Convexification and Global Optimization of Nonlinear Programs

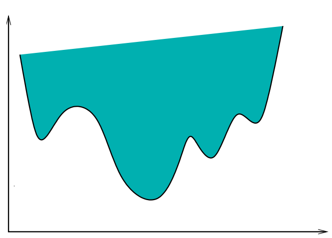
Mohit Tawarmalani
mtawarma@mgmt.purdue.edu
Krannert School of Management
Purdue University

Chemnitz 2004

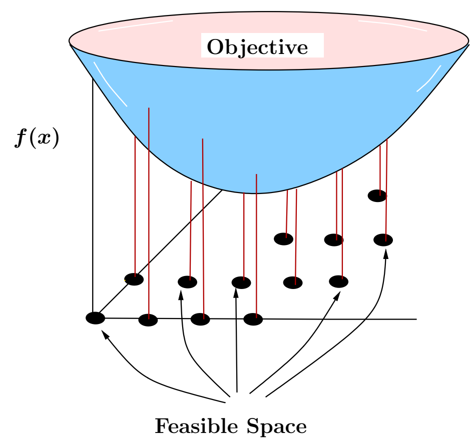
Mixed-Integer NLP

(P)	**min**	$f(x,y)$	**Objective Function**
	s.t.	$g(x,y) \leq 0$	**Constraints**
		$x \in \mathbb{R}^n$	**Continuous Variables**
		$y \in \mathbb{Z}^p$	**Integrality Restrictions**

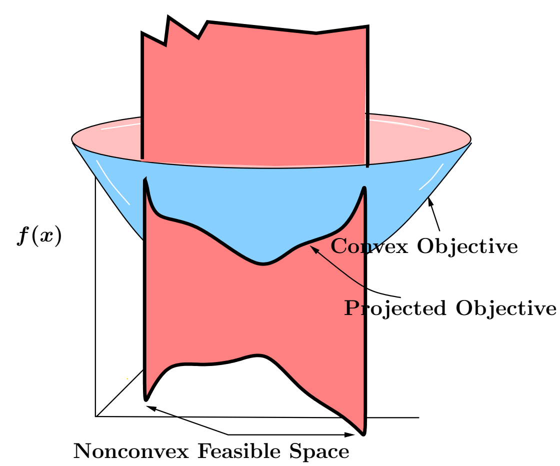
Challenges:



Multimodal Objective



Integrality



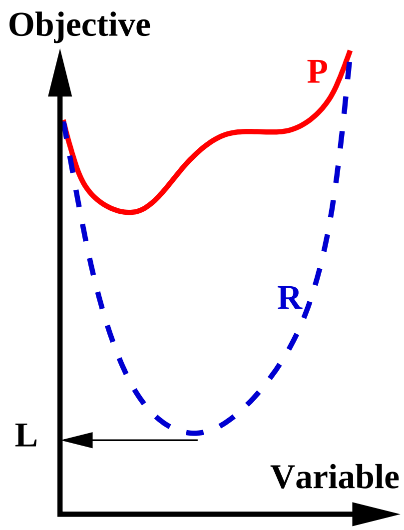
Nonconvex Constraints

DETERMINISTIC ALGORITHMS

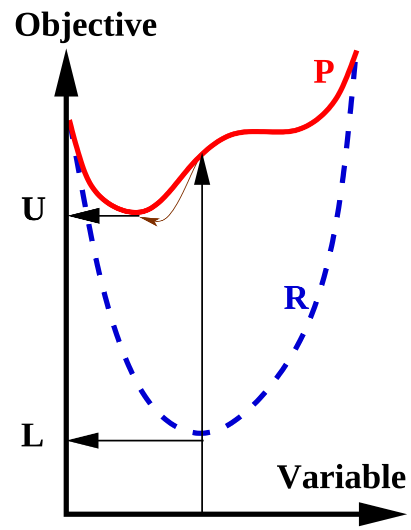
- **Branch-and-Bound**
  - **Bound problem over successively refined partitions**
    - » **Falk and Soland, 1969**
    - » **McCormick, 1976**
- **Convexification**
  - **Outer-approximate with increasingly tighter convex programs**
  - **Tuy, 1964**
  - **Sherali and Adams, 1994**
- **Decomposition**
  - **Project out some variables by solving subproblem**
    - » **Duran and Grossmann, 1986**
    - » **Visweswaran and Floudas, 1990**

- **Our approach**
  - **Branch-and-bound**
  - **Separable and differentiable reformulation**
  - **Constraint propagation & duality-based reduction**
  - **Convex envelopes and convexification**
- **Tawarmalani, M. and N. V. Sahinidis, *Convexification and Global Optimization in Continuous and Mixed-Integer Nonlinear Programming,* Kluwer Academic Publishers, Nov. 2002.**

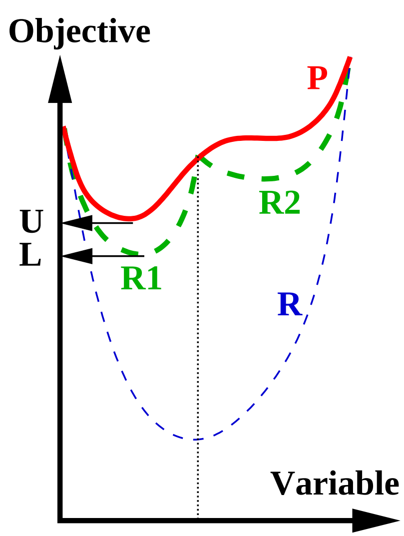
Branch and Bound Algorithm



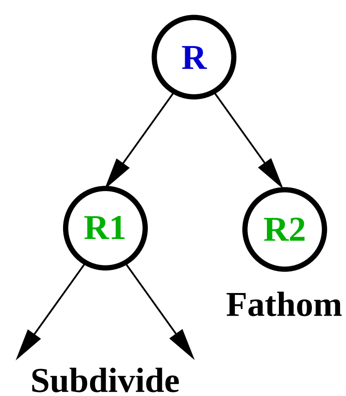
a. Lower Bounding



b. Upper Bounding



c. Domain Subdivision



d. Search Tree

Functional Decomposition

$$f(x, y, z, w) = \sqrt{\exp(xy + z \ln w) z^3}$$

$$\overbrace{\underbrace{\overbrace{\left(\exp(\underbrace{\underbrace{xy}_{x_1} + \overbrace{z \underbrace{\ln w}_{x_2}}^{x_3}}_{x_4})\right.}^{x_5} \; \overbrace{z^3}^{x_6}}_{x^7} \left.\right)^{0.5}}^{f}$$

$$\begin{aligned}
x_1 &= xy \\
x_2 &= \ln(w) \\
x_3 &= z x_2 \\
x_4 &= x_1 + x_3 \\
x_5 &= \exp(x_4) \\
x_6 &= z^3 \\
x_7 &= x_5 x_6 \\
f &= \sqrt{x_7}
\end{aligned}$$

- Introduce variables for intermediate quantities
- Retain terms with known convex envelopes
- Bound bilinear terms using McCormick's envelopes

Tawarmalani and Sahinidis, Mathematical Programming, 2004

Branch And Reduce Optimization Navigator

Components

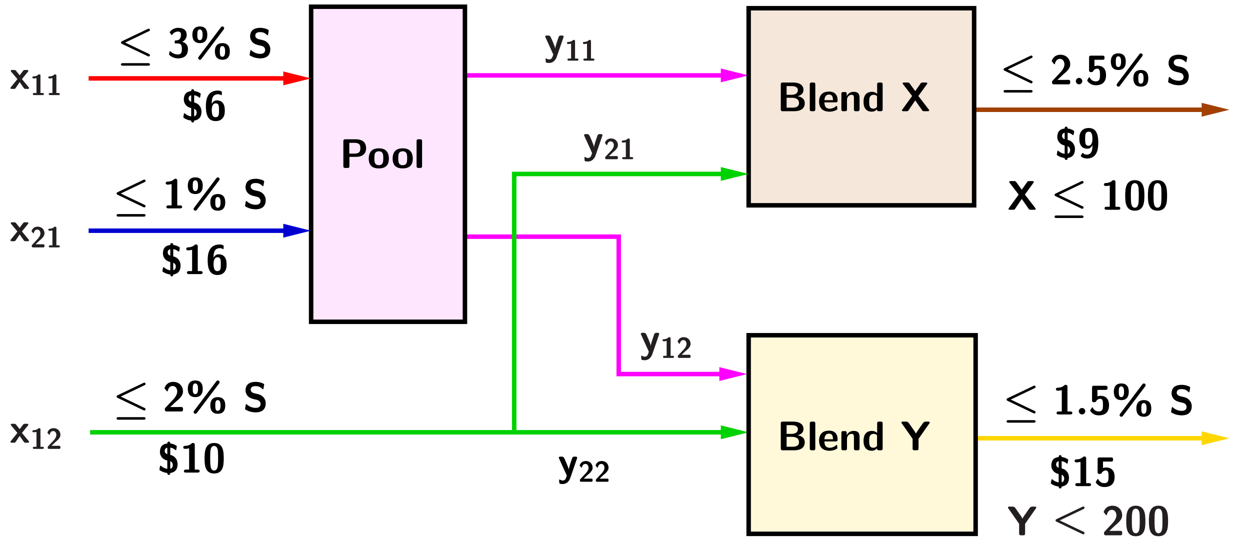
- Modeling Language
- Preprocessor
- Range Reduction
- Interval Arithmetic
- Automatic Differentiator
- IEEE Exception Handling
- Cutting Planes

