Introduction to Algorithms

Kostja Siefen

Senior Technical Account Manager

GUROBI OPTIMIZATION

The World's Fastest Solver

About the Speaker

Kostja Siefen

Senior Technical Account Manager

Gurobi Optimization

Agenda

1. Linear programming (LP)
    a. Define LP
    b. LP examples
    c. Algorithms (high-level)

2. Mixed-integer programming (MIP)
    a. Define MIP
    b. MIP example

3. Basic techniques for solving MIP
    a. Bounding the optimal solution
    b. Bound-and-bound
    c. Improvements (heuristics, cuts, etc.)
    d. Termination criteria

4. Log file interpretation
5. Outlook: Tuning

Introducing LP

What is LP?

$$
\begin{array}{lrcl}
\min\limits_{x \in \mathbb{R}^n} & f(x) & & \\
\textbf{subject to} & h_1(x) & \leq & 0 \\
 & h_2(x) & \leq & 0 \\
 & & \vdots & \\
 & h_m(x) & \leq & 0
\end{array}
$$

- **This is a Linear Programming (LP) instance if:**
  - The objective function $f(x)$ is linear
    - e.g., $f(x) = 3x1 + 2x2$
  - The functions $h_1(x), \ldots, h_m(x)$ are linear
    - e.g., $h_1(x) = x1 - 17x2 + 10$, etc.

Quiz time!

$$
\begin{array}{rrcrcrcr}
\min\limits_{x\in\mathbb{R}^2} & x_1 & - & x_2 & & & & \\
\text{subject to} & x_1 & + & 17x_2 & - & 10 & \leq & 0 \\
 & 2x_1 & + & 3x_2 & - & 5 & \leq & 0 \\
 & -x_1 & & & & & \leq & 0 \\
 & & & & & -x_2 & \leq & 0
\end{array}
$$

Quiz time!

$$
\begin{array}{rrcrcrcr}
\min\limits_{x \in \mathbb{R}^2} & \boxed{x_1^2} & - & x_2 & & & & \\
\text{subject to} & x_1 & + & 17x_2 & - & 10 & \leq & 0 \\
 & 2x_1 & + & 3x_2 & - & 5 & \leq & 0 \\
 & -x_1 & & & & & \leq & 0 \\
 & & & & & -x_2 & \leq & 0
\end{array}
$$

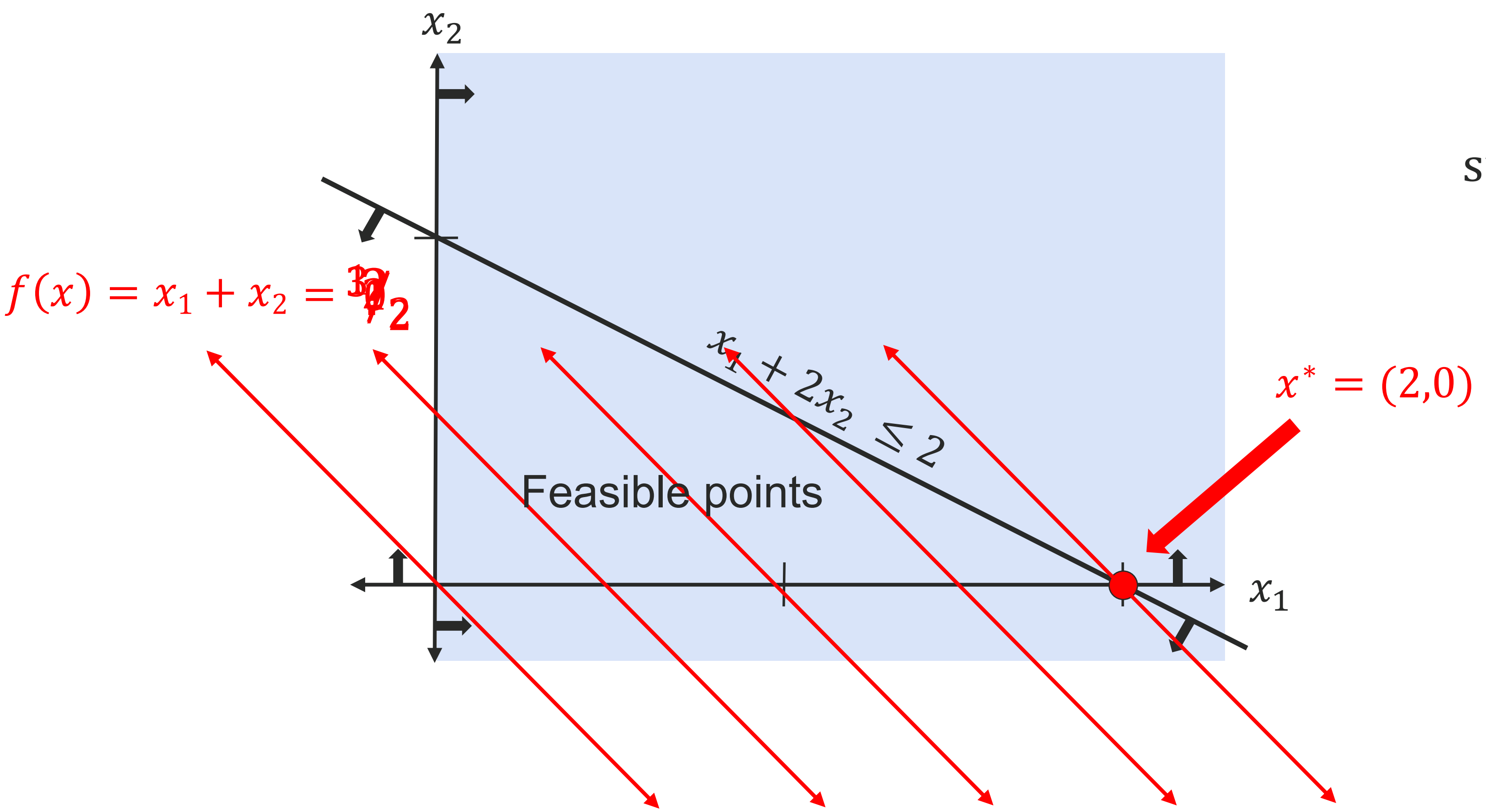
Quiz time!

$$
\begin{array}{rrcrcrcc}
\min\limits_{x\in\mathbb{R}^2} & |x_1 & - & x_2| & & & & \\
\text{subject to} & x_1 & + & 17x_2 & - & 10 & \leq & 0 \\
 & 2x_1 & + & 3x_2 & - & 5 & \leq & 0 \\
 & -x_1 & & & & & \leq & 0 \\
 & & & & & -x_2 & \leq & 0
\end{array}
$$

Quiz time!

$$
\begin{aligned}
\max_{x\in\mathbb{R}^2} \quad & x_1 - x_2 & \iff \quad & -\min_{x\in\mathbb{R}^2} -(x_1 - x_2) \\
\text{subject to} \quad & x_1 + 17x_2 \leq 10 & \iff \quad & x_1 + 17x_2 - 10 \leq 0 \\
& 2x_1 + 3x_2 - 5 \geq 0 & \iff \quad & -2x_1 - 3x_2 + 5 \leq 0 \\
& 4x_1 + x_2 - 9 = 0 & \iff \quad & \begin{cases} 4x_1 + x_2 - 9 \leq 0, \\ -4x_1 - x_2 + 9 \leq 0, \end{cases} \\
& -x_1 \leq 0 & & \\
& -x_2 \leq 0 & &
\end{aligned}
$$

LP by picture

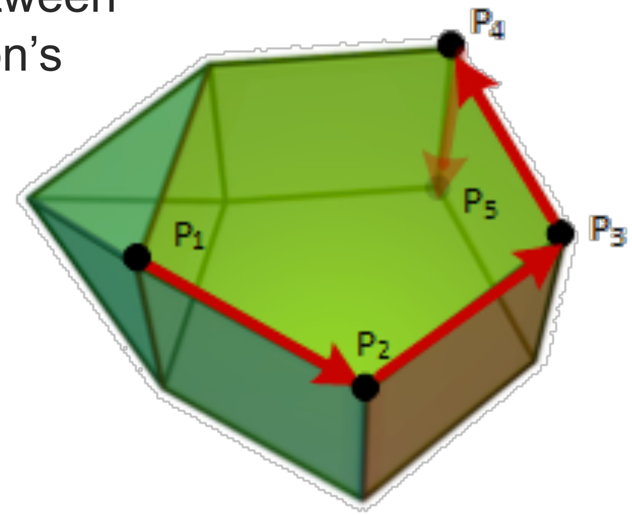


$$
\begin{array}{llllll}
\max\limits_{x\in\mathbb{R}^2} & x_1 & + & x_2 & & \\
\text{subject to} & x_1 & + & 2x_2 & \leq & 2 \\
 & x_1 & & & \geq & 0 \\
 & & & x_2 & \geq & 0
\end{array}
$$

