Overview of Generalized Disjunctive Programming

Ignacio E. Grossmann
Center for Advanced Process Decision-making
Carnegie Mellon University
Pittsburgh, PA 15213, USA

EWO Seminar

January 22, 2009

Motivation

- **EWO Problems involve Discrete/Continuous Optimization**
  - **Linear/Nonlinear models**
  - **0-1 and continuous decisions**

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

Alternative approach:

- **Logic-based: Generalized Disjunctive Programming (GDP)**

- **Outline**
  - **Modeling with GDP (LOGMIP)**
  - **Solution methods (relaxations)**
  - **Example supply chain with contracts for purchases**
  - **New frontiers:**

    - Improved relaxations (basic steps) (Strip packing problem)

    - Nonconvex GDPs

MI(N)LP

- **Mixed-Integer Linear/Nonlinear Programming**

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

$$s.t. \quad h(x,y) = 0$$ *Equality Constraints*

$$g(x,y) \le 0$$ *Inequality Constraints*

$$x \in R^n, \quad y \in \{0,1\}^m$$

MILP: ***f(x,y)*** **and** ***g(x,y)*** **linear in** ***x*** **and** ***y***

MINLP: ***f(x,y)*** **and** ***g(x,y)*** **– commonly assumed to be convex and bounded**

often not true in practice => nonconvex

f(x,y) **and** ***g(x,y)*** **commonly linear in** ***y*** ***often true in practice***

Mixed-integer Nonlinear Programming

Codes MILP:
Commercial : CPLEX, XPRESS, (GUROBI ?) XA, OSL
Open Source (COIN-OR) : CBC, SYMPHONY

Codes MINLP:
SBB ***GAMS simple B&B***
MINLP-BB (AMPL) ***Fletcher and Leyffer (1999)***

Bonmin (COIN-OR) ***Bonami et al (2006)***
FilMINT ***Linderoth and Leyffer (2006)***

DICOPT (GAMS) ***Viswanathan and Grossman (1990)***
AOA (AIMSS)

α–ECP ***Westerlund and Peterssson (1996)***
MINOPT ***Schweiger and Floudas (1998)***

BARON ***Sahinidis et al. (1998)***
Couenne (COIN-OR) ***Belottti et al. (2008)***

Global solvers

Generalized Disjunctive Programming

Motivation

1. **Facilitate modeling of discrete/continuous optimization problems through use algebraic constraints and *symbolic logic expressions***
2. **Improve combinatorial search effort**
3. **Improve handling nonlinearities**

Basic References

Definition of GDP

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

Logic-based Outer-Approximation

Turkay, M. and I.E. Grossmann, "Logic-Based MINLP Algorithms For the Optimal Synthesis of Process Networks," *Computers and Chemical Engineering* , 20, 959-978 (1996).

Convex hull nonlinear disjunctions

Lee, S. and I.E. Grossmann, "New Algorithms for Nonlinear Generalized Disjunctive Programming," *Computers and Chemical Engineering*, 24, pp.2125-2141 (2000).

Established relation with DP

Sawaya, N. and I.E. Grossmann, "Reformulations, Relaxations and Cutting Planes for Linear Generalized Disjunctive Programming," submitted for publication *Math. Prog.* (2008).

LOGMIP

Vecchietti, A. and I.E. Grossmann, "LOGMIP: A Disjunctive 0-1 Nonlinear Optimizer for Process Systems Models, *Computers and Chemical Engineering* 23, 555-565 (1999).

Chemical ENGINEERING

CAPD CENTER

Background reading

Balas E., "Disjunctive Programming and a Hierarchy of Relaxations for Discrete Continuous Optimization Problems", ***SIAM J. Alg. Disc. Meth*****., Vol. 6, No. 3, 1985.**

Hooker J., "Logic-based methods for optimization: combining optimization and constraint satisfaction", John Wiley & Sons, 2000.

Jeroslow R.G., "Logic-Based Decision Support: Mixed-Integer Model Formulation", ***Annals of Discrete Mathematics*****, 40, North Holland (Amsterdam), 1989.**

"Typical" disjunction for scheduling

Product A and Product B must be processed on same line

Sequencing decision:

A before B OR B before A

Let T_A, T_B be starting times (*variables*)

Let p_A, p_B be processing times (*parameters*)

Disjunction

$$\left[T_A + p_A \le T_B\right] \vee \left[T_B + p_B \le T_A\right]$$

A before B OR B before A

T_A T_B T_B T_A

Generalized Disjunctive Programming (GDP)

Quantitative/Symbolic Optimization Model

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

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

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

OR operator $\longrightarrow$

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

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**

Carnegie Mellon

Small GDP Problem

$$\min Z = c + 2x_1 + x_2$$

$$s.t.$$

$$\begin{bmatrix} Y_1 \\ -x_1 + x_2 + 2 \le 0 \\ c = 5 \end{bmatrix} \vee \begin{bmatrix} Y_2 \\ 2 - x_2 \le 0 \\ c = 7 \end{bmatrix}$$

$$\begin{bmatrix} Y_3 \\ x_1 - x_2 \le 1 \end{bmatrix} \vee \begin{bmatrix} \neg Y_3 \\ x_1 = 0 \end{bmatrix}$$

$$Y_1 \wedge \neg Y_2 \Rightarrow \neg Y_3$$

$$\neg(Y_2 \wedge Y_3)$$

$$0 \le x_1 \le 5,\ 0 \le x_2 \le 5,\ c \ge 0$$

$$Y_j \in \{true, false\},\ j = 1, 2, 3.$$

Carnegie Mellon

Input file LOGMIP (GAMS)

```
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 PEQUE2 /ALL/;
SOLVE PEQUE2 USING MIP MINIMIZING Z;
```

Big-M Formulation

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/

Generalized Disjunctive Programming (GDP)

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

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

OR operator →

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

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**

Solution methods: Relaxation?

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* $Q_i : P_1 \vee P_2 \vee \ldots \vee P_r$ $\Rightarrow \sum_{i \in Nonneg} y_i + \sum_{i \in Neg} (1 - y_i) \geq 1$
and P_i is a *literal*

Linear inequality

Steps to obtain CNF **(Clocksin, Melish, 1981)**

1. Replace implication by disjunction

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

$\neg$ ***negation***

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

If prod A or prod B implies reactor 3 or reactor 4

P_1 P_2 P_3 P_4

$P_1 \vee P_2 \Rightarrow P_3 \vee P_4$

1. Remove implication

$\neg(P_1 \vee P_2) \vee (P_3 \vee P_4)$

2. Apply De Morgan's

$(\neg P_1 \wedge \neg P_2) \vee (P_3 \vee P_4)$

3. Distribute OR over AND

$\underbrace{(\neg P_1 \vee P_3 \vee P_4)}_{\text{clause 1}} \wedge \underbrace{(\neg P_2 \vee P_3 \vee P_4)}_{\text{clause 2}}$ ***=> CNF !***

from clause 1

$1 - y_1 + y_3 + y_4 \geq 1$

from clause 2

$1 - y_2 + y_3 + y_4 \geq 1$

$$\boxed{\begin{array}{l} y_3 + y_4 \geq y_1 \\ y_3 + y_4 \geq y_2 \end{array}}$$

Carnegie Mellon

Big-M MI(N)LP (BM)

- **MINLP reformulation of GDP**

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

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

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

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

$$A\lambda \le a$$

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

Big-M Parameter

Logic constraints

LP/NLP Relaxation $0 \le \lambda_{jk} \le 1$

Carnegie Mellon

Convex Hull Formulation (CH)

- **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** v^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

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

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

Convex for g(x) convex

  - **λ_j - weights for linear combination**

Remark I

For linear disjunctions

$$\bigvee_{j \in J_k} \left[A_{jk} x \le b_{jk} \right]$$

Convex-hull $\quad set \;\; g_{jk}(x) = A_{jk} x - b_{jk}$

$$x = \sum_{j \in J_k} \nu_{jk}$$

$$A_{jk} \nu_{jk} \le b_{jk} \lambda_{jk} \quad j \in J_k$$

Balas (1985)

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

$$0 \le \lambda_{jk} \le 1 \quad j \in J_k$$

Remark II

For special cases disaggregated variables can be eliminated

Example

$$\begin{bmatrix} Y \\ L \le x \le U \end{bmatrix} \vee \begin{bmatrix} \neg Y \\ x = 0 \end{bmatrix}$$

Convex hull formulation

$$x = v^1 + v^2$$

$$L\lambda \le v^1 \le U\lambda$$

$$v^2 = 0 * (1 - \lambda)$$

$$\lambda = 0,1$$

Since $v^2 = 0 \Rightarrow$

$$\boxed{\begin{array}{l} L\lambda \le x \le U\lambda \\ \lambda = 0,1 \end{array}}$$

w-MIP representable (Raman, G, 1994)

See also John Hooker EWO Seminar
Tour Modeling Techniques

Remark III

Nonlinear disjunctions

How to implement $\lambda_{jk} g_{jk}(\nu_{jk}/\lambda_{jk}) \le 0$ for λ_{jk} zero ?

Replace $\lambda_{jk} g_{jk}(\nu_{jk}/\lambda_{jk}) \le 0$ where $0 \le \nu_{jk} \le U\lambda_{jk}$ by:

$$\boxed{((1-\varepsilon)\lambda_{jk}+\varepsilon)(g_{jk}(\nu_{jk}/((1-\varepsilon)\lambda_{jk}+\varepsilon))) - \varepsilon g_{jk}(0)(1-\lambda_{jk}) \le 0}$$

Furman, Sawaya & Grossmann (2009)

a. Exact approximation of the original constraints as $\varepsilon \to 0$.

b. The constraints are exact at $\lambda_{jk} = 0$ and at $\lambda_{jk} = 1$ regardless of value of ε.

$$if\ \lambda_{jk} = 0, \Rightarrow (\varepsilon)(g_{jk}(0)) - \varepsilon g_{jk}(0) = 0 \le 0$$

$$if\ \lambda_{jk} = 1, \Rightarrow ((1)(g_{jk}(\nu_{jk}/(1)) - \varepsilon g_{jk}(0)(0) = (1) g_{jk}(\nu_{jk}/(1)) \le 0$$

c. The LHS of the new constraint is **convex**.

