

Computing with multi-row Gomory cuts

Daniel G. Espinoza

Universidad de Chile, FCFM, DII

MIP 2008, Columbia University, New York, USA,
August 4, 2008

Outline

- 1 Introduction and Theory
- 2 The Experiment Settings
- 3 The Results

Introduction

- General cutting planes central for practical IP performance.
- Most important family are Gomory cuts (Bixby et al. 2006).
- Much research on extensions, but little practical impact.
- Most attempts focused on cuts derived from single-row systems.

Introduction

- General cutting planes central for practical IP performance.
- Most important family are Gomory cuts (Bixby et al. 2006).
- Much research on extensions, but little practical impact.
- Most attempts focused on cuts derived from single-row systems.

Introduction

- General cutting planes central for practical IP performance.
- Most important family are Gomory cuts (Bixby et al. 2006).
- Much research on extensions, but little practical impact.
- Most attempts focused on cuts derived from single-row systems.

Introduction

- General cutting planes central for practical IP performance.
- Most important family are Gomory cuts (Bixby et al. 2006).
- Much research on extensions, but little practical impact.
- Most attempts focused on cuts derived from single-row systems.

Introduction

- Recent work by Andersen et. al [2005], Cornuejols and Borozan [2007], Gomory [2007], have extended our understanding of multi-row relaxations.
- From the theory side, multi-row relaxations are known to be central in describing convex hull of sets:
 - Cook, Kannan and Schrijver example has infinite CG-rank.
 - Single cut from two rows is not a Gomory cut relaxation.
 - Theorem: For any $n \times m$ matrix A , there is a $(n+1) \times m$ matrix B such that $\text{CG-rank}(B) > \text{CG-rank}(A)$.
- Our goal is to test if these new ideas may have a practical impact.

Introduction

- Recent work by Andersen et. al [2005], Cornuejols and Borozan [2007], Gomory [2007], have extended our understanding of multi-row relaxations.
- From the theory side, multi-row relaxations are known to be central in describing convex hull of sets:
 - Cook, Kannan and Schrijver example has infinite CG-rank.
 - Single cut from two rows gives complete description.
 - Extensions show need for cuts derived from n rows.
- Our goal is to test if these new ideas may have a practical impact.

Introduction

- Recent work by Andersen et. al [2005], Cornuejols and Borozan [2007], Gomory [2007], have extended our understanding of multi-row relaxations.
- From the theory side, multi-row relaxations are known to be central in describing convex hull of sets:
 - Cook, Kannan and Schrijver example has infinite CG-rank.
 - Single cut from two rows gives complete description.
 - Extensions show need for cuts derived from n rows.
- Our goal is to test if these new ideas may have a practical impact.

Introduction

- Recent work by Andersen et. al [2005], Cornuejols and Borozan [2007], Gomory [2007], have extended our understanding of multi-row relaxations.
- From the theory side, multi-row relaxations are known to be central in describing convex hull of sets:
 - Cook, Kannan and Schrijver example has infinite CG-rank.
 - Single cut from two rows gives complete description.
 - Extensions show need for cuts derived from n rows.
- Our goal is to test if these new ideas may have a practical impact.

Introduction

- Recent work by Andersen et. al [2005], Cornuejols and Borozan [2007], Gomory [2007], have extended our understanding of multi-row relaxations.
- From the theory side, multi-row relaxations are known to be central in describing convex hull of sets:
 - Cook, Kannan and Schrijver example has infinite CG-rank.
 - Single cut from two rows gives complete description.
 - Extensions show need for cuts derived from n rows.
- Our goal is to test if these new ideas may have a practical impact.

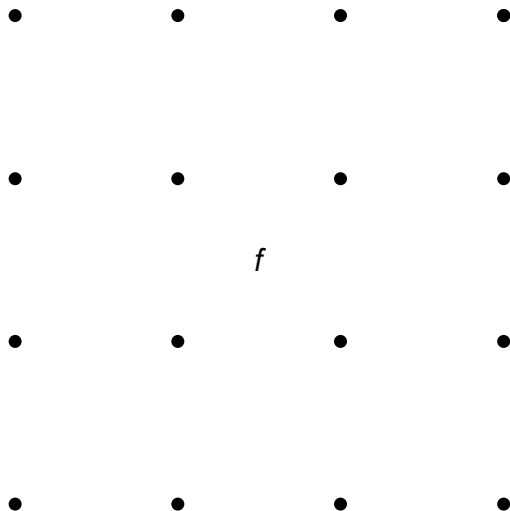
Introduction

- Recent work by Andersen et. al [2005], Cornuejols and Borozan [2007], Gomory [2007], have extended our understanding of multi-row relaxations.
- From the theory side, multi-row relaxations are known to be central in describing convex hull of sets:
 - Cook, Kannan and Schrijver example has infinite CG-rank.
 - Single cut from two rows gives complete description.
 - Extensions show need for cuts derived from n rows.
- Our goal is to test if these new ideas may have a practical impact.

How we derive cuts from two row systems?

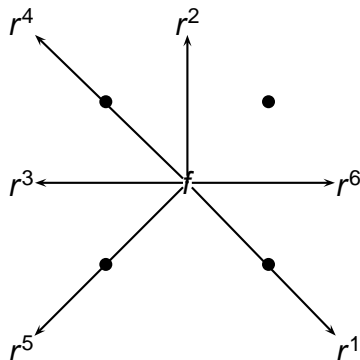
- We consider $x, s \in \mathbb{Z}^2 \times \mathbb{R}_+^J$.

How we derive cuts from two row systems?



- We consider $x, s \in \mathbb{Z}^2 \times \mathbb{R}_+^J$.
- Assume $f, r^j \in \mathbb{Q}^2$, $f \notin \mathbb{Z}^2$.

How we derive cuts from two row systems?

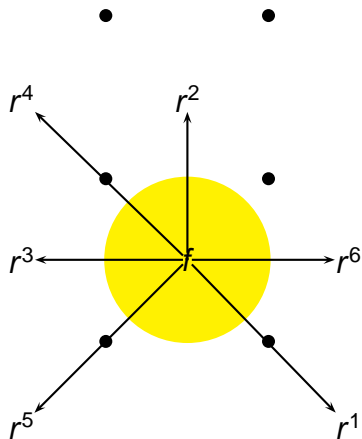


- We consider $x, s \in \mathbb{Z}^2 \times \mathbb{R}_+^J$.
- Assume

$$x = f + \sum_{j \in J} r^j s_j$$

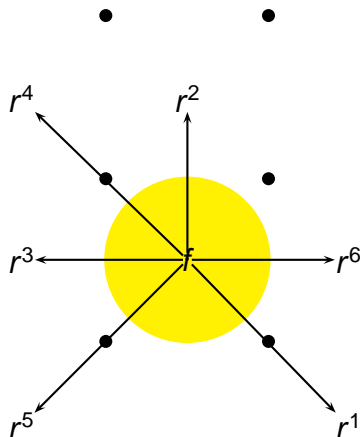
...

How we derive cuts from two row systems?



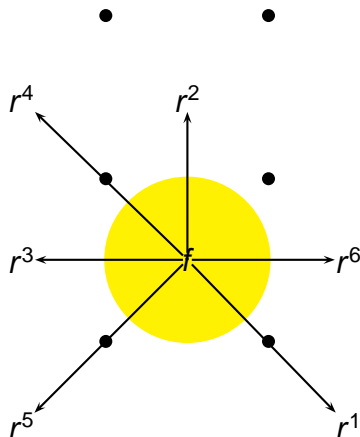
- We consider $x, s \in \mathbb{Z}^2 \times \mathbb{R}_+^J$.
- Let $C \subset \mathbb{R}^2$ s.t. $f \in C$, convex and $\text{int}(C) \cap \mathbb{Z}^2 = \emptyset$.

How we derive cuts from two row systems?