Algorithmic LP ideas

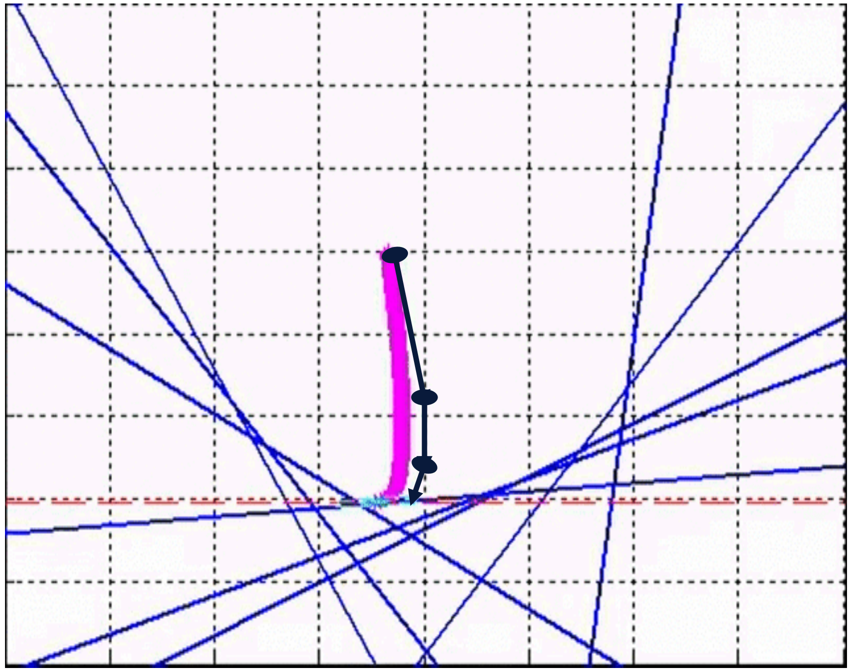
$$\min_{x \in \mathbb{R}^2} \quad c^T x$$
$$\text{subject to} \quad Ax \leq b$$

Traverses edges between polyhedron's vertices



Simplex

Generates sequence of points in interior of polyhedron



Barrier

img src: mathstools.com

Introducing MIP

What is MILP?

$$
\begin{array}{llll}
\min\limits_{x\in\mathbb{R}^n} & f(x) & & \\
\textbf{subject to} & h_1(x) & \leq & 0 \\
 & h_2(x) & \leq & 0 \\
 & & \vdots & \\
 & h_m(x) & \leq & 0 \\
\end{array}
$$

Some x_i are integer

- **This is a Mixed-Integer Linear Programming (MILP) instance if:**
  - The objective function $f(x)$ is linear
    - e.g., $f(x) = 3x1 + 2x2$
  - The functions $h_1(x), \dots, h_m(x)$ are linear
    - e.g., $h_1(x) = x1 - 17x2 + 10$, etc.
- **MILP = LP + integrality constraints!**

MIP example: furniture manufacturing

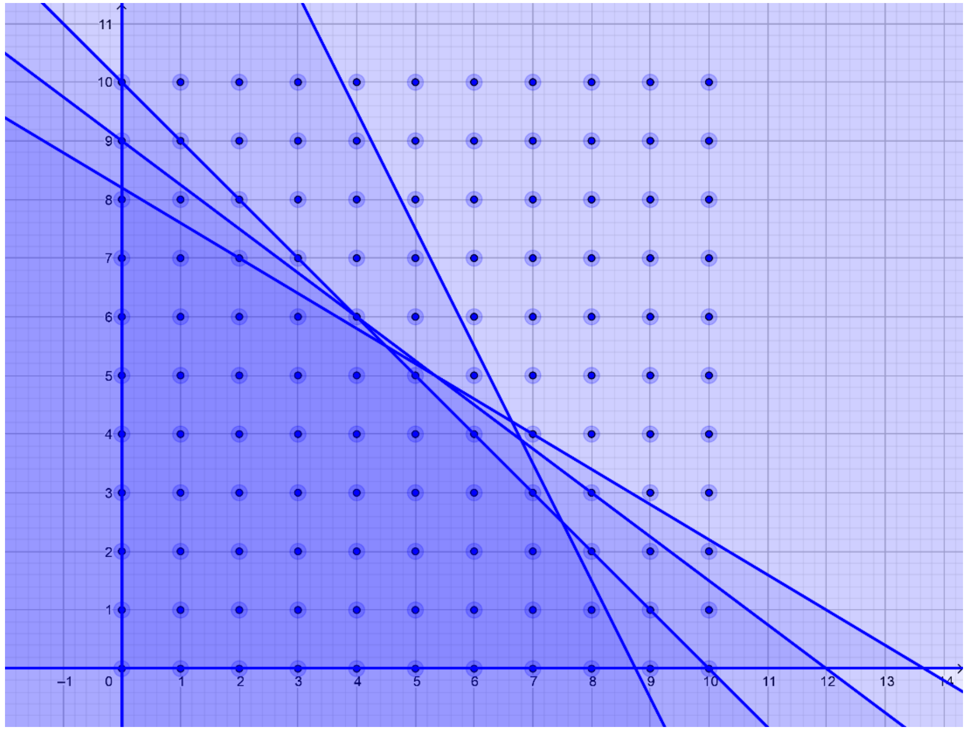
Variables:
- x: number of stools we make
- y: number of chairs we make

~~**Solution:**~~ $\cancel{x^* = 4.5, y^* = 5.5}$

	Stool (x)	Chair (y)	Available
Profit	\$10	\$11	↑↑↑

$$
\begin{array}{lrcrcl}
\Leftrightarrow & \max_{x,y} & 10x & + & 11y & \\
\Leftrightarrow & \text{subject to} & x & + & y & \leq 10 \\
\Leftrightarrow & & 3x & + & 4y & \leq 36 \\
\Leftrightarrow & & 3x & + & 5y & \leq 41 \\
\Leftrightarrow & & 4x & + & 2y & \leq 35 \\
& & \text{integer} & & x, y & \geq 0
\end{array}
$$

MIP by picture



$$
\begin{array}{rrcrcr}
\max_{x,y} & {\color{red}10x} & {\color{red}+} & {\color{red}11y} & & \\
\text{subject to} & x & + & y & \leq & 10 \\
 & 3x & + & 4y & \leq & 36 \\
 & 3x & + & 5y & \leq & 41 \\
 & 4x & + & 2y & \leq & 35 \\
 & & & \text{integer } x, y & \geq & 0
\end{array}
$$

Why not enumerate (try) all the integer feasible points?

Why not enumerate all combinations of 0-1 values?

GUROBI OPTIMIZATION

- **Try all 2^n points on Summit supercomputer at ORNL**
  - 200 petaflops (2×10^{17})

n	$0-1$ points to check	Time
10	1024	0 seconds
60	1.15×10^{18}	6 seconds
70	1.18×10^{21}	1.6 hours
120	1.33×10^{36}	210 billion years

Solving MIP

Bounds and MIP Gap

Let $f(x,y) \coloneqq 10x + 11y$

$$
\begin{aligned}
z^* \coloneqq \max_{x,y} \quad & 10x + 11y \\
\text{s.t.} \quad & x + y \leq 10 \\
& 3x + 4y \leq 36 \\
& 3x + 5y \leq 41 \\
& 4x + 2y \leq 35 \\
& \text{integer } x, y \geq 0
\end{aligned}
$$

$0 \leq z^* = \ ?$

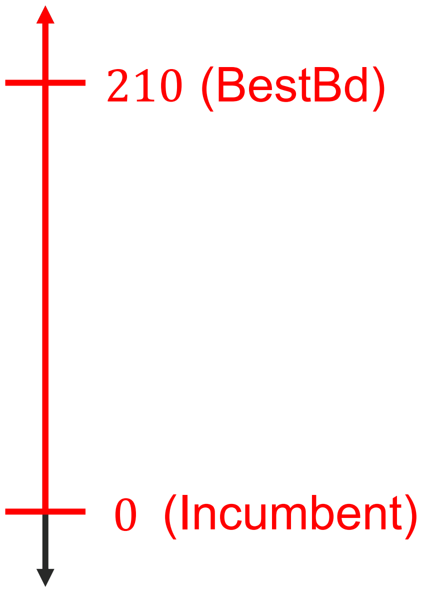
0 (Incumbent)

Can we try something simple?