Carnegie Mellon

MI(N)LP Reformulation (CH)

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

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

$$\boxed{\begin{aligned} & x = \sum_{j \in J_k} \nu^{jk}, \; k \in K \\ & 0 \le \nu^{jk} \le \lambda_{jk} U_{jk}, \; j \in J_k, k \in K \\ & \sum_{j \in J_k} \lambda_{jk} = 1 \;, k \in K \\ & \lambda_{jk} g_{jk}(\nu^{jk} / \lambda_{jk}) \le 0, \quad j \in J_k, k \in K \end{aligned}}$$

$$A\lambda \le a$$

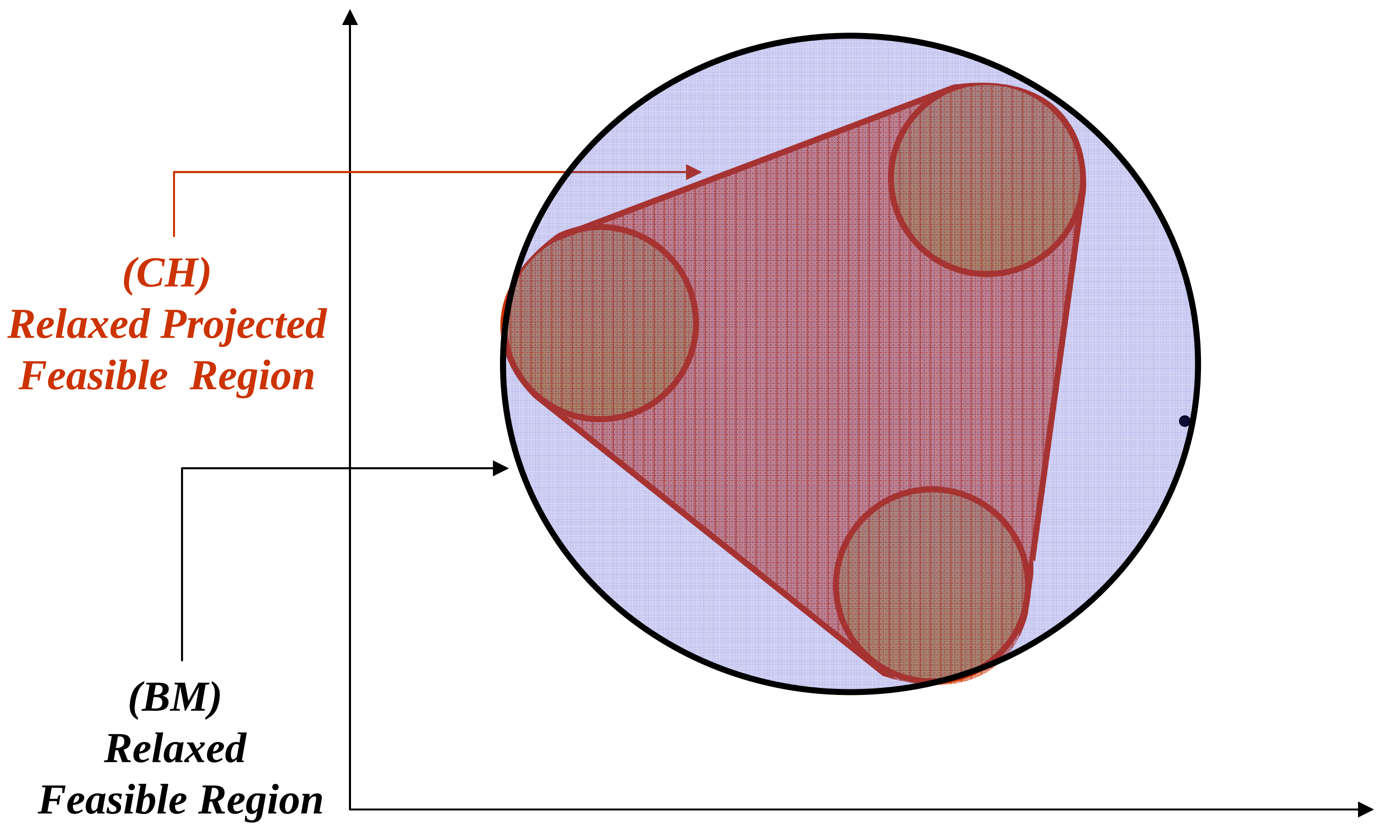
$$x, \nu^{jk} \ge 0, \; \lambda_{jk} = 0,1 \;\; j \in J_k, k \in K$$

LP/NLP Relaxation $\quad 0 \le \lambda_{jk} \le 1$

Convex Hull vs Big-M

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



Methods Generalized Disjunctive Programming

GDP

Logic based methods

Branch and bound

(Lee & Grossmann, 2000)

Decomposition

Outer-Approximation

(Turkay & Grossmann, 1997)

Reformulation MI(N)LP

Convex-hull

Big-M

LP-based B&B
Outer-Approximation
Extended Cutting Plane

A Branch and Bound Algorithm for GDP

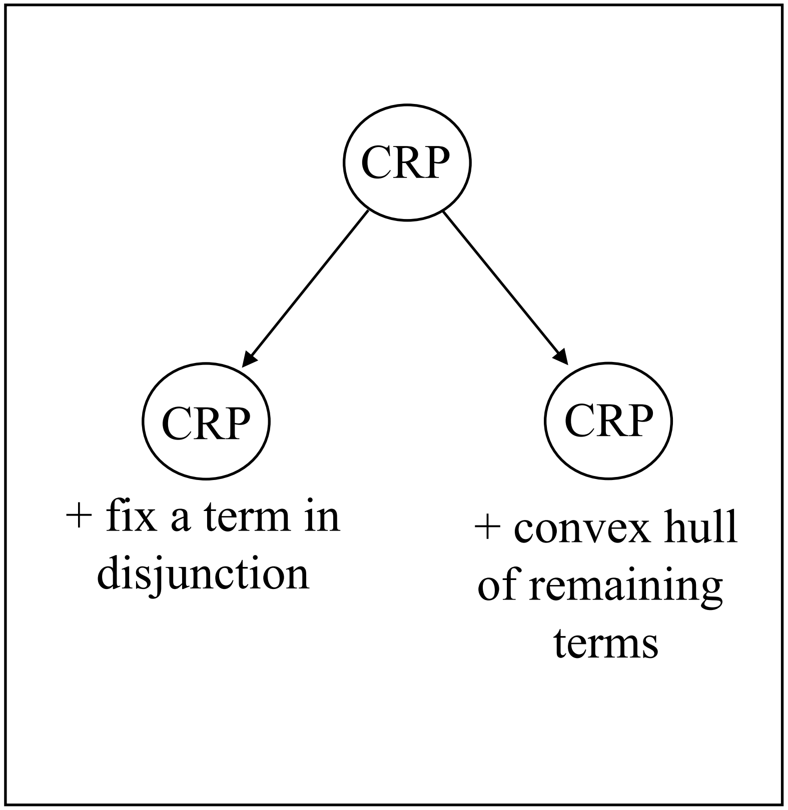
- **Tree Search**
  - **NLP subproblem at each node**
- **Solve Relaxation GDP (CRP)**
  - **lower bound**

- **Branching Rule**
  - **Set the largest λ_j as 1**
  - **Dichotomy rule**
- **Logic inference**

