Deterministic Global Optimization: Advances in Theory and Applications

Christodoulos A. Floudas

Stephen C. Macaleer `63 Professor in Engineering and Applied Science
Professor of Chemical and Biological Engineering

Princeton University

- **Department of Chemical and Biological Engineering**
- **Program of Applied and Computational Mathematics**
- **Department of Operations Research and Financial Engineering**
- **Center for Quantitative Biology**

(Adjiman, Androulakis, Akrotirianakis, Gounaris, Meyer, Maranas, Misener)

Computer Aided Systems Laboratory

Interface

- Chemical Engineering
- Applied Mathematics
- Operations Research
- Computer Science
- Computational Chemistry
- Computational Biology

Research Areas

Product & Process Systems Engineering

Computational Biology & Genomics

Themes

Mathematical Modeling, Optimization Theory & Algorithms

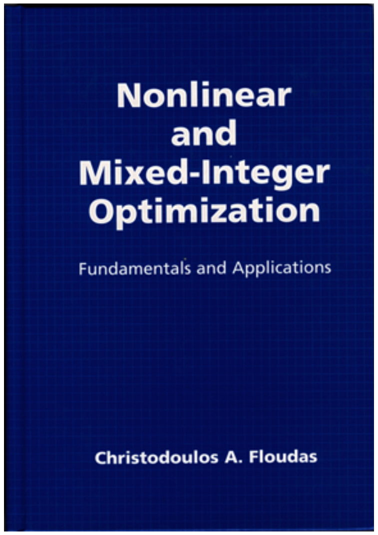
Discovery at the Macroscopic Level

Discovery at the Microscopic Level

Computer Aided Systems Laboratory

Mathematical Modeling, Optimization Theory and Algorithms

Discrete-Continuous Nonlinear Optimization

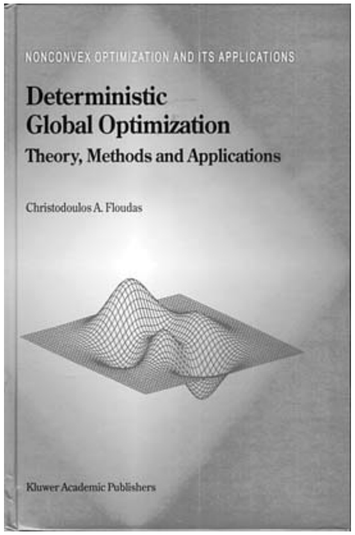


Computational Tools

MINOPT

- Nonideal Separation Systems
- Energy Recovery Networks
- Reactors / Reaction Networks
- Mass Exchange Networks
- Metabolic Pathways / Networks

Deterministic Global Optimization



Computational Tools

aBB

sMIN-aBB

gMIN-aBB

GOP

ASTRO-FOLD

- C2 NLPs
- MINLPs
- DAEs
- Bilevel
- Enclosure of All Solutions
- Pooling / Blending
- Parameter Estimation
- Protein Folding

Computer Aided Systems Laboratory

Discovery at the Macroscopic Level

Product and Process Design and Synthesis

- Methods for Design under Uncertainty
- Parameter Estimation under Uncertainty
- Topology in Metabolic Networks
- Nonideal Separations
- Phase & Chemical Reaction Equilibrium
- Parameter Estimation of Algebraic and Dynamic Models

Computational Tools

MINOPT

aBB

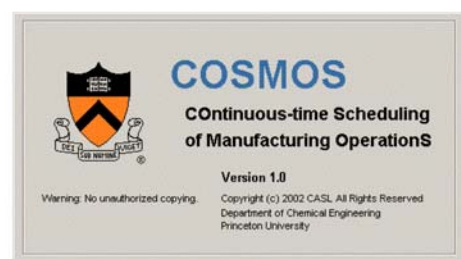
sMIN-aBB

gMIN-aBB

Product and Process Operations: Scheduling, Planning & Uncertainty

- Short Term Scheduling
- Medium Range Scheduling
- Reactive Scheduling
- Scheduling under Uncertainty
- Design and Scheduling under Uncertainty
- Long term Planning

Computational Tools



Computer Aided Systems Laboratory

Discovery at the Microscopic Level

Bioinformatics and Computational Genomics

- Structure Prediction in Protein Folding
  - Secondary Structure
  - Tertiary Structure
- Structure Refinement for NMR
- Dynamics of Protein Folding
- Protein-Protein Interactions
- De Novo Protein Design
- Topology of Signal Transduction Networks and Metabolic Pathways
- Peptide & Protein Identification via Tandem Mass Spectroscopy

Computational Tools

ASTRO-FOLD

In Silico ProtDIS

Outline

- **Deterministic Global Optimization**: Objectives & Motivation
- Convex Envelopes:
  - Trilinear Monomials
  - Edge Concave functions
- Piecewise linearization of Bilinear terms
- Checking Convexity: Products of Univariate Functions
- PαBB: Piecewise Quadratic Perturbations
- Pooling Problems: Standard, Generalized & Extended
- Conclusions

Deterministic Global Optimization: Objectives

♦ **Objective 1**

Determine a global minimum of the objective function subject to the set of constraints

♦ **Objective 2**

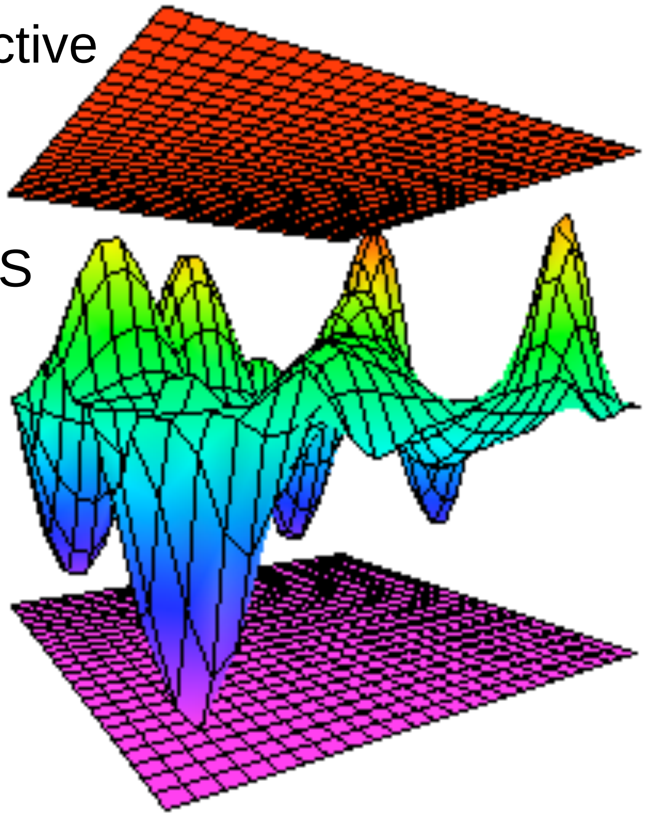
Determine LOWER and UPPER BOUNDS on the global minimum

♦ **Objective 3**

Identify good quality solutions (i.e., local minima close to the global minimum)

♦ **Objective 4**

Enclose ALL SOLUTIONS of constrained systems of equations



Objective 2
Objective 3

→

Major Importance in Engineering Applications

Deterministic Global Optimization: C^2 NLPs

Formulation

$$\min_{\mathbf{x}} \quad f(\mathbf{x})$$
$$\text{s.t.} \quad \mathbf{h}(\mathbf{x}) = \mathbf{0}$$
$$\mathbf{g}(\mathbf{x}) \leq \mathbf{0}$$
$$\mathbf{x} \in \mathbf{X} \subseteq R^n$$

$$f, \; \mathbf{h}, \; \mathbf{g} \in C^2$$

Application Areas

- **Phase Equilibrium Problems**
  - Minimum Gibbs Free Energy
  - Tangent Plane Stability
- **Pooling/Blending**
- **Parameter Estimation &**
- **Data Reconciliation**
  - Physical Properties
- **Design Under Uncertainty**
- **Robust Stability of Control Systems**
- **Structure Prediction in Clusters**
- **Structure Prediction in Molecules**
- **Protein Folding**
- **Peptide Docking**
- **NMR Structure Refinement**
- **Prediction of Crystal Structure**

Deterministic Global Optimization: MINLPs

Formulation

$$\min_{\mathbf{x},\mathbf{y}} \quad f(\mathbf{x},\mathbf{y})$$

$$\text{s.t.} \quad \mathbf{h}(\mathbf{x},\mathbf{y}) = \mathbf{0}$$

$$\mathbf{g}(\mathbf{x},\mathbf{y}) \le \mathbf{0}$$

$$\mathbf{x} \in \mathbf{X} \subseteq R^n$$

$$\mathbf{y} \quad \text{INTEGER}$$

continuous relaxations

$$f, \ \mathbf{h}, \ \mathbf{g} \in C^2$$

Application Areas

- **Process Synthesis Problems**
  - HENs
  - Separations/Complex Columns
  - Reactor Networks
  - Flowsheets
- **Scheduling, Design, Synthesis of Batch and Continuous Processes**
- **Planning**
- **Synthesis Under Uncertainty**
- **Design, Synthesis of Materials**
- **Metabolic Pathways**
- **Circuit Design**
- **Layout Problems**
- **Nesting of Arbitrary Objects**

Deterministic Global Optimization: Bilevel Nonlinear Optimization, BNLPs

Formulation

$$
\begin{aligned}
&\min_{\mathbf{x},\mathbf{y}} \quad F(\mathbf{x},\mathbf{y}) \\
\text{s.t.} \quad & \mathbf{H}(\mathbf{x},\mathbf{y}) = \mathbf{0} \\
& \mathbf{G}(\mathbf{x},\mathbf{y}) \le \mathbf{0} \\
& \min_{\mathbf{y}} \quad f(\mathbf{x},\mathbf{y}) \\
& \quad \text{s.t.} \quad \mathbf{h}(\mathbf{x},\mathbf{y}) = \mathbf{0} \\
& \quad \phantom{\text{s.t.} \quad} \mathbf{g}(\mathbf{x},\mathbf{y}) \le \mathbf{0} \\
& \mathbf{x} \in \mathbf{X} \subseteq R^{n1}, \mathbf{y} \in \mathbf{Y} \subseteq R^{n2}
\end{aligned}
$$

Application Areas

- Economics
- Civil Engineering
- Aerospace
- Chemical Engineering
  - Design Under Uncertainty : Flexibility Analysis
  - Chemical Equilibrium Process Design
  - Location/Allocation in Exploration
  - Interaction of Design with Control
  - Optimal Pollution Control
  - Molecular Design
  - Pipe Network Optimization

Recent Reviews

- Floudas, Akrotirianakis, Caratzoulas, Meyer, Kallrath (2005), "Global Optimization in the 21st Century", *Computers & Chemical Engineering*, 29(6), 1185-1202.
- Floudas, and Gounaris, (2009), "A Review of Recent Advances In Global Optimization", *Journal of Global Optimization*, 45, 3-38.

Outline

- **Deterministic Global Optimization**: Objectives & Motivation
- **Convex Envelopes:**
    - **Trilinear Monomials**
    - **Edge Concave functions**
- Piecewise linearization of Bilinear terms
- Checking Convexity: Products of Univariate Functions
- PαBB: Piecewise Quadratic Perturbations
- Pooling Problems: Standard, Generalized & Extended
- Conclusions

Convex Envelopes for Trilinear Monomials

(Meyer and Floudas, *JOGO*, 2003)

- The convex envelope of a trilinear monomial is polyhedral over a coordinate aligned hyper-rectangular domain.

- A triangulation of the domain defines the convex envelope of the monomial.

- The correct triangulation is determined by a set of conditions related to the minimal affine dependencies of the vertices of the hyper-rectangle.

- An explicit set of formulae for the elements of the convex envelope is defined for each set of conditions.

Convex Envelopes for Trilinear Monomials

(Meyer and Floudas, *JOGO*, 2003)

Positive Bounds

If $\underline{x} \geq 0$, $\underline{y} \geq 0$ and $\underline{z} \geq 0$ and the auxiliary conditions apply:

$$\overline{x}\underline{y}\underline{z} + \underline{x}\overline{y}\overline{z} \leq \underline{x}\overline{y}\underline{z} + \overline{x}\underline{y}\overline{z} \qquad \overline{x}\underline{y}\underline{z} + \underline{x}\overline{y}\overline{z} \leq \overline{x}\overline{y}\underline{z} + \underline{x}\underline{y}\overline{z}$$

the linear equalities defining the facets of the convex envelope are:

$$w = \underline{y}\underline{z}x + \underline{x}\underline{z}y + \underline{x}\underline{y}z - 2\underline{x}\underline{y}\underline{z}$$

$$w = \overline{y}\overline{z}x + \overline{x}\overline{z}y + \overline{x}\overline{y}z - 2\overline{x}\overline{y}\overline{z}$$

$$w = \underline{y}\overline{z}x + \underline{x}\overline{z}y + \overline{x}\underline{y}z - \underline{x}\underline{y}\overline{z} - \overline{x}\underline{y}\overline{z}$$

$$w = \overline{y}\underline{z}x + \overline{x}\underline{z}y + \underline{x}\overline{y}z - \overline{x}\overline{y}\underline{z} - \underline{x}\overline{y}\underline{z}$$

$$w = \frac{\theta}{\overline{x} - \underline{x}}x + \overline{x}\underline{z}y + \overline{x}\underline{y}z + \left(-\frac{\theta\underline{x}}{\overline{x} - \underline{x}} - \overline{x}\overline{y}\underline{z} - \overline{x}\underline{y}\overline{z} + \underline{x}\overline{y}\overline{z} \right)$$

$$w = \frac{\theta}{\underline{x} - \overline{x}}x + \underline{x}\overline{z}y + \underline{x}\overline{y}z + \left(-\frac{\theta\overline{x}}{\underline{x} - \overline{x}} - \underline{x}\underline{y}\overline{z} - \underline{x}\overline{y}\underline{z} + \overline{x}\underline{y}\underline{z} \right)$$

$$\textit{where} \;\; \theta = \underline{x}\underline{y}\overline{z} - \overline{x}\underline{y}\underline{z} - \underline{x}\overline{y}\overline{z} + \underline{x}\overline{y}\underline{z}$$

Convex Envelopes for Trilinear Monomials

(Meyer and Floudas, *JOGO*, 2003)

Illustration

To construct the concave envelope of $x_1 x_2 x_3$ for $(x_1, x_2, x_3) \in [1,2]\times[1,2]\times[2,4]$. We substitute $y \leftarrow x_1$, $x \leftarrow x_2$, and $z \leftarrow x_3$ and check conditions:

$$\bar{x}\underline{y}\underline{z} + \underline{x}\bar{y}\bar{z} \le \underline{x}\bar{y}\underline{z} + \bar{x}\underline{y}\bar{z} \qquad \bar{x}\underline{y}\underline{z} + \underline{x}\bar{y}\bar{z} \le \bar{x}\bar{y}\underline{z} + \underline{x}\underline{y}\bar{z}$$

which translate into,

$$\bar{x}_2\underline{x}_1\underline{x}_3 + \underline{x}_2\bar{x}_1\bar{x}_3 \le \underline{x}_2\bar{x}_1\underline{x}_3 + \bar{x}_2\underline{x}_1\bar{x}_3$$

$$(3)(1)(2)+(1)(2)(4)\le(1)(2)(2)+(3)(1)(4)$$

$$14 \le 16$$

and,

$$\bar{x}_2\underline{x}_1\underline{x}_3 + \underline{x}_2\bar{x}_1\bar{x}_3 \le \underline{x}_2\underline{x}_1\bar{x}_3 + \bar{x}_2\bar{x}_1\underline{x}_3$$

$$(3)(1)(2)+(1)(2)(4)\le(1)(1)(4)+(3)(2)(2)$$

$$14 \le 16$$

Both conditions hold, so we can use the substitutions in the facet defining equations.

Convex Envelopes for Trilinear Monomials

(Meyer and Floudas, *JOGO*, 2003)

Facet Defining Equations

$$w = 2x_2 + 2x_1 + 1x_3 - 4,$$

$$w = 8x_2 + 12x_1 + 6x_3 - 48,$$

$$w = 4x_2 + 4x_1 + 3x_3 - 16,$$

$$w = 4x_2 + 6x_1 + 2x_3 - 16,$$

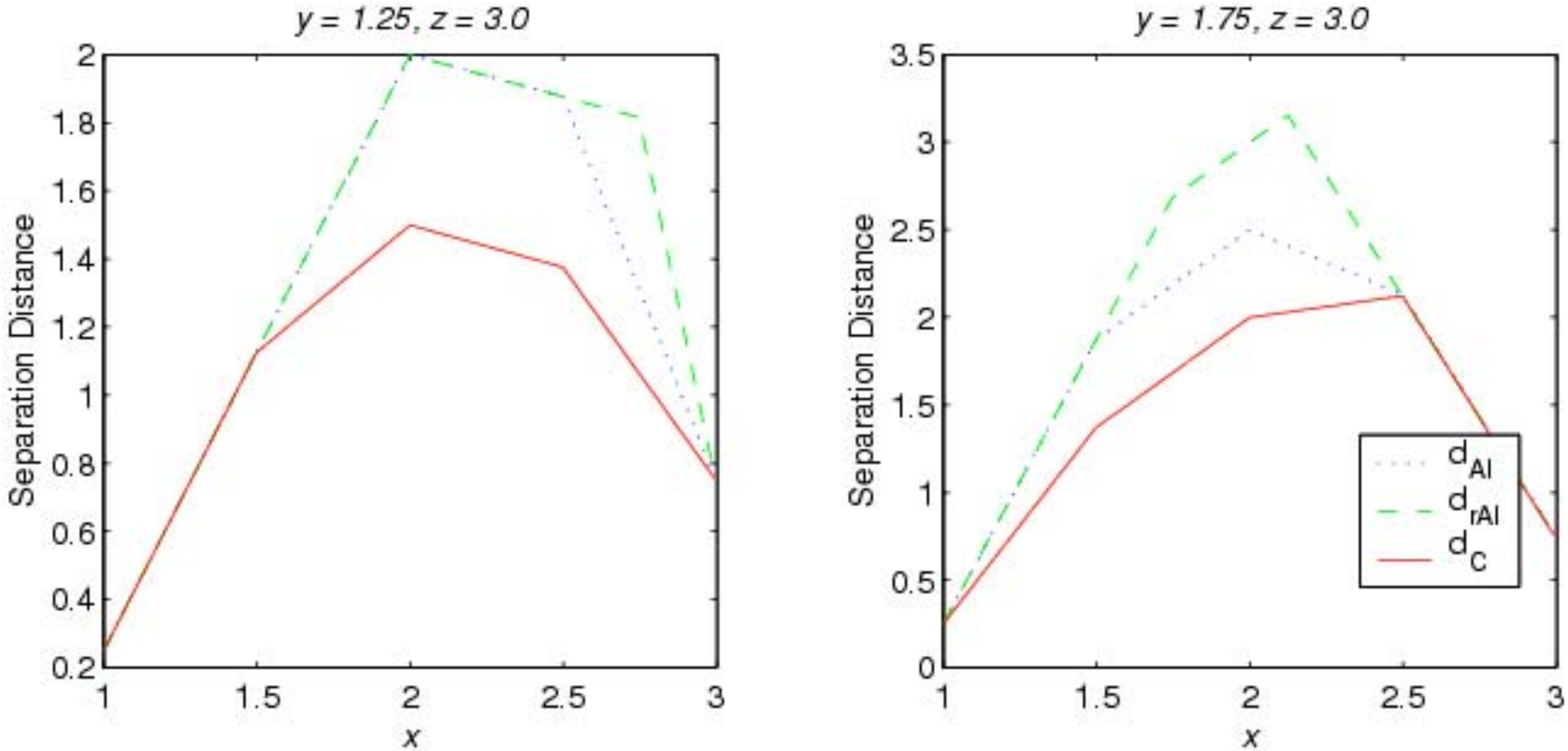
$$w = 5x_2 + 6x_1 + 3x_3 - 21,$$

$$w = 3x_2 + 4x_1 + 2x_3 - 11.$$

Comparison with Lower Bounding Approximations

The separation distance between the function *xyz* and the convex envelope (d_C) is compared with the separation distance between *xyz* and:

- the Arithmetic Interval lower bounding approximation (d_{AI}) and,
- the Recursive Arithmetic Interval lower bounding approximation (d_{rAI}).



Convex Envelopes for Edge-Concave Functions

(Meyer and Floudas, *Math. Programming*, 2005)

Definition:

Edge-concave functions are a class of functions that admit a **vertex polyhedral convex envelope** (Tardella, 2008)

Several classes of functions are edge-concave on certain domains:

- Concave functions over polytopes
- Multilinear functions over hypercubes (Rikun, 1997)

Theorem (Tardella, 2003): Function f(**x**) defined on a box is **edge-concave iff it is componentwise concave**. When f(**x**) is also twice continuously differentiable, **edge-concavity** is equivalent to:

$$f_{x_i,x_i}(\mathbf{x}) \leq 0 \quad \forall i = 1, \cdots, n$$

Convex Envelopes for Edge-Concave Functions

(Meyer and Floudas, *Math. Programming*, 2005)

Edge-concave function $f : conv(V) \rightarrow R$

Set of vertices of hyperrectangle $V = \{x^1, x^2, ..., x^{2^n}\} \subseteq R^n$

ALGORITHM

Step1: Dominance Relations

Evaluate function at each vertex point x^i
and determine the dominant subsets $X = \{x^i : i \in P(\lambda)\} \subseteq V$

Convex Envelopes for Edge-Concave Functions

(Meyer and Floudas, *Math. Programming*, 2005)

Edge-concave function $f : conv(V) \rightarrow R$

Set of vertices of hyperrectangle $V = \{x^1, x^2, \ldots, x^{2^n}\} \subseteq R^n$

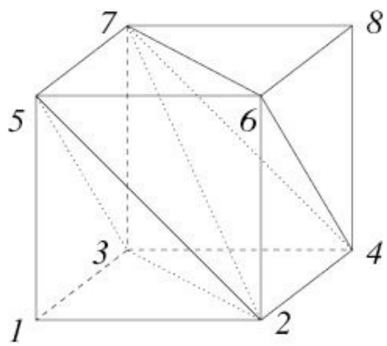
ALGORITHM

Step1: Dominance Relations

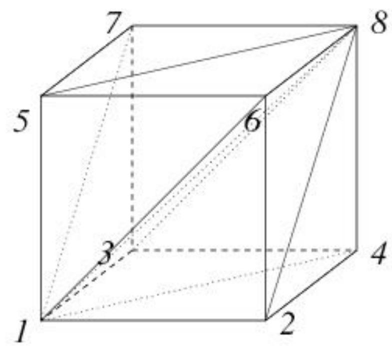
Evaluate function at each vertex point x^i and determine the dominant subsets $X = \{x^i : i \in P(\lambda)\} \subseteq V$

Step2: Triangulation Class

Determine the triangulation type
(6 triangulation types for 3D cube)



(a) Triangulation type **A**



(b) Triangulation type **B**

Convex Envelopes for Edge-Concave Functions

(Meyer and Floudas, *Math. Programming*, 2005)

Edge-concave function $f : conv(V) \to R$

Set of vertices of hyperrectangle $V = \{x^1, x^2, \ldots, x^{2^n}\} \subseteq R^n$

ALGORITHM

Step1: Dominance Relations	Evaluate function at each vertex point x^i and determine the dominant subsets $X = \{x^i : i \in P(\lambda)\} \subseteq V$
Step2: Triangulation Class	Determine the triangulation type (6 triangulation types for 3D cube)
Step3: Reorientation	Apply Transformation: Representative triangulation → Current triangulation

Convex Envelopes for Edge-Concave Functions

(Meyer and Floudas, *Math. Programming*, 2005)

Edge-concave function $f : conv(V) \to R$

Set of vertices of hyperrectangle $V = \{x^1, x^2, \ldots, x^{2^n}\} \subseteq R^n$

ALGORITHM

Step1: Dominance Relations	Evaluate function at each vertex point x^i and determine the dominant subsets $X = \{x^i : i \in P(\lambda)\} \subseteq V$
Step2: Triangulation Class	Determine the triangulation type (6 triangulation types for 3D cube)
Step3: Reorientation	Apply Transformation: Representative triangulation → Current triangulation
Step4: Compute Facets	Calculate FDH from the cells of the current triangulation

Solve linear system of equations:

$$\begin{bmatrix} 1 & x_1^{i_1} & x_2^{i_1} & x_3^{i_1} \\ 1 & x_1^{i_2} & x_2^{i_2} & x_3^{i_2} \\ 1 & x_1^{i_3} & x_2^{i_3} & x_3^{i_3} \\ 1 & x_1^{i_4} & x_2^{i_4} & x_3^{i_4} \end{bmatrix} \begin{bmatrix} \pi_0 \\ \pi_1 \\ \pi_2 \\ \pi_3 \end{bmatrix} = \begin{bmatrix} f(x^{i_1}) \\ f(x^{i_2}) \\ f(x^{i_3}) \\ f(x^{i_4}) \end{bmatrix}$$

FDH is: $w = \langle \pi, x \rangle + \pi_0$

Convex Envelopes for Edge-Concave Functions

(Meyer and Floudas, *Math. Programming*, 2005)

Consider function f(**x**):

$$\begin{aligned} f(x_1, x_2, x_3) &= 2 \cdot x_1 \cdot x_2 + 2 \cdot x_1 \cdot x_3 + 2 \cdot x_2 \cdot x_3 - 0.2 \cdot x_1 \cdot x_2 \cdot x_3 \\ x_i &\in [-10, 10], \; i = 1, 2, 3 \end{aligned}$$

