

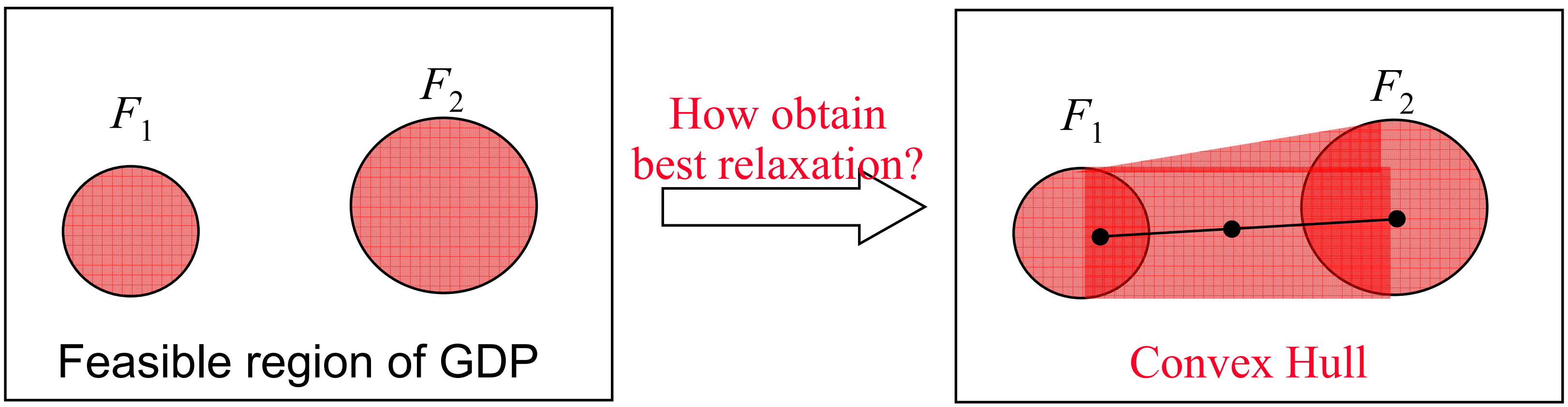
$$Ay \le a$$ **Logic constraints**

$$x \ge 0, y_{jk} \in \{0,1\}$$

NLP Relaxation $0 \le y_{jk} \le 1$

Nonlinear Convex Hull Relaxation

- **Disjunction** $(x \in F1) \vee (x \in F2)$



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

Carnegie Mellon

Convex Hull Formulation

- Consider **Disjunction** $k \in K$

$$\bigvee_{j \in J_k} \begin{bmatrix} Y_{jk} \\ g_{jk}(x) \le 0 \\ c = \gamma_{jk} \end{bmatrix}$$

- **Theorem**: Convex Hull of Disjunction k *(Lee, Grossmann, 2000)*
  - **Disaggregated variables** ν^j

$$\{(x,c) \mid x = \sum_{j \in J_k} v^{jk}, \quad c_k = \sum_{j \in J} \lambda_{jk} \gamma_{jk},$$

$$0 \le v^{jk} \le \lambda_{jk} U_{jk},\ j \in J_k$$

=> Convex Constraints

$$\sum_{j \in J_k} \lambda_{jk} = 1, \quad 0 < \lambda_{jk} \le 1,$$

$$\lambda_{jk} g_{jk}(v^{jk} / \lambda_{jk}) \le 0,\ j \in J_k \}$$

  - λ_j - weights for linear combination

- Generalization of Stubbs and Mehrotra (1999)

Remarks

1. $h(\nu, \lambda) = \lambda g(\nu / \lambda)$

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

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

3. **Implementation**

   $$\lambda_{jk} g_{jk}(\nu^{jk} / (\lambda_{jk} + \varepsilon)) \le 0$$

   ε **tolerance (e.g. 0.0001)**
   should be smaller than integer tolerance

4. **For linear constraints convex hull reduces to result by *Balas (1985)***

Note. For linear disjunctions $\underset{i \in D}{\vee}\left[\sum a_{ij}\, x_j \le b_i\right]$

above reduces to

$$x_j = \sum_i z_j^i \quad \mathrm{j} \in \mathrm{N}$$

$$\sum_{j \in N} a_{ij} z_j^i \le b_i y_i \quad \mathrm{i} \in \mathrm{D}$$

$$\sum_i \mathrm{y}_\mathrm{i} = 1$$

Balas (1985)

$$y_i \ge 0 \quad \mathrm{i} \in \mathrm{D}$$

Convex Relaxation Problem (CRP)

CRP:

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

$$s.t. \qquad r(x) \le 0$$

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

$$0 \le \nu^{jk} \le \lambda_{jk} U_{jk}, j \in J_k, k \in K$$

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

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

Convex Hull Formulation

$$A\lambda \le a$$

Logic constraints

$$x, \nu^{jk} \ge 0, \ 0 \le \lambda_{jk} \le 1, \ j \in J_k, k \in K$$

- **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):***

RBM:

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

$$s.t. \quad r(x) \le 0$$

$$g_{jk}(x) \le M_{jk}(1-y_{jk}) \quad j\in J_k, k\in K$$

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

$$Ay \le a$$

$$x \ge 0, \; 0 \le y_{jk} \le 1$$

Methods Generalized Disjunctive Programming

GDP

Key: relaxation

Logic based methods

Reformulation MINLP

Outer-Approximation
Generalized Benders
Extended Cutting Plane

Branch and bound

(Lee & Grossmann, 2000)

Decomposition

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

Convex-hull

Cutting plane

Big-M

(Sawaya & Grossmann, 2004)

Carnegie Mellon

MINLP Reformulation

Specify in CRP λ as 0-1 variables

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

$$s.t. \qquad r(x)\le 0$$

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

$$0 \le \nu^{jk} \le \lambda_{jk}U_{jk},\ j\in J_k,\ k \in K$$

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

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

$$A\lambda \le a$$

$$x,\ \nu^{jk} \ge 0,\ \boxed{\lambda_{jk} = 0,1}\ j\in J_k,\ k \in K$$

Cutting Planes for Linear Generalized Disjunctive Programming

GDP Model:

Sawaya, Grossmann (2004)

$$\min Z = \sum_{k \in K} c_k + h^T x$$ *Objective Function*

$$\text{s.t.} \qquad Bx \le b$$ *Common Constraints*

OR Operator

$$\bigvee_{j \in J_k} \begin{bmatrix} Y_{jk} \\ A_{jk} x \le a_{jk} \\ c_k = \gamma_{jk} \end{bmatrix} \quad k \in K$$ *Disjunctive Constraints*

$$\Omega(Y) = True$$ *Logic Constraints*

$$x \in R^n,\ Y_{jk} \in \{True, False\},\ c_k \in R$$

$$j \in J_k,\ k \in K$$

Boolean Variables

Reformulations as MILP

Big-M

$$\text{Min } Z = \sum_{k \in K} \sum_{j \in J_k} \gamma_{jk} \lambda_{jk} + h^T x$$

Big-M parameters

$$\begin{aligned}
s.t. \quad & Bx \le b \\
& A_{jk} x - a_{jk} \le M_{jk} (1-\lambda_{jk}) && j \in J_k,\, k \in K \\
& \sum_{j \in J_k} \lambda_{jk} = 1 && k \in K \\
& D\lambda \le d \\
& x \in R^n,\ \lambda_{jk} \in \{0,1\} && j \in J_k,\, k \in K
\end{aligned} \qquad \text{(BM)}$$

Convex Hull

$$\text{Min } Z = \sum_{k \in K} \sum_{j \in J_k} \gamma_{jk} \lambda_{jk} + h^T x$$

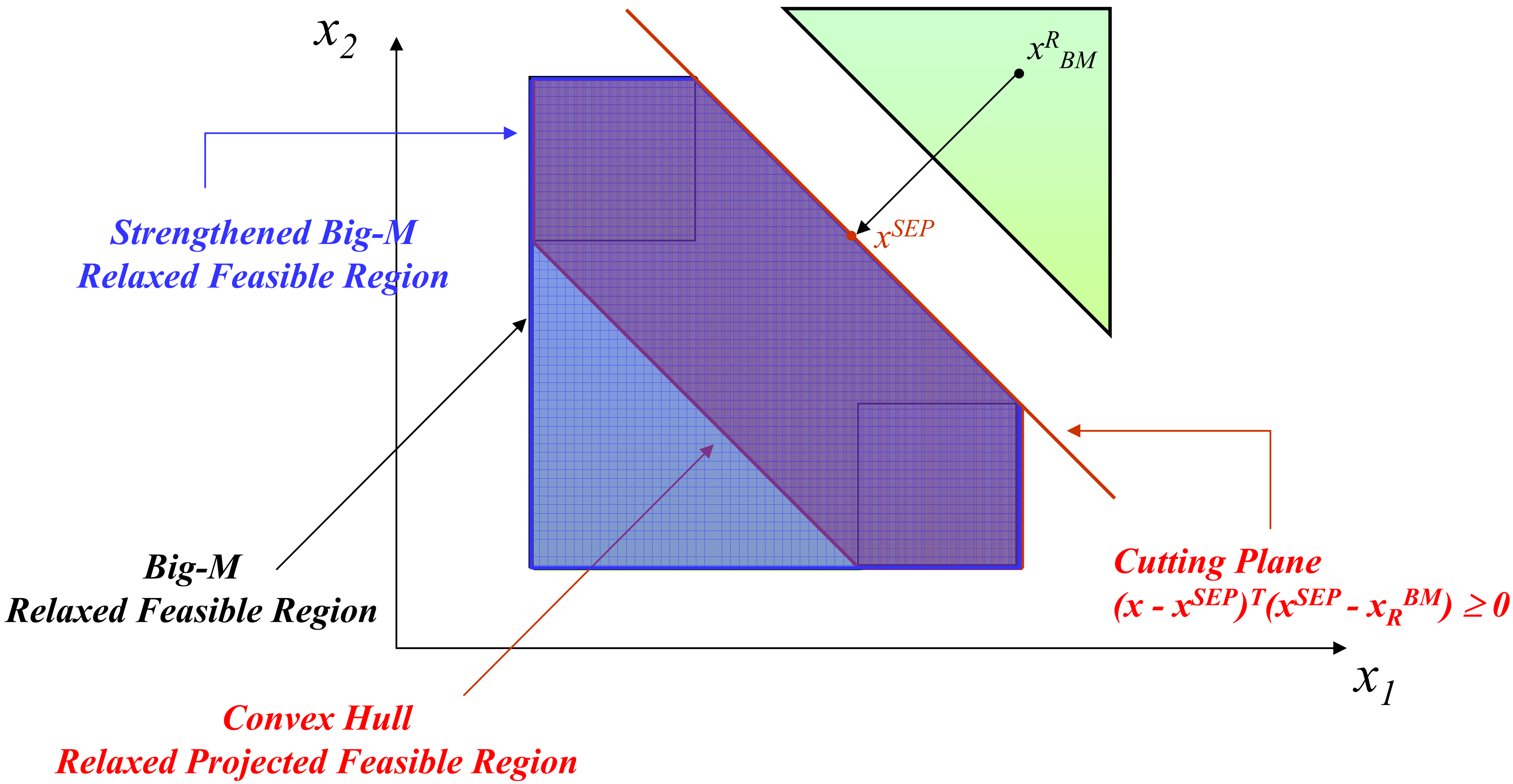
Disaggregated variables

$$\begin{aligned}
s.t. \quad & Bx \le b \\
& A_{jk} \nu^{jk} - a_{jk} \lambda_{jk} \le 0 && j \in J_k,\, k \in K \\
& x = \sum_{j \in J_k} \nu^{jk} && k \in K \\
& 0 \le \nu^{jk} \le \lambda_{jk} U_{jk} && j \in J_k,\, k \in K \\
& \sum_{j \in J_k} \lambda_{jk} = 1 && k \in K \\
& D\lambda \le d \\
& x \in R^n,\ \nu^{jk} \in R^n_+,\ \lambda_{jk} \in \{0,1\} && j \in J_k,\, k \in K
\end{aligned} \qquad \text{(CH)}$$

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) ***(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.

$$
\begin{array}{llr}
\textit{Min } Z = \Phi(x) & & \text{(SEP)} \\
\text{s.t.} \quad Bx \le b & & \\
\quad A_{jk}\, \nu^{jk} - a_{jk}\, \lambda_{jk} \le 0 & & j \in J_k,\ k \in K \\
\quad x = \sum_{j \in J_k} \nu^{jk} & & k \in K \\
\quad 0 \le \nu^{jk} \le \lambda_{jk}\, U_{jk} & & j \in J_k,\ k \in K \\
\quad \sum_{j \in J_k} \lambda_{jk} = 1 & & k \in K \\
\quad D\lambda \le d & & \\
\quad x \in R^n,\ \nu^{jk} \in R^n_{+},\ 0 \le \lambda_{jk} \le 1 & & j \in J_k,\ k \in K
\end{array}
$$

Note: $\Phi(x)$ *can be represented by either the Euclidean norm* $(?\ x - x_R^{BM}?\ )$ *(NLP) or the Infinity norm* $(\max_i |x_i - x_{iR}^{BM}|)$ *(LP).*

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 ξ such that

$$\xi^T (z^{SEP} - z^{BM}) \geq 0$$

is a valid linear inequality, where ξ 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,

$$\xi \equiv (\mu^+ - \mu^-)$$

$$\begin{array}{llll} Min\ u & & & \text{Lagrange Multipliers} \\ s.t & u \geq z_i - z_i^{BM} & i \in I & \leftarrow \mu^+ \\ & u \geq -z_i + z_i^{BM} & i \in I & \leftarrow \mu^- \\ & \text{Feasible region of (SEP)} & & \end{array}$$

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

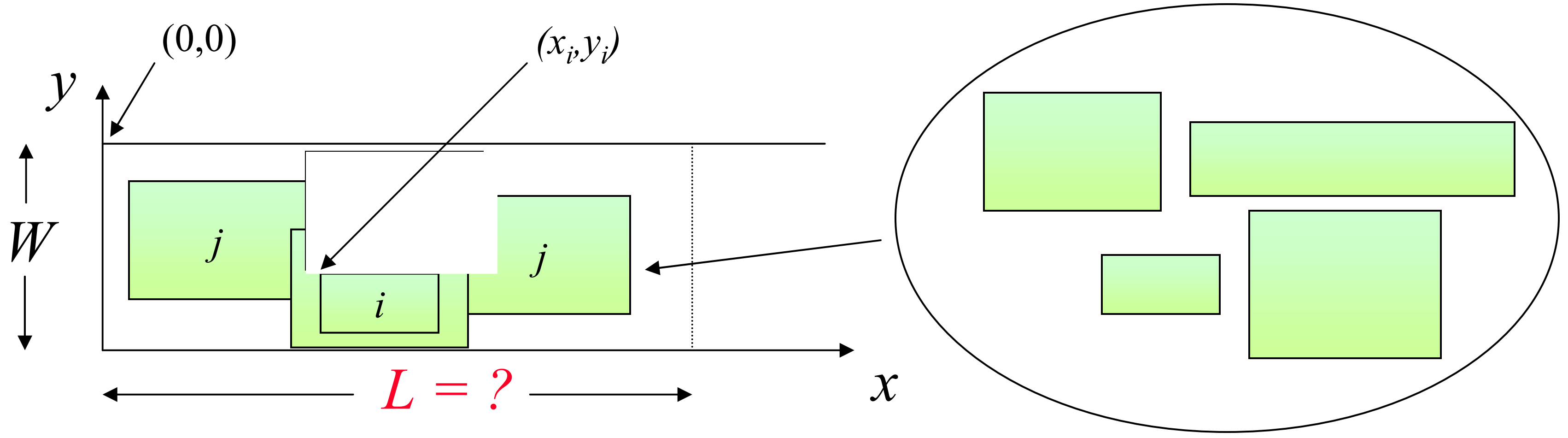
$$\xi \equiv (\mu^+ - \mu^-)$$

$$\begin{array}{llll} Min\ \Sigma u_i & & & \text{Lagrange Multipliers} \\ s.t & u_i \geq z_i - z_i^{BM} & i \in I & \leftarrow \mu^+ \\ & u_i \geq -z_i + z_i^{BM} & i \in I & \leftarrow \mu^- \\ & \text{Feasible region of (SEP)} & & \end{array}$$

Strip-packing Problem

Problem statement: *Hifi M. (1998)*

- 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.