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

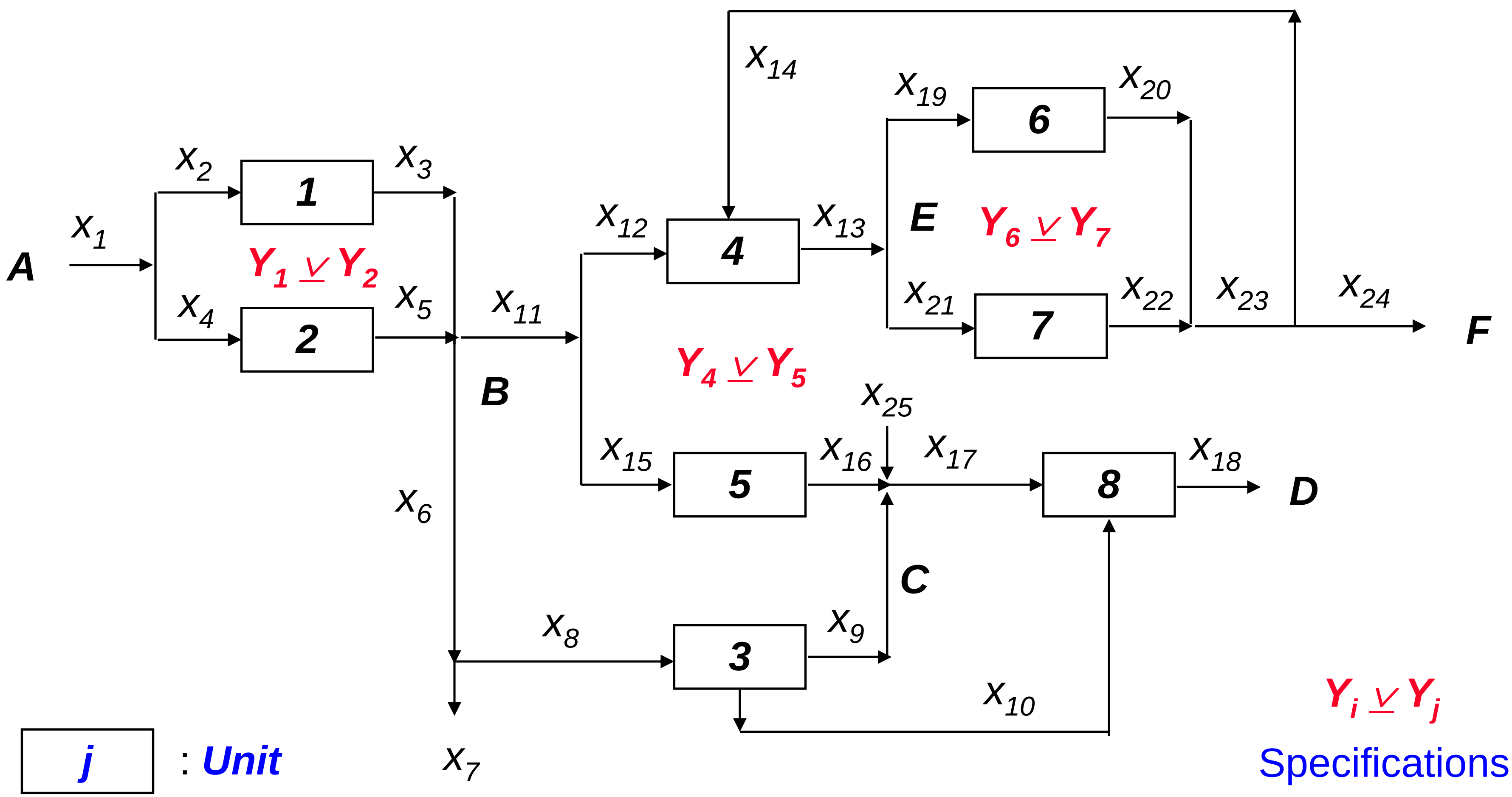
  When all the terms are fixed

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



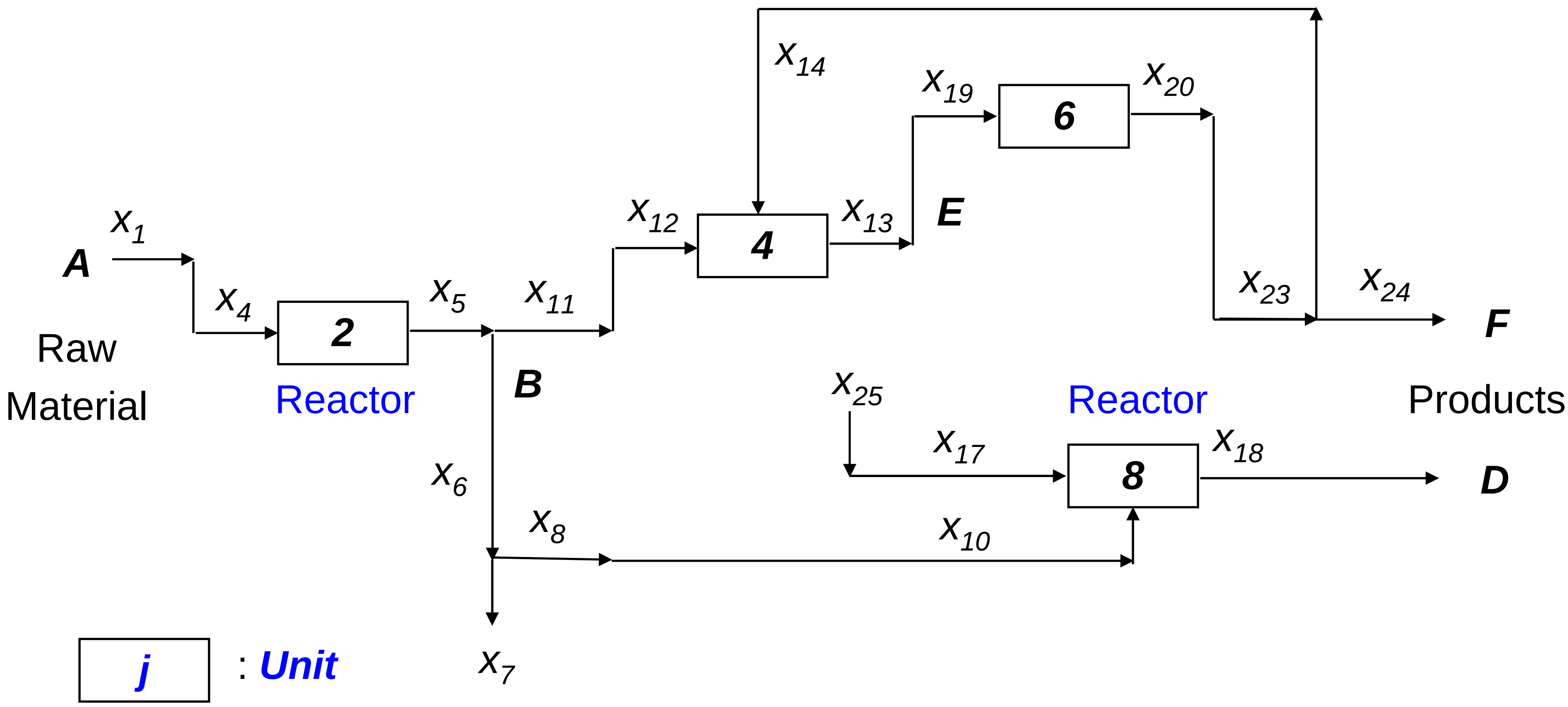
Process Network with Fixed Charges

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

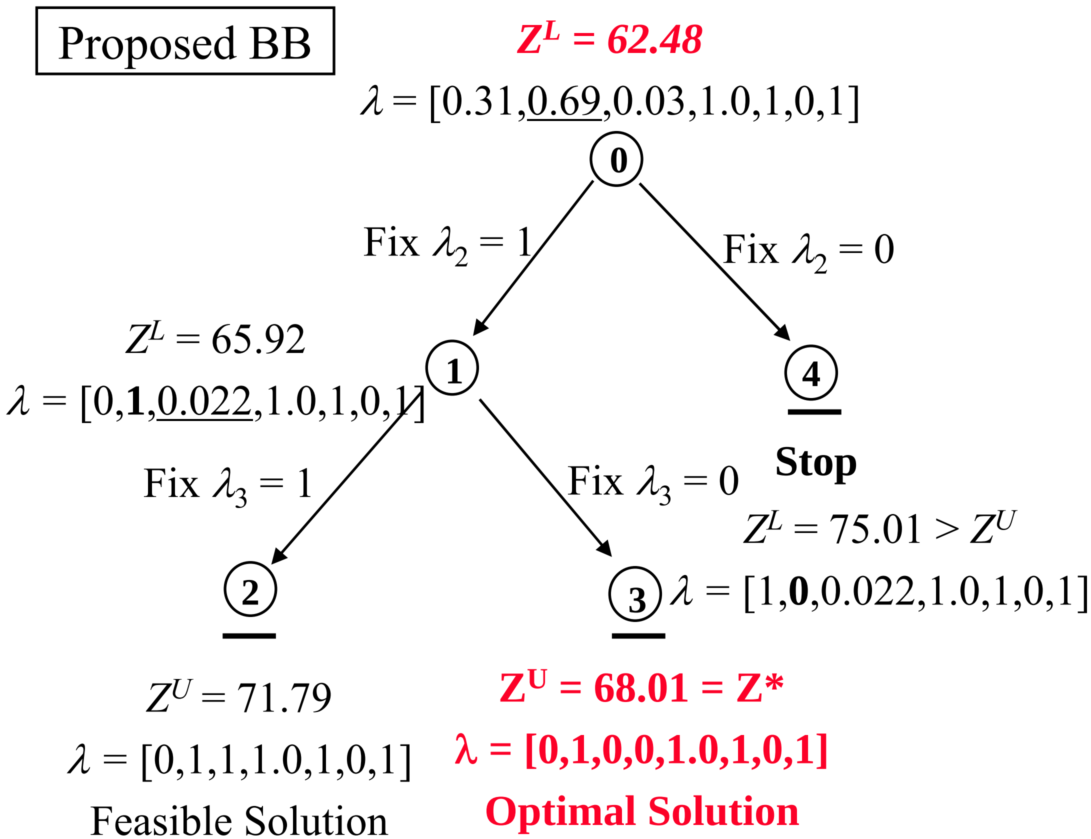


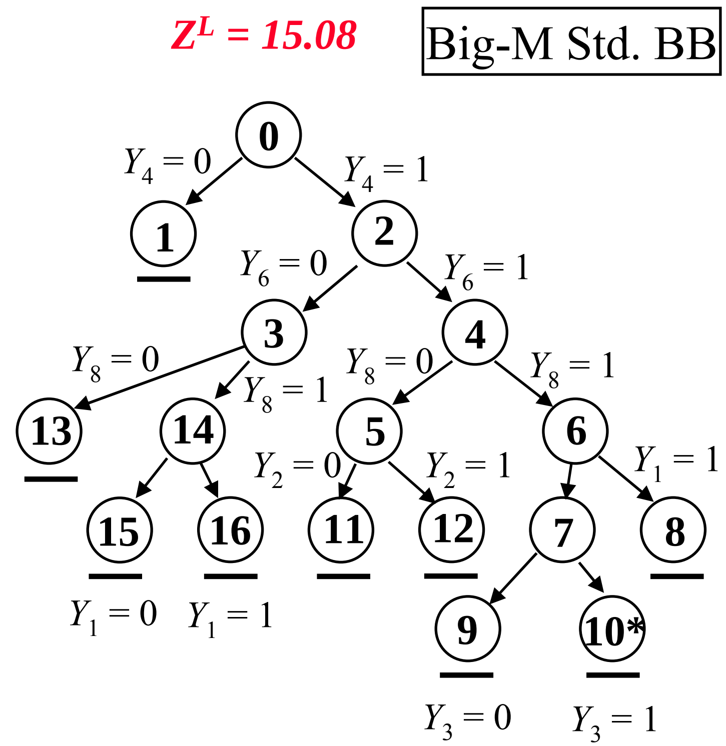
Optimal solution

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



Proposed BB Method





- 5 nodes vs. 17 nodes of Standard BB (lower bound = 15.08)