The function must be **edge-concave** because:

$$\frac{\partial^2 f}{\partial x_1^2} = \frac{\partial^2 f}{\partial x_2^2} = \frac{\partial^2 f}{\partial x_3^2} = 0$$

Convex Envelopes for Edge-Concave Functions

(Meyer and Floudas, *Math. Programming*, 2005)

$$f(x_1, x_2, x_3) = 2 \cdot x_1 \cdot x_2 + 2 \cdot x_1 \cdot x_3 + 2 \cdot x_2 \cdot x_3 - 0.2 \cdot x_1 \cdot x_2 \cdot x_3$$
$$x_i \in [-10,\ 10],\ i = 1, 2, 3$$

ALGORITHM

Step1: Dominance Relations	Function is perturbed so that non-dominated and dominated subsets coincide
Step2: Triangulation Class	Dominance relations match vertex pattern of triangulation type A
Step3: Reorientation	Standard vertex orientation [1 2 3 4 5 6 7 8] becomes problem-specific orientation [1 3 2 4 5 7 6 8]
Step4: Compute Facets	FDH is calculated using cells of triangulation A:

$$\begin{aligned}
f(\mathbf{x}) &\geq -16 + 8 \cdot x_1 + 10.4 \cdot x_2 + 4 \cdot x_3 \\
f(\mathbf{x}) &\geq -80 + 16 \cdot x_1 + 12.8 \cdot x_2 + 10 \cdot x_3 \\
f(\mathbf{x}) &\geq -180 + 20 \cdot x_1 + 18 \cdot x_2 + 20 \cdot x_3 \\
f(\mathbf{x}) &\geq -160 + 16 \cdot x_1 + 16.8 \cdot x_2 + 20 \cdot x_3 \\
f(\mathbf{x}) &\geq -100 + 20 \cdot x_1 + 14 \cdot x_2 + 10 \cdot x_3
\end{aligned}$$

Convex Envelopes for Edge-Concave Functions

(Meyer and Floudas, *Math. Programming*, 2005)

$$
\begin{aligned}
f(x_1, x_2, x_3) &= 2 \cdot x_1 \cdot x_2 + 2 \cdot x_1 \cdot x_3 + 2 \cdot x_2 \cdot x_3 - 0.2 \cdot x_1 \cdot x_2 \cdot x_3 \\
x_i &\in [-10,\, 10]\,,\; i = 1, 2, 3
\end{aligned}
$$

$$
\min f(x_1, x_2, x_3) = f(-10, 10, -10) = f(-10, -10, 10) = f(10, -10, -10) = -400
$$

Comparison of underestimation techniques:	Lower Bnd	CPU
Edge-concave technique (Meyer and Floudas, 2005):	-400	0.01 s (GAMS)
Recursive arithmetic intervals (Maranas and Floudas, 1995):	-600	0.01 s (GAMS)
Second-order semidefinite relaxation (Henrion et al., 2007):	-405.72	0.22 s (Matlab)
Third-order semidefinite relaxation (Henrion et al., 2007):	-400	0.40 s (Matlab)

Convex Envelopes for Edge-Concave Functions

(Meyer and Floudas, *Math. Programming*, 2005)

Now consider function g(**x**):

$$\begin{aligned} g(x_1, x_2, x_3) \; = \; & 1.7502 \cdot x_1 - 0.6031 \cdot x_1 \cdot x_2 - 0.0403 \cdot x_1 \cdot x_3 + 0.0738 \cdot x_1 \cdot x_2^2 + \\ & 0.0116 \cdot x_1 \cdot x_2 \cdot x_3 - 0.0026 \cdot x_1 \cdot x_2^3 - 0.0010 \cdot x_1 \cdot x_2^2 \cdot x_3 \end{aligned}$$

$$x_1 \in [0,\, 2]\,,\; x_2 \in [6.4,\, 10]\,,\; x_3 \in [0,\, 5]$$

g(x) is not edge-concave because one of the partials is sometimes greater than 0:

$$\frac{\partial^2 g}{\partial x_2^2} \; = \; 2 \cdot 0.0738 \cdot x_1 + 6 \cdot (-0.0026) \cdot x_1 \cdot x_2 + 2 \cdot (-0.0010) \cdot x_1 \cdot x_3 \nleq 0$$

But g(**x**) can be written as the sum of an **edge-concave** function and an extra term:

$$\begin{aligned} g(x_1, x_2, x_3) \; = \; & (1.7502 \cdot x_1 - 0.6031 \cdot x_1 \cdot x_2 - 0.0403 \cdot x_1 \cdot x_3 + 0.0490 \cdot x_1 \cdot x_2^2 + \\ & 0.0116 \cdot x_1 \cdot x_2 \cdot x_3 - 0.0026 \cdot x_1 \cdot x_2^3 - 0.0010 \cdot x_1 \cdot x_2^2 \cdot x_3) + 0.0248 \cdot x_1 \cdot x_2^2 \end{aligned}$$

Convex Envelopes for Edge-Concave Functions

(Meyer and Floudas, *Math. Programming*, 2005)

$$g(x_1, x_2, x_3) = (1.7502 \cdot x_1 - 0.6031 \cdot x_1 \cdot x_2 - 0.0403 \cdot x_1 \cdot x_3 + 0.0490 \cdot x_1 \cdot x_2^2 + 0.0116 \cdot x_1 \cdot x_2 \cdot x_3 - 0.0026 \cdot x_1 \cdot x_2^3 - 0.0010 \cdot x_1 \cdot x_2^2 \cdot x_3) + 0.0248 \cdot x_1 \cdot x_2^2$$

$$x_1 \in [0,\ 2],\ x_2 \in [6.4,\ 10],\ x_3 \in [0,\ 5]$$

Comparison of underestimation techniques:

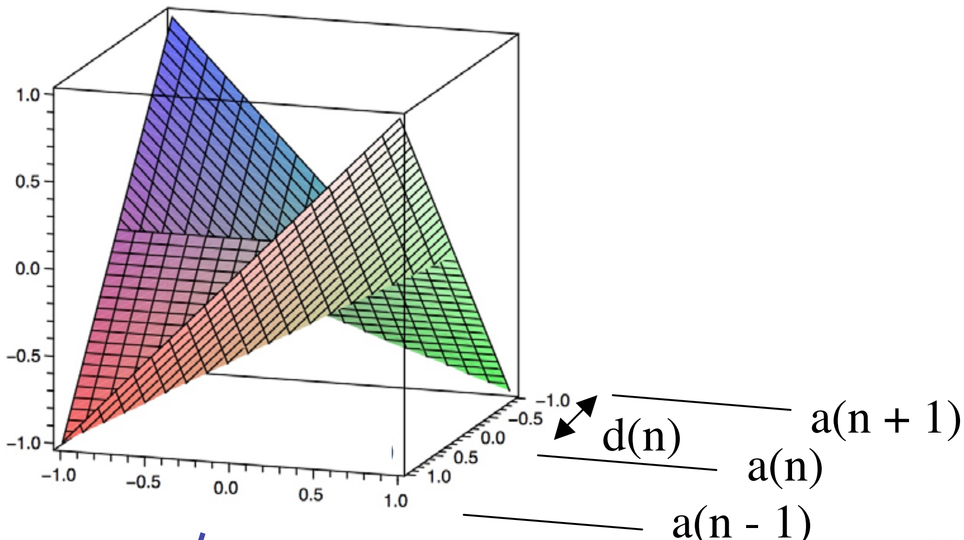
	Lower Bnd	CPU
Global solution	**0**	
Edge-concave algorithm with an extra term underestimated using recursive arithmetic (Meyer and Floudas, 2005) **:**	**-10.61**	**0.01 s (GAMS)**
Recursive arithmetic intervals only (Maranas and Floudas, 1995) **:**	**-13.56**	**0.01 s (GAMS)**
Second-order semidefinite relaxation (Henrion et al., 2007) **:**	**-infinity**	**0.56 s (Matlab)**
Third-order semidefinite relaxation (Henrion et al., 2007) **:**	**0**	**0.44 s (Matlab)**

Outline

- **Deterministic Global Optimization**: Objectives & Motivation
- Convex Envelopes:
    - Trilinear Monomials
    - Edge Concave functions
- **Piecewise linearization of Bilinear terms**
- Checking Convexity: Products of Univariate Functions
- PαBB: Piecewise Quadratic Perturbations
- Pooling Problems: Standard, Generalized & Extended
- Conclusions

Relaxation Development: Disjunctive Formulation

Balas [1979]; Floudas [1995]; Wicaksono & Karimi, *AIChE J*. [2008]



- Partition x-variables
- **Allow exactly one active relaxation for each domain point**

$$\bigvee_{n} \left[\begin{array}{l} W(n) \\ z \geq x \cdot y^L + a(n) \cdot y - a(n) \cdot y^L \\ z \geq x \cdot y^U + a(n+1) \cdot y - a(n+1) \cdot y^U \\ z \leq x \cdot y^L + a(n+1) \cdot y - a(n+1) \cdot y^L \\ z \leq x \cdot y^U + a(n) \cdot y - a(n) \cdot y^U \\ a(n) \leq x \leq a(n+1) \\ y^L \leq y \leq y^U \end{array} \right]$$

Piecewise Relaxation of Bilinear Programs

- 10 relaxation schemes from Wicaksono & Karimi, *AIChE J.* [2008] & 5 additional schemes from Gounaris, Misener, Floudas, *Ind. Eng. Chem. Res.* [2009] using *ab initio* domain partitioning
- 3 formulation classes:
  - big-M
  - convex hull
  - incremental cost
- Multiple design choices:
  - choice of which variable to partition
  - number of partition segments
  - uniform grid or not

Piecewise Relaxation of Bilinear Programs

- 10 relaxation schemes from Wicaksono & Karimi, *AIChE J.* [2008] & 5 additional schemes from Gounaris et al., *Ind. Eng. Chem. Res.* [2009] using *ab initio* domain partitioning
- **3 formulation classes:**
  - **big-M**
  - **convex hull**
  - **incremental cost**
- Multiple design choices:
  - choice of which variable to partition
  - number of partition segments
  - uniform grid or not

Relaxation Development: Big-M Reformulation

Meyer & Floudas, *AIChE J*. [2006]; Wicaksono & Karimi, *AIChE J*. [2008]

$$\bigvee_n \begin{bmatrix} W(n) \\ z \geq x \cdot y^L + a(n) \cdot y - a(n) \cdot y^L \\ z \geq x \cdot y^U + a(n+1) \cdot y - a(n+1) \cdot y^U \\ z \leq x \cdot y^L + a(n+1) \cdot y - a(n+1) \cdot y^L \\ z \leq x \cdot y^U + a(n) \cdot y - a(n) \cdot y^U \\ a(n) \leq x \leq a(n+1) \\ y^L \leq y \leq y^U \end{bmatrix}$$

$$\lambda(n) = \begin{cases} 1 & \text{if } a(n) \leq x \leq a(n+1) \\ 0 & \text{else} \end{cases} \qquad \forall\, n = 1, \cdots, N$$

$$\textstyle\sum_{n=1}^{N} \lambda(n) = 1$$

$$a(n) \cdot \lambda(n) + x^L\,[1 - \lambda(n)] \leq x \leq a(n+1) \cdot \lambda(n) + x^U\,[1 - \lambda(n)] \quad \forall\, n = 1, \cdots, N$$

$$z \geq x \cdot y^L + a(n) \cdot y - a(n) \cdot y^L - M \cdot [1 - \lambda(n)] \qquad \forall\, n = 1, \cdots, N$$
$$z \geq x \cdot y^U + a(n+1) \cdot y - a(n+1) \cdot y^U - M \cdot [1 - \lambda(n)] \qquad \forall\, n = 1, \cdots, N$$
$$z \leq x \cdot y^L + a(n+1) \cdot y - a(n+1) \cdot y^L + M \cdot [1 - \lambda(n)] \qquad \forall\, n = 1, \cdots, N$$
$$z \leq x \cdot y^U + a(n) \cdot y - a(n) \cdot y^U + M \cdot [1 - \lambda(n)] \qquad \forall\, n = 1, \cdots, N$$

$$x^L \leq x \leq x^U; \quad y^L \leq y \leq y^U$$

Relaxation Development: Big-M Reformulation

Meyer & Floudas, *AIChE J.* [2006]; Wicaksono & Karimi, *AIChE J.* [2008]

$$\bigvee_n \begin{bmatrix} W(n) \\ z \geq x \cdot y^L + a(n) \cdot y - a(n) \cdot y^L \\ z \geq x \cdot y^U + a(n+1) \cdot y - a(n+1) \cdot y^U \\ z \leq x \cdot y^L + a(n+1) \cdot y - a(n+1) \cdot y^L \\ z \leq x \cdot y^U + a(n) \cdot y - a(n) \cdot y^U \\ a(n) \leq x \leq a(n+1) \\ y^L \leq y \leq y^U \end{bmatrix}$$

$$\lambda(n) = \begin{cases} 1 & \text{if } a(n) \leq x \leq a(n+1) \\ 0 & \text{else} \end{cases} \qquad \forall\, n = 1, \cdots, N$$

$$\sum_{n=1}^{N} \lambda(n) = 1$$

$$a(n) \cdot \lambda(n) + x^L \left[1 - \lambda(n)\right] \leq x \leq a(n+1) \cdot \lambda(n) + x^U \left[1 - \lambda(n)\right] \quad \forall\, n = 1, \cdots, N$$

$$z \geq x \cdot y^L + a(n) \cdot y - a(n) \cdot y^L - M \cdot [1 - \lambda(n)] \qquad \forall\, n = 1, \cdots, N$$

$$z \geq x \cdot y^U + a(n+1) \cdot y - a(n+1) \cdot y^U - M \cdot [1 - \lambda(n)] \qquad \forall\, n = 1, \cdots, N$$

$$z \leq x \cdot y^L + a(n+1) \cdot y - a(n+1) \cdot y^L + M \cdot [1 - \lambda(n)] \qquad \forall\, n = 1, \cdots, N$$

$$z \leq x \cdot y^U + a(n) \cdot y - a(n) \cdot y^U + M \cdot [1 - \lambda(n)] \qquad \forall\, n = 1, \cdots, N$$

$$x^L \leq x \leq x^U; \quad y^L \leq y \leq y^U$$

Relaxation Development:
Big-M Reformulation

Meyer & Floudas, *AIChE J.* [2006]; Wicaksono & Karimi, *AIChE J.* [2008]

$$\bigvee_n \begin{bmatrix} W(n) \\ z \geq x \cdot y^L + a(n) \cdot y - a(n) \cdot y^L \\ z \geq x \cdot y^U + a(n+1) \cdot y - a(n+1) \cdot y^U \\ z \leq x \cdot y^L + a(n+1) \cdot y - a(n+1) \cdot y^L \\ z \leq x \cdot y^U + a(n) \cdot y - a(n) \cdot y^U \\ a(n) \leq x \leq a(n+1) \\ y^L \leq y \leq y^U \end{bmatrix}$$

$$\lambda(n) = \begin{cases} 1 & \text{if } a(n) \leq x \leq a(n+1) \\ 0 & \text{else} \end{cases} \qquad \forall\, n = 1, \cdots, N$$

$$\sum_{n=1}^{N} \lambda(n) = 1$$

$$a(n) \cdot \lambda(n) + x^L\,[1 - \lambda(n)] \leq x \leq a(n+1) \cdot \lambda(n) + x^U\,[1 - \lambda(n)] \quad \forall\, n = 1, \cdots, N$$

$$z \geq x \cdot y^L + a(n) \cdot y - a(n) \cdot y^L - M \cdot [1 - \lambda(n)] \qquad \forall\, n = 1, \cdots, N$$

$$z \geq x \cdot y^U + a(n+1) \cdot y - a(n+1) \cdot y^U - M \cdot [1 - \lambda(n)] \qquad \forall\, n = 1, \cdots, N$$

$$z \leq x \cdot y^L + a(n+1) \cdot y - a(n+1) \cdot y^L + M \cdot [1 - \lambda(n)] \qquad \forall\, n = 1, \cdots, N$$

$$z \leq x \cdot y^U + a(n) \cdot y - a(n) \cdot y^U + M \cdot [1 - \lambda(n)] \qquad \forall\, n = 1, \cdots, N$$

$$x^L \leq x \leq x^U; \quad y^L \leq y \leq y^U$$

Relaxation Development:
Convex Hull Reformulation

Karuppiah & Grossmann, *Comput. Chem. Eng.* [2006]; Wicaksono & Karimi, *AIChE J.* [2008]

$$\bigvee_n \begin{bmatrix} W(n) \\ z \geq x \cdot y^L + a(n) \cdot y - a(n) \cdot y^L \\ z \geq x \cdot y^U + a(n+1) \cdot y - a(n+1) \cdot y^U \\ z \leq x \cdot y^L + a(n+1) \cdot y - a(n+1) \cdot y^L \\ z \leq x \cdot y^U + a(n) \cdot y - a(n) \cdot y^U \\ a(n) \leq x \leq a(n+1) \\ y^L \leq y \leq y^U \end{bmatrix}$$

$$\lambda(n) = \begin{cases} 1 & \text{if } a(n) \leq x \leq a(n+1) \\ 0 & \text{else} \end{cases} \qquad \forall\, n = 1, \cdots, N$$

$$\textstyle\sum_{n=1}^{N} \lambda(n) = 1$$

$$x = \textstyle\sum_{n=1}^{N} u(n) \quad \text{where} \quad a(n) \cdot \lambda(n) \leq u(n) \leq a(n+1) \cdot \lambda(n) \quad \forall\, n = 1, \cdots, N$$

$$y^L \cdot \lambda(n) \leq v(n) \leq y^U \cdot \lambda(n) \qquad \forall\, n = 1, \cdots, N$$

$$z \geq \textstyle\sum_{n=1}^{N} \left[u(n) \cdot y^L + a(n) \cdot v(n) - a(n) \cdot y^L \cdot \lambda(n)\right]$$
$$z \geq \textstyle\sum_{n=1}^{N} \left[u(n) \cdot y^U + a(n+1) \cdot v(n) - a(n+1) \cdot y^U \cdot \lambda(n)\right]$$
$$z \leq \textstyle\sum_{n=1}^{N} \left[u(n) \cdot y^L + a(n+1) \cdot v(n) - a(n+1) \cdot y^L \cdot \lambda(n)\right]$$
$$z \leq \textstyle\sum_{n=1}^{N} \left[u(n) \cdot y^U + a(n) \cdot v(n) - a(n) \cdot y^U \cdot \lambda(n)\right]$$

$$x^L \leq x \leq x^U; \quad y^L \leq y \leq y^U$$

Relaxation Development:
Convex Hull Reformulation

Karuppiah & Grossmann, *Comput. Chem. Eng.* [2006]; Wicaksono & Karimi, *AIChE J.* [2008]

$$\bigvee_n \begin{bmatrix} W(n) \\ z \geq x \cdot y^L + a(n) \cdot y - a(n) \cdot y^L \\ z \geq x \cdot y^U + a(n+1) \cdot y - a(n+1) \cdot y^U \\ z \leq x \cdot y^L + a(n+1) \cdot y - a(n+1) \cdot y^L \\ z \leq x \cdot y^U + a(n) \cdot y - a(n) \cdot y^U \\ a(n) \leq x \leq a(n+1) \\ y^L \leq y \leq y^U \end{bmatrix}$$

$$\lambda(n) = \begin{cases} 1 & \text{if } a(n) \leq x \leq a(n+1) \\ 0 & \text{else} \end{cases} \qquad \forall\, n = 1, \cdots, N$$

$$\textstyle\sum_{n=1}^{N} \lambda(n) = 1$$

$$x = \textstyle\sum_{n=1}^{N} u(n) \quad \text{where} \quad a(n) \cdot \lambda(n) \leq u(n) \leq a(n+1) \cdot \lambda(n) \qquad \forall\, n = 1, \cdots, N$$

$$y^L \cdot \lambda(n) \leq v(n) \leq y^U \cdot \lambda(n) \qquad \forall\, n = 1, \cdots, N$$

$$z \geq \textstyle\sum_{n=1}^{N} \left[u(n) \cdot y^L + a(n) \cdot v(n) - a(n) \cdot y^L \cdot \lambda(n)\right]$$

$$z \geq \textstyle\sum_{n=1}^{N} \left[u(n) \cdot y^U + a(n+1) \cdot v(n) - a(n+1) \cdot y^U \cdot \lambda(n)\right]$$

$$z \leq \textstyle\sum_{n=1}^{N} \left[u(n) \cdot y^L + a(n+1) \cdot v(n) - a(n+1) \cdot y^L \cdot \lambda(n)\right]$$

$$z \leq \textstyle\sum_{n=1}^{N} \left[u(n) \cdot y^U + a(n) \cdot v(n) - a(n) \cdot y^U \cdot \lambda(n)\right]$$

$$x^L \leq x \leq x^U; \quad y^L \leq y \leq y^U$$

Relaxation Development:

Convex Hull Reformulation

Karuppiah & Grossmann, *Comput. Chem. Eng.* [2006]; Wicaksono & Karimi, *AIChE J.* [2008]

$$\bigvee_n \begin{bmatrix} W(n) \\ z \geq x \cdot y^L + a(n) \cdot y - a(n) \cdot y^L \\ z \geq x \cdot y^U + a(n+1) \cdot y - a(n+1) \cdot y^U \\ z \leq x \cdot y^L + a(n+1) \cdot y - a(n+1) \cdot y^L \\ z \leq x \cdot y^U + a(n) \cdot y - a(n) \cdot y^U \\ a(n) \leq x \leq a(n+1) \\ y^L \leq y \leq y^U \end{bmatrix}$$

$$\lambda(n) = \begin{cases} 1 & \text{if } a(n) \leq x \leq a(n+1) \\ 0 & \text{else} \end{cases} \qquad \forall\, n = 1, \cdots, N$$

$$\textstyle\sum_{n=1}^{N} \lambda(n) = 1$$

$$x = \textstyle\sum_{n=1}^{N} u(n) \quad \text{where} \quad a(n) \cdot \lambda(n) \leq u(n) \leq a(n+1) \cdot \lambda(n) \qquad \forall\, n = 1, \cdots, N$$

$$y^L \cdot \lambda(n) \leq v(n) \leq y^U \cdot \lambda(n) \qquad \forall\, n = 1, \cdots, N$$

Activates one

$$z \geq \textstyle\sum_{n=1}^{N} \left[u(n) \cdot y^L + a(n) \cdot v(n) - a(n) \cdot y^L \cdot \lambda(n)\right]$$
$$z \geq \textstyle\sum_{n=1}^{N} \left[u(n) \cdot y^U + a(n+1) \cdot v(n) - a(n+1) \cdot y^U \cdot \lambda(n)\right]$$
$$z \leq \textstyle\sum_{n=1}^{N} \left[u(n) \cdot y^L + a(n+1) \cdot v(n) - a(n+1) \cdot y^L \cdot \lambda(n)\right]$$
$$z \leq \textstyle\sum_{n=1}^{N} \left[u(n) \cdot y^U + a(n) \cdot v(n) - a(n) \cdot y^U \cdot \lambda(n)\right]$$

$$x^L \leq x \leq x^U; \quad y^L \leq y \leq y^U$$

Formulating an Underestimator (nf4r)

nf4r

$$\lambda(n) = \begin{cases} 1 & \text{if } a(n) \le x \le a(n+1) \\ 0 & \text{else} \end{cases} \qquad \forall\, n = 1, \cdots, N$$

$$\textstyle\sum_{n=1}^{N} \lambda(n) = 1$$

$$\textstyle\sum_{n=1}^{N} a(n)\cdot\lambda(n) \le x \le \sum_{n=1}^{N} a(n+1)\cdot\lambda(n)$$

$$y = y^L + \textstyle\sum_{n=1}^{N} \Delta y(n) \quad \text{where } 0 \le \Delta y(n) \le (y^U - y^L)\cdot\lambda(n) \quad \forall\, n = 1, \cdots, N$$

$$z \ge y^L \cdot x + \textstyle\sum_{n=1}^{N} [a(n)\cdot \Delta y(n)]$$
$$z \ge y^U \cdot x + \textstyle\sum_{n=1}^{N} [a(n+1)\cdot [\Delta y(n) - (y^U - y^L)\cdot\lambda(n)]]$$
$$z \le y^L \cdot x + \textstyle\sum_{n=1}^{N} [a(n+1)\cdot \Delta y(n)]$$
$$z \le y^U \cdot x + \textstyle\sum_{n=1}^{N} [a(n)\cdot [\Delta y(n) - (y^U - y^L)\cdot\lambda(n)]]$$

$$x^L \le x \le x^U; \quad y^L \le y \le y^U$$

Gounaris et al., *Ind. Eng. Chem. Res.* [2009]

nf4

$$\lambda(n) = \begin{cases} 1 & \text{if } a(n) \le x \le a(n+1) \\ 0 & \text{else} \end{cases} \qquad \forall\, n = 1, \cdots, N$$

$$\textstyle\sum_{n=1}^{N} \lambda(n) = 1$$

$$x = \textstyle\sum_{n=1}^{N} [a(n)\cdot\lambda(n)] + \Delta x \quad \text{where } 0 \le \Delta x \le \sum_{n=1}^{N} [d(n)\cdot\lambda(n)]$$

$$y = y^L + \textstyle\sum_{n=1}^{N} \Delta y(n) \quad \text{where } 0 \le \Delta y(n) \le (y^U - y^L)\cdot\lambda(n) \qquad \forall\, n = 1, \cdots, N$$

$$z = y^L \cdot x + \textstyle\sum_{n=1}^{N} [a(n)\cdot \Delta y(n)] + \Delta z$$
$$\Delta z \ge 0$$
$$\Delta z \ge (y^U - y^L)\left[x - \textstyle\sum_{n=1}^{N} [a(n+1)\cdot\lambda(n)]\right] + \sum_{n=1}^{N} [d(n)\cdot \Delta y(n)]$$
$$\Delta z \le \Delta x \cdot (y^U - y^L)$$
$$\Delta z \le \textstyle\sum_{n=1}^{N} [d(n)\cdot \Delta y(n)]$$

$$x^L \le x \le x^U; \quad y^L \le y \le y^U$$

Wicaksono & Karimi, *AIChE J.* [2008]

