Identifying Rank 2 Chvátal-Gomory Inequalities

Brady Hunsaker

University of Pittsburgh

Craig Tovey

Ellis Johnson

Georgia Institute of Technology

hunsaker@engr.pitt.edu

Outline

- Motivation
- Formulation of problems
- Computational issues
- User interface
- Examples
- Open-source tools used
- Future development

Chvátal-Gomory inequalities

- Valid inequalities formed by using known valid inequalities.
- Technique explicitly presented by Chvátal (1973); implicit in earlier work by Gomory (1958).

1. Known valid inequalities: $\sum_{j=1}^{n} a_{ij}x_j \leq b_i$, $i = 1, \ldots, m$, and $x_j \geq 0, j = 1, \ldots, n$.
2. Select nonnegative multipliers $u_1, \ldots, u_m$.
3. The resulting Chvátal-Gomory inequality is

$$\sum_{j=1}^{n} \left\lfloor \sum_{i=1}^{m} u_i a_{ij} \right\rfloor x_j \leq \left\lfloor \sum_{i=1}^{m} u_i b_i \right\rfloor .$$

Chvátal-Gomory rank

Use C-G inequalities and original inequalities to generate new C-G inequalities.

- The C-G rank of the initial inequalities and any inequality dominated by a nonnegative linear combination of the initial inequalities is 0.
- Any inequality which can be generated as a C-G inequality based on rank 0 inequalities—but is not rank 0 itself—has C-G rank 1.
- A valid inequality has C-G rank k if it can be generated as a C-G inequality based on inequalities of rank $0, 1, \ldots, k-1$, but does not have rank less than k.

Motivation

Given an IP instance and a valid inequality, what is its Chvátal-Gomory rank?

- This question is often considered when new valid inequalities are proposed for a problem.
- How much tighter is the rank 1 polyhedron than the LP relaxation?
- Determining an upper bound on the rank is often done by trial-and-error.
- Lower bound proofs are generally even harder.
- Not aware of any existing computational method for testing rank 2.

Formulations and Computational Issues

An LP for Rank 1

- Candidate inequality: $\sum_{j=1}^{n} f_j x_j \leq g$ with $f_j \in \mathbf{Z}, g \in \mathbf{Z}$.
- If rank 1 C-G inequality, then there exist nonnegative u_i such that $\sum_{i=1}^{m} u_i a_{ij} \geq f_j$ for all j and such that $\sum_{i=1}^{m} u_i b_i < g + 1$.
- LP for “best” multipliers:

$$\begin{array}{lrcll} \min & \sum_{i=1}^{m} b_i u_i & & & \\ s.t. & \sum_{i=1}^{m} u_i a_{ij} & \geq & f_j & j = 1, \ldots, n \\ & u_i & \geq & 0 & i = 1, \ldots, m \end{array}$$

- If the optimal objective value satisfies $z \leq g$, then it is rank 0. If $g < z < g + 1$, then the inequality is rank 1. Otherwise it is rank > 1 or invalid.

Column generation for Rank 2

- Can form a similar LP for rank 2, but need a multiplier variable for each C-G inequality of rank 1.
- Use the same LP with the following column generation for a rank 1 inequality $\sum_{j=1}^{n} \alpha_j x_j \leq \beta$. π is the vector of dual prices for the base LP.

$$
\begin{array}{lrcll}
\max & \sum_{j=1}^{n} \pi_j \alpha_j - \beta & & & \\
s.t. & \sum_{i=1}^{m} u_i a_{ij} & = & \hat{\alpha}_j & j = 1, \ldots, n \\
 & \sum_{i=1}^{m} u_i b_i & = & \hat{\beta} & \\
 & \hat{\alpha}_j & \geq & \alpha_j & j = 1, \ldots, n \\
 & \lfloor \hat{\beta} \rfloor & \leq & \beta & \\
 & u_i & \geq & 0 & i = 1, \ldots, m \\
 & \alpha_j, \beta & \text{integer} & &
\end{array}
$$

- But this is not a linear IP.

Column generation for Rank 2

- $\lfloor \hat{\beta} \rfloor \leq \beta$ is the same as $\hat{\beta} - 1 < \beta$
- Replace that with $\hat{\beta} - 1 + \epsilon \leq \beta$, where ϵ is small

$$\begin{array}{rrcll} \max & \sum_{j=1}^{n} \pi_j \alpha_j - \beta & & & \\ s.t. & \sum_{i=1}^{m} u_i a_{ij} & = & \hat{\alpha}_j & j = 1, \ldots, n \\ & \sum_{i=1}^{m} u_i b_i & = & \hat{\beta} & \\ & \hat{\alpha}_j & \geq & \alpha_j & j = 1, \ldots, n \\ & \hat{\beta} - 1 + \epsilon & \leq & \beta & \\ & u_i & \geq & 0 & i = 1, \ldots, m \\ & \alpha_j, \beta & \text{integer} & & \end{array}$$

Optimization over Rank 1

- Think of π as a point in $\mathbf{R}^n$. Then we may view the column generation as separation.
- This allows us to separate over rank 1 inequalities.