Set of small rectangles

GDP Model For Strip-packing Problem

$$\mathit{Min}\ Z = L \qquad \text{(SP-GDP)}$$

$$\text{s.t.} \quad L \ge x_i + l_i \qquad i \in N$$

$$\begin{bmatrix} Y^1_{ij} \\ x_i + l_i \le x_j \end{bmatrix} \vee \begin{bmatrix} Y^2_{ij} \\ x_j + l_j \le x_i \end{bmatrix} \vee \begin{bmatrix} Y^3_{ij} \\ y_i - h_i \ge y_j \end{bmatrix} \vee \begin{bmatrix} Y^4_{ij} \\ y_j - h_j \ge y_i \end{bmatrix} \qquad i,j \in N,\ i < j$$

$$0 \le x_i \le U_i - l_i \qquad i \in N$$

$$h_i \le y_i \le W \qquad i \in N$$

$$x_i,\ y_i \in R \qquad i \in N$$

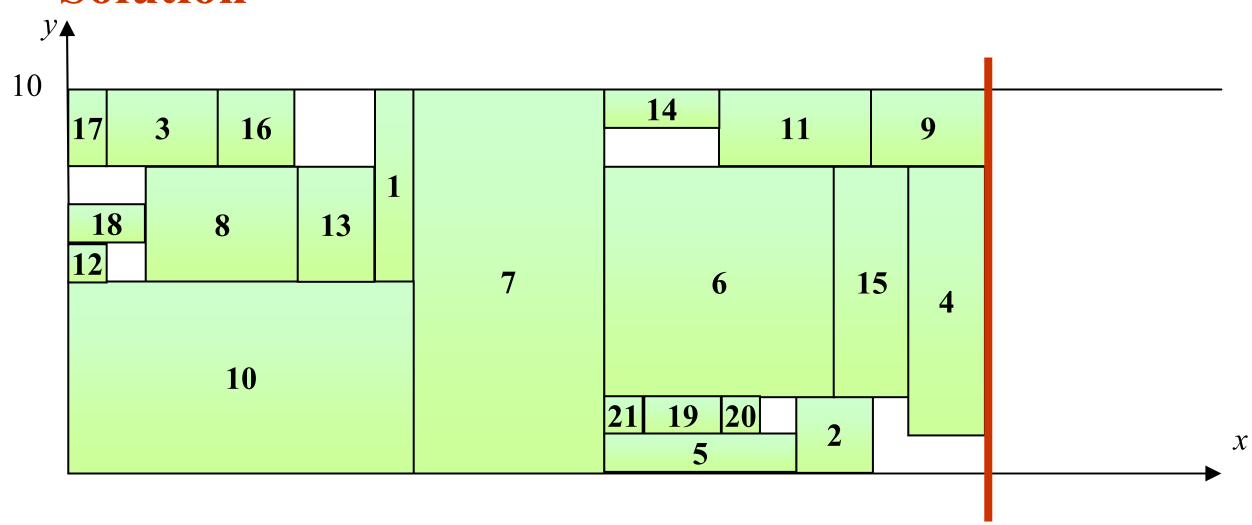
$$Y^1_{ij},\ Y^2_{ij},\ Y^3_{ij},\ Y^4_{ij} \in \{True, False\} \qquad i,j \in N,\ i < j$$

21-rectangle Strip-packing Problem

Problem Size

	Total number of constraints	Total number of variables	Number of discrete variables
Convex Hull	5272	4244	840
Big-M	1072	884	840

Solution



Optimal Length: 24

Numerical Results

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

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

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

Results also for retrofit, scheduling problems

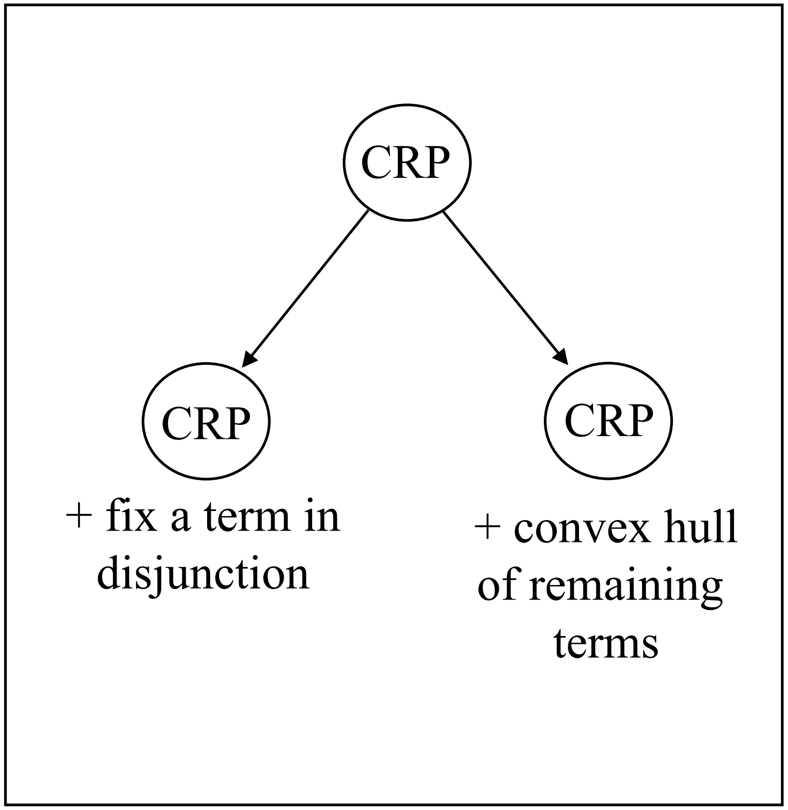
A Branch and Bound Algorithm for GDP

- Tree Search
  - NLP subproblem at each node
- Solve CRP of GDP
  - lower bound
- Branching Rule
  - Set the largest λ_j as 1 or 0
  - Dichotomy rule
- Logic inference

  CNF unit resolution (Raman & Grosmann, 1993)
- Depth first search

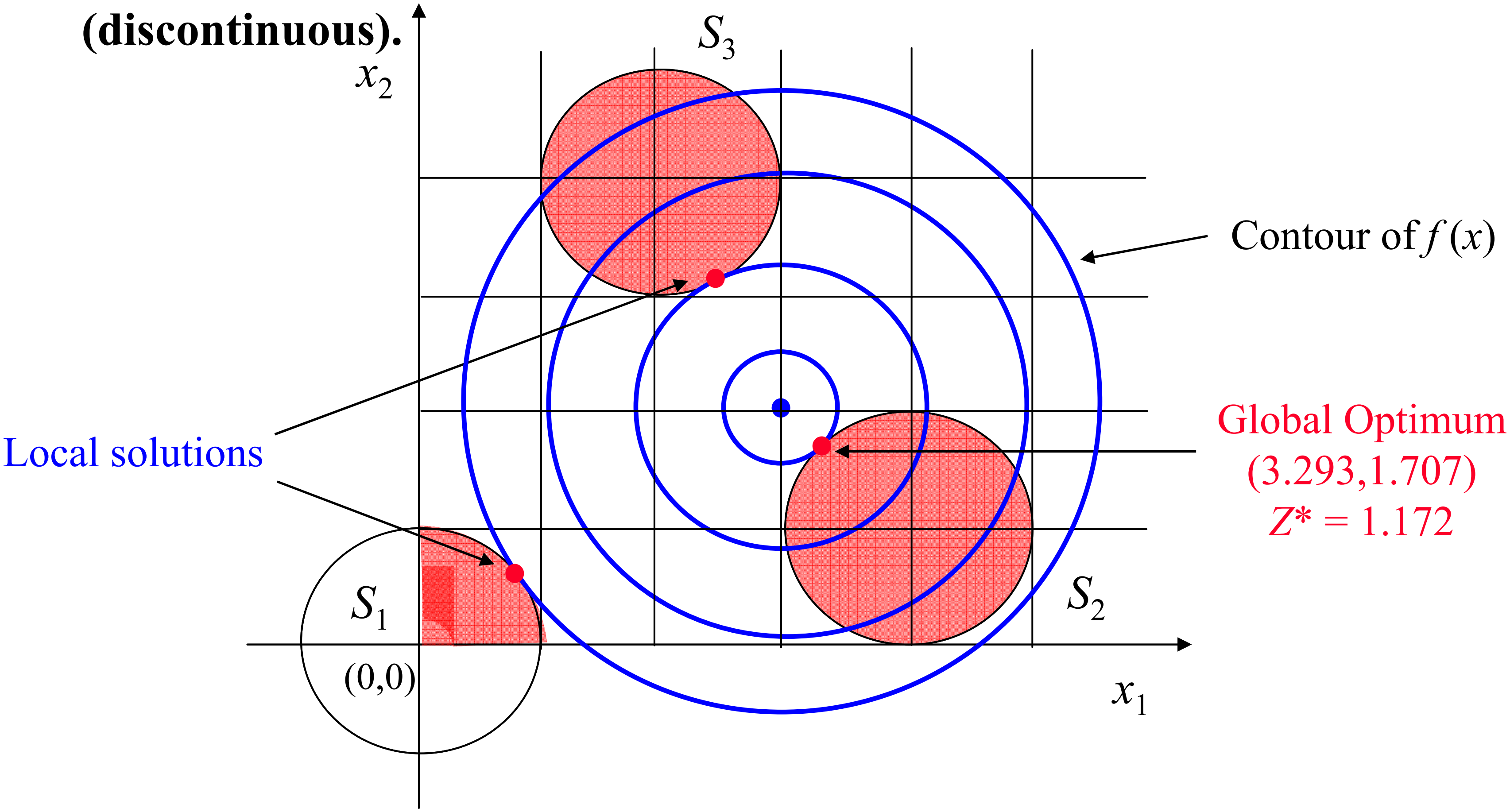
  When all the terms are fixed

  upper bound
- Repeat Branching until $Z^L > Z^U$.

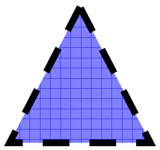


GDP Example

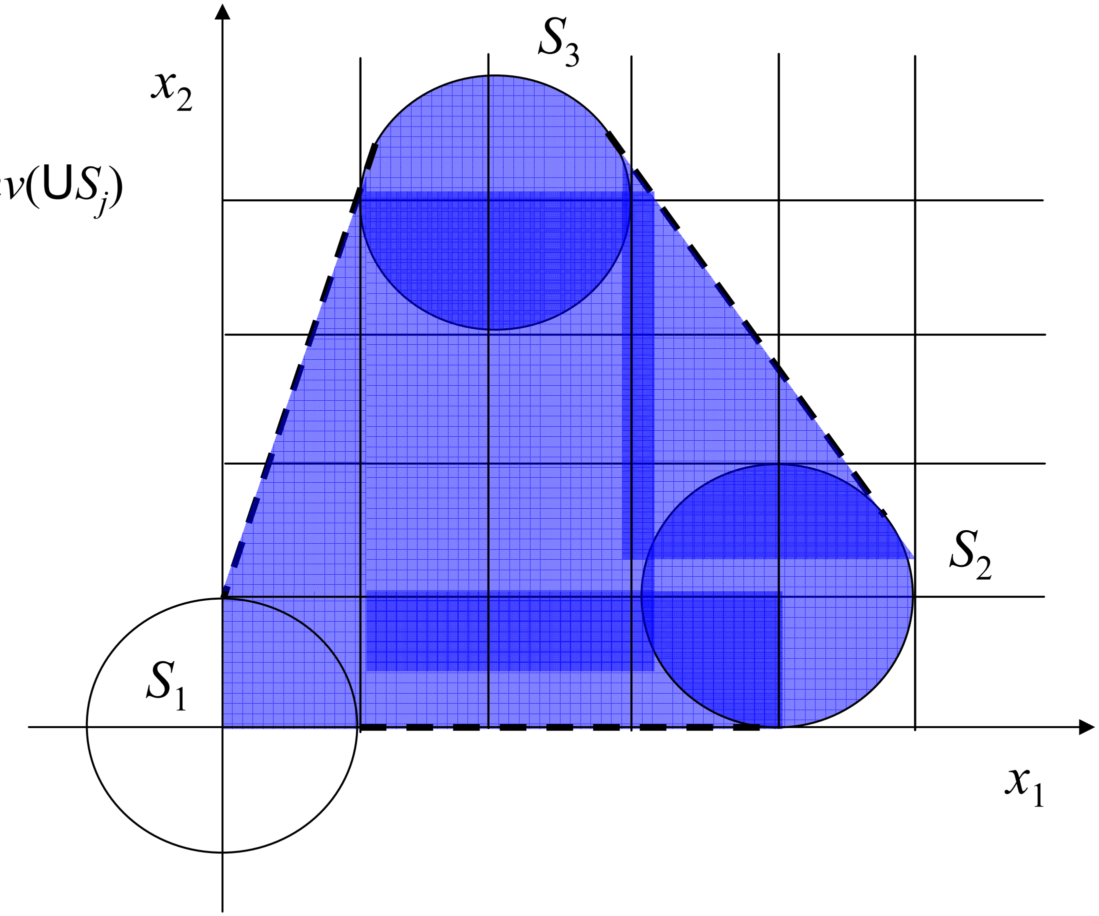
- **Find x ≥ 0, $(x \in S_1) \vee (x \in S_2) \vee (x \in S_3)$ to minimize $Z = (x_1 - 3)^2 + (x_2 - 2)^2 + c$**
  - **Objective Function = continuous function + fixed charge (discontinuous).**