Formulating an Underestimator (nf4r)

nf4r

$$\lambda(n) = \begin{cases} 1 & \text{if } a(n) \le x \le a(n+1) \\ 0 & \text{else} \end{cases} \qquad \forall\, n = 1, \cdots, N$$

$$\sum_{n=1}^{N} \lambda(n) = 1$$

$$\sum_{n=1}^{N} a(n) \cdot \lambda(n) \le x \le \sum_{n=1}^{N} a(n+1) \cdot \lambda(n)$$

$$y = y^L + \sum_{n=1}^{N} \Delta y(n) \quad \text{where } 0 \le \Delta y(n) \le (y^U - y^L) \cdot \lambda(n) \quad \forall\, n = 1, \cdots, N$$

$$z \ge y^L \cdot x + \sum_{n=1}^{N} [a(n) \cdot \Delta y(n)]$$
$$z \ge y^U \cdot x + \sum_{n=1}^{N} [a(n+1) \cdot [\Delta y(n) - (y^U - y^L) \cdot \lambda(n)]]$$
$$z \le y^L \cdot x + \sum_{n=1}^{N} [a(n+1) \cdot \Delta y(n)]$$
$$z \le y^U \cdot x + \sum_{n=1}^{N} [a(n) \cdot [\Delta y(n) - (y^U - y^L) \cdot \lambda(n)]]$$

$$x^L \le x \le x^U; \quad y^L \le y \le y^U$$

Gounaris et al., *Ind. Eng. Chem. Res.* [2009]

Eliminate Δz variable

nf4

$$\lambda(n) = \begin{cases} 1 & \text{if } a(n) \le x \le a(n+1) \\ 0 & \text{else} \end{cases} \qquad \forall\, n = 1, \cdots, N$$

$$\sum_{n=1}^{N} \lambda(n) = 1$$

$$x = \sum_{n=1}^{N} [a(n) \cdot \lambda(n)] + \Delta x \quad \text{where } 0 \le \Delta x \le \sum_{n=1}^{N} [d(n) \cdot \lambda(n)]$$

$$y = y^L + \sum_{n=1}^{N} \Delta y(n) \quad \text{where } 0 \le \Delta y(n) \le (y^U - y^L) \cdot \lambda(n) \qquad \forall\, n = 1, \cdots, N$$

$$z = y^L \cdot x + \sum_{n=1}^{N} [a(n) \cdot \Delta y(n)] + \Delta z$$
$$\Delta z \ge 0$$
$$\Delta z \ge (y^U - y^L) \left[x - \sum_{n=1}^{N} [a(n+1) \cdot \lambda(n)]\right] + \sum_{n=1}^{N} [d(n) \cdot \Delta y(n)]$$
$$\Delta z \le \Delta x \cdot (y^U - y^L)$$
$$\Delta z \le \sum_{n=1}^{N} [d(n) \cdot \Delta y(n)]$$

$$x^L \le x \le x^U; \quad y^L \le y \le y^U$$

Wicaksono & Karimi, *AIChE J.* [2008]

Relaxation Development:
Incremental Cost Reformulation

Wicaksono & Karimi, *AIChE J*. [2008]

$$\bigvee_n \begin{bmatrix} W(n) \\ z \geq x \cdot y^L + a(n) \cdot y - a(n) \cdot y^L \\ z \geq x \cdot y^U + a(n+1) \cdot y - a(n+1) \cdot y^U \\ z \leq x \cdot y^L + a(n+1) \cdot y - a(n+1) \cdot y^L \\ z \leq x \cdot y^U + a(n) \cdot y - a(n) \cdot y^U \\ a(n) \leq x \leq a(n+1) \\ y^L \leq y \leq y^U \end{bmatrix}$$

$$\theta(n) \begin{cases} 1 & \text{if } x \geq a(n+1) \\ 0 & \text{else} \end{cases} \qquad 1 \leq n \leq (N-1)$$

$$\theta(n) \geq \theta(n+1) \qquad 1 \leq n \leq (N-2)$$

$$x = x^L + \textstyle\sum_{n=1}^{N} [d(n) \cdot \Delta u(n)] \qquad 0 \leq \Delta u(n) \leq 1$$

$$0 \leq \Delta u(N) \leq \theta(N-1) \leq \Delta u(N-1) \leq \cdots \leq \Delta u(2) \leq \theta(1) \leq \Delta u(1) \leq 1$$

$$z = x^L \cdot y + (x - x^L) \cdot y^L + \textstyle\sum_{n=1}^{N} d(n) \cdot \Delta w(n)$$

$$\Delta w(n) \geq \Delta v(n) \qquad \forall\, n < N$$

$$\Delta w(1) \geq (y^U - y^L) \cdot \Delta u(1) + y - y^U$$

$$\Delta w(n) \geq (y^U - y^L) \cdot [\Delta u(n) - \theta(n-1)] + \Delta v(n-1) \qquad \forall\, n > 1$$

$$\Delta w(1) \leq y - y^L$$

$$\Delta w(n) \leq \Delta v(n-1) \qquad \forall\, n > 1$$

$$\Delta w(N) \leq (y^U - y^L) \cdot \Delta u(N)$$

$$\Delta w(n) \leq (y^U - y^L) \cdot [\Delta u(n) - \theta(n)] + \Delta v(n) \qquad \forall\, n < N$$

$$x^L \leq x \leq x^U; \quad y^L \leq y \leq y^U$$

Relaxation Development:
Incremental Cost Reformulation

Wicaksono & Karimi, *AIChE J*. [2008]

$$\bigvee_n \begin{bmatrix} W(n) \\ z \geq x \cdot y^L + a(n) \cdot y - a(n) \cdot y^L \\ z \geq x \cdot y^U + a(n+1) \cdot y - a(n+1) \cdot y^U \\ z \leq x \cdot y^L + a(n+1) \cdot y - a(n+1) \cdot y^L \\ z \leq x \cdot y^U + a(n) \cdot y - a(n) \cdot y^U \\ a(n) \leq x \leq a(n+1) \\ y^L \leq y \leq y^U \end{bmatrix}$$

Alternative binary variable

$$\theta(n) \begin{cases} 1 & \text{if } x \geq a(n+1) \\ 0 & \text{else} \end{cases} \qquad 1 \leq n \leq (N-1)$$

$$\theta(n) \geq \theta(n+1) \qquad 1 \leq n \leq (N-2)$$

$$x = x^L + \textstyle\sum_{n=1}^{N} [d(n) \cdot \Delta u(n)] \qquad 0 \leq \Delta u(n) \leq 1$$

$$0 \leq \Delta u(N) \leq \theta(N-1) \leq \Delta u(N-1) \leq \cdots \leq \Delta u(2) \leq \theta(1) \leq \Delta u(1) \leq 1$$

$$z = x^L \cdot y + (x - x^L) \cdot y^L + \textstyle\sum_{n=1}^{N} d(n) \cdot \Delta w(n)$$

$$\Delta w(n) \geq \Delta v(n) \qquad \forall\, n < N$$

$$\Delta w(1) \geq (y^U - y^L) \cdot \Delta u(1) + y - y^U$$

$$\Delta w(n) \geq (y^U - y^L) \cdot [\Delta u(n) - \theta(n-1)] + \Delta v(n-1) \qquad \forall\, n > 1$$

$$\Delta w(1) \leq y - y^L$$

$$\Delta w(n) \leq \Delta v(n-1) \qquad \forall\, n > 1$$

$$\Delta w(N) \leq (y^U - y^L) \cdot \Delta u(N)$$

$$\Delta w(n) \leq (y^U - y^L) \cdot [\Delta u(n) - \theta(n)] + \Delta v(n) \qquad \forall\, n < N$$

$$x^L \leq x \leq x^U; \quad y^L \leq y \leq y^U$$

Relaxation Development:
Incremental Cost Reformulation

Wicaksono & Karimi, *AIChE J.* [2008]

$$\bigvee_n \left[\begin{array}{l} W(n) \\ z \geq x \cdot y^L + a(n) \cdot y - a(n) \cdot y^L \\ z \geq x \cdot y^U + a(n+1) \cdot y - a(n+1) \cdot y^U \\ z \leq x \cdot y^L + a(n+1) \cdot y - a(n+1) \cdot y^L \\ z \leq x \cdot y^U + a(n) \cdot y - a(n) \cdot y^U \\ a(n) \leq x \leq a(n+1) \\ y^L \leq y \leq y^U \end{array} \right]$$

Uses 1 fewer dimension

$$\theta(n) \begin{cases} 1 & \text{if } x \geq a(n+1) \\ 0 & \text{else} \end{cases} \qquad 1 \leq n \leq (N-1)$$

$$\theta(n) \geq \theta(n+1) \qquad 1 \leq n \leq (N-2)$$

$$x = x^L + \textstyle\sum_{n=1}^{N} [d(n) \cdot \Delta u(n)] \qquad 0 \leq \Delta u(n) \leq 1$$

$$0 \leq \Delta u(N) \leq \theta(N-1) \leq \Delta u(N-1) \leq \cdots \leq \Delta u(2) \leq \theta(1) \leq \Delta u(1) \leq 1$$

$$z = x^L \cdot y + (x - x^L) \cdot y^L + \textstyle\sum_{n=1}^{N} d(n) \cdot \Delta w(n)$$

$$\Delta w(n) \geq \Delta v(n) \qquad \forall\, n < N$$

$$\Delta w(1) \geq (y^U - y^L) \cdot \Delta u(1) + y - y^U$$

$$\Delta w(n) \geq (y^U - y^L) \cdot [\Delta u(n) - \theta(n-1)] + \Delta v(n-1) \qquad \forall\, n > 1$$

$$\Delta w(1) \leq y - y^L$$

$$\Delta w(n) \leq \Delta v(n-1) \qquad \forall\, n > 1$$

$$\Delta w(N) \leq (y^U - y^L) \cdot \Delta u(N)$$

$$\Delta w(n) \leq (y^U - y^L) \cdot [\Delta u(n) - \theta(n)] + \Delta v(n) \qquad \forall\, n < N$$

$$x^L \leq x \leq x^U; \quad y^L \leq y \leq y^U$$

Relaxation Development:
Incremental Cost Reformulation

Wicaksono & Karimi, *AIChE J*. [2008]

$$\bigvee_n \begin{bmatrix} W(n) \\ z \geq x \cdot y^L + a(n) \cdot y - a(n) \cdot y^L \\ z \geq x \cdot y^U + a(n+1) \cdot y - a(n+1) \cdot y^U \\ z \leq x \cdot y^L + a(n+1) \cdot y - a(n+1) \cdot y^L \\ z \leq x \cdot y^U + a(n) \cdot y - a(n) \cdot y^U \\ a(n) \leq x \leq a(n+1) \\ y^L \leq y \leq y^U \end{bmatrix}$$

$$\theta(n) \begin{cases} 1 & \text{if } x \geq a(n+1) \\ 0 & \text{else} \end{cases} \qquad 1 \leq n \leq (N-1)$$

$$\theta(n) \geq \theta(n+1) \qquad 1 \leq n \leq (N-2)$$

And avoids disaggregating the variable y

$$x = x^L + \sum_{n=1}^{N} [d(n) \cdot \Delta u(n)] \qquad 0 \leq \Delta u(n) \leq 1$$

$$0 \leq \Delta u(N) \leq \theta(N-1) \leq \Delta u(N-1) \leq \cdots \leq \Delta u(2) \leq \theta(1) \leq \Delta u(1) \leq 1$$

$$z = x^L \cdot y + (x - x^L) \cdot y^L + \sum_{n=1}^{N} d(n) \cdot \Delta w(n)$$

$$\Delta w(n) \geq \Delta v(n) \qquad \forall\, n < N$$

$$\Delta w(1) \geq (y^U - y^L) \cdot \Delta u(1) + y - y^U$$

$$\Delta w(n) \geq (y^U - y^L) \cdot [\Delta u(n) - \theta(n-1)] + \Delta v(n-1) \qquad \forall\, n > 1$$

$$\Delta w(1) \leq y - y^L$$

$$\Delta w(n) \leq \Delta v(n-1) \qquad \forall\, n > 1$$

$$\Delta w(N) \leq (y^U - y^L) \cdot \Delta u(N)$$

$$\Delta w(n) \leq (y^U - y^L) \cdot [\Delta u(n) - \theta(n)] + \Delta v(n) \qquad \forall\, n < N$$

$$x^L \leq x \leq x^U; \quad y^L \leq y \leq y^U$$

Formulating an Underestimator (nf7r)

nf7r

$$\theta(n) \begin{cases} 1 & \text{if } x \geq a(n+1) \\ 0 & \text{else} \end{cases} \qquad 1 \leq n \leq (N-1)$$

$$\theta(n) \geq \theta(n+1) \qquad 1 \leq n \leq (N-2)$$

$$x^L + \sum_{n=1}^{N-1} d(n) \cdot \theta(n) \leq x \leq a(1) + \sum_{n=1}^{N-1} d(n+1) \cdot \theta(n)$$

$$0 \leq \Delta v(N-1) \leq \Delta v(N-2) \leq \cdots \leq \Delta v(2) \leq \Delta v(1) \leq y - y^L$$

$$\Delta v(1) \geq (y^U - y^L) \cdot \theta(1) + y - y^U$$
$$\Delta v(n) \geq (y^U - y^L) \cdot [\theta(n) - \theta(n-1)] + \Delta v(n-1) \qquad 2 \leq n \leq N-1$$
$$\Delta v(N-1) \leq (y^U - y^L) \cdot \theta(N-1)$$

$$z \geq y^L \cdot x + x^L \cdot y - x^L \cdot y^L + \sum_{n=1}^{N} [d(n) \cdot \Delta v(n)]$$
$$z \leq y^L \cdot x + x^L \cdot y - x^L \cdot y^L + d(1) \cdot (y - y^U) + \sum_{n=1}^{N-1} d(n+1) \cdot \Delta v(n)$$
$$z \leq y^U \cdot x + x^L \cdot y - x^L \cdot y^U + \sum_{n=1}^{N-1} d(n) \cdot \left[\Delta v(n) - (y^U - y^L) \cdot \theta(n)\right]$$
$$z \geq y^U \cdot x + x^L \cdot y - x^L \cdot y^U + d(1) \cdot (y - y^L) + \sum_{n=1}^{N-1} d(n+1) \cdot \left[\Delta v(n) - (y^U - y^L) \cdot \theta(n)\right]$$

Gounaris et al., *Ind. Eng. Chem. Res.* [2009]

nf7

$$\theta(n) \begin{cases} 1 & \text{if } x \geq a(n+1) \\ 0 & \text{else} \end{cases} \qquad 1 \leq n \leq (N-1)$$

$$\theta(n) \geq \theta(n+1) \qquad 1 \leq n \leq (N-2)$$

$$x = x^L + \sum_{n=1}^{N-1} [d(n) \cdot \theta(n)] + \Delta x$$

$$z = x \cdot y^L + x^L \cdot y - x^L \cdot y^L + \sum_{n=1}^{N-1} [d(n) \cdot \Delta v(n)] + \Delta w$$

$$0 \leq \Delta v(N-1) \leq \Delta v(N-2) \leq \cdots \leq \Delta v(2) \leq \Delta v(1) \leq y - y^L$$

$$\Delta v(1) \geq (y^U - y^L) \cdot \theta(1) + y - y^U$$
$$\Delta v(n) \geq (y^U - y^L) \cdot [\theta(n) - \theta(n-1)] + \Delta v(n-1) \qquad 2 \leq n \leq N-1$$
$$\Delta v(N-1) \leq (y^U - y^L) \cdot \theta(N-1)$$

$$0 \leq \Delta x \leq d(1) + \sum_{n=1}^{N-1} [\theta(n) \cdot (d(n+1) - d(n))]$$

$$0 \leq \Delta y \leq y^U - y^L$$

$$\Delta w(n) \leq d(1) \cdot (y - y^L) + \sum_{n=1}^{N-1} [\Delta v(n) \cdot (d(n+1) - d(n))]$$
$$\Delta w(n) \leq (y^U - y^L) \cdot \Delta x$$
$$\Delta w(n) \geq \Delta x \cdot (y^U - y^L) + d(1) \cdot (y - y^U) + \sum_{n=1}^{N-1} [(\Delta v(n) - (y^U - y^L) \cdot \theta(n)) \cdot (d(n+1) - d(n))]$$

Wicaksono & Karimi, *AIChE J.* [2008]

Formulating an Underestimator (nf7r)

nf7r

$$\theta(n) \begin{cases} 1 & \text{if } x \geq a(n+1) \\ 0 & \text{else} \end{cases} \qquad 1 \leq n \leq (N-1)$$

$$\theta(n) \geq \theta(n+1) \qquad 1 \leq n \leq (N-2)$$

$$x^L + \sum_{n=1}^{N-1} d(n) \cdot \theta(n) \leq x \leq a(1) + \sum_{n=1}^{N-1} d(n+1) \cdot \theta(n)$$

$$0 \leq \Delta v(N-1) \leq \Delta v(N-2) \leq \cdots \leq \Delta v(2) \leq \Delta v(1) \leq y - y^L$$

$$\Delta v(1) \geq (y^U - y^L) \cdot \theta(1) + y - y^U$$
$$\Delta v(n) \geq (y^U - y^L) \cdot [\theta(n) - \theta(n-1)] + \Delta v(n-1) \qquad 2 \leq n \leq N-1$$
$$\Delta v(N-1) \leq (y^U - y^L) \cdot \theta(N-1)$$

$$z \geq y^L \cdot x + x^L \cdot y - x^L \cdot y^L + \sum_{n=1}^{N} [d(n) \cdot \Delta v(n)]$$
$$z \leq y^L \cdot x + x^L \cdot y - x^L \cdot y^L + d(1) \cdot (y - y^U) + \sum_{n=1}^{N-1} d(n+1) \cdot \Delta v(n)$$
$$z \leq y^U \cdot x + x^L \cdot y - x^L \cdot y^U + \sum_{n=1}^{N-1} d(n) \cdot [\Delta v(n) - (y^U - y^L) \cdot \theta(n)]$$
$$z \geq y^U \cdot x + x^L \cdot y - x^L \cdot y^U + d(1) \cdot (y - y^L) + \sum_{n=1}^{N-1} d(n+1) \cdot [\Delta v(n) - (y^U - y^L) \cdot \theta(n)]$$

Gounaris et al., *Ind. Eng. Chem. Res.* [2009]

Remove Δx & Δw variables from the formulation

nf7

$$\theta(n) \begin{cases} 1 & \text{if } x \geq a(n+1) \\ 0 & \text{else} \end{cases} \qquad 1 \leq n \leq (N-1)$$

$$\theta(n) \geq \theta(n+1) \qquad 1 \leq n \leq (N-2)$$

$$x = x^L + \sum_{n=1}^{N-1} [d(n) \cdot \theta(n)] + \Delta x$$

$$z = x \cdot y^L + x^L \cdot y - x^L \cdot y^L + \sum_{n=1}^{N-1} [d(n) \cdot \Delta v(n)] + \Delta w$$

$$0 \leq \Delta v(N-1) \leq \Delta v(N-2) \leq \cdots \leq \Delta v(2) \leq \Delta v(1) \leq y - y^L$$

$$\Delta v(1) \geq (y^U - y^L) \cdot \theta(1) + y - y^U$$
$$\Delta v(n) \geq (y^U - y^L) \cdot [\theta(n) - \theta(n-1)] + \Delta v(n-1) \qquad 2 \leq n \leq N-1$$
$$\Delta v(N-1) \leq (y^U - y^L) \cdot \theta(N-1)$$

$$0 \leq \Delta x \leq d(1) + \sum_{n=1}^{N-1} [\theta(n) \cdot (d(n+1) - d(n))]$$

$$0 \leq \Delta y \leq y^U - y^L$$

$$\Delta w(n) \leq d(1) \cdot (y - y^L) + \sum_{n=1}^{N-1} [\Delta v(n) \cdot (d(n+1) - d(n))]$$
$$\Delta w(n) \leq (y^U - y^L) \cdot \Delta x$$
$$\Delta w(n) \geq \Delta x \cdot (y^U - y^L) + d(1) \cdot (y - y^U) + \sum_{n=1}^{N-1} [(\Delta v(n) - (y^U - y^L) \cdot \theta(n)) \cdot (d(n+1) - d(n))]$$

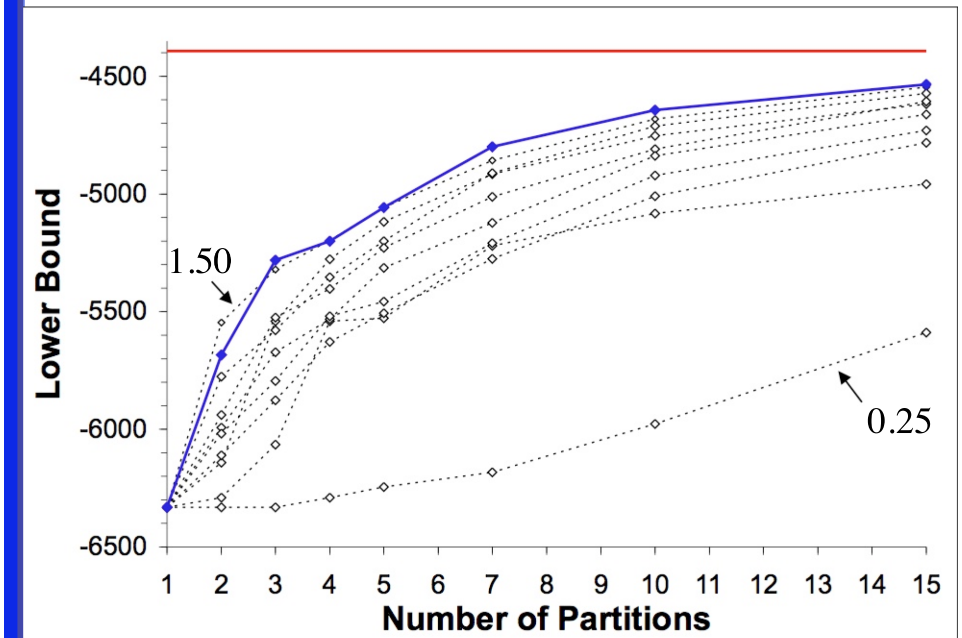
Wicaksono & Karimi, *AIChE J.* [2008]

Piecewise Relaxation of Bilinear Programs

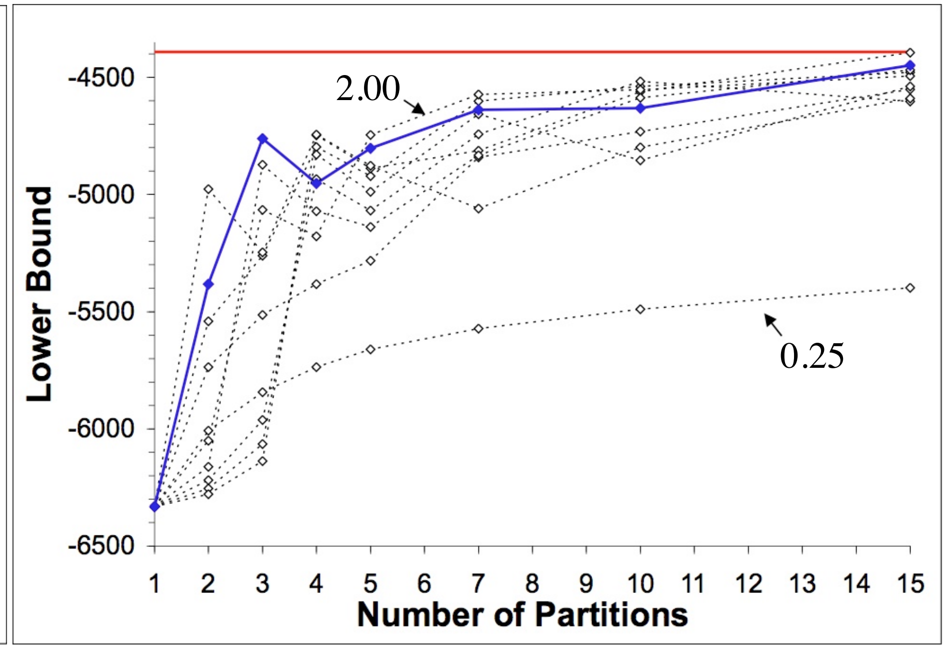
- 10 relaxation schemes from Wicaksono & Karimi, *AIChE J.* [2008] & 5 additional schemes from Gounaris et al., *Ind. Eng. Chem. Res.* [2009] using *ab initio* domain partitioning
- 3 formulation classes:
  - big-M
  - convex hull
  - incremental cost
- **Multiple design choices:**
  - **choice of which variable to partition**
  - **number of partition segments**
  - **uniform grid or not**

Applying Relaxation to a Representative Benchmark Problem [Audet et al., *Manag. Sci.* 2004]

‘y’- variant



‘p’- variant



Global minimum

Underestimate using uniform partitioning

Variable length partitioning controlled by parameter γ

[Wicaksono & Karimi, *AIChE J.*, 2008]:

$$x_n = x^L + \left(\frac{n}{N}\right)^{\gamma}(x^U - x^L)\ , \quad 0 \le n \le N$$

Comparison of the Relaxation Formulations

Number of runs* that failed to reach optimality within total CPU time limit of 1 h

Class	Formulation	Partitioning level N						
		2	3	4	5	7	10	15
Big–M	bm	–	–	11	20	33	66	140
	nf1	–	–	11	22	31	63	124
	nf2g	–	–	11	21	33	59	114
	nf2	–	–	10	21	31	62	118
Convex Combination	ch	–	–	1	5	8	18	43
	tch	–	–	1	5	12	27	71
	nf3	–	–	–	4	7	15	53
	nf4l	–	–	1	5	7	11	28
	nf4	–	–	–	4	7	10	25
	nf4r	–	–	1	5	7	8	23
Incremental Cost	nf5	–	–	–	4	6	9	32
	nf6	–	–	–	3	6	8	20
	nf6t	–	–	–	3	6	9	20
	nf7	–	–	–	–	5	8	19
	nf7r	–	–	–	1	6	8	16

*Out of a total of 480 runs for each entry.