Logic-based Outer Approximation

Main point: avoids solving MINLP in full space

Turkay, Grossmann (1997)

NLP Subproblem: (reduced)

$$\min Z = \sum_{k \in SD} c_k + f(x)$$
$$s.t.\quad g(x) \le 0$$
$$\left.\begin{array}{l} h_{ik}(x) \le 0 \\ c_k = \gamma_{ik} \end{array}\right\} for\, Y_{\hat{i}k} = true\quad \hat{i} \in D_k, k \in SD \qquad \text{(NLPD)}$$
$$\left.\begin{array}{l} B^i x = 0 \\ c_k = 0 \end{array}\right\} for\, Y_{ik} = false\quad i \in D_k, i \ne \hat{i}, k \in SD$$
$$x \in R^n,\ c_i \in R^m,$$

Redundant constraints are eliminated with false values

Master Problem:

$$Min\quad Z = \sum_k c_k + \alpha$$
$$s.t.\quad \left.\begin{array}{l} \alpha \ge f(x^l) + \nabla f(x^l)^T (x - x^l) \\ g(x^l) + \nabla g(x^l)^T (x - x^l) \le 0 \end{array}\right\} l = 1,\ldots,L \qquad \text{(MGDP)}$$
$$\bigvee_{i \in D_k} \begin{bmatrix} Y_{ik} \\ h_{ik}(x^l) + \nabla h_{ik}(x^l)^T (x - x^l) \le 0 \\ l \in L_{ik} \\ c_k = \gamma_{ik} \end{bmatrix} \quad k \in SD$$
$$\Omega(Y) = True$$
$$\alpha \in R,\ x \in R^n,\ c \in R^m,\ Y \in \{true, false\}^m$$

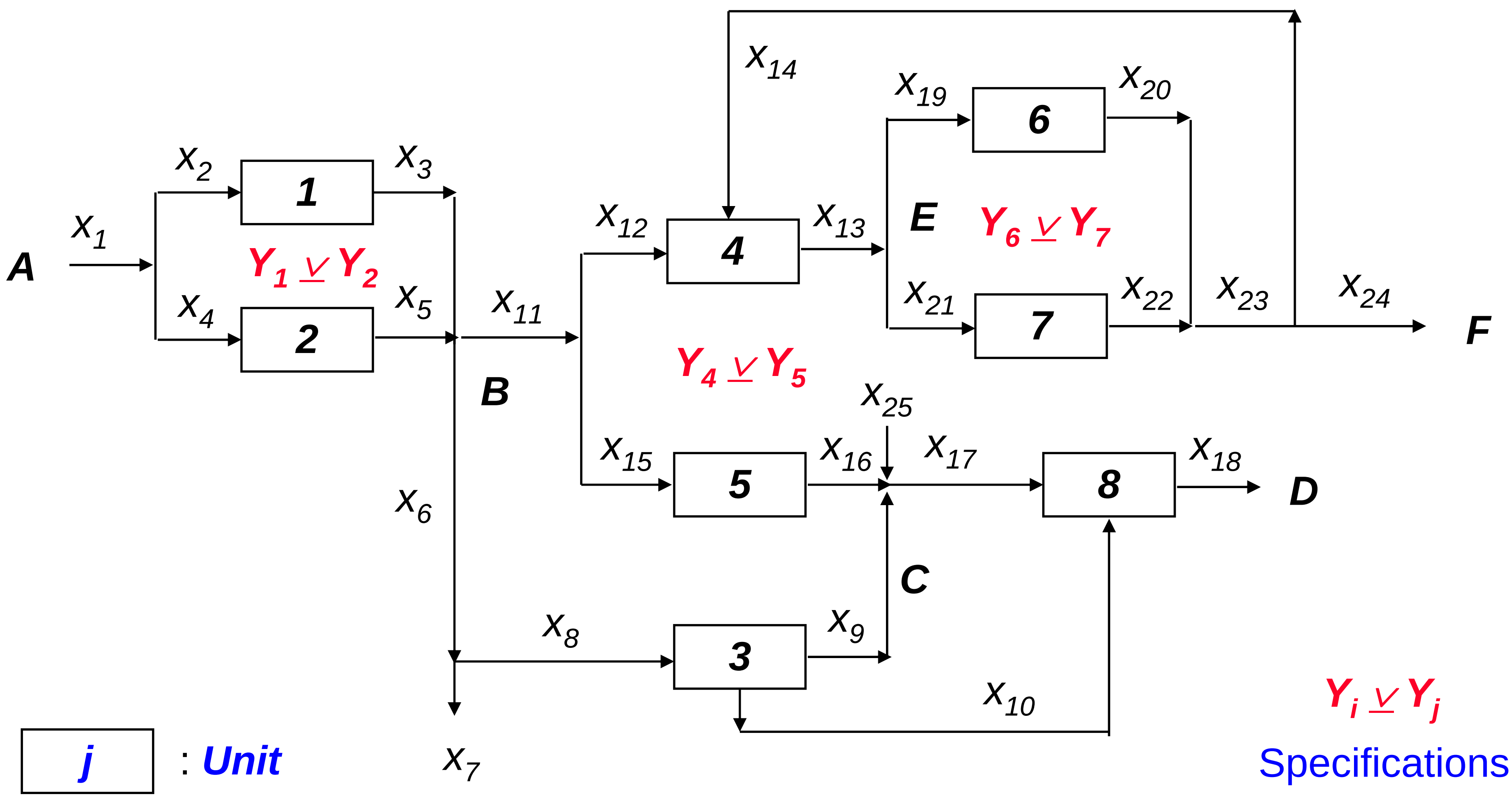
Master problem solved with disjunctive branch and bound or with MILP reformulation