Capabilities

- Fully automated MINLP solver
- Specialized modules: IP, LMP, IQP, FCP, . . .
- Links to CPLEX, OSL, MINOS, SNOPT
- Expandable Branch and Bound Framework

- GAMS links since November 2000; AIMMS link in 2004

Pooling Problem: (Haverly 1978)



$$\min \quad \overbrace{6x_{11} + 16x_{21} + 10x_{12}}^{\text{cost}} - \overbrace{9(y_{11} + y_{21})}^{X\text{-revenue}} - \overbrace{15(y_{12} + y_{22})}^{Y\text{-revenue}}$$

s.t.

Sulfur Mass Balance

$$q = \frac{3x_{11} + x_{21}}{y_{11} + y_{12}}$$

Quality Requirements

$$q y_{11} + 2y_{21} \leq 2.5(y_{11} + y_{21})$$

$$q y_{12} + 2y_{22} \leq 1.5(y_{12} + y_{22})$$

Mass Balance

$$x_{11} + x_{21} = y_{11} + y_{12}$$

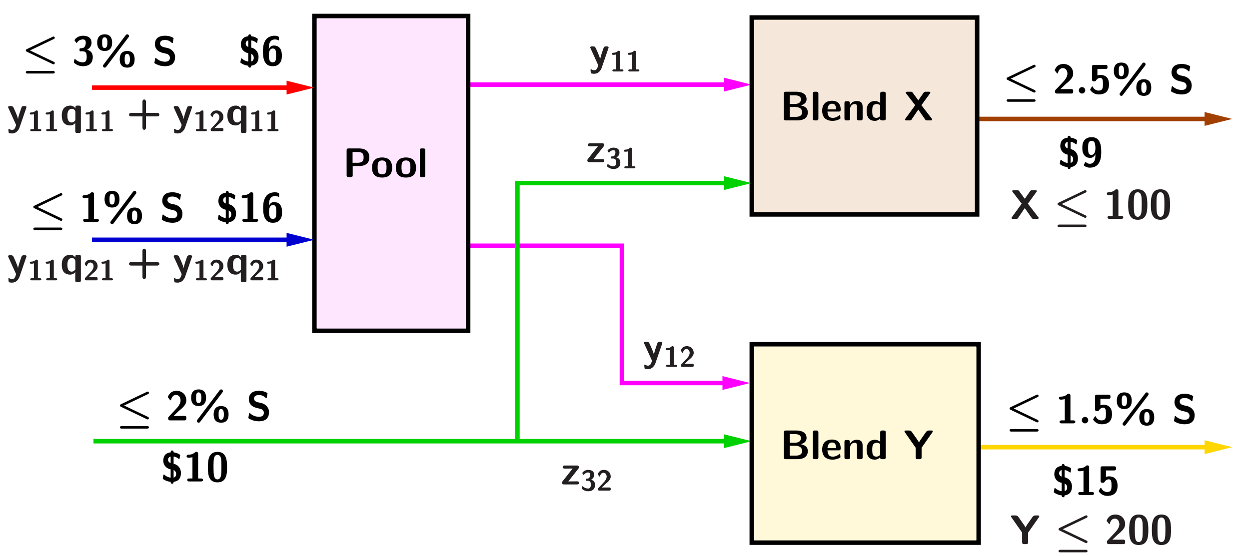
$$x_{12} = y_{21} + y_{22}$$

Demands

$$y_{11} + y_{21} \leq 100$$

$$y_{12} + y_{22} \leq 200$$

Pooling Problem: (Ben Tal 1994)



$$\min \overbrace{6\,(y_{11}q_{11} + y_{12}q_{11}) + 16\,(y_{11}q_{21} + y_{12}q_{21}) + 10\,(z_{31} + z_{32})}^{\text{cost}}$$

$$- \overbrace{9(y_{11} + y_{21})}^{X\text{-revenue}} - \overbrace{15(x_{12} + x_{22})}^{Y\text{-revenue}}$$

s.t. **Mass Balance**

$$q_{11} + q_{21} = 1$$

Quality Requirements

$$-0.5z_{31} + 3y_{11}q_{11} + y_{11}q_{21} \le 2.5y_{11}$$

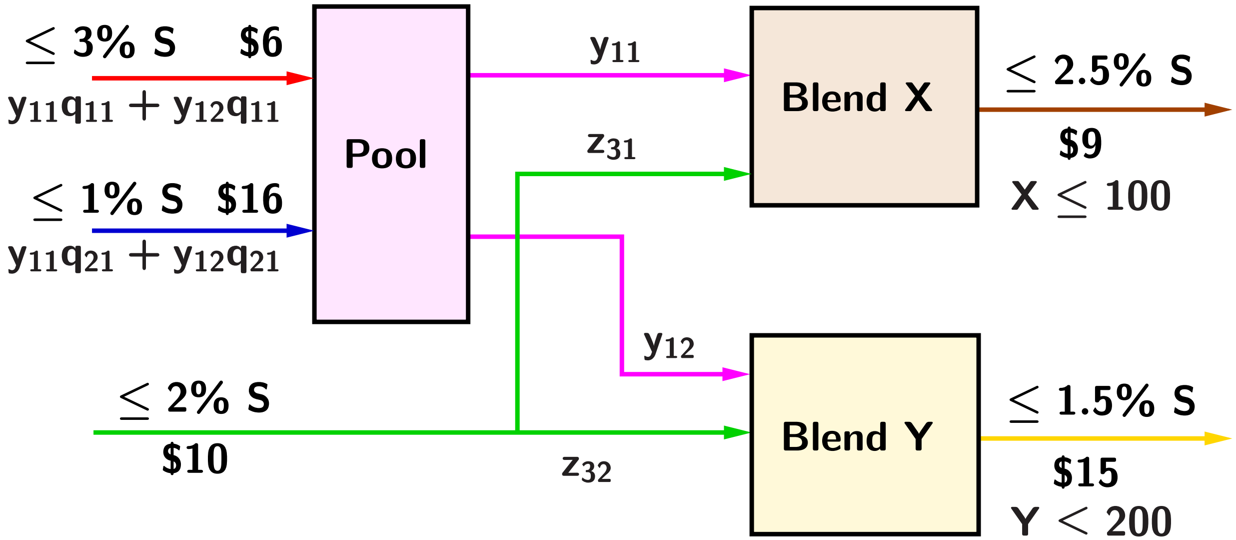
$$0.5z_{32} + 3y_{12}q_{11} + y_{12}q_{21} \le 1.5y_{12}$$

Demands

$$y_{11} + z_{31} \le 100$$

$$y_{12} + z_{32} \le 200$$

Pooling Problem: pq formulation



$$\min \quad \overbrace{6\,(y_{11}q_{11} + y_{12}q_{11}) + 16\,(y_{11}q_{21} + y_{12}q_{21}) + 10\,(z_{31} + z_{32})}^{\text{cost}}$$

$$- \overbrace{9(y_{11} + y_{21})}^{X\text{-revenue}} - \overbrace{15(x_{12} + x_{22})}^{Y\text{-revenue}}$$

s.t.