To compare the formulations, finely partition the bilinear terms in the test case pooling problems and stress test the relaxation formulations to see which ones most often solve within a time limit.

Gounaris et al., *Ind. Eng. Chem. Res.* [2009]

Comparison of the Relaxation Formulations

Number of runs* that failed to reach optimality within total CPU time limit of 1 h

Class	Formulation	Partitioning level N						
		2	3	4	5	7	10	15
Big–M	bm	–	–	11	20	33	66	140
	nf1	–	–	11	22	31	63	124
	nf2g	–	–	11	21	33	59	114
	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	62	118
Convex Combination	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	18	43
	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	27	71
	nf3	–	–	–	4	7	15	53
	nf4l	–	–	1	5	7	11	28
	nf4	–	–	–	4	7	10	25
	nf4r	–	–	1	5	7	8	23
Incremental Cost	nf5	–	–	–	4	6	9	32
	nf6	–	–	–	3	6	8	20
	nf6t	–	–	–	3	6	9	20
	nf7	–	–	–	–	5	8	19
	nf7r	–	–	–	1	6	8	16

*Out of a total of 480 runs for each entry.

Use the most reliable formulations for large-scale pooling problems

Gounaris et al., *Ind. Eng. Chem. Res.* [2009]

To compare the formulations, finely partition the bilinear terms in the test case pooling problems and stress test the relaxation formulations to see which ones most often solve within a time limit.

Comparison of the Relaxation Formulations

Number of runs* that failed to reach optimality within total CPU time limit of 1 h

Class	Formulation	Partitioning level N						
		2	3	4	5	7	10	15
Big–M	bm	–	–	11	20	33	66	140
	nf1	–	–	11	22	31	63	124
	nf2g	–	–	11	21	33	59	114
	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	62	118
Convex Combination	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	18	43
	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	27	71
	nf3	–	–	–	4	7	15	53
	nf4l	–	–	1	5	7	11	28
	nf4	–	–	–	4	7	10	25
	nf4r	–	–	1	5	7	8	23
Incremental Cost	nf5	–	–	–	4	6	9	32
	nf6	–	–	–	3	6	8	20
	nf6t	–	–	–	3	6	9	20
	nf7	–	–	–	–	5	8	19
	nf7r	–	–	–	1	6	8	16

*Out of a total of 480 runs for each entry.

Use the most reliable formulations for large-scale pooling problems

To compare the formulations, finely partition the bilinear terms in the test case pooling problems and stress test the relaxation formulations to see which ones most often solve within a time limit.

Gounaris et al., *Ind. Eng. Chem. Res.* [2009]

Outline

- **Deterministic Global Optimization**: Objectives & Motivation
- Convex Envelopes:
  - Trilinear Monomials
  - Edge Concave functions
- Piecewise linearization of Bilinear terms
- **Checking Convexity: Products of Univariate Functions**
- PαBB: Piecewise Quadratic Perturbations
- Pooling Problems: Standard, Generalized & Extended
- Conclusions

Convexity of Products of Univariate Functions

(Gounaris and Floudas, *JOTA*, 2008)

$$f(\underline{x}) = \prod_{i=1}^{N} f_i(x_i) = f_1(x_1)\, f_2(x_2) \ldots .\, f_n(x_n)$$

When is f(x) convex?

Convexity of Products of Univariate Functions

(Gounaris and Floudas, *JOTA*, 2008)

$$f(\underline{x}) = \prod_{i=1}^{N} f_i(x_i) = f_1(x_1)\, f_2(x_2) \ldots.\, f_n(x_n)$$

When is f(x) convex?

Sufficient Conditions

- **Every factor should be strictly positive**
- **Every factor should be strictly convex**

} **An even number of factors are allowed to instead be strictly negative and strictly concave**

- **For every factor:** $f_i(x_i)\, f_i^{//}(x_i) - \left(f_i^{/}(x_i)\right)^2 \geq 0$

Convexity of Products of Univariate Functions

(Gounaris and Floudas, *JOTA*, 2008)

$$f(\underline{x}) = \prod_{i=1}^{N} f_i(x_i) = f_1(x_1)\, f_2(x_2) \ldots . f_n(x_n)$$

When is f(x) convex?

Sufficient Conditions

- **Every factor should be strictly positive**
- **Every factor should be strictly convex**

} **An even number of factors are allowed to instead be strictly negative and strictly concave**

- **For every factor**: $f_i(x_i)\, f_i^{//}(x_i) - \left(f_i^{/}(x_i)\right)^2 \geq 0$

These conditions are in fact necessary if all factors share the same functional form

Convexity of Products of Univariate Functions

(Gounaris and Floudas, *JOTA*, 2008)

$$f(\underline{x}) = \left\{ \frac{1.8}{x_2} + 1.2x_2 - \frac{3\log(1-x_1)}{x_2} - 2x_2\log(1-x_1) \right\} \frac{e^{x_3 - x_4}}{x_4^{1.2}}$$

Is f(x) convex in $\left[\frac{1}{3}, \frac{2}{3}\right]^4$?

Convexity of Products of Univariate Functions

(Gounaris and Floudas, *JOTA*, 2008)

$$f(\underline{x}) = \left\{ \frac{1.8}{x_2} + 1.2x_2 - \frac{3\log(1-x_1)}{x_2} - 2x_2\log(1-x_1) \right\} \frac{e^{x_3 - x_4}}{x_4^{1.2}}$$

$$= f_1(x_1)\, f_2(x_2)\, f_3(x_3)\, f_4(x_4)$$

Is f(x) convex in $\left[\frac{1}{3}, \frac{2}{3}\right]^4$?

$$f_1(x_1) = 0.6 - \log(1-x_1)$$

$$f_2(x_2) = \frac{2x_2^2 + 3}{x_2}$$

$$f_3(x_3) = e^{x_3}$$

$$f_4(x_4) = \frac{e^{-x_4}}{x_4^{1.2}}$$

Yes!

….because all four functions satisfy the sufficient conditions in [1/3,2/3]

Outline

- **Deterministic Global Optimization**: Objectives & Motivation
- Convex Envelopes:
  - Trilinear Monomials
  - Edge Concave functions
- Piecewise linearization of Bilinear terms
- Checking Convexity: Products of Univariate Functions
- **PαBB: Piecewise Quadratic Perturbations**
- Pooling Problems: Standard, Generalized & Extended
- Conclusions

PαBB:
Piecewise Quadratic Perturbations

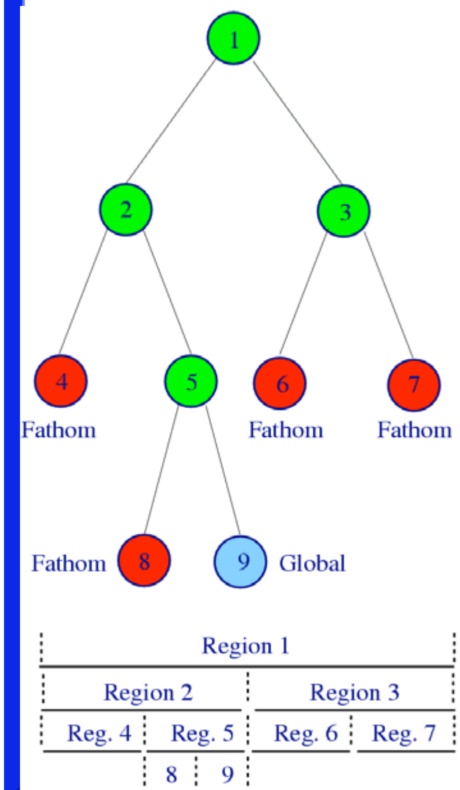
Christodoulos A. Floudas
Princeton University

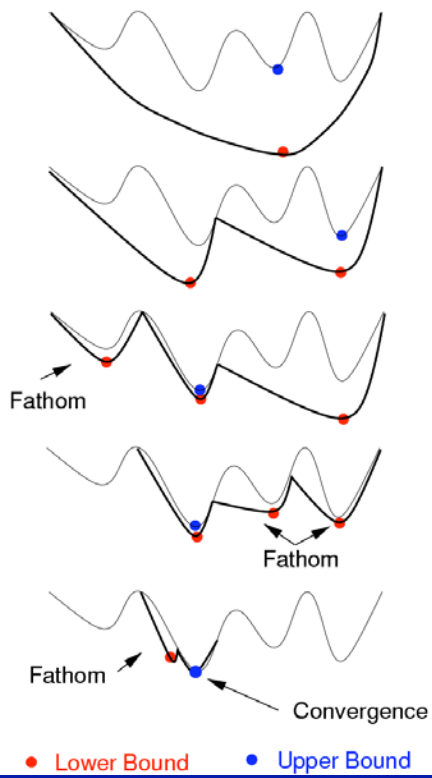
C^2 NLPs - The αBB Framework

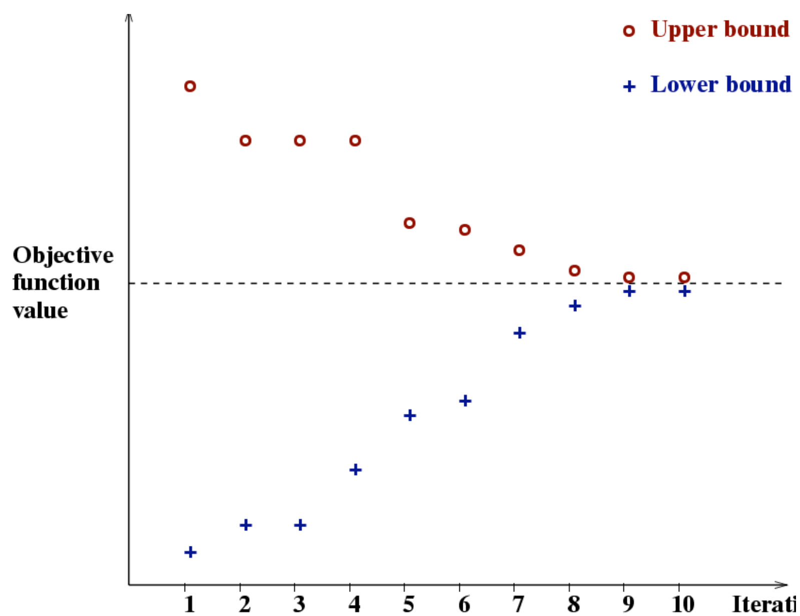
$$\begin{aligned} \min_{\mathbf{x}} \quad & f(\mathbf{x}) \\ \text{s.t.} \quad & \mathbf{h}(\mathbf{x}) = \mathbf{0} \\ & \mathbf{g}(\mathbf{x}) \le \mathbf{0} \\ & \mathbf{x} \in \mathbf{X} \subseteq \mathcal{R}^n \end{aligned}$$

f , $\mathbf{h}$, $\mathbf{g}$ twice continuously differentiable

- Based on a branch-and-bound framework
- Upper bound on the global solution is obtained by solving the full nonconvex problem to local optimality
- Lower bound is determined by solving a valid convex underestimation of the original problem
- Convergence is obtained by successive subdivision of the region at each level in the brand & bound tree
- Guaranteed ε-convergence for C^2 NLPs







Convex Lower Bounding: The αBB Framework

(Androulakis et al., *JOGO*,1995; Adjiman et al.,*Comp.&Chem.Eng*. 1998)

- Decompose each constraint into a sum of terms

$$f(x) = f_L(x) + f_C(x) + \sum_{i \in B} a_i x_1^i x_2^i + \sum_{i \in T} b_i x_1^i x_2^i x_3^i$$

LINEAR CONVEX BILINEAR TRILINEAR

$$+ \sum_{i \in F} c_i \frac{x_1^i}{x_2^i} + \sum_{i \in FT} d_i \frac{x_1^i x_2^i}{x_3^i} + \sum_{i \in S} f_S^i(x)$$

FRACTIONAL FRACTIONAL TRILINEAR SIGNOMIAL

$$+ \sum_{i \in UC} f_{UC}^i(x^i) + f_{GNC}(x)$$

UNIVARIATE CONCAVE GENERAL NONCONVEX

- Develop valid convex underestimators for each term

Convex Lower Bounding: The αBB Framework

Linear Terms
Convex Terms } → unchanged

Bilinear Terms (McCormick, 1976; Al Kayyal, Falk, 1983)

Define $w_B = x_1 x_2$ and introduce: Convex Envelope

$$
\begin{aligned}
w_B &\geq x_1^L x_2 + x_2^L x_1 - x_1^L x_2^L \\
w_B &\geq x_1^U x_2 + x_2^U x_1 - x_1^U x_2^U \\
w_B &\leq -x_1^U x_2 - x_2^L x_1 + x_1^U x_2^L \\
w_B &\leq -x_1^L x_2 - x_2^U x_1 + x_1^L x_2^U
\end{aligned}
$$

Key Property (Androulakis et al., 1995)

$$
\max_{x_1, x_2} (x_1 x_2 - w_B) = \frac{(x_1^U - x_1^L)(x_2^U - x_2^L)}{4}
$$

Convex Lower Bounding: The αBB Framework

General C^2 Nonconvex Terms

(Maranas, Floudas, 1994; Androulakis et. al, 1995)

$$L(x) = f_{GNC}(x) - \sum_{i=1}^{n} \alpha_i (x_i^U - x_i)(x_i - x_i^L)$$

$$\alpha \geq \max\left\{0, -\frac{1}{2} \min_{i, x^L \leq x \leq x^U} \lambda_{i,H}(x)\right\}$$

P1: $f_{GNC}(x) \geq L(x)$

P2: $f_{GNC}(x) = L(x)$ at corner points

P3: $L(x)$ is convex in $\left[x^L, x^U\right]$

P4: $L^{D_1}(x) \geq L^{D_2}(x)$ where $D_1 \subseteq D_2$

P5: Maximum Separation Distance

$$\max\,(f_{GNC}(x) - L(x)) = \frac{1}{4}\sum_{i=1}^{n} \alpha_i (x_i^U - x_i^L)^2$$

P6: Convexity of *L(x)*

$$L(x) \text{ is convex if } H_{GNC}(x) + 2Diag(\alpha_i)$$
$$\text{is positive semidefinite } \quad \forall\, \alpha \in \left[x^L, x^U\right]$$

Rigorous Calculations of α: The αBB Framework

(Adjiman, Floudas, 1996; Adjiman et al., 1998a,b)

Key Ideas

- Derive Hessian matrix, $H(x)$, of $f_{GNC}(x)$
- Compute INTERVAL Hessian in $[x^L, x^U]$

$$[H(x)]_{ij} = [h_{ij}^L(x), h_{ij}^U(x)]$$

- $H \subseteq [H]$
- Compute α : $[H] + 2Diag(\alpha)$ is P.S.D.

Uniform Diagonal Shift Matrix

O(*n*²) Methods

- Gerschgorin Theorem

O(*n*³) Methods

- Hertz
- Lower Bounding Hessian
- Mori-Kokane
- E-Matrix Approach

Non-Uniform Diagonal Shift Matrix

- **Scaled Gerschgorin Theorem**
- H-Matrix
- **Semi-definite Programming**

Scaled Gerschgorin Theorem: The αBB Framework

Gerschgorin Theorem for real matrices:

$$\lambda_{\min} \geq \min_i \left(h_{ii} - \sum_{j \neq i} |h_{ii}| \right)$$

Theorem for Interval Matrices (Adjiman et al., 1998a,b)

$$\alpha_i = \max\left[0, -\frac{1}{2}\left[h_{ii}^L - \sum_{j \neq i} \max\left(|h_{ii}^L|, |h_{ii}^U|\right)\frac{d_j}{d_i}\right]\right]$$

⬇

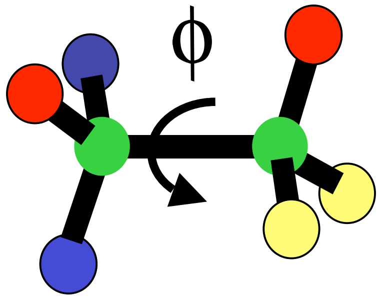
$$[H_{NT}] + 2Diag(\alpha_i) \text{ is positive semidefinite}$$

- *d* is a positive vector
  Use $d_i = 1$ or $d_i = x_i^U - x_i^L$

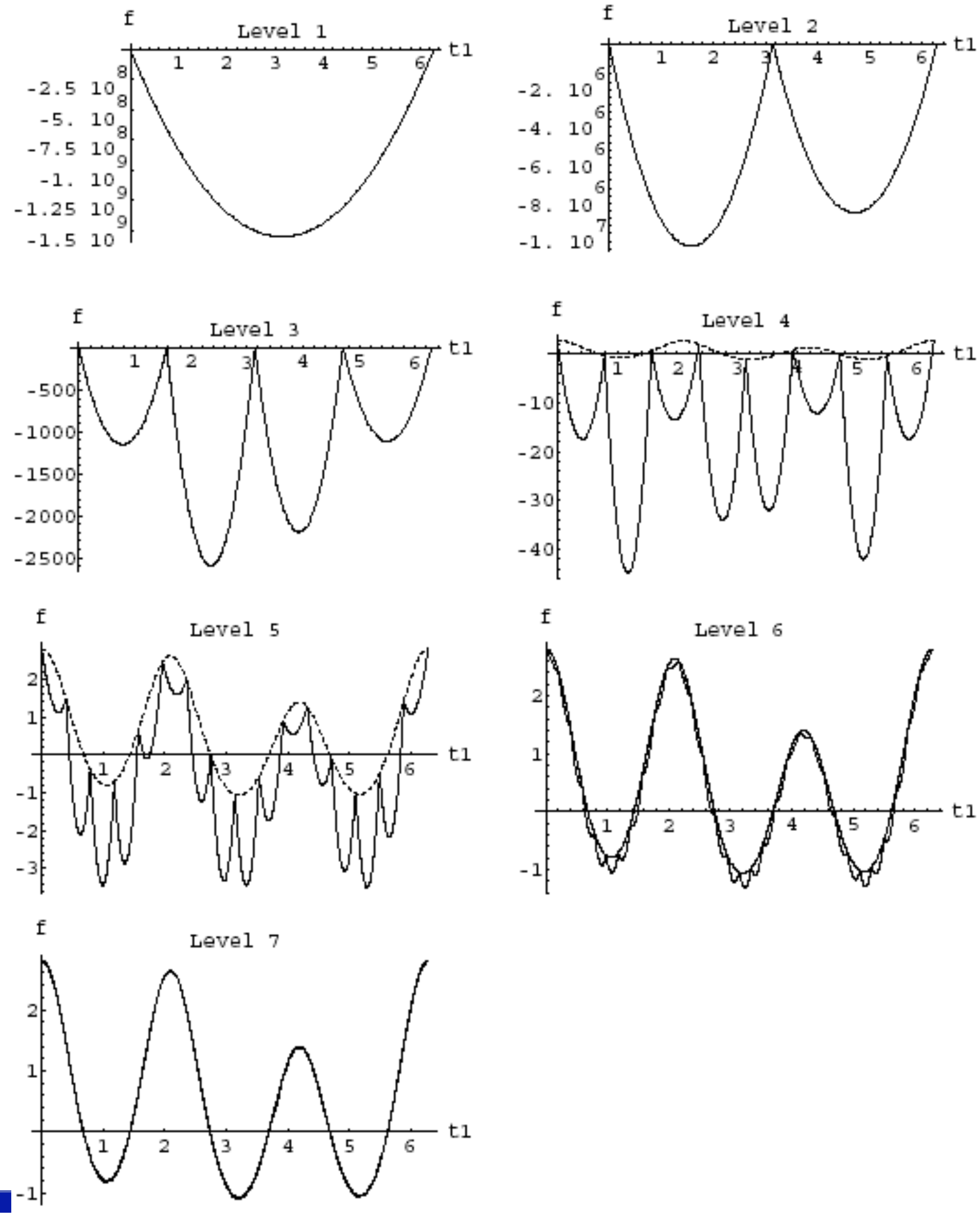
Inexpensive and simple technique

C2 NLPs - Illustrative Example

Pseudoethane



$$
\begin{aligned}
f(\phi) = {} & \frac{588600}{\left(3r^2 - 4\cos\theta\, r^2 - 2\left(\sin^2\theta\cos\left(\phi - \frac{2\pi}{3}\right) - \cos^2\theta\right) r^2\right)^6} \\
& - \frac{1079.1}{\left(3r^2 - 4\cos\theta\, r^2 - 2\left(\sin^2\theta\cos\left(\phi - \frac{2\pi}{3}\right) - \cos^2\theta\right) r^2\right)^3} \\
& + \frac{600800}{\left(3r^2 - 4\cos\theta\, r^2 - 2\left(\sin^2\theta\cos\phi - \cos^2\theta\right) r^2\right)^6} \\
& - \frac{1071.5}{\left(3r^2 - 4\cos\theta\, r^2 - 2\left(\sin^2\theta\cos\phi - \cos^2\theta\right) r^2\right)^3} \\
& + \frac{481300}{\left(3r^2 - 4\cos\theta\, r^2 - 2\left(\sin^2\theta\cos\left(\phi + \frac{2\pi}{3}\right) - \cos^2\theta\right) r^2\right)^6} \\
& - \frac{1064.6}{\left(3r^2 - 4\cos\theta\, r^2 - 2\left(\sin^2\theta\cos\left(\phi + \frac{2\pi}{3}\right) - \cos^2\theta\right) r^2\right)^3}
\end{aligned}
$$



αBB Underestimator: Room for Improvement?

$$q(x) = \sum_{i=1}^{n} \alpha_i \left(x_i - \underline{x}_i\right) \cdot \left(\overline{x}_i - x_i\right)$$

- Curvature of the perturbation function is constant.
- The eigenvectors of the Hessian matrix of the perturbation function are aligned with the coordinate axes.

A Refinement of the αBB Underestimator

Meyer, Floudas, *JOGO,* (2005)

Central Idea

- Partition the domain into subregions.
- Calculate the α parameters in each subregion.
- Construct an underestimator for the whole domain using these α's.

Properties of the Underestimator Function

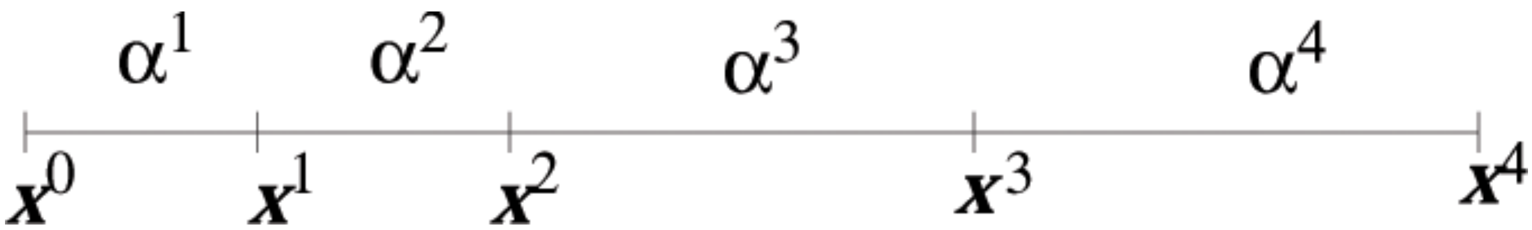
- smoothness
- convexity
- underestimation

Structure of the Underestimator Function

- sum of piecewise quadratic univariate functions
- underestimator matches function at vertices

Piecewise C²-Continuous Underestimator

- Partition interval $[\underline{x}_i, \overline{x}_i]$ into N_i subintervals.
- Endpoints of the subintervals: $x_i^0, x_i^1, \cdots, x_i^{N_i}$.



A smooth convex underestimator $f(x)$ in an interval $x \in [\underline{x}, \overline{x}]$:

$$\phi(x) := f(x) - q(x)$$

$$q(x) := \sum_{i=1}^{n} q_i^k(x_i) \qquad \text{for } x_i \in \left[x_i^{k-1}, x_i^k\right]$$

$$q_i^k(x_i) := \alpha_i^k \left(x_i - x_i^{k-1}\right) \cdot \left(x_i^k - x_i\right) + \beta_i^k x_i + \gamma_i^k$$

Joining the Pieces

- Smoothness: function q_i^k and their gradients must match at the internal endpoints x_i^k.
- Tight at extrema: $q_i(x) = 0$ at $\{\underline{x}_i, \overline{x}_i\}$.

$$q_i^k(x_i^k) = q_i^{k+1}(x_i^k) \quad \text{for all } k = 1, \cdots, N_i - 1$$

$$\frac{dq_i^k(x_i^k)}{dx_i} = \frac{dq_i^{k+1}(x_i^k)}{dx_i} \quad \text{for all } k = 1, \cdots, N_i - 1$$

$$q_i^1(x_i^0) = 0$$

$$q_i^{N_i}(x_i^{N_i}) = 0$$

Expands to a linear system in β and γ.