Carnegie Mellon

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

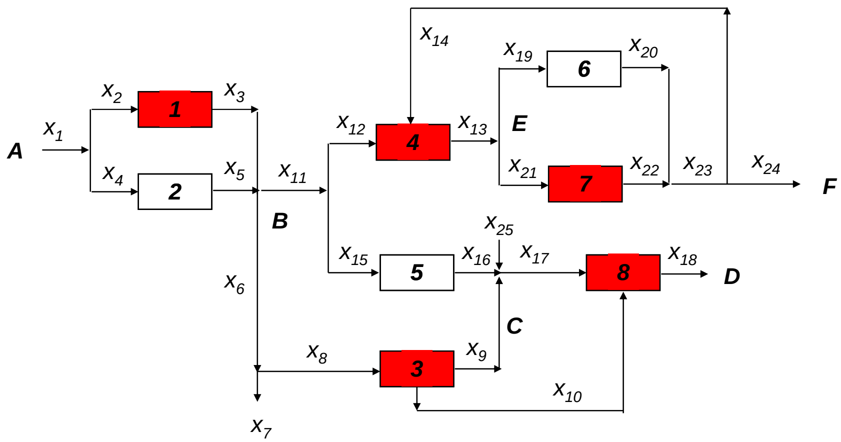
Process Network with Fixed Charges

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

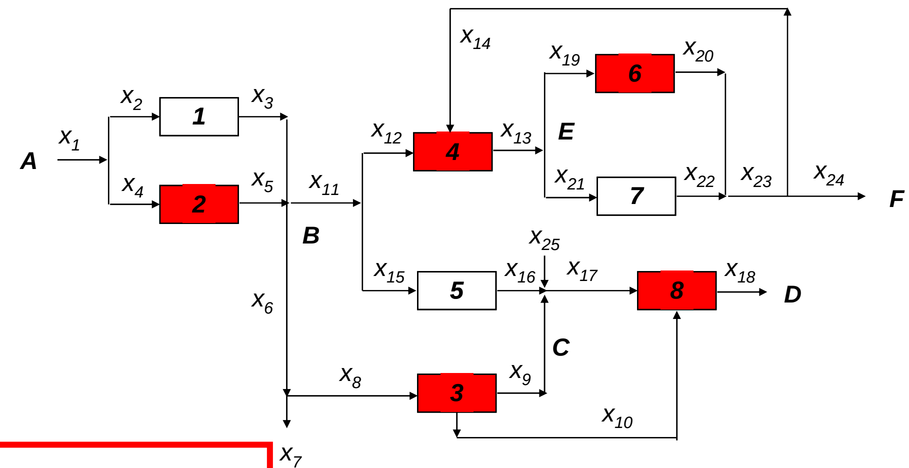


LOGMIP- Logic Based OA

NLP1 = 73.7



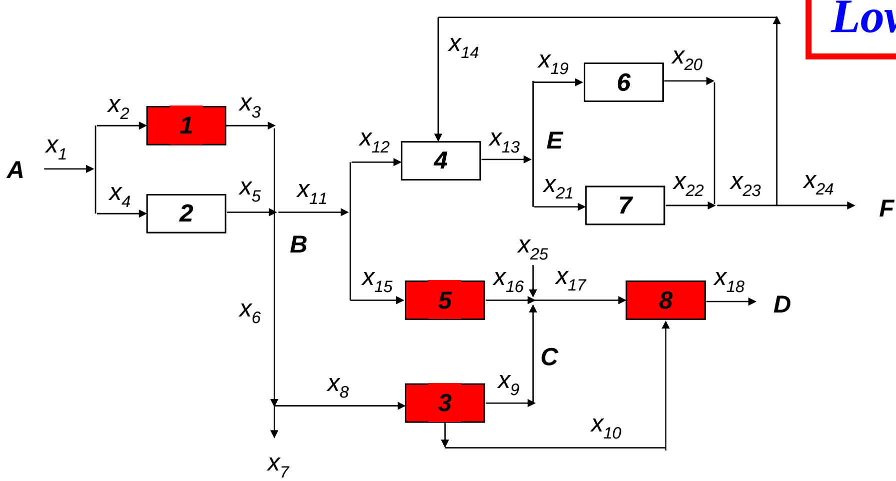
NLP2 = 103.6



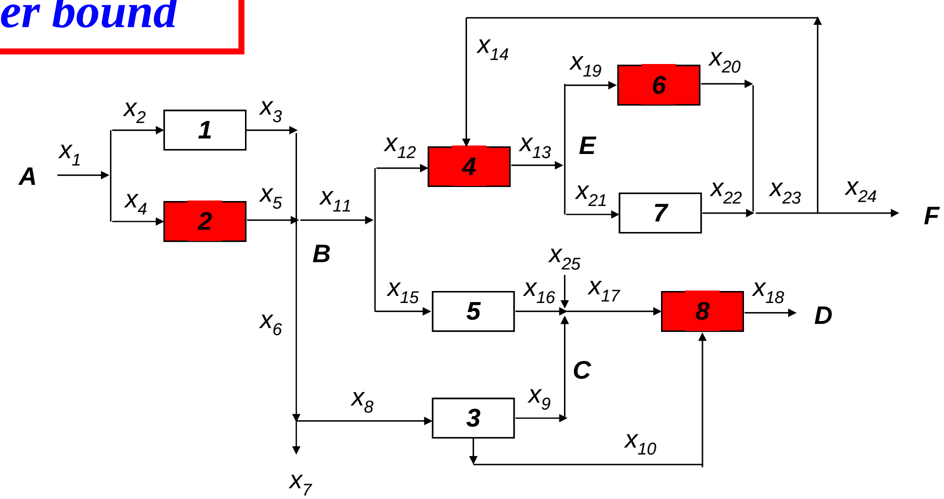
MIP = 67.9

Lower bound

NLP3 = 133.8



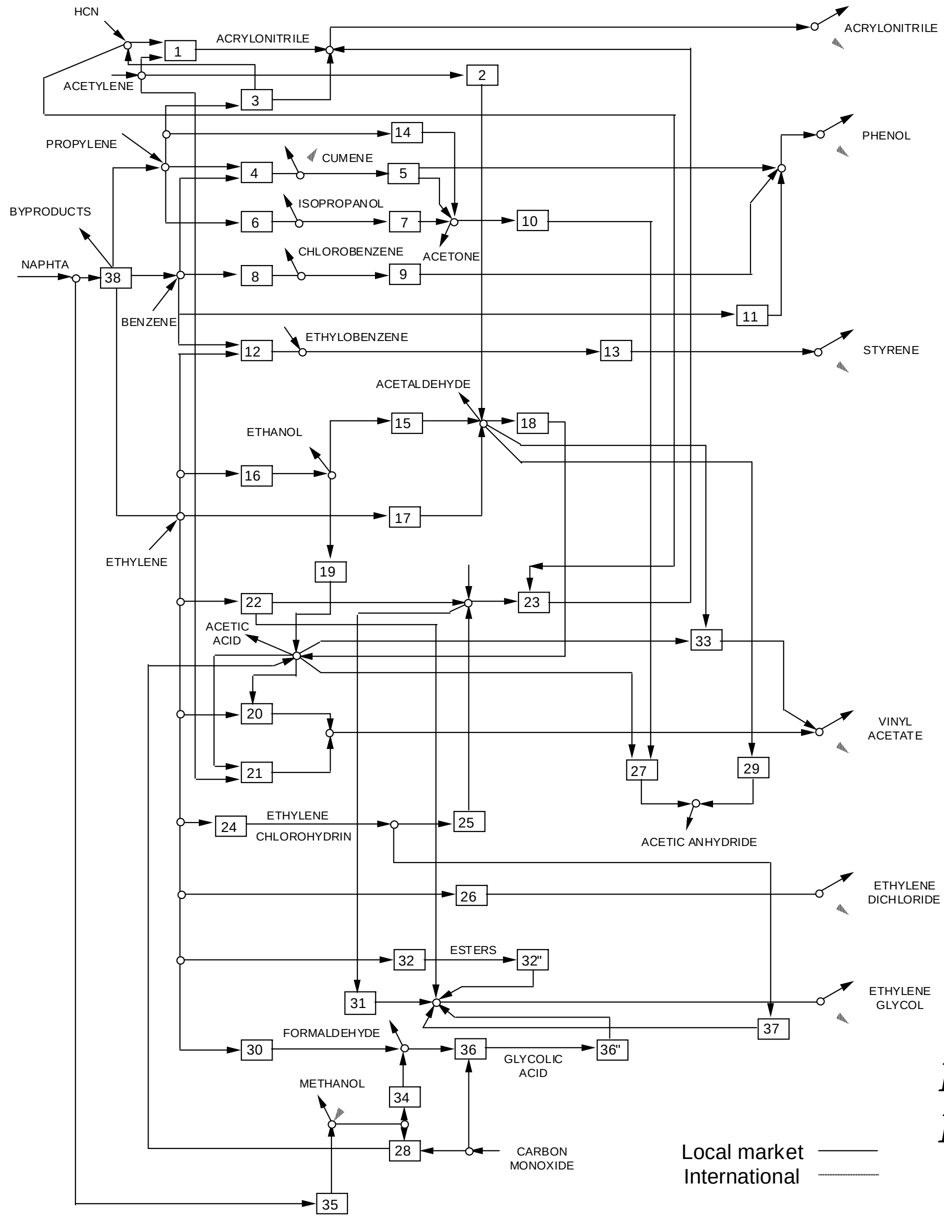
NLP4 = 68.0



Optimum!

Carnegie Mellon

Example: Optimal Production Planning with Contracts



LP Multiperiod Planing Model

38 processes

28 chemicals

10 months

Possibility contracts:

Naphtha, Ethylene, Acetylene

Purchases:

Fixed price

Discount after amount

Bulk discount after amount

Contracts fixed time

Park, Park, Melle, Grossmann (2006)
I&EC Res., 45, 5013-5026

Carnegie Mellon

Multiperiod Production Planning LP Model

Fixed price purchases

$$Max\ PROFIT = \sum_{j\in J}\sum_{t\in T}\psi_{jt}S_{jt} - \sum_{j\in J}\sum_{t\in T}\phi_{jt}P_{jt}$$
$$-\sum_{i\in I}\sum_{j\in JM_i}\sum_{t\in T}\delta_{it}W_{ijt} - \sum_{j\in J}\sum_{t\in T}\xi_{jt}V_{jt} - \sum_{j\in J}\sum_{t\in T}\theta_{jt}SF_{jt}$$

Mass balance process $W_{ijt} = \mu_{ij}W_{ij't} \quad i\in I,\ j\in J_i,\ j'\in JM_i,\ t\in T$

Capacity $W_{ijt} \le Q_{it} \quad i\in I,\ j\in JM_i,\ t\in T$

Mass balance chemicals $V_{j,t-1} + \sum_{i\in O_j} W_{ijt} + P_{jt} = V_{jt} + \sum_{i\in I_j} W_{ijt} + S_{jt} \quad j\in J,\ t\in T$

Shortfalls $SF_{jt} \ge d_{jt}^U - S_{jt} \quad j\in J,\ t\in T$

Purchases $a_{jt}^L \le P_{jt} \le a_{jt}^U$

Sales $d_{jt}^L \le S_{jt} \le d_{jt}^U$

$$\left.\begin{array}{l} a_{jt}^L \le P_{jt} \le a_{jt}^U \\ d_{jt}^L \le S_{jt} \le d_{jt}^U \end{array}\right\} \quad j\in J,\ t\in T$$

Limit inventory $V_{jt} \le V_{jt}^U \quad j\in J,\ t\in T$

$$0 \le SF_{jt} \le SF_{jt}^U \quad j\in J,\ t\in T$$

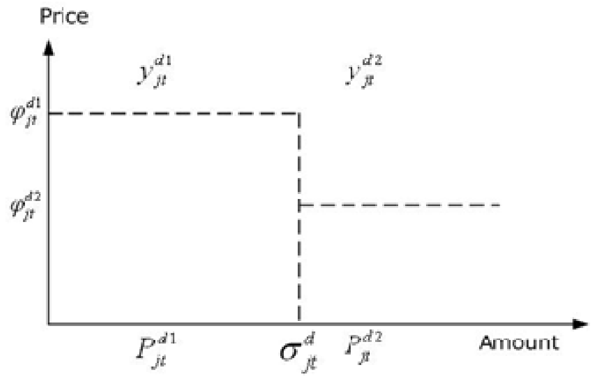
$$S_{jt}, P_{jt}, W_{it}, V_{jt} \ge 0$$

What if prices not fixed but given by contracts?

Carnegie Mellon

. Discount after σ_{jt}^{d} amount.