Mass Balance

$$q_{11} + q_{21} = 1$$

Quality Requirements

$$-0.5z_{31} + 3y_{11}q_{11} + y_{11}q_{21} \le 2.5y_{11}$$
$$0.5z_{32} + 3y_{12}q_{11} + y_{12}q_{21} \le 1.5y_{12}$$

Demands

$$y_{11} + z_{31} \le 100$$
$$y_{12} + z_{32} \le 200$$

Convexification Constraints

$$q_{11}y_{11} + q_{21}y_{11} = y_{11}$$
$$q_{11}y_{12} + q_{21}y_{12} = y_{12}$$

RELAXATION-ONLY CONSTRAINTS

- Can strengthen relaxation by adding to the model:
  - Nonlinear reformulations (RLT)
  - First-order optimality conditions
  - Problem-specific optimality conditions and symmetry-breaking constraints
- Traditionally, modeling languages for optimization pass single model
- RELAXATION_ONLY_EQUATIONS construct added to BARON's modeling language
- Strengthen relaxation without complicating local search

LOCAL SEARCH WITH CONOPT

	q-formulation			pq-formulation		
Problem	Objective	CPU s	Iter	Objective	CPU s	Iter
adhya1	-68.74	0.01	9	-56.67	0.00	5
adhya2	0.00	0.01	4	0.00	0.01	3
adhya3	-65.00	0.03	12	-57.74	0.02	7
adhya4	-470.83	0.01	9	-470.83	0.02	9
bental4	0.00	0.01	3	0.00	0.00	3
bental5	-2900.00	0.02	9	-2700.00	0.03	18
foulds2	-1000.00	0.00	6	-600.00	0.01	14
foulds3	-6.50	0.04	6	-6.50	0.09	9
foulds4	-6.00	0.04	6	-6.50	0.16	23
foulds5	-7.00	0.04	7	-6.50	0.06	7
haverly1	-400.00	0.00	5	0.00	0.00	3
haverly2	-400.00	0.00	5	0.00	0.01	3
haverly3	-750.00	0.01	8	0.00	0.00	3
rt97	inf	0.00	4	-4330.78	0.01	8
sum	-6074.07*	0.22	89	-3904.74*	0.41	107

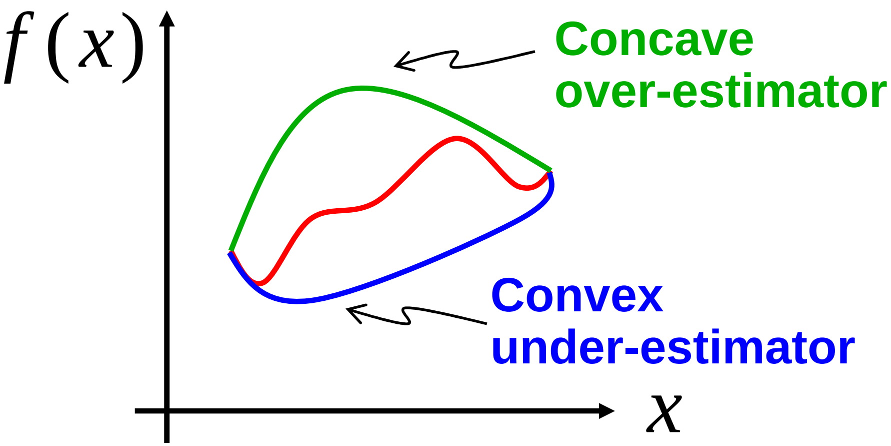
*: Not including rt97.

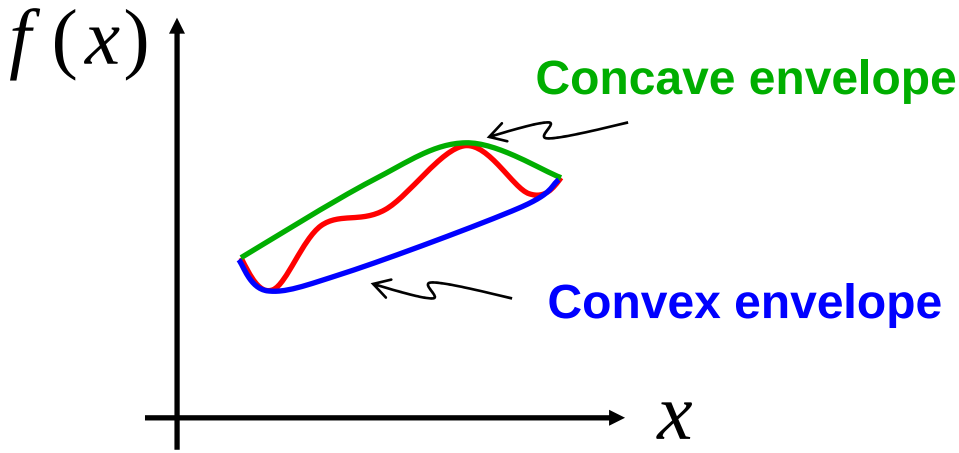
GLOBAL SEARCH WITH BARON

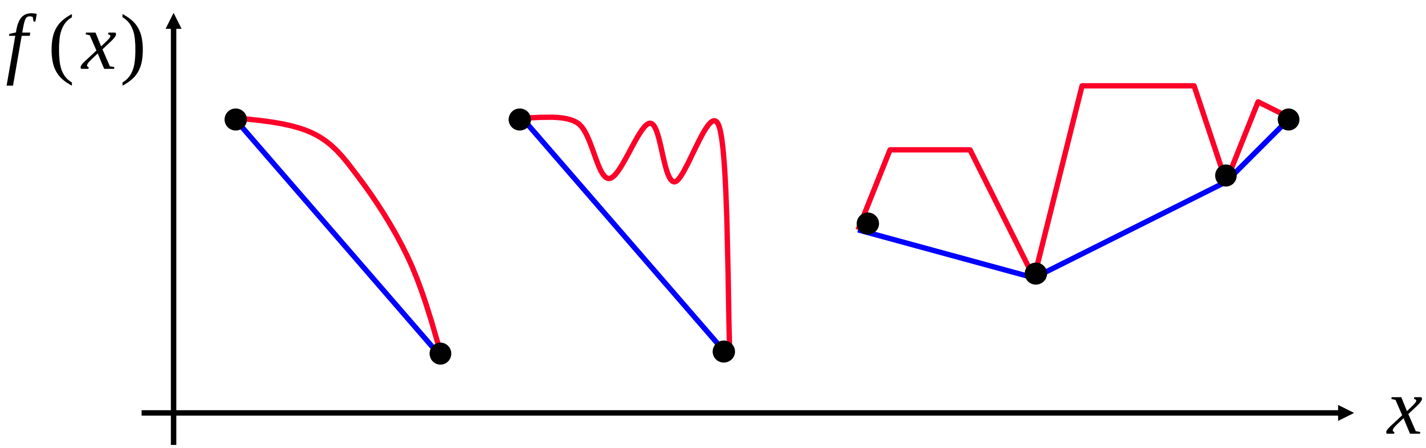
Problem	Strategy 1				Strategy 2				Strategy 3			
	N_t	N_o	N_m	CPU s	N_t	N_o	N_m	CPU s	N_t	N_o	N_m	CPU s
adhya1	573	550	50	17	30	24	7	1	28	24	7	0.5
adhya2	501	338	41	20	17	13	4	1	17	13	4	0.5
adhya3	9248*	2404	1800*	1200*	31	1	6	1.5	31	1	6	1.5
adhya4	6129*	-1	1620*	1200*	1	1	1	1.5	1	1	1	1
bental4	101	101	14	0.5	1	-1	1	0.5	1	-1	1	0.5
bental5	6445*	901	3815*	1200*	-1	-1	0	0.5	-1	-1	0	0
foulds2	1061	977	106	16	-1	-1	0	0	-1	-1	0	0
foulds3	348*	91	260*	1200*	-1	-1	0	1	-1	-1	0	5
foulds4	326*	262	246*	1200*	-1	-1	0	1	-1	-1	0	1
foulds5	389*	316	287*	1200*	-1	-1	0	1	-1	-1	0	1
haverly1	25	6	5	0	1	1	1	0	1	1	1	0
haverly2	17	1	5	0	1	1	1	0	1	1	1	0
haverly3	3	1	2	0	1	1	1	0	1	1	1	0
rt97	5629	2836	609	173.5	13	6	4	0.5	13	6	4	0.5
sum	30795	8783	8860	7427	91	42	26	10	89	42	26	12

*: Run did not terminate within 1200 seconds.

TIGHT RELAXATIONS







Convex/concave envelopes often finitely generated

The Generating Set of a Function

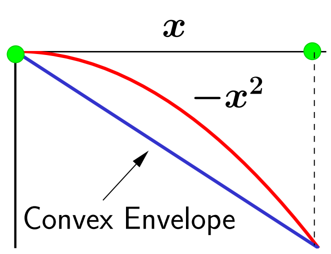
Definition:* The generating set of the epigraph of a function $g(x)$ over a compact convex set C is defined as

$$G_C^{\text{epi}}(g) = \left\{ x \;\middle|\; (x, y) \in \text{vert}\Big(\text{epi conv}\big(g(x)\big)\Big) \right\},$$

where vert$(\cdot)$ is the set of extreme points of $(\cdot)$.