Formulae for β and γ

Linear System

$$\beta_i^k x_i^k + \gamma_i^k = \beta_i^{k+1} x_i^{k+1} + \gamma_i^{k+1} \qquad \text{for all } k = 1, \ldots, N_i$$

$$-\alpha_i^k \left(x_i^k - x_i^{k-1}\right) + \beta_i^k = \alpha_i^{k+1}\left(x_i^{k+1} - x_i^k\right) + \beta_i^{k+1} \qquad \text{for all } k = 1, \ldots, N_i$$

$$\beta_i^1 x_i^0 + \gamma_i^0 = 0$$

$$\beta_i^{N_i} x_i^{N_i} + \gamma_i^{N_i} = 0$$

Solution

$$\beta_i^1 = \left(\sum_{k=1}^{N_i - 1} s_i^k \left(x_i^k - x_i^{N_i}\right)\right) \Big/ \left(x_i^{N_i} - x_i^0\right)$$

$$\beta_i^k = \beta_i^1 + \sum_{j=1}^{k-1} s_i^j \qquad \text{for all } k = 2, \ldots, N_i$$

$$\gamma_i^k = -\gamma_i^1 x_i^0 - \sum_{j=1}^{k-1} s_i^j x_i^j \qquad \text{for all } k = 1, \ldots, N_i$$

where $s_i^k = -\alpha_i^k \left(x_i^k - x_i^{k-1}\right) - \alpha_i^{k+1}\left(x_i^{k+1} - x_i^k\right)$.

Illustration: Lennard-Jones Potential Energy Function

$f(x) = \frac{1}{x^{12}} - \frac{2}{x^{6}}$ in the interval $[\underline{x}, \overline{x}] = [0.85, 2.00]$.

- First term: convex, dominates when x is small
- Second term: concave, dominates when x is large

Minimum eigenvalues:

$$\min f'' = \begin{cases} \frac{156}{\overline{x}^{14}} - \frac{84}{\overline{x}^{8}} & \text{if} \quad \overline{x} \le 1.21707 \\ -7.47810 & \text{if} \quad [\underline{x}, \overline{x}] \ni 1.21707 \\ \frac{156}{\underline{x}^{14}} - \frac{84}{\underline{x}^{8}} & \text{if} \quad \underline{x} \ge 1.21707 \end{cases}$$

Illustration: Lennard-Jones

Standard αBB underestimator:

$$f(x) - \frac{7.47810}{2}\left(\overline{x} - x\right)\cdot\left(x - \underline{x}\right)$$

2 subinterval underestimator:

k	x^k	min f''	α^k	β^k	γ^k
0	0.850				
1	1.425	-7.47810	3.73905	1.62764	-1.38349
2	2.000	-3.84462	1.92231	-1.62764	3.25528

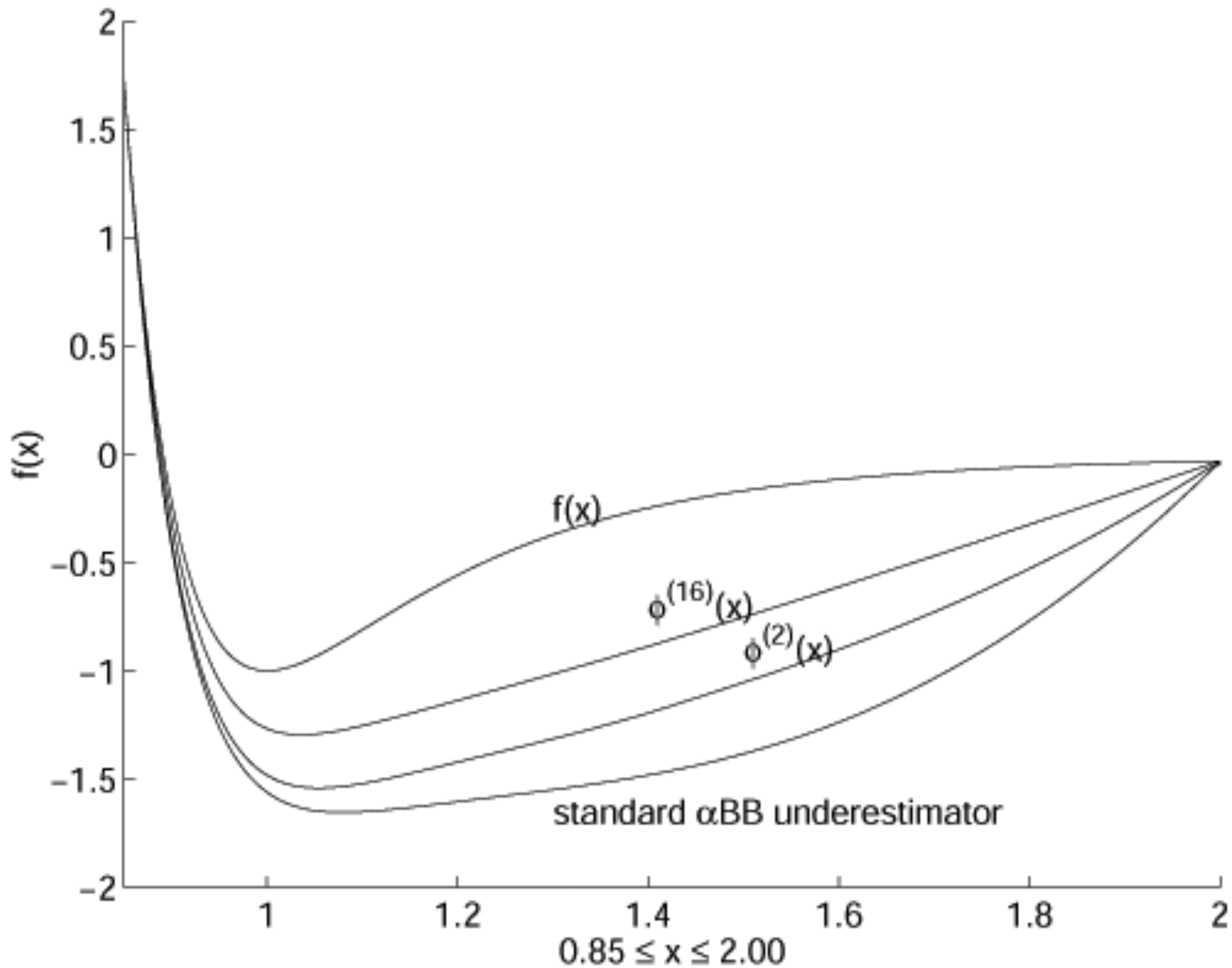
Underestimator when $0.850 \le x \le 1.425$:

$$f(x) - \left(3.73905\left(1.425 - x\right)\cdot\left(x - 0.850\right) + 1.62764x - 1.38349\right)$$

Underestimator when $1.425 \le x \le 2.00$:

$$f(x) - \left(1.92231\left(2.000 - x\right)\cdot\left(x - 1.425\right) - 1.62764x + 3.25528\right)$$

Illustration: Lennard-Jones



Outline

- Deterministic Global Optimization: Objectives & Motivation
- Convex Envelopes:
  - Trilinear Monomials
  - Edge Concave functions
- Piecewise linearization of Bilinear terms
- Checking Convexity: Products of Univariate Functions
- PαBB: Piecewise Quadratic Perturbations
- **Pooling Problems: Standard, Generalized & Extended - Towards addressing Large-Scale Global Optimization Problems**
- Conclusions

Motivation:

Globally Optimizing Standard, Generalized, and Extended Pooling Problems

- **Applications** of pooling problems include:
  - **Refining & Petrochemical**
    - **Crude Oil Scheduling**
    - **Combining Process Streams into Products**
  - **Wastewater Treatment**
    - **Removing heavy metals, organic matter, etc. from process streams**
  - **Supply Chain Operations**
  - **Communications**
- **Pooling** is necessitated by **limited storage conditions** requiring blending multiple streams into **intermediate nodes** or ***pools***

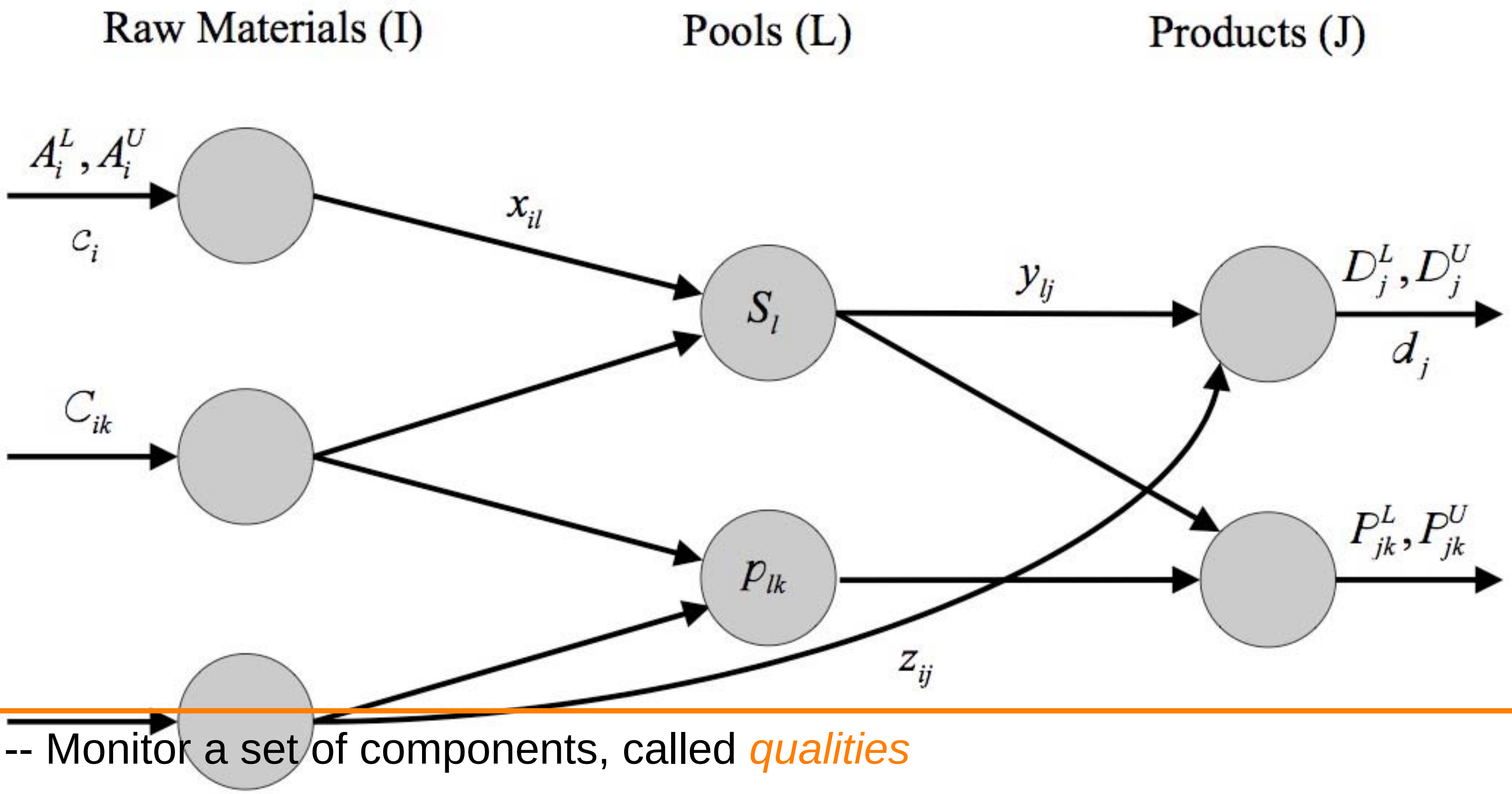
Relevant Publications

- **Misener & Floudas.** Global Optimization of Large-Scale Generalized Pooling Problems: Quadratically Constrained MINLP Models. *Ind. Eng. Chem. Res.* 49:5424-5438, 2010.
- **Misener, Gounaris & Floudas.** Mathematical Modeling and Global Optimization of Large-Scale Extended Pooling Problems with the (EPA) Complex Emissions Constraints. *Comp. Chem. Eng.,* 34, 1432-1456, 2010.
- **Misener & Floudas.** Advances for the Pooling Problem: Modeling, Global Optimization, and Computational Studies. *Appl. Comput. Math.,* 8:3-22, 2009.
- **Gounaris, Misener & Floudas.** Computational Comparison of Piecewise-Linear Relaxations for Pooling Problems. *Ind. Eng. Chem. Res.,* 48:5742-5766, 2009.

PRINCETON UNIVERSITY

Standard Pooling Problem:
NLP with Quadratic Nonconvex Terms

Haverly, *ACM SIGMAP Bulletin* [1978]; Floudas & Aggarwal, *Comput. Chem. Eng.* [1990]; Floudas & Visweswaran, *Comput. Chem. Eng.* [1990]; Lodwick, *ORSA J. Comput.* [1992]; Ben-Tal et al., *Math. Prog.* [1994]; Adhya et al., *Ind. Eng. Chem. Res.* [1999]; Foulds et al., *Optimization* [1992]; Quesada & Grossmann, *Comput. Chem. Eng.* [1995]; Tawarmalani & Sahinidis [2002]; Meyer & Floudas, *AIChE J.* [2006]; Pham et al., *Ind. Eng. Chem. Res.* [2009]

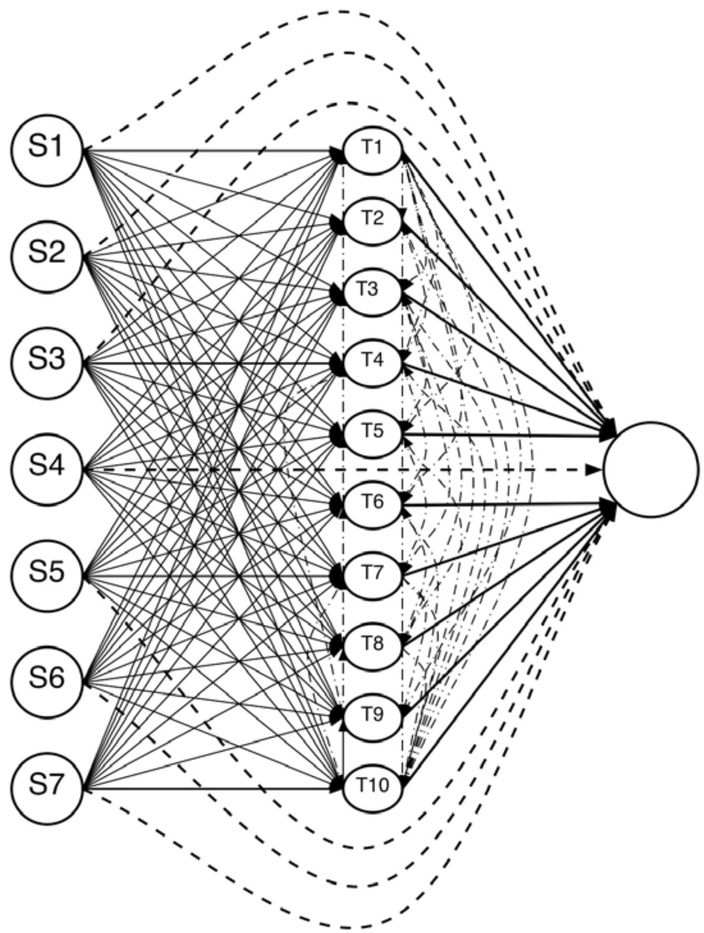


-- Monitor a set of components, called *qualities*

-- Assume linear blending at each intermediate and output node

Generalized Pooling Problems:
MINLP with Quadratic Nonconvex Terms

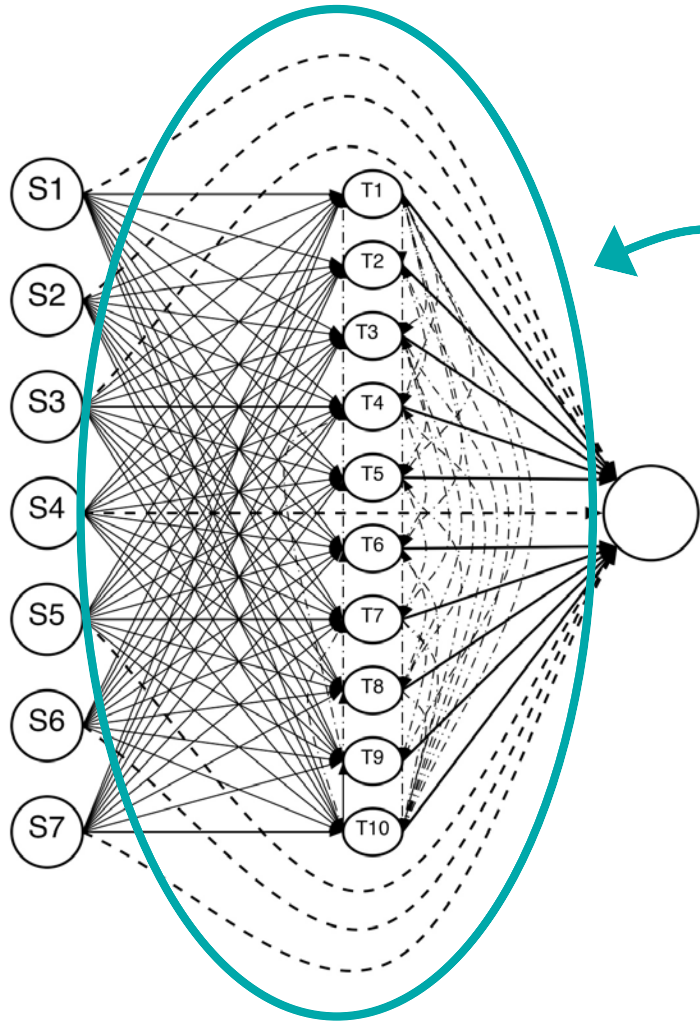
Galan & Grossmann, *Ind. Eng. Chem. Res.* [1998]; Bagajewicz, *Comp. Chem. Eng.* [2000]; Lee & Grossmann, *Comp. Chem. Eng.* [2003]; Audet et al., *Manag. Sci.* [2004]; Meyer & Floudas, *AIChE J.* [2006]; Karuppiah & Grossmann, *Comp. Chem. Eng.* [2006]



- In the combinatorially complex generalized pooling problem, **the network topology is a decision variable**
- Each stream & pool is assigned a binary decision variable & associated cost
- Meyer & Floudas, *AIChE J.* [2006] solved a 4-plant industrial problem to a 1.2% gap

Generalized Pooling Problems

Galan & Grossmann, *Ind. Eng. Chem. Res.* [1998]; Bagajewicz, *Comp. Chem. Eng.* [2000]; Lee & Grossmann, *Comp. Chem. Eng.* [2003]; Audet et al., *Manag. Sci.* [2004]; Meyer & Floudas, *AIChE J.* [2006]; Karuppiah & Grossmann, *Comp. Chem. Eng.* [2006]



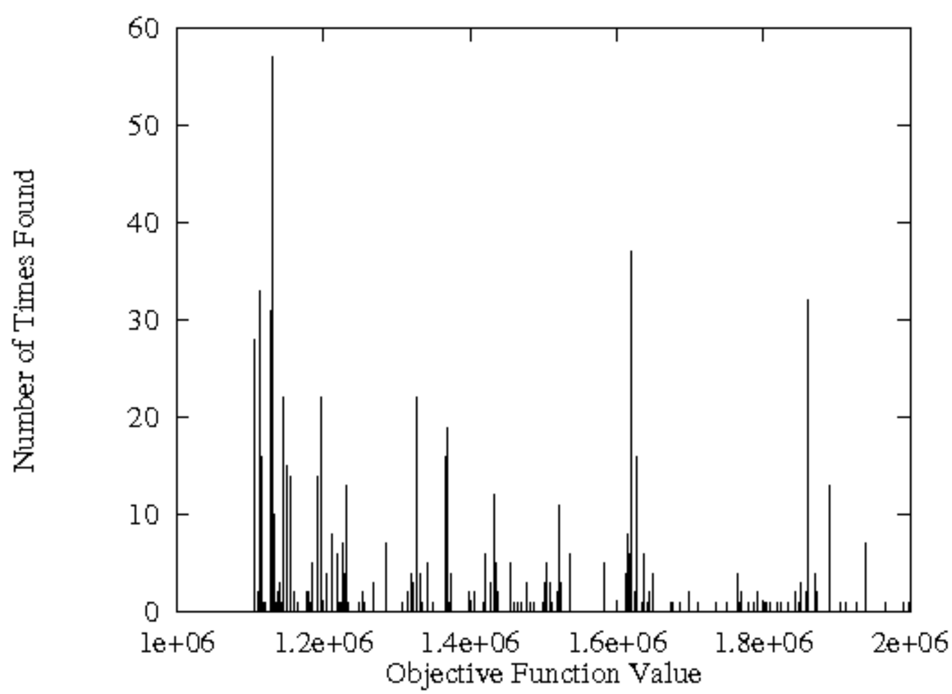
In the combinatorially complex generalized pooling problem, **the network topology is a decision variable**

Each stream & pool is assigned a binary decision variable & associated cost

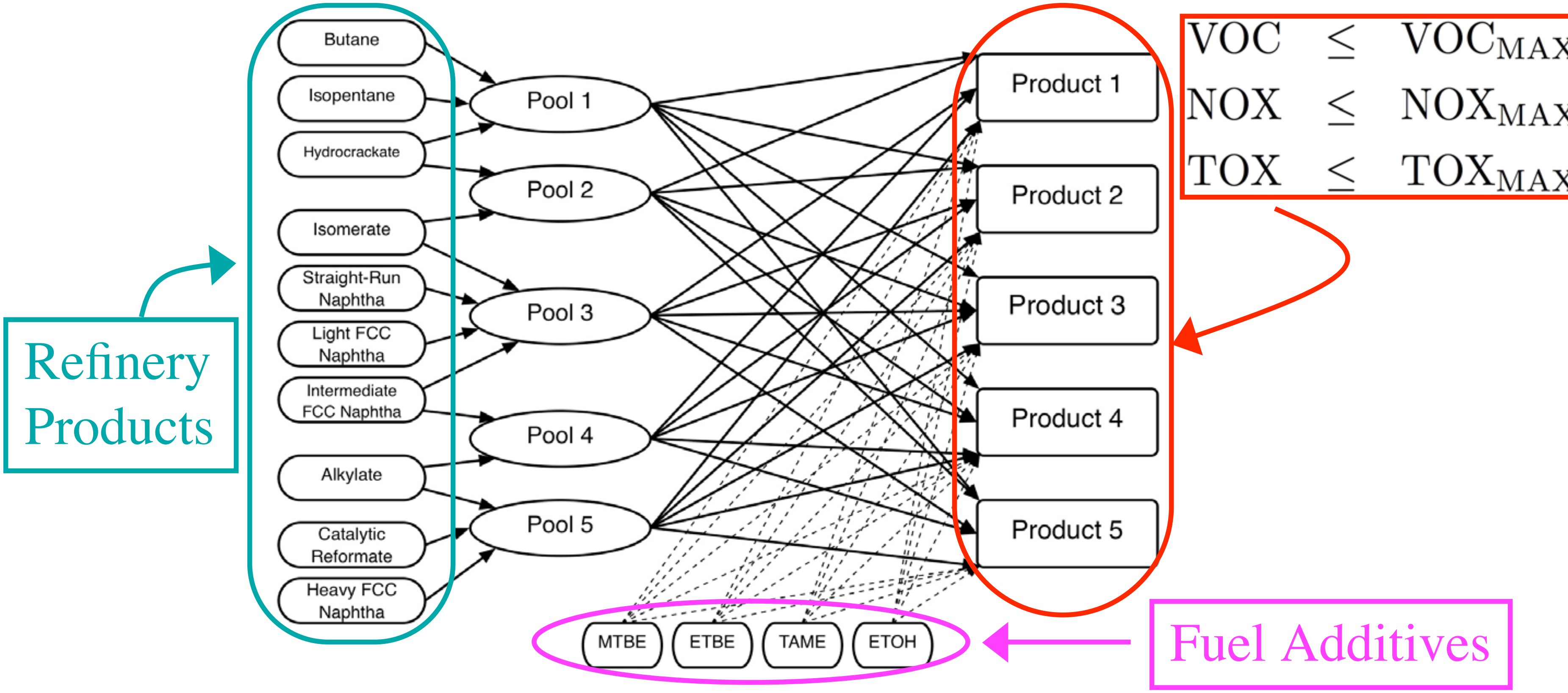
Meyer & Floudas, *AIChE J.* [2006] solved a 4-plant industrial problem to a 1.2% gap

Solutions Using GAMS/DICOPT and Random Starting Points

- Continuous variables initialized with uniformly distributed random numbers
- Binary variables initialized by rounding the uniformly distributed numbers in [0,1] to the nearest integer
- **DICOPT used to solve problem from 1000 starting points.**
- **Number of times best known solution was found: 0.**