- $x = 0, y = 0$ is a feasible solution!
- Then z^* must *at least as large* as f(0,0) = 0
- Currently, (0,0) is our *incumbent* solution

Bounds and MIP Gap

$$0 \le z^* \le 210$$



$$
\begin{array}{rl}
z^* := \max\limits_{x,y} & \boxed{10x + 11y} \\
\text{s.t.} & \boxed{x + y \le 10} \\
& 3x + 4y \le 36 \\
& 3x + 5y \le 41 \\
& 4x + 2y \le 35 \\
& \text{integer } x, y \ge 0
\end{array}
$$

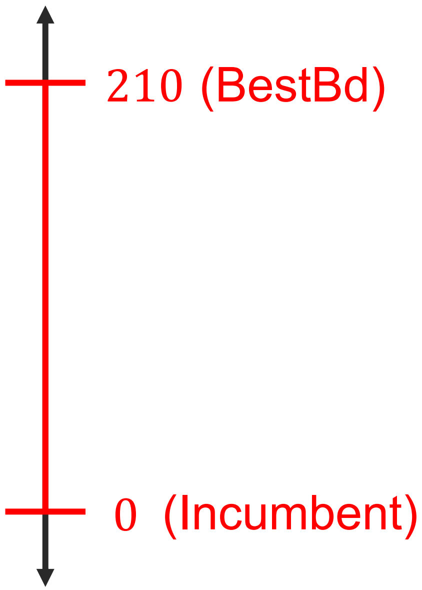
$$x + y \le 10$$
$$\Downarrow$$
$$x \le 10 \text{ and } y \le 10$$
$$\Downarrow$$
$$10x \le 100 \text{ and } 11y \le 110$$
$$\Downarrow$$
$$f(x,y) = 10x + 11y \le 210$$

What about upper bounds?

- Use linear constraints to get an upper bound on the optimal objective z^*
- *All* solutions have objective $z^* \le 210$
  - This is currently our *best (upper) bound*

Bounds and MIP Gap

$$0 \le z^* \le 210$$



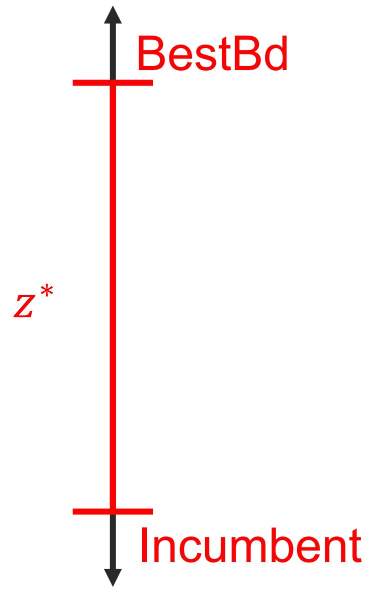
$$
\begin{array}{rrcrcrcr}
z^* := \max\limits_{x,y} & 10x & + & 11y & & \\
\text{s.t.} & x & + & y & \le & 10 \\
 & 3x & + & 4y & \le & 36 \\
 & 3x & + & 5y & \le & 41 \\
 & 4x & + & 2y & \le & 35 \\
\end{array}
$$

integer $x, y \ge 0$

How good is the incumbent?

- Measure the quality by the *(relative) MIP gap:* $\frac{|\text{BestBd} - \text{Incumbent}|}{|\text{Incumbent}|} \times 100\%$
- Current gap is $\frac{210-0}{0} \times 100\%$ (undefined or $+\infty$)
  - This is a very poor gap!

Bounds and MIP Gap



$$
\begin{array}{rrcrcrcr}
z^* := \max\limits_{x,y} & 10x & + & 11y & & \\
\text{s.t.} & x & + & y & \leq & 10 \\
 & 3x & + & 4y & \leq & 36 \\
 & 3x & + & 5y & \leq & 41 \\
 & 4x & + & 2y & \leq & 35 \\
\end{array}
$$

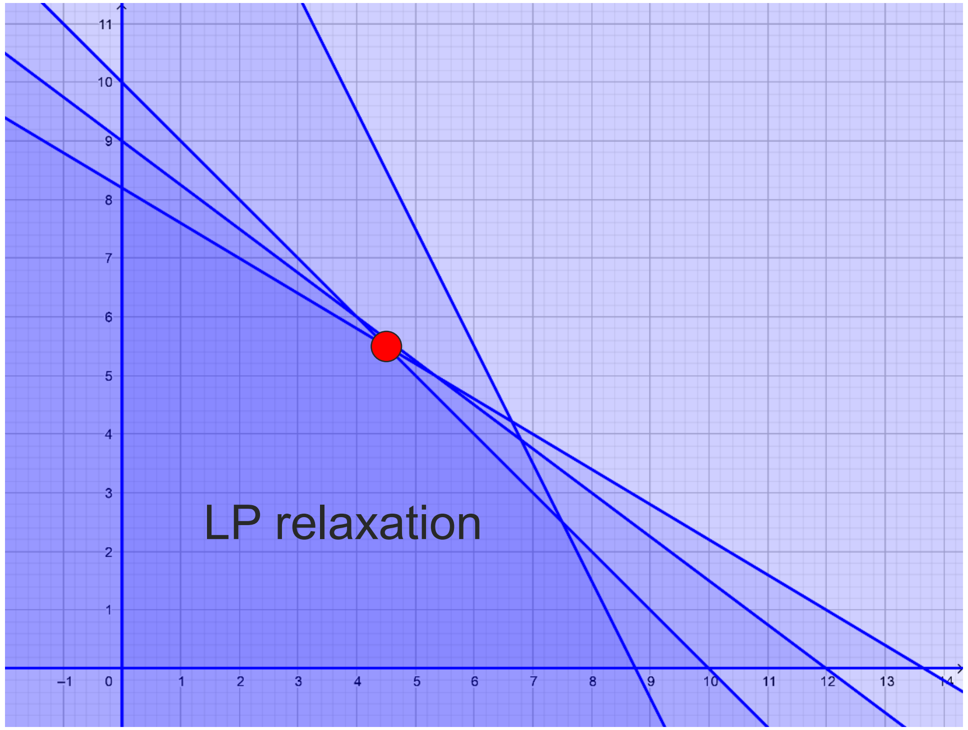
$$\text{integer } x, y \geq 0$$

Can we do better?

- A MIPGap of 0 means the incumbent solution is optimal
  - This happens when the bound and the incumbent have equal value
- Our goal: try to improve the bound and/or incumbent

Upper bound from relaxation

GUROBI OPTIMIZATION

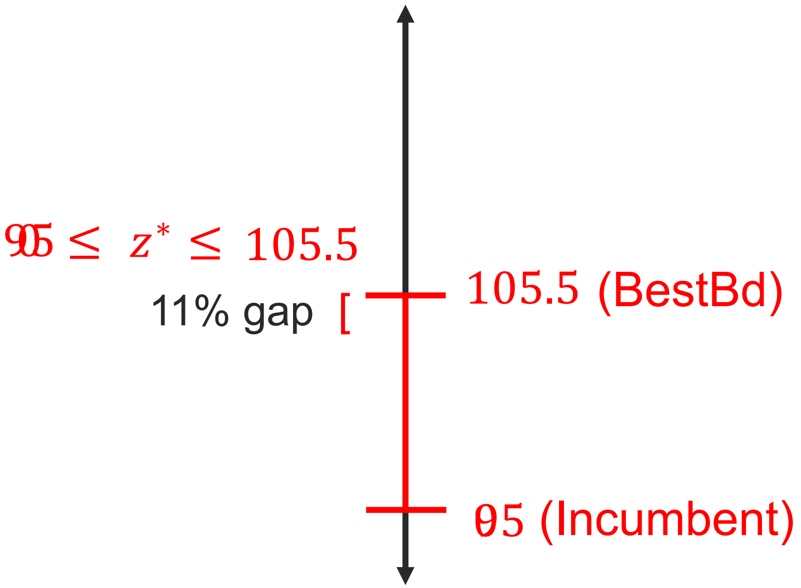


$$
\begin{array}{rrcrcr}
z^{LP} := \max\limits_{x,y} & \color{red}{10x} & \color{red}{+} & \color{red}{11y} & & \\
\text{s.t.} & x & + & y & \leq & 10 \\
& 3x & + & 4y & \leq & 36 \\
& 3x & + & 5y & \leq & 41 \\
& 4x & + & 2y & \leq & 35 \\
& & & \cancel{\text{integer}}\ x, y & \geq & 0
\end{array}
$$

Can we improve our upper bound?