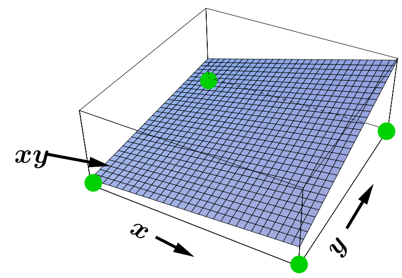
Examples:

$g(x) = -x^2$



$G_{[0,6]}^{\text{epi}}(g) = \{0\} \cup \{6\}$

$g(x) = xy$

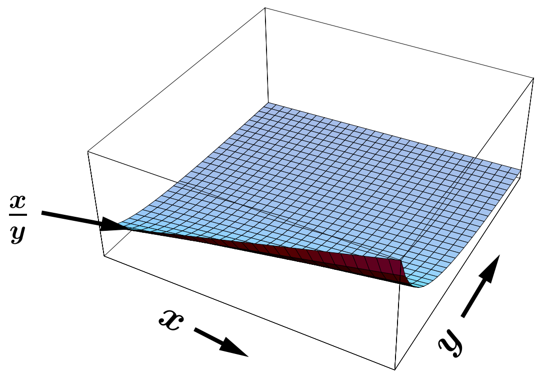


$G_{[1,4]^2}^{\text{epi}}(g) = \{1, 1\} \cup \{1, 4\} \cup \{4, 1\} \cup \{4, 4\}$

Identifying the Generating Set

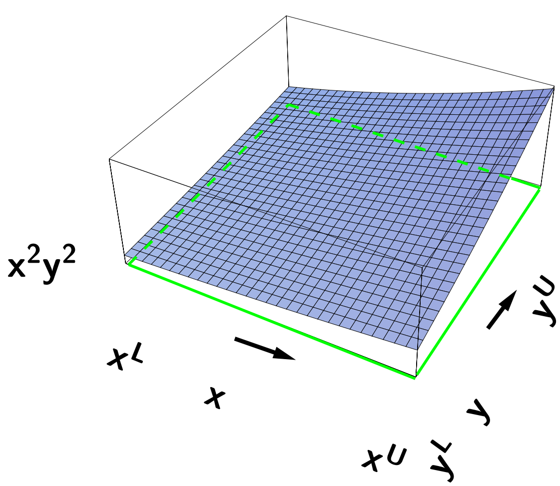
Characterization: $x_0 \notin G_C^{\text{epi}}(g)$ if and only if there exists $X \subseteq C$ and $x_0 \notin G_X^{\text{epi}}(g)$.

Example I: X is linear joining (x^L, y^0) and (x^U, y^0)



$$G^{\text{epi}}(x/y) = \{(x,y) \mid x \in \{x^L, x^U\}\}$$

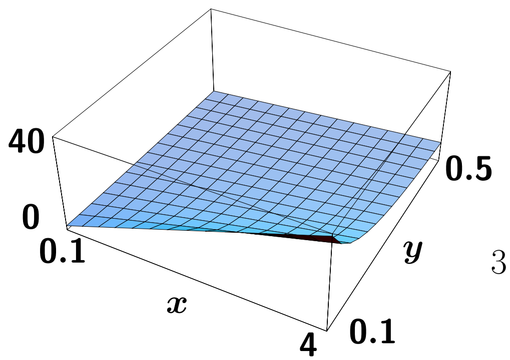
Example II: X is ϵ neighborhood of (x^0, y^0)



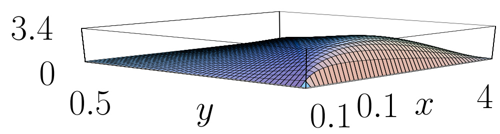
$$G^{\text{epi}}(x^2y^2) = \{(x,y) \mid x \in \{x^L, x^U\}\} \cup \{(x,y) \mid y \in \{y^L, y^U\}\}$$

Properties: Envelope of x/y

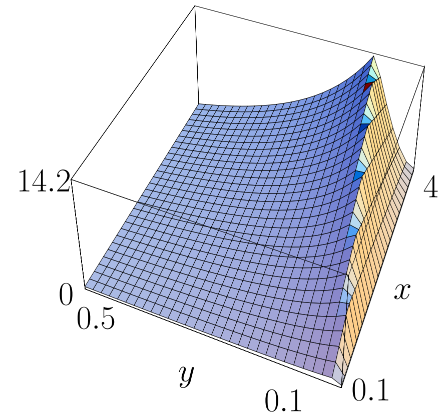
Comparison of Tightness:



Ratio: x/y



x/y – Envelope



x/y – Factorable

Maximum Gap: Envelope and Factorable Relaxation (using McCormick):

Point:

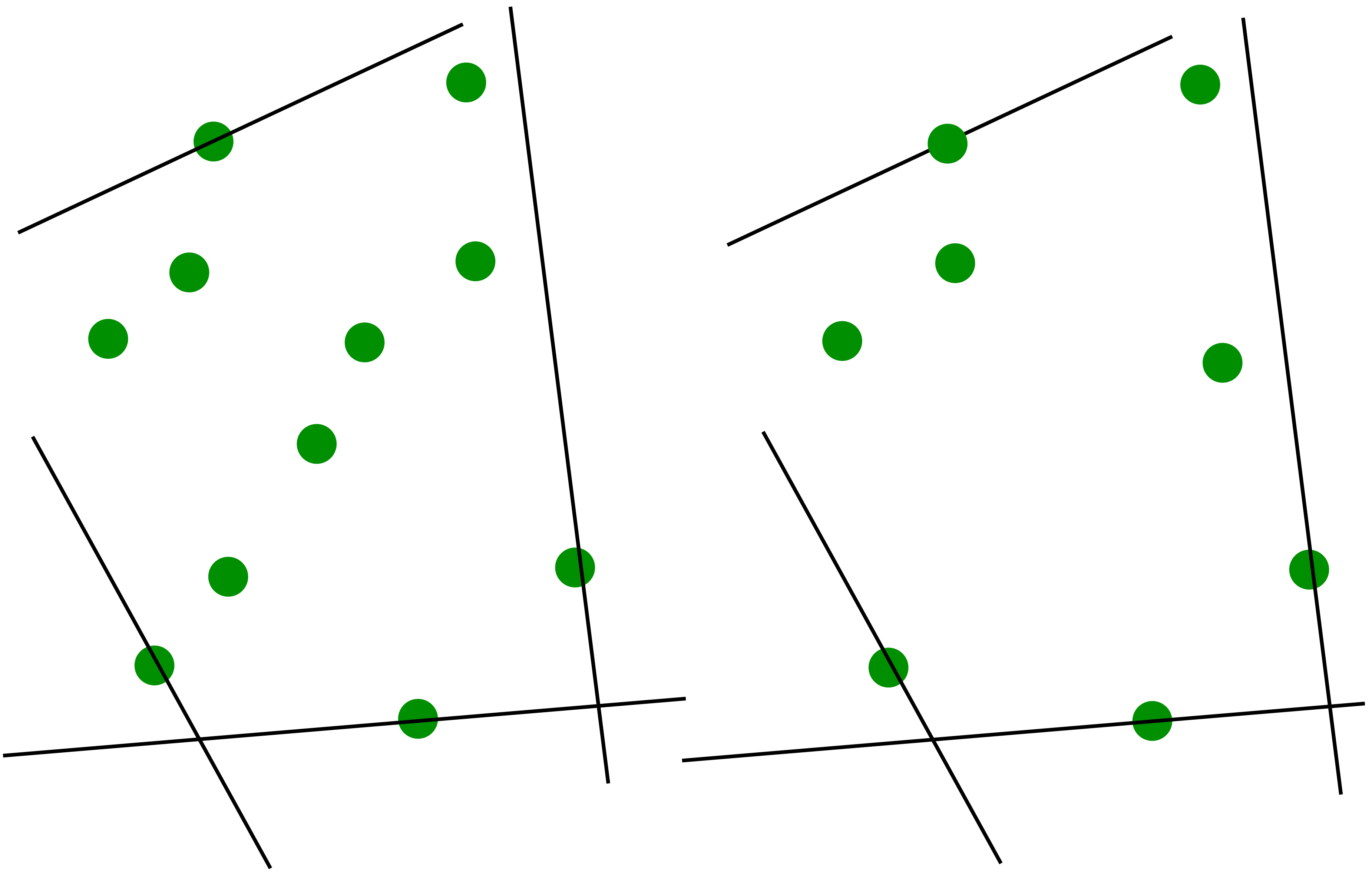
$$\left(x^U, y^L + \frac{y^L(y^U - y^L)(x^U y^U - x^L y^L)}{x^U {y^U}^2 - x^L {y^L}^2}\right)$$

Gap:

$$\frac{x^U (y^U - y^L)^2 (x^U y^U - x^L y^L)^2}{y^L y^U (2x^U y^U - x^L y^L - x^U y^L)(x^U {y^U}^2 - x^L {y^L}^2)}$$

Tawarmalani and Sahinidis, Mathematical Programming, 2002

A Geometric Perspective



Example: General Multilinear Functions

Definition: A general multilinear function

1. defined over a Cartesian product of polytopes $P = P_1 \times \cdots \times P_n$.
2. For any j, the function is linear in x_j ($x_j \in P_j$) when all x_k, $k \neq j$ ($x_k \in P_k$) are fixed.