Example : convex hull



Convex hull = $conv(\cup S_j)$



Carnegie Mellon

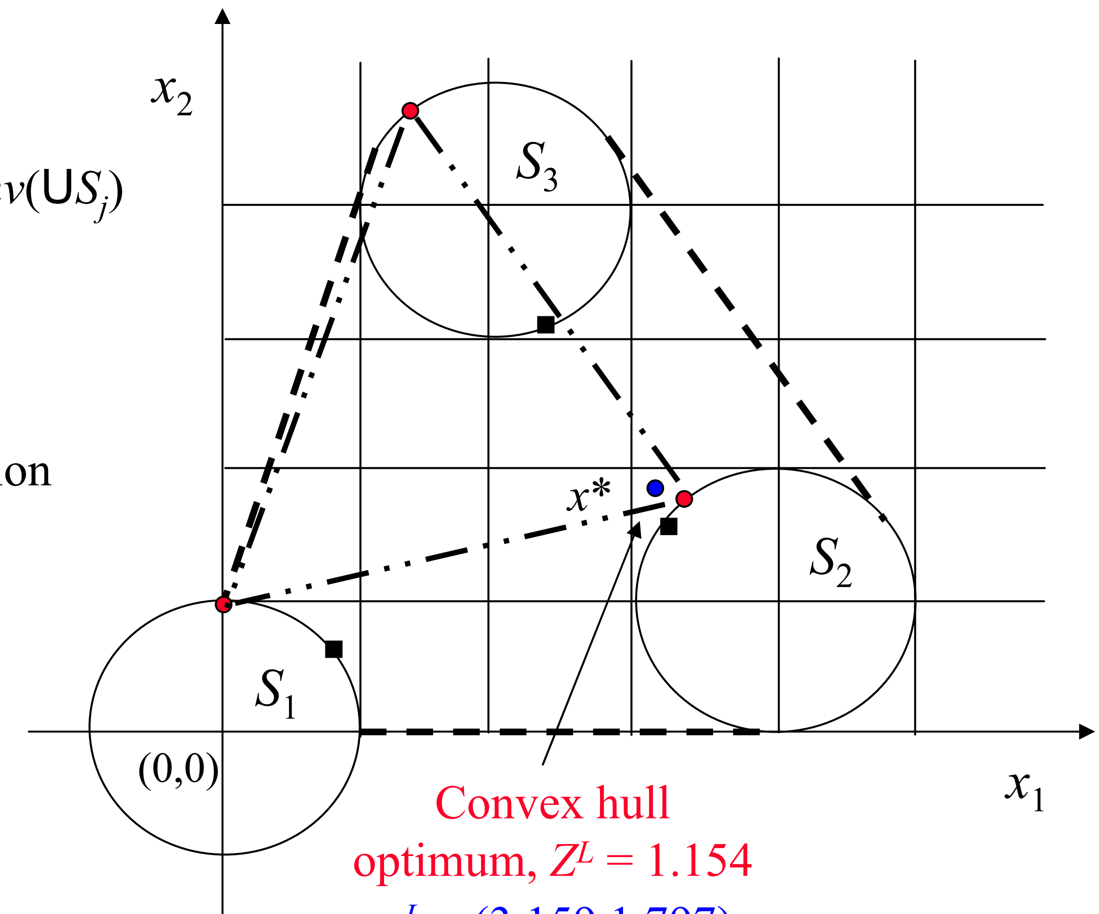
Example: CRP solution

Convex hull = $conv(\cup S_j)$

Convex combination of z_j

$z_j = v^j/\lambda_j$

Local solution point



Convex hull optimum, $Z^L = 1.154$

$x^L = (3.159, 1.797)$

Infeasible to GDP

Weight

$\lambda_1 = 0.016$

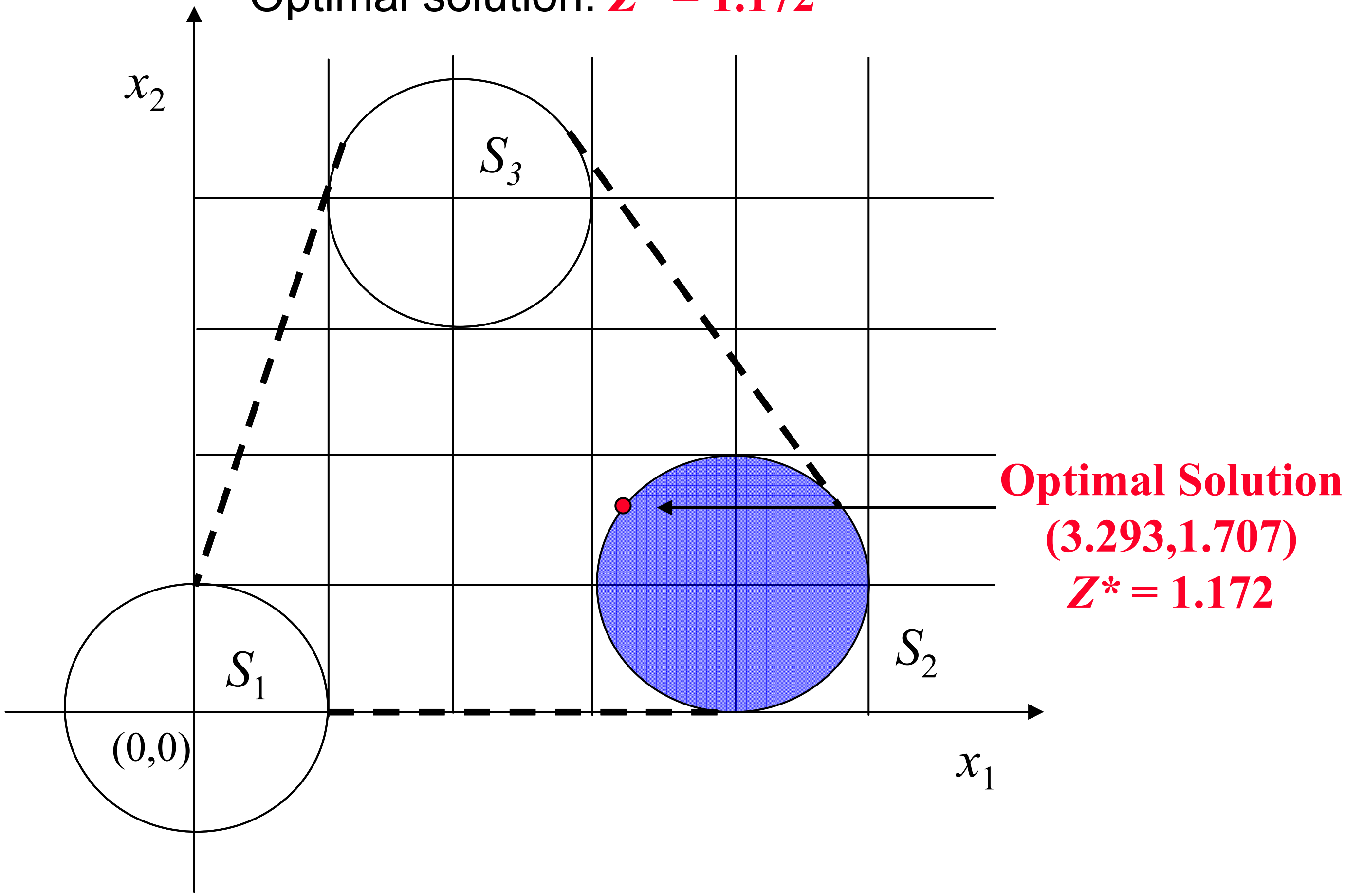
$\lambda_2 = $ **0.955**

$\lambda_3 = 0.029$

Carnegie Mellon

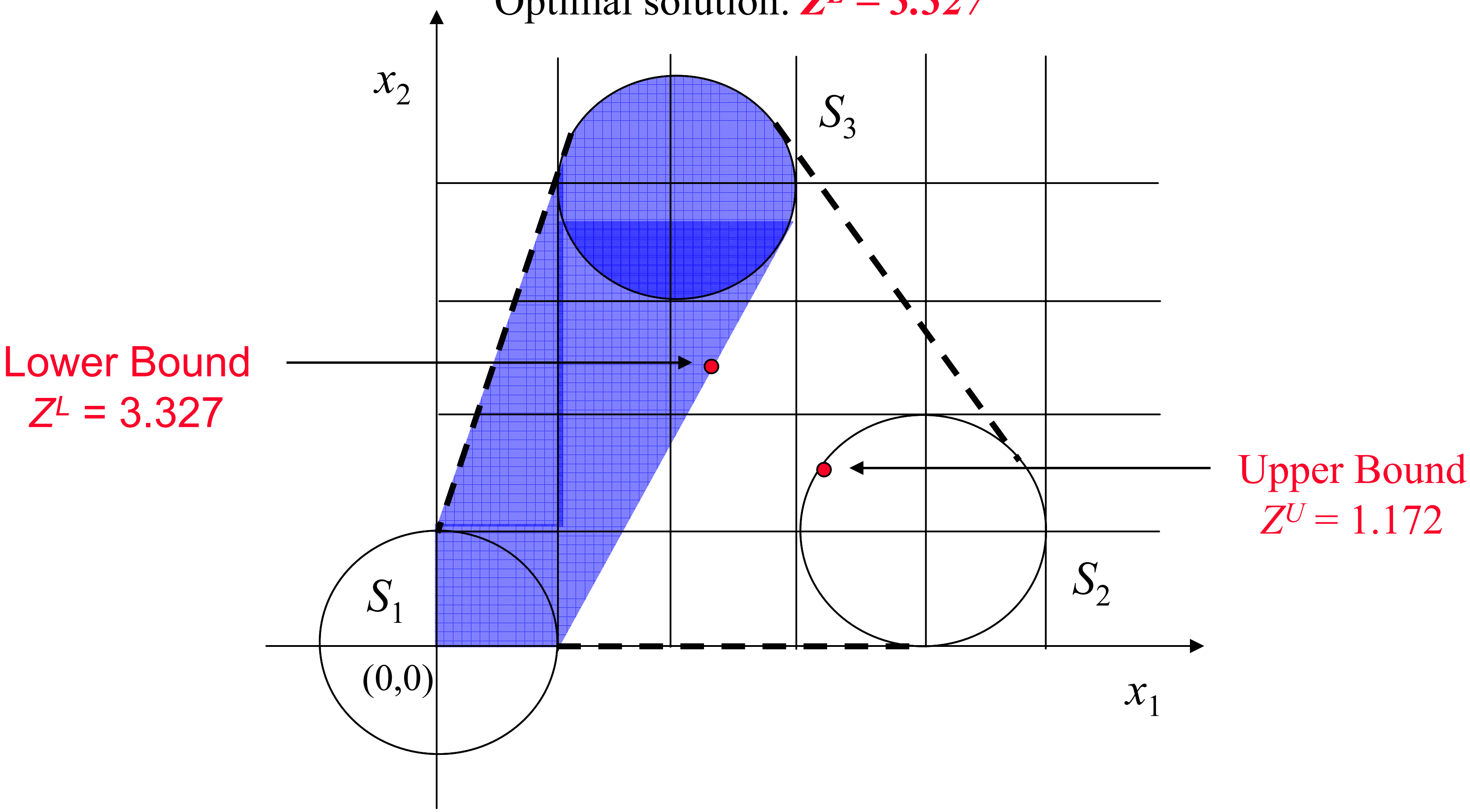
Example : branch and bound

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



Example : branch and bound

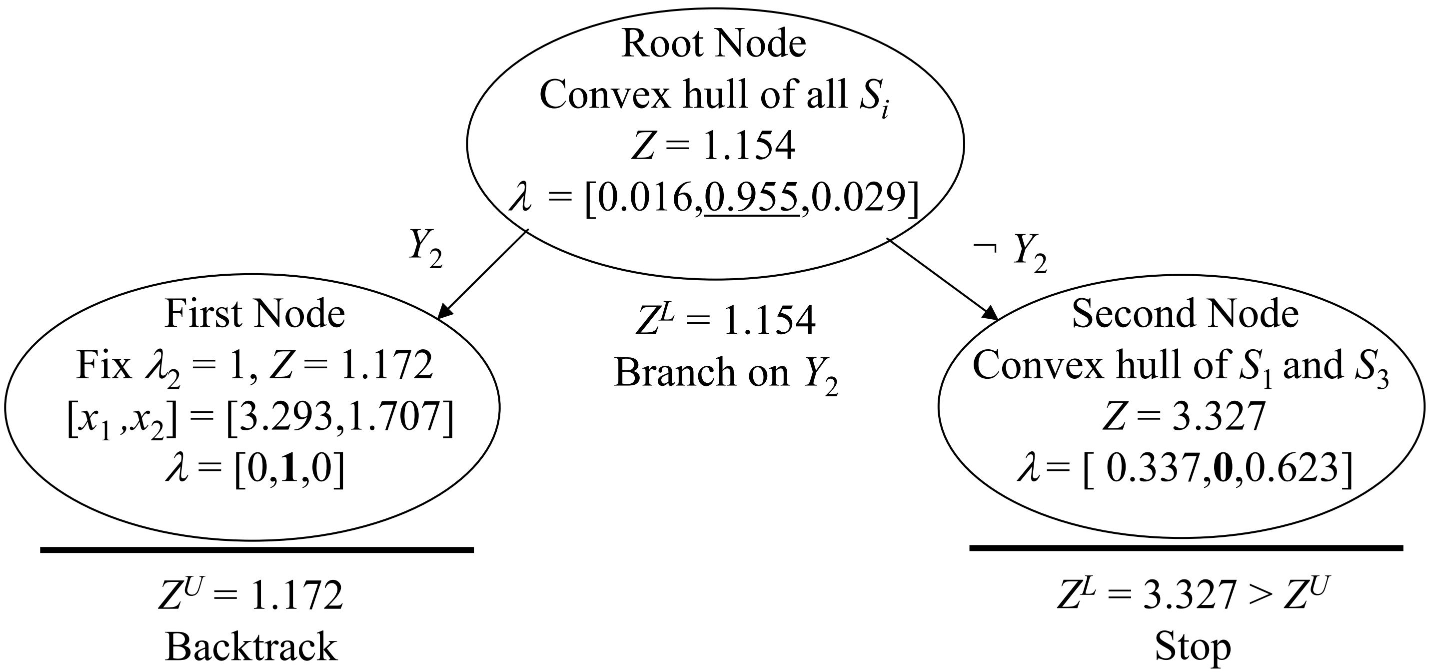
Second Node: ***conv(S_1 U S_3)***
Optimal solution: ***Z^L = 3.327***



Carnegie Mellon

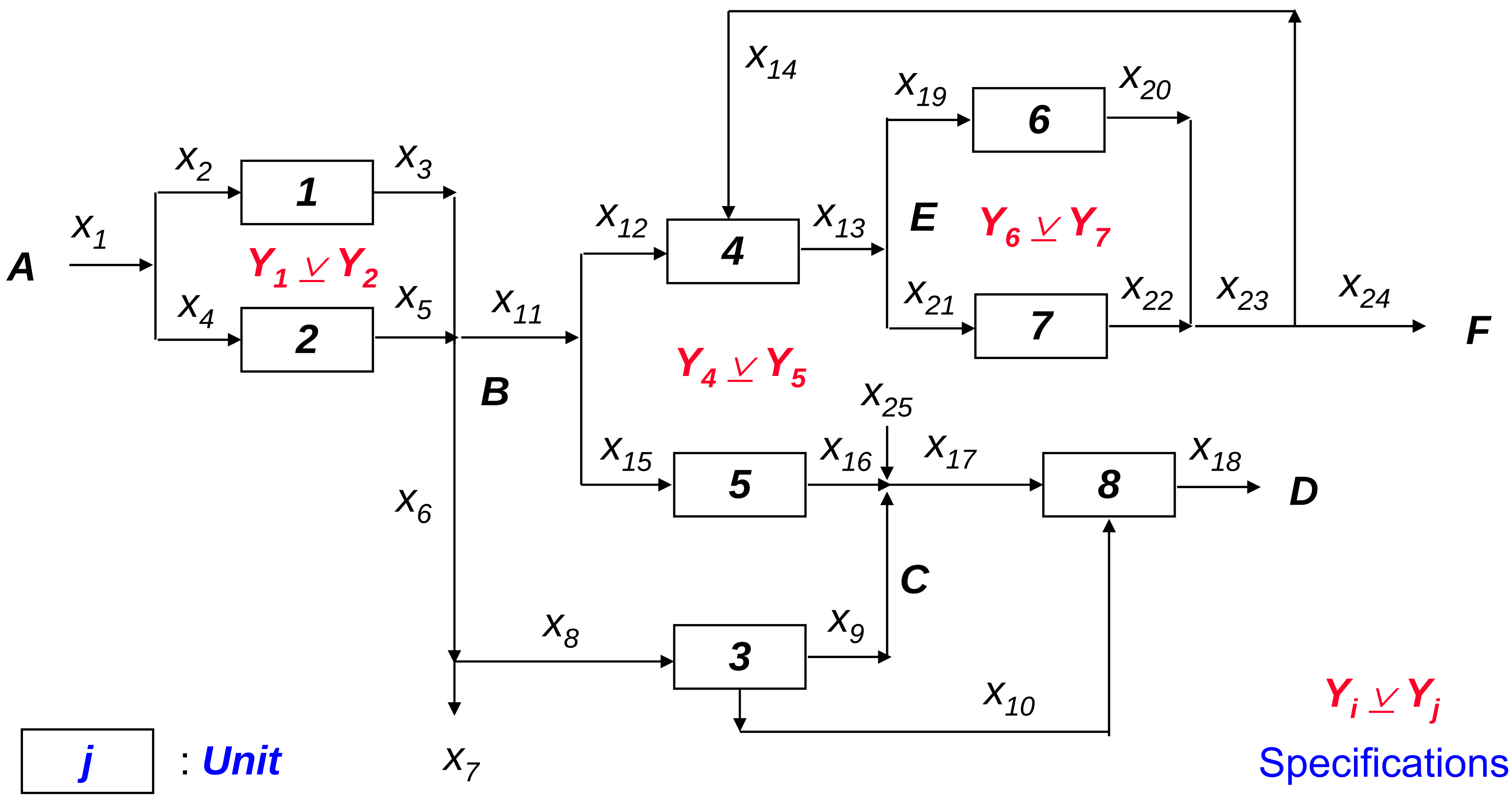
Example: Search Tree

- **Branching Rule:** λ_j - the weight of disaggregated variable
  - Fix Y_j as true: fix λ_j as 1.