- The integrality constraints are hard…let's temporarily ignore them
- *LP (root) relaxation*: relax (remove) integrality restrictions
- Optimal solution is $(4.5, 5.5)$
  $\Longrightarrow z^{\mathrm{LP}} = f(4.5,5.5) = 105.5$ is an upper bound on z^*!

Lower bound from heuristics



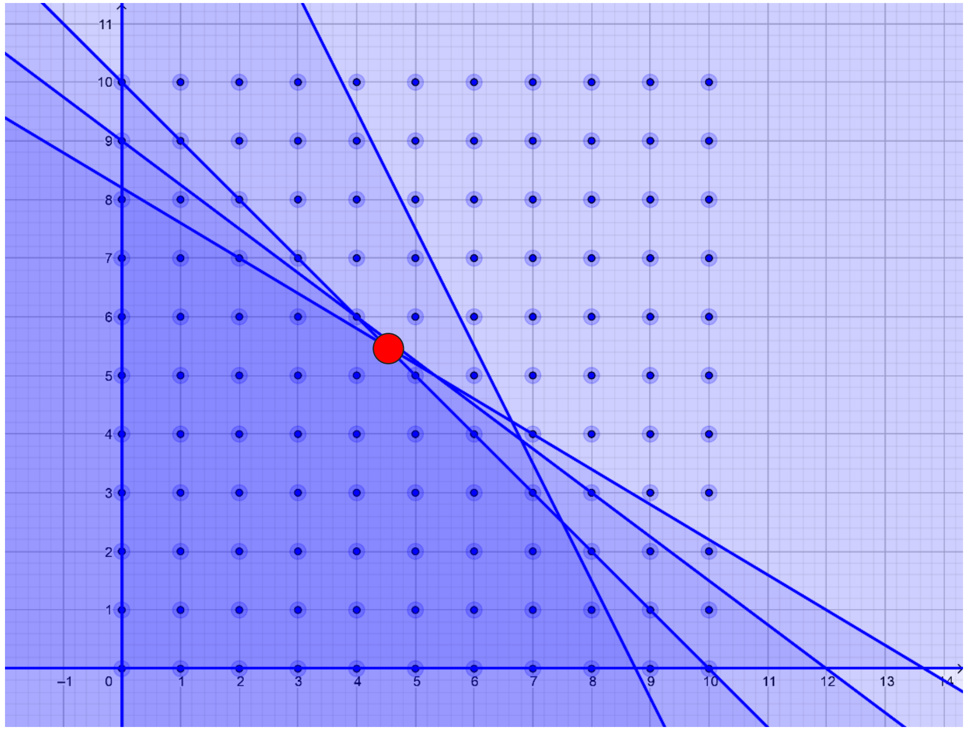
$$
\begin{array}{rrcrcl}
z^* := \max\limits_{x,y} & 10x & + & 11y & & \\
\text{s.t.} & x & + & y & \leq & 10 \\
 & 3x & + & 4y & \leq & 36 \\
 & 3x & + & 5y & \leq & 41 \\
 & 4x & + & 2y & \leq & 35 \\
 & & & \text{integer } x, y & \geq & 0
\end{array}
$$

Can we improve our incumbent?

- Let's "guess" at a good solution
  - This is commonly called a "heuristic"
- Try: round down the relaxed solution $(4.5, 5.5)$:

  $\Longrightarrow (\lfloor 4.5 \rfloor,\ \lfloor 5.5 \rfloor) = (4, 5)$ is feasible!
- $f(4, 5)\ =\ 95$…we have improved our incumbent!

Cuts



$$
\begin{array}{rrcrcrcr}
z^* := \max\limits_{x,y} & 10x & + & 11y & & \\
\text{s.t.} & x & + & y & \leq & 10 \\
& 3x & + & 4y & \leq & 36 \\
& 3x & + & 5y & \leq & 41 \\
& 4x & + & 2y & \leq & 35
\end{array}
$$

integer $x, y \geq 0$

LP optimum:
$x = 4.5, y = 5.5$

Fractional solutions are bad!

- Is there an inequality for our MIP that "cuts off" the LP optimum $(4.5, 5.5)$?
- *Cuts*: inequalities that tighten our relaxation of the MIP feasible region
- Let's try to derive a cut that is violated by $(4.5, 5.5)$!

Cuts

$$\max_{x, y, s_1, s_2, s_3, s_4} \quad 10x + 11y$$

$$\max_{x,y} \quad 10x + 11y$$

s. t.

$$\begin{aligned} x &+ y \le 10 \\ 3x &+ 4y \le 36 \\ 3x &+ 5y \le 41 \\ 4x &+ 2y \le 35 \end{aligned}$$

$$\begin{aligned} x &+ y + s_1 = 10 \\ 3x &+ 4y + s_2 = 36 \\ 3x &+ 5y + s_3 = 41 \\ 4x &+ 2y + s_4 = 35 \end{aligned}$$

$s_1, s_2, s_3, s_4, x, y \ge 0$

integer $x, y \ge 0$

Satisfied without slack at $(4.5, 5.5)$

LP optimum:
$x = 4.5, y = 5.5$

1. Aggregate

$$-3x - 3y - 3s_1 = -30$$

and

$$3x + 5y + s_3 = 41$$

$$\Downarrow$$

$$2y - 3s_1 + s_3 = 11$$

$$\Downarrow$$

$$y - \frac{3}{2}s_1 + \frac{1}{2}s_3 = \frac{11}{2}$$

2. Weaken

$$y - \frac{3}{2}s_1 + \frac{1}{2}s_3 = \frac{11}{2}$$

$$\Downarrow$$

$$y + \left\lfloor -\frac{3}{2} \right\rfloor s_1 + \left\lfloor \frac{1}{2} \right\rfloor s_3 \le \frac{11}{2}$$

$$y - 2(10 - x - y) \le 5$$

$$\Downarrow$$

$$y - 2s_1 \le \frac{11}{2}$$

$$2x + 3y \le 25$$

3. Strengthen

$$y - 2s_1 \le \frac{11}{2}$$

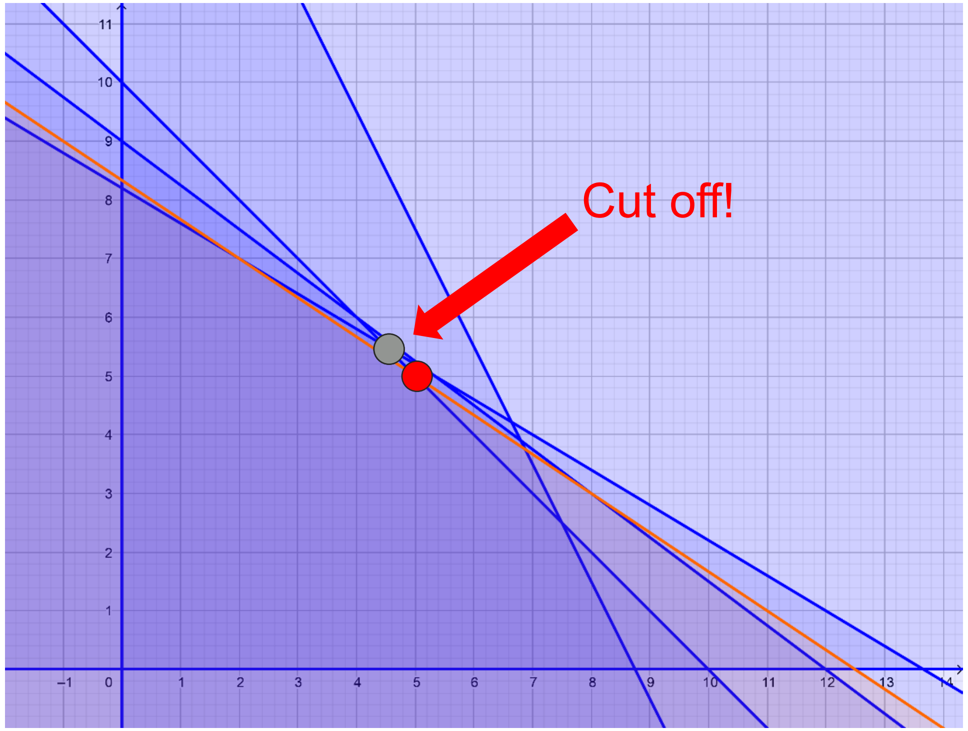
$$\Downarrow$$

$$y - 2s_1 \le \left\lfloor \frac{11}{2} \right\rfloor = 5$$

New inequality!