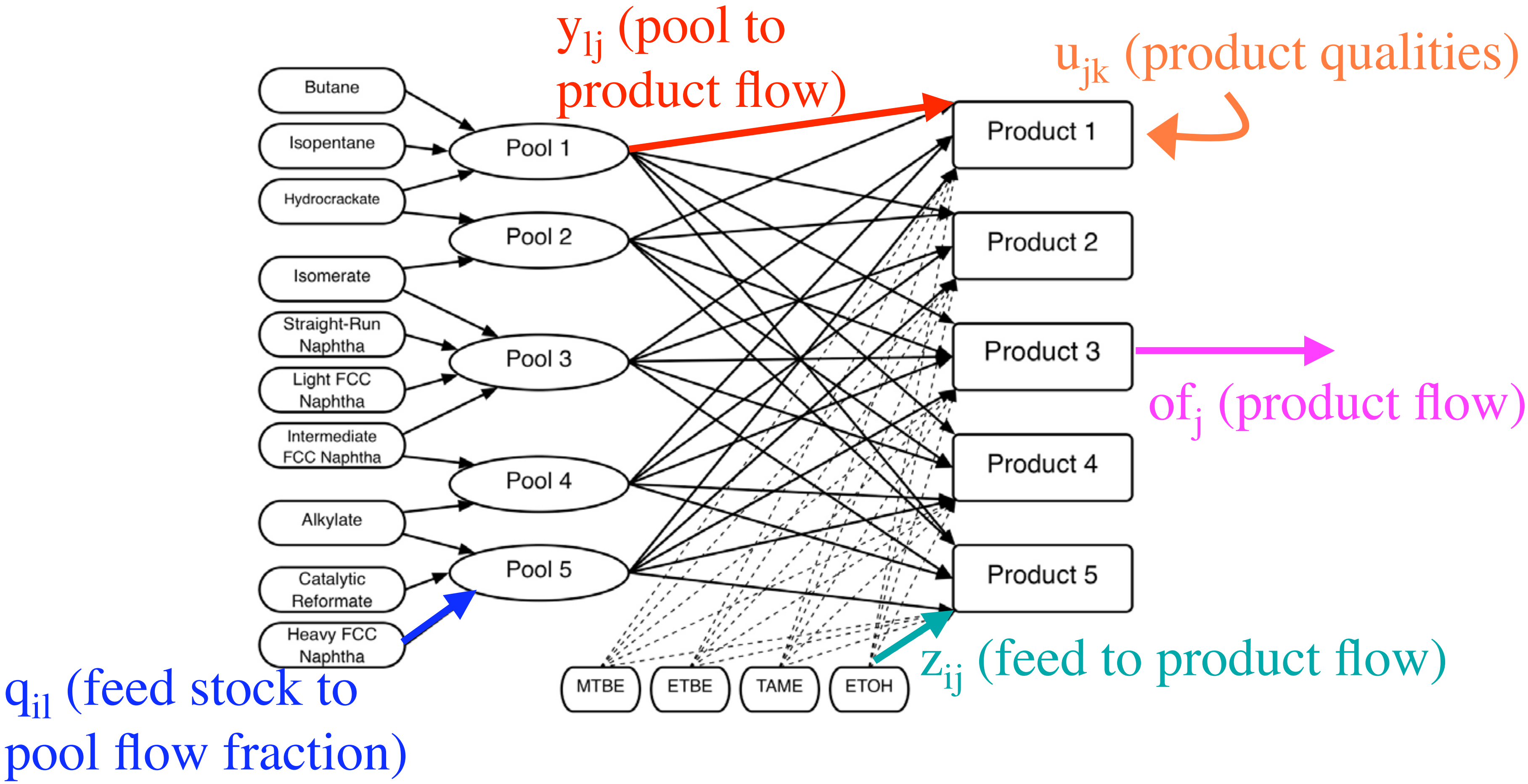


The Extended Pooling Problem: MINLP with General Nonconvex Constraints



- Given refinery exit streams, **meet volatile organics, NO_X, & toxic emissions standards** for each gasoline blend according to the EPA Complex Emissions Model & legislative bounds

Extended Pooling Problem: Standard Backbone



- **Monitor the set of 11 components** in the EPA Complex Emissions Model
- Assume **linear blending** at each intermediate & output node for all components except Reid Vapor Pressure (RVP), which blends nonlinearly

The Extended Pooling Problem

- The extended pooling problem incorporates the EPA Complex Emissions Model Constraints and associated legislative bounds on volatile organics (VOC), NO_X, and toxic (TOX) emissions into the constraint set:

$$
\begin{aligned}
\mathrm{VOC} &\leq \mathrm{VOC}_{\mathrm{MAX}} \\
\mathrm{NOX} &\leq \mathrm{NOX}_{\mathrm{MAX}} \\
\mathrm{TOX} &\leq \mathrm{TOX}_{\mathrm{MAX}}
\end{aligned}
$$

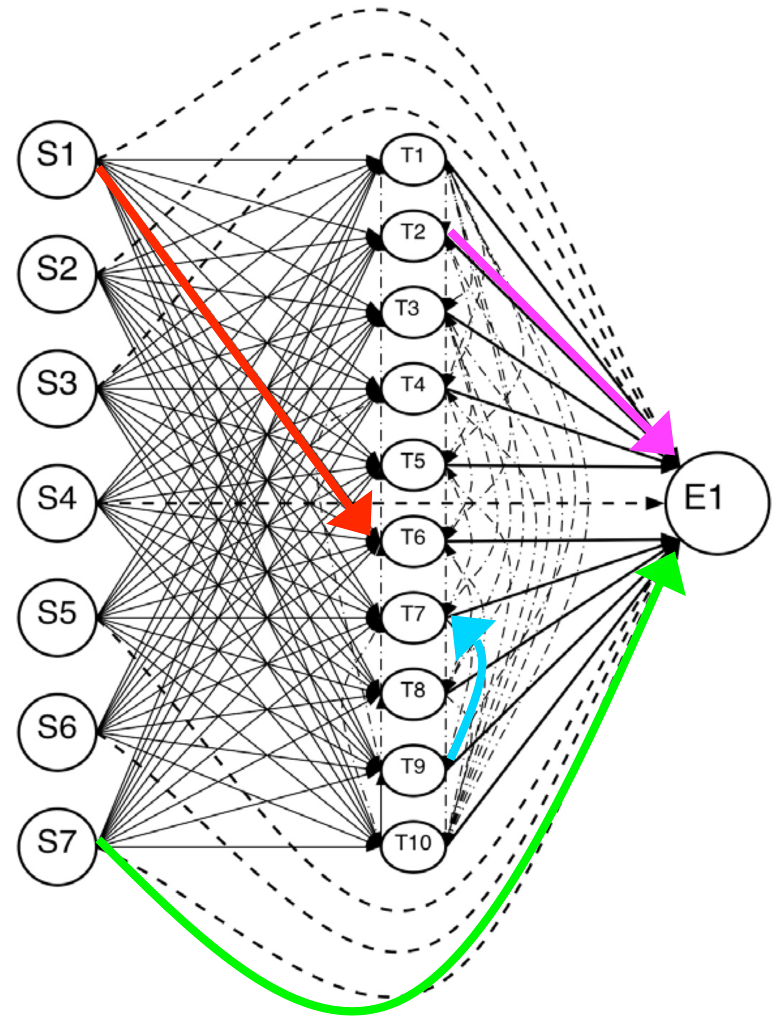
Outline

- Deterministic Global Optimization: Objectives & Motivation
- Convex Envelopes:
  - Trilinear Monomials
  - Edge Concave functions
- Piecewise linearization of Bilinear terms
- Checking Convexity: Products of Univariate Functions
- PαBB: Piecewise Quadratic Perturbations
- **Pooling Problems: Standard, Generalized & Extended - Towards addressing Large-Scale Global Optimization Problems**
- Conclusions

Generalized Pooling Problem

Meyer & Floudas, *AIChE J.* [2006]; Misener & Floudas, *Ind. Eng. Chem. Res.* [2010]

- Topology: 7 sources; 1 or 2 sinks; multiple possible treatment plants
- Possible connections: source to plant; source to sink; plant to plant; plant to sink



Problem Definition [Meyer & Floudas, *AIChE J.* 2006]

$$\min_{a,b,c,d,q,y^a,y^b,y^c,y^d,y^e} \quad z^P = \sum_{s\in S}\sum_{e\in E} c^a_{se} a_{se} + \sum_{t\in T}\sum_{e\in E} c^b_{te} b_{te}$$
$$+ \sum_{t\in T}\sum_{t'\in T\setminus\{t\}} (c^c_{tt'} + c^e_{t'}) c_{tt'} + \sum_{s\in S}\sum_{t\in T} (c^d_{st} + c^e_t) d_{st}$$
$$+ \sum_{s\in S}\sum_{e\in E} c^{ya}_{se} y^a_{se} + \sum_{t\in T}\sum_{e\in E} c^{yb}_{te} y^b_{te}$$
$$+ \sum_{t\in T}\sum_{t'\in T\setminus\{t\}} c^{yc}_{tt'} y^c_{tt'} + \sum_{s\in S}\sum_{t\in T} c^{yd}_{st} y^d_{st}$$
$$+ \sum_{t\in T} c^{ye}_t y^e_t$$

$$a_{se} - y^a_{se}\bar{a}_{se} \le 0$$
$$\underline{a}_{se} y^a_{se} - a_{se} \le 0$$
$$b_{te} - y^b_{te}\bar{b}_{te} \le 0$$
$$\underline{b}_{te} y^b_{te} - b_{te} \le 0$$
$$c_{tt'} - y^c_{tt'}\bar{c}_{tt'} \le 0$$
$$\underline{c}_{tt'} y^c_{tt'} - c_{tt'} \le 0$$
$$d_{st} - y^d_{st}\bar{d}_{st} \le 0$$
$$\underline{d}_{st} y^d_{st} - d_{st} \le 0$$

$$\sum_{s\in S} a_{se} q^{\text{source}}_{cs} + \sum_{t\in T} b_{te} q_{ct} \le q^{\max}_c$$

$$\sum_{s\in S} d_{st} + \sum_{t'\in T\setminus\{t\}} c_{t't} - \bar{e}_t y^e_t \le 0$$

$$\times \left(\sum_{s\in S} a_{se} \right.$$

$$c_{t't} + \underline{e}_t y^e_t \le 0$$

The objective, which represents water treatment cost, reflects both the variable costs of flow rates & the fixed costs of activating each plant or connection

$$q_{ct}\left(\sum_{s\in S} d_{st} + \sum_{t'\in T\setminus\{t\}} c_{t't} \right) = (1 - r_{ct})$$
$$\times \left(\sum_{t'\in T\setminus\{t\}} c_{t't} q_{ct'} + \sum_{s\in S} d_{st} q^{\text{source}}_{cs} \right)$$

$$\sum_{t'\in T\setminus\{t\}} c_{t't} - \sum_{t'\in T\setminus\{t\}} c_{tt'} + \sum_{s\in S} d_{st} = \sum_{e\in E} b_{te}$$

$$y^c_{tt'} + y^c_{t't} \le 1$$

Problem Definition [Meyer & Floudas, *AIChE J.* 2006]

$$\min_{a,b,c,d,q,y^a,y^b,y^c,y^d,y^e} \quad z^P = \sum_{s\in S}\sum_{e\in E} c^a_{se} a_{se} + \sum_{t\in T}\sum_{e\in E} c^b_{te} b_{te} + \sum_{t\in T}\sum_{t'\in T\setminus\{t\}} (c^c_{tt'} + c^e_{t'}) c_{tt'} + \sum_{s\in S}\sum_{t\in T} (c^d_{st} + c^e_t) d_{st}$$
$$+ \sum_{s\in S}\sum_{e\in E} c^{ya}_{se} y^a_{se} + \sum_{t\in T}\sum_{e\in E} c^{yb}_{te} y^b_{te} + \sum_{t\in T}\sum_{t'\in T\setminus\{t\}} c^{yc}_{tt'} y^c_{tt'} + \sum_{s\in S}\sum_{t\in T} c^{yd}_{st} y^d_{st} + \sum_{t\in T} c^{ye}_t y^e_t$$

$$a_{se} - y^a_{se}\bar{a}_{se} \le 0$$
$$\underline{a}_{se} y^a_{se} - a_{se} \le 0$$
$$b_{te} - y^b_{te}\bar{b}_{te} \le 0$$
$$\underline{b}_{te} y^b_{te} - b_{te} \le 0$$
$$c_{tt'} - y^c_{tt'}\bar{c}_{tt'} \le 0$$
$$\underline{c}_{tt'} y^c_{tt'} - c_{tt'} \le 0$$
$$d_{st} - y^d_{st}\bar{d}_{st} \le 0$$
$$\underline{d}_{st} y^d_{st} - d_{st} \le 0$$
$$\sum_{s\in S} d_{st} + \sum_{t'\in T\setminus\{t\}} c_{t't} - \bar{e}_t y^e_t \le 0$$
$$-\sum_{s\in S} d_{st} - \sum_{t'\in T\setminus\{t\}} c_{t't} + \underline{e}_t y^e_t \le 0$$

$$\sum_{s\in S} a_{se} q^{\text{source}}_{cs} + \sum_{t\in T} b_{te} q_{ct} \le q^{\max}_c \times \left(\sum_{s\in S} a_{se} + \sum_{t\in T} b_{te} \right)$$

$$q_{ct}\left(\sum_{s\in S} d_{st} + \sum_{t'\in T\setminus\{t\}} c_{t't} \right) = (1 - r_{ct}) \times \left(\sum_{t'\in T\setminus\{t\}} c_{t't} q_{ct'} + \sum_{s\in S} d_{st} q^{\text{source}}_{cs} \right)$$

Bilinear terms in the model

$$\sum_{e\in E} a_{se} + \sum_{t\in T} d_{st} = f^{\text{source}}_s$$
$$\sum_{t'\in T\setminus\{t\}} c_{t't} - \sum_{t'\in T\setminus\{t\}} c_{tt'} + \sum_{s\in S} d_{st} = \sum_{e\in E} b_{te}$$
$$y^c_{tt'} + y^c_{t't} \le 1$$

Problem Definition [Misener & Floudas, *Ind. Eng. Chem. Res.* 2010]

$$\min z_p = \sum_{\substack{s\in S\\ e\in E}} \left(c^a_{s,e}\cdot a_{s,e} + c^{ya}_{s,e}\cdot y^a_{s,e} \right) + \sum_{\substack{t\in T\\ e\in E}} \left(c^b_{t,e}\cdot b_{t,e} + c^{yb}_{t,e}\cdot y^b_{t,e} \right) +$$

$$\sum_{\substack{t\in T\\ t'\in T\setminus\{t\}}} \left(c^c_{t,t'}\cdot c_{t,t'} + c^{yc}_{t,t'}\cdot y^c_{t,t'} \right) + \sum_{\substack{t\in T\\ s\in S}} \left(c^d_{s,t}\cdot d_{s,t} + c^{yd}_{s,t}\cdot y^d_{s,t} \right) +$$

$$\sum_{t\in T} \left(c^e_t \cdot \left[\sum_{t'\in T\setminus\{t\}} c_{t,t'} + \sum_{e\in E} b_{t,e} \right] + c^{ye}_t \cdot y^e_t \right)$$

$$y^a_{s,e}\cdot \underline{a}_{s,e} \le a_{s,e} \le y^a_{s,e}\cdot \overline{a}_{s,e}$$

$$y^b_{t,e}\cdot \underline{b}_{t,e} \le b_{t,e} \le y^b_{t,e}\cdot \overline{b}_{t,e}$$

$$y^c_{t,t'}\cdot \underline{c}_{t,t'} \le c_{t,t'} \le y^c_{t,t'}\cdot \overline{c}_{t,t'}$$

$$y^d_{s,t}\cdot \underline{d}_{s,t} \le d_{s,t} \le y^d_{s,t}\cdot \overline{d}_{s,t}$$

$$y^e_t\cdot \underline{e}_t \le \sum_{t'\in T\setminus\{t\}} c_{t,t'} + \sum_{e\in E} b_{t,e} \le y^e_t\cdot \overline{e}_t$$

$$(1 - r_{c,t})\cdot \left[\sum_{s\in S} q^{\text{source}}_{c,s}\cdot d_{s,t} + \sum q_{c,t'}\cdot c_{t',t} \right] =$$

$$q_{c,t}\cdot \left[\sum_{t'\in T} \ldots \right]$$

$$f^{\text{source}}_s = \sum a_{s,e} + \sum d_{s,t}$$

$$\ldots \sum_{e\in E} b_{t,e}$$

$$q^{\max}_{c,e}\cdot \left(\sum_{s\in S} a_{s,e} + \sum_{t\in T} b_{t,e} \right) \ge$$

$$\sum_{s\in S} q^{\text{source}}_{c,s}\cdot a_{s,e} + \sum_{t\in T} q_{c,t}\cdot b_{t,e}$$

$$\sum_{s\in S} f_s \ldots \sum_{\substack{s\in S\\ e\in E}} a_{s,e} + \sum_{\substack{t\in T\\ e\in E}} b_{t,e}$$

$$f^{\text{sink,max}}_e \ge \sum_{s\in S} a_{s,e} + \sum_{t\in T} b_{t,e}$$

$$y^c_{t,t'} + y^c_{t',t} \le 1$$

The **objective** and the **discrete decisions** of activating or not each pipe or treatment plant remain the same as the Meyer & Floudas, *AIChE J.* [2006] formulation

Problem Definition [Misener & Floudas, *Ind. Eng. Chem. Res.* 2010]

$$\min z_p = \sum_{\substack{s\in S\\ e\in E}} \left(c^a_{s,e} \cdot a_{s,e} + c^{ya}_{s,e} \cdot y^a_{s,e} \right) + \sum_{\substack{t\in T\\ e\in E}} \left(c^b_{t,e} \cdot b_{t,e} + c^{yb}_{t,e} \cdot y^b_{t,e} \right) +$$

$$\sum_{\substack{t\in T\\ t'\in T\setminus\{t\}}} \left(c^c_{t,t'} \cdot c_{t,t'} + c^{yc}_{t,t'} \cdot y^c_{t,t'} \right) + \sum_{\substack{t\in T\\ s\in S}} \left(c^d_{s,t} \cdot d_{s,t} + c^{yd}_{s,t} \cdot y^d_{s,t} \right) +$$

$$q_{c,t} \cdot \left[\sum_{t'\in T\setminus\{t\}} c_{t,t'} + \sum_{e\in E} b_{t,e} \right] \quad \forall\, c \in C;\ t \in T$$

$$q^{\max}_{c,e} \cdot \left(\sum_{s\in S} a_{s,e} + \sum_{t\in T} b_{t,e} \right) \geq \sum_{s\in S} q^{\text{source}}_{c,s} \cdot a_{s,e} + \sum_{t\in T} q_{c,t} \cdot b_{t,e}$$

The **material balances** are equivalent to the Meyer & Floudas, *AIChE J*. [2006] formulation except that they now permit multiple sinks (output nodes)

$$y^a_{s,e} \cdot \underline{a}_{s,e} \leq a_{s,e} \leq y^a_{s,e} \cdot \overline{a}_{s,e}$$

$$y^b_{t,e} \cdot \underline{b}_{t,e} \leq b_{t,e} \leq y^b_{t,e} \cdot \overline{b}_{t,e}$$

$$y^c_{t,t'} \cdot \underline{c}_{t,t'} \leq c_{t,t'} \leq y^c_{t,t'} \cdot \overline{c}_{t,t'}$$

$$y^d_{s,t} \cdot \underline{d}_{s,t} \leq d_{s,t} \leq y^d_{s,t} \cdot \overline{d}_{s,t}$$

$$y^e_t \cdot \underline{e}_t \leq \sum_{t'\in T\setminus\{t\}} c_{t,t'} + \sum_{e\in E} b_{t,e} \leq y^e_t \cdot \overline{e}_t$$

$$f^{\text{source}}_s = \sum_{e\in E} a_{s,e} + \sum_{t\in T} d_{s,t}$$

$$\sum_{s\in S} d_{s,t} + \sum_{t'\in T\setminus\{t\}} c_{t',t} = \sum_{t'\in T\setminus\{t\}} c_{t,t'} + \sum_{e\in E} b_{t,e}$$

$$\sum_{s\in S} f^{\text{source}}_s = \sum_{\substack{s\in S\\ e\in E}} a_{s,e} + \sum_{\substack{t\in T\\ e\in E}} b_{t,e}$$

$$f^{\text{sink,max}}_e \geq \sum_{s\in S} a_{s,e} + \sum_{t\in T} b_{t,e}$$

$$y^c_{t,t'} + y^c_{t',t} \leq 1$$

Problem Definition [Misener & Floudas, *Ind. Eng. Chem. Res.* 2010]

$$\min z_p = \sum_{\substack{s \in S \\ e \in E}} \left(c^a_{s,e} \cdot a_{s,e} + c^{ya}_{s,e} \cdot y^a_{s,e} \right) + \sum_{\substack{t \in T \\ e \in E}} \left(c^b_{t,e} \cdot b_{t,e} + c^{yb}_{t,e} \cdot y^b_{t,e} \right) +$$

$$\sum_{\substack{t \in T \\ t' \in T \setminus \{t\}}} \left(c^c_{t,t'} \cdot c_{t,t'} + c^{yc}_{t,t'} \cdot y^c_{t,t'} \right) + \sum_{\substack{t \in T \\ s \in S}} \left(c^d_{s,t} \cdot d_s \ldots \right.$$

$$\sum_{t \in T} \left(c^e_t \cdot \left[\sum_{t' \in T \setminus \{t\}} c_{t,t'} + \sum_{e \in E} b_{t,e} \right] + c^{ye}_t \cdot y^e_t \right)$$

$$y^a_{s,e} \cdot \underline{a}_{s,e} \leq a_{s,e} \leq y^a_{s,e} \cdot \overline{a}_{s,e}$$

$$\ldots \sum_{t' \in T \setminus \{t\}} \ldots \sum_{e \in E} \ldots$$

The equations **monitoring material balances** across treatment units & **limiting emitted contaminants** are also equivalent to the Meyer & Floudas, *AIChE J*. [2006] formulation, but use fewer bilinear terms

$$(1 - r_{c,t}) \cdot \left[\sum_{s \in S} q^{\text{source}}_{c,s} \cdot d_{s,t} + \sum_{t' \in T \setminus \{t\}} q_{c,t'} \cdot c_{t',t} \right] =$$

$$q_{c,t} \cdot \left[\sum_{t' \in T \setminus \{t\}} c_{t,t'} + \sum_{e \in E} b_{t,e} \right] \quad \forall c \in C;\ t \in T$$

$$q^{\max}_{c,e} \cdot \left(\sum_{s \in S} a_{s,e} + \sum_{t \in T} b_{t,e} \right) \geq$$

$$\sum_{s \in S} q^{\text{source}}_{c,s} \cdot a_{s,e} + \sum_{t \in T} q_{c,t} \cdot b_{t,e}$$

$$f^{\text{source}}_s = \sum_{e \in E} a_{s,e} + \sum_{t \in T} d_{s,t}$$

$$\sum_{s \in S} d_{s,t} + \sum_{t' \in T \setminus \{t\}} c_{t',t} = \sum_{t' \in T \setminus \{t\}} c_{t,t'} + \sum_{e \in E} b_{t,e}$$

$$\sum_{s \in S} f^{\text{source}}_s = \sum_{\substack{s \in S \\ e \in E}} a_{s,e} + \sum_{\substack{t \in T \\ e \in E}} b_{t,e}$$

$$f^{\text{sink,max}}_e \geq \sum_{s \in S} a_{s,e} + \sum_{t \in T} b_{t,e}$$

$$y^c_{t,t'} + y^c_{t',t} \leq 1$$

Sizes of the Case Studies

	Topology				# Eqs	# Variables		# Bilinear
	$\|C\|$	$\|S\|$	$\|T\|$	$\|E\|$		Contin.	Binary	Terms
4-Plant	3	7	4	1	150	63	55	48
10-Plant	3	7	10	1	516	207	187	300
15-Plant	3	7	15	1	986	382	352	675
20-Plant FDC	3	7	20	2	1663	634	594	1260
20-Plant DDC	3	7	20	2	1663	634	594	1260

Regulated qualities

Treatment plant options

Environmental sinks

Th… …n 2009 can
… gap to **1.22% in 1561.6 CPU s**
[Meyer & Floudas, *AIChE J.* 2006]

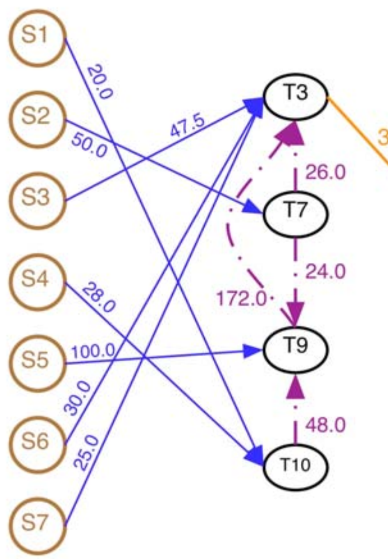
Sizes of the Case Studies

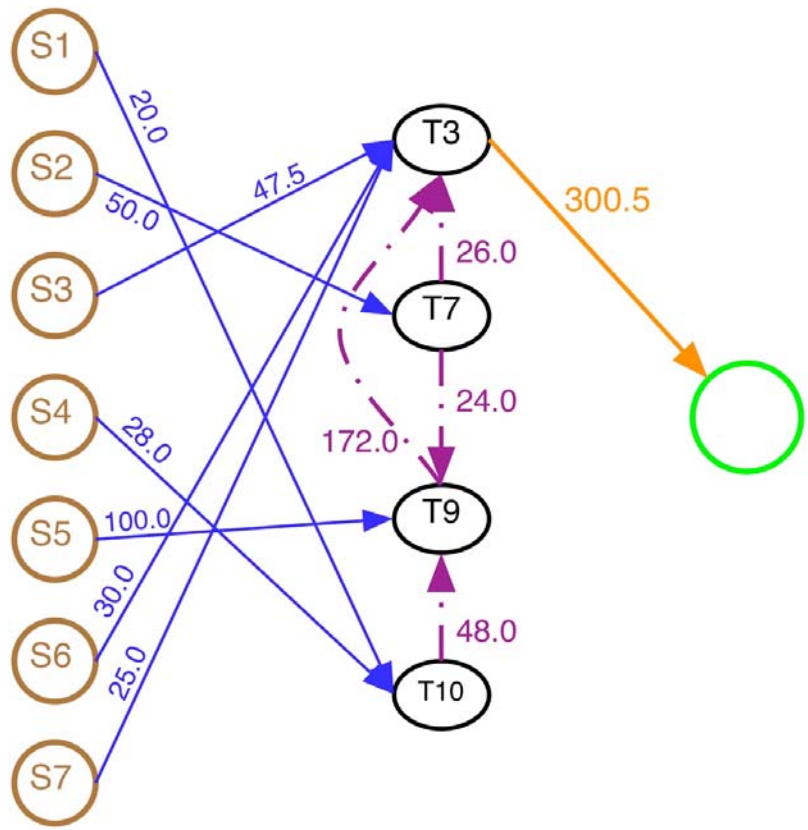
	Topology				# Eqs	# Variables		# Bilinear
	$\|C\|$	$\|S\|$	$\|T\|$	$\|E\|$		Contin.	Binary	Terms
4-Plant	3	7	4	1	150	63	55	48
10-Plant	3	7	10	1	516	207	187	300
15-Plant	3	7	15	1	986	382	352	675
20-Plant FDC	3	7	20	2	1663	634	594	1260
20-Plant DDC	3	7	20	2	1663	634	594	1260