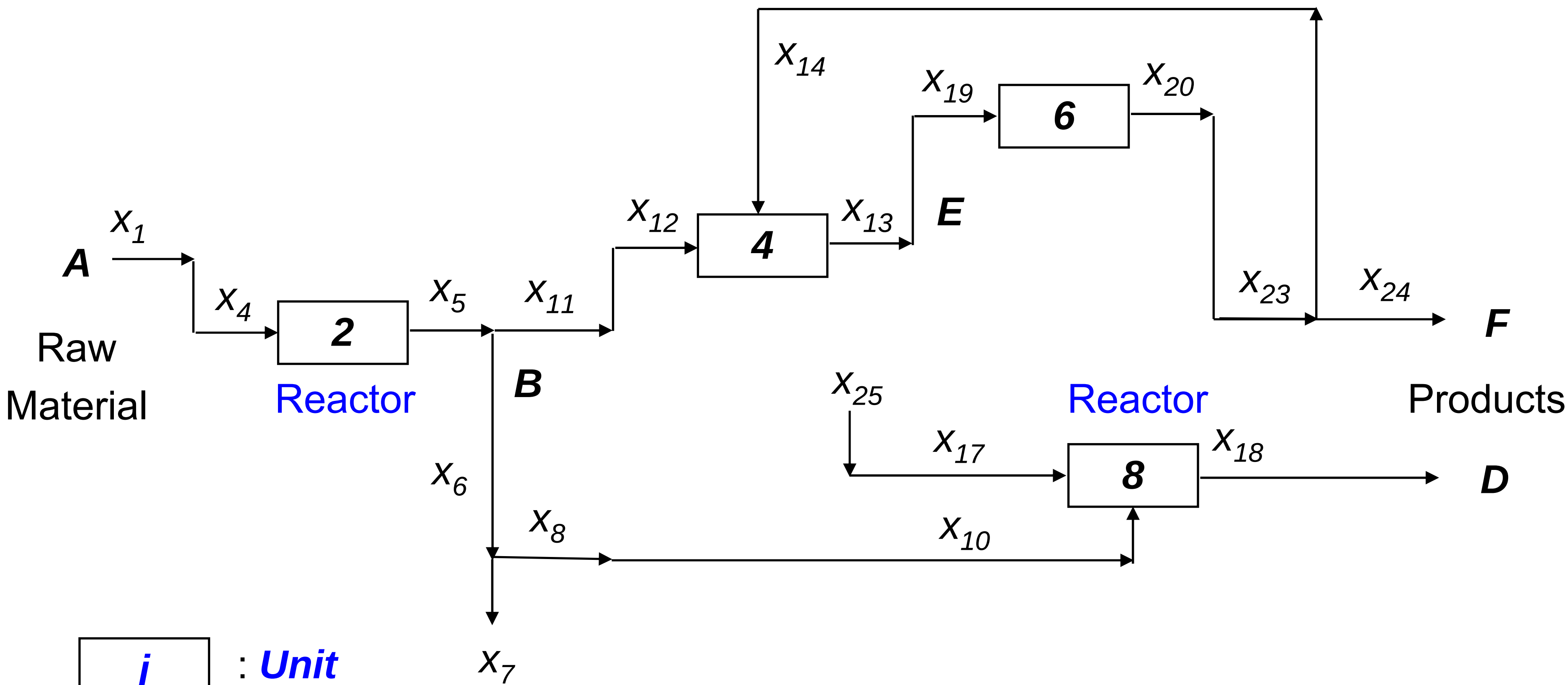
Process Network with Fixed Charges

- *Türkay and Grossmann (1997)*
  - **Superstructure of the process**



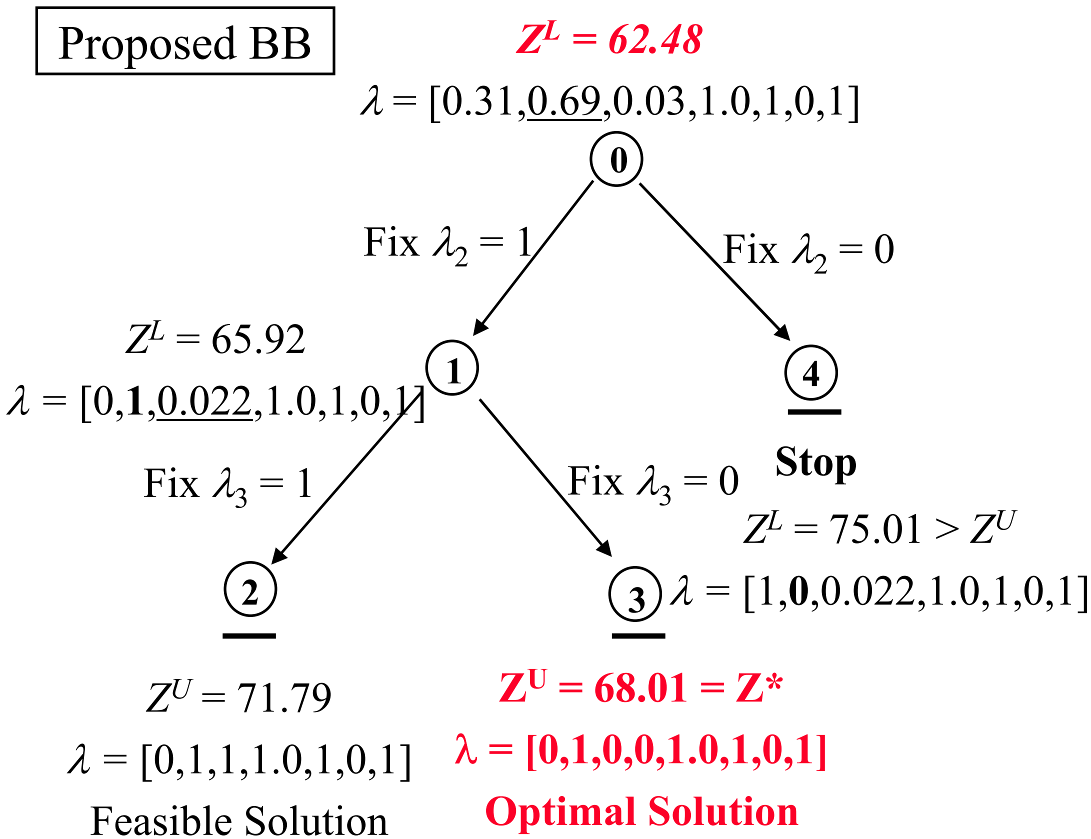
Optimal solution

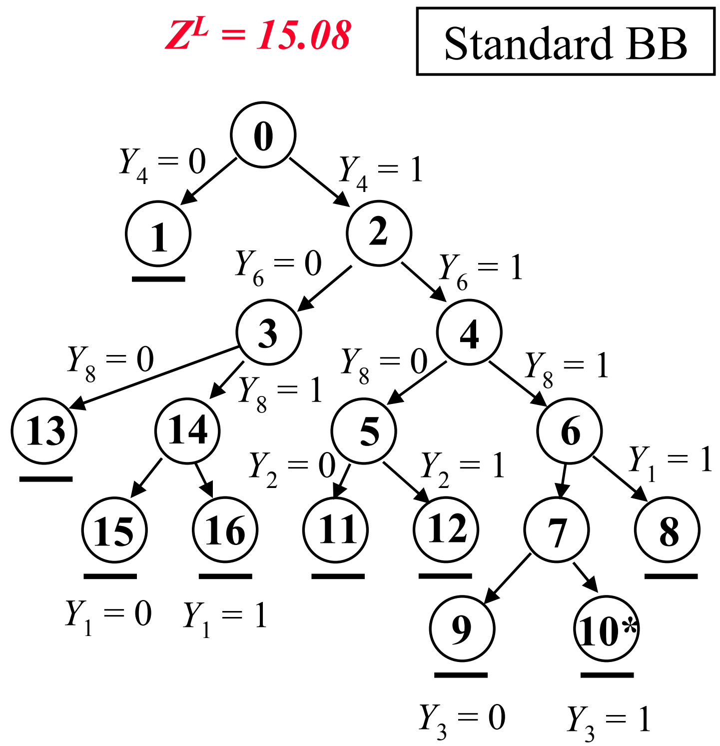
- **Minimum Cost: $ 68.01M/year**



Carnegie Mellon

Proposed BB Method





- **5 nodes vs. 17 nodes of Standard BB (lower bound = 15.08)**
- **CPU time : 2.578 vs. 3.383 of Standard BB (300MHz Pentium II PC)**

Carnegie Mellon

Logic-based Outer Approximation

Turkay, Grossmann (1997)

NLP Subproblem:

$$\min Z = \sum_{k \in SD} c_k + f(x)$$

$$s.t. \quad g(x) \le 0$$

$$\left.\begin{array}{l} h_{ik}(x) \le 0 \\ c_k = \gamma_{ik} \end{array}\right\} for\, Y_{\hat{i}k} = true \;\; \hat{i} \in D_k, k \in SD \qquad \text{(NLPD)}$$

$$\left.\begin{array}{l} B^i x = 0 \\ c_k = 0 \end{array}\right\} for\, Y_{ik} = false \;\; i \in D_k, i \ne \hat{i}, k \in SD$$

$$x \in R^n, \;\; c_i \in R^m,$$

Redundant constraints eliminated

MILP Master Problem:

$$Min \quad Z = \sum_k c_k + \alpha$$

$$s.t. \quad \left.\begin{array}{l} \alpha \ge f(x^l) + \nabla f(x^l)^T (x - x^l) \\ g(x^l) + \nabla g(x^l)^T (x - x^l) \le 0 \end{array}\right\} \; l = 1, \ldots, L \qquad \text{(MGDP)}$$

$$\bigvee_{i \in D_k} \begin{bmatrix} Y_{ik} \\ h_{ik}(x^l) + \nabla h_{ik}(x^l)^T (x - x^l) \le 0 \\ l \in L_{ik} \\ c_k = \gamma_{ik} \end{bmatrix} \quad k \in SD$$

$$\Omega(Y) = True$$

$$\alpha \in R, \;\; x \in R^n, \; c \in R^m, \; Y \in \{true, false\}^m$$

Carnegie Mellon

Proceed as in OA. Requires initialization NLPs

LogMIP

Aldo Vecchietti, INGAR

Part of GAMS Modeling System

-Disjunctions specified with IF Then ELSE statements

```
DISJUNCTION D1(I,K,J);
D1(I,K,J)
    with (L(I,K,J)) IS
IF Y(I,K,J) THEN
    NOCLASH1(I,K,J);
ELSE
    NOCLASH2(I,K,J);
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/

Small Example