Cuts

GUROBI OPTIMIZATION



LP optimum:

$x = 4.5, y = 5.5$

$$
\begin{array}{rrcrcrcr}
z^* := \max\limits_{x,y} & 10x & + & 11y & & \\
\text{s.t.} & x & + & y & \leq & 10 \\
& 3x & + & 4y & \leq & 36 \\
& 3x & + & 5y & \leq & 41 \\
& 4x & + & 2y & \leq & 35 \\
& 3x & + & 2y & \leq & 25 \\
& & & \text{integer } x, y & \geq & 0
\end{array}
$$

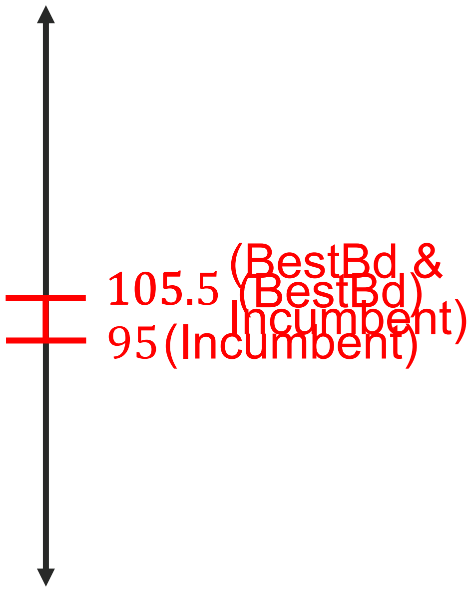
Is this new inequality any good?

- It cuts off the previous optimal solution!
- Solving the new, tightened relaxation yields a new upper bound
- New upper bound on z^* is $10(5) + 11(5) = 105$

Improving bounds

$105 \leq z^* \leq 105.5$

Gap=0%

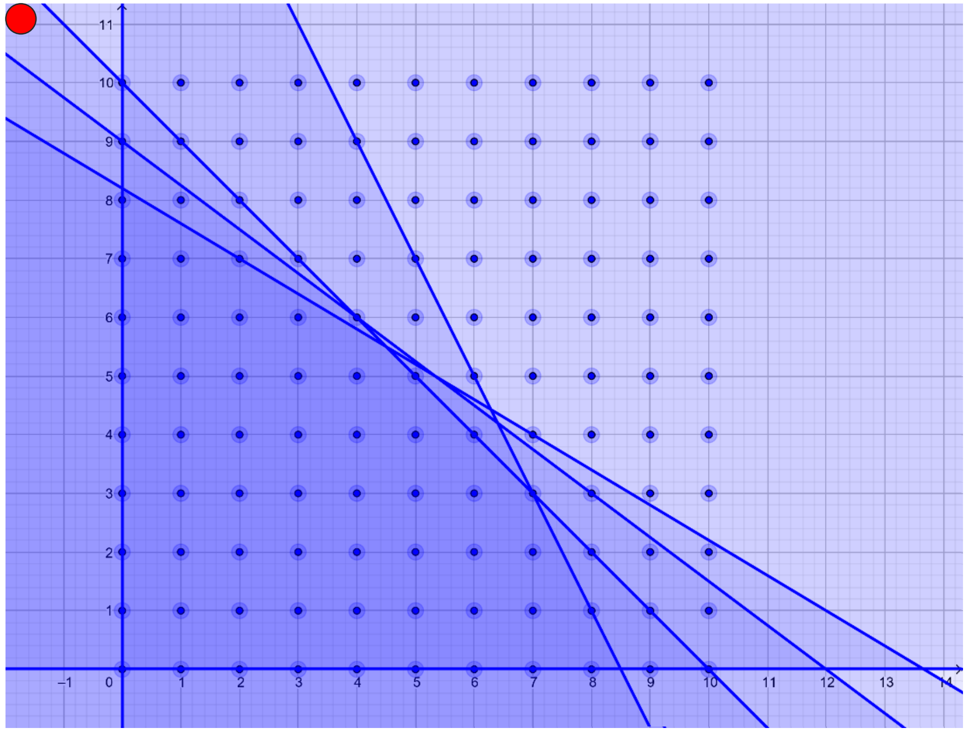


$$
\begin{array}{rrcrcr}
z^* := \max_{x,y} & 10x & + & 11y & & \\
\text{s.t.} & x & + & y & \leq & 10 \\
 & 3x & + & 4y & \leq & 36 \\
 & 3x & + & 5y & \leq & 41 \\
 & 4x & + & 2y & \leq & 35 \\
 & & \text{integer} & x, y & \geq & 0
\end{array}
$$

What can we say about z^*?

- New upper bound: $z^* \leq 105$
- But the optimal solution to the relaxation is $(5, 5)$
    - This is feasible to the original problem!
    - 105 is a new incumbent objective value
- **We have solved the problem: $z^* = 105$**

Presolve

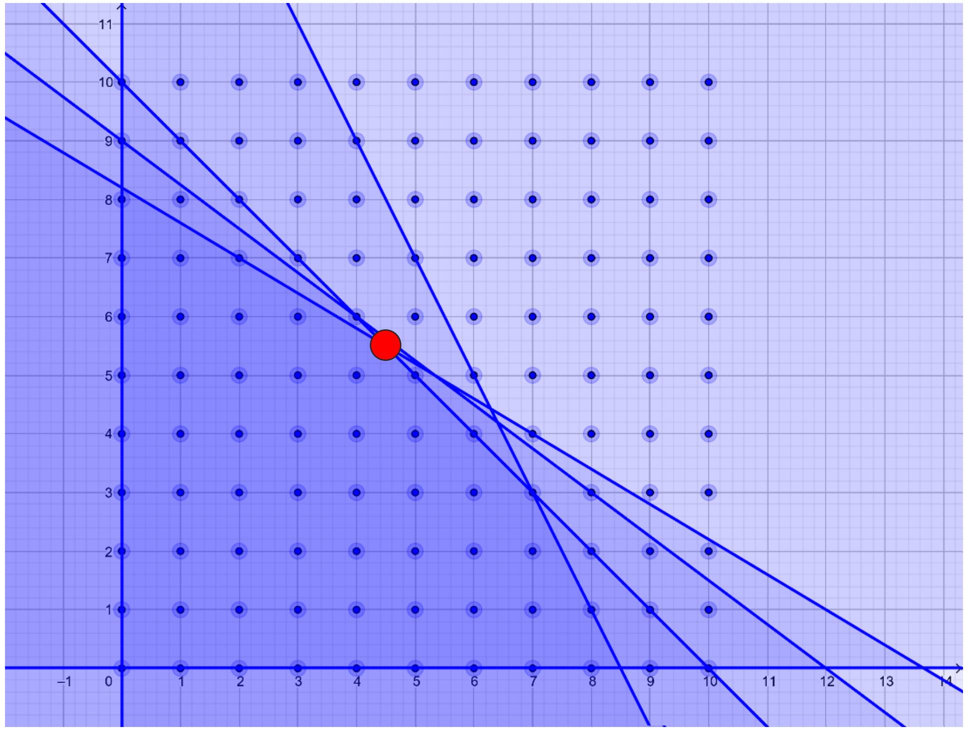


$$
\begin{array}{rlrcrcr}
z^* := \max\limits_{x,y} & 10x & + & 11y & & \\
\text{s.t.} & x & + & y & \leq & 10 \\
 & 3x & + & 4y & \leq & 36 \\
 & 3x & + & 5y & \leq & 41 \\
 & 4x & + & 2y & \leq & 35 \\
 & & & \text{integer } x, y & \geq & 0
\end{array}
$$

Can we tighten the formulation just by looking at it?