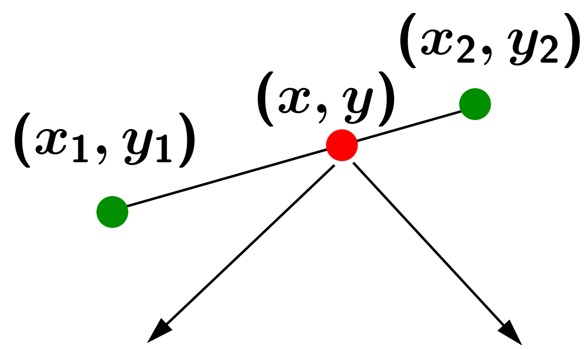
Known Fact: There exists an extreme point of P where the general multilinear function is optimized (minimized/maximized).

Simple Consequence: The following set can be convexified using disjunctive programming on "lifted" extreme points of P.

$$\begin{aligned} z_1 &= L_1(x_1, \ldots, x_n) \\ &\vdots \\ z_m &= L_m(x_1, \ldots, x_n) \\ x_i &\in P_i \end{aligned}$$

since $\sum_{i=1}^{n} \alpha_i x_i + \sum_{i=1}^{m} \beta_i L_i(x_1, \cdots, x_m)$ is also general multilinear.

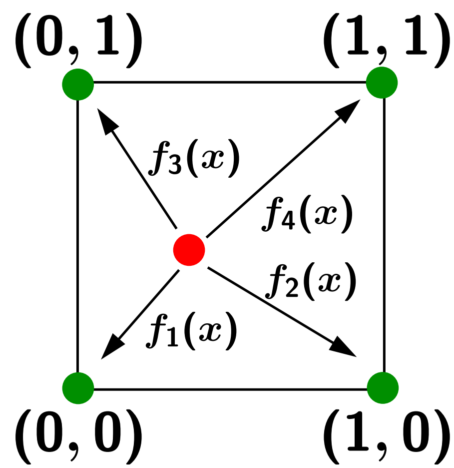
Inclusion Certificates: Two Points in a Plane



$$f_1(x,y) = \frac{x_2 - x}{x_2 - x_1} \quad f_2(x,y) = \frac{x - x_1}{x_2 - x_1}$$

Inclusion Certificate Has Different Forms but is Unique in this Case!

Inclusion Certificate: 0-1 Rectangle



Two Possible Certificates:

1. $f_1(x) = (1 - x_1)(1 - x_2), f_2(x) = x_1(1 - x_2), f_3(x) = (1 - x_1)x_2, f_4(x) = x_1x_2$
2. If $x_1 + x_2 - 1 > 0$,

   $$f_1(x) = 0,\ f_2(x) = 1 - x_2,\ f_3(x) = 1 - x_1,\ f_4(x) = x_1 + x_2 - 1$$

   else

   $$f_1(x) = 1 - x_1 - x_2,\ f_2(x) = x_1,\ f_3(x) = x_2,\ f_4(x) = 0$$

Certificate is not necessarily unique!

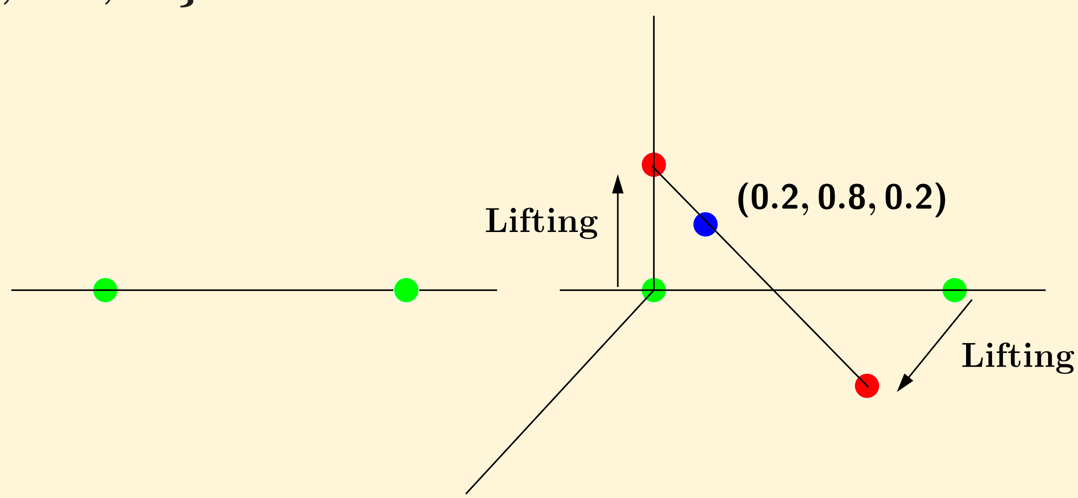
Convexifying Disjunctions

Theorem: (An informal statement) Consider the collection $\mathcal{C}$. The convex hull of $(x, \lambda_1, \ldots, \lambda_m)$ where $\lambda_1, \ldots, \lambda_m$ is any set of convex multipliers that provides a certificate of the inclusion of x in conv($\mathcal{C}$) is the same as the convex hull of the lifted sets $\{(x_i, e_i) \mid x_i \in C_i\}$ which in turn is:

$$\text{conv}\left\{(x, \lambda_1, \ldots, \lambda_m) \;\middle|\; \sum_{i=1}^{m} \lambda_i = 1, \sum_{i=1}^{m} \lambda_i x_i = x, \lambda_i \geq 0, \lambda_i x_i \in \lambda_i C_i, \right.$$
$$\left. \text{and } \lambda_i x_i = \lambda_i x \text{ if } i \in J \right\}$$

where $J \subseteq \{1, \ldots, m\}$.

Example:



The Motivation/Use of Theorem

The set:

$$z_1 \geq \frac{x_1 x_2}{y_1}$$

$$z_2 = x_1 x_2$$

$$z_3 = x_1 x_2 x_3$$

$$z_4 \leq \frac{x_1 x_2 - x_1}{y_1 y_2}$$

$$0 \leq x_1, x_2, x_3 \leq 1$$

$$0 < y^L \leq y_1, y_2 \leq y^U$$

can be convexified by disjunctive programming restricting x to binary values. The disjunctive sets that need to be considered correspond to the following points/sets in x-space: $(1, 1, 1)$, $(1, 1, 0)$, $(1, 0, x_3)$ and $(0, x_2, x_3)$. The convex multipliers for the first two sets can be identified with z_3, $z_2 - z_3$ respectively.

General (non-unit) hypercubes can be handled similarly

Example Contd.

The convex hull is the convex hull of the following disjunction:

(I)

$$z_1 \geq \frac{1}{y_1}$$

$$z_4 \leq 0$$

$$y^L \leq y_1, y_2 \leq y^U$$

$$(x_1, x_2, x_3, z_2, z_3) = (1, 1, 1, 1, 1)$$

(II)

$$z_1 \geq \frac{1}{y_1}$$

$$z_4 \leq 0$$

$$y^L \leq y_1, y_2 \leq y^U$$

$$(x_1, x_2, x_3, z_2, z_3) = (1, 1, 0, 1, 0)$$

(III)

$$z_1 \geq 0$$

$$z_4 \leq \frac{-1}{y_1 y_2}$$

$$y^L \leq y_1, y_2 \leq y^U$$

$$0 \leq x_3 \leq 1$$

$$(x_1, x_2, z_2, z_3) = (1, 0, 0, 0)$$

(IV)

$$z_1 \geq 0$$

$$z_4 \leq 0$$

$$y^L \leq y_1, y_2 \leq y^U$$

$$0 \leq x_2, x_3 \leq 1$$

$$(x_1, z_2, z_3) = (0, 0, 0, 0)$$

Proof of Equivalence

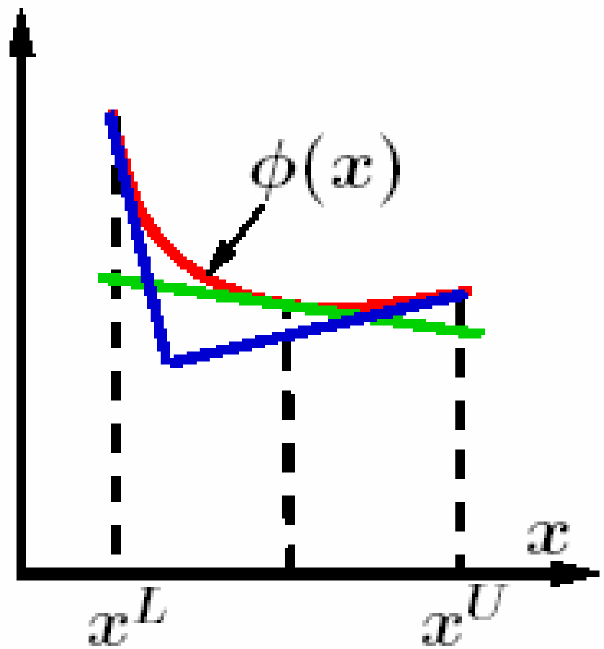
- ($\subseteq$) The convex hull of disjunctive set is a subset of the convex hull of the example constraint set
- ($\supseteq$) The equality follows since the example constraint set is a subset of the set we constructed. Consider $\lambda_I = x_1x_2x_3$, $\lambda_{II} = x_1x_2(1 - x_3)$, $\lambda_{III} = x_1(1 - x_2)$, $\lambda_{IV} = 1 - x_1$.