The best-known algorithm ... can reduce the 4-Plant test ...
[Meyer & Floudas, *AIChE J.* 2006]

But the problems we actually want to address are much larger

Optimal Case Study Topologies





4 & 10 Plant C

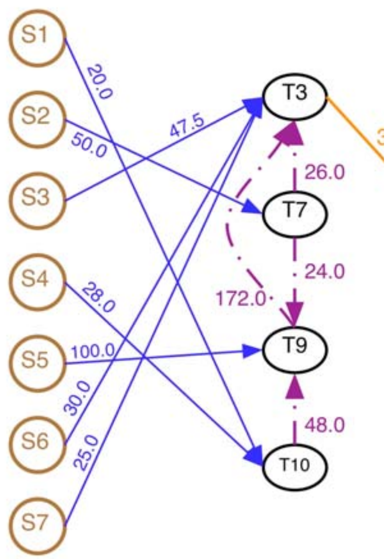
z = 1.086 * 10^6

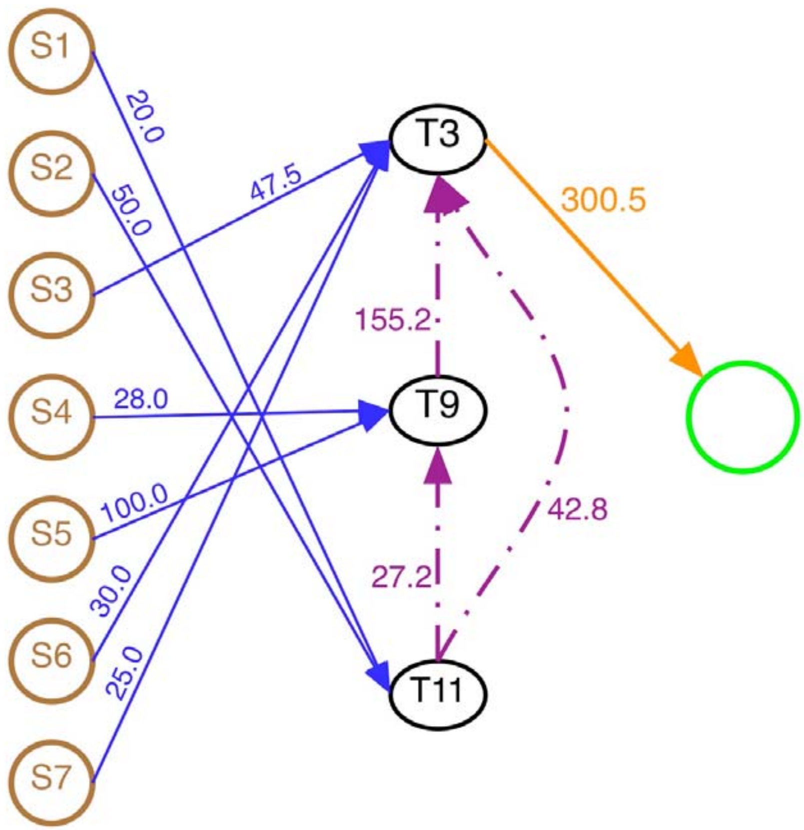
4-Plant

10-Pla

4 Plant Case; z = 1.086 * 10^6; 150 Eqs, 63 Con Vars, 55 Bin Vars, 48 Bilin Terms, **38.2 CPU s**

Optimal Case Study Topologies





4 & 10 Plant C

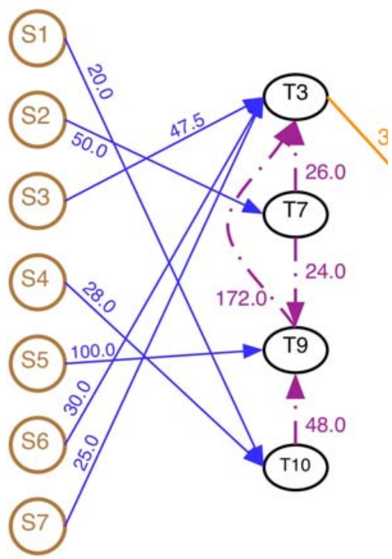
$z = 1.086 * 10^6$

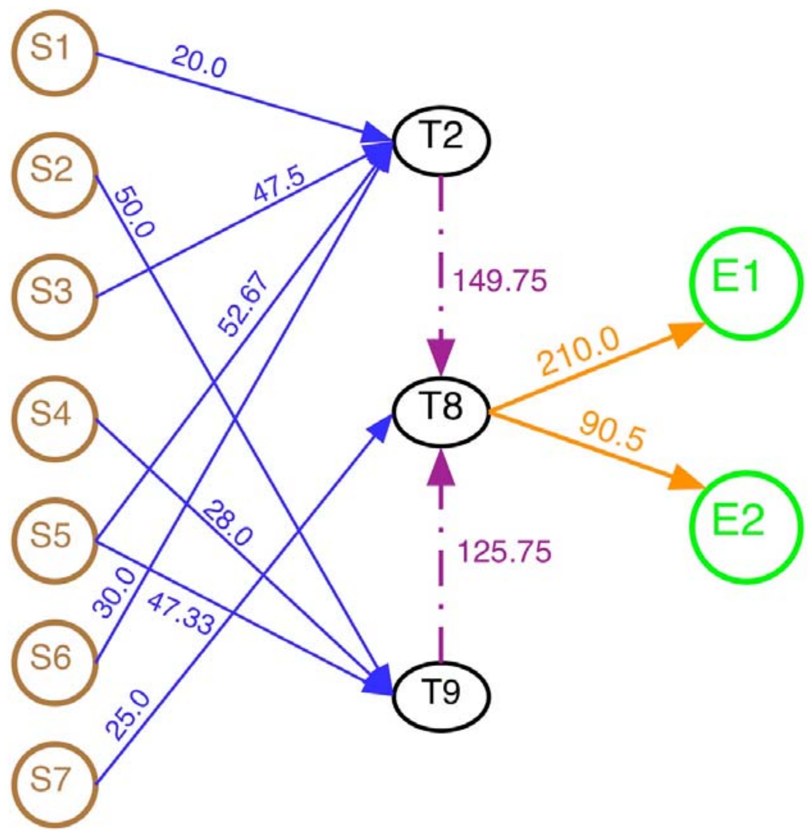
4-Plant

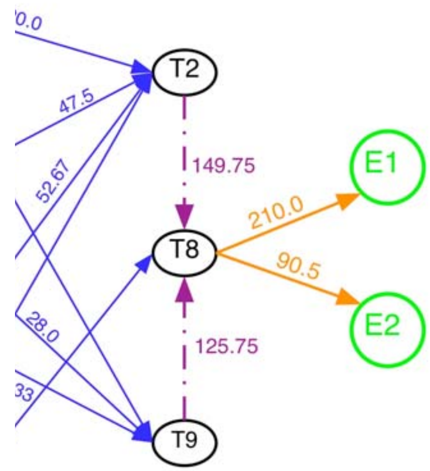
10-Pla

15 Plant Case; $z = 9.437 * 10^6$; 986 Eqs, 382 Cont Vars, 352 Bin Vars, 675 Bil Terms, **2490 CPU s**

Optimal Case Study Topologies







4 & 10 Plant C

$z = 1.086 * 10^6$

4-P

10-]

Plant Cases

$= 1.375 * 10^6$

10^6

20 Plant Cases; 1663 Eqs, 634 Cont Vars, 594 Bin Vars, 1260 Bil Terms; $z_1 = 1.375 * 10^6$; $z_2 = 1.416 * 10^6$

Optimizing the 4-Plant Test Case

Table 4: Branch & Bound Results for the 4 Plant Test Case (Serial Optimization)

N	Root Node		#	Termination			CPU Time (s)	
	Relaxation	CPU (s)	Nodes	Lower Bnd	Upper Bnd	% Gap	Solving	Total
McC	$7.219 \cdot 10^5$	0.1	5000*	$1.082 \cdot 10^6$	$1.106 \cdot 10^6$	2.2	773.9	7112.5
RLT	$7.223 \cdot 10^5$	0.6	1712	$1.085 \cdot 10^6$	$1.086 \cdot 10^6$	0.1	783.6	2720.0
3	$9.607 \cdot 10^5$	2.9	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	190.1
5	$9.896 \cdot 10^5$	5.9	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	38.2
7	$1.005 \cdot 10^6$	5.6	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	39.1
10	$1.041 \cdot 10^6$	16.6	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	54.1
15	$1.086 \cdot 10^6$	109.3	1	$1.086 \cdot 10^6$	$1.086 \cdot 10^6$	< 0.1	109.31	111.6

*Branch-and-bound tree limited to 5000 nodes

Trade-off: Rlxns with many partitions tend to be tighter but rlxns with few partitions tend to solve quickly

- The best known algorithms in 2003 run on a machine in 2009 can reduce the 4-Plant test case gap to **1.22% in 1561.6 CPU s** [Meyer & Floudas, *AIChE J.* 2006]

Optimizing the 4-Plant Test Case

Table 4: Branch & Bound Results for the 4 Plant Test Case (Serial Optimization)

N							CPU Time (s) Solving	Total
M							773.9	7112.5
R							783.6	2720.0
3	$9.607 \cdot 10^5$	2.9	95	$1.085 \cdot 10^6$	$1.086 \cdot 10^6$	0.1	86.36	190.1
5	$9.896 \cdot 10^5$	5.9	7	$1.085 \cdot 10^6$	$1.086 \cdot 10^6$	0.1	25.07	38.2
7	$1.005 \cdot 10^6$	5.6	5	$1.085 \cdot 10^6$	$1.086 \cdot 10^6$	0.1	33.25	39.1
10							50.37	54.1
15							109.31	111.6

*Bran

The new algorithms we have explored can address the same problem to a **0.1% gap in 38 CPU s** (a ten-fold improvement in one-fortieth the time)

Balancing this trade-off with intermediate partitioning generates the smallest total CPU time

- The best known algorithms in 2003 run on a machine in 2009 can reduce the 4-Plant test case gap to **1.22% in 1561.6 CPU s** [Meyer & Floudas, *AIChE J.* 2006]

Optimizing the 15-Plant Test Case

Using our experience with the 4- & 10-Plant test cases, solve the 15-Plant test case with N = 5 on the same Linux workstation & converge to **0.1% in 2489.76 s**. Moving the same problem to parallel CPLEX on a Beowulf cluster confirms that N = 5 is still appropriate.

Table 6: Branch & Bound Results for the 15 Plant Test Case (8-Threaded Parallel Optimization)

N	Root Node Relaxation	# Nodes	Termination Lower Bnd	Termination Upper Bnd	Termination % Gap	Total CPU s
McC	$7.129 \cdot 10^5$	891	$9.424 \cdot 10^5$	$9.437 \cdot 10^5$	0.1	15940.90
RLT	$7.132 \cdot 10^5$	967	$9.343 \cdot 10^5$	$9.437 \cdot 10^5$	0.1	66592.25
3	$9.002 \cdot 10^5$	7	$9.343 \cdot 10^5$	$9.437 \cdot 10^5$	0.1	1141.98
5	$9.345 \cdot 10^5$	1	$9.345 \cdot 10^5$	$9.437 \cdot 10^5$	< 0.1	784.81
7	$9.408 \cdot 10^5$	9	$9.343 \cdot 10^5$	$9.437 \cdot 10^5$	0.1	1874.73
10	$9.345 \cdot 10^5$	1	$9.345 \cdot 10^5$	$9.437 \cdot 10^5$	< 0.1	912.03
15	$9.345 \cdot 10^5$	1	$9.345 \cdot 10^5$	$9.437 \cdot 10^5$	< 0.1	2397.24

Optimizing the 20-Plant Test Cases: Fixed Disposal Costs

The test cases with 1260 bilinear terms are challenging, so employ additional strategies:

(1) **Limit partitioned terms** to those related to commonly-used plants

(2) Solve MILP relaxation to a tight gap only in later nodes of the BB tree

N	# Nodes	Termination: Lo[illegible]	Termination: [illegible]	[illegible]	Total [illegible]
McC	2462	1.[illegible]	[illegible]	[illegible]	[illegible]
RLT	8166	$1.268 \cdot 10^6$	[illegible]	[illegible]	[illegible]
Partitioning on all 20 Treatmen[illegible]					
$N = 3$	49	$1.338 \cdot 10^6$	[illegible]	[illegible]	[illegible]
Limited Partitioning					
$N = 3$	331	$1.362 \cdot 10^6$	$1.375 \cdot 10^6$	0.9	360000*
$N = 5$	17	$1.210 \cdot 10^6$	$1.555 \cdot 10^6$	22.2	360000*

*Branch-and-bound tree limited to 3.6×10^5 CPU s (100 hours)

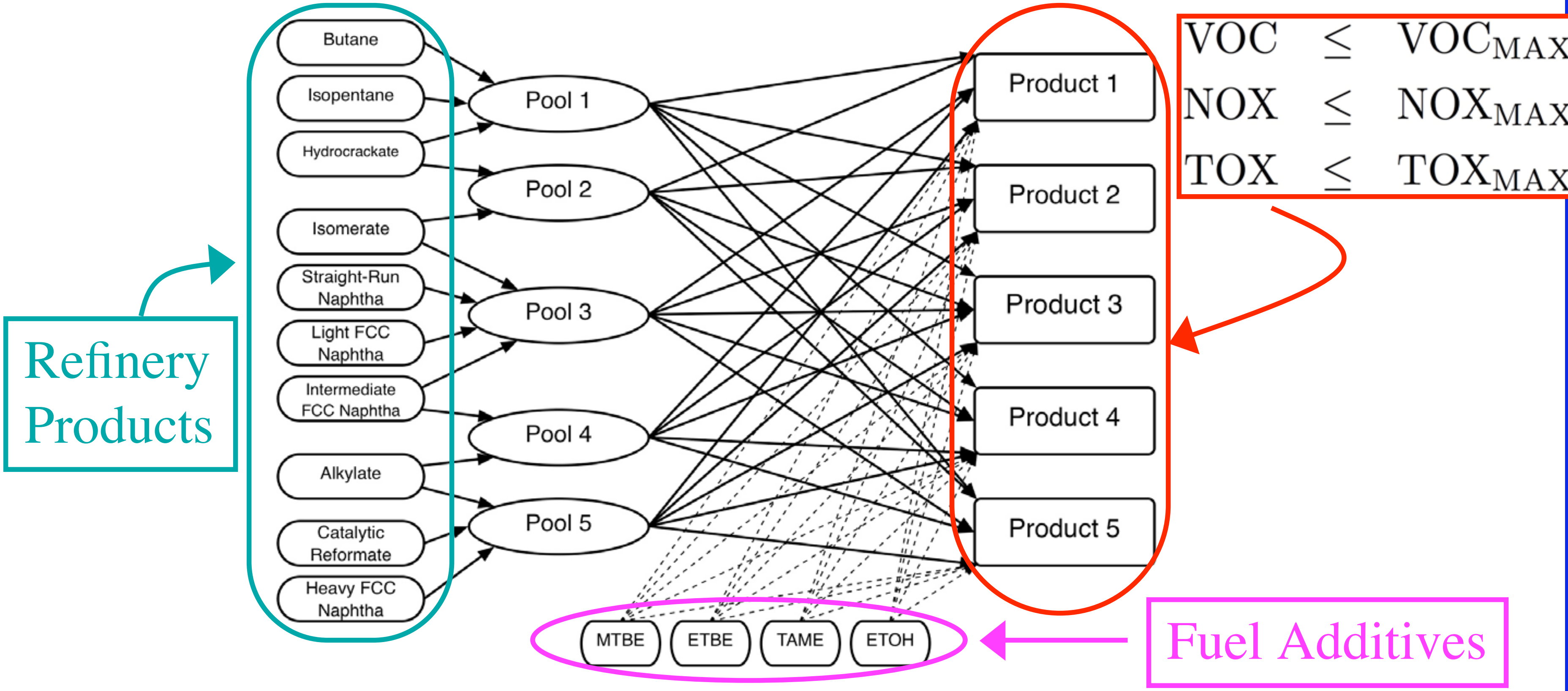
Partitioning on a **selected portion of the bilinear terms** helps the convergence

For a topology this large, N = 5 no longer solves in a reasonable time

Outline

- Deterministic Global Optimization: Objectives & Motivation
- Convex Envelopes:
  - Trilinear Monomials
  - Edge Concave functions
- Piecewise linearization of Bilinear terms
- Checking Convexity: Products of Univariate Functions
- PαBB: Piecewise Quadratic Perturbations
- **Pooling Problems: Standard, Generalized & Extended - Towards addressing Large-Scale Global Optimization Problems**
- Conclusions

The Extended Pooling Problem



- Given refinery exit streams, **meet volatile organics, NO_X, & toxic emissions standards** for each gasoline blend according to the EPA Complex Emissions Model & legislative bounds

MINLP Model of the Extended Pooling Problem

- To integrate the EPA Complex Emissions Model into problem framework, introduce outflow variable (of_j) from each product:

$$of_j = \sum_{l:(l,j)\in T_Y} y_{l,j} + \sum_{i:(i,j)\in T_Z} z_{i,j} \quad \forall\, j$$

and define the product qualities ($u_{j,k}$) that **blend linearly**:

$$(u_{j,k})\cdot(of_j) = \sum_{\substack{l:(l,j)\in T_Y\\ i:(i,l)\in T_X}} C_{i,k}\cdot q_{i,l}\cdot y_{l,j} + \sum_{i:(i,j)\in T_Z} C_{i,k}\cdot z_{i,j} \quad \forall\, j, \forall\, k$$

and Reid Vapor Pressure, which **blends by a power law**:

$$\widehat{u_{j,3}}\cdot(of_j) = \sum_{i:(i,l)\in T_X}\sum_{l:(l,j)\in T_Y} C_{i,3}^{1.25}\cdot q_{i,l}\cdot y_{l,j} + \sum_{i:(i,j)\in T_Z} C_{i,3}^{1.25}\cdot z_{i,j} \quad \forall\, j$$

$$\widehat{u_{j,3}} = u_{j,3}^{1.25}$$

MINLP Model of the Extended Pooling Problem

- Relate the outflow product qualities ($u_{j,k}$) to the EPA Model variables using the specifications in 40 CFR80.45

Many variables in the EPA Complex Emissions Model depend on the range of the quality (*e.g.*, the toxics model 300°F distillation fraction is set to 95 when the quality is greater than 95)

$\mathrm{OXY}_j = u_{j,1} \; \forall j \qquad \mathrm{BEN}_j = \ldots$

In the EPA Model, Reid Va[...] Pressure is set to 8.7 psi in t[...]

$\mathrm{RVP}_j = \begin{cases} \ldots \\ 8.7 & \text{Winter} \end{cases} \; \forall j$

$u_{j,5} - 95 \geq (u^{\mathrm{L}}_{j,5} - 95) \cdot y_{\mathrm{E300},j}$

$u_{j,5} \ldots$

Some of the variables used in the EPA Model match the true quality

$$\begin{aligned}
\ldots &\leq (u^{\mathrm{U}}_{j,5} - 95) \cdot y_{\mathrm{E300},j} \\
\mathrm{E300}^T_j - 95 &\geq (u^{\mathrm{L}}_{j,5} - 95) \cdot y_{\mathrm{E300},j} \\
\mathrm{E300}^T_j - u_{j,5} &\leq (u^{\mathrm{U}}_{j,5} - u^{\mathrm{L}}_{j,5}) \cdot (1 - y_{\mathrm{E300},j}) \\
\mathrm{E300}^T_j - u_{j,5} &\geq (u^{\mathrm{L}}_{j,5} - u^{\mathrm{U}}_{j,5}) \cdot (1 - y_{\mathrm{E300},j})
\end{aligned}$$

The Extended Pooling Problem

$$\mathrm{TOX}_j = \mathrm{BENZ}_j + \mathrm{FORM}_j + \mathrm{ACET}_j + \mathrm{BUTA}_j + 10 \cdot \mathrm{NEBENZ}_j + \mathrm{POM}_j \quad \forall j$$

There are three regulated components in the EPA Model:

volatile organics, NO_X, & toxic emissions

Toxics emissions (TOX_j) is the sum of six components:

exhaust benzene ($BENZ_j$), formaldehyde ($FORM_j$), acetaldehyde ($ACET_j$), 1,3-butadiene ($BUTA_j$), nonexhaust benzene ($NEBENZ_j$), & polycyclic organic matter (POM_j)

Emissions standards must be met for each product *j*

The Extended Pooling Problem

Each of the three regulated emissions is modeled with a nonlinear expression. The toxics model is the only convex one of the three.

$$\mathrm{TOX}_j = \mathrm{BENZ}_j + \mathrm{FORM}_j + \mathrm{ACET}_j + \mathrm{BUTA}_j + 10 \cdot \mathrm{NEBENZ}_j + \mathrm{POM}_j \quad \forall j$$

$$\mathrm{BENZ}_j = \sum_{e=1}^{2} \frac{\mathrm{BENZ}(b) \cdot w_e^T}{e^{b_e(b)}} \times \exp\{t_{BE,e,j}\},$$

$$t_{BE,e,j} = c_{e,1}^{BE}\mathrm{OXY}_j + c_{e,2}^{BE}\mathrm{SUL}_j + c_{e,3}^{BE}\mathrm{E300}_j + c_{e,4}^{BE}\mathrm{ARO}_j + c_{e,5}^{BE}\mathrm{BEN}_j$$

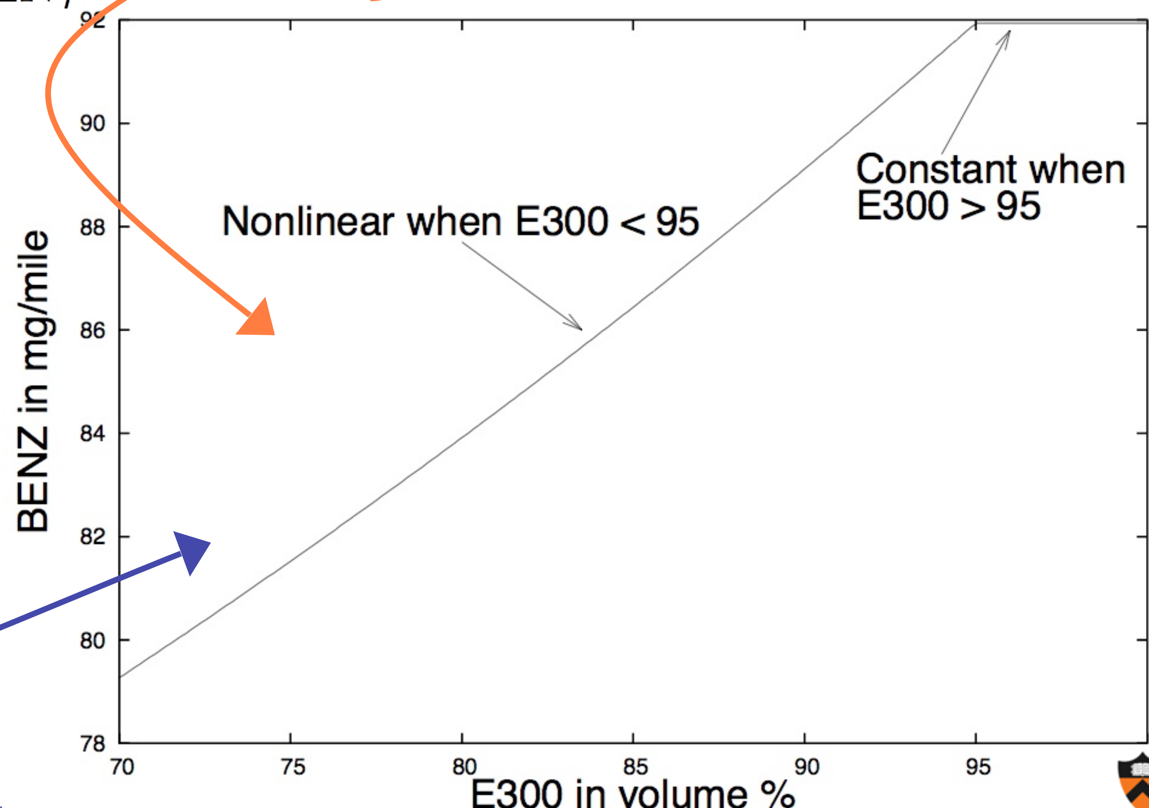
nonlinear (albeit convex) equation

The Extended Pooling Problem

$$\mathrm{BENZ}_j = \sum_{e=1}^{2} \frac{\mathrm{BENZ}(b)\cdot w_e^T}{e^{b_e(b)}} \times \exp\{t_{BE,e,j}\},$$

$$t_{BE,e,j} = c_{e,1}^{BE}\mathrm{OXY}_j + c_{e,2}^{BE}\mathrm{SUL}_j + c_{e,3}^{BE}\mathrm{E300}_j + c_{e,4}^{BE}\mathrm{ARO}_j + c_{e,5}^{BE}\mathrm{BEN}_j$$