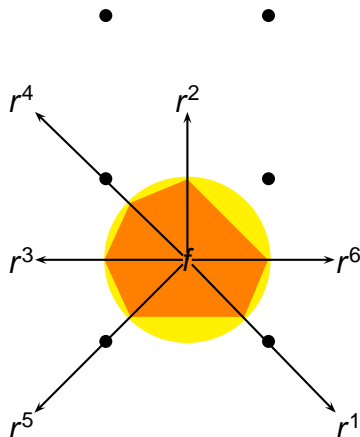
- We consider $x, s \in \mathbb{Z}^2 \times \mathbb{R}_+^J$.
- Let $C \subset \mathbb{R}^2$ s.t. $f \in C$, convex and $\text{int}(C) \cap \mathbb{Z}^2 = \emptyset$
- Let $\alpha_j = \max\{\alpha : f + \alpha r^j \in C\}$.

How we derive cuts from two row systems?



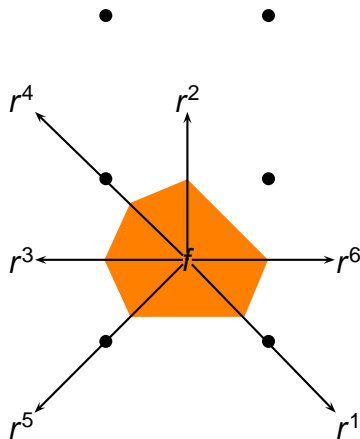
- We consider $x, s \in \mathbb{Z}^2 \times \mathbb{R}_+^J$.
- Let $C \subset \mathbb{R}^2$ s.t. $f \in C$, convex and $\text{int}(C) \cap \mathbb{Z}^2 = \emptyset$
- Let $\alpha_j = \max\{\alpha : f + \alpha r^j \in C\}$
- Let $S = \{s \in \mathbb{R}_+^J : \sum_{j \in J} \frac{s_j}{\alpha_j} \leq 1\}$.

How we derive cuts from two row systems?



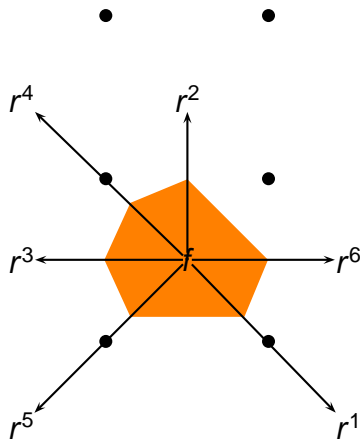
- We consider $x, s \in \mathbb{Z}^2 \times \mathbb{R}_+^J$.
- Let $C \subset \mathbb{R}^2$ s.t. $f \in C$, convex and $\text{int}(C) \cap \mathbb{Z}^2 = \emptyset$
- Let $\alpha_j = \max\{\alpha : f + \alpha r^j \in C\}$
- Let $S = \{s \in \mathbb{R}_+^J : \sum_{j \in J} \frac{s_j}{\alpha_j} \leq 1\}$.
- $\Rightarrow C' := f + r \cdot S \subseteq C$.

How we derive cuts from two row systems?



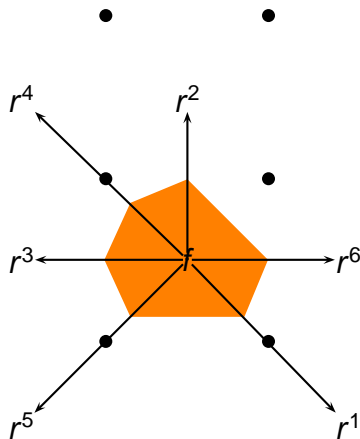
- We consider $x, s \in \mathbb{Z}^2 \times \mathbb{R}_+^J$.
- Let $C \subset \mathbb{R}^2$ s.t. $f \in C$, convex and $\text{int}(C) \cap \mathbb{Z}^2 = \emptyset$
- Let $\alpha_j = \max\{\alpha : f + \alpha r^j \in C\}$
- Let $S = \{s \in \mathbb{R}_+^J : \sum_{j \in J} \frac{s_j}{\alpha_j} \leq 1\}$.
- $\sum_{j \in J} \frac{s_j}{\alpha_j} \geq 1$ is valid

How we derive cuts from two row systems?



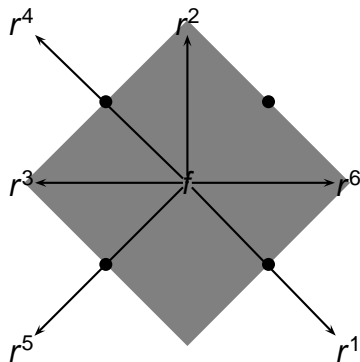
- We consider $x, s \in \mathbb{Z}^2 \times \mathbb{R}_+^J$.
- $\sum_{j \in J} \frac{s_j}{\alpha_j} \geq 1$ is valid
- How much better can we make C' ?

How we derive cuts from two row systems?



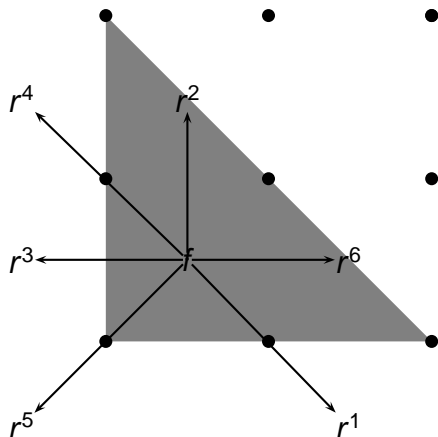
- We consider $x, s \in \mathbb{Z}^2 \times \mathbb{R}_+^J$.
- $\sum_{j \in J} \frac{s_j}{\alpha_j} \geq 1$ is valid
- C' should be a maximal convex, lattice free set containing f

How we derive cuts from two row systems?



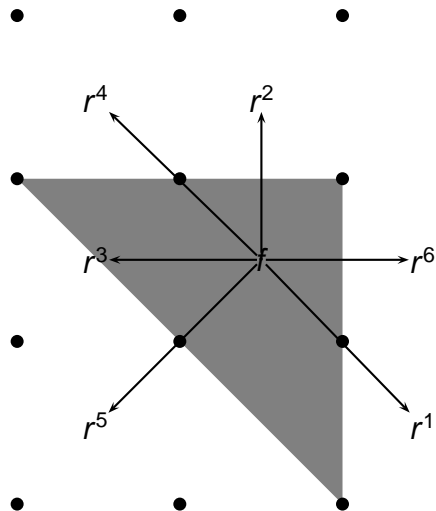
- We consider $x, s \in \mathbb{Z}^2 \times \mathbb{R}_+^J$.
- $\sum_{j \in J} \frac{s_j}{\alpha_j} \geq 1$ is valid
- C' should be a maximal convex, lattice free set containing f
- A quadrilateral or Gomory set
- completely symmetric

How we derive cuts from two row systems?



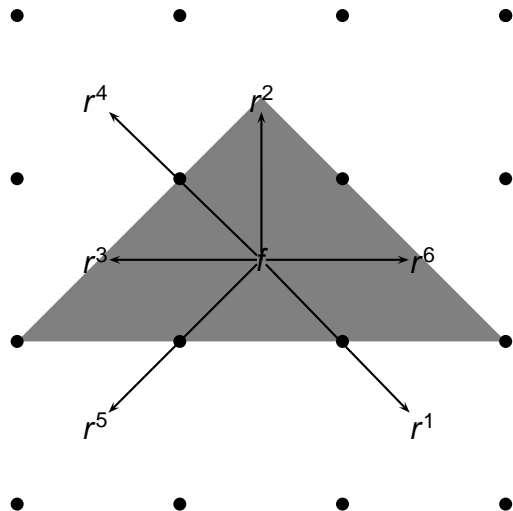
- We consider $x, s \in \mathbb{Z}^2 \times \mathbb{R}_+^J$.
- $\sum_{j \in J} \frac{s_j}{\alpha_j} \geq 1$ is valid
- C' should be a maximal convex, lattice free set containing f
- A Type 1 triangle

How we derive cuts from two row systems?



