# Multiperiod Blend Scheduling Problem

***Juan Pablo Ruiz***
***Ignacio E. Grossmann***

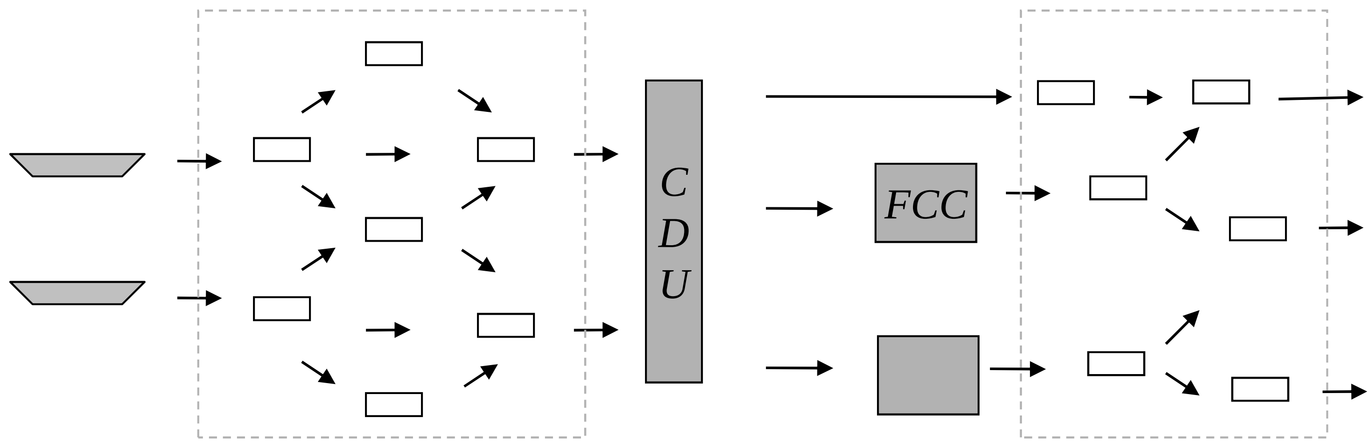
***Department of Chemical Engineering***
***Center for Advanced Process Decision-making***
***Carnegie Mellon University***
***Pittsburgh, PA 15213***

# Motivation

- **Large cost savings** can be achieved if the correct blending decisions are taken.

- **Models highly nonconvex** → **global optimization** techniques required.

- **Efficient solution methods** for large scale systems **remains as a challenge**…



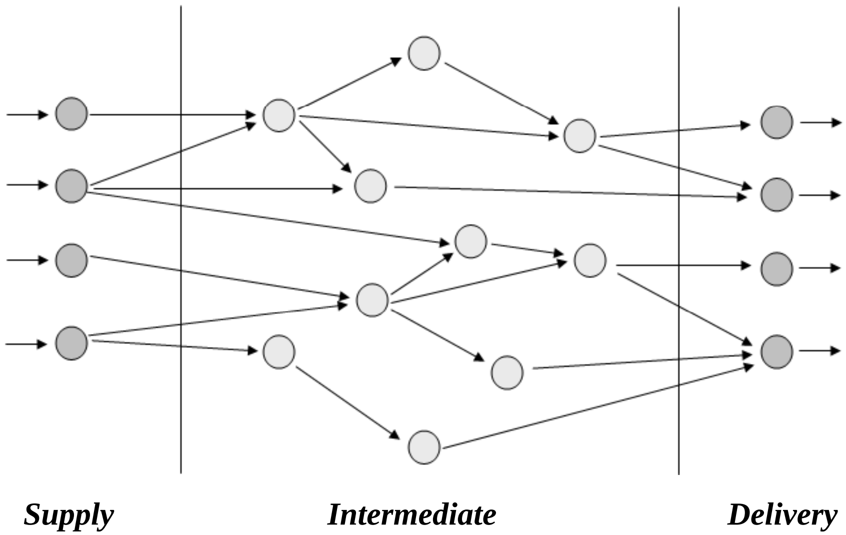

**Blending Upstream the CDU**

**Blending Downstream the CDU**

**Goal:** Develop **tools and strategies** aiming at **improving the efficiency** of the solution methods for the global optimization of the **multiperiod blend scheduling problem**

# General Problem Topology

The **general case** of a blending problem can be represented schematically as follows




**Remarks:**

Examples of **supply nodes**:
- tanks loaded by ships
- feedstocks downstream the CDU

Examples of **delivery nodes**:
- tanks feeding the CDU
- tanks delivering to final customers

*Main Model Assumptions*

- The **quality** of each stream/inventory is **constant** for a given **period**.
- A tank can **receive or deliver** in a given period of time but **not both**.
- **Supply** tanks keep a **constant quality**.
- **Delivery** tanks keep the **quality** within a given **range**.
- Streams **entering delivery** tanks should satisfy a **quality condition.**

# Work lines - Summary

## Alternative Formulations

- Proposed formulations given in the **space of properties and total flows** and in the **space of individual property flows**
- Reduced the number of bilinear terms by using **GDP formulations**
- Explored the use of **redundant constraints** to improve the relaxations

## Solution Methods

- Proposed a **Logic Based Outer Approximation** method to find local solutions
- Proposed a **Lagrangian Decomposition** method to find global solutions

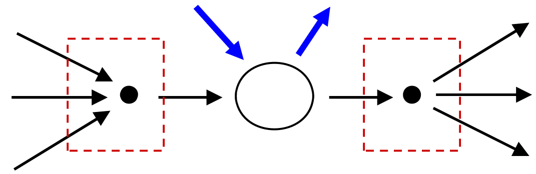
***Main Focus***

## Novel Relaxation Strategies

- Proposed the use of new relaxations based on **vector space properties**

# Alternative Formulations

# Different space formulations



### *Mixer*

***Formulation I***
Specific Property – Total Flow

$$\sum_{\substack{j' \\ (j'j)\in E}} F_{j'jt} S_{qj't} = Tf_{qjt}^{in}$$

**Nonconvex**

***Formulation II***
Total Property Flow

$$\sum_{\substack{j' \\ (j'j)\in E}} f_{q\,j'jt} = Tf_{qjt}^{in}$$

### *Splitter*

***Formulation I***
Specific Property – Total Flow

$$\sum_{\substack{j' \\ (jj')\in E}} F_{jj't} = TF_{jt}^{out}, S_{qj}$$

***Formulation II***
Total Property Flow

**Nonconvex**

$$f_{q\,j'jt} = \xi_{jj't} Tf_{qjt}^{out}$$

$$\sum_{(jj')\in E} \xi_{jj't} = 1$$

*Formulation I and II are **equivalent** but with **different relaxations !***

# Redundant Constraints*

The **total property flow** to the delivery site is constrained by an upper and lower bound. This information is lost when $CF^{B}_{qjj't-1} = C^{B}_{qjt-1} F^{B}_{qjj't-1}$ is relaxed.