- *Tighten*: reduce the feasible region of relaxation without removing any feasible points
- $4x$ is even, and so is $2y$…
- This doesn't change optimal solution relaxation

Branch-and-bound

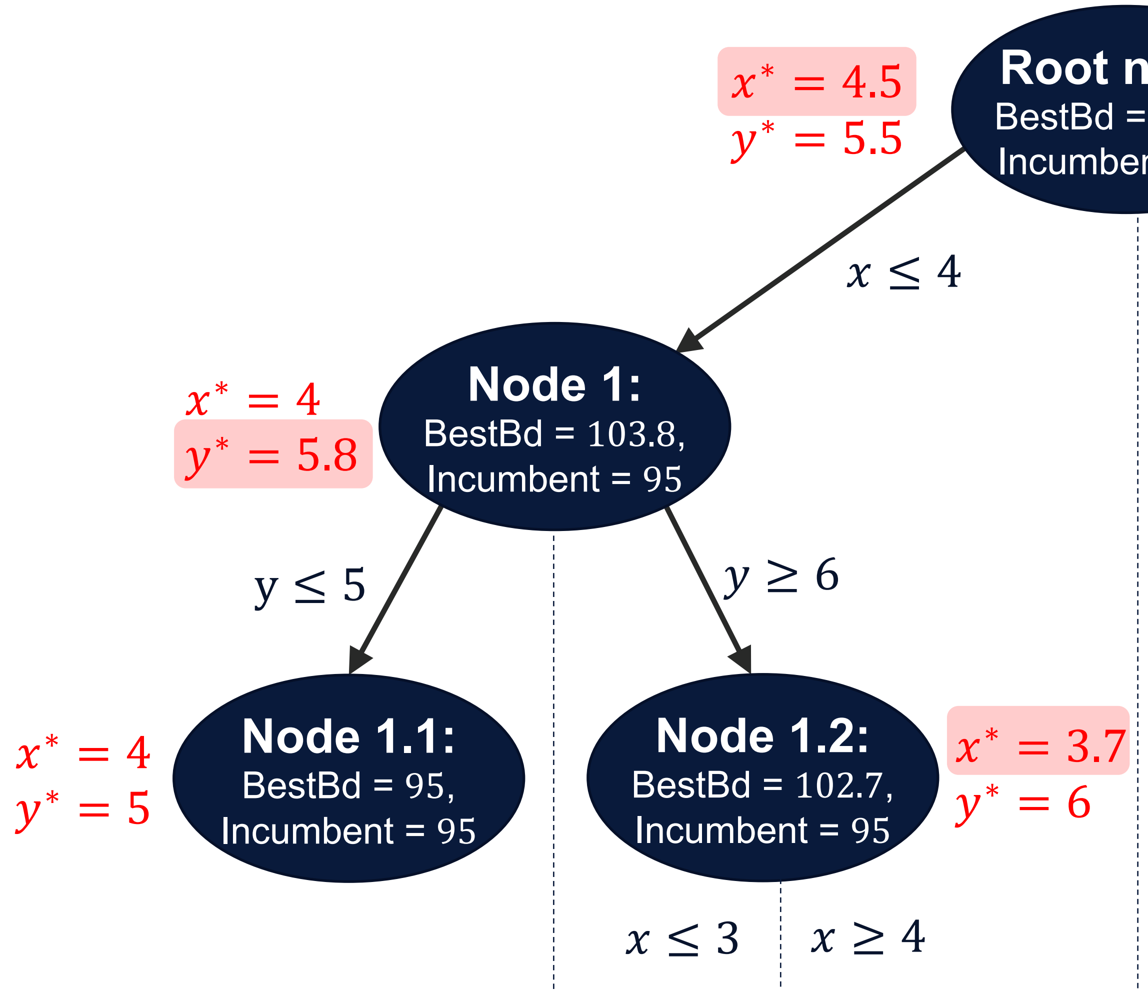


$$
\begin{array}{rrcrcr}
z^* := \max_{x,y} & 10x & + & 11y & & \\
\text{s.t.} & x & + & y & \leq & 10 \\
& 3x & + & 4y & \leq & 36 \\
& 3x & + & 5y & \leq & 41 \\
& 4x & + & 2y & \leq & 34 \\
& & & \text{integer } x, y & \geq & 0
\end{array}
$$

Can we combine these ideas with brute-force enumeration?

- Yes! For example, consider the relaxation solution $(x, y) = (4.5, 5.5)$
  - x has to be integer. So is $x \leq 4$ or $x \geq 5$?
  - Let's investigate!

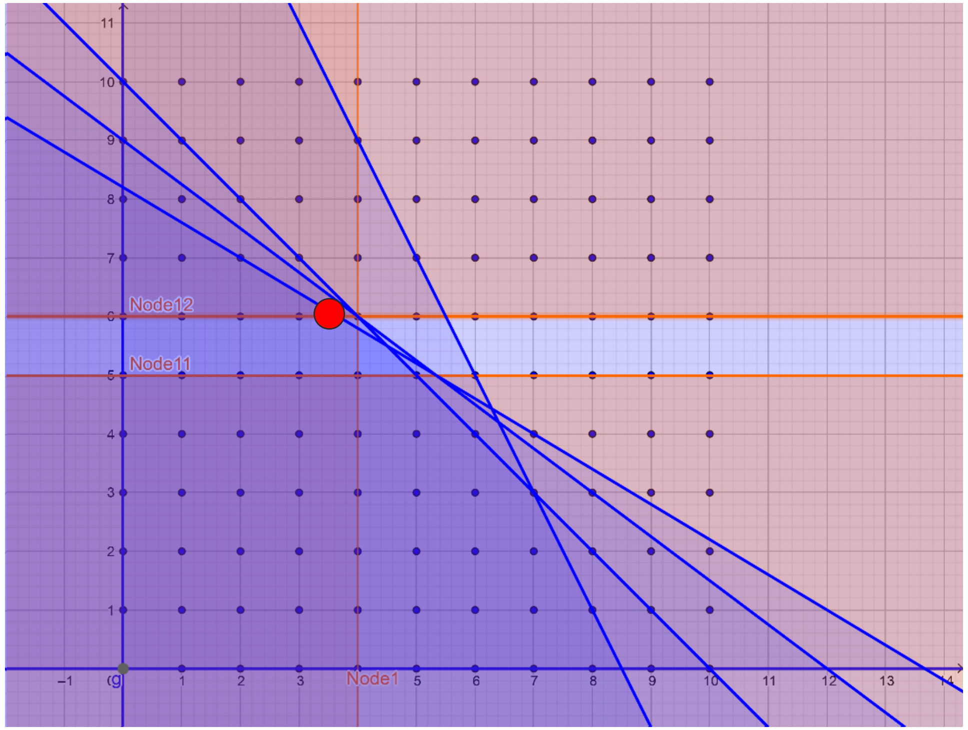
Branch-and-bound tree



$$
\begin{array}{rrcrcr}
z^* := \max\limits_{x,y} & 10x & + & 11y & & \\
\text{s.t.} & x & + & y & \leq & 10 \\
 & 3x & + & 4y & \leq & 36 \\
 & 3x & + & 5y & \leq & 41 \\
 & 4x & + & 2y & \leq & 34 \\
 & & & \text{integer } x, y & \geq & 0 \\
 & & & & & x \leq 4 \\
 & & & & & y \geq 6
\end{array}
$$

Branch-and-bound tree

GUROBI
OPTIMIZATION



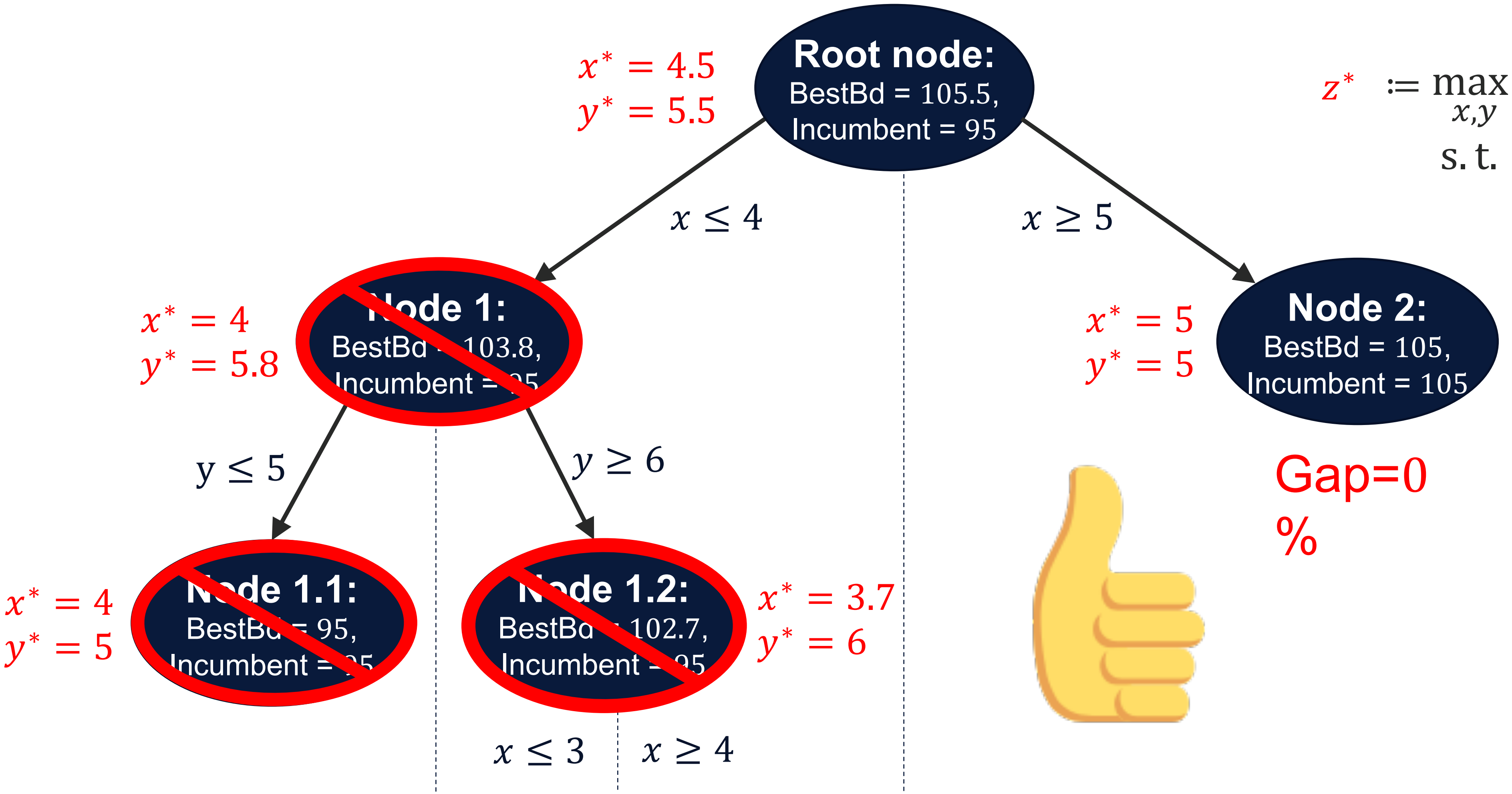
Geometrically, what does this look like?