Price drops after amount σ



Disjunctive model

$$COST_{jt}^{d} = \varphi_{jt}^{d1} P_{jt}^{d1} + \varphi_{jt}^{d2} P_{jt}^{d2} \quad j \in JR, t \in T$$

$$\begin{bmatrix} y_{jt}^{d1} \\ 0 \le P_{jt}^{d1} \le \sigma_{jt}^{d} \\ P_{jt}^{d2} = 0 \end{bmatrix} \vee \begin{bmatrix} y_{jt}^{d2} \\ P_{jt}^{d1} = \sigma_{jt}^{d} \\ P_{jt}^{d2} \ge 0 \end{bmatrix} \quad j \in JR, t \in T$$

$$P_{jt}^{d} = P_{jt}^{d1} + P_{jt}^{d2} \quad j \in JR, t \in T$$

MILP model (Convex hull)

$$COST_{jt}^{d} = \varphi_{jt}^{d1} P_{jt}^{d1} + \varphi_{jt}^{d2} P_{jt}^{d2} \quad j \in JR, t \in T$$

$$P_{jt}^{d} = P_{jt}^{d1} + P_{jt}^{d2} \quad j \in JR, t \in T$$

$$P_{jt}^{d1} = P_{jt}^{d11} + P_{jt}^{d12} \quad j \in JR, t \in T$$

$$0 \le P_{jt}^{d11} \le y_{jt}^{d1} \sigma_{jt}^{d} \quad j \in JR, t \in T$$

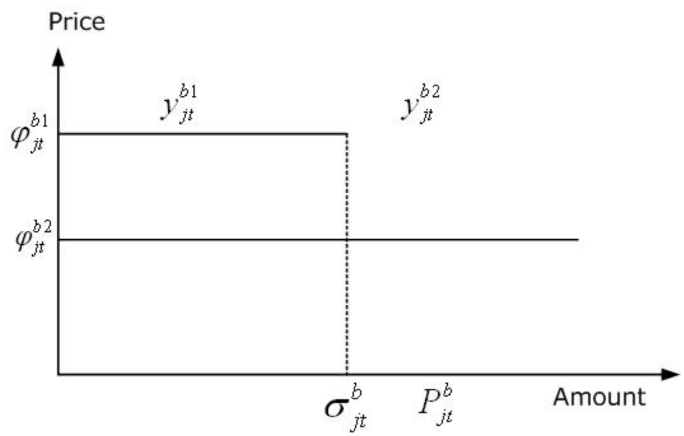
$$P_{jt}^{d12} = y_{jt}^{d2} \sigma_{jt}^{d} \quad j \in JR, t \in T$$

$$0 \le P_{jt}^{d2} \le y_{jt}^{d2} U_{jt}^{d} \quad j \in JR, t \in T$$

$$y_{jt}^{d1} + y_{jt}^{d2} = y_{jt}^{d} \quad j \in JR, t \in T$$

$$y_{jt}^{d1}, y_{jt}^{d2} \in \{0,1\}$$

Bulk discount



Price for total amount drops after amount σ

Disjunctive Model

$$\begin{bmatrix} y_{jt}^{b1} \\ COST_{jt}^{b} = \varphi_{jt}^{b1} P_{jt}^{b} \\ 0 \le P_{jt}^{b} \le \sigma_{jt}^{b} \end{bmatrix} \vee \begin{bmatrix} y_{jt}^{b2} \\ COST_{jt}^{b} = \varphi_{jt}^{b2} P_{jt}^{b} \\ P_{jt}^{b} \ge \sigma_{jt}^{b} \end{bmatrix} \quad j \in JR, t \in T$$

MILP Model (Convex Hull)

$$COST_{jt}^{b} = \varphi_{jt}^{b1} P_{jt}^{b1} + \varphi_{jt}^{b2} P_{jt}^{b2} \quad j \in JR, t \in T$$

$$P_{jt}^{b} = P_{jt}^{b1} + P_{jt}^{b2} \quad j \in JR, t \in T$$

$$0 \le P_{jt}^{b1} \le \sigma_{jt}^{b} y_{jt}^{b1} \quad j \in JR, t \in T$$

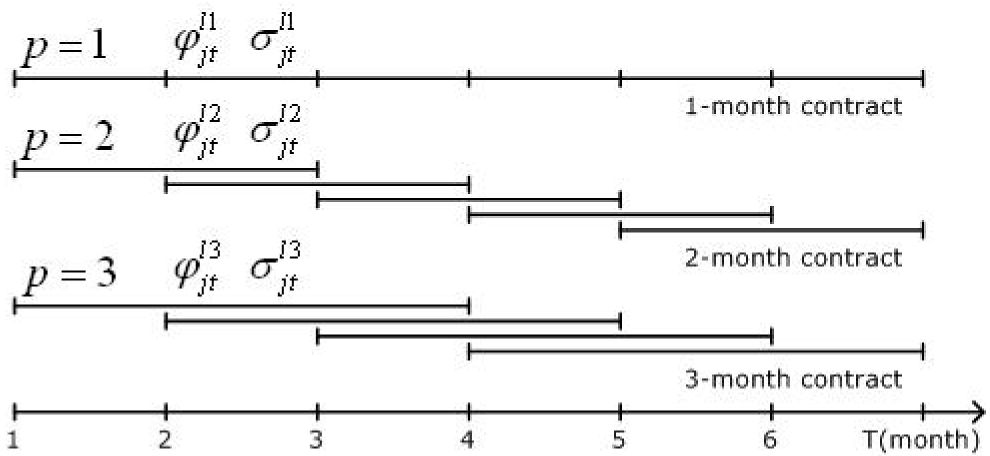
$$\sigma_{jt}^{b} y_{jt}^{b2} \le P_{jt}^{b2} \le U_{jt}^{b} y_{jt}^{b2} \quad j \in JR, t \in T$$

$$y_{jt}^{b1} + y_{jt}^{b2} = y_{jt}^{b} \quad j \in JR, t \in T$$

$$y_{jt}^{b1}, y_{jt}^{b2} \in \{0,1\}$$

Fixed-duration contracts

Price gradually drops if amount σ purchased fixed number of months



Disjunctive Model

$$\begin{bmatrix} y_{jt}^{l1} \\ COST_{jt}^{l} = \varphi_{jt}^{l1} P_{jt}^{l} \\ P_{jt}^{l} \ge \sigma_{jt}^{l1} \end{bmatrix} \vee \begin{bmatrix} y_{jt}^{l2} \\ COST_{jt}^{l} = \varphi_{jt}^{l2} P_{jt}^{l} \\ COST_{j,t+1}^{l} = \varphi_{jt}^{l2} P_{j,t+1}^{l} \\ P_{jt}^{l} \ge \sigma_{jt}^{l2} \\ P_{j,t+1}^{l} \ge \sigma_{jt}^{l2} \end{bmatrix} \vee \begin{bmatrix} y_{jt}^{l3} \\ COST_{jt}^{l} = \varphi_{jt}^{l3} P_{jt}^{l} \\ COST_{j,t+1}^{l} = \varphi_{jt}^{l3} P_{j,t+1}^{l} \\ COST_{j,t+2}^{l} = \varphi_{jt}^{l3} P_{j,t+2}^{l} \\ P_{jt}^{l} \ge \sigma_{jt}^{l3} \\ P_{j,t+1}^{l} \ge \sigma_{jt}^{l3} \\ P_{j,t+2}^{l} \ge \sigma_{jt}^{l3} \end{bmatrix} \quad j \in JR, t \in T$$

MILP Model

$$COST_{jt}^{l} = \sum_{p \in LC} \sum_{\tau \in T_t^p} \varphi_{j\tau}^{lp} P_{jt\tau}^{lp} \qquad j \in JR, t \in T$$

$$P_{jt}^{l} = \sum_{p \in LC} \sum_{\tau \in T_t^p} P_{jt\tau}^{lp} \qquad j \in JR, t \in T$$

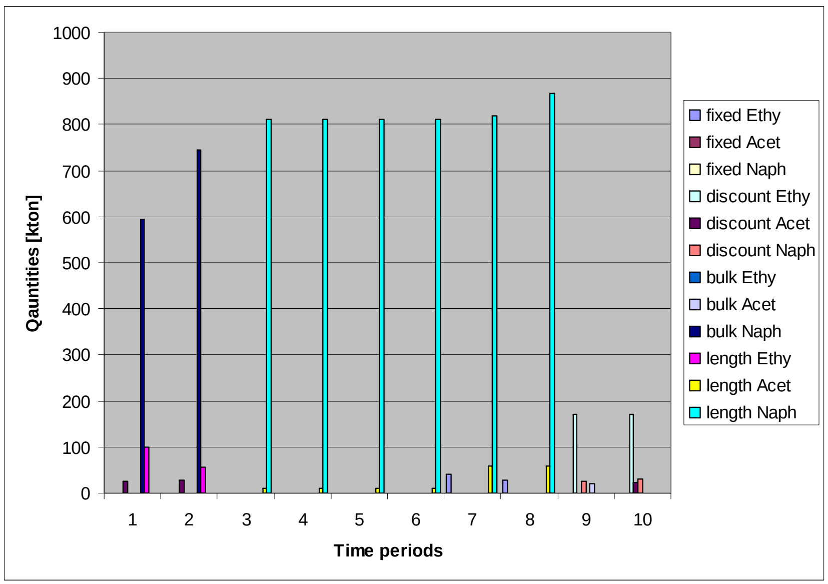
$$\sigma_{j\tau}^{lp} y_{j\tau}^{lp} \le P_{jt\tau}^{lp} \le U_j^l y_{j\tau}^{lp} \qquad j \in JR, t \in T_\tau^p \subset T, \tau \in T_t^p \subset T, p \in LC$$

$$\sum_{p \in LC} y_{j\tau}^{lp} \le y_{j\tau}^{l} \qquad j \in JR, \tau \in T$$

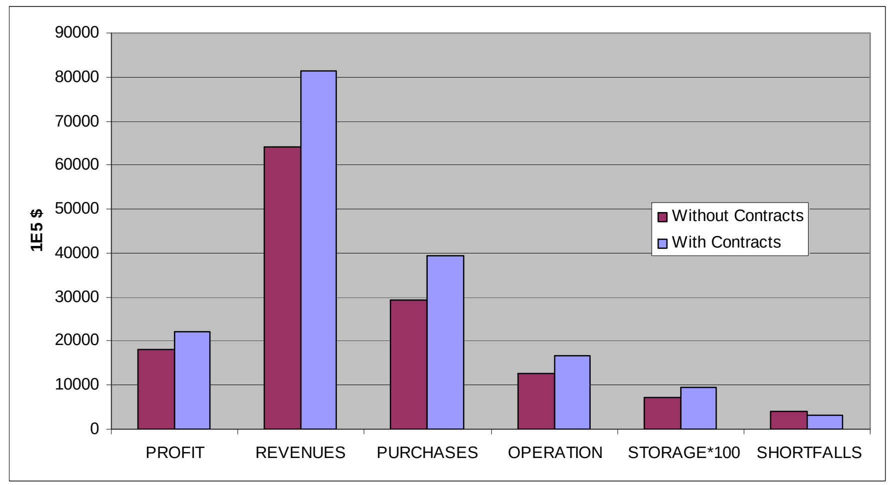
$$y_{j\tau}^{lp} \in \{0,1\}, LC = \{1,2,3\}$$