$$C^{L}_{qj'} F_{jj't} - M(1 - x_{jj't}) \le CF^{B}_{qjj't-1} \qquad \forall q \in Q, j \in J^{B}, j' \in J^{D}, (jj') \in E, t \in T$$

$$CF^{B}_{qjj't-1} \le C^{U}_{qj'} F_{jj't} + M(1 - x_{jj't}) \qquad \forall q \in Q, j \in J^{B}, j' \in J^{D}, (jj') \in E, t \in T$$

For any two properties q and q' (in any stream or inventory) the ratio between the **total property flow** of q to q' is the same as the ratio between **specific property value**. This is lost when the problem is relaxed.

$$CF^{B}_{qjj't-1} C^{B}_{q'jt-1} = CF^{B}_{q'jj't-1} C^{B}_{qjt-1} \qquad \forall q, q' \in Q, j \in J^{B}, j' \in (J^{D} \cup J^{B}), (jj') \in E, t \in T$$

$$CI^{B}_{qjt-1} C^{B}_{q'jt-1} = CI^{B}_{q'jt-1} C^{B}_{qjt-1} \qquad \forall q, q' \in Q, j \in J^{B}, t \in T$$

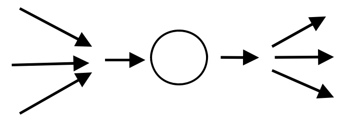
The **property balance** around each "splitter" should be held. This is lost when the bilinear terms are relaxed.

$$\sum_{\substack{j' \\ (j'j) \in E}} f_{qj'jt} = Tf^{out}_{qjt} \qquad \forall q \in Q, j \in J^{B}, t \in T$$

** Ruiz and Grossmann, 2010 "Using redundancy to strengthen the relaxation of nonconvex MINLPs"*
**To appear in Computers and Chemical Engineering Journal**

# Generalized Disjunctive Programming

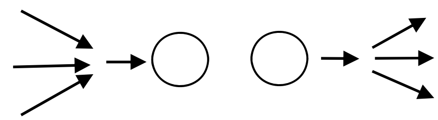
***Traditional MINLP Formulation***



***One general state***

$$C^{B}_{qjt} I_{jt} = C^{B}_{qjt-1} I_{jt-1} + \sum_{\substack{j' \in J^{P} \\ (j'j) \in E}} C^{P}_{qj'} F_{j'jt} + \sum_{\substack{j' \in J^{B} \\ (j'j) \in E}} C^{B}_{qjt-1'} F_{j'jt} - \sum_{\substack{j' \\ (j'j) \in E}} C^{B}_{qjt-1'} F_{j'jt} \quad \forall q \in Q, j \in J^{B}, t \in T$$

***Proposed GDP Formulation***



***Two states***

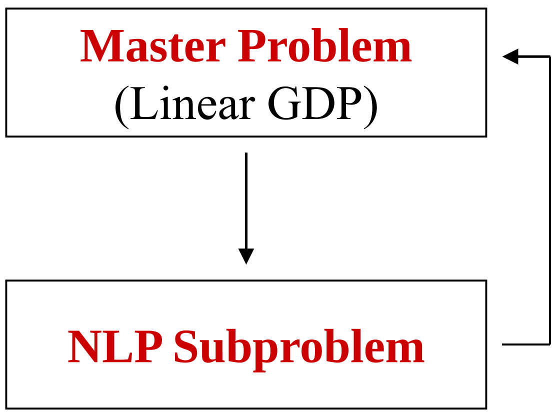
$$\begin{bmatrix} Y_{jt} \\ C^{B}_{qjt} I_{jt} = C^{B}_{qjt-1} I_{jt-1} + \sum_{\substack{j' \in J^{P} \\ (j'j) \in E}} C^{P}_{qj'} F_{j'jt} + \\ \sum_{\substack{j' \in J^{B} \\ (j'j) \in E}} C^{B}_{qjt-1'} F_{j'jt} \quad \forall q \in Q, j \in J^{B}, t \in T \end{bmatrix} \vee \begin{bmatrix} \neg Y_{jt} \\ C^{B}_{qjt} = C^{B}_{qjt-1'} \quad \forall q \in Q, j \in J^{B}, t \in T \end{bmatrix}$$

***By exploiting the underlying logic structure of the problem, a reduction of the number of bilinear terms can be achieved***

# Solution Methods

# Logic Based Outer-Approximation

*Outline of the Logic Based OA*




*No guaranty a global solution!*

***LGDP Master:***

- Relax bilinear terms using McCormick envelopes.
- Solve MIP using the Hull Reformulation.

***NLP Subproblem:***

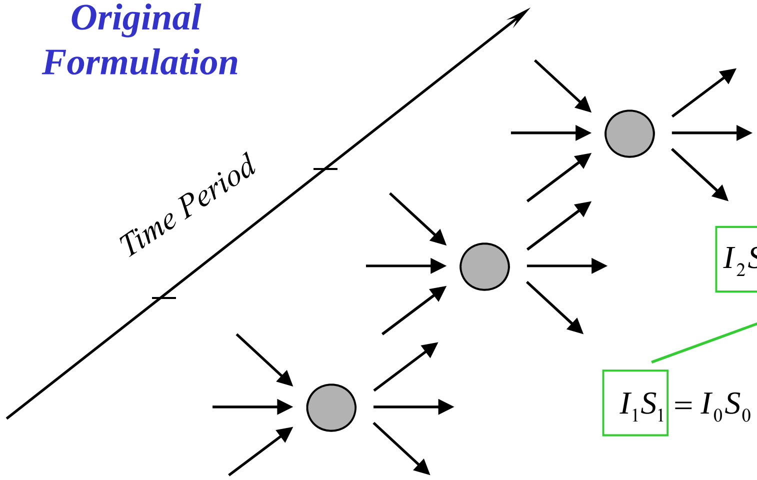
- Fix boolean variables from master problem.
- Eliminate not active disjuncts
- Solve small NLP formulation

***Iteration Step:***

- Generate linear cuts on solution point of NLP subproblem.

# Lagrangian decomposition

*Original Formulation*



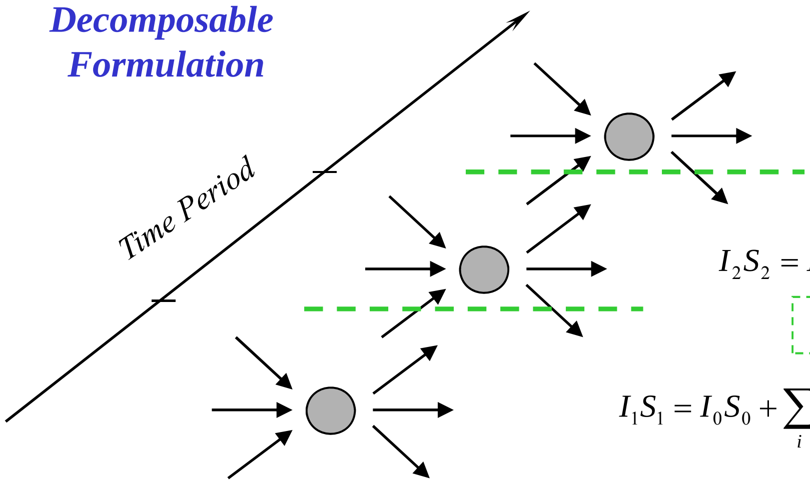

$$I_3S_3 = \boxed{I_2S_2} + \sum_i Fin_{i3}S_{i3} - \sum_i Fout_{i3}S_3$$

$$\boxed{I_2S_2} = \boxed{I_1S_1} + \sum_i Fin_{i2}S_{i2} - \sum_i Fout_{i2}S_2$$

$$\boxed{I_1S_1} = I_0S_0 + \sum_i Fin_{i1}S_{i1} - \sum_i Fout_{i1}S_1$$

*Constraints are* ***linked*** *together by the* ***inventory*** *and* ***composition*** *variables*