Certificates of rank

Rank	Certificate
≤ 1	Multipliers for rank 0 inequalities
> 1	Point x that satisfies rank 0 inequalities and $\sum_{j=1}^{n} f_j x_j \geq g + 1$
≤ 2	Multipliers for rank 0 and rank 1 inequalities; also certificates of rank ≤ 1 for each rank 1 inequality used

- Each of the certificates above is succinct.
- Unfortunately, the “certificate” for rank > 2 has to prove an IP infeasibility.

Tolerances & bounds are important

Potential numerical trouble due to the interaction of ϵ, the particular instance, and the tolerances of the solver.

- Although the epsilon was written as an absolute epsilon, it is really a relative epsilon.
- May need to try several values of ϵ.
- Need upper bounds on the multipliers u_i: no multiplier needs to be greater than 1.

User Interface and Examples

Input may be AMPL or MPS

Two inputs are necessary:

IP instance This defines the rank 0 constraints. Two possible formats:

- GNU MathProg (subset of AMPL)
- MPS

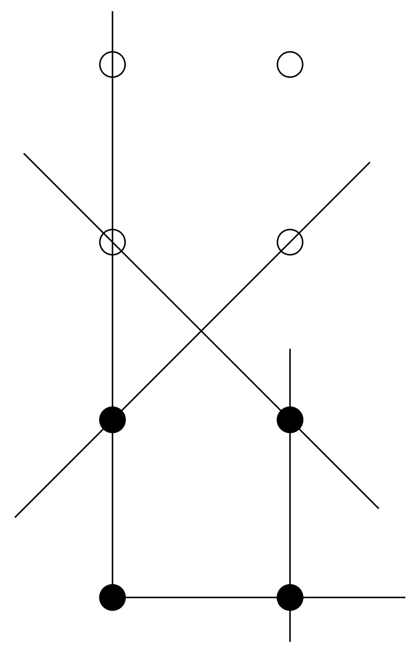
Constraints to test Specified in a text file containing a vector of constraint coefficients on each line.

Output is text, including certificates of rank.

Example: High rank

Textbook example:

$$\begin{aligned} tx_1 + x_2 &\leq 1 + t \\ -tx_1 + x_2 &\leq 1 \\ x_1 &\leq 1 \\ x_1, x_2 &\geq 0 \end{aligned}$$



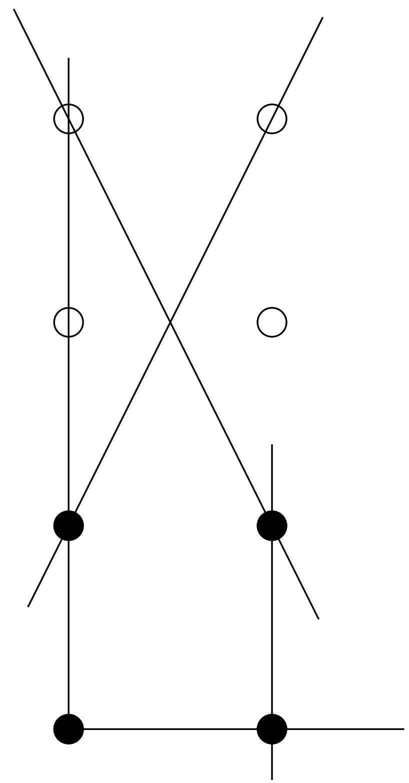
Valid inequality: $x_2 \leq 1$ has rank t.

High rank: $t = 1$ output

```
Coin0001I At line 7 NAME          classic1
Coin0001I At line 8 ROWS
Coin0001I At line 13 COLUMNS
Coin0001I At line 17 RHS
Coin0001I At line 20 ENDATA
Coin0002I Problem classic1 has 3 rows, 2 columns and 5 elements
Coin0008I classic1 read with 0 errors
=======================================================
Testing constraint: 0 1 <= 1
Clp0005I 0  Objective 0 Primal infeas 1 (1)
Clp0005I 2  Objective 1.5
Clp0000I Optimal - objective value 1.5
Rank of inequality is 1
Multipliers to achieve rank 1:
0.5: 1 1  2 1
0.5: 1 -1  2 1
```