Computational Results

Case	0-1 variables	Cont. variables	Constraints	CPU time [s]	Time periods	Profit [10^3 $]
1	0	12,606	13,416	0.18	10	18,085.95
2	6,160	40,606	46,002	0.95	10	22,073.06



Purchases raw materials



Comparison without/with contracts

Computational Results

```
MODEL STATISTICS

BLOCKS OF EQUATIONS          47     SINGLE EQUATIONS       46,002
BLOCKS OF VARIABLES          30     SINGLE VARIABLES       40,606
NON ZERO ELEMENTS        85,033     DISCRETE VARIABLES      6,160

S O L V E      S U M M A R Y

     MODEL   MULT                 OBJECTIVE  NPV
     TYPE    MIP                  DIRECTION  MAXIMIZE
     SOLVER  CPLEX                FROM LINE  878

**** SOLVER STATUS     1 NORMAL COMPLETION
**** MODEL STATUS      1 OPTIMAL
**** OBJECTIVE VALUE           22073.0600

 RESOURCE USAGE, LIMIT          0.470     10000.000
 ITERATION COUNT, LIMIT      1437       1000000
```

ILOG CPLEX Dec 1, 2008 22.9.2 LNX 7311.8080 LX3 x86/Linux
Cplex 11.2.0, GAMS Link 34

MIP Presolve eliminated 44425 rows and 38571 columns.
Reduced MIP has 909 rows, 1367 columns, and 3267 nonzeros.
Reduced MIP has 270 binaries, 0 generals, 0 SOSs, and 0 indicators.

Implied bound cuts applied: 3
Flow cuts applied: 40
Gomory fractional cuts applied: 15
MIP Solution: 22073.060039 (959 iterations, 21 nodes)

Linear Generalized Disjunctive Programming
LGDP Model

Sawaya N.W. and Grossmann I.E. (2008)

$$Min\ Z = \sum_{k \in K} c_k + d^T x$$ **Objective function**

$$s.t. \qquad Bx \geq b$$ **Common constraints**

$$\bigvee_{j \in J_k} \begin{bmatrix} Y_{jk} \\ A^{jk} x \geq a^{jk} \\ c_k = \gamma_{jk} \end{bmatrix} \qquad k \in K$$ **Disjunctive constraints**

$$\underline{\vee}_{j \in J_k} Y_{jk} \qquad k \in K$$

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

$$x^L \leq x \leq x^U$$ ***Continuous variables***

$$Y_{jk} \in \{True, False\} \qquad j \in J_k, k \in K$$ ***Boolean variables***

$$c_k \in \mathbb{R}^1 \qquad k \in K$$

Logical OR operator

Can we obtain stronger relaxations?

How to deal with Boolean and logic constraints in Disjunctive Programming?

Reformulating LGDP into Disjunctive Programming Formulation

$$
\begin{aligned}
& Min\ Z = \sum_{k \in K} c_k + d^T x \\
& s.t. \quad Bx \ge b \\
& \bigvee_{j \in J_k} \begin{bmatrix} Y_{jk} \\ A^{jk} x \ge a^{jk} \\ c_k = \gamma_{jk} \end{bmatrix} && k \in K \\
& \underset{j \in J_k}{\veebar} Y_{jk} && k \in K \\
& \Omega(Y) = True \\
& x^L \le x \le x^U \\
& Y_{jk} \in \{True, False\} && j \in J_k, k \in K \\
& c_k \in \mathbb{R}^1 && k \in K
\end{aligned}
$$

$$
\begin{aligned}
& Min\ Z = \sum_{k \in K} c_k + d^T x \\
& s.t. \quad Bx \ge b \\
& \bigvee_{j \in J_k} \begin{bmatrix} \lambda_{jk} = 1 \\ A^{jk} x \ge a^{jk} \\ c_k = \gamma_{jk} \end{bmatrix} && k \in K \\
& \sum_{j \in J_k} \lambda_{jk} = 1 && k \in K \\
& H\lambda \ge h \\
& x^L \le x \le x^U \\
& 0 \le \lambda_{jk} \le 1 && j \in J_k, k \in K \\
& c_k \in \mathbb{R}^1 && k \in K
\end{aligned}
$$

LGDP

LDP ***=> Integrality λ guaranteed***

Proposition. LGDP and LDP have equivalent solutions.

Equivalent Forms in DP Through Basic Steps

There are many forms between CNF and DNF that are equivalent

Regular Form (RF): form represented by intersection of unions of polyhedra

Thus the RF is:

$$F = \bigcap_{t \in T} S_t$$

where for $t \in T, \quad S_t = \bigcup_{i \in Q_t} P_i, \quad P_i$ a polyhedron, $i \in Q_t$.

Proposition 1 (Theorem 2.1 in Balas (1979)). *Let F be a disjunctive set in RF. Then F can be brought to DNF by* $\underline{|T|-1}$ *<u>recursive applications of the following basic steps,</u> which preserve regularity:*

For some $r, s \in T, r \neq s,$ *bring* $S_r \cap S_s$ *to DNF, by replacing it with:*

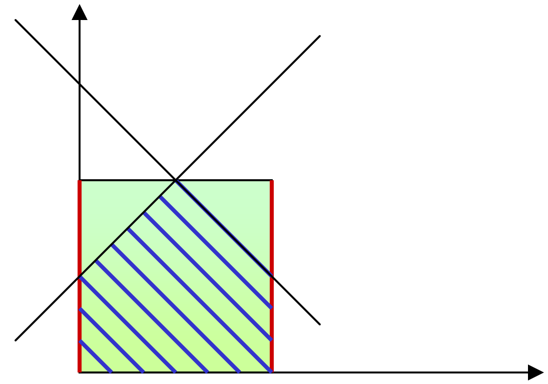
$$S_{rs} = \bigcup_{\substack{i \in Q_r \\ t \in Q_s}} (P_i \cap P_t).$$

$\Rightarrow$ ***as basic steps are performed tighter relaxations are obtained***

Illustrative Example: Hierarchy of Relaxations

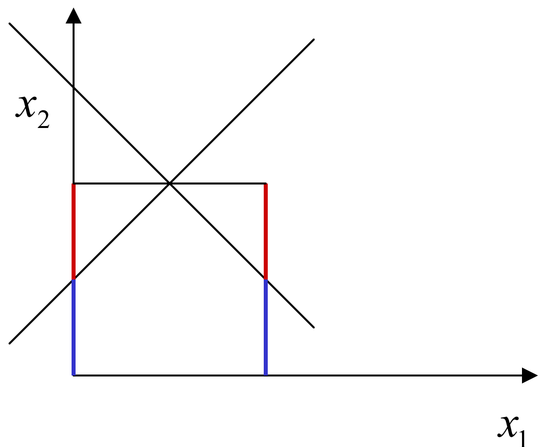
$$x_1 - x_2 + 0.5 \geq 0$$
$$-x_1 - x_2 + 1 \geq 0$$
$$\begin{bmatrix} x_1 = 0 \\ 0 \leq x_2 \leq 1 \end{bmatrix} \vee \begin{bmatrix} x_1 = 1 \\ 0 \leq x_2 \leq 1 \end{bmatrix}$$

Convex Hull of disjunction



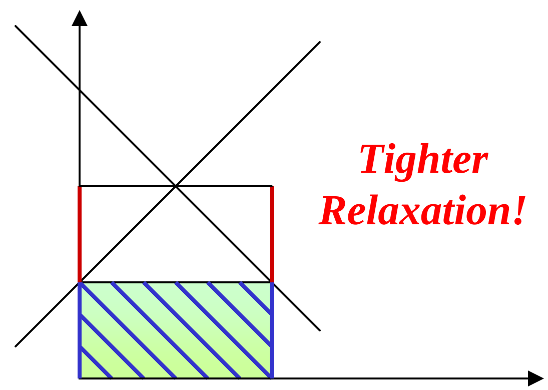
LP Relaxation

Application of 2 Basic Steps



$$\begin{bmatrix} x_1 - x_2 + 0.5 \geq 0 \\ -x_1 - x_2 + 1 \geq 0 \\ x_1 = 0 \\ 0 \leq x_2 \leq 1 \end{bmatrix} \vee \begin{bmatrix} x_1 - x_2 + 0.5 \geq 0 \\ -x_1 - x_2 + 1 \geq 0 \\ x_1 = 1 \\ 0 \leq x_2 \leq 1 \end{bmatrix}$$

Convex Hull of disjunction



Tighter Relaxation!