*Decomposable Formulation*




***Dualize***

$$I_3S_3 = I'_2\,S'_2 + \sum_i Fin_{i3}S_{i3} - \sum_i Fout_{i3}S_3$$

$$I'_2 = I_2 \;\; ; \;\; S'_2 = S_2$$

$$I_2S_2 = I'_1\,S'_1 + \sum_i Fin_{i2}S_{i2} - \sum_i Fout_{i2}S_2$$

$$I'_1 = I_1 \;\; ; \;\; S'_1 = S_1$$

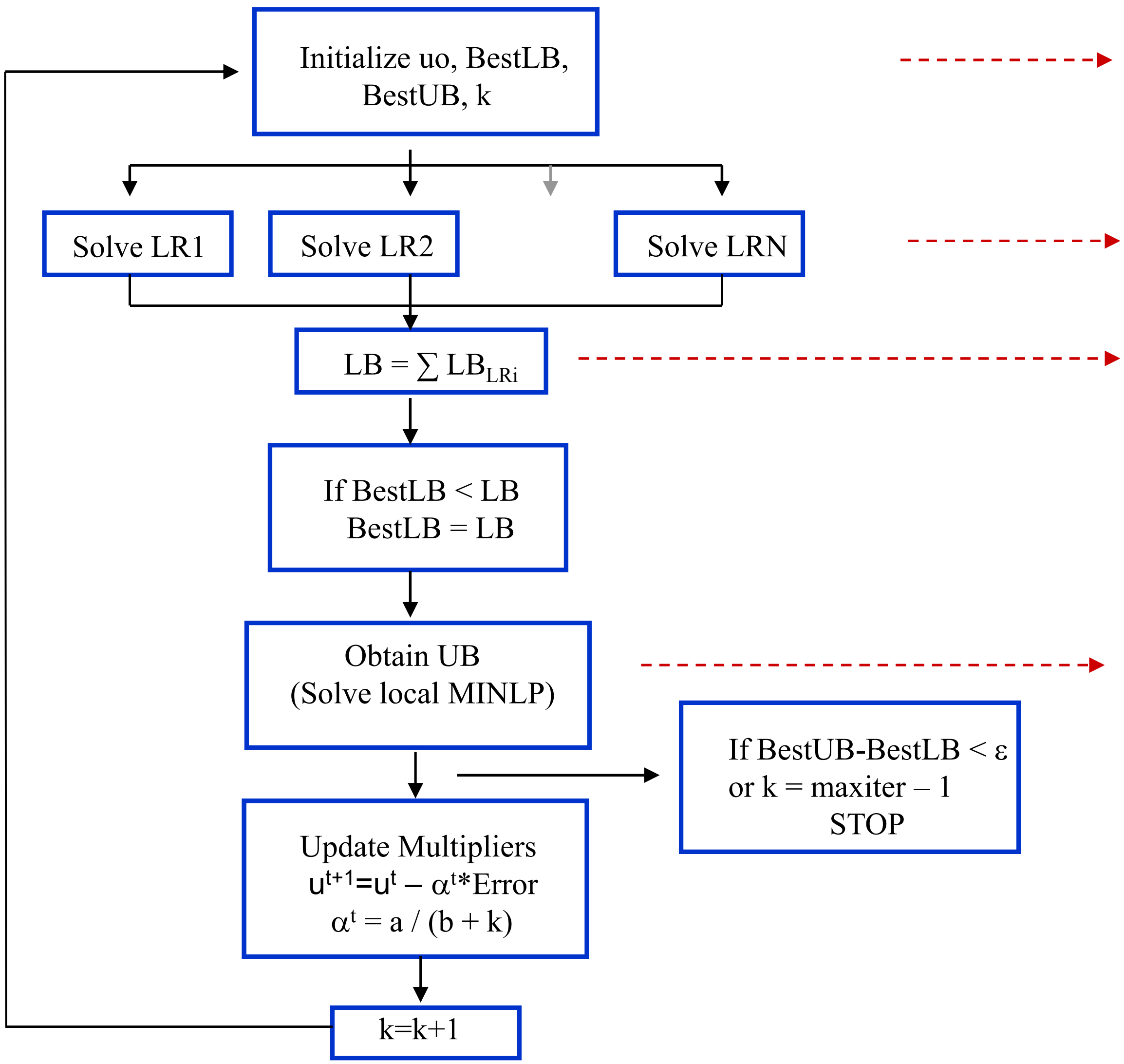
$$I_1S_1 = I_0S_0 + \sum_i Fin_{i1}S_{i1} - \sum_i Fout_{i1}S_1$$

***Duplicating inventory and composition*** *variables and* ***dualizing*** *the correspondent* ***equalities*** *leads to a temporal* ***decomposable*** *structure*

# Lagrangian decomposition

## *(Algorithm)*

### *Outline of the Lagrangian decomposition method*




***uo*** *represents the dual multipliers;* ***BestLB****, the best lower bound;* ***BestUB****, the best upper bound and* ***k****, the iter counter*

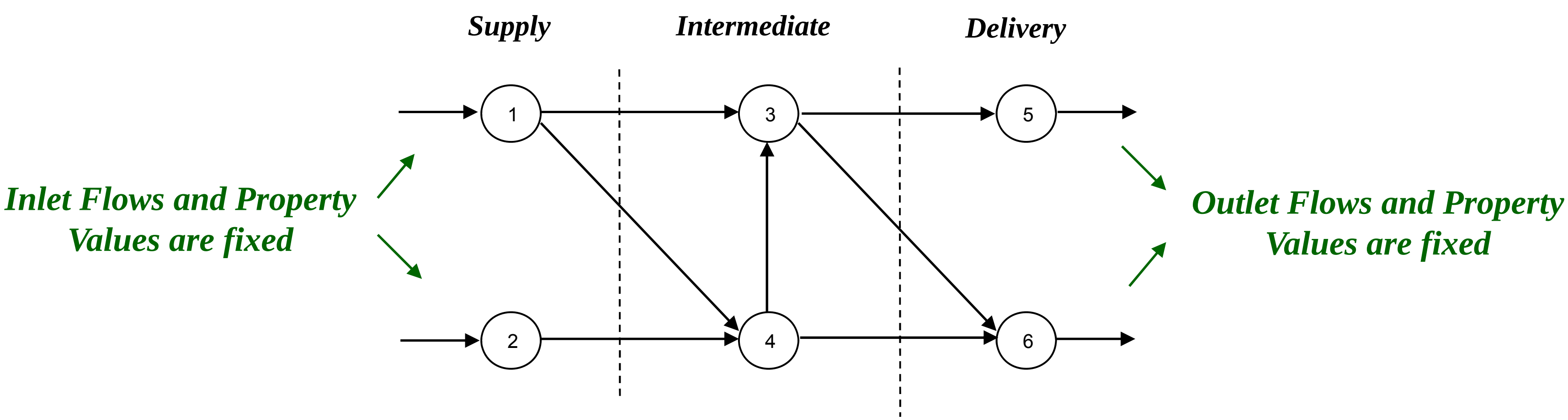
*Each* ***subproblem (LRi)*** *from the decomposition is solved*