```
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;
```

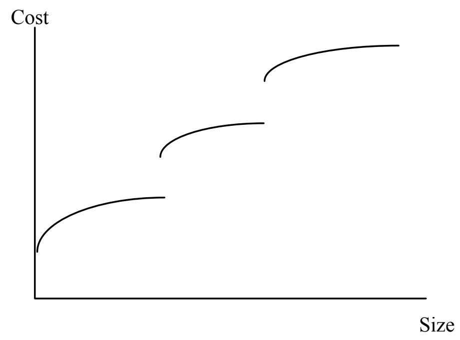
Structural Optimization of Vinyl Chloride Plant

Major options:

Direct Chlorination vs. Oxychlorination
Air vs. Oxygen
Pressure Pyrolisis
Separation sequence

Optimization with discontinuous cost models:

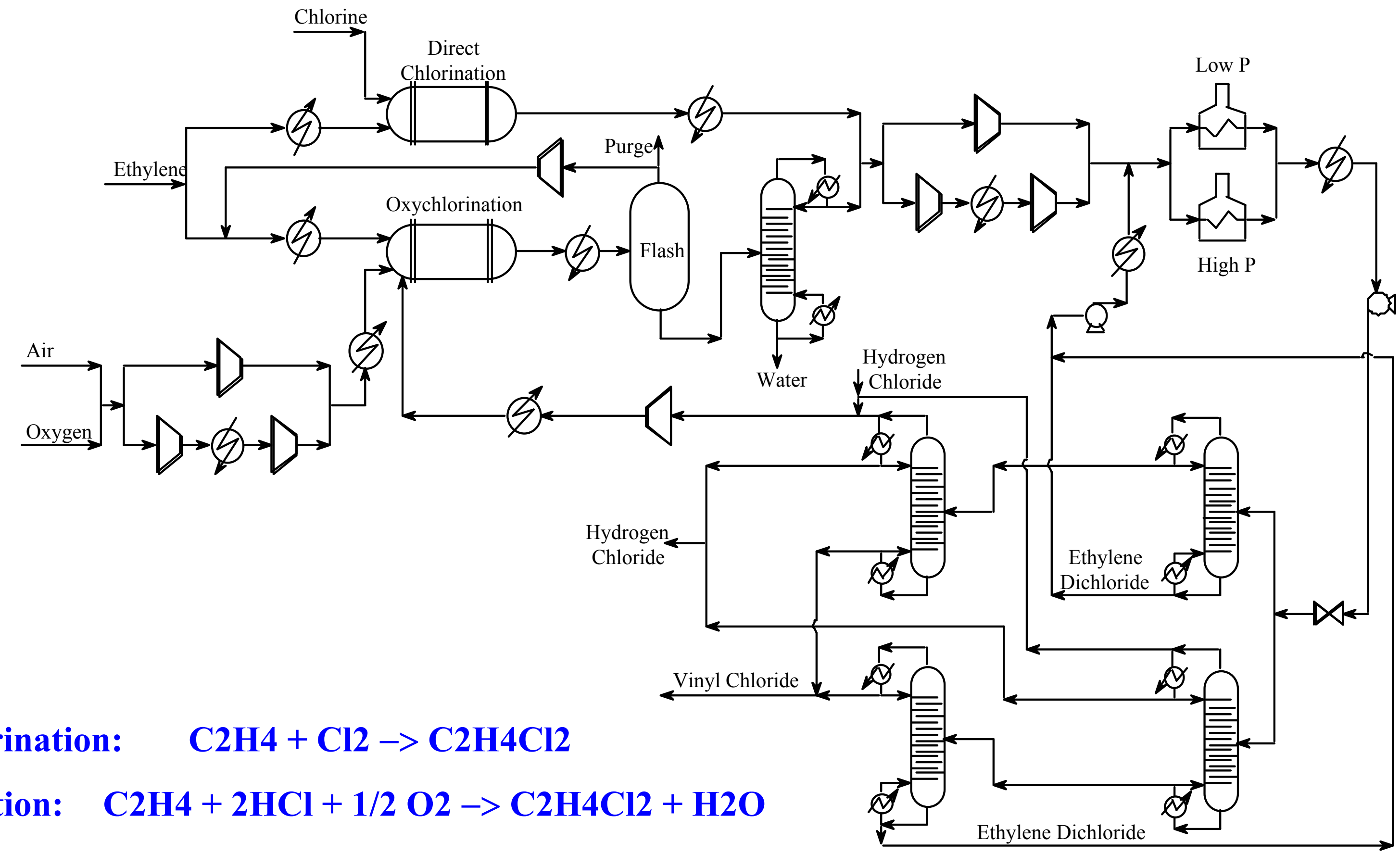
- *Multiple size regions*
- *Pressure, temperature factors*



Process Flowsheet Synthesis

Superstructure Vinyl Chloride Monomer

(Turkay & Grossmann, 1997)

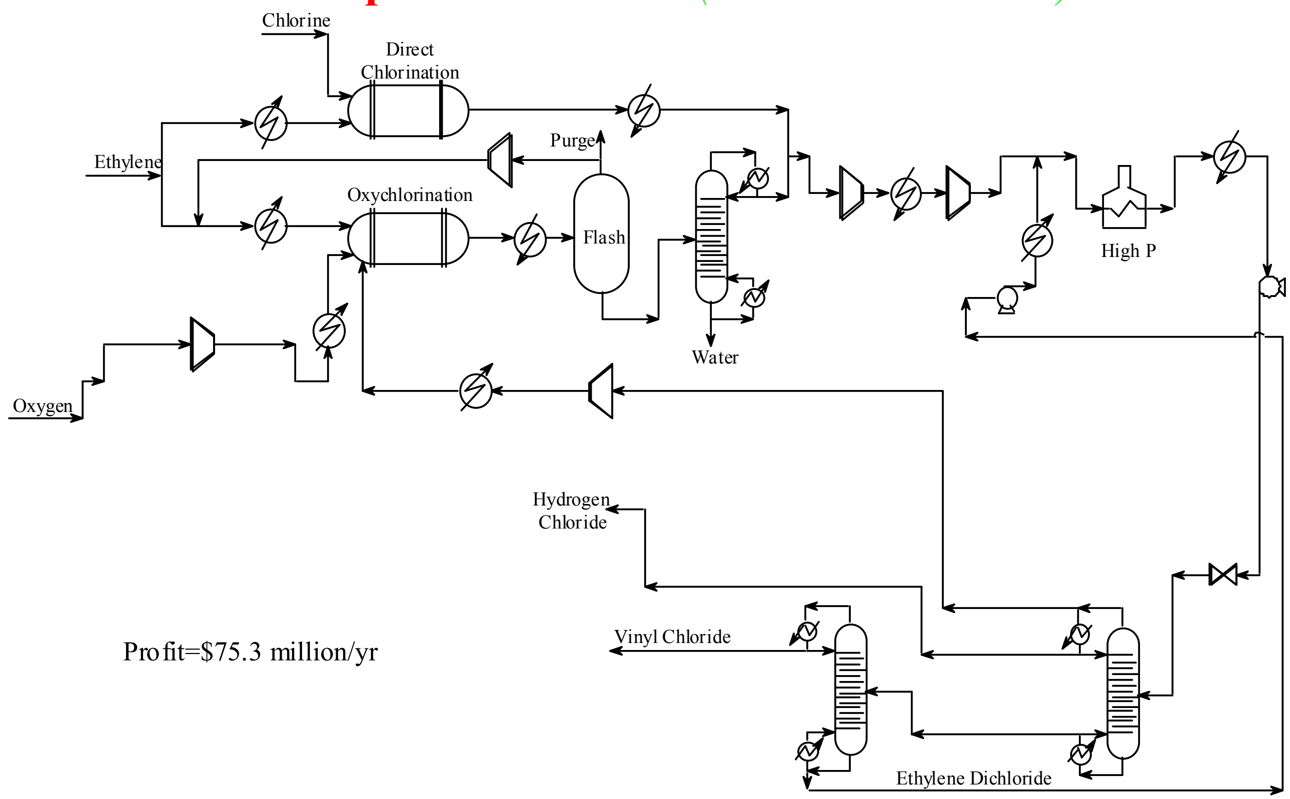


Direct chlorination: **C2H4 + Cl2 –> C2H4Cl2**

Oxychlorination: **C2H4 + 2HCl + 1/2 O2 –> C2H4Cl2 + H2O**

Pyrolisis: **H4Cl2 –> C2H3Cl + HCl**

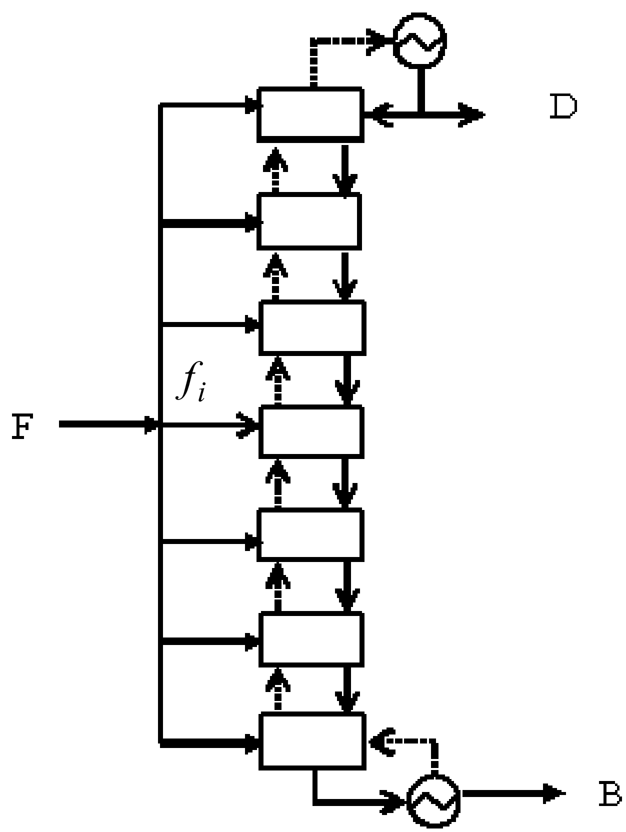
Optimal Solution *(CPU-time: 3.8min)*



Profit=$75.3 million/yr

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

Optimal Feedtray Location



Sargent & Gaminibandara (1976)

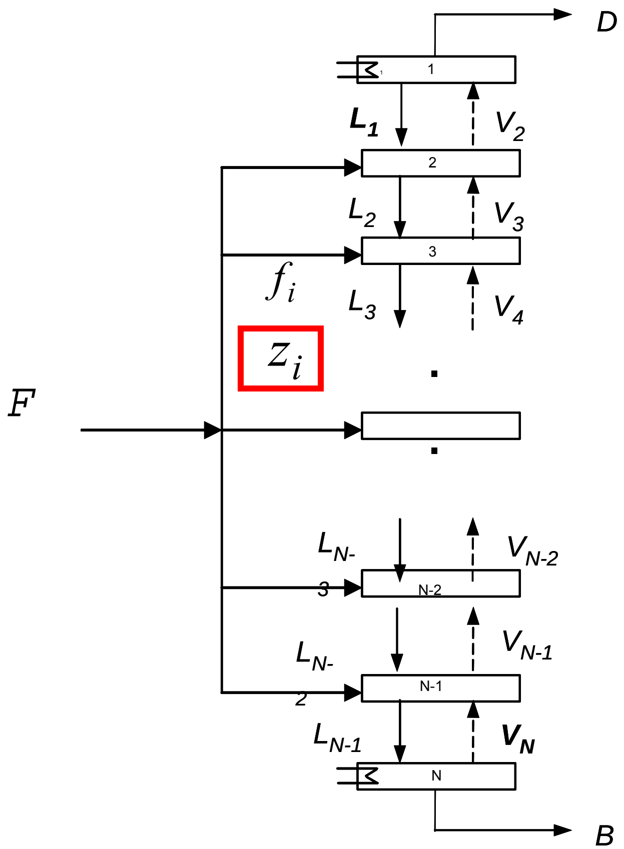
NLP Formulation

$$\begin{aligned} & Min\ cost \\ st\ \ & MESH\ eqtns \\ & \sum_{i \in LOC} f_i = F \end{aligned}$$

***NLP** VMP: Variable-Metric Projection*

Optimal Feedtray Location (Cont)

Viswanathan & Grossmann (1990)



MINLP Formulation

$$Min\ cost$$

$$st\ \ MESH\ eqtns$$

$$\sum_{i \in LOC} z_i = 1$$

$$\sum_{i \in LOC} f_i = F$$

$$f_i - F z_i \leq 0 \quad i \in LOC$$

$$z_i = 0,1 \quad i \in LOC$$

MINLP DICOPT: AP-Outer Approximation-ER

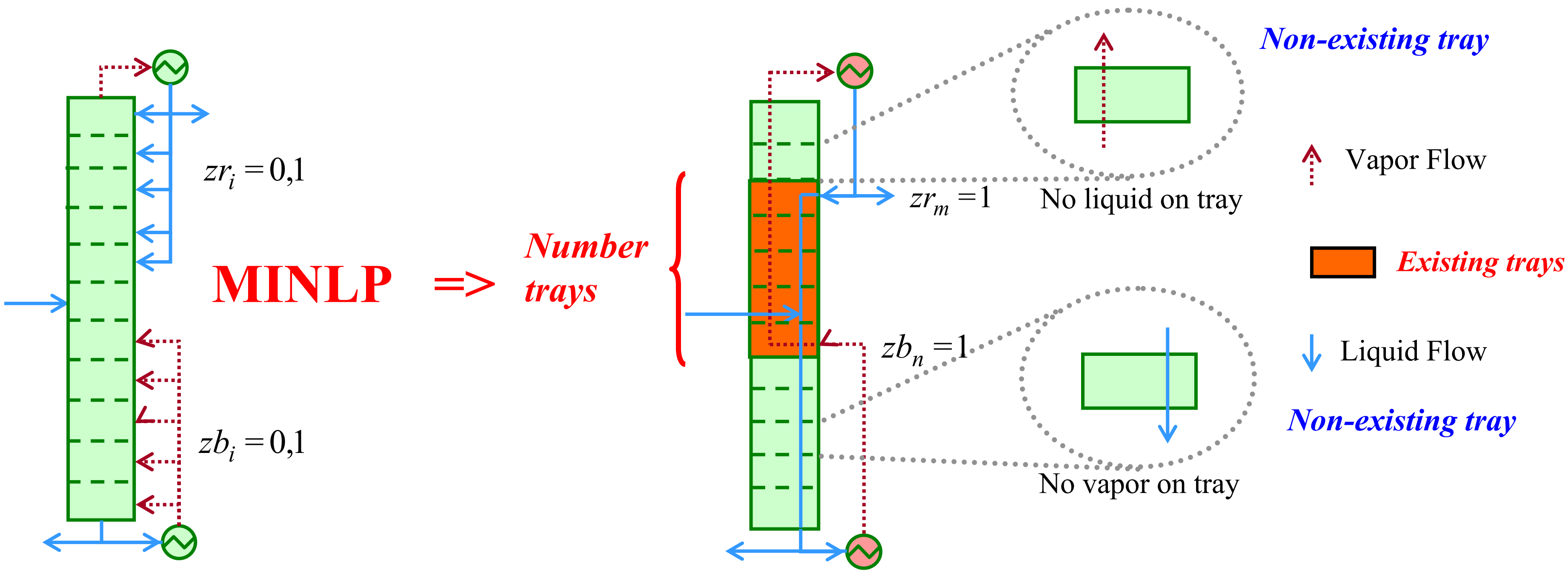
Remark: MINLP solves as relaxed NLP!

Feed tray composition tends to match composition of feed

Carnegie Mellon

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.

Disjunctive Programming Model

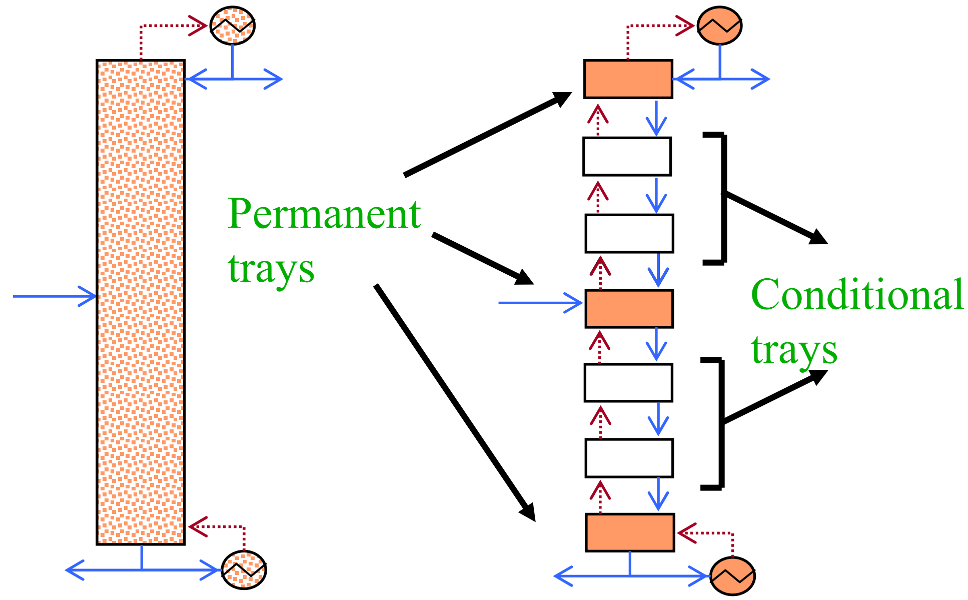
Yeomans & Grossmann (2000)

Permanent trays:

Feed, reboiler, condenser

Conditional trays:

Intermediate trays might be selected or not.

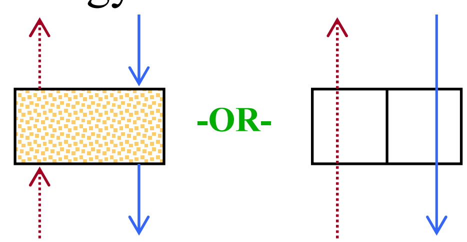


Trays not allowed to "disappear" from column:

VLE mass transfer if selected.

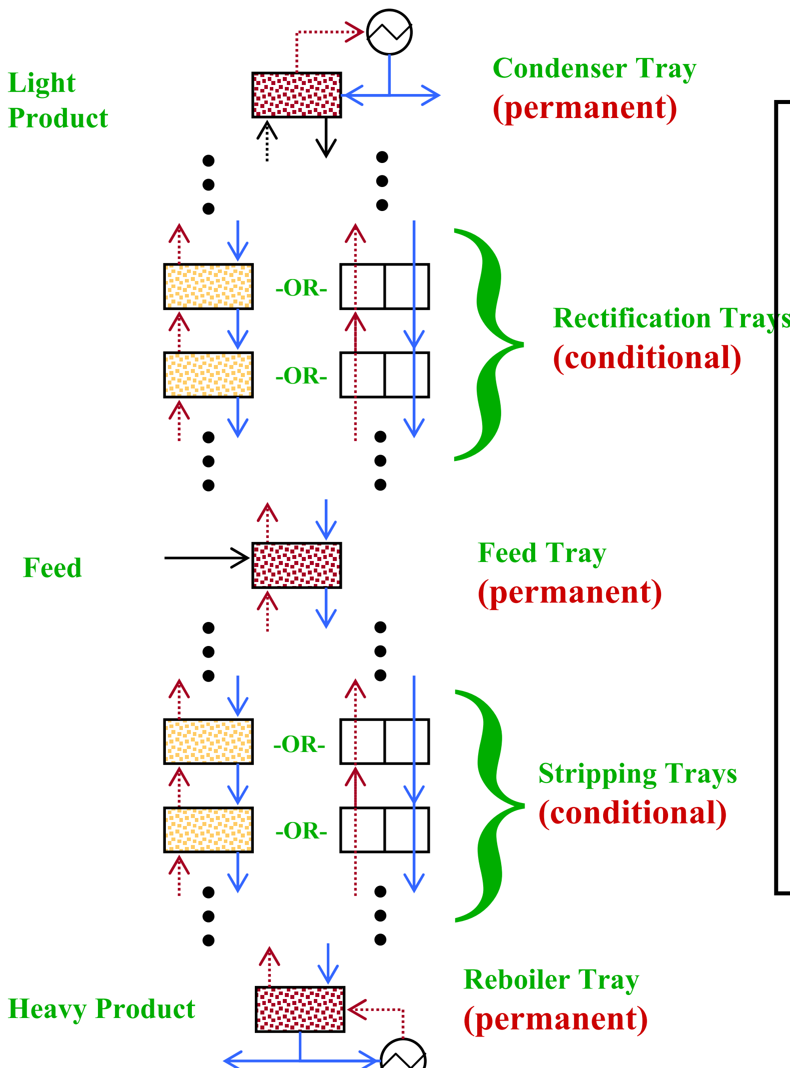
No VLE, trivial mass/energy balance if not selected

Disjunction **VLE**

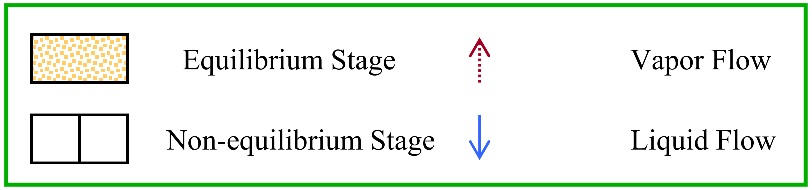


-OR- **NOT VLE (tray bypass)**

Single Column GDP Model



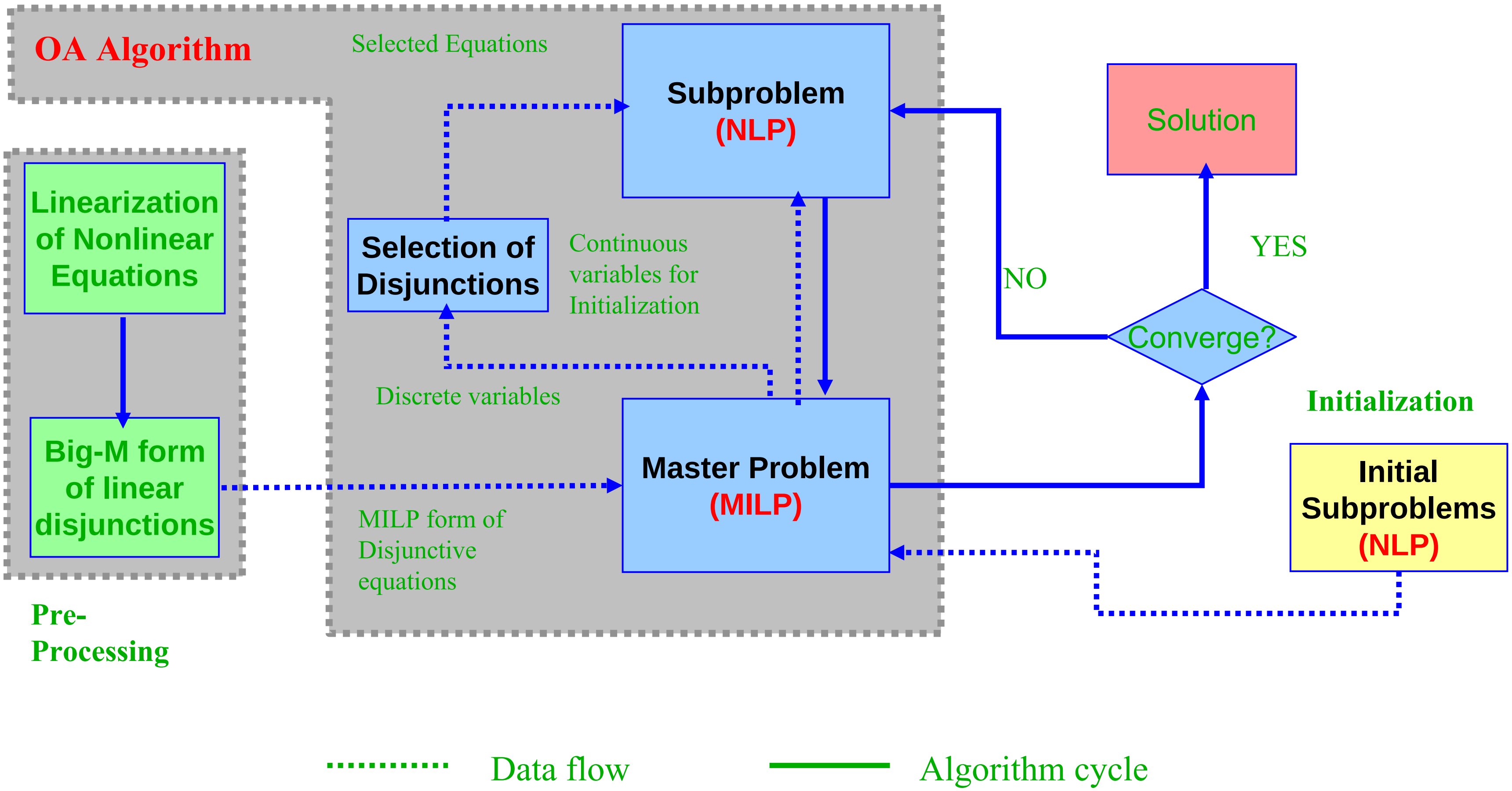
- Permanent and conditional trays:
  - MESH equations for condenser, reboiler and feed trays
  - Mass & energy balances for rectification and stripping trays.
- Conditional trays only:
  - Apply VLE constraints (Y_n=True) or not (Y_n=False)
  - Use disjunctions as modeling tool.



Carnegie Mellon

Logic-based OA Algorithm

Turkay, Grossmann (1996)



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	
