Chemical ENGINEERING
Carnegie Mellon

ingar

Computational Experience with Logmip Solving Linear and Nonlinear Disjunctive Programming Problems

Aldo Vecchietti, INGAR – Instituto de Desarrollo y Diseño
E-mail: aldovec@ceride.gov.ar
Ignacio E. Grossmann, Carnegie Mellon University,
E-mail: grossmann@cmu.edu
Alexander Meeraus, GAMS Development
E-mail: alex@gams.com

GAMS

Chemical ENGINEERING
Carnegie Mellon

LogMIP
Logical Mixed Integer Programming

ingar

http://www.ceride.gov.ar/logmip/

Motivation

- Modeling framework for facilitating the formulation of models that can be expressed in terms of algebraic, disjunctive and symbolic logic expressions

- Language compiler within GAMS for disjunctions expressions, logic constraints and logic propositions.

- Solution algorithms and techniques for solving linear /nonlinear disjunctive programming problems.

GAMS

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 $\quad 0 \le y_{jk} \le 1$

Chemical ENGINEERING
Carnegie Mellon

Convex Hull Formulation

ingar

- 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

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

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

=> **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)

GAMS

Convex Relaxation Problem (CRP)

CRP:

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

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

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

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

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

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

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).*
- **Remark**: *MINLP reformulation by setting* $\lambda_{jk} = 0,1$

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

Logic based methods

Branch and bound
(Lee & Grossmann, 2000)

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

Reformulation MI(N)LP
Outer-Approximation
Generalized Benders
Extended Cutting Plane

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

LogMIP

GAMS

LogMIP Language Syntax

Carnegie Mellon

Properties of LogMIP language

- Simple syntax
- Semantic close to disjunction expression to model blocks, exclusive between them
- The syntax must be known for a regular user
- The syntax construction must allow the definition of embeded disjunctions

Notation

Symbol	Meaning
< >	The phrase enclosed is a syntactic rule
Word	Token
[]	Optional expression
{}	Expression that can be repeated
( )	Expression enclosed can be grouped
::=	Define like
|	OR

Rules describing LogMIP syntax

<LOGMIP Model> ::=
<Disjunction Declaration> {**;** <LogMIP Model>}
| <Disjunction Definition> {**;** <LogMIP Model>}
;

<Disjunction Declaration> ::=
disjunction identifier {**, identifier**} **;**
;

<Disjunction Definition> ::=
disjunction identifier is <If Sentence>
;

<If Sentence> ::=
if <condition> **then** <components> **else** <components> **end if ;**
| **if** <condition> **then** <components> {**elsif** <condition> **then** <components>} **end if**
;

<components> ::=
entity {**; entity**} [**;** <If Sentence>]
;

GAMS

Examples

$$\begin{bmatrix} True \\ Constraint\ \ 1 \end{bmatrix} \vee \begin{bmatrix} Fal\ se \\ Constraint\ \ 2 \end{bmatrix}$$

Modeling two terms disjunction

IF (condition$_1$) THEN

Constraints to be considered when condition$_1$ is True

ELSE

Constraints to be considered when condition$_1$ is False

END IF

$$\begin{bmatrix} 1 \\ Constraints\ 1 \end{bmatrix} \vee \begin{bmatrix} 2 \\ Constraints\ 2 \end{bmatrix} \vee .. \vee \begin{bmatrix} N \\ Constraints\ N \end{bmatrix}$$

Modeling a multi-term disjunction

IF (condition$_1$) THEN

Constraints to be considered when condition$_1$ is True

ELSIF (condition2) THEN

Constraints to be considered when condition$_1$ is True

ELSIF (condition3) THEN

...

ELSIF (conditionN) THEN

Constraints to be considered when condition$_1$ is True

END IF

Small Example

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

s.a.:

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

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

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

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

$$0 \leq x_1 \leq 5,\ 0 \leq x_2 \leq 5,\ c \geq 0$$

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

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;
```

LogMIP

Recent developments

-Disjunctions specified with IF Then ELSE statements

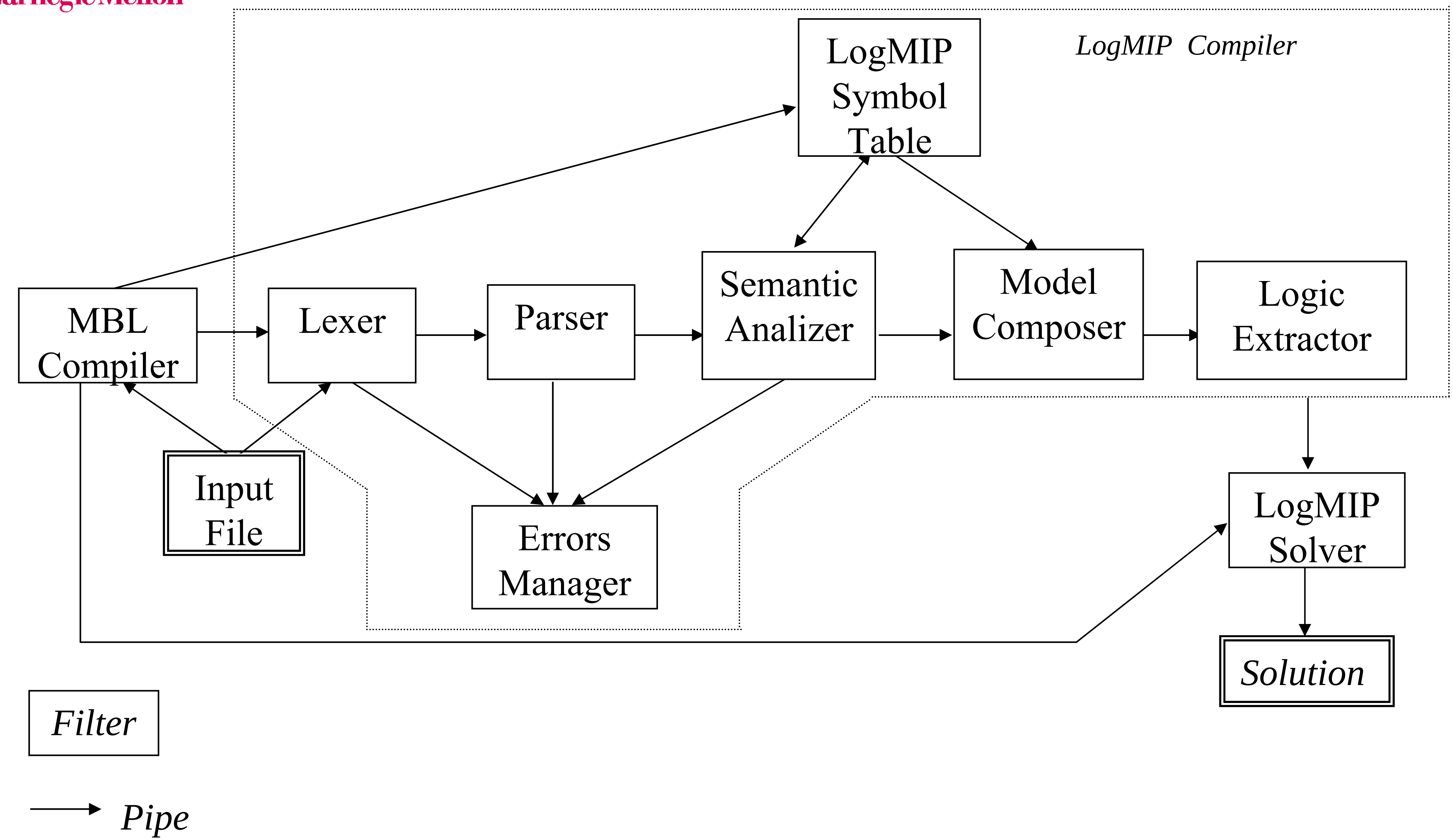
```
DISJUNCTION D1(I,K,J);
D1(I,K,J)
        with (L(I,K,J)) IS
IF Y(I,K,J) THEN
        NOCLASH1(I,K,J);
ELSE
        NOCLASH2(I,K,J);
ENDIF;
```

-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

LogMIP Compiler Architecture



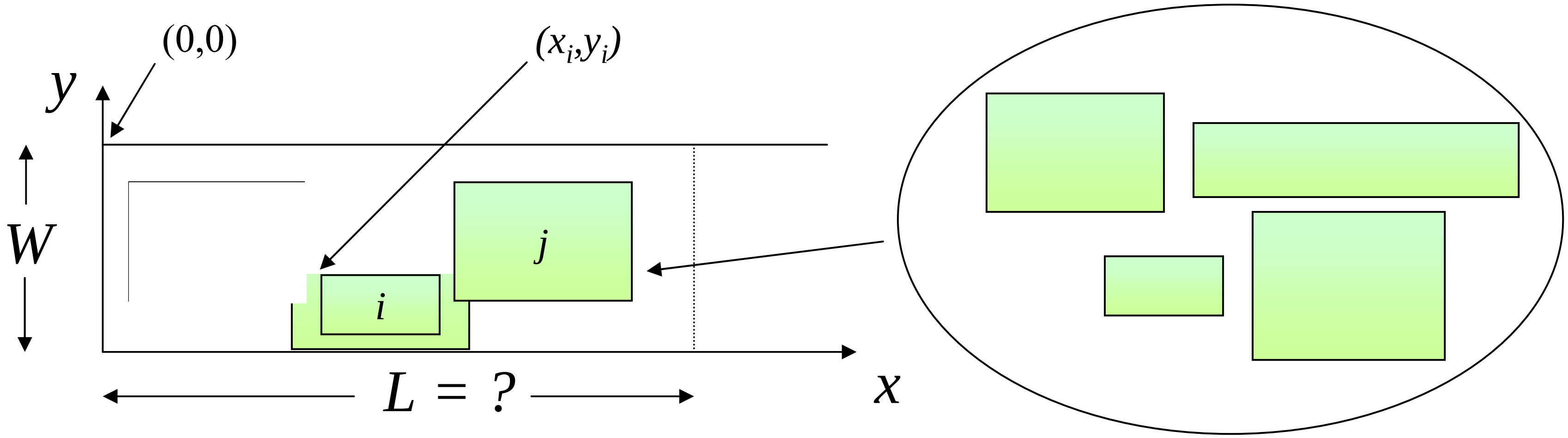
Strip-packing Problem

Chemical ENGINEERING Carnegie Mellon

ingar

Problem statement: *Hifi (1998)*

- **Given a set of small rectangles with width w_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

GAMS

GDP Model For Strip-packing Problem

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

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

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

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

Objective function
Minimize length

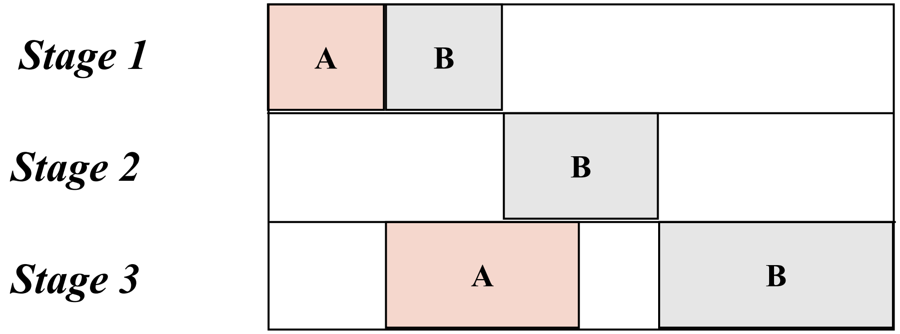
Disjunctive constraints
No overlap between rectangles

Bounds on variables

Zero-wait Job-shop Scheduling Problem

Problem Statement: ***Raman R. & Grossmann I.E. (1994)***

- **Set of jobs $i \in I$ must be processed sequentially on a set of consecutive stages $j \in J$.**
  - **All jobs can be sequenced on a subset of stages $j \in J(i)$.**
  - **Zero-wait transfer is assumed between stages.**