*A* ***lower bound for the original nonconvex*** *problem can be obtained by adding up the solution of each (LRi) (i.e.* $LB_{LRi}$*)*

*Any local optimization algorithm can be used to find an UB. (e.g.* ***The logic based outer-approximation*** *applied on the GDP formulation)*

# Illustrative Example

The **implementation** of the **Lagrangian Decomposition** method has been tested in the following simple case




**Topology**
(allowed connections)

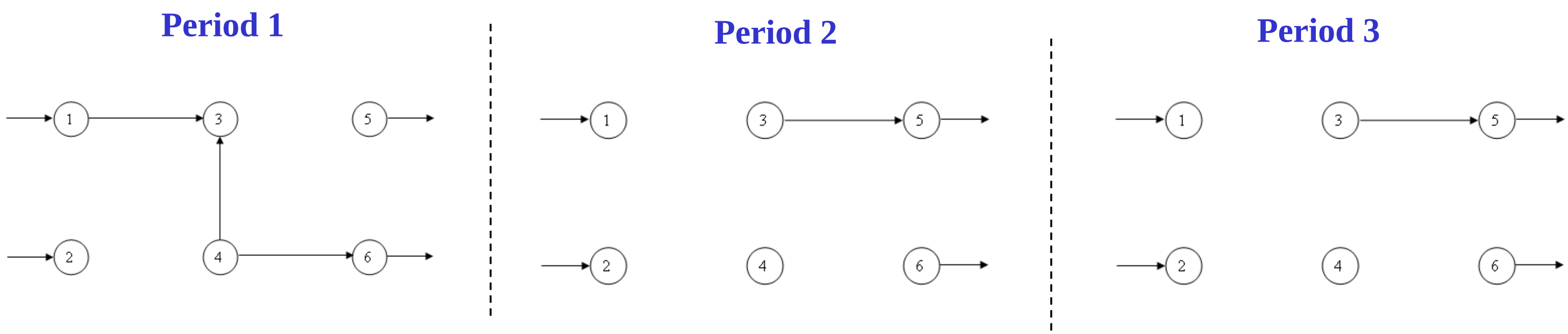
**Network Description:**

- **Two Supply, Intermediate and Delivery nodes**
- **Two properties transported**
- **Three time periods**

# Numerical Results

## Lagrangian Decomposition

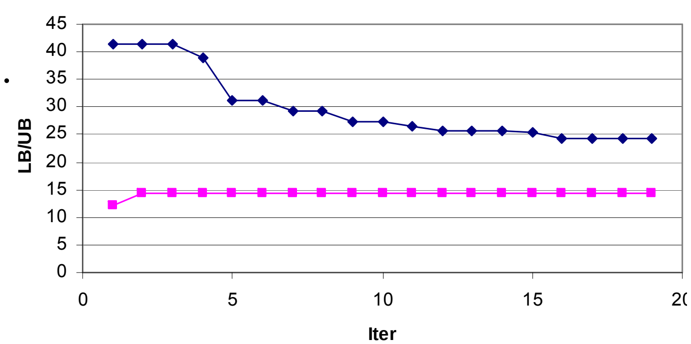
***Representation of the nonzero flow streams in the different periods for the global optimal solution***




***Global Solution (Z = 14.22) (verified with BARON)***

### *Remarks*

- Forced to stop after **20 iterations** (no improvement observed).
- **Finds the global solution** (**Z = 14.22**)
- The existence of the **duality gap is due to the nonconvex** nature of the problem