Benzene is a component of the toxic emissions model



Logical disjunctions extend the accurate range of the model but introduce nonconvexities into the optimization problem

The Extended Pooling Problem

Volatile organics is another of the 3 regulated components in the EPA Model (in addition to NO_X, & toxic emissions)

$$\mathrm{VOC}_j = \mathrm{VOCE}_j + \mathrm{VOCNE}_j$$

In the summer, the non-exhaust volatile organic emissions are a **non-convex quadratic function** of Reid Vapor Pressure

$$\mathrm{VOCNE}_j = \begin{cases} 0.0 & \text{Winter} \\ \alpha_1^V + \alpha_2^V \cdot \mathrm{RVP}_j + \alpha_3^V \cdot \mathrm{RVP}_j^2 & \text{Summer} \end{cases} \quad \forall j$$

The Extended Pooling Problem

Volatile organics is another of the 3 regulated components in the EPA Model (in addition to NO_X, & toxic emissions)

$$\mathrm{VOC}_j = \mathrm{VOCE}_j + \mathrm{VOCNE}_j$$

Exhaust volatile organic emissions: **product of an exponential & polynomial function**

$$\mathrm{VOCE}_j = \sum_{e=1}^{2} \frac{\mathrm{VOC}(b) \cdot w_e^V}{e^{v_e(b)}} \cdot \exp\{t_{V,e,j}\} \cdot \{1 + f^V_{\mathrm{EXT},e,1}(\mathrm{E200}_j) + f^V_{\mathrm{EXT},e,2}(\mathrm{E300}_j, \mathrm{ARO}_j)\}$$

$$t_{V,e,j} = c^V_{e,1} \cdot \mathrm{OXY}_j + c^V_{e,2} \cdot \mathrm{SUL}^V_j + c^V_{e,3} \cdot \mathrm{RVP}_j + c^V_{e,4} \cdot \mathrm{E200}^V_j + c^V_{e,5} \cdot \mathrm{E300}^V_j + c^V_{e,6} \cdot \mathrm{ARO}^V_j + c^V_{e,7} \cdot \mathrm{OLE}^V_j + c^V_{e,8} \cdot (\mathrm{E200}^V_j)^2 + c^V_{e,9} \cdot (\mathrm{E300}^V_j)^2 + c^V_{e,10} \cdot \mathrm{ARO}^V_j \cdot \mathrm{E300}^V_j$$

The function in the exponential is itself nonconvex

Extended Pooling Problem:
Characterizing Nonlinear Terms

Small, Mid-Size & Large Test Cases have 108, 180 & 640 nonlinear terms, respectively

TABLE 1. Topology of the Extended Pooling Problem Test Cases

Underestimate the Nonlinear Terms using …

	Stocks	Pools	Products
Small Case	[illegible]	[illegible]	2
Mid-Size Case	[illegible]	[illegible]	3
Large Case	[illegible]	5	1[illegible]

Outer Approximation

Edge-Concave Techniques (Meyer & Floudas; 2005)

Piecewise-Linear Relaxations (Gounaris, Misener, Floudas; 2009)

…e Extended Pooling Problem

	Variables		Nonlinear Terms			
	Contin	Binary	Bilinear	Quad	Expon	Higher-Order Poly
Small Case	212	30	62	12	24	8
Mid-Size Case	331	45	111	18	36	12
Large Case	1104	150	410	60	120	40

Extended Pooling Problem:
Relaxation of EPA Model for NEBENZ

- The paradigm of edge-concavity efficiently generates a tight lower bound on the EPA Model of nonexhaust benzene (a toxic emissions component)
- The EPA nonexhaust benzene model would be edge-concave iff:

Generate a tight relaxation by subtracting a term from NEBENZ_j to satisfy the 2nd derivative property. Derive convex hull of the edge-concave portion using the method of Meyer & Floudas, *Math. Prog.* [2005] & relax the remaining portion with recursive arithmetic

arithmetic [Maranas & Floudas, *J. Global Optim.* 1995; Ryoo & Sahinidis, *J. Global Optim.*, 2001]

$$\frac{\partial^2 \cdots}{\partial \text{BEN}_j^2} = 0 \leq 0 \quad \forall j$$

The 1st equation is not valid, so NEBENZ is not edge concave. But:

$$\text{NEBENZ}_j - \alpha_4' \cdot \text{RVP}_j^2 \cdot \text{BEN}_j$$

is edge-concave when:

$$\alpha_4' = \alpha_4^{NB} + 3 \cdot \alpha_6^{NB} \cdot \text{RVP}_j^{\text{L}} + \alpha_7^{NB} \cdot \text{MTB}_j^{\text{L}}$$

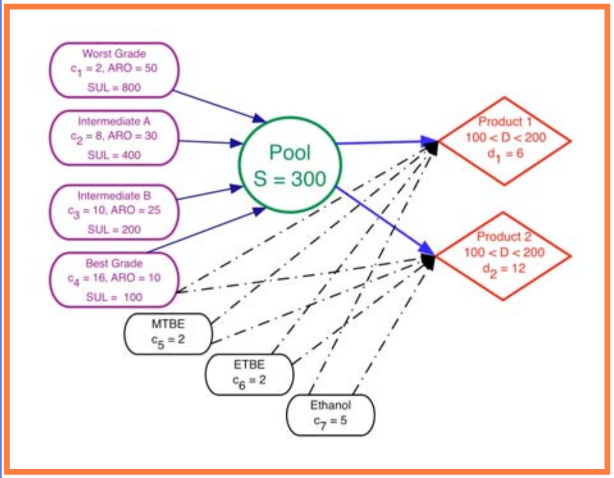
Extended Pooling Problem:
Relaxation of EPA Model for NEBENZ

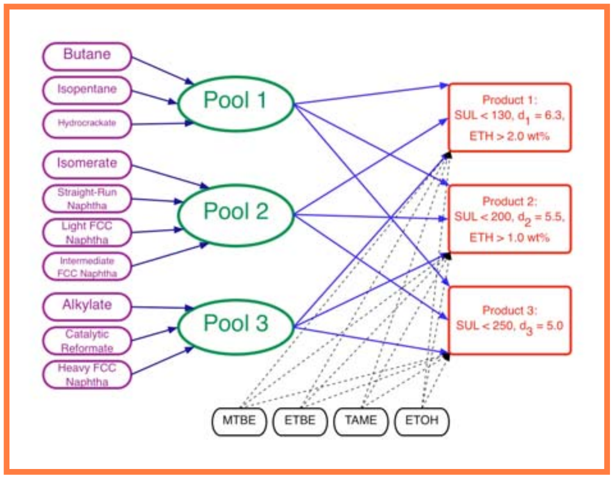
Tardella, *Discret. Appl. Math.* [1988]; Tardella [2003, 2008]; Maranas & Floudas, *J. Global Optim.* [1995]; Ryoo & Sahinidis, *J. Global Optim.* [2001]; Meyer & Floudas, *Math. Prog.* [2005]

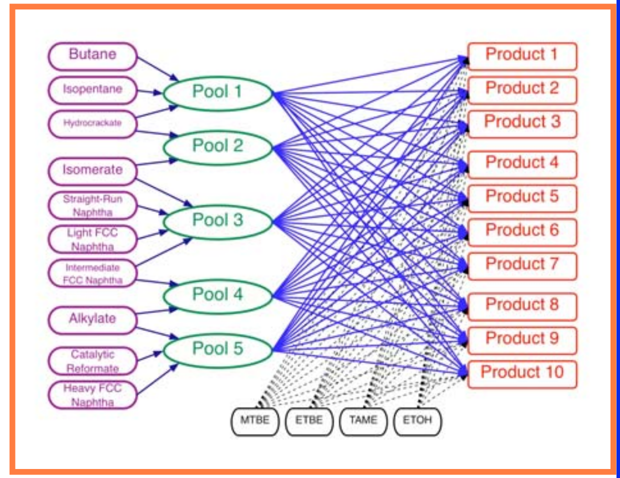
- Generating the convex hull of the edge-concave portion with the Meyer & Floudas, *Math. Prog.* [2005] method produces 22% improvement in the relaxation lower bound without requiring extra time:

	Recursive Arith Rlx	Edge-Concave Based Relax.
LB of NEBENZ in Reg1 (mg/mile)	-13.56	-10.61
CPU (s)	0.01	0.01
LB of NEBENZ in Reg2 (mg/mile)	-11.49	-9.01
CPS (s)	< 0.01	< 0.01

Extended Pooling Problem:
Characterizing the Case Studies





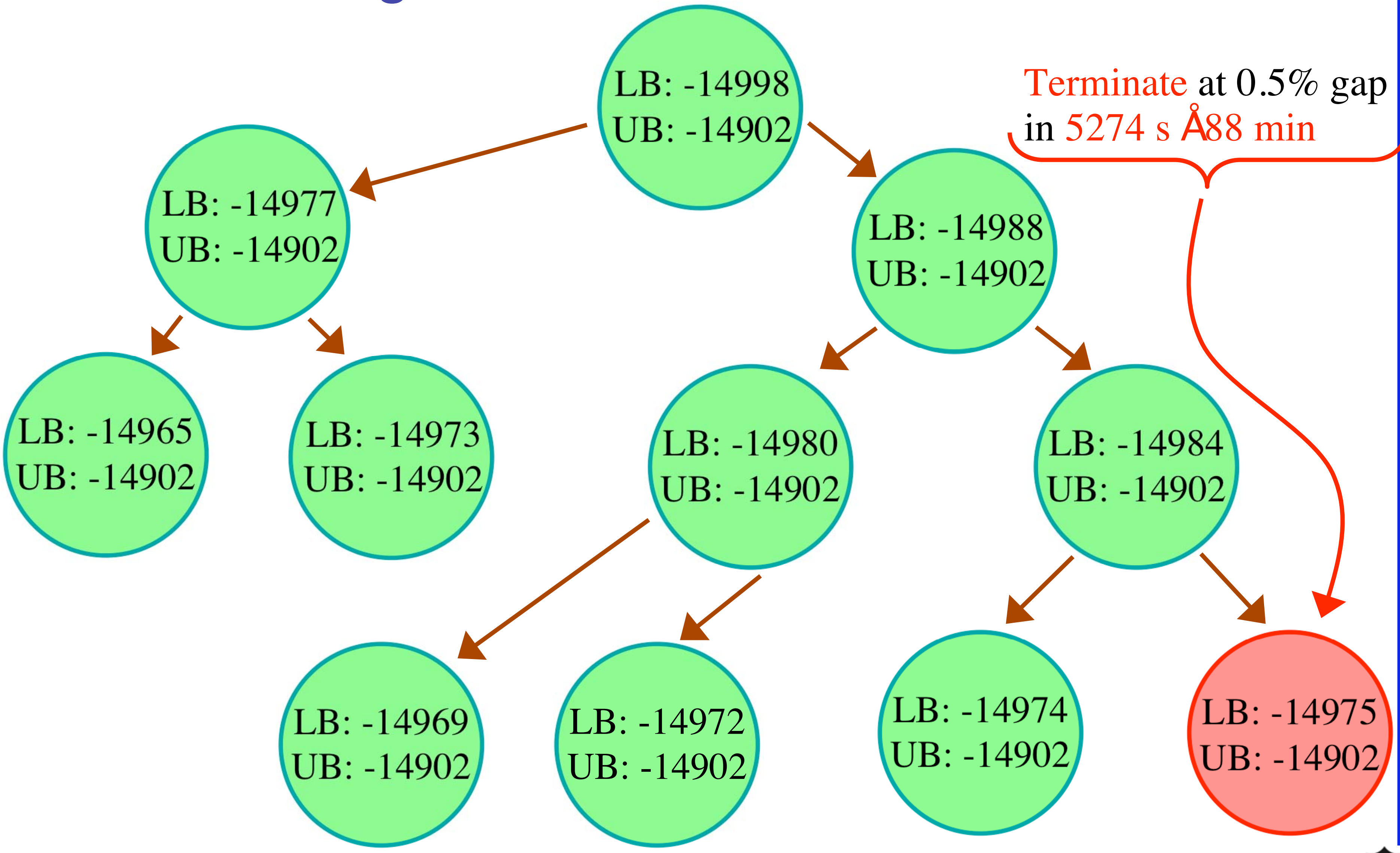


- **Small**
  - –214 Contin Vars
  - –30 Binary Vars
  - –108 Nonlin Terms

- **Mid-Size**
  - –331 Contin Vars
  - –45 Binary Vars
  - –180 Nonlin Terms

- **Large**
  - –1104 Contin Vars
  - –150 Binary Vars
  - –640 Nonlin Terms

Extended Pooling Problem:
Large Case: Branch & Bound



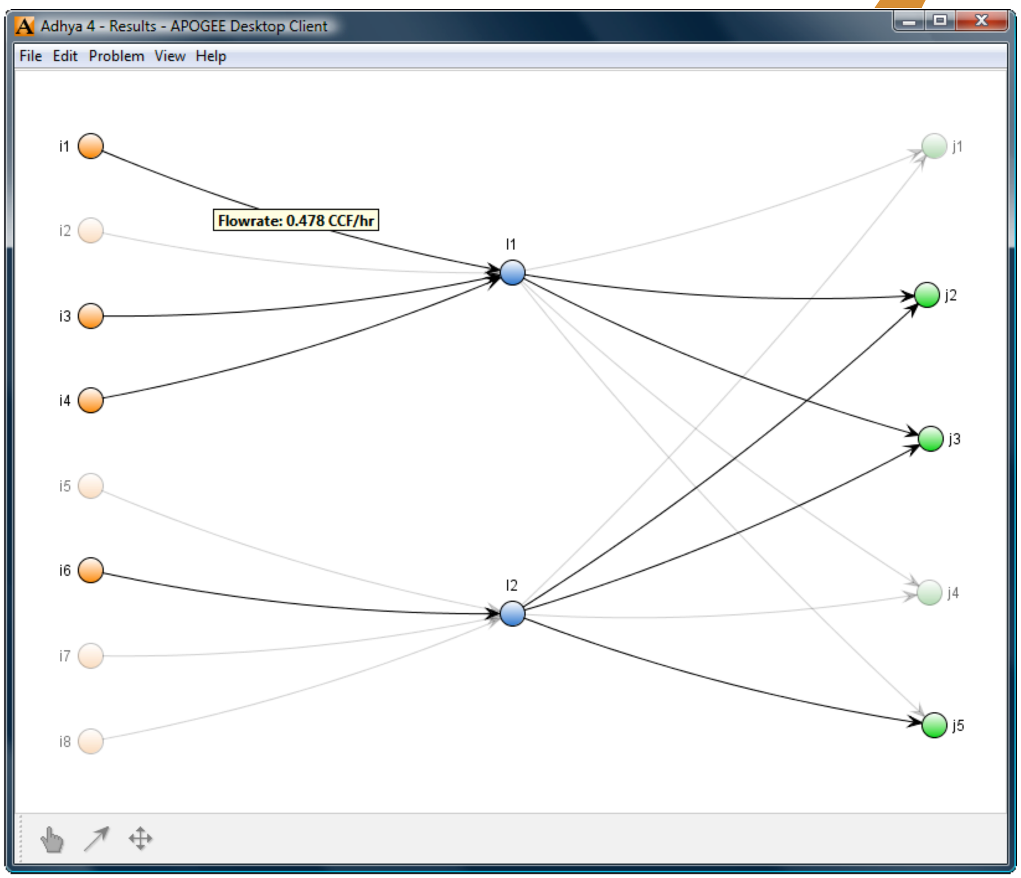
APOGEE

helios.princeton.edu/APOGEE/

Algorithms for Pooling-problem global Optimization in GEneralized and Extended classes

APOGEE Desktop Client

- Submit problem specifications to the solver.



Problem specification

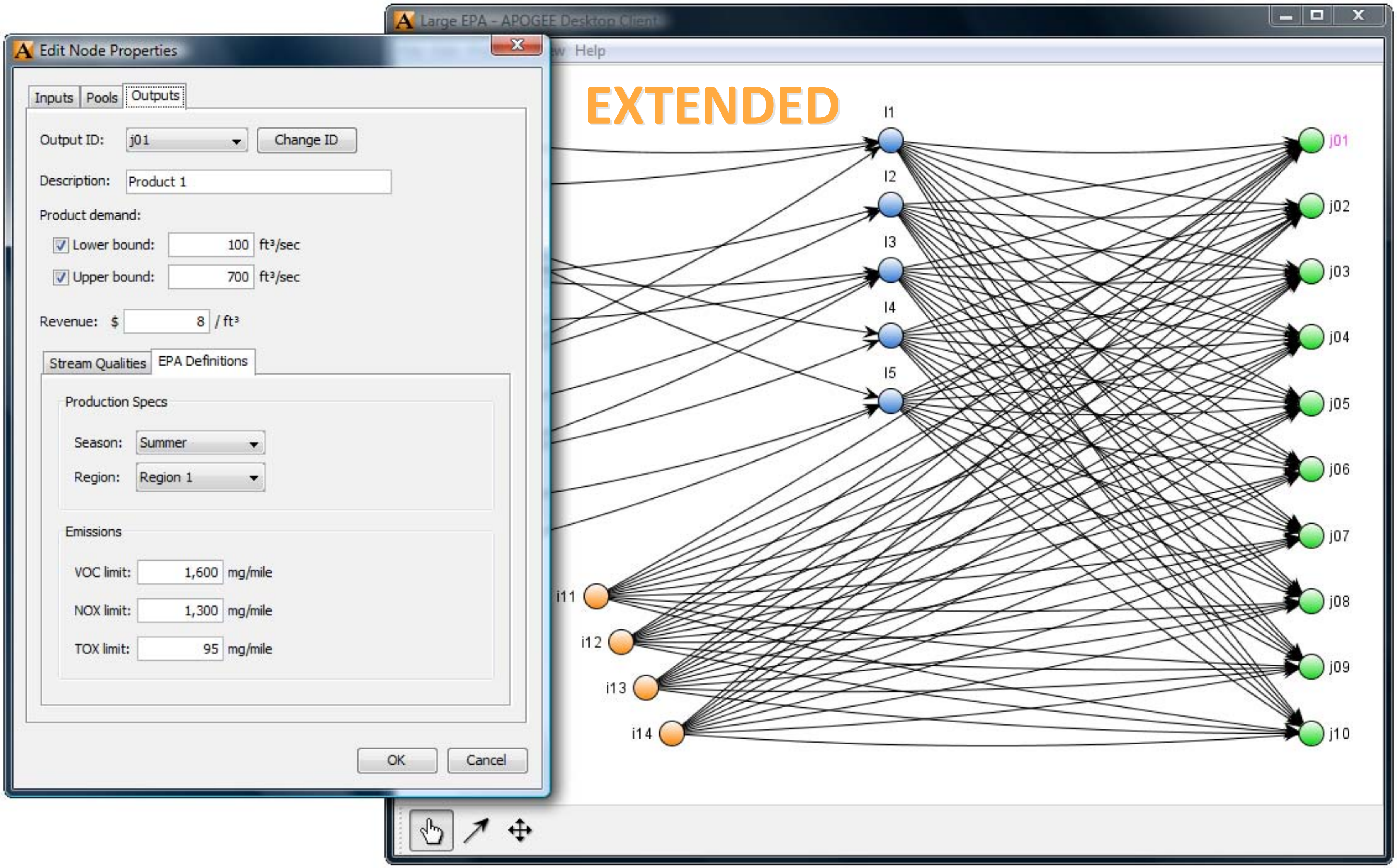
Solver

- Piecewise-linear and edge-concave relaxations
- Branch-and-bound algorithm

Results

Problem: Adhya, Tawarmalani, and Sahinidis, *Ind. Eng. Chem. Res.* 38: 1965-72, 1999

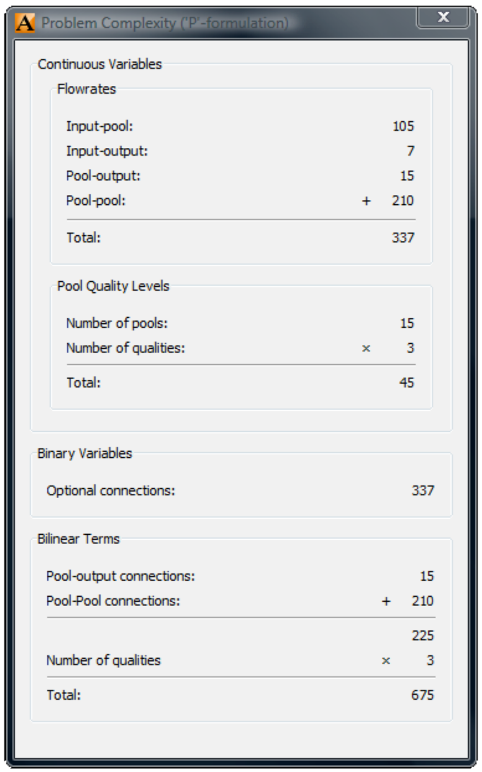
Globally Optimize Pooling Problems in Three Classes

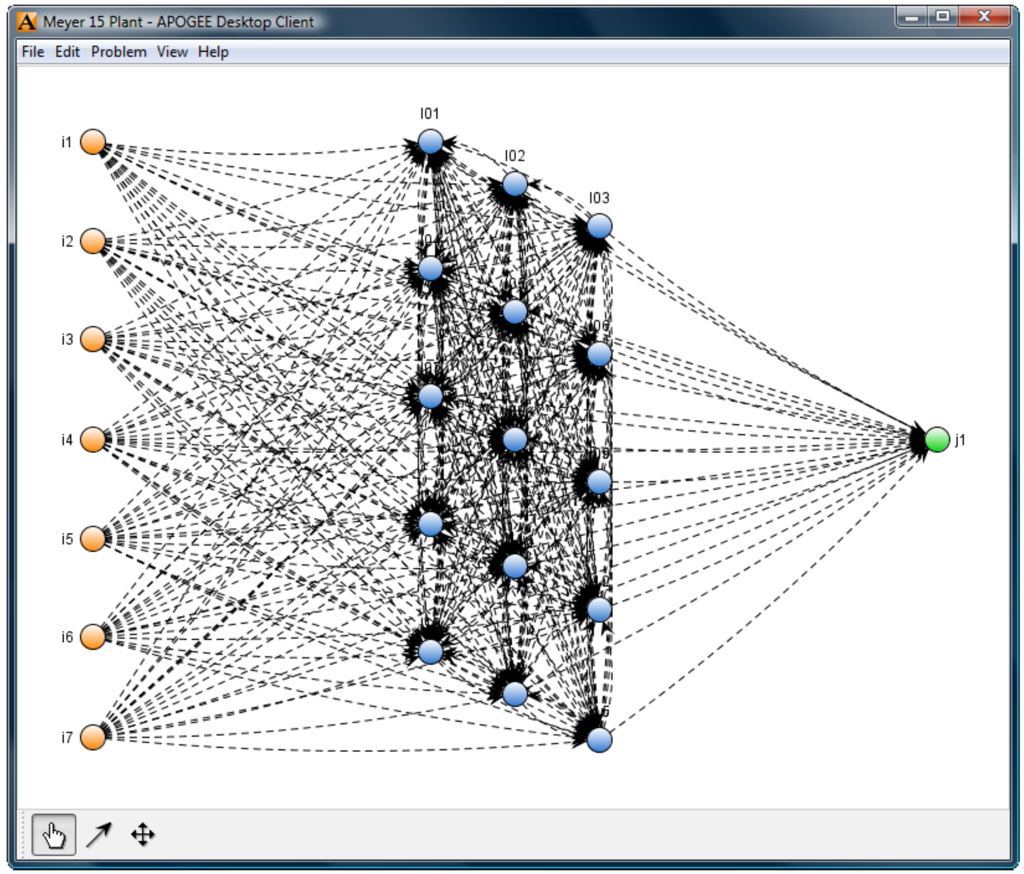


[illegible]

Problem Complexity

- Monitor complexity as problem is constructed.
- Globally optimize large-scale instances.





Problem: Meyer and Floudas. *AIChE J.* **2006**, *52*, 3027-41.
Problem: Ruiz and Grossmann. *Comput Chem Eng*, **2010**, *34*, 914-30.

Conclusions

- Motivational Areas & Review of contributions
- Convex Envelopes: Trilinear Monomials;
  Edge Concave Functions
- Piecewise Relaxation of Bilinear Terms
- Checking Convexity:Products of Univariate Functions
- PαBB: Piecewise Quadratic Perturbation Based αBB
- Generalized & Extended Pooling Problems: Large Scale Global Optimization Successes

Exciting theoretical and algorithmic advances with potential impact on several application areas

Acknowledgements

Claire S. Adjiman	Imperial College
Ioannis Akrotirianakis	SAS
Ioannis P. Androulakis	Rutgers University
Stavros Caratzoulas	University of Delaware
William R. Esposito	Praxair
Chrysanthos Gounaris	Princeton University
Zeynep H. Gumus	Cornell Medical School
Steven T. Harding	Process Combinatorics
Marianthi G. Ierapetritou	Rutgers University
Josef Kallrath	BASF
John L. Klepeis	D.E. Shaw
Xiaoxia Lin	Harvard Medical School
Costas D. Maranas	Penn State University
Clifford A. Meyer	Dana Farber Cancer Institute
Ruth Misener	Princeton University
Arnold Neumaier	University of Vienna
Heather D. Schafroth	Cornell University
Karl M. Westerberg	CCSF
Carl A. Schweiger	Pavilion Technologies

National Science Foundation
National Institutes of Health
AspenTech, BASF