- **Objective is to obtain a schedule that minimizes makespan.**



Chemical ENGINEERING
Carnegie Mellon

ingar

GDP Model For Zero-wait Job-shop Scheduling Problem

$$\begin{aligned}
&Min \quad ms \\
&s.t. \quad ms \geq t_i + \sum_{\forall j \in J(i)} TAU_{ij} \qquad \forall i \in I \\
&\begin{bmatrix} Y^1_{ik} \\ t_i + \sum_{\substack{\forall m \in J(i) \\ m \leq j}} TAU_{im} \leq t_k + \sum_{\substack{\forall m \in J(k) \\ m < j}} TAU_{km} \end{bmatrix} \vee \begin{bmatrix} Y^2_{ik} \\ t_k + \sum_{\substack{\forall m \in J(k) \\ m \leq j}} TAU_{km} \leq t_i + \sum_{\substack{\forall m \in J(i) \\ m < j}} TAU_{im} \end{bmatrix} \quad \forall j \in C_{ik}, \forall i,k \in I, i<k \\
&ms, t_i \in \mathbf{R}^1_+ ,\ Y^1_{ik}, Y^2_{ik} \in \{True, False\} \qquad \forall i,k \in I, i<k
\end{aligned}$$

Objective function
Minimize makespan

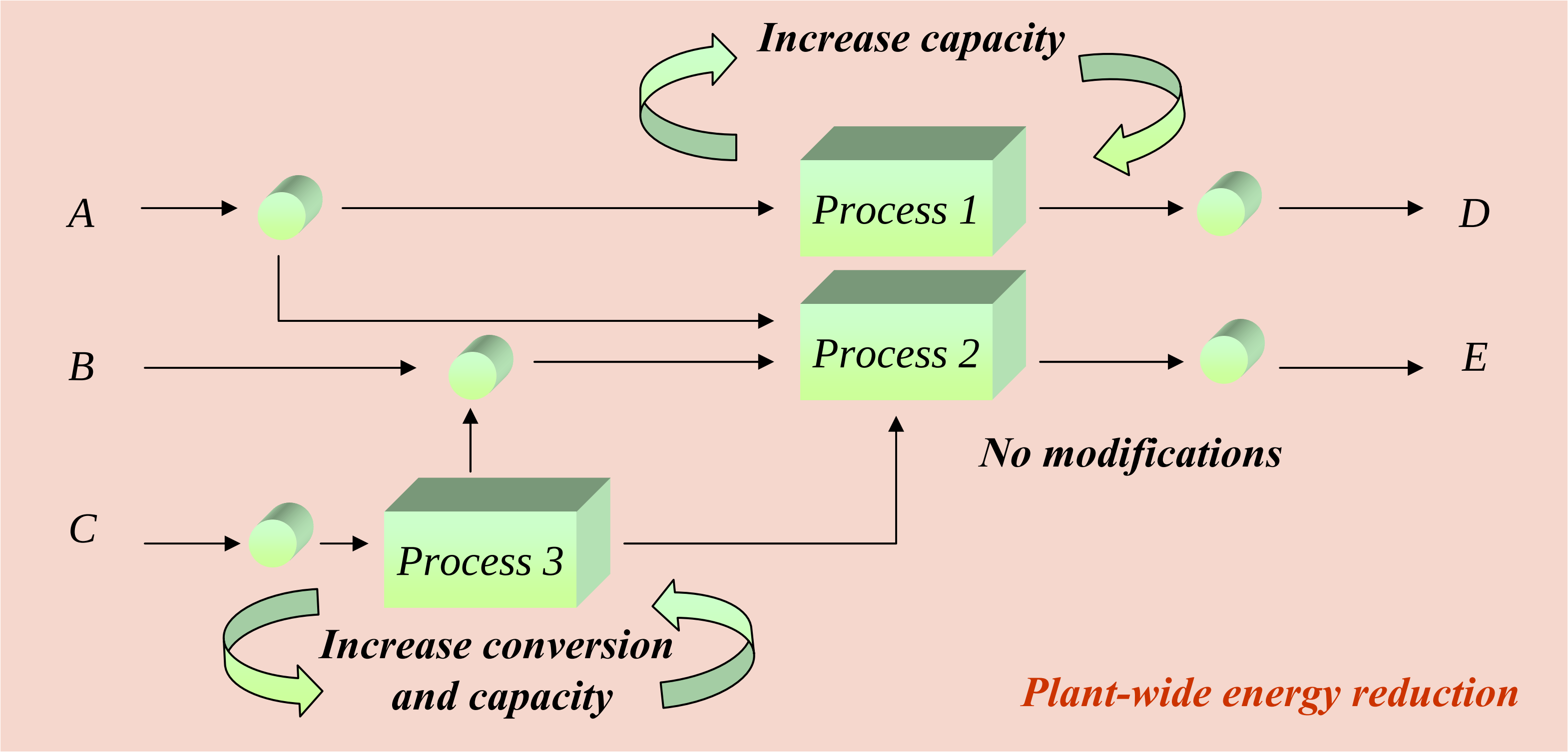
Disjunctive constraints
No clashes between jobs

GAMS

Retrofit Planning Problem

Problem Statement: *Jackson J. & Grossmann I.E. (2002)*

- **Retrofit: Redesign of existing plant.**
  - **Improvements such as higher yield, increased capacity, energy reduction.**
- **Objective is to identify modifications that maximize economic potential, given time horizon and limited capital investments.**



GDP Model For Retrofit Planning Problem

Objective function
Maximize economic potential

$$\text{Min} \sum_{\forall t\in T}\sum_{\forall s\in S_{prod}} PR_s^t mf_s^t - \sum_{\forall t\in T}\sum_{\forall s\in S_{raw}} PR_s^t mf_s^t - \sum_{\forall t\in T} PRST qst^t - \sum_{\forall t\in T} PRWT qwt^t - \sum_{\forall t\in T}\sum_{\forall p\in P} fc_p^t - \sum_{\forall t\in T} ec^t$$

Common constraints
Mass balances