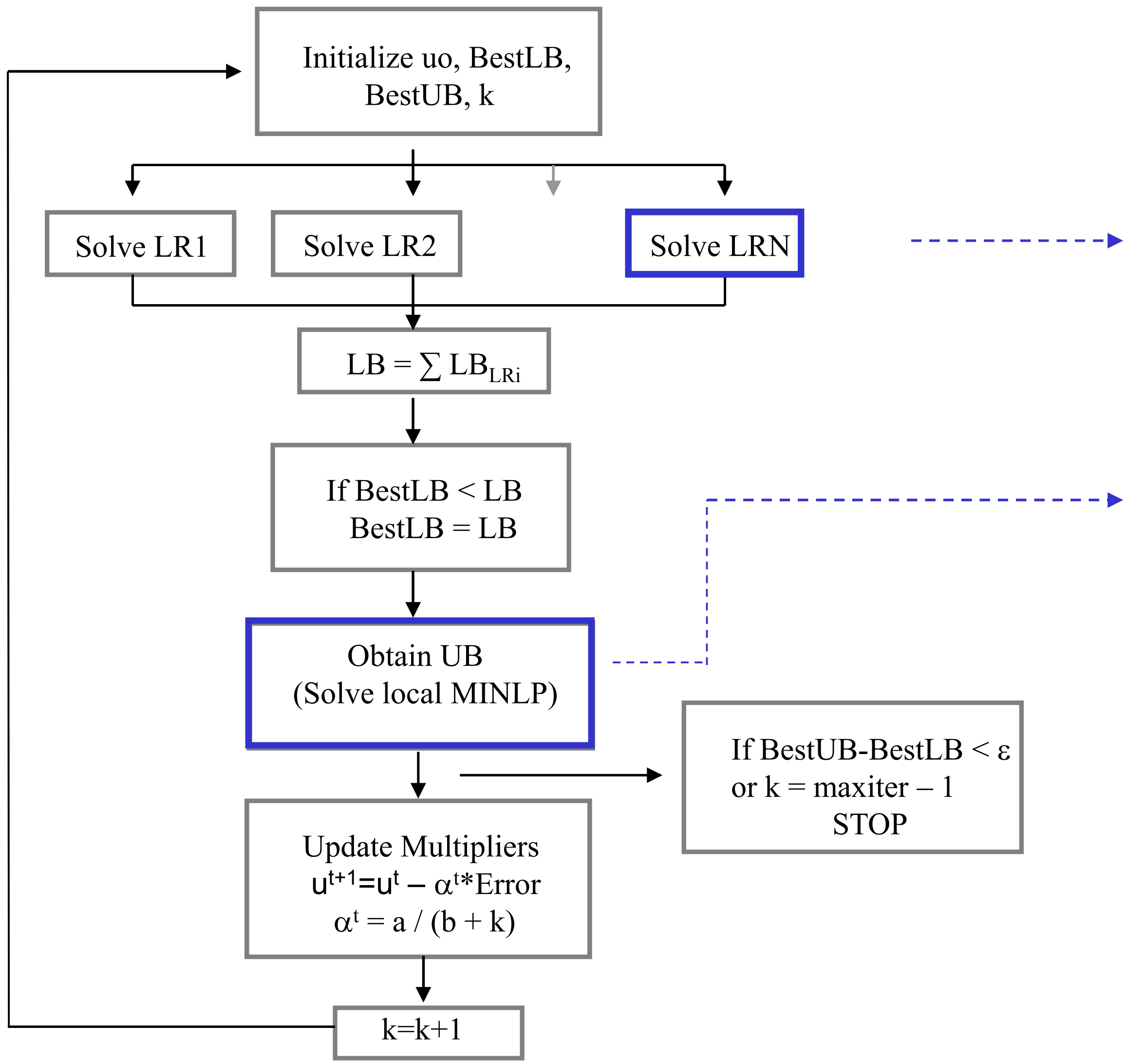
# Generation of Real-world Instances

| Structure | | | Lower Bound | Upper Bound | Bounds on Variables |
|---|---|---|---|---|---|
| Structure | Nodes | $J^P, J^B, J^D$ | 4 | 10 | |
| Structure | Properties | Q | 4 | 10 | |
| Structure | Periods | T | 40 | 100 | |
| Structure | Edges | E | 40% of all possible edges | 100% of all possible edges | |
| | | $P_{jt}$ | 0 | 100 | Bounds on Variables |
| | | $D_{jt}$ | 0 | 100 | Bounds on Variables |
| | | $S^P_{qj}$ | 0 | 1 | Bounds on Variables |
| | | $S^{B0}_{qj}$ | 0 | 1 | Bounds on Variables |
| | | $S^L_{qj}, S^U_{qj}$ | 0 | 1 | Bounds on Variables |
| | | $I^L_j, I^U_j, I^0_j$ | 0 | 100 | Bounds on Variables |
| | | $F^L_{jj'}, F^U_{jj'}$ | 0 | 100 | Bounds on Variables |

***This table was used to generate random instances***

# Observations

## Numerical tests using Lagrangian Relaxation with temporal decomposition



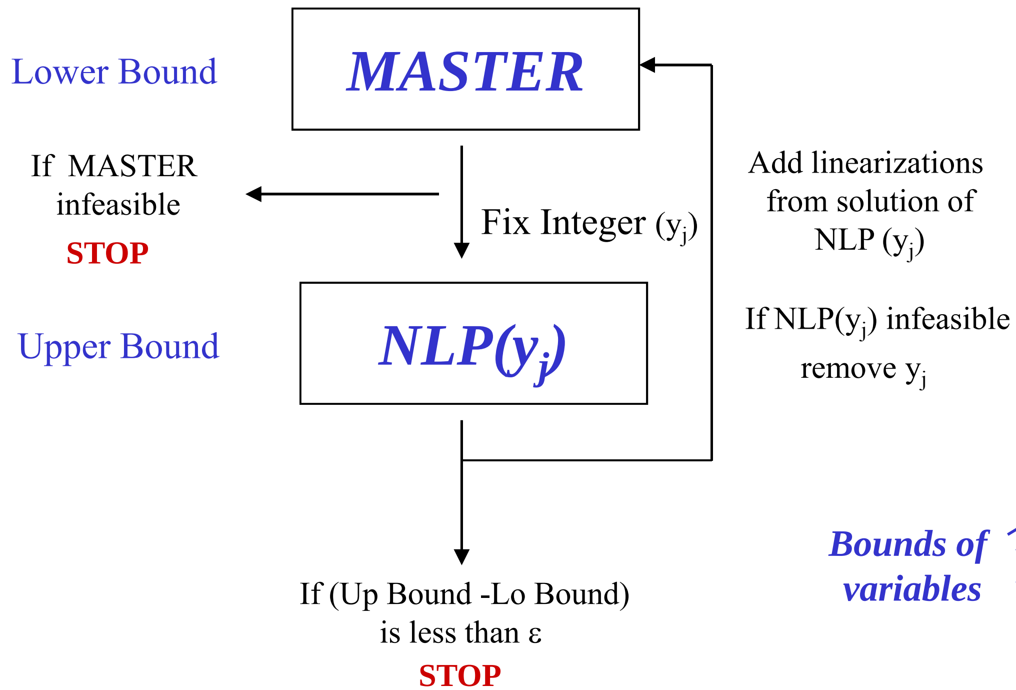

**1- High computational time required in sub-problems (> 5min)**

**2- Difficulties to find local solutions**

***How do we tackle these issues?***

# Finding Local Solutions

## Outer Approximation Method




***The MASTER problem can be tightened by adding McCormick Convex envelopes for the bilinear terms***

$$f_{qijt} \le F_{ijt} C^{UP}_{qit-1} + C_{qit-1} F^{LO}_{ij} - C^{UP}_{qit-1} F^{LO}_{ij}$$

$$f_{qijt} \le F_{ijt} C^{LO}_{qit-1} + C_{qit-1} F^{UP}_{ij} - C^{LO}_{qit-1} F^{UP}_{ij}$$

$$f_{qijt} \ge F_{ijt} C^{UP}_{qit-1} + C_{qit-1} F^{UP}_{ij} - C^{UP}_{qit-1} F^{UP}_{ij}$$

$$f_{qijt} \ge F_{ijt} C^{LO}_{qit-1} + C_{qit-1} F^{LO}_{ij} - C^{LO}_{qit-1} F^{LO}_{ij}$$

***Bounds of variables***

Similarly for terms $I_{qit} C_{qit}$

### *Remarks*

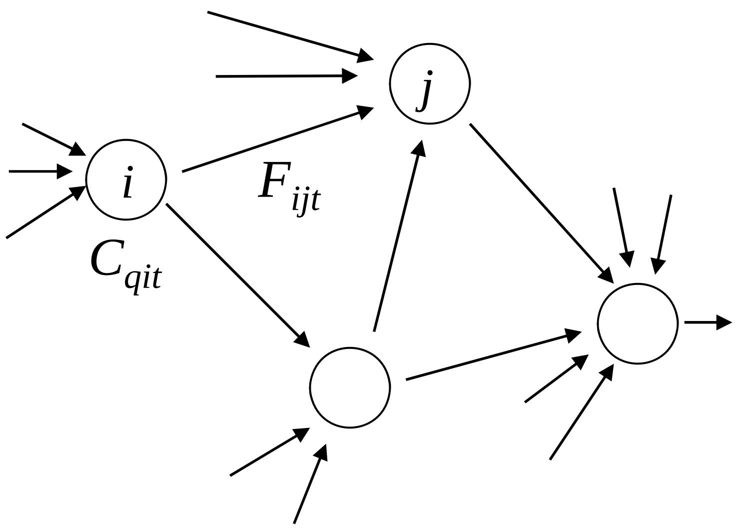
- **Reducing the number of bilinear** terms in NLP($y_j$) leads to a more **robust formulation**
- Having **good bounds** for the variables is of main importance to find **tight relaxations**