Some Observations

- Presence of $z_1z_2 \geq \frac{x_1x_2}{y_1}$ does not change the convex hull ...
- Potential use of RELAXATION_ONLY_CONSTRAINTS

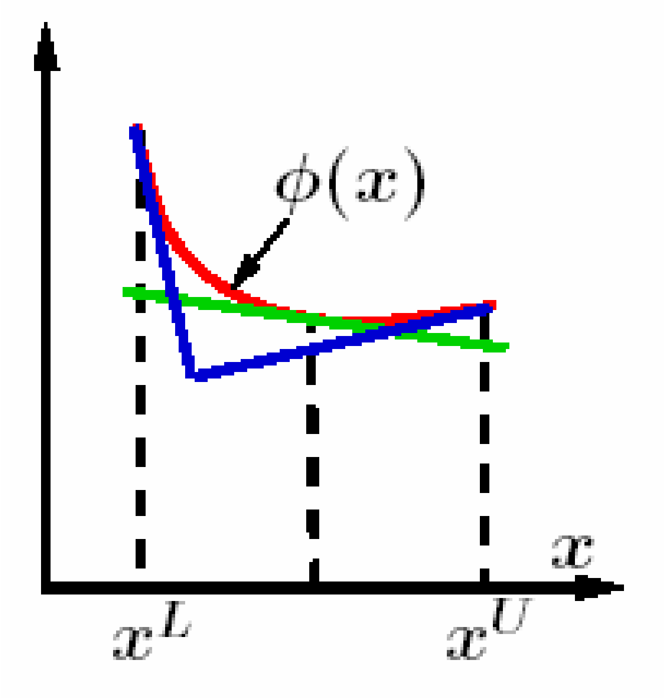
POLYHEDRAL OUTER-APPROXIMATION

- **Convex NLP solvers are not as robust as LP solvers**
- **Linear programs can be solved efficiently**
- **Outer-approximate convex relaxation by polyhedron**



Developed a sandwich algorithm that enjoys quadratic convergence (Tawarmalani and Sahinidis, 2004)

EXPLOITING CONVEXITY

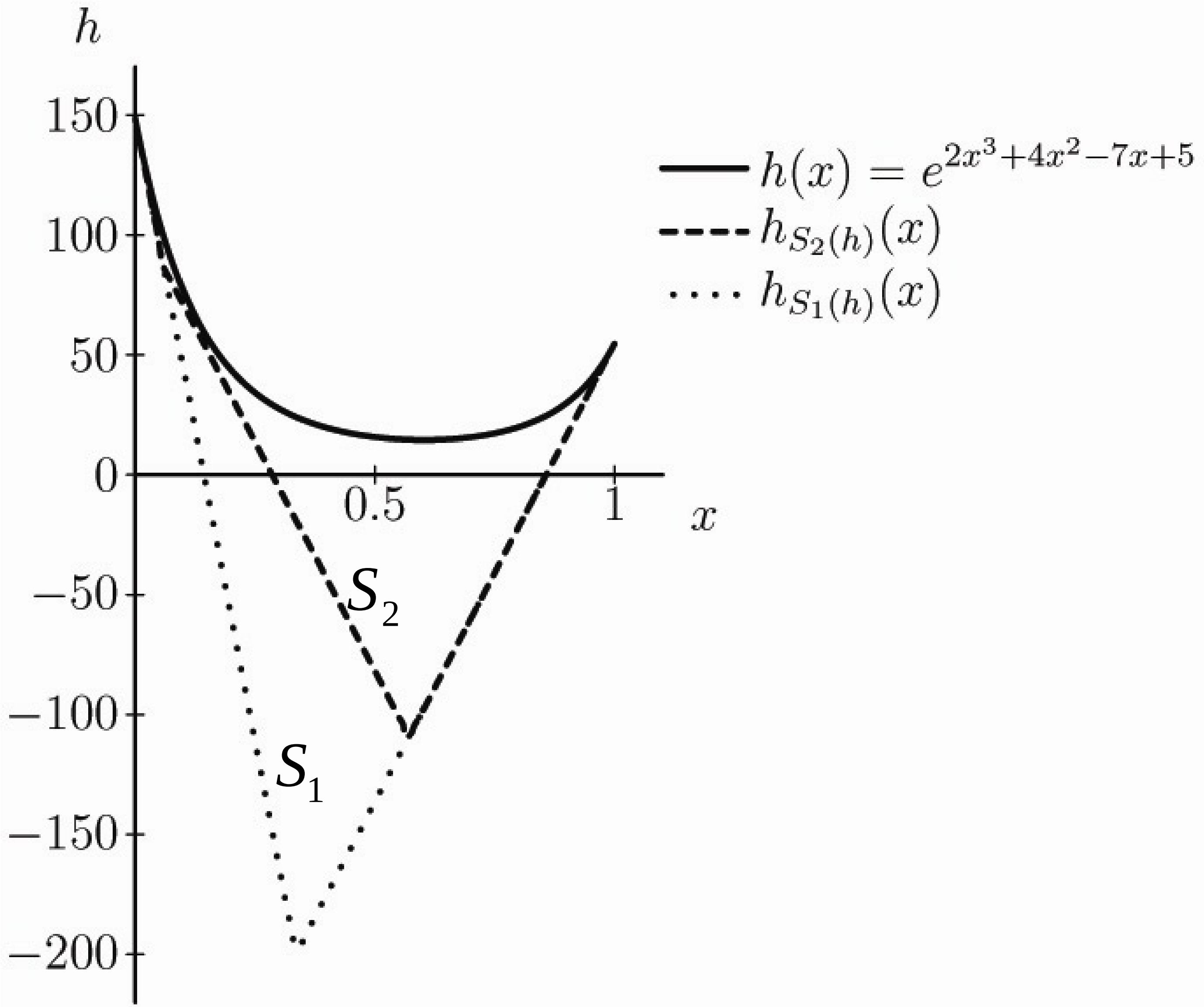


- **Polyhedral relaxations of univariate functions facilitate reliable lower bounding via fast LP routines**
- **Outstanding issues:**
  - **Weakening of lower bound**
  - **Polyhedral relaxations of convex multivariate functions**
    - » **Gruber (1993), Böröczky and Reitzner (2004)**