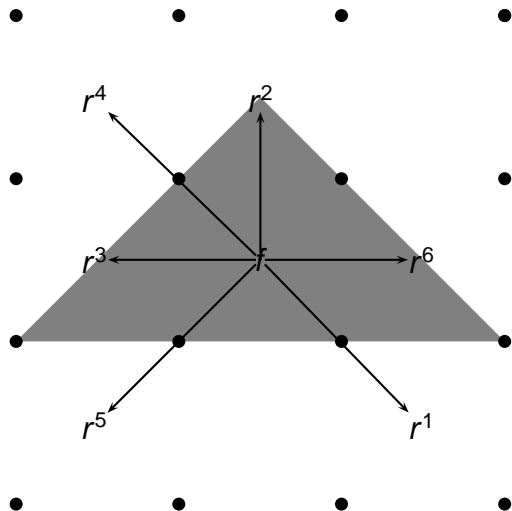
- We consider $x, s \in \mathbb{Z}^2 \times \mathbb{R}_+^J$.
- $\sum_{j \in J} \frac{s_j}{\alpha_j} \geq 1$ is valid
- C' should be a maximal convex, lattice free set containing f
- A Type 1 triangle
- 2^n possible configurations ($n =$ number of rows)

How we derive cuts from two row systems?



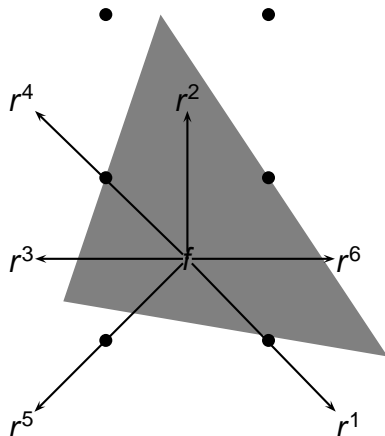
- We consider $x, s \in \mathbb{Z}^2 \times \mathbb{R}_+^J$.
- $\sum_{j \in J} \frac{s_j}{\alpha_j} \geq 1$ is valid
- C' should be a maximal convex, lattice free set containing f
- A Type 2 triangle
- $n!2^n$ possible configurations

How we derive cuts from two row systems?



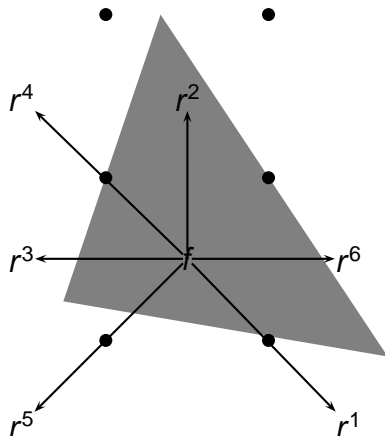
- We consider $x, s \in \mathbb{Z}^2 \times \mathbb{R}_+^J$.
- $\sum_{j \in J} \frac{s_j}{\alpha_j} \geq 1$ is valid
- For non-dominance, every edge must contain integer point in relative interior

How we derive cuts from two row systems?



- We consider $x, s \in \mathbb{Z}^2 \times \mathbb{R}_+^J$.
- $\sum_{j \in J} \frac{s_j}{\alpha_j} \geq 1$ is valid
- Still, far too many possible sets

How we derive cuts from two row systems?



- We consider $x, s \in \mathbb{Z}^2 \times \mathbb{R}_+^J$.
- $\sum_{j \in J} \frac{s_j}{\alpha_j} \geq 1$ is valid
- Still, far too many possible sets
- All these ideas can be extended to $x, f, r^j \in \mathbb{Q}^q$ and to $|J| = \infty$.

The Problem:

How we apply this?

Basic Problem:

$$\begin{aligned}
 \max \quad & cx \\
 \text{s.t.} \quad & Ax = b \\
 & x_i \in \mathbb{Z}_+ \quad \forall i \in I, \mathbf{x} \in \mathbb{R}_+^n
 \end{aligned} \tag{1}$$

where $I \subseteq \{1, \dots, n\}$, $A \in \mathbb{Q}^{m \times n}$ is of full row rank, $c \in \mathbb{Q}^n$, $b \in \mathbb{Q}^m$, and $x \in \mathbb{Q}^n$.

A first relaxation:

$$\begin{aligned}
 x_{B'} &= f + \sum_{j \in N} r^j x_j \\
 x_N &\geq 0, x_i \in \mathbb{Z} \forall i \in B'
 \end{aligned} \tag{2}$$

Where B is a basic solution, $B' = B \cap I$.

The Problem:

How we apply this?

Basic Problem:

$$\begin{aligned}
 \max \quad & cx \\
 \text{s.t.} \quad & Ax = b \\
 & x_i \in \mathbb{Z}_+ \quad \forall i \in I, \mathbf{x} \in \mathbb{R}_+^n
 \end{aligned} \tag{1}$$

where $I \subseteq \{1, \dots, n\}$, $A \in \mathbb{Q}^{m \times n}$ is of full row rank, $c \in \mathbb{Q}^n$, $b \in \mathbb{Q}^m$, and $\mathbf{x} \in \mathbb{Q}^n$.

A first relaxation:

$$\begin{aligned}
 \mathbf{x}_{B'} &= \mathbf{f} + \sum_{j \in N} r^j \mathbf{x}_j \\
 \mathbf{x}_N &\geq \mathbf{0}, \mathbf{x}_i \in \mathbb{Z} \forall i \in B'
 \end{aligned} \tag{2}$$

Where B is a basic solution, $B' = B \cap I$.

The Problem:

The math behind it

Gomory-Johnson Infinite group relaxation:

$$R_f : \quad \begin{array}{l} \mathbf{x} = \mathbf{f} + \sum_{\text{finite}} r s_r \\ \mathbf{x} \in \mathbb{Z}^q \quad \quad \mathbf{s} \in \mathbb{R}_+^{\mathbb{Q}^q} \end{array} \quad (3)$$

Theorem (Minimal valid inequalities for R_f (CB-2007))

If $\mathbf{f} \notin \mathbb{Z}^q$, any minimal valid inequality that cuts off $(\mathbf{f}, 0)$:

● Is of the form $\sum_{r \in \mathbb{Q}^q} \psi(r) s_r \geq 1$.

The math behind it

Gomory-Johnson Infinite group relaxation:

$$R_f : \quad \begin{array}{l} \mathbf{x} = f + \sum_{\text{finite}} r s_r \\ \mathbf{x} \in \mathbb{Z}^q \quad \mathbf{s} \in \mathbb{R}_+^{\mathbb{Q}^q} \end{array} \quad (3)$$

Theorem (Minimal valid inequalities for R_f (CB-2007))

If $f \notin \mathbb{Z}^q$, any minimal valid inequality that cuts off $(f, 0)$:

- 1 Is of the form $\sum_{\text{finite}} \psi(r) s_r \geq 1$.
- 2 If $B_\psi = \{\mathbf{x} \in \mathbb{Q}^p : \psi(\mathbf{x} - f) \leq 1\}$, then B_ψ is convex, with no integral point in its interior. Furthermore $f \in B_\psi$.
- 3 If ψ is finite, then ψ is a continuous nonnegative homogeneous convex piecewise linear function with at most 2^q pieces.
- 4 If ψ is finite, then f is in the interior of B_ψ and B_ψ is a polyhedron of at most 2^q facets, and each of its facets contains an integral point in its relative interior.

The math behind it

Gomory-Johnson Infinite group relaxation:

$$R_f : \quad \begin{array}{l} \mathbf{x} = f + \sum_{\text{finite}} r \mathbf{s}_r \\ \mathbf{x} \in \mathbb{Z}^q \quad \mathbf{s} \in \mathbb{R}_+^{\mathbb{Q}^q} \end{array} \quad (3)$$

Theorem (Minimal valid inequalities for R_f (CB-2007))

If $f \notin \mathbb{Z}^q$, any minimal valid inequality that cuts off $(f, 0)$:

- 1 Is of the form $\sum_{\text{finite}} \psi(r) \mathbf{s}_r \geq 1$.
- 2 If $B_\psi = \{\mathbf{x} \in \mathbb{Q}^p : \psi(\mathbf{x} - f) \leq 1\}$, then B_ψ is convex, with no integral point in its interior. Furthermore $f \in B_\psi$.
- 3 If ψ is finite, then ψ is a continuous nonnegative homogeneous convex piecewise linear function with at most 2^q pieces.
- 4 If ψ is finite, then f is in the interior of B_ψ and B_ψ is a polyhedron of at most 2^q facets, and each of its facets contains an integral point in its relative interior.