# Finding Local Solutions

## Tighter bounds for variables (I)

***Observation***

If **two streams** are **mixed** together, the **concentration** of any given component in the mixture is always **higher/lower** than the **minimum/maximum** concentration in the streams




***Mathematical Representation***

$$C_{qit} \geq \min_{(i,j)\in E} (C_{qjt-1}) \qquad \forall q,i,t$$

$$C_{qit} \leq \max_{(i,j)\in E} (C_{qjt-1}) \qquad \forall q,i,t$$

***How can we use it to infer bounds for the compositions?***

# Finding Local Solutions

## Tighter bounds for variables (II)

***Lower Bounds***

$$C_{qit}^{LO} = C_{qi0} \qquad \forall q,i,t=1$$

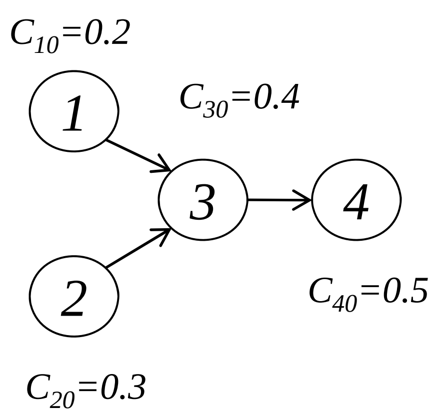
$$C_{qit}^{LO} \geq \min_{(i,j)\in E}(C_{qjt-1}^{LO}) \qquad \forall q,i,t>1$$

***Upper Bounds***

$$C_{qit}^{UP} = C_{qi0} \qquad \forall q,i,t=1$$

$$C_{qit}^{UP} \leq \max_{(i,j)\in E}(C_{qjt-1}^{UP}) \qquad \forall q,i,t>1$$

### *Illustrative Example*



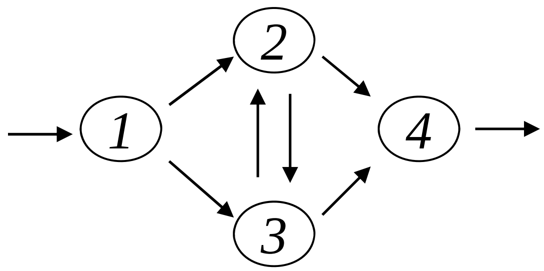

| | t=0 | | t=1 | | t=2 | |
|---|---|---|---|---|---|---|
| | LO | UP | LO | UP | LO | UP |
| Node 1 | 0.2 | 0.2 | 0.2 | 0.2 | 0.2 | 0.2 |
| Node 2 | 0.3 | 0.3 | 0.3 | 0.3 | 0.3 | 0.3 |
| Node 3 | 0.4 | 0.4 | 0.2 | 0.4 | 0.2 | 0.4 |
| Node 4 | 0.5 | 0.5 | 0.4 | 0.5 | 0.2 | 0.5 |

***Lower and upper bound tightening can be achieved in the preprocessing step***

# Finding Local Solutions

## Tighter bounds for variables (III)

### *Performance Analysis*




*Predicted lower bounds at first* ***MASTER*** *problem*

| | *Global Optimum* | *Using original bounds* | *Using inferred bounds* |
|---|---|---|---|
| *Instance 1* | -2900 | -4958 | -4083 |
| *Instance 2* | -10900 | -20958 | -14650 |

### *Remark*

**Improvements** in the bounds prediction can be obtained if lower/upper bounds of **flows and inventory** levels are **considered**

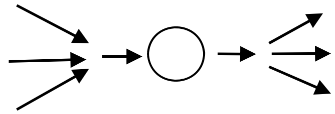
$$C_{qit} \geq \min\left(\frac{I_{qit-1}C_{qit-1} + \sum_{(i,j)\in E} F_{jit}C_{qjt-1}}{I_{qit}}\right) \geq \frac{I^{LO}_{qit-1}C^{LO}_{qit-1} + \sum_{(i,j)\in E} F^{LO}_{jit}C^{LO}_{qjt-1}}{I^{UP}_{qit}} \qquad \forall q,i,t$$

$$C_{qit} \leq \max\left(\frac{I_{qit-1}C_{qit-1} + \sum_{(i,j)\in E} F_{jit}C_{qjt-1}}{I_{qit}}\right) \leq \frac{I^{UP}_{qit-1}C^{UP}_{qit-1} + \sum_{(i,j)\in E} F^{UP}_{jit}C^{UP}_{qjt-1}}{I^{LO}_{qit}} \qquad \forall q,i,t$$

# Finding Local Solutions

## Reduced number of bilinear terms

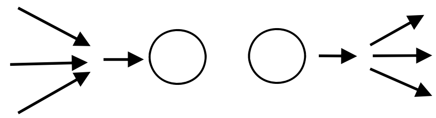
***Traditional MINLP Formulation***



***One general state***

$$C^B_{qjt} I_{jt} = C^B_{qjt-1} I_{jt-1} + \sum_{\substack{j'\in J^P \\ (j'j)\in E}} C^P_{qj'} F_{j'jt} + \sum_{\substack{j'\in J^B \\ (j'j)\in E}} C^B_{qjt-1'} F_{j'jt} - \sum_{\substack{j' \\ (j'j)\in E}} C^B_{qjt-1'} F_{j'jt} \quad \forall q \in Q, j \in J^B, t \in T$$

***Proposed GDP Formulation***



***Two states***

$$\begin{bmatrix} Y_{jt} \\ C^B_{qjt} I_{jt} = C^B_{qjt-1} I_{jt-1} + \sum_{\substack{j'\in J^P \\ (j'j)\in E}} C^P_{qj'} F_{j'jt} + \\ \sum_{\substack{j'\in J^B \\ (j'j)\in E}} C^B_{qjt-1'} F_{j'jt} \quad \forall q \in Q, j \in J^B, t \in T \end{bmatrix} \vee \begin{bmatrix} \neg Y_{jt} \\ C^B_{qjt} = C^B_{qjt-1'} \quad \forall q \in Q, j \in J^B, t \in T \end{bmatrix}$$

***By exploiting the underlying logic structure of the problem, a reduction of the number of bilinear terms can be achieved***