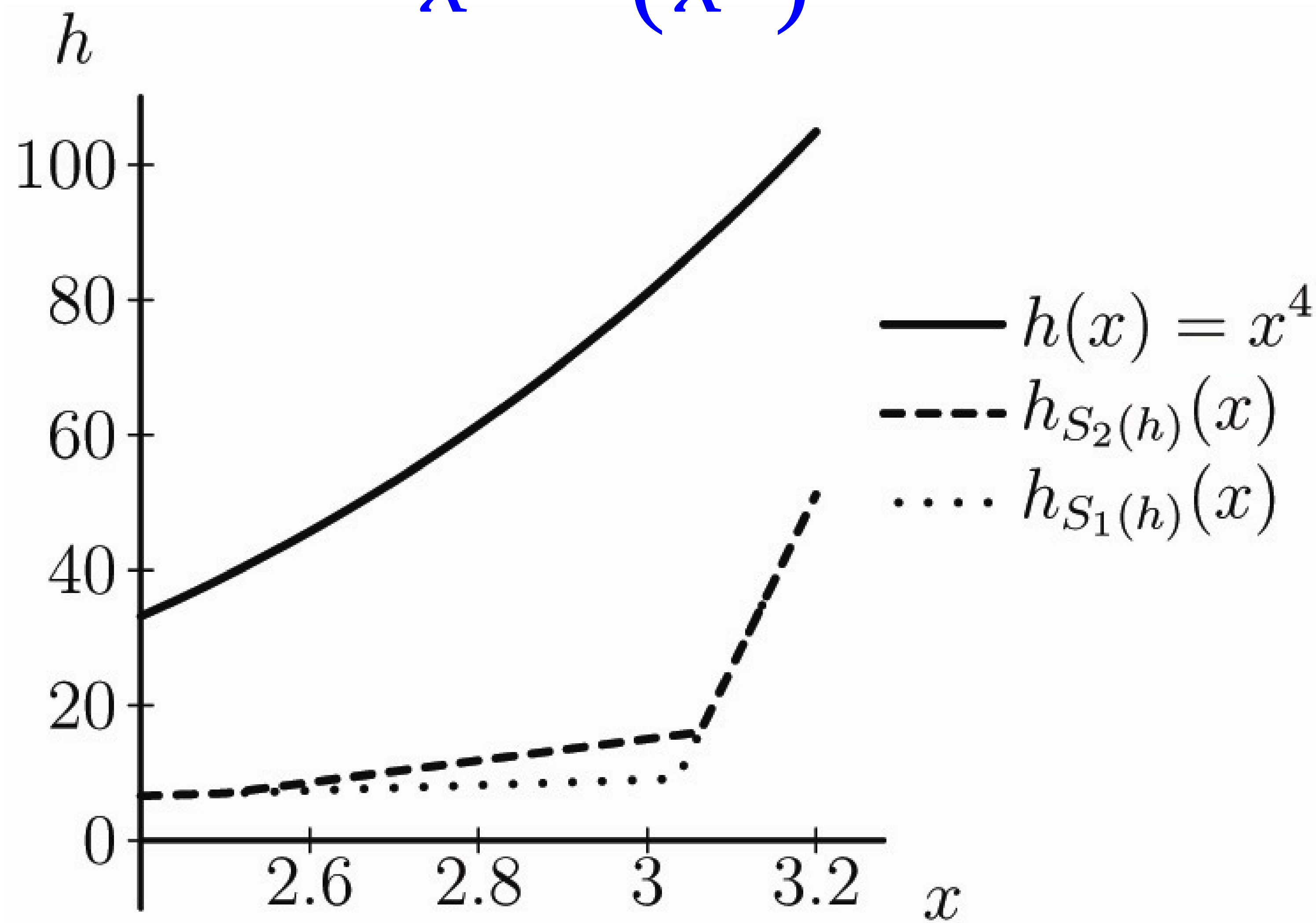
RECURSIVE FUNCTIONAL COMPOSITIONS

- **Consider *h*=*g*(*f*), where**
  - ***g* and *f* are multivariate convex functions**
  - ***g* is non-decreasing in the range of each nonlinear components of *f***
- ***h* is convex**
- **Two outer approximations of the composite function *h*:**
  - **S1: a single-step procedure that constructs supporting hyperplanes of *h* at a predetermined number of points**
  - **S2: a two-step procedure that constructs supporting hyperplanes for *g* and *f* at corresponding points**

THEOREM: S2 is Contained in S1

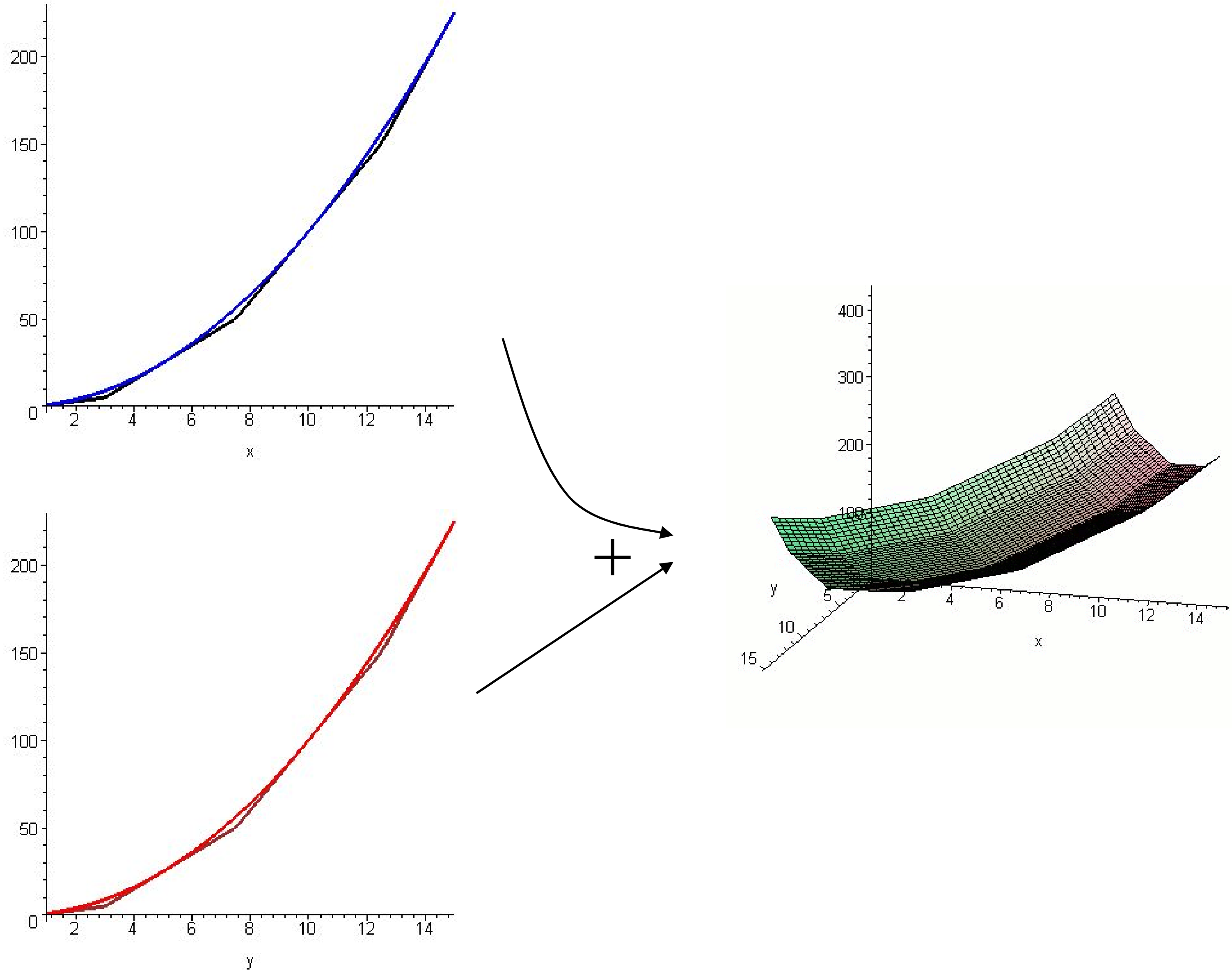


$$x^4 = (x^2)^2$$



Outer - approximating x^4 at $x = 1$ and $x = 4$

OUTER APPROXIMATION OF x^2+y^2



TWO-STEP IS BETTER

- **Theorem:** **An exponential number of supporting hyperplanes in S1 may be required to cover S2**

  $h = f_1(x_1) + \ldots + f_m(x_m)$ **where each** f_i **is strictly convex**

- **Separable functions are quite common in nonconvex optimization**
- **S2 has the potential of providing much tighter polyhedral outer approximations than S1 with a comparable number of supporting hyperplanes**

AUTOMATIC DETECTION AND EXPLOITATION OF CONVEXITY

- **Composition rule: *h* = *g*(*f*), where**
  - ***g* and *f* are multivariate convex functions**
  - ***g* is non-decreasing in the range of each nonlinear component of *f***
- **Subsumes many known rules for detecting convexity/concavity**
  - **However, logexp(*x*) = log(e^{x_1} + … + e^{x_n})**
  - **CONVEX_EQUATIONS modeling language construct**

ILLUSTRATIVE EXAMPLE 1: CUTS FROM CONVEX ENVELOPES

$$\min \quad x^2 - 100x + y^2 - 30y + 1000\frac{x}{y}$$

$$\text{s.t.} \quad 0 \leq x \leq 1000$$

$$1 \leq y \leq 1000$$

Solution -1118 at (34.3, 31.8)

Iteration	Lower bound	Relaxation optimal solution
1	-7500.9	$x_1 = (65.3, 66.5)$
2	-3832.2	$x_2 = (33.1, 34.1)$
3	-2839.5	$x_3 = (49.2, 19.0)$
4	-2325.7	$x_4 = (41.1, 25.6)$
5	-2057.5	$x_5 = (37.1, 22.3)$
6	-2041.1	$x_6 = (39.1, 23.9)$

Cutting planes reduce root-node gap by 86%

With cuts: 7 nodes
Without: 47 nodes

ILLUSTRATIVE EXAMPLE 2: CONVEX_EQUATIONS CONSTRUCT

$$\begin{aligned}\min \quad & 100\log(e^{x_1}+e^{x_2}+e^{x_3})+x_1^2-40x_1\\ & +x_2x_3-10\log(x_1)+x_2^2-20x_2-50x_3\\ \text{s.t.} \quad & 1\leq x_1\leq 10\\ & 1\leq x_2\leq 10\\ & 1\leq x_3\leq 10\end{aligned}$$