The math behind it

Gomory-Johnson Infinite group relaxation:

$$R_f : \quad \begin{array}{l} \mathbf{x} = f + \sum_{\text{finite}} r s_r \\ \mathbf{x} \in \mathbb{Z}^q \quad \quad \mathbf{s} \in \mathbb{R}_+^{\mathbb{Q}^q} \end{array} \quad (3)$$

Theorem (Minimal valid inequalities for R_f (CB-2007))

If $f \notin \mathbb{Z}^q$, any minimal valid inequality that cuts off $(f, 0)$:

- 1 Is of the form $\sum_{\text{finite}} \psi(r) s_r \geq 1$.
- 2 If $B_\psi = \{\mathbf{x} \in \mathbb{Q}^p : \psi(\mathbf{x} - f) \leq 1\}$, then B_ψ is convex, with no integral point in its interior. Furthermore $f \in B_\psi$.
- 3 If ψ is finite, then ψ is a continuous nonnegative homogeneous convex piecewise linear function with at most 2^q pieces.
- 4 If ψ is finite, then f is in the interior of B_ψ and B_ψ is a polyhedron of at most 2^q facets, and each of its facets contains an integral point in its relative interior.

The math behind it

Gomory-Johnson Infinite group relaxation:

$$R_f : \quad \begin{array}{l} \mathbf{x} = f + \sum_{\text{finite}} r s_r \\ \mathbf{x} \in \mathbb{Z}^q \quad \mathbf{s} \in \mathbb{R}_+^{\mathbb{Q}^q} \end{array} \quad (3)$$

Theorem (Minimal valid inequalities for R_f (CB-2007))

If $f \notin \mathbb{Z}^q$, any minimal valid inequality that cuts off $(f, 0)$:

- 1 Is of the form $\sum_{\text{finite}} \psi(r) s_r \geq 1$.
- 2 If $B_\psi = \{\mathbf{x} \in \mathbb{Q}^p : \psi(\mathbf{x} - f) \leq 1\}$, then B_ψ is convex, with no integral point in its interior. Furthermore $f \in B_\psi$.
- 3 If ψ is finite, then ψ is a continuous nonnegative homogeneous convex piecewise linear function with at most 2^q pieces.
- 4 If ψ is finite, then f is in the interior of B_ψ and B_ψ is a polyhedron of at most 2^q facets, and each of its facets contains an integral point in its relative interior.

The math behind it

Gomory-Johnson Infinite group relaxation:

$$R_f : \quad \begin{array}{l} \mathbf{x} = f + \sum_{\text{finite}} r s_r \\ \mathbf{x} \in \mathbb{Z}^q \quad \mathbf{s} \in \mathbb{R}_+^{\mathbb{Q}^q} \end{array} \quad (3)$$

Theorem (Minimal valid inequalities for R_f (CB-2007))

If $f \notin \mathbb{Z}^q$, any minimal valid inequality that cuts off $(f, 0)$:

- 1 Is of the form $\sum_{\text{finite}} \psi(r) s_r \geq 1$.
- 2 If $B_\psi = \{\mathbf{x} \in \mathbb{Q}^p : \psi(\mathbf{x} - f) \leq 1\}$, then B_ψ is convex, with no integral point in its interior. Furthermore $f \in B_\psi$.
- 3 If ψ is finite, then ψ is a continuous nonnegative homogeneous convex piecewise linear function with at most 2^q pieces.
- 4 If ψ is finite, then f is in the interior of B_ψ and B_ψ is a polyhedron of at most 2^q facets, and each of its facets contains an integral point in its relative interior.

Selecting a set B_ψ

- Three kind of maximal convex lattice free sets:

- Type 1 n -dimensional simplex:

$$T1_n = \{x \in \mathbb{R}^n : x_i \geq 0, \sum_{i=1}^n x_i \leq n\}$$

• Each face has exactly one more lattice point than the previous face.
 • 2^n possible orientations.

- The n -dimensional hyper cube:

- Type-2 n -dimensional simplex:

- $\text{vol}(T1_n) = \text{vol}(G_n) = \text{vol}(T2_n) \Theta\left(2^{-\frac{n(n-1)}{2}}\right)$.

Selecting a set B_ψ

- Three kind of maximal convex lattice free sets:
 - Type 1 n -dimensional simplex:
 - $T1_n := \{x \in \mathbb{R}^n : x_i \geq 0, \sum_{i=1}^n x_i \leq n\}$
 - Each facet has exactly one point in its relative interior.
 - 2^n possible *orientations*.
 - The n -dimensional hyper cube:
- Type-2 n -dimensional simplex:

- $\text{vol}(T1_n) = \text{vol}(G_n) = \text{vol}(T2_n) \Theta\left(2^{-\frac{n(n-1)}{2}}\right).$

Selecting a set B_ψ

- Three kind of maximal convex lattice free sets:
 - Type 1 n -dimensional simplex:
 - $T1_n := \{x \in \mathbb{R}^n : x_i \geq 0, \sum_{i=1}^n x_i \leq n\}$
 - Each facet has exactly one point in its relative interior.
 - 2^n possible *orientations*.
 - The n -dimensional hyper cube:
- Type-2 n -dimensional simplex:

- $\text{vol}(T1_n) = \text{vol}(G_n) = \text{vol}(T2_n) \Theta\left(2^{-\frac{n(n-1)}{2}}\right).$

Selecting a set B_ψ

- Three kind of maximal convex lattice free sets:
 - Type 1 n -dimensional simplex:
 - $T1_n := \{x \in \mathbb{R}^n : x_i \geq 0, \sum_{i=1}^n x_i \leq n\}$
 - Each facet has exactly one point in its relative interior.
 - 2^n possible *orientations*.
 - The n -dimensional hyper cube:

- Type-2 n -dimensional simplex:

- $\text{vol}(T1_n) = \text{vol}(G_n) = \text{vol}(T2_n) \Theta\left(2^{-\frac{n(n-1)}{2}}\right).$

Selecting a set B_ψ

- Three kind of maximal convex lattice free sets:
 - Type 1 n -dimensional simplex:
 - $T1_n := \{x \in \mathbb{R}^n : x_i \geq 0, \sum_{i=1}^n x_i \leq n\}$
 - Each facet has exactly one point in its relative interior.
 - 2^n possible *orientations*.
 - The n -dimensional hyper cube:
 - $G_n = \sum_{i=1}^n x_i e_i + \sum_{i=1}^n y_i e_{-i}, x_i, y_i \in [-1, 1]$
 - Type-2 n -dimensional simplex:

- $\text{vol}(T1_n) = \text{vol}(G_n) = \text{vol}(T2_n) \Theta(2^{-\frac{n(n-1)}{2}})$.

Selecting a set B_ψ

- Three kind of maximal convex lattice free sets:
 - Type 1 n -dimensional simplex:
 - $T1_n := \{x \in \mathbb{R}^n : x_i \geq 0, \sum_{i=1}^n x_i \leq n\}$
 - Each facet has exactly one point in its relative interior.
 - 2^n possible *orientations*.
 - The n -dimensional hyper cube:
 - $G_n := \frac{1}{2} + \{x \in \mathbb{R}^n : \sum_{i=1}^n s_i x_i \leq \frac{n}{2}, \forall s \in \{-1, 1\}^n\}$
 - Each facet has exactly one point in its relative interior.
 - Completely symmetric.
 - Type-2 n -dimensional simplex:

- $\text{vol}(T1_n) = \text{vol}(G_n) = \text{vol}(T2_n) \Theta\left(2^{-\frac{n(n-1)}{2}}\right).$

Selecting a set B_ψ

- Three kind of maximal convex lattice free sets:
 - Type 1 n -dimensional simplex:
 - $T1_n := \{x \in \mathbb{R}^n : x_i \geq 0, \sum_{i=1}^n x_i \leq n\}$
 - Each facet has exactly one point in its relative interior.
 - 2^n possible *orientations*.
 - The n -dimensional hyper cube:
 - $G_n := e_{\frac{1}{2}} + \{x \in \mathbb{R}^n : \sum_{i=1}^n s_i x_i \leq \frac{n}{2}, \forall s \in \{-1, 1\}^n\}$
 - Each facet has exactly one point in its relative interior.
 - Completely symmetric.
 - Type-2 n -dimensional simplex:

- $\text{vol}(T1_n) = \text{vol}(G_n) = \text{vol}(T2_n) \Theta\left(2^{-\frac{n(n-1)}{2}}\right).$

Selecting a set B_ψ

- Three kind of maximal convex lattice free sets:
 - Type 1 n -dimensional simplex:
 - $T1_n := \{x \in \mathbb{R}^n : x_i \geq 0, \sum_{i=1}^n x_i \leq n\}$
 - Each facet has exactly one point in its relative interior.
 - 2^n possible *orientations*.
 - The n -dimensional hyper cube:
 - $G_n := e_{\frac{1}{2}} + \{x \in \mathbb{R}^n : \sum_{i=1}^n s_i x_i \leq \frac{n}{2}, \forall s \in \{-1, 1\}^n\}$
 - Each facet has exactly one point in its relative interior.
 - Completely symmetric.
 - Type-2 n -dimensional simplex:

$$\bullet \text{vol}(T1_n) = \text{vol}(G_n) = \text{vol}(T2_n) \Theta\left(2^{-\frac{n(n-1)}{2}}\right).$$

Selecting a set B_ψ

- Three kind of maximal convex lattice free sets:
 - Type 1 n -dimensional simplex:
 - $T1_n := \{x \in \mathbb{R}^n : x_i \geq 0, \sum_{i=1}^n x_i \leq n\}$
 - Each facet has exactly one point in its relative interior.
 - 2^n possible *orientations*.
 - The n -dimensional hyper cube:
 - $G_n := e_2^1 + \{x \in \mathbb{R}^n : \sum_{i=1}^n s_i x_i \leq \frac{n}{2}, \forall s \in \{-1, 1\}^n\}$
 - Each facet has exactly one point in its relative interior.
 - Completely symmetric.

- Type-2 n -dimensional simplex:

$$T_2 := \{x \in \mathbb{R}^n : \sum_{i=1}^n x_i \leq n, x_i \in \{-1, 0, 1\}\}$$

- $\text{vol}(T1_n) = \text{vol}(G_n) = \text{vol}(T2_n) \Theta(2^{-\frac{n(n-1)}{2}})$.

Selecting a set B_ψ

- Three kind of maximal convex lattice free sets:
 - Type 1 n -dimensional simplex:
 - $T1_n := \{x \in \mathbb{R}^n : x_i \geq 0, \sum_{i=1}^n x_i \leq n\}$
 - Each facet has exactly one point in its relative interior.
 - 2^n possible *orientations*.
 - The n -dimensional hyper cube:
 - $G_n := e^1_2 + \{x \in \mathbb{R}^n : \sum_{i=1}^n s_i x_i \leq \frac{n}{2}, \forall s \in \{-1, 1\}^n\}$
 - Each facet has exactly one point in its relative interior.
 - Completely symmetric.
 - Type-2 n -dimensional simplex:
 - $T2_n := \{x \in \mathbb{R}^n : (R_i) \sum_{j=1}^{i-1} x_j - x_i \leq i-1, \forall i \in N+1\}$
 - Facet (R_i) has $\max\{2^{n-i}, 1\}$ (interior) integer points.
 - Has $2^n n!$ possible *orientations*
- $\text{vol}(T1_n) = \text{vol}(G_n) = \text{vol}(T2_n) \Theta(2^{-\frac{n(n-1)}{2}})$.

Selecting a set B_ψ

- Three kind of maximal convex lattice free sets:
 - Type 1 n -dimensional simplex:
 - $T1_n := \{x \in \mathbb{R}^n : x_i \geq 0, \sum_{i=1}^n x_i \leq n\}$
 - Each facet has exactly one point in its relative interior.
 - 2^n possible *orientations*.
 - The n -dimensional hyper cube:
 - $G_n := e^1_2 + \{x \in \mathbb{R}^n : \sum_{i=1}^n s_i x_i \leq \frac{n}{2}, \forall s \in \{-1, 1\}^n\}$
 - Each facet has exactly one point in its relative interior.
 - Completely symmetric.
 - Type-2 n -dimensional simplex:
 - $T2_n := \{x \in \mathbb{R}^n : (R_i) \sum_{j=1}^{i-1} x_j - x_i \leq i-1, \forall i \in N+1\}$
 - Facet (R_i) has $\max\{2^{n-i}, 1\}$ (interior) integer points.
 - Has $2^n n!$ possible *orientations*
- $\text{vol}(T1_n) = \text{vol}(G_n) = \text{vol}(T2_n) \Theta(2^{-\frac{n(n-1)}{2}})$.

Selecting a set B_ψ

- Three kind of maximal convex lattice free sets:
 - Type 1 n -dimensional simplex:
 - $T1_n := \{x \in \mathbb{R}^n : x_i \geq 0, \sum_{i=1}^n x_i \leq n\}$
 - Each facet has exactly one point in its relative interior.
 - 2^n possible *orientations*.
 - The n -dimensional hyper cube:
 - $G_n := e^1_2 + \{x \in \mathbb{R}^n : \sum_{i=1}^n s_i x_i \leq \frac{n}{2}, \forall s \in \{-1, 1\}^n\}$
 - Each facet has exactly one point in its relative interior.
 - Completely symmetric.
 - Type-2 n -dimensional simplex:
 - $T2_n := \{x \in \mathbb{R}^n : (R_i) \sum_{j=1}^{i-1} x_j - x_i \leq i-1, \forall i \in N+1\}$
 - Facet (R_i) has $\max\{2^{n-i}, 1\}$ (interior) integer points.
 - Has $2^n n!$ possible *orientations*
- $\text{vol}(T1_n) = \text{vol}(G_n) = \text{vol}(T2_n) \Theta(2^{-\frac{n(n-1)}{2}})$.

Selecting a set B_ψ

- Three kind of maximal convex lattice free sets:
 - Type 1 n -dimensional simplex:
 - $T1_n := \{x \in \mathbb{R}^n : x_i \geq 0, \sum_{i=1}^n x_i \leq n\}$
 - Each facet has exactly one point in its relative interior.
 - 2^n possible *orientations*.
 - The n -dimensional hyper cube:
 - $G_n := e^1_2 + \{x \in \mathbb{R}^n : \sum_{i=1}^n s_i x_i \leq \frac{n}{2}, \forall s \in \{-1, 1\}^n\}$
 - Each facet has exactly one point in its relative interior.
 - Completely symmetric.
 - Type-2 n -dimensional simplex:
 - $T2_n := \{x \in \mathbb{R}^n : (R_i) \sum_{j=1}^{i-1} x_j - x_i \leq i-1, \forall i \in N+1\}$
 - Facet (R_i) has $\max\{2^{n-i}, 1\}$ (interior) integer points.
 - Has $2^n n!$ possible *orientations*
- $\text{vol}(T1_n) = \text{vol}(G_n) = \text{vol}(T2_n) \Theta(2^{-\frac{n(n-1)}{2}})$.

Selecting a set B_ψ

- Three kind of maximal convex lattice free sets:
 - Type 1 n -dimensional simplex:
 - $T1_n := \{x \in \mathbb{R}^n : x_i \geq 0, \sum_{i=1}^n x_i \leq n\}$
 - Each facet has exactly one point in its relative interior.
 - 2^n possible *orientations*.
 - The n -dimensional hyper cube:
 - $G_n := e^1_2 + \{x \in \mathbb{R}^n : \sum_{i=1}^n s_i x_i \leq \frac{n}{2}, \forall s \in \{-1, 1\}^n\}$
 - Each facet has exactly one point in its relative interior.
 - Completely symmetric.
 - Type-2 n -dimensional simplex:
 - $T2_n := \{x \in \mathbb{R}^n : (R_i) \sum_{j=1}^{i-1} x_j - x_i \leq i-1, \forall i \in N+1\}$
 - Facet (R_i) has $\max\{2^{n-i}, 1\}$ (interior) integer points.
 - Has $2^n n!$ possible *orientations*
- $\text{vol}(T1_n) = \text{vol}(G_n) = \text{vol}(T2_n) \Theta(2^{-\frac{n(n-1)}{2}})$.