$$
\begin{array}{llrcrcr}
\color{red}{z^*} & := \max\limits_{x,y} & \color{red}{10x} & \color{red}{+} & \color{red}{11y} & & \\
& \text{s.t.} & x & + & y & \leq & 10 \\
& & 3x & + & 4y & \leq & 36 \\
& & 3x & + & 5y & \leq & 41 \\
& & 4x & + & 2y & \leq & 34 \\
& & & & \text{integer } x, y & \geq & 0
\end{array}
$$

$x \leq 4$

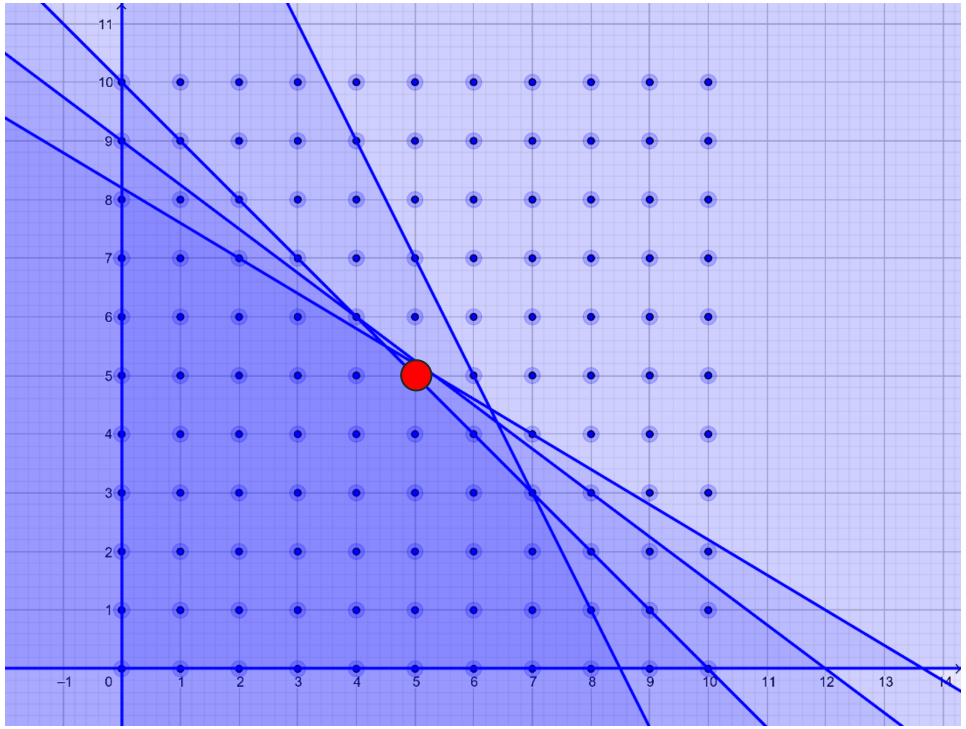
$y \geq 6$

Branch-and-bound tree



$$
\begin{array}{rrcrcll}
z^* := \max\limits_{x,y} & 10x & + & 11y & & \\
\text{s.t.} & x & + & y & \leq & 10 \\
& 3x & + & 4y & \leq & 36 \\
& 3x & + & 5y & \leq & 41 \\
& 4x & + & 2y & \leq & 34 \\
& & & \text{integer } x, y & \geq & 0 \\
& & & & & x \geq 5
\end{array}
$$

Branch-and-bound



$$
\begin{array}{rlrcrcr}
z^* := & \max_{x,y} & 10x & + & 11y & & \\
& \text{s.t.} & x & + & y & \leq & 10 \\
& & 3x & + & 4y & \leq & 36 \\
& & 3x & + & 5y & \leq & 41 \\
& & 4x & + & 2y & \leq & 34 \\
& & & & \text{integer } x, y & \geq & 0
\end{array}
$$

Looks like a lot of work!

- For the computer – not for you!
- Gurobi does presolve, cuts, heuristics, branch-and-bound and more “auto-magically”!

Termination criteria

- MIP gap
  - E.g., terminate if *relative* MIP gap $\leq$ 1% or *absolute* MIP gap $\leq$ 10,
- Time spent solving
  - E.g., terminate after 3600 seconds have elapsed
- Number of nodes explored
  - E.g., terminate after solver visits 100,000 nodes
- Number of feasible solutions found
- …and more!

See the Termination section of the **Parameters** page in the Reference Manual

Gurobi logs

Recognize these ideas in the Gurobi logs

Log file structure

LP

1. User input + model statistics
2. Presolve
3. Simplex or Barrier iterations
4. Termination statistics

MILP

1. User input + model statistics
2. Presolve
3. Root node relaxation
4. Branch-and-bound tree exploration
5. Termination statistics

Simplex log

```
Iteration    Objective       Primal Inf.    Dual Inf.      Time
       0    1.7748600e+04   6.627132e+03   0.000000e+00      0s
    9643    1.1574611e+07   1.418653e+03   0.000000e+00      5s
   14440    1.1607748e+07   4.793500e+00   0.000000e+00     10s
   15213    1.1266396e+07   0.000000e+00   0.000000e+00     11s

Solved in 15213 iterations and 10.86 seconds
Optimal objective  1.126639605e+07
```

Barrier log

```
Barrier statistics:
 AA' NZ      : 2.836e+03
 Factor NZ   : 3.551e+03 (roughly 40 MBytes of memory)
 Factor Ops  : 1.739e+05 (less than 1 second per iteration)
 Threads     : 4

                Objective                Residual
Iter       Primal          Dual         Primal    Dual     Compl     Time
   0   1.30273209e+06  0.00000000e+00  5.90e+02 0.00e+00  7.32e+00   12s
   1   1.04326180e+05 -5.84079103e+02  4.84e+01 1.69e+00  5.95e-01   12s
   2   9.46325157e+03 -4.40392705e+02  2.92e+00 1.35e+00  5.46e-02   12s
   3   3.66683689e+03  9.27381244e+02  1.94e-01 5.35e-01  1.41e-02   12s
   4   3.37449982e+03  1.79938013e+03  1.29e-01 2.41e-01  7.64e-03   12s
   5   3.13244138e+03  1.90266941e+03  8.89e-02 2.07e-01  6.00e-03   12s
   6   2.71282610e+03  2.11401255e+03  3.20e-02 1.15e-01  2.96e-03   12s
   7   2.48856811e+03  2.18107490e+03  1.06e-02 7.26e-02  1.56e-03   12s
   8   2.35427593e+03  2.21183615e+03  3.20e-03 4.52e-02  7.36e-04   12s
   9   2.30239737e+03  2.22464753e+03  1.53e-03 2.38e-02  4.03e-04   12s
  10   2.25547118e+03  2.23096162e+03  3.00e-04 1.40e-02  1.30e-04   12s
  11   2.24052450e+03  2.23917612e+03  4.10e-06 6.33e-04  7.20e-06   12s
  12   2.23967243e+03  2.23966346e+03  2.01e-08 5.01e-06  4.82e-08   12s
  13   2.23966667e+03  2.23966666e+03  1.11e-10 1.14e-13  4.81e-11   13s

Barrier solved model in 13 iterations and 12.51 seconds
Optimal objective 2.23966667e+03
```