$$\text{s.t.} \quad mf_s^t = f_s^t MW_s \qquad \forall s\in S, \forall t\in T$$

$$mf_s^t \geq DEM_s^t \qquad \forall s\in S_{prod}, \forall t\in T$$

$$mf_s^t \leq SUP_s^t \qquad \forall s\in S_{raw}, \forall t\in T$$

$$\sum_{\forall s\in S_{n_{in}}} mf_s^t = \sum_{\forall s\in S_{n_{out}}} mf_s^t \qquad \forall n\in N, \forall t\in T$$

$$\sum_{\forall s\in S_{p_{in}}} mf_s^t = \sum_{\forall s\in S_{p_{out}}} mf_s^t + unrct_p^t \qquad \forall p\in P, \forall t\in T$$

Disjunctive constraints
Conversion/Capacity scenarios

$$\bigvee_{\forall m\in M}\begin{bmatrix} Y_{pm}^t \\ f_s^t = f_{p_{lmt}}^t\left(\dfrac{GMA_s^t}{GMA_{p_{lmt}}}\right)ETA_{pm}^t \\ \sum_{\forall s\in S_{p_{in}}} mf_s^t \leq CAP_{pm}^t \end{bmatrix} \qquad \forall s\in S_{p_{out}}, \forall p\in P, \forall t\in T$$

$$\bigvee_{\forall m\in M}\begin{bmatrix} W_{pm}^t \\ fc_p^t = FC_{pm}^t \end{bmatrix} \qquad \forall p\in P, \forall t\in T$$

Disjunctive constraints
Energy reduction scenarios

$$q_{sk}^t = mf_s^t CP_s (T_{s_{out_k}}^t - T_{s_{in_k}}^t) \qquad \forall s\in S_{cold}, \forall k\in K, \forall t\in T$$

$$q_{sk}^t = mf_s^t CP_s (T_{s_{in_k}}^t - T_{s_{out_k}}^t) \qquad \forall s\in S_{hot}, \forall k\in K, \forall t\in T$$

$$\begin{bmatrix} X_1^t \\ qst^t = \sum_{\forall k\in K}\sum_{\forall s\in S_{cold}} q_{sk}^t \\ qwt^t = \sum_{\forall k\in K}\sum_{\forall s\in S_{hot}} q_{sk}^t \end{bmatrix} \vee \begin{bmatrix} X_2^t \\ r_k^t - r_{k-1}^t - qst_k^t + qwt_k^t = \sum_{\forall s\in S_{hot}} q_{sk}^t - \sum_{\forall s\in S_{cold}} q_{sk}^t \\ qst^t = \sum_{\forall k\in K} qst_k^t \\ qwt^t = r_{|K|}^t + \sum_{\forall k\in K} qwt_k^t \end{bmatrix} \qquad \forall k\in K, \forall t\in T$$