# Finding Local Solutions

## Numerical Results

### *Performance Analysis*

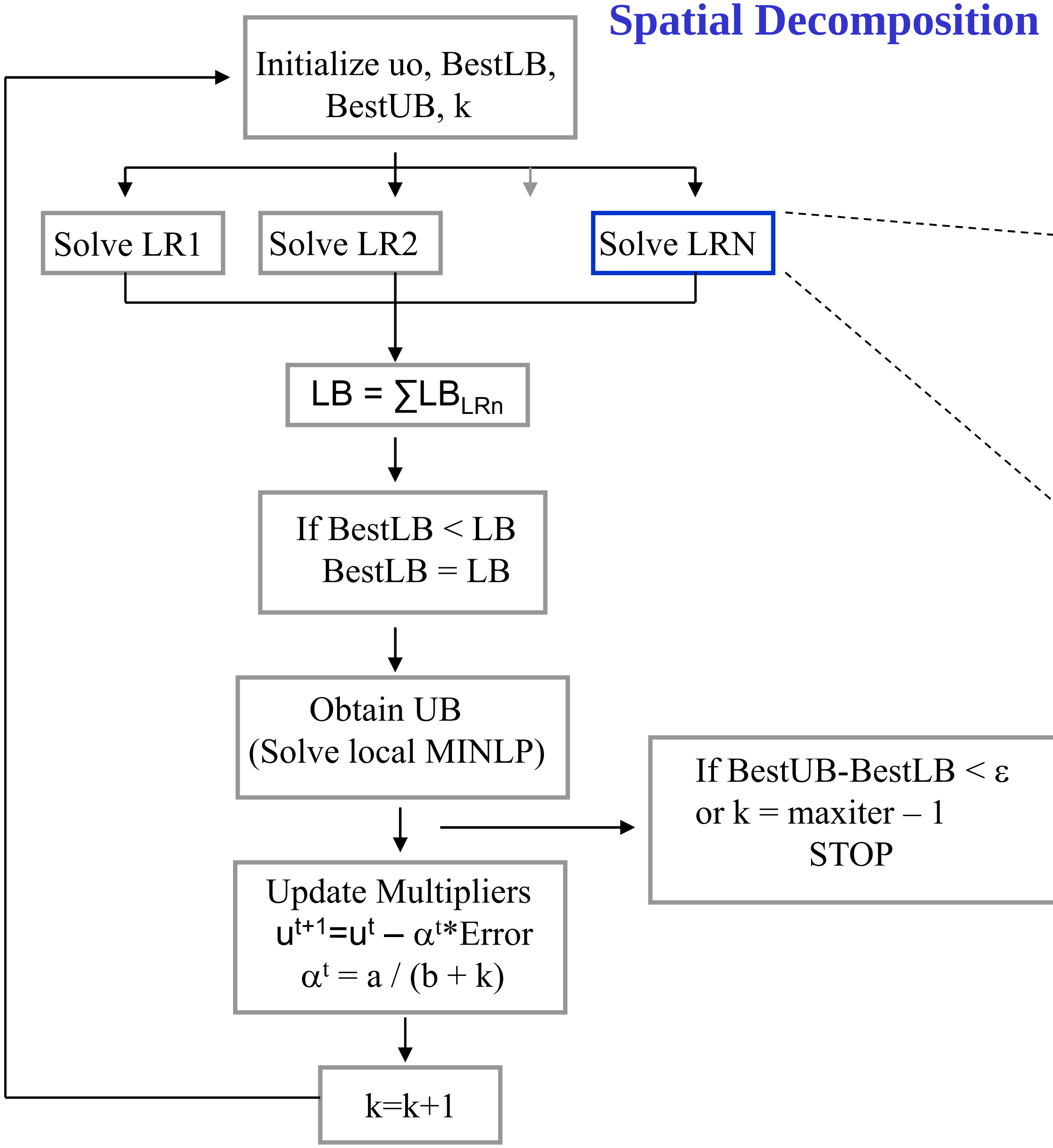
- 11 **random instances**
- Outer approximation solver **DICOPT(GAMS)**
- Three different formulations (all using McCormick envelopes):
  - **1**- Original MINLP
  - **2**- Formulation with reduced number of bilinear terms
  - **3**- Formulation with reduced number of bilinear terms plus bound tightening
- Forced to **stop after 10 iterations or 30 minutes**

### *Remarks*

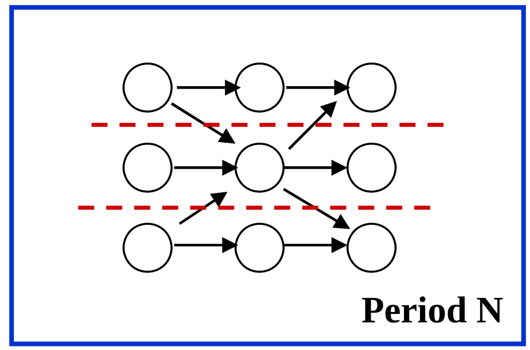
- Formulation (**2**) and (**3**) found **feasible solutions** in more than **70%** of instances
- Formulation (**3**) **outperformed** Formulation (**2**) in **20%** of the instances
- Formulation (**1**) led to a large number of **“false” infeasible** problems