Continuous Variables	1597
Constraints	1544
Total CPU time (s)	**1.12**
Model Description	
Continuous Variables	**2933**
Binary Variables	**60**
Constraints	**2862**
Nonlinear nonzero elements	5656
Number of iterations	10
NLP CPU time (s)	9.14
MILP CPU time (s)	16.97
Total CPU time (s)	**401**
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.***

Carnegie Mellon

Constraint (Logic) Programming

Van Hentenryck (1988), Puget (1994), Hooker (2000)

Variables: Continuous, integer, boolean

Constraints:

- **Algebraic** $h(x) \le 0$
- **Disjunctions** $[\, A_1 x \le b_1 \,] \vee [\, A_2 x \le b_2 \,]$
- **Conditional** If $g(x) \le 0$ then $r(x) \le 0$

Unusual Operators:

- **All different** 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:

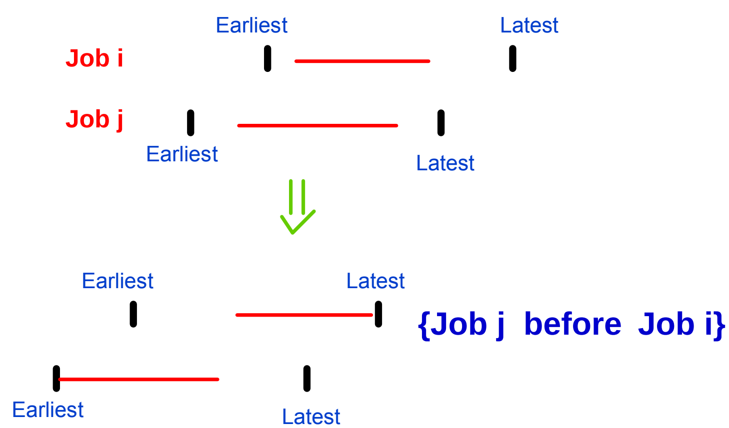
Domain Reduction

Reduction of bounds/discrete values:

a) Tighten bounds linear/monotonic functions

b)"edge-finding" for jobshop scheduling

{Job i before Job j} OR {Job j before Job i}

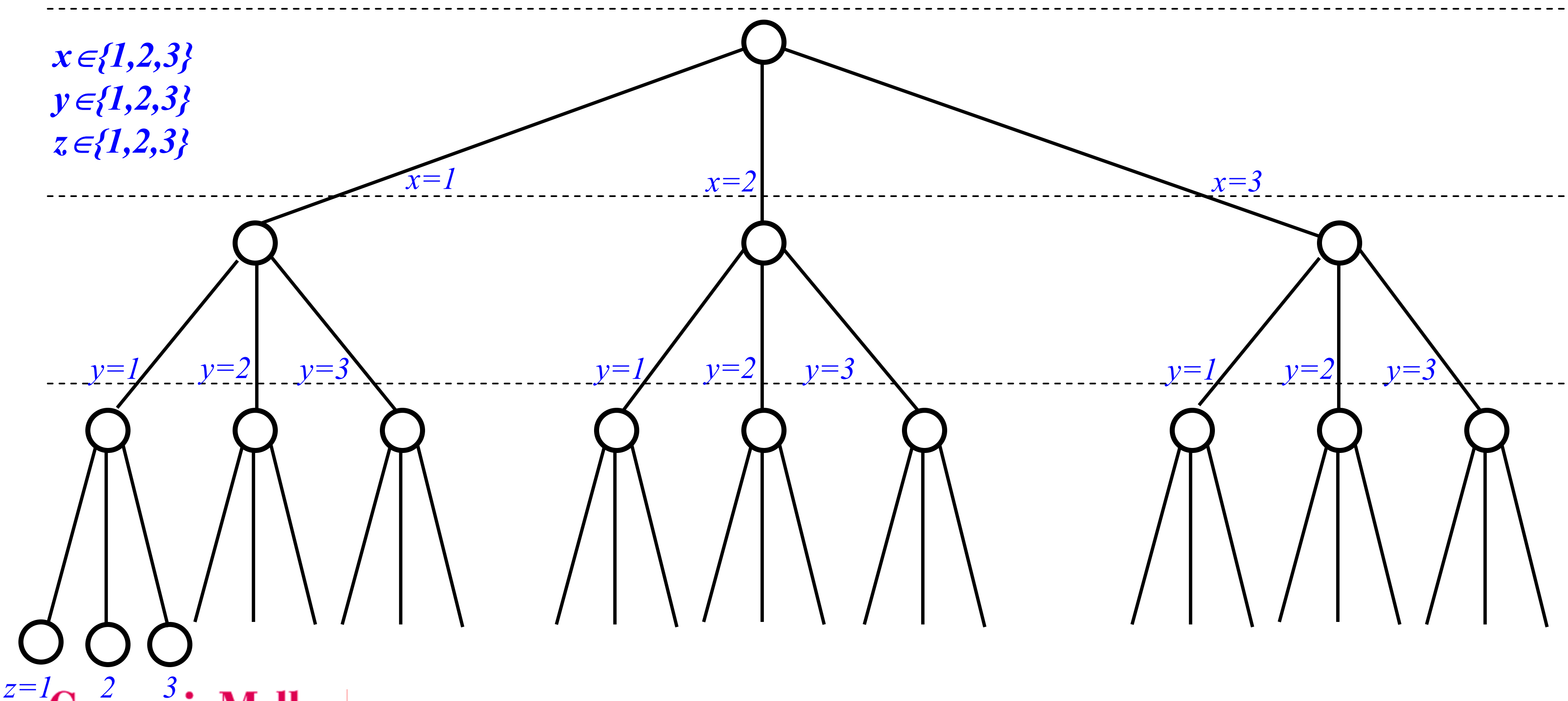


Constraint Programming

Problem

Find values for x, y and z in {1,2,3} that satisfy

$$x - y = 1$$
$$x + y + z = 6$$



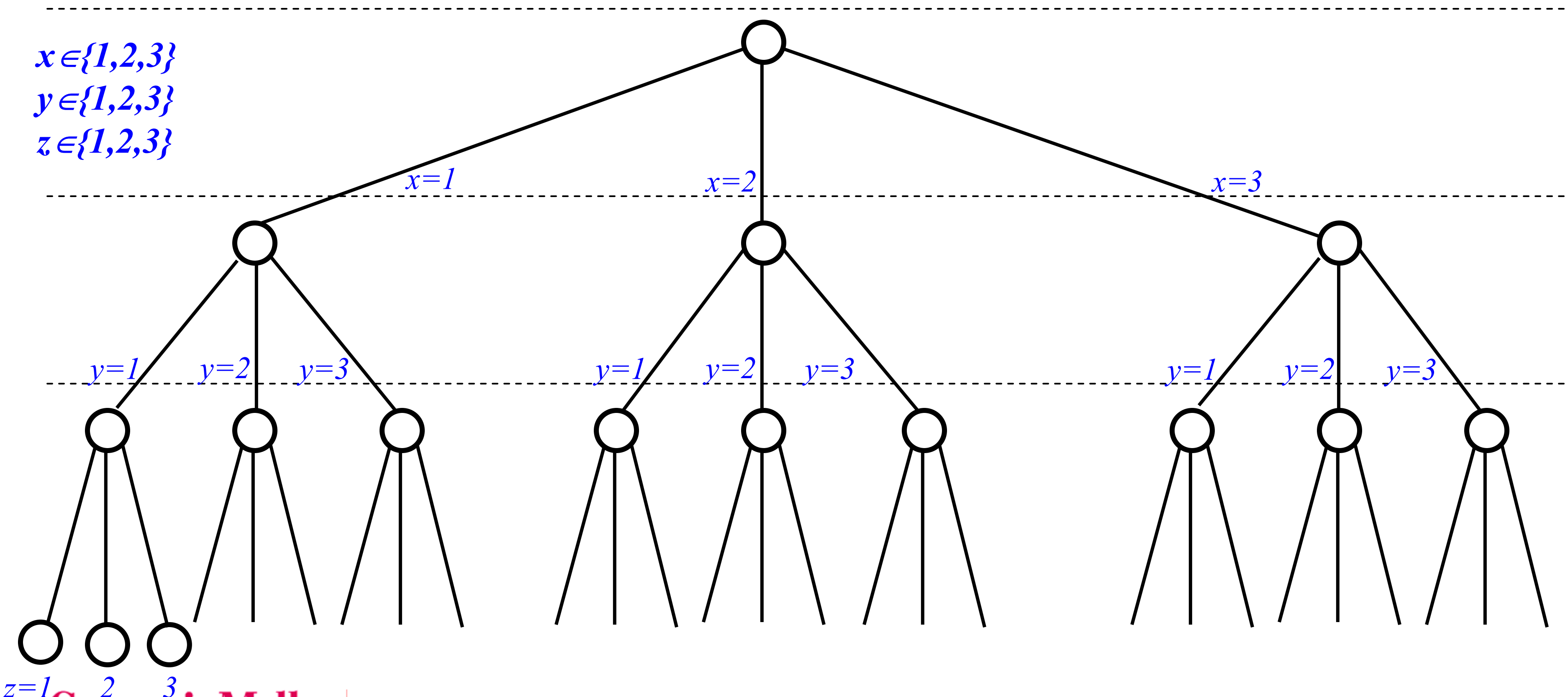
Carnegie Mellon

Constraint Programming

$x - y = 1$ $\Rightarrow x \neq 1$

$x + y + z = 6$

$x \in \{1,2,3\}$
$y \in \{1,2,3\}$
$z \in \{1,2,3\}$



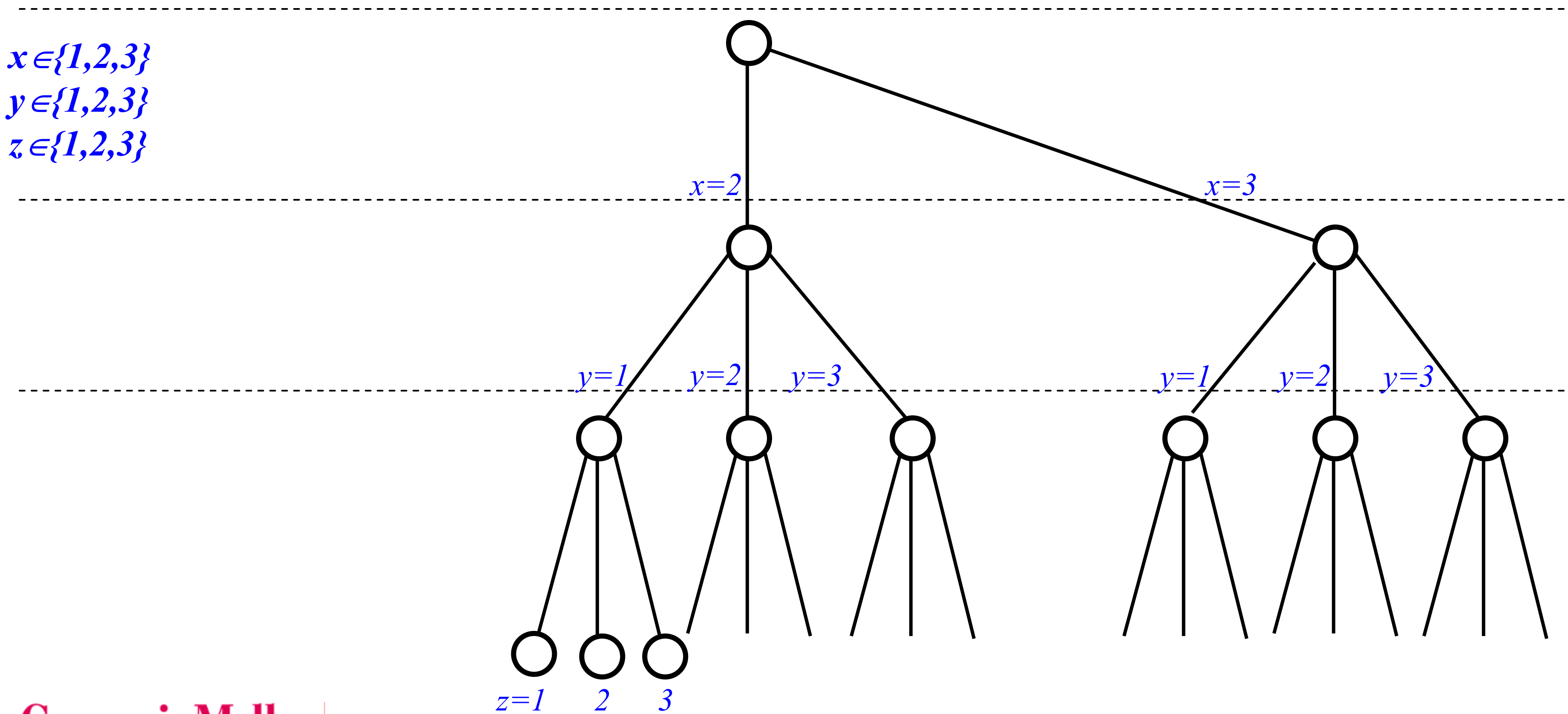
Constraint Programming

$x - y = 1$ $\Rightarrow x \neq 1$

$x=2 \Rightarrow y = 1$

$x=3 \Rightarrow y = 2$

$x + y + z = 6$



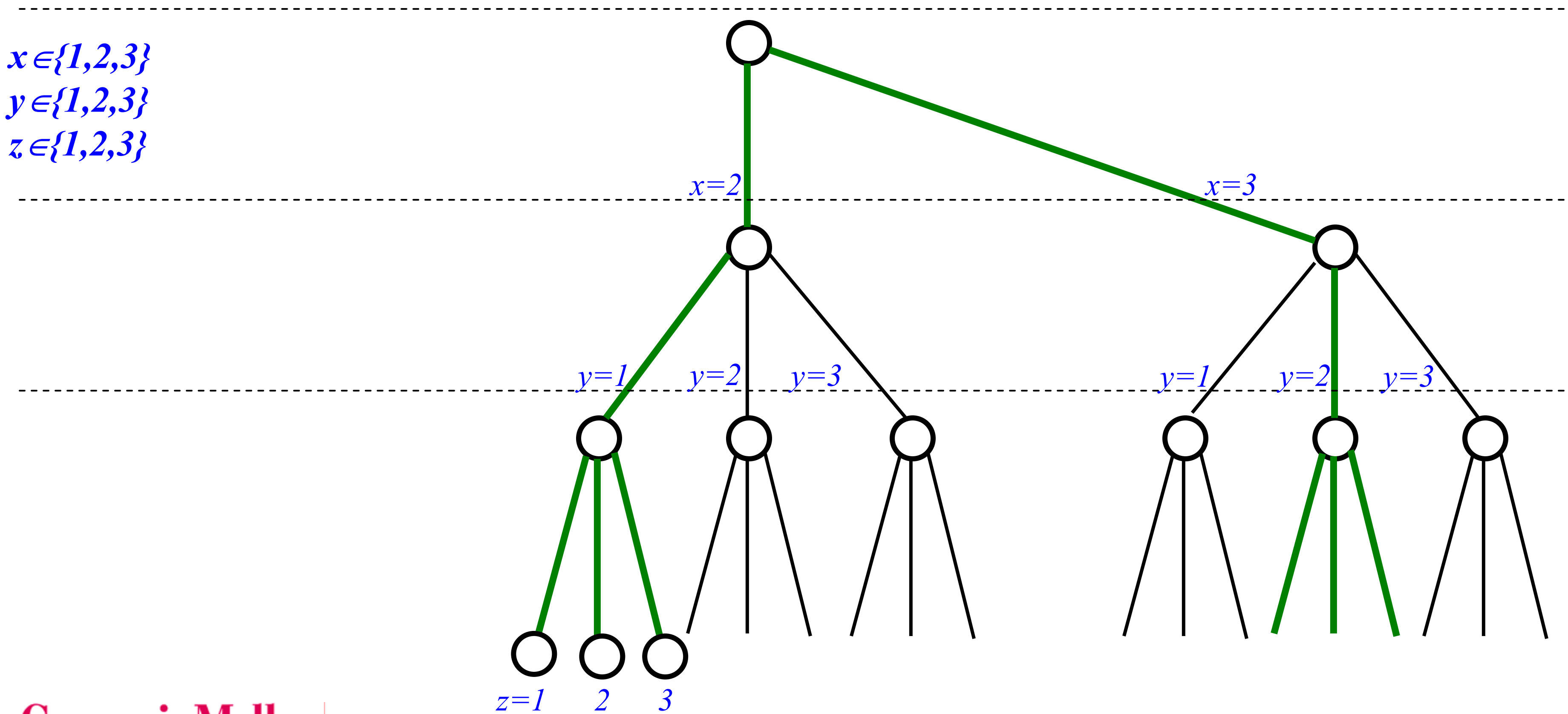
Constraint Programming

$x - y = 1$ $\Rightarrow x \neq 1$

$x=2 \Rightarrow y = 1$

$x=3 \Rightarrow y = 2$

$x + y + z = 6$



Constraint Programming

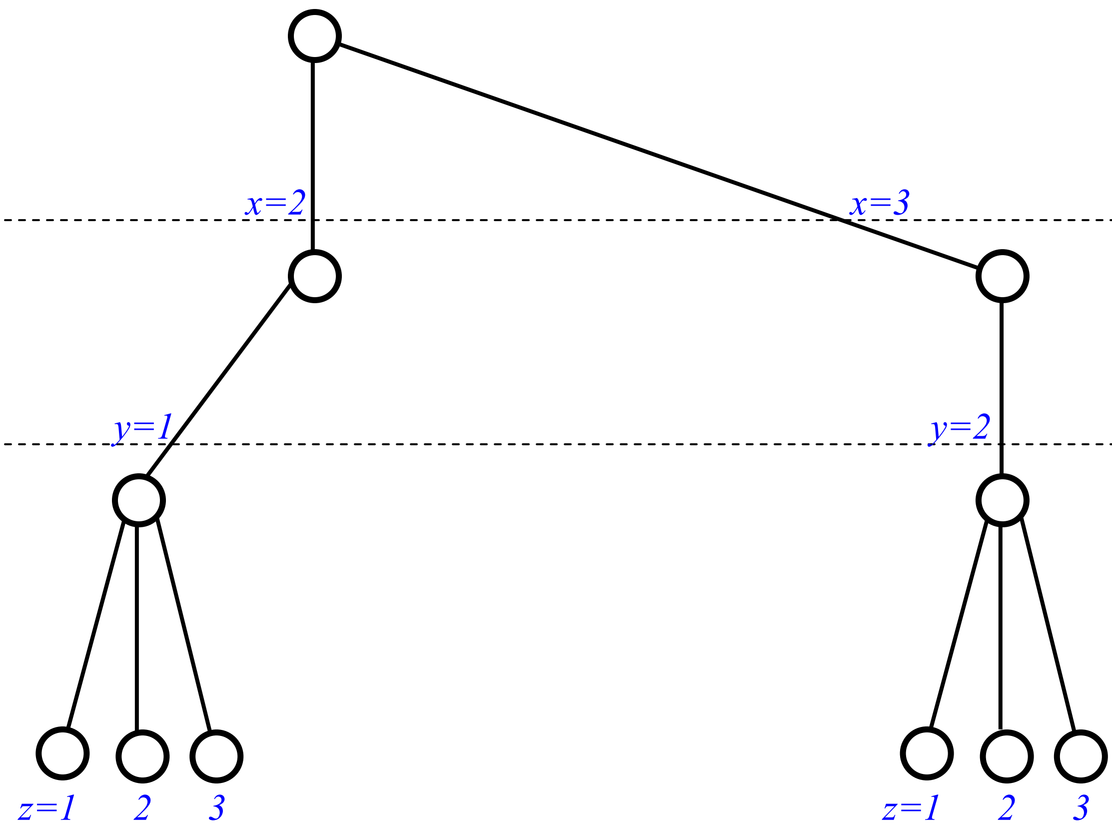
$\boldsymbol{x - y = 1}$ $\quad \Rightarrow x \neq 1$

$x=2 \Rightarrow y = 1$

$x=3 \Rightarrow y = 2$

$\boldsymbol{x + y + z = 6}$

$x \in \{1,2,3\}$
$y \in \{1,2,3\}$
$z \in \{1,2,3\}$



Constraint Programming

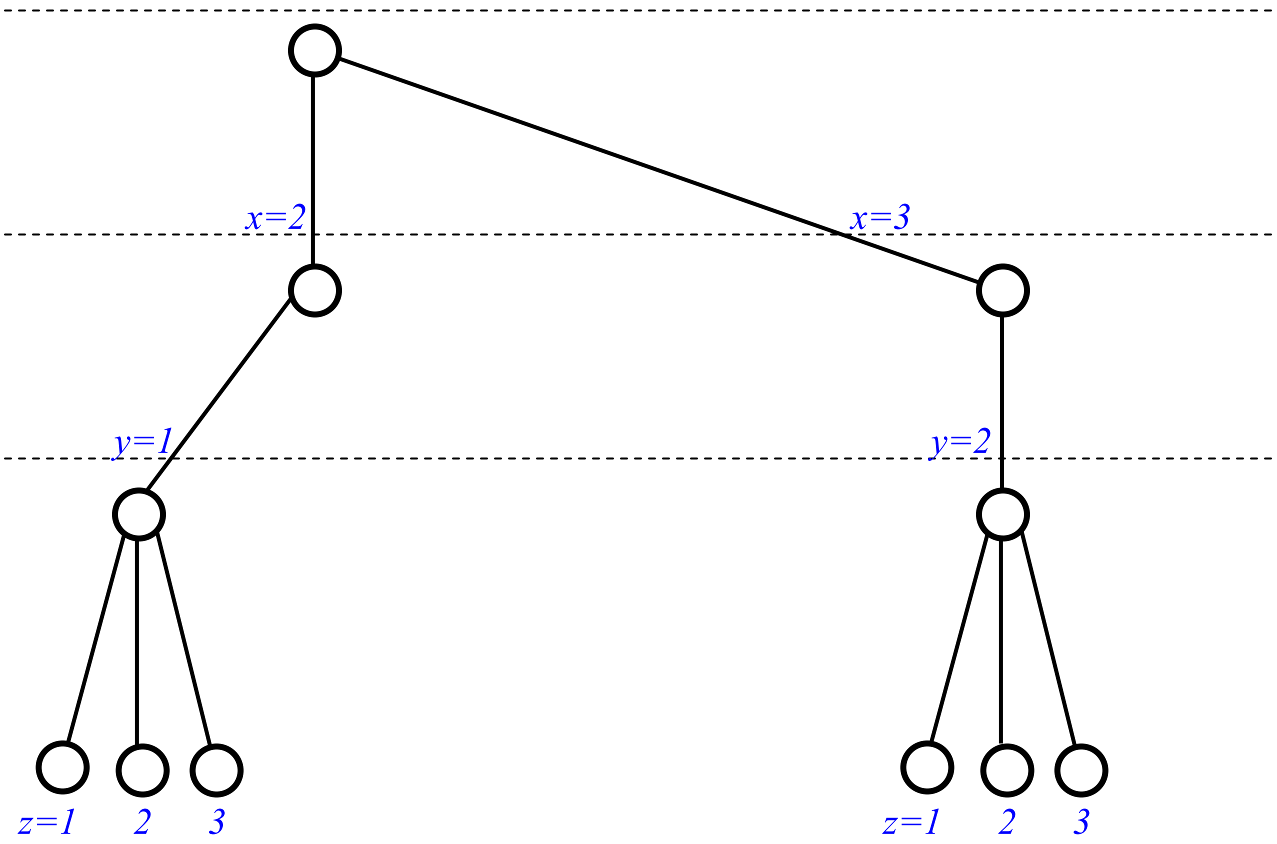
$x - y = 1 \quad \Rightarrow x \neq 1$

$x=2 \Rightarrow y = 1$

$x=3 \Rightarrow y = 2$

$x + y + z = 6$

$x \in \{1,2,3\}$
$y \in \{1,2,3\}$
$z \in \{1,2,3\}$



Carnegie Mellon

Constraint Programming

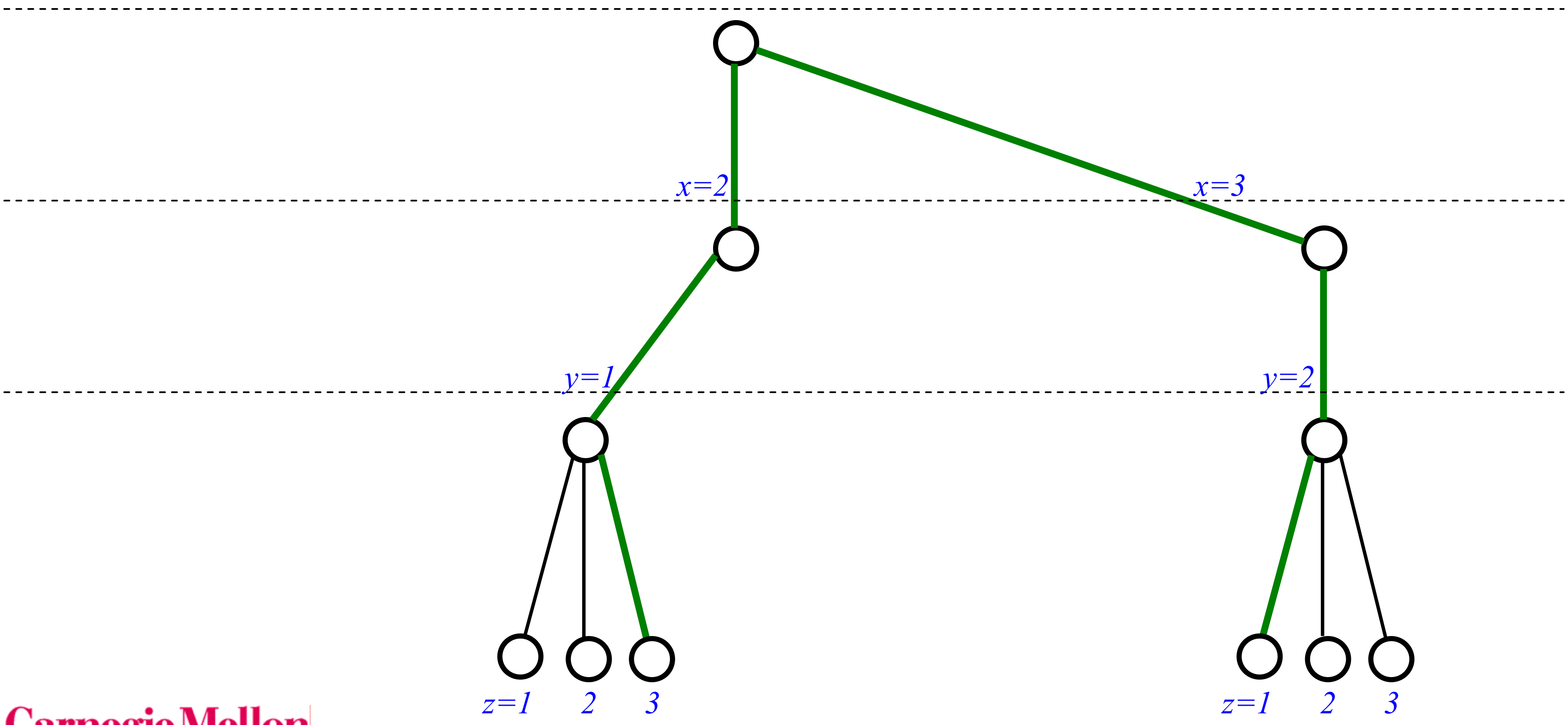
$x - y = 1 \quad \Rightarrow x \neq 1$

$x=2 \Rightarrow y = 1$

$x=3 \Rightarrow y = 2$

$\boxed{x + y + z = 6} \Rightarrow (x = 2) \wedge (y = 1) \Rightarrow z = 3$

$(x = 3) \wedge (y = 2) \Rightarrow z = 1$



Constraint Programming

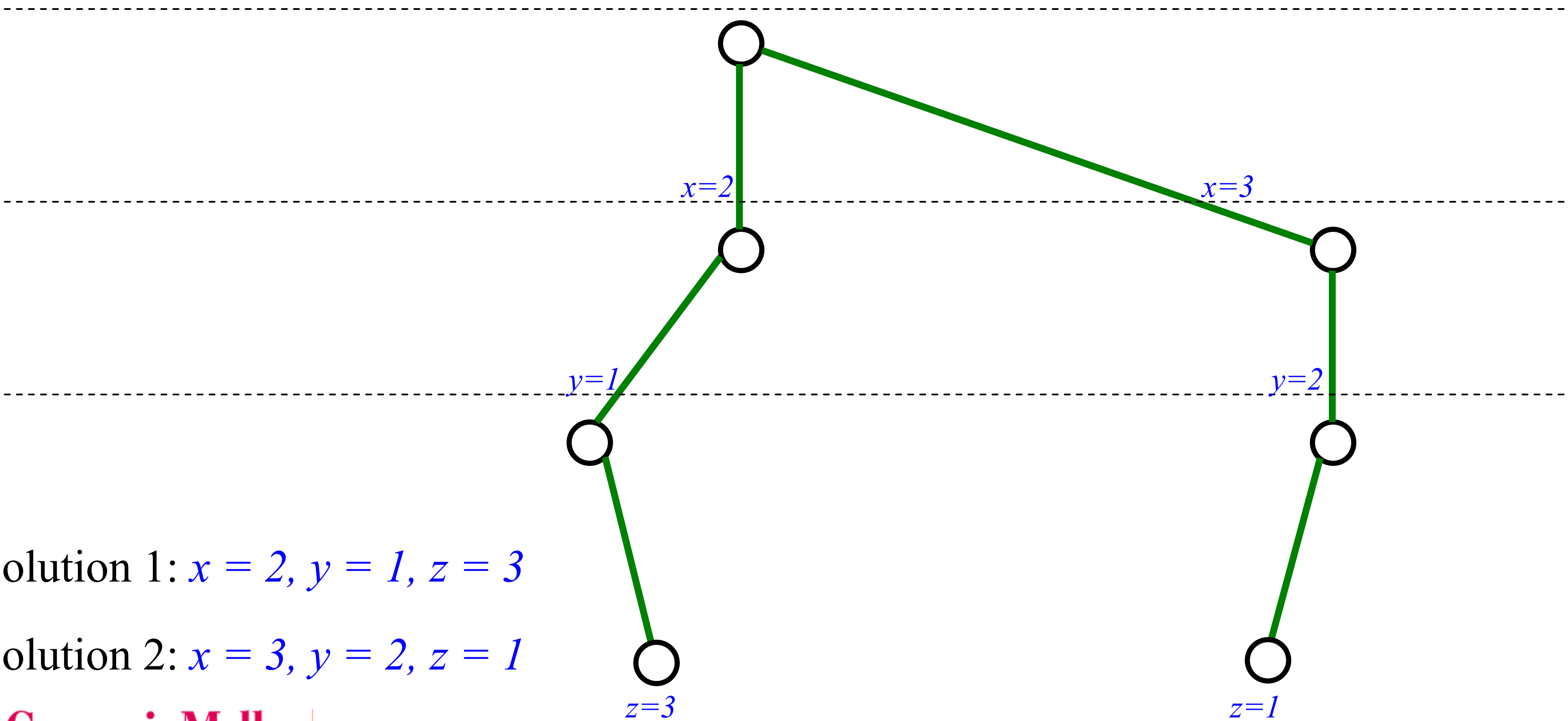
$x - y = 1 \quad \Rightarrow x \neq 1$

$x=2 \Rightarrow y = 1$