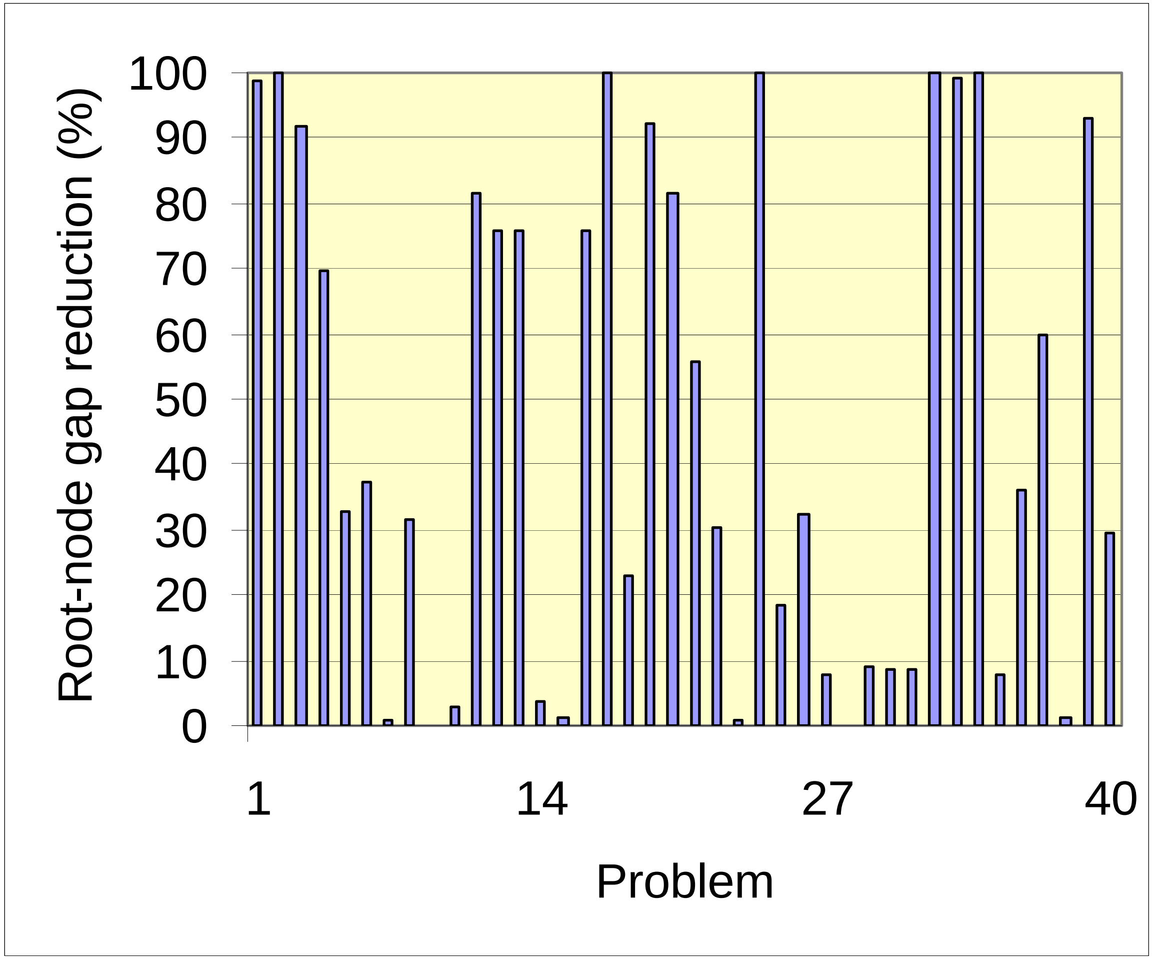
Solution 83
at (34.3, 31.8)

Iteration	Lower bound 1	Lower bound 2
1	−586.9	62.5
2	−514.9	78.9
3	−445.8	79.6
4	−436.2	80.2
5	−432.9	80.6

Cutting planes reduce root-node gap by 99.5%

With cuts: 35 nodes
Without: 1793 nodes

ROOT-NODE GAP REDUCTIONS FOR PROBLEMS FROM `globallib`



Range from 0.05% to 100%. Average 48%.

SOLUTION TO GLOBALITY

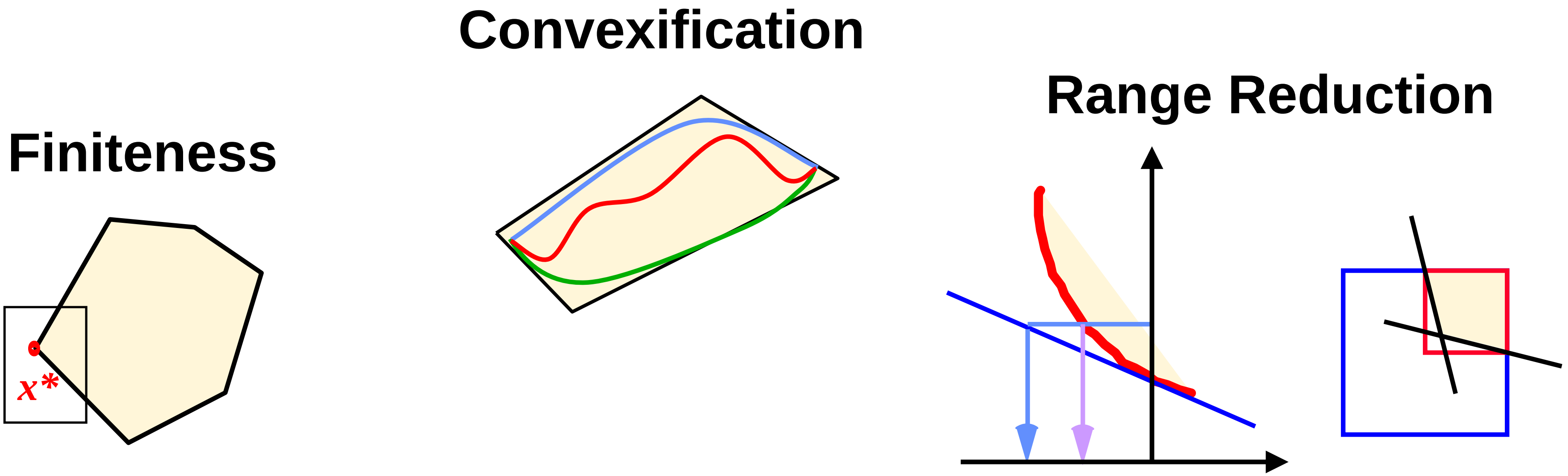
				Without cuts			With cuts		
Problem	m	n	n_d	N_t	N_m	CPU s	N_t	N_m	CPU s
du-opt	9	20	13	82520	44734	25200	79	25	157
du-opt5	9	20	12	106299	33954	25200	78	15	92
elf	38	54	24	866	91	31	698	89	95
enpro48	215	154	92	4215	434	165	702	92	84
enpro56	192	128	73	4609	477	182	1354	99	101
ex1233	64	52	12	28122	652	922	2198	310	1246
ex6_2_14	2	4		739	65	7	2713	115	38
ex8_4_7	40	62		12497	7021	25200	10011	6658	1883
fac1	18	22	6	78055	8842	143	1	1	0
fac2	33	66	12	4653901	101202	25200	27	8	1
fac3	33	66	12	11516667	101172	25200	24	7	0
gtm	24	63		3229450	87622	25200	1	1	0
himmel16	21	18		1211	152	27	915	116	81
linear	20	24		1904473	19706	24403	59472	955	2772
parallel	115	205	25	651	65	198	555	72	194
ravem	186	112	53	778	134	30	246	52	18
spectra2	73	70	30	1963	330	54	43	10	6
synheat	64	56	12	2739	81	191	1159	99	541
tls4	64	105	89	191760	4357	4106	172015	4807	12295
Average	64	68	33	1148501	21636	9561	13278	712	1032

19 PROBLEMS FROM `globallib` AND `minlplib`

	Minimum	Maximum	Average
Constraints	2	215	64
Variables	4	205	68
Discrete variables	0	92	33

	Without cuts	With cuts	% reduction
Nodes	1,148,501	13,278	99
Nodes in memory	21,636	712	97
CPU sec	9,561	1,032	89

BRANCH-AND-REDUCE



- Separable reformulation leads to automatic convexity exploitation and tight relaxations
- Polyhedral cutting planes significantly reduce relaxation gaps, memory and CPU requirements
- Modeling constructs facilitate exploitation of relaxation-only and convex constraints
- Implemented in BARON 7.2