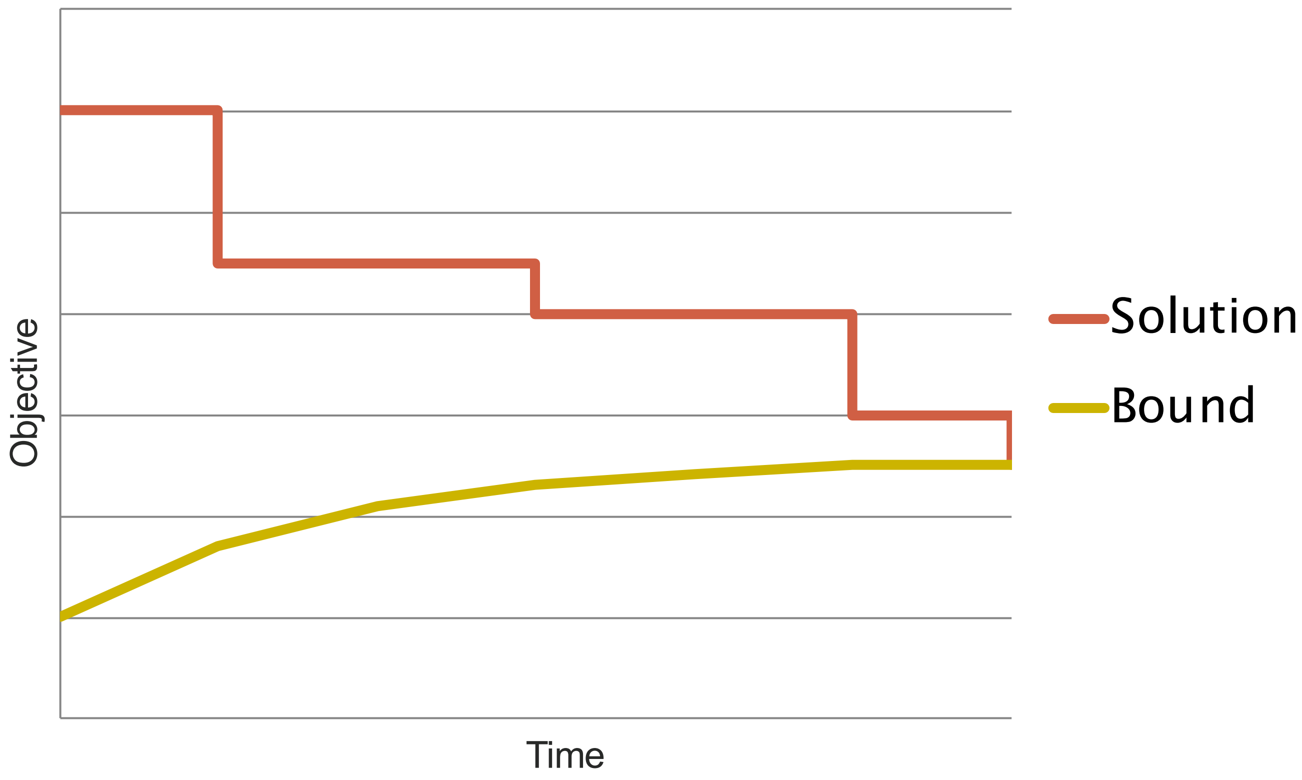
Barrier crossover log

```
Barrier solved model in 13 iterations and 12.51 seconds
Optimal objective 2.23966667e+03


Root crossover log...

      40 DPushes remaining with DInf 0.0000000e+00                13s
       0 DPushes remaining with DInf 7.8159701e-14                13s

    1176 PPushes remaining with PInf 0.0000000e+00                13s
       0 PPushes remaining with PInf 0.0000000e+00                13s

  Push phase complete: Pinf 0.0000000e+00, Dinf 1.2079227e-13     13s


Root simplex log...

Iteration    Objective       Primal Inf.    Dual Inf.      Time
    1219    2.2396667e+03   0.000000e+00   0.000000e+00     13s
    1219    2.2396667e+03   0.000000e+00   0.000000e+00     13s

Root relaxation: objective 2.239667e+03, 1219 iterations, 0.43 seconds
```

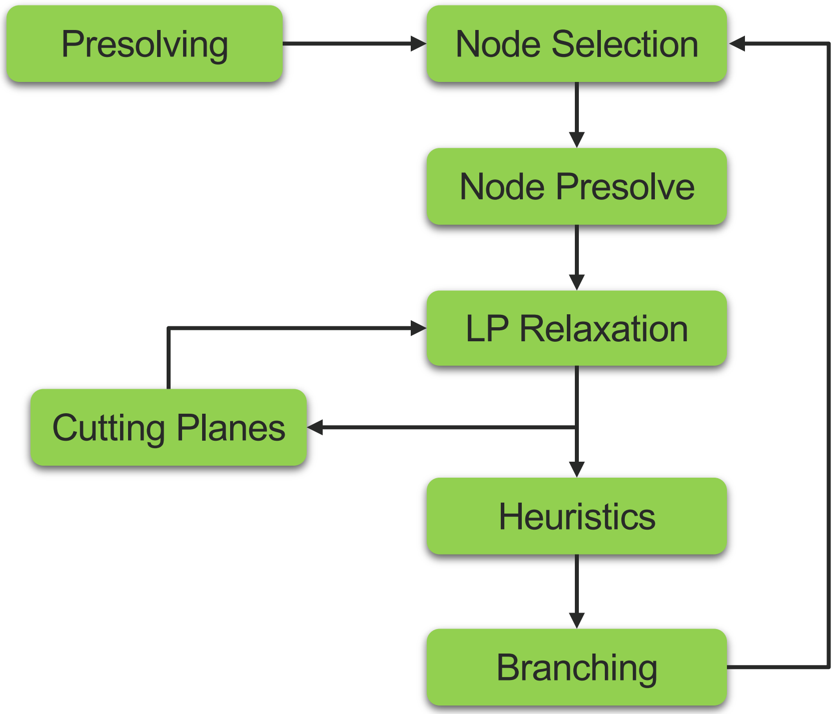
MIP tree exploration

```
    Nodes    |    Current Node    |     Objective Bounds      |     Work
 Expl Unexpl |  Obj  Depth IntInf | Incumbent    BestBd   Gap | It/Node Time


     0     0 8.0000e+08    0   72 3.1334e+09 8.0000e+08  74.5%     -    0s
H    0     0                    2.400019e+09 8.0000e+08  66.7%     -    0s
     0     0 8.0000e+08    0   72 2.4000e+09 8.0000e+08  66.7%     -    0s
H    0     0                    2.066683e+09 8.0000e+08  61.3%     -    0s
     0     0 8.0000e+08    0   72 2.0667e+09 8.0000e+08  61.3%     -    0s
     0     0 8.0000e+08    0   72 2.0667e+09 8.0000e+08  61.3%     -    0s
     0     2 8.0000e+08    0   72 2.0667e+09 8.0000e+08  61.3%     -    0s
H  796   711                    2.050016e+09 8.0000e+08  61.0%   4.4    0s
H  796   675                    2.050016e+09 8.0000e+08  61.0%   4.4    0s
* 3353  2003             109    2.000021e+09 8.0000e+08  60.0%   4.5    0s
H 5097  3281                    2.000019e+09 8.0000e+08  60.0%   4.1    0s
H 5571  3544                    1.950018e+09 8.0000e+08  59.0%   4.0    0s
* 6228  4017              82    1.940019e+09 8.0000e+08  58.8%   3.9    1s
* 6347  4038              97    1.925019e+09 8.0000e+08  58.4%   3.9    1s
H 6422  3912                    1.833347e+09 8.0000e+08  56.4%   3.9    1s
* 7999  4981             104    1.816683e+09 8.0000e+08  56.0%   3.8    1s
H 9958  6572                    1.800016e+09 8.0000e+08  55.6%   3.8    1s
```

Solving a MIP model



Branch-and-cut



Branch-and-cut