$x=3 \Rightarrow y = 2$

$\boxed{x + y + z = 6} \Rightarrow (x = 2) \wedge (y = 1) \Rightarrow z = 3$

$(x = 3) \wedge (y = 2) \Rightarrow z = 1$



Solution 1: $x = 2, y = 1, z = 3$

Solution 2: $x = 3, y = 2, z = 1$

Carnegie Mellon

OPL *(ILOG OPL Studio)* Van Hentenryck (1999)

Single language for linear and integer prpgramming, 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:

$$
\begin{aligned}
(M1): \min\ & c^T x \\
s.t.\ \ & Ax + By + Cv \le a \\
& A'x + B'y + C'v \le a' \\
& x \in \{0,1\}^n,\ y \in \{0,1\}^m,\ v \in R^p
\end{aligned}
$$

Complicating rows

Complicating variables (not in objective function)

CP:

$$
\begin{aligned}
(M2): \min\ & f(\bar{x}) \\
s.t.\ \ & G(\bar{x},\bar{y},\bar{v}) \le 0 \\
& \bar{x},\bar{y},\bar{v} \in D
\end{aligned}
$$

Hybrid:

$$
\begin{aligned}
(M3): \min\ & c^T x \\
s.t.\ \ & \left.Ax + By + Cv \le a\right\} \text{MILP (optimality)} \\
& \left.\begin{aligned} & x \Leftrightarrow \bar{x} \\ & \bar{G}(\bar{x},\bar{y},\bar{v}) \le 0 \end{aligned}\right\} \text{CP (feasibility)} \\
& x \in \{0,1\}^n,\ y \in \{0,1\}^m,\ v \in R^p \\
& \bar{x},\bar{y},\bar{v} \in D
\end{aligned}
$$

Decomposition Hybrid Model

MILP

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

$$\begin{aligned} &\min \; c^T x \\ &s.t. \;\; Ax + By + Cv \leq a \\ &\qquad Q^k x \leq q^k \;\; \forall k \in \{1,2,\ldots K-1\} \\ &\qquad x \in \{0,1\}^n, \;\; y \in \{0,1\}^m, \; v \in R^p \end{aligned}$$

No Solution → ***Infeasible***

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

CP

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

$$\begin{aligned} &\text{Find} \;\; \bar{y}, \bar{v} \\ &s.t. \;\; G(\bar{x}^K, \bar{y}, \bar{v}) \leq 0 \\ &\qquad \bar{y}, \bar{v} \in D \end{aligned}$$

Infeasible →

Generate cuts

$Q^K x \leq q^K$

Set $K = K + 1$

Feasible →

Optimal Solution

$$\sum_{i \in T^k} x_i - \sum_{i \in F^k} x_i \leq B^k - 1$$

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

No-good Cuts (weak)

Balas & Jeroslow (1972)

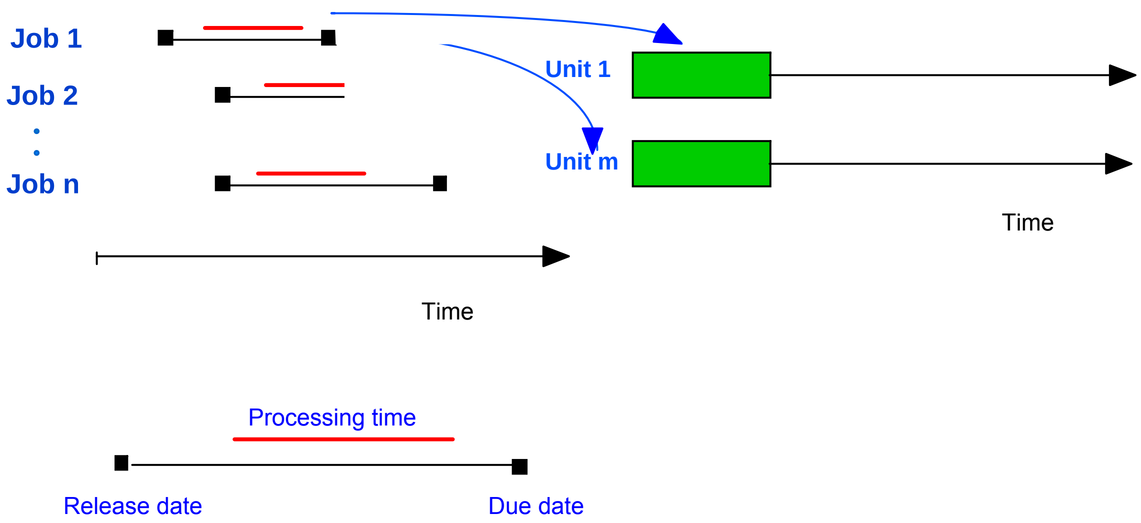
Scheduling of parallel units (Single Stage)

(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



MILP Optimization Model

$$x_{im} = \begin{cases} 1 \ \textit{if task i to unit m} \\ 0 \ \textit{otherwise} \end{cases} \qquad ts_i = \textit{start time task } i$$

$$\min \sum_{i \in I} \sum_{m \in M} C_{im} x_{im}$$ **Cost processing**

$$s.t. \quad ts_i \geq r_i$$ **Earliest start**

$$ts_i \leq d_i - \sum_{m \in M} p_{im} x_{im} \quad \forall i \in I$$ **Latest start**

$$\sum_{m \in M} x_{im} = 1 \quad \forall i \in I$$ **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