$$\begin{bmatrix} V_1^t \\ ec^t = EFC_1^t \end{bmatrix} \vee \begin{bmatrix} V_2^t \\ ec^t = EFC_2^t \end{bmatrix} \qquad \forall t\in T$$

Common constraint *Investment limit*

$$\sum_{\forall p\in P} fc_p^t + ec^t + \sum_{\forall s\in S_{raw}} PR_s^t mf_s^t + PRST qst^t + PRWT qwt^t \leq INV^t \qquad \forall t\in T$$

Logic constraints
Connect disjunctions

$$Y_{pm}^t \rightarrow \bigwedge_{\tau>t} Y_{pm}^\tau \qquad \forall p\in P, \forall t\in T, \forall m\in M\setminus m_1$$

$$W_{pm}^t \rightarrow \bigwedge_{\tau\neq t} W_{pl}^t \qquad \forall p\in P, \forall t\in T, \forall m\in M\setminus m_1$$

$$Y_{pl}^t \rightarrow W_{pl}^t \qquad \forall p\in P, \forall t\in T$$

$$Y_{pm}^t \bigwedge_{\tau<t} \neg Y_{pm}^\tau \rightarrow W_{pm}^t \qquad \forall p\in P, \forall t\in T, \forall m\in M\setminus m_1$$

$$X_j^t \rightarrow \bigwedge_{\tau>t} X_j^\tau \qquad \forall t\in T, \forall j\in J\setminus j_1$$

$$V_j^t \rightarrow \bigwedge_{\tau\neq t} V_i^t \qquad \forall t\in T, \forall j\in J\setminus j_1$$

$$X_1^t \rightarrow V_1^t \qquad \forall t\in T$$

$$X_j^t \bigwedge_{\tau<t} \neg X_j^\tau \rightarrow V_j^t \qquad \forall t\in T, \forall j\in J\setminus j_1$$

$$mf_s^t, f_s^t \in \mathbf{R}^1_+ \qquad \forall s\in S, \forall t\in T$$

$$f_{lmt,p}^t, unrct_p^t, fc_p^t \in \mathbf{R}^1_+ \qquad \forall p\in P, \forall t\in T$$

$$q_{sk}^t \in \mathbf{R}^1_+ \qquad \forall s\in S, \forall k\in K, \forall t\in T$$

$$qst^t, qwt^t, ec^t \in \mathbf{R}^1_+ \qquad \forall t\in T$$

$$qst_k^t, qwt_k^t, r_k^t \in \mathbf{R}^1_+ \qquad \forall k\in K, \forall t\in T$$

$$Y_{pm}^t, W_{pm}^t \in \{True, False\} \qquad \forall p\in P, \forall t\in T, \forall m\in M$$

$$X_j^t, V_j^t \in \{True, False\} \qquad \forall j\in J, \forall t\in T$$