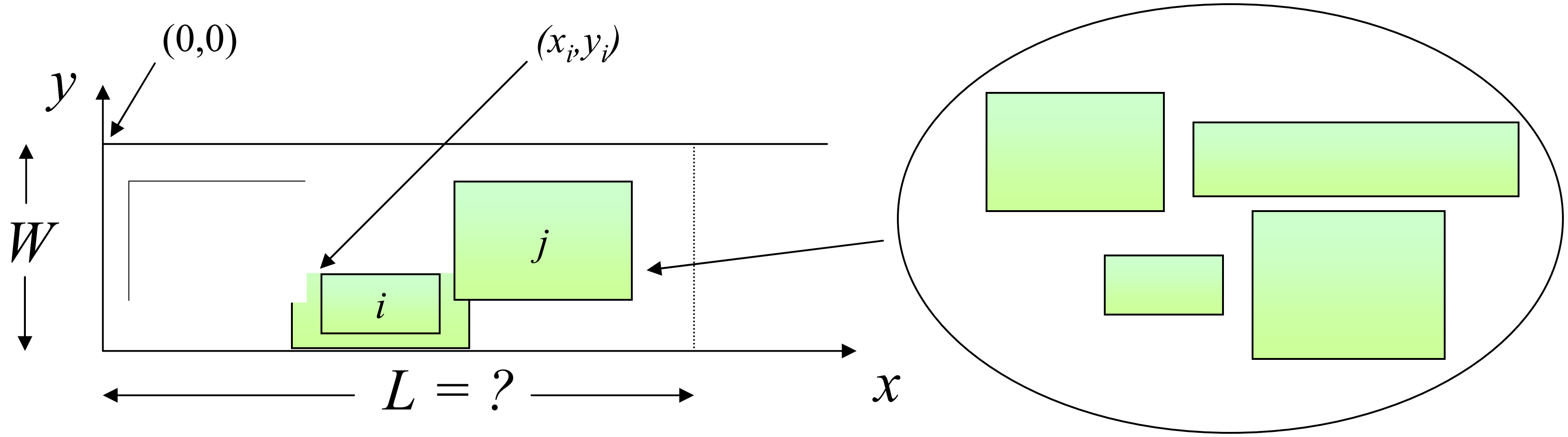
Numerical Example: Strip-packing problem

Problem statement: *Hifi (1998)*

Given a set of small rectangles with width H_i and length L_i.

Large rectangular strip of fixed width W and unknown length L.

Objective is to fit small rectangles onto strip without overlap and rotation while minimizing length L of the strip.



Set of small rectangles

GDP/DP Model for Strip-packing problem

$$
\begin{array}{lll}
Min & lt & \\
s.t. & lt \geq x_i + L_i & \forall i \in N
\end{array}
$$

$$
\begin{bmatrix} Y^1_{ij} \\ x_i + L_i \leq x_j \end{bmatrix} \vee \begin{bmatrix} Y^2_{ij} \\ x_j + L_j \leq x_i \end{bmatrix} \vee \begin{bmatrix} Y^3_{ij} \\ y_i - H_i \geq y_j \end{bmatrix} \vee \begin{bmatrix} Y^4_{ij} \\ y_j - H_j \geq y_i \end{bmatrix} \quad \forall i, j \in N, i < j
$$

$$
\begin{array}{ll}
x_i \leq UB_i - L_i & \forall i \in N \\
H_i \leq y_i \leq W & \forall i \in N
\end{array}
$$

$$
lt, x_i, y_i \in \mathbb{R}^1_+ \,,\; Y^1_{ij}, Y^2_{ij}, Y^3_{ij}, Y^4_{ij} \in \{True, False\} \qquad \forall i, j \in N, i < j
$$

Objective function
Minimize length

Disjunctive constraints
No overlap between rectangles

Bounds on variables

25 Rectangle Problem Optimal solution= 31

Original CH		Strengthened
1,112 0-1 variables	=>	1,112 0-1 variables
4,940 cont vars		5,783 cont vars
7,526 constraints		8,232 constraints
LP relaxation = 9		LP relaxation = 27!

31 Rectangle Problem Optimal solution= 38

Original CH		Strengthened
2,256 0-1 variables	=>	2,256 0-1 variables
9,716 cont vars		11,452 cont vars
14,911 constraints		15,624 constraints
LP relaxation = 10.64		LP relaxation = 33!

Nonconvex GDP

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

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

Disjunctions

OR operator → $$\bigvee_{j \in J_k} \begin{bmatrix} Y_{jk} \\ g_{jk}(x) \le 0 \\ c_k = \gamma_{jk} \end{bmatrix} \quad k \in K$$

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

$$x \in R^n, \; c_k \in R^1$$

$$Y_{jk} \in \{ true, false \}$$

f, g and r: nonconvex

Carnegie Mellon

Convex Underestimator GDP (R)

- **Introducing convex underestimators**

$$\min Z = \sum_k c_k + \overline{f}(x)$$

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

$$\bigvee_{j \in J_k} \begin{bmatrix} Y_{jk} \\ \overline{g}_{jk}(x) \le 0 \\ c_k = \gamma_{jk} \end{bmatrix} \quad k \in K$$

$$\Omega(Y) = true$$

$$x \in R^n, \; c_k \in R^1$$

$$Y_{jk} \in \{true, false\}$$

$$\overline{f}, \overline{r} \text{ and } \overline{g} : convex$$

Convex underestimators

Bilinear: Linear

McCormick (1976), Al-Khayyal (1992)

Linear fractional: Convex nonlinear

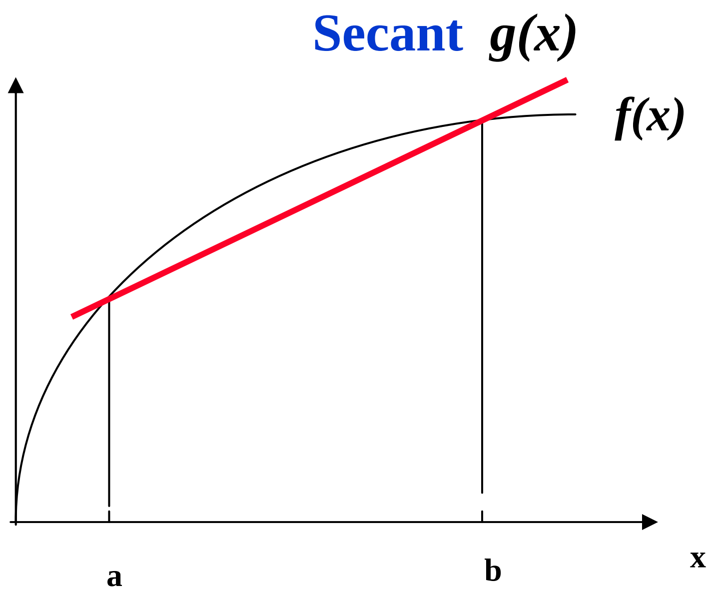
Quesada and Grossmann (1995)

Concave separable: Linear secant

◆**Problem (R) yields a valid lower bound to Problem (GDP)**

Convex envelopes

Concave function



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

Bilinear

$$w = xy$$

$$x^L \le x \le x^U \qquad y^L \le y \le y^U$$

McCormick convex envelopes

$$w \ge x^L y + y^L x - x^L y^L$$

$$w \ge x^U y + y^U x - x^U y^U$$

$$w \le x^L y + y^U x - x^L y^U$$

$$w \le x^U y + y^L x - x^U y^L$$

For other convex envelopes/underestimators see:

Tawarmalani, M. and N. V. Sahinidis, *Convexification and Global Optimization in Continuous and Mixed-Integer Nonlinear Programming: Theory, Algorithms, Software, and Applications*, Vol. 65, *Nonconvex Optimization And Its Applications* series, Kluwer Academic Publishers, Dordrecht, 2002

Proposed framework to obtain stronger relaxations for nonconvex GDP

(Bilinear and Concave GDP)

(Ruiz & Grossmann, 2008)

The **framework** consists of **two main phases**:

1- Generate a **valid Linear Generalized Disjunctive Program** (***LGDP)*** *relaxation for the nonconvex GDP problem* (e.g. bilinear and concave).

2- Strengthen the *continuous relaxation of the* ***LGDP*** obtained in phase 1 by **applying a set of basic steps**

Relaxation Results

	Global Optimum	Lower Bound (Lee & Grossmann Relaxation)	Lower Bound (Proposed Relaxation)	DNF Lower Bound
Example 1	-1.01	-1.28	-1.10	-1.10
Example 2	6.31	5.65	6.08	6.08
Example 3	114384.78	91671.18	94925.77	97858.86
Example 4	1214.87	400.66	431.90	431.90
Example 5	-4640	-5515	-5468	-5241

Remarks

-Proposed methodology leads to improvements in the lower bounds.

-Often, it is not necessary to reach the DNF form to have good lower bounds. Note that examples 1, 2 and 4 show that the lower bound is the same as the lower bound of the DNF

-The lower bound of the DNF is the best lower bound attainable for a given LGDP.

Conclusions

GDP modeling framework

- Provides a logic-based framework for *linear and nonlinear* discrete-continuous optimization
- big-M and convex hull alternative formulations *different relaxations*
- Solution methods: *reformulation, branch and bound, decomposition*
- Numerical example of *planning problem with contracts* has shown the impact of the convex hull formulation *(with help of CPLEX)*

Unified Linear GDP with Disjunctive Programming

- Developed *DP equivalent* formulation for GDP
- Numerical results have shown *great improvement in lower bound for strip packing problem*

Nonconvex GDPs

- *Tighter lower bounds* can be obtained in bilinear and concave problems by applying basic steps