Carnegie Mellon

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

$$x_{im} = \{0,1\},\ y_{ii'} = \{0,1\}, \qquad ts_i \geq 0$$

Hybrid Optimization Model

Assignment orders to units: Mixed-integer linear programming

$$x_{im} = \begin{cases} 1 \; if \; job \; i \; to \; unit \; m \\ 0 \; otherwise \end{cases}$$

$$\begin{aligned}
&\min \sum_{i \in I} \sum_{m \in M} C_{im} x_{im} \\
&s.t. \;\; ts_i \geq r_i \\
&\qquad ts_i \leq d_i - \sum_{m \in M} p_{im} x_{im} \quad \forall i \in I \\
&\qquad \sum_{m \in M} x_{im} = 1 \quad \forall i \in I \\
&\qquad \sum_{i \in I} x_{im} p_{im} \leq \max_i \{d_i\} - \min_i \{r_i\} \quad \forall m \in M
\end{aligned}$$

Sequencing of orders in each unit

Constraint Programming

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

$$i.start \geq r_i \quad \forall i \in I$$

$$i.start \leq d_i - p_{z_i} \quad \forall i \in I$$

$$i.duration = p_{z_i}$$

$$i\ requires\ t_{z_i} \quad \forall i \in I$$

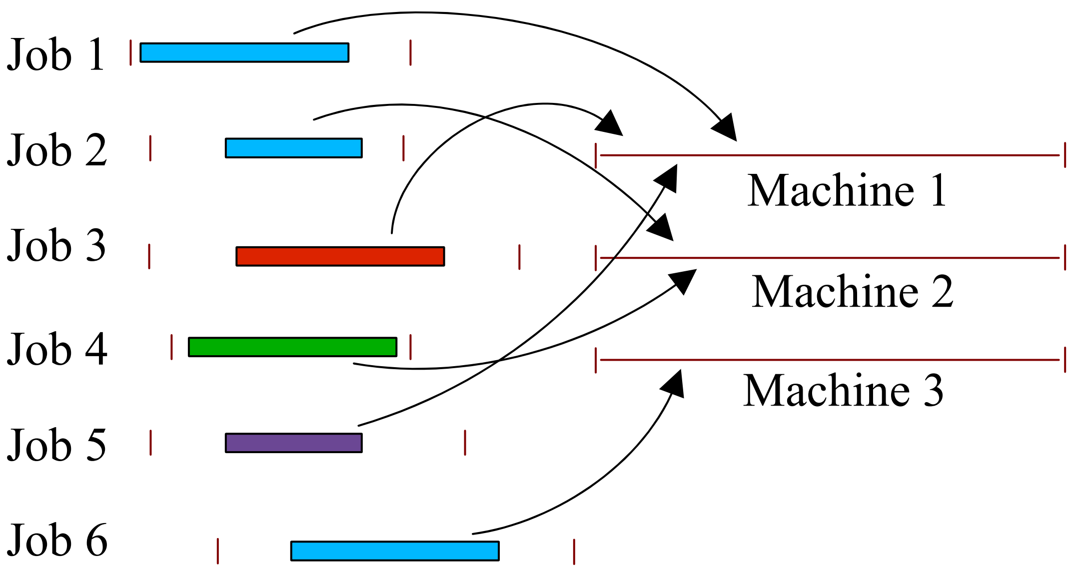
Global constraint *(implicit)*

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

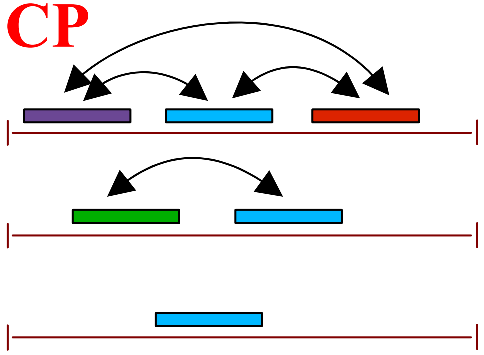
$$z_i \in M \ \forall i \in I; i.start \in D, i.duration \in D \quad \forall i \in I$$

Decomposition Strategy

MILP



CP

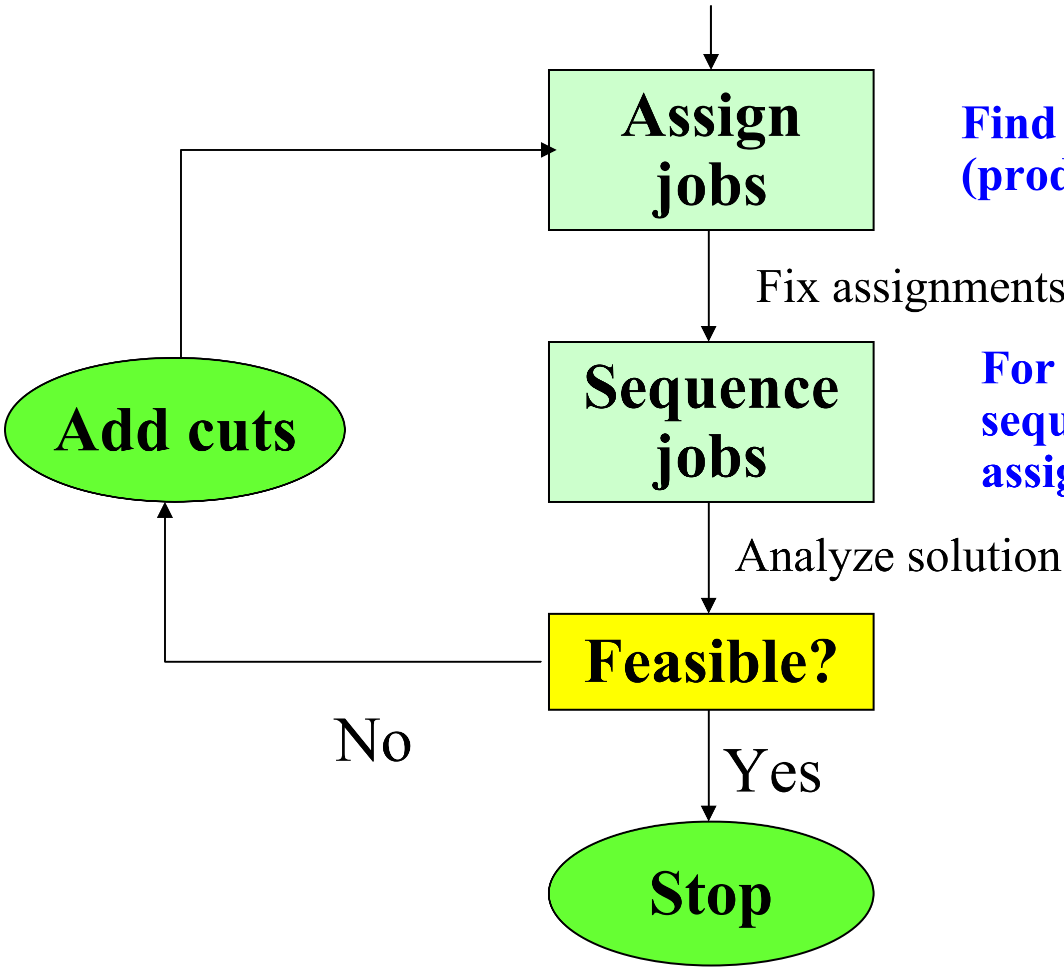


Assignment

Sequencing

Decomposition Strategy

Separate problem into assignment and sequencing problems



Find the minimum assignment (production) cost MILP

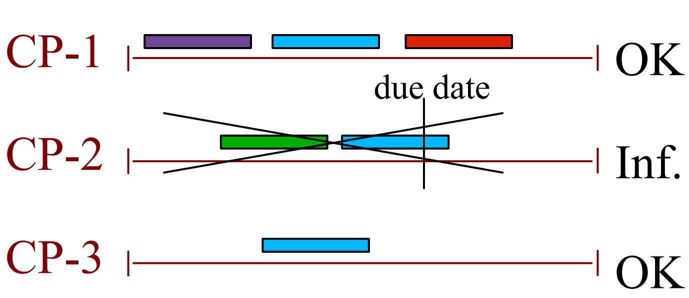
For each unit find a feasible sequence for the chosen assignment CP

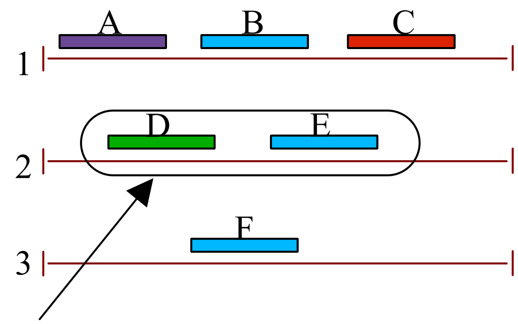
Cuts

Cuts for infeasible assignments:

$$\sum_{i \in I_m^k} x_{im} \le B_k^m - 1 \quad \forall m \in M$$

$$I_m^k = \left\{ i \,\middle|\, x_{im}^k = 1 \right\}, \; B_k^m = \left| I_m^k \right|$$





Cut out this assignment

$(x_{D2} + x_{E2} \le 1)$

- 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

	2 machines 3 jobs		3 machines 7 jobs		3 machines 12 jobs		5 machines 15 jobs		5 machines 20 jobs	
Problem	Set 1	Set 2	Set 1	Set 2	Set 1	Set 2	Set 1	Set 2	Set 1	Set 2
MILP	0.04	0.04	0.31	0.27	926.3	199.9	1784.7	73.3	18142.7	102672.3
CLP	0	0.02	0.04	0.14	3.84	0.38	553.5	9.28	68853.5	2673.9
Hybrid	0.02	0.01	0.52	0.02	4.18	0.02	2.25	0.04	14.13	0.41

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

CPLEX 6.5/Sun Ultra 60

	2M, 3J		3M, 7J		3M, 12J		5M, 15J		5M, 20J	
Problem	Set 1	Set 2	Set 1	Set 2	Set 1	Set 2	Set 1	Set 2	Set 1	Set 2
Hybrid	0.00	0.01	0.51	0.02	5.36	0.03	0.64	0.92	36.63	4.79

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