Chemical ENGINEERING

Carnegie Mellon

ingar

GAMS

Computational Results

Linear Disjunctive Problems

Problem	# disc. var.	# var.	# Equat.	CPU BigM (sec.)	iter. BigM	nodes BigM	CPU CH (sec.)	iter CH	nodes CH
cut-1	24	34	30	0.05	64	32	0.11	92	0
cut-2	180	202	236	7.19	29,196	4673	43.78	44,434	873
jobshop-1	12	21	13	0.05	5	0	0.001	9	0
jobshop-2	245	253	78	0.16	341	86	0.93	2155	200
jobshop-3	320	319	219	3.57	10,034	2209	154.45	207,605	20,600
jobshop-4	125	131	106	0.11	288	55	2.25	3035	299
retrofit	72	160	211	0.72	1449	136	0.11	122	0
retrofitNS	336	1685	2935	1810.11	6,995,748	494,291	2.08	3019	18
pipeline	387	1640	3385	327.52	89,820	13,657	940.65	420,574	23,750

Performance of big-M and Convex Hull (CH) is problem dependent

CPLEX 7.5

Computational Results

Nonlinear Disjunctive Problems

Problem	# disc. var.	# var.	# Equat.	# nlp initial.	#nlp total	# master	CPU master (sec.)	CPU nlp (sec.)
8 processes	8	42	70	3	4	1	1.2	0.22
batch-design	54	113	217	1	2	2	0.82	0.18
spectroscopy	30	99	162	1	14	14	1.71	0.74
methanol	17	310	557	2	5	3	0.55	1.27

Logic-based Outer-Approximation (CPLEX 7.5/CONOPT)

Conclusions

1. LogMIP provides unique capability for formulating and solving discrete/continuous optimization problems (GDPs) in GAMS environment:
    - Handling disjunctions
    - Handling symbolic logic propositions

2. Numerical results show following:
    - For linear GDPs performance of big-M and convex hull is problem dependent
    - For nonlinear GDPs robustness increased with logic-based outer-approximation

3. Future work:
    - Extend big-M and convex hull reformulation to nonlinear GDPs
    - Branch and bound for linear and nonlinear GDPs

LogMIP can be downloaded from http://www.ceride.gov.ar/logmip/