How we separate:

How we separate:

- Relaxation selection:

How we separate:

How we separate:

- Relaxation selection:
 - Sort fractional integer variables by pseudo cost.

How we separate:

How we separate:

- Relaxation selection:
 - Sort fractional integer variables by pseudo cost.
 - Discard numerically ill tableau rows ($|a| \leq 10^{-6} \Rightarrow a = 0$).

How we separate:

How we separate:

- Relaxation selection:
 - Sort fractional integer variables by pseudo cost.
 - Discard numerically ill tableau rows ($|a| \leq 10^{-6} \Rightarrow a = 0$).
 - Store tableau rows (in the previous order) that have a ratio $\leq 2^{12}$.

How we separate:

How we separate:

- Relaxation selection:
 - Sort fractional integer variables by pseudo cost.
 - Discard numerically ill tableau rows
($|a| \leq 10^{-6} \Rightarrow a = 0$).
 - Store tableau rows (in the previous order) that have a ratio $\leq 2^{12}$.
 - Up to n tableau rows are found, or no more available.

How we separate:

How we separate:

- Relaxation selection:
 - Sort fractional integer variables by pseudo cost.
 - Discard numerically ill tableau rows
($|a| \leq 10^{-6} \Rightarrow a = 0$).
 - Store tableau rows (in the previous order) that have a ratio $\leq 2^{12}$.
 - Up to n tableau rows are found, or no more available.
- Orientation selection:

How we separate:

How we separate:

- Relaxation selection:
 - Sort fractional integer variables by pseudo cost.
 - Discard numerically ill tableau rows
($|a| \leq 10^{-6} \Rightarrow a = 0$).
 - Store tableau rows (in the previous order) that have a ratio $\leq 2^{12}$.
 - Up to n tableau rows are found, or no more available.
- Orientation selection: Try all!

How we separate:

How we separate:

- Relaxation selection:
 - Sort fractional integer variables by pseudo cost.
 - Discard numerically ill tableau rows ($|a| \leq 10^{-6} \Rightarrow a = 0$).
 - Store tableau rows (in the previous order) that have a ratio $\leq 2^{12}$.
 - Up to n tableau rows are found, or no more available.
- Orientation selection: Try all!
- Cut selection:

How we separate:

How we separate:

- Relaxation selection:
 - Sort fractional integer variables by pseudo cost.
 - Discard numerically ill tableau rows ($|a| \leq 10^{-6} \Rightarrow a = 0$).
 - Store tableau rows (in the previous order) that have a ratio $\leq 2^{12}$.
 - Up to n tableau rows are found, or no more available.
- Orientation selection: Try all!
- Cut selection:
 - Discard cuts with ratio $\geq 2^{15}$.

How we separate:

How we separate:

- Relaxation selection:
 - Sort fractional integer variables by pseudo cost.
 - Discard numerically ill tableau rows ($|a| \leq 10^{-6} \Rightarrow a = 0$).
 - Store tableau rows (in the previous order) that have a ratio $\leq 2^{12}$.
 - Up to n tableau rows are found, or no more available.
- Orientation selection: Try all!
- Cut selection:
 - Discard cuts with ratio $\geq 2^{15}$.
 - Select at most N cuts minimizing $1/\max_abs$.

How we separate:

How we separate:

- Relaxation selection:
 - Sort fractional integer variables by pseudo cost.
 - Discard numerically ill tableau rows ($|a| \leq 10^{-6} \Rightarrow a = 0$).
 - Store tableau rows (in the previous order) that have a ratio $\leq 2^{12}$.
 - Up to n tableau rows are found, or no more available.
- Orientation selection: Try all!
- Cut selection:
 - Discard cuts with ratio $\geq 2^{15}$.
 - Select at most N cuts minimizing $1/\max_abs$.
 - Allow multiple rounds.

What we compare:

What we compare:

- We implement the cut-generation procedure as a CPLEX cut-callback.
- Use it under default CPLEX 11.0 settings (including cut generation) and preprocessing.
- We compare:
 - CPLEX 11.0
 - CPLEX 11.0 with our cut-generation procedure
 - CPLEX 11.0 with our cut-generation procedure and preprocessing
- All runs with two hours time limit.
- Base results will be CPLEX defaults with pre-processing.
- Cut generation done only at root node.

What we compare:

What we compare:

- We implement the cut-generation procedure as a CPLEX cut-callback.
- Use it under default CPLEX 11.0 settings (including cut generation) and preprocessing.
- We compare:
 - root LP value.
 - total number of nodes (excluding those that are pruned by the cut-generation procedure).
- All runs with two hours time limit.
- Base results will be CPLEX defaults with pre-processing.
- Cut generation done only at root node.

What we compare:

What we compare:

- We implement the cut-generation procedure as a CPLEX cut-callback.
- Use it under default CPLEX 11.0 settings (including cut generation) and preprocessing.
- We compare:
 - root LP value.
 - final running time for those that finish the run.
 - final B&B bound after time limit.
- All runs with two hours time limit.
- Base results will be CPLEX defaults with pre-processing.
- Cut generation done only at root node.

What we compare:

What we compare:

- We implement the cut-generation procedure as a CPLEX cut-callback.
- Use it under default CPLEX 11.0 settings (including cut generation) and preprocessing.
- We compare:
 - root LP value.
 - final running time for those that finish the run.
 - final B&B bound after time limit.
- All runs with two hours time limit.
- Base results will be CPLEX defaults with pre-processing.
- Cut generation done only at root node.

What we compare:

What we compare:

- We implement the cut-generation procedure as a CPLEX cut-callback.
- Use it under default CPLEX 11.0 settings (including cut generation) and preprocessing.
- We compare:
 - root LP value.
 - final running time for those that finish the run.
 - final B&B bound after time limit.
- All runs with two hours time limit.
- Base results will be CPLEX defaults with pre-processing.
- Cut generation done only at root node.

What we compare:

What we compare:

- We implement the cut-generation procedure as a CPLEX cut-callback.
- Use it under default CPLEX 11.0 settings (including cut generation) and preprocessing.
- We compare:
 - root LP value.
 - final running time for those that finish the run.
 - final B&B bound after time limit.
- All runs with two hours time limit.
- Base results will be CPLEX defaults with pre-processing.
- Cut generation done only at root node.

What we compare:

- We implement the cut-generation procedure as a CPLEX cut-callback.
- Use it under default CPLEX 11.0 settings (including cut generation) and preprocessing.
- We compare:
 - root LP value.
 - final running time for those that finish the run.
 - final B&B bound after time limit.
- All runs with two hours time limit.
- Base results will be CPLEX defaults with pre-processing.
- Cut generation done only at root node.

What we compare:

What we compare:

- We implement the cut-generation procedure as a CPLEX cut-callback.
- Use it under default CPLEX 11.0 settings (including cut generation) and preprocessing.
- We compare:
 - root LP value.
 - final running time for those that finish the run.
 - final B&B bound after time limit.
- All runs with two hours time limit.
- Base results will be CPLEX defaults with pre-processing.
- Cut generation done only at root node.

What we compare:

What we compare:

- We implement the cut-generation procedure as a CPLEX cut-callback.
- Use it under default CPLEX 11.0 settings (including cut generation) and preprocessing.
- We compare:
 - root LP value.
 - final running time for those that finish the run.
 - final B&B bound after time limit.
- All runs with two hours time limit.
- Base results will be CPLEX defaults with pre-processing.
- Cut generation done only at root node.