# *Solution of $LR_i$ sub-problems*

## Spatial Decomposition




***Spatial Decomposition***




***How do we decompose it spatially?***

# Solution of LRi sub-problems

## Minimal cut-edge with fixed nodes

*Objective:* *Minimize the edges that cross the boundaries of each subset*

**Incidence Matrix**

$$\min \sum_{ijk} A_{ij}(y_{ik} - z_{ijk})$$

$$s.t. \quad \sum_{k} y_{ik} = 1 \qquad \forall i$$

$$\sum_{i} y_{ik} = \alpha_k \qquad \forall k$$

**Number of nodes in disjoint subsets**

**If $y_{ik}$ = 1 then the node i belongs to the subset k**

$$1 - y_{ik} - y_{jk} + z_{ijk} \geq 0 \qquad \forall ijk$$

$$y_{ik} \geq z_{ijk} \qquad \forall ijk$$

$$y_{jk} \geq z_{ijk} \qquad \forall ijk$$

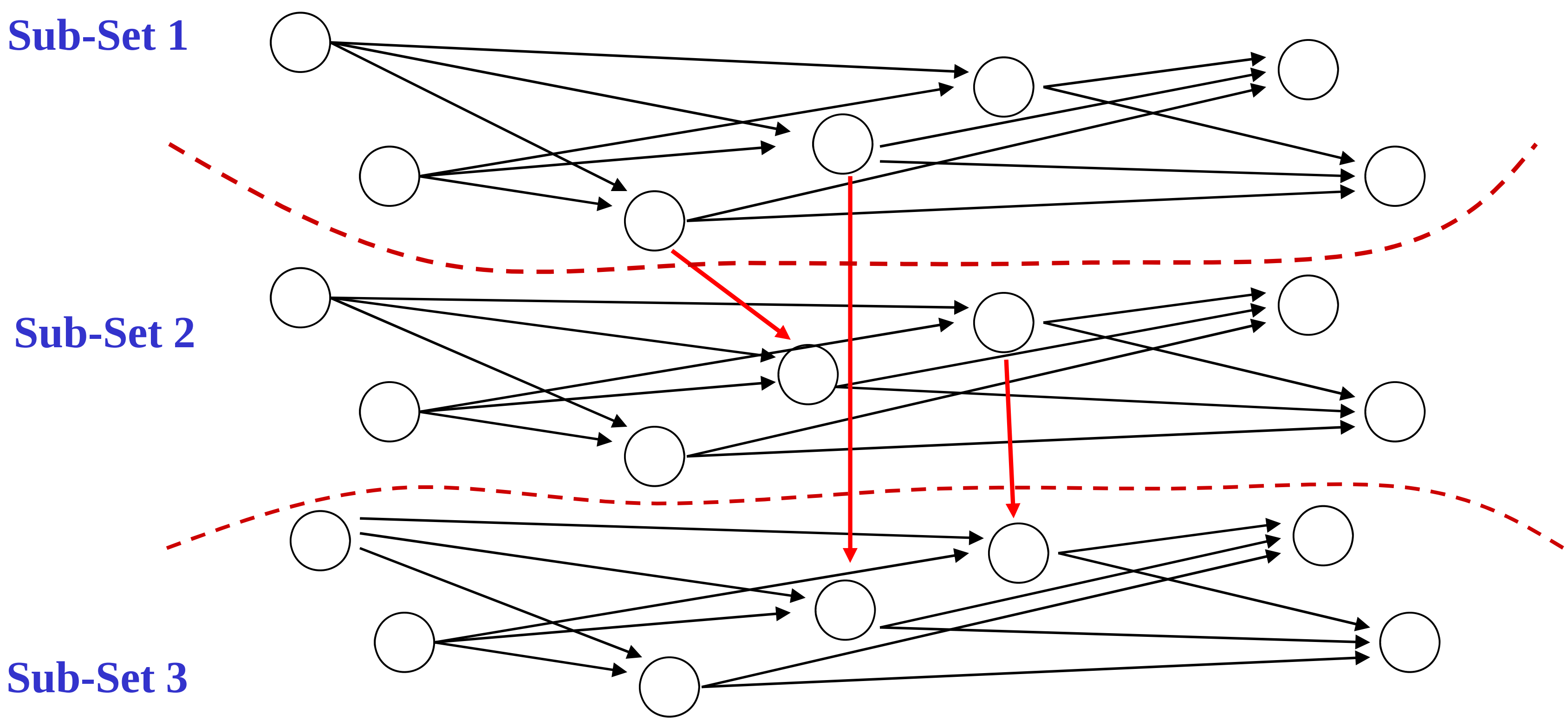
→ $z_{ijk} = y_{ik} y_{jk}$

$$\sum_{ij} z_{ijk} = \alpha_k^2 \qquad \forall k$$

→ **cut!**

# Solution of LRi sub-problems

## Minimal cut-edge with fixed nodes example




Dualized constraints necessary: **3(n+1)**
(*n: number of properties considered*)

# Solution of $LR_i$ sub-problems

## Numerical Results

-Baron takes 347 seconds **(~6min)** to solve the problem with a solution of **20954.8**

-The spatial decomposition solves the problem in **1 iteration**:

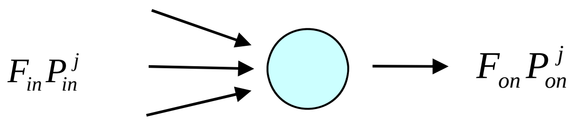
| | | |
|---|---|---|
| MIP separation problem: | | 5 seconds |
| Sub-problem 1: | (sol: 6096.0) | 1.6 seconds |
| Sub-problem 2: | (sol: 11451.8) | 1.4 seconds |
| Sub-problem 3: | (sol: 3407.0) | 1.5 seconds |
| TOTAL: | **(sol: 20954.8)** | **9.5 seconds** |

***Remarks:***

- Even though it is not expected for general problems to converge in one iteration, **even with 15 iterations, the time necessary would be ~1 min**

# Novel Relaxations

# Vector space properties to strengthen the relaxation




***Main building block of a process network***

### *Algebraic Representation*

$$\sum_{i \in I_n} (F_{in} P_{in}^{j}) - F_{on} P_{in}^{j} = 0$$

$$\sum_{i \in I_n} F_{in} - F_{on} = 0$$

### *Vectorial Representation*

$$v_{F_n} \perp v_E$$

$$v_{P_n^j} \perp v_{F_n}$$

→ *Vector Space Interaction Exposed*

$$v_{F_n} = (F_1, F_2, \ldots, F_{|I_n|}, F_o)$$

$$v_E = (1, 1, \ldots, 1, -1)$$

$$v_{P_n^j} = (P_1^j, P_2^j, \ldots, P_{|I_n|}^j, -P_o^j)$$

### *Exploit interaction to develop cuts (3-D Case)*

$$v_{F_n} \perp v_E, \quad v_{P_n^j} \perp v_{F_n} \quad \Rightarrow \quad v_F \times v_P = \alpha v_F$$

$$\alpha = \frac{\| v_P \| \sqrt{3} \sin \theta}{\| v_F \|}, \quad 0 \le \theta \le 2\pi$$

**Cuts** → *(for a given j and n)*

$$\beta_i \le \alpha F_i^{up} + \alpha^{lo} F_i - \alpha^{lo} F_i^{up}$$

$$\beta_i \le \alpha F_i^{lo} + \alpha^{up} F_i - \alpha^{up} F_i^{lo}$$

$$\beta_i \ge \alpha F_i^{up} + \alpha^{up} F_i - \alpha^{up} F_i^{up}$$

$$\beta_i \ge \alpha F_i^{lo} + \alpha^{lo} F_i - \alpha^{lo} F_i^{lo}$$

$$\beta_1 = P_o - P_2$$

$$\beta_2 = P_1 - P_o$$

$$\beta_o = P_1 - P_2$$

# Numerical Results*

Table – Comparison of the performance of proposed approach with traditional relaxations

| | | Traditional Approach | | | Proposed Approach | | |
|---|---|---|---|---|---|---|---|
| Instance | GO | LB | Nodes | Time(s) | LB | Nodes | Time(s) |
| 1 | 82.78 | 78.25 | 11 | 20 | 81.6 | 4 | 9 |
| 2 | 151.96 | 142.10 | 35 | 30 | 144.30 | 17 | 17 |
| 3 | 5.19 | 4.86 | 87 | 109 | 5.05 | 6 | 22 |
| 4 | 9.79 | 8.22 | 650 | 542 | 8.78 | 365 | 325 |
| 5 | 12.60 | 12.48 | 5 | 19 | 12.60 | 1 | 17 |
| 6 | -264.01 | -574.5 | 877 | 596 | -530.0 | 460 | 321 |
| 7 | -1308.0 | -1985.3 | 97 | 77 | -1930.0 | 63 | 56 |

*All problems were solved using a Pentium(R) CPU 3.4 GHz and 1GB RAM*

***Pooling problems!***

*Ruiz J.P. and Grossmann I.E. 2010, “Exploiting Vector Space Properties for the Global Optimization of Process Networks” , Optimization Letters

# Remarks

- Proposed formulations given in the **space of properties and total flows** and in the **space of individual property flows**
- Reduced the number of bilinear terms by using **GDP formulations**
- Explored the use of **redundant constraints** to improve the relaxations
- Proposed a **Logic Based Outer Approximation** method to find local solutions
- Proposed a **Lagrangian Decomposition** method to find global solutions
- Proposed the use of new relaxations based on **vector space properties**

## *Future Work*

- Implement **spatial decomposition** of the sub-problems within the **global** optimization **framework**.

- Add **cuts to strengthen relaxation for LR** (from Vector Space Analysis?)