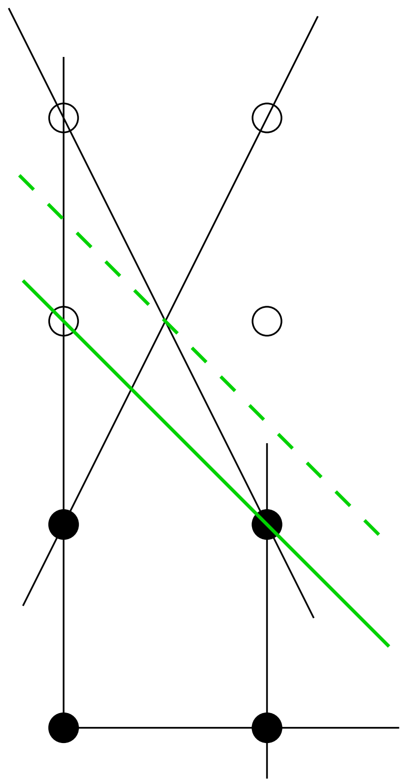
High rank: $t = 2$ output excerpts

```
Testing constraint: 0 1 <= 1
Rank of inequality is at least 2
The following point satisfies all rank 0 inequalities
and proves rank 2 or greater:
1 0.5  2 2
Adding row: 1 1 <= 2
Multipliers to create rank 1 row:
0.874997000012: 1 2  2 1
0.374997000012: 1 -2  2 1
Rank 2
Multipliers to achieve rank 2:
0.333333333333: 1 -2  2 1
0.666666666667: 1 1  2 1
```



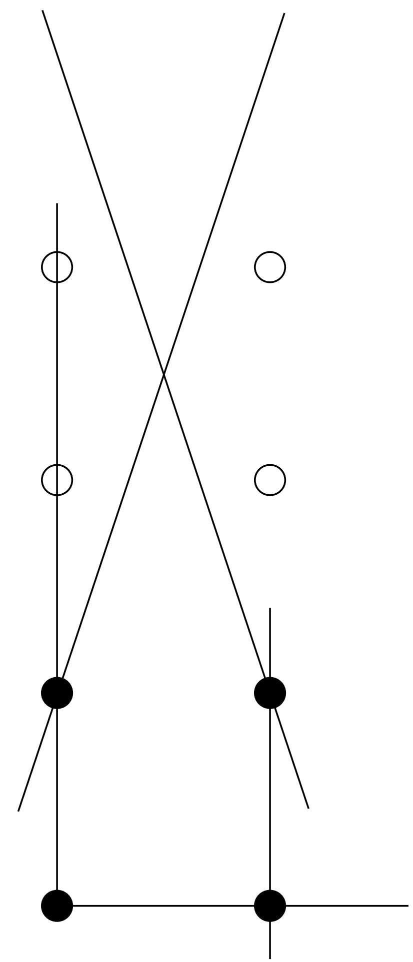
High rank: $t = 2$ output excerpts

```
Testing constraint: 0 1 <= 1
Rank of inequality is at least 2
The following point satisfies all rank 0 inequalities
and proves rank 2 or greater:
1 0.5  2 2
Adding row: 1 1 <= 2
Multipliers to create rank 1 row:
0.874997000012: 1 2  2 1
0.374997000012: 1 -2  2 1
Rank 2
Multipliers to achieve rank 2:
0.333333333333: 1 -2  2 1
0.666666666667: 1 1  2 1
```



High rank: $t = 3$ output excerpts

```
Testing constraint: 0 1 <= 1
Rank of inequality is at least 2
The following point satisfies all rank 0 inequalities
and proves rank 2 or greater:
1 0.5 2 2.5
Adding row: 0 1 <= 2
Multipliers to create rank 1 row:
0.59999760001: 1 3  2 1
0.59999760001: 1 -3  2 1
Rank 3 or more
```



Example: Knapsack

$$\begin{aligned} \sum_{i=1}^{12} i x_i &\leq 12 \\ x_i &\geq 0 \end{aligned}$$

Facet Coefficients	RHS	C-G Rank
0 0 0 0 0 1 2 2 2 2 2 2	2	
0 0 0 0 1 1 1 2 2 2 2 2	2	
0 0 0 2 2 3 4 4 6 6 6 6	6	
0 0 1 1 2 2 2 3 3 4 4 4	4	
0 0 1 2 3 3 3 4 5 6 6 6	6	
0 0 3 4 5 6 7 8 9 12 12 12	12	
0 1 1 2 2 3 4 4 5 5 6 6	6	
0 1 3 4 4 6 8 8 9 11 12 12	12	

Example: Knapsack

$$\begin{aligned} \sum_{i=1}^{12} i x_i &\leq 12 \\ x_i &\geq 0 \end{aligned}$$

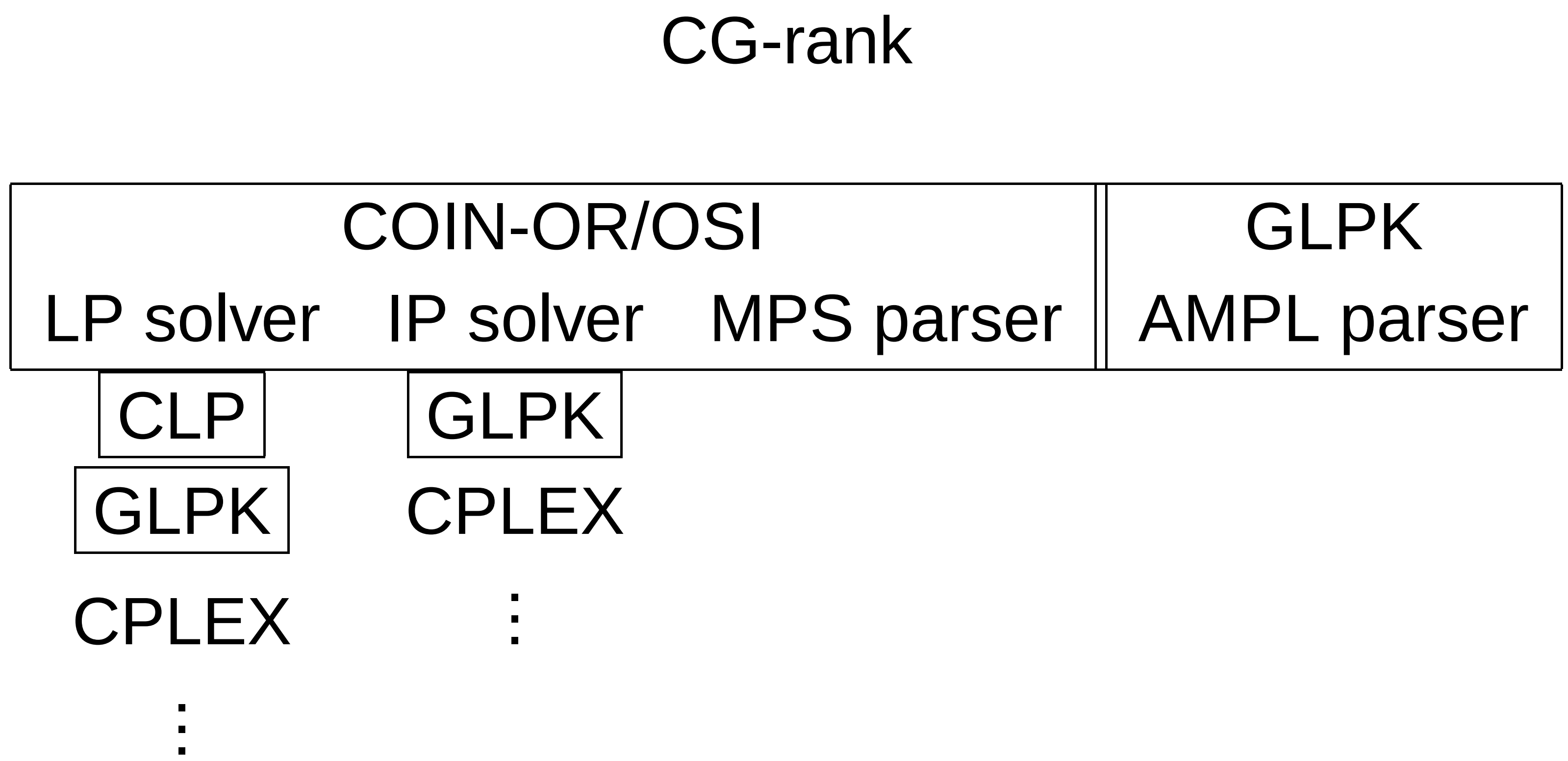
Facet Coefficients	RHS	C-G Rank
0 0 0 0 0 1 2 2 2 2 2 2	2	2
0 0 0 0 1 1 1 2 2 2 2 2	2	2
0 0 0 2 2 3 4 4 6 6 6 6	6	3+
0 0 1 1 2 2 2 3 3 4 4 4	4	1
0 0 1 2 3 3 3 4 5 6 6 6	6	2
0 0 3 4 5 6 7 8 9 12 12 12	12	3+
0 1 1 2 2 3 4 4 5 5 6 6	6	1
0 1 3 4 4 6 8 8 9 11 12 12	12	3+

Open-source tools used

CG-rank

LP solver   IP solver   MPS parser   AMPL parser

Open-source tools used



Availability and license

- Source code will be available at http://www.engr.pitt.edu/hunsaker/cg-rank/
- Licensed under GNU GPL.
- Also need to download COIN-OR/OSI and GLPK in order to compile
- Email hunsaker@engr.pitt.edu for assistance

Future work

- Improve column generation to allow for larger instances
- Allow test constraints to be specified in MPS or GNU MathProg
- Make linking with GLPK optional
- Optimization over rank 1 inequalities
- Extend to rank 3