What we compare:

Where we compare:

- Instances from MIPLIB 3.0, MIPLIB 2003, and others.
- Total of 173 problems.
- Discarded the following problems:
 - `air`
 - `air1`
 - `air2`
 - `air3`
 - `air4`
 - `air5`
 - `air6`
 - `air7`
 - `air8`
 - `air9`
 - `air10`
 - `air11`
 - `air12`
 - `air13`
 - `air14`
 - `air15`
 - `air16`
 - `air17`
 - `air18`
 - `air19`
 - `air20`
 - `air21`
 - `air22`
 - `air23`
 - `air24`
 - `air25`
 - `air26`
 - `air27`
 - `air28`
 - `air29`
 - `air30`
 - `air31`
 - `air32`
 - `air33`
 - `air34`
 - `air35`
 - `air36`
 - `air37`
 - `air38`
 - `air39`
 - `air40`
 - `air41`
 - `air42`
 - `air43`
 - `air44`
 - `air45`
 - `air46`
 - `air47`
 - `air48`
 - `air49`
 - `air50`
 - `air51`
 - `air52`
 - `air53`
 - `air54`
 - `air55`
 - `air56`
 - `air57`
 - `air58`
 - `air59`
 - `air60`
 - `air61`
 - `air62`
 - `air63`
 - `air64`
 - `air65`
 - `air66`
 - `air67`
 - `air68`
 - `air69`
 - `air70`
 - `air71`
 - `air72`
 - `air73`
 - `air74`
 - `air75`
 - `air76`
 - `air77`
 - `air78`
 - `air79`
 - `air80`
 - `air81`
 - `air82`
 - `air83`
 - `air84`
 - `air85`
 - `air86`
 - `air87`
 - `air88`
 - `air89`
 - `air90`
 - `air91`
 - `air92`
 - `air93`
 - `air94`
 - `air95`
 - `air96`
 - `air97`
 - `air98`
 - `air99`
 - `air100`
 - `air101`
 - `air102`
 - `air103`
 - `air104`
 - `air105`
 - `air106`
 - `air107`
 - `air108`
 - `air109`
 - `air110`
 - `air111`
 - `air112`
 - `air113`
 - `air114`
 - `air115`
 - `air116`
 - `air117`
 - `air118`
 - `air119`
 - `air120`
 - `air121`
 - `air122`
 - `air123`
 - `air124`
 - `air125`
 - `air126`
 - `air127`
 - `air128`
 - `air129`
 - `air130`
 - `air131`
 - `air132`
 - `air133`
 - `air134`
 - `air135`
 - `air136`
 - `air137`
 - `air138`
 - `air139`
 - `air140`
 - `air141`
 - `air142`
 - `air143`
 - `air144`
 - `air145`
 - `air146`
 - `air147`
 - `air148`
 - `air149`
 - `air150`
 - `air151`
 - `air152`
 - `air153`
 - `air154`
 - `air155`
 - `air156`
 - `air157`
 - `air158`
 - `air159`
 - `air160`
 - `air161`
 - `air162`
 - `air163`
 - `air164`
 - `air165`
 - `air166`
 - `air167`
 - `air168`
 - `air169`
 - `air170`
 - `air171`
 - `air172`
 - `air173`
- Differences under 5% are not considered in the averages.
- Final test set has 96 problems.

What we compare:

Where we compare:

- Instances from MIPLIB 3.0, MIPLIB 2003, and others.
- Total of 173 problems.
- Discarded the following problems:
 - Solved by all configurations in under 5 seconds: 48.
 - 100% solved by all configurations: 14.
 - Solved by all configurations within 5% difference: 111.
- Differences under 5% are not considered in the averages.
- Final test set has 96 problems.

What we compare:

Where we compare:

- Instances from MIPLIB 3.0, MIPLIB 2003, and others.
- Total of 173 problems.
- Discarded the following problems:
 - Solved by all configurations in under 5 seconds: 48.
 - LP root gap less than 0.1% in all configurations: 26.
 - Two unstable problems discarded, roll3000, l152av.
- Differences under 5% are not considered in the averages.
- Final test set has 96 problems.

What we compare:

Where we compare:

- Instances from MIPLIB 3.0, MIPLIB 2003, and others.
- Total of 173 problems.
- Discarded the following problems:
 - Solved by all configurations in under 5 seconds: 48.
 - LP root gap less than 0.1% in all configurations: 26.
 - Two unstable problems discarded, roll3000, l152av.
- Differences under 5% are not considered in the averages.
- Final test set has 96 problems.

What we compare:

Where we compare:

- Instances from MIPLIB 3.0, MIPLIB 2003, and others.
- Total of 173 problems.
- Discarded the following problems:
 - Solved by all configurations in under 5 seconds: 48.
 - LP root gap less than 0.1% in all configurations: 26.
 - Two unstable problems discarded, roll3000, l152av.
- Differences under 5% are not considered in the averages.
- Final test set has 96 problems.

What we compare:

Where we compare:

- Instances from MIPLIB 3.0, MIPLIB 2003, and others.
- Total of 173 problems.
- Discarded the following problems:
 - Solved by all configurations in under 5 seconds: 48.
 - LP root gap less than 0.1% in all configurations: 26.
 - Two unstable problems discarded, roll3000, l152av.
- Differences under 5% are not considered in the averages.
- Final test set has 96 problems.

What we compare:

Where we compare:

- Instances from MIPLIB 3.0, MIPLIB 2003, and others.
- Total of 173 problems.
- Discarded the following problems:
 - Solved by all configurations in under 5 seconds: 48.
 - LP root gap less than 0.1% in all configurations: 26.
 - Two unstable problems discarded, roll3000, l152av.
- Differences under 5% are not considered in the averages.
- Final test set has 96 problems.

What we compare:

Where we compare:

- Instances from MIPLIB 3.0, MIPLIB 2003, and others.
- Total of 173 problems.
- Discarded the following problems:
 - Solved by all configurations in under 5 seconds: 48.
 - LP root gap less than 0.1% in all configurations: 26.
 - Two unstable problems discarded, roll3000, l152av.
- Differences under 5% are not considered in the averages.
- Final test set has 96 problems.

What we compare:

Tested settings

- Will compare several configurations:
 - default: Cplex 11.0 defaults with pre-processing.
 - General naming scheme will be C-Tt-Nn:
 - $C \in \{ \text{Gomory (G)}, T1, T2 \}$, indicate the type of lattice-free set used.
 - $t \in \{2, 5, 10\}$, is an upper bound on the number of tableau rows selected at each round of cuts at the root node.
 - $n \in \{5, 20, 100\}$, is an upper bound on the number of cuts to be added at the root node.

What we compare:

Tested settings

- Will compare several configurations:
 - default: Cplex 11.0 defaults with pre-processing.
 - General naming scheme will be C-Tt-Nn:
 - $C \in \{ \text{Gomory } (G), T1, T2 \}$, indicate the type of lattice-free set used.
 - $t \in \{2, 5, 10\}$, is an upper bound on the number of tableau rows selected at each round of cuts at the root node.
 - $n \in \{5, 20, 100\}$, is an upper bound on the number of cuts to be added at the root node.

What we compare:

Tested settings

- Will compare several configurations:
 - default: Cplex 11.0 defaults with pre-processing.
 - General naming scheme will be C-Tt-Nn:
 - $C \in \{ \text{Gomory } (G), T1, T2 \}$, indicate the type of lattice-free set used.
 - $t \in \{2, 5, 10\}$, is an upper bound on the number of tableau rows selected at each round of cuts at the root node.
 - $n \in \{5, 20, 100\}$, is an upper bound on the number of cuts to be added at the root node.

What we compare:

Tested settings

- Will compare several configurations:
 - default: Cplex 11.0 defaults with pre-processing.
 - General naming scheme will be C-Tt-Nn:
 - $C \in \{ \text{Gomory (G)}, T1, T2 \}$, indicate the type of lattice-free set used.
 - $t \in \{2, 5, 10\}$, is an upper bound on the number of tableau rows selected at each round of cuts at the root node.
 - $n \in \{5, 20, 100\}$, is an upper bound on the number of cuts to be added at the root node.

What we compare:

Tested settings

- Will compare several configurations:
 - default: Cplex 11.0 defaults with pre-processing.
 - General naming scheme will be C-Tt-Nn:
 - $C \in \{ \text{Gomory (G)}, T1, T2 \}$, indicate the type of lattice-free set used.
 - $t \in \{2, 5, 10\}$, is an upper bound on the number of tableau rows selected at each round of cuts at the root node.
 - $n \in \{5, 20, 100\}$, is an upper bound on the number of cuts to be added at the root node.

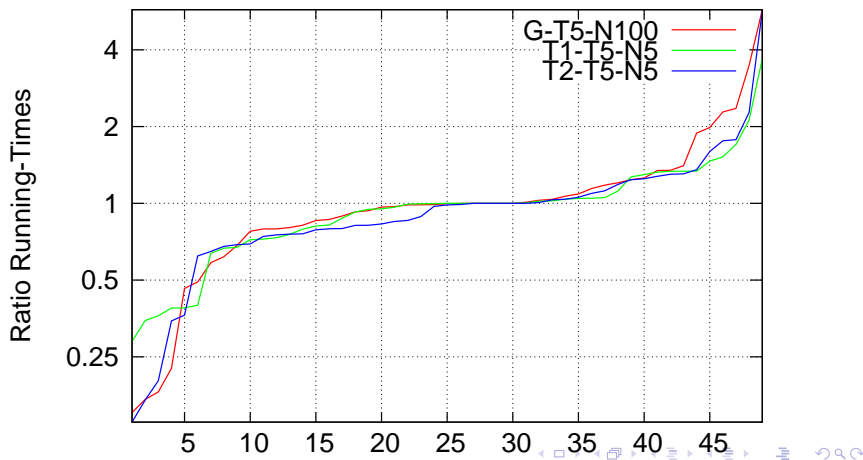
What we compare:

Tested settings

- Will compare several configurations:
 - default: Cplex 11.0 defaults with pre-processing.
 - General naming scheme will be C-Tt-Nn:
 - $C \in \{ \text{Gomory } (G), T1, T2 \}$, indicate the type of lattice-free set used.
 - $t \in \{2, 5, 10\}$, is an upper bound on the number of tableau rows selected at each round of cuts at the root node.
 - $n \in \{5, 20, 100\}$, is an upper bound on the number of cuts to be added at the root node.

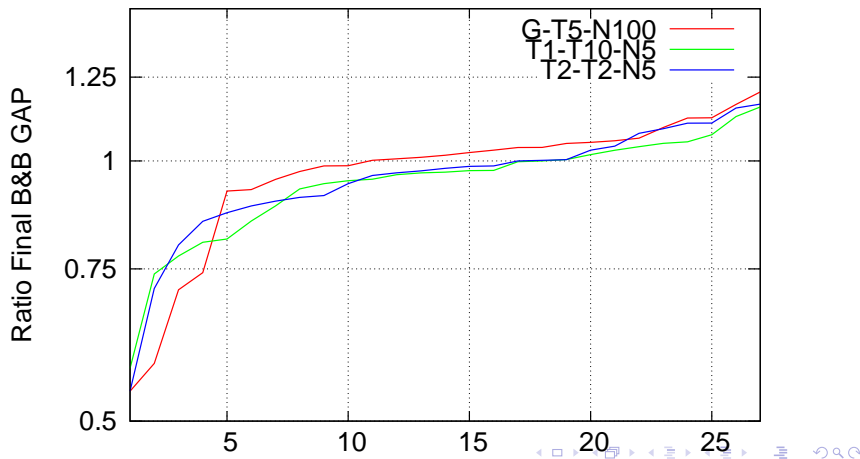
The Results

Overall speed-up (49 instances) 6.8%, 8.3%,
11.8%



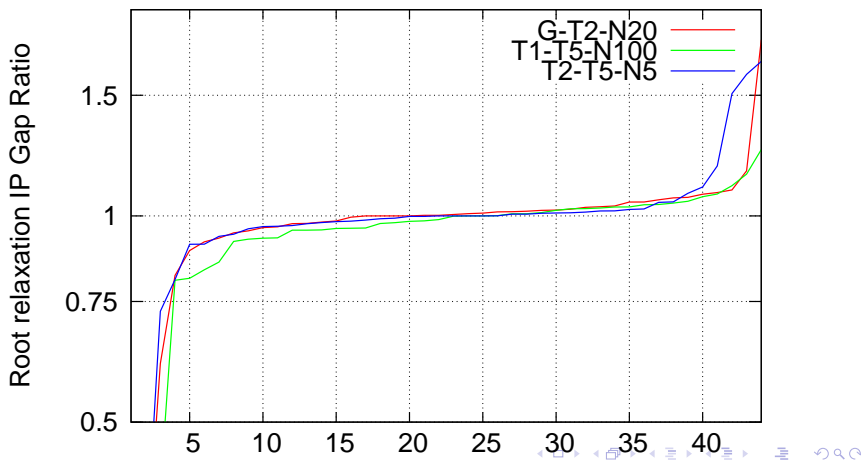
The Results

Closed B&B GAP_{MIP} (27 instances) 3.7%,
6.0%, 4.5%

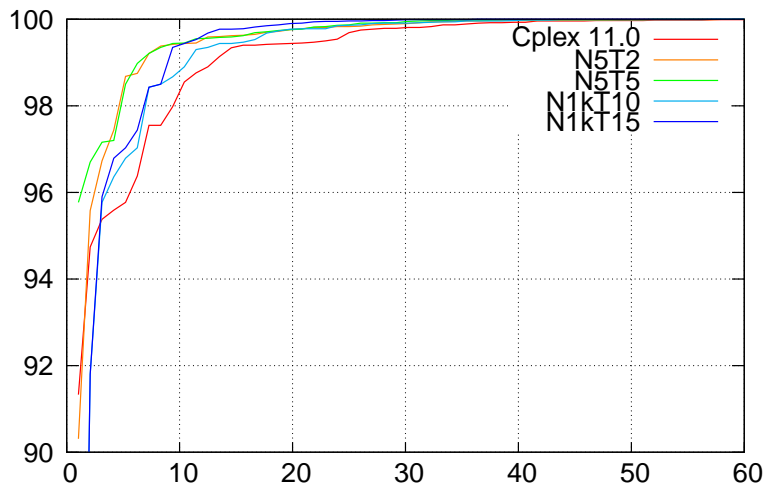


The Results

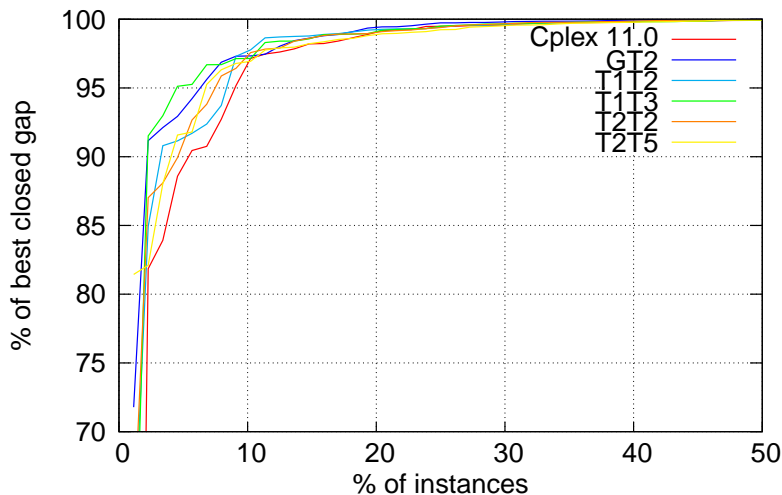
Closed Root GAP_{LP} (44 instances) 7.2%,
8.3%, 5.7%



Performance Profile



Performance Profile II



Conclusions

- Of all tested configurations, only two had worst results on Root LP gap and on B&B gap, and eight had worst results on speed.
- Although the improvements are not dramatic, they still are important.
- Lots of testing of parameters.
- Numerical issues are important!
- Could we do a full separation?

Conclusions

- Of all tested configurations, only two had worst results on Root LP gap and on B&B gap, and eight had worst results on speed.
- Although the improvements are not dramatic, they still are important.
- Lots of testing of parameters.
- Numerical issues are important!
- Could we do a full separation?

Conclusions

- Of all tested configurations, only two had worst results on Root LP gap and on B&B gap, and eight had worst results on speed.
- Although the improvements are not dramatic, they still are important.
- Lots of testing of parameters.
- Numerical issues are important!
- Could we do a full separation?

Conclusions

- Of all tested configurations, only two had worst results on Root LP gap and on B&B gap, and eight had worst results on speed.
- Although the improvements are not dramatic, they still are important.
- Lots of testing of parameters.
- Numerical issues are important!
- Could we do a full separation?

Conclusions

- Of all tested configurations, only two had worst results on Root LP gap and on B&B gap, and eight had worst results on speed.
- Although the improvements are not dramatic, they still are important.
- Lots of testing of parameters.
- Numerical issues are important!
- Could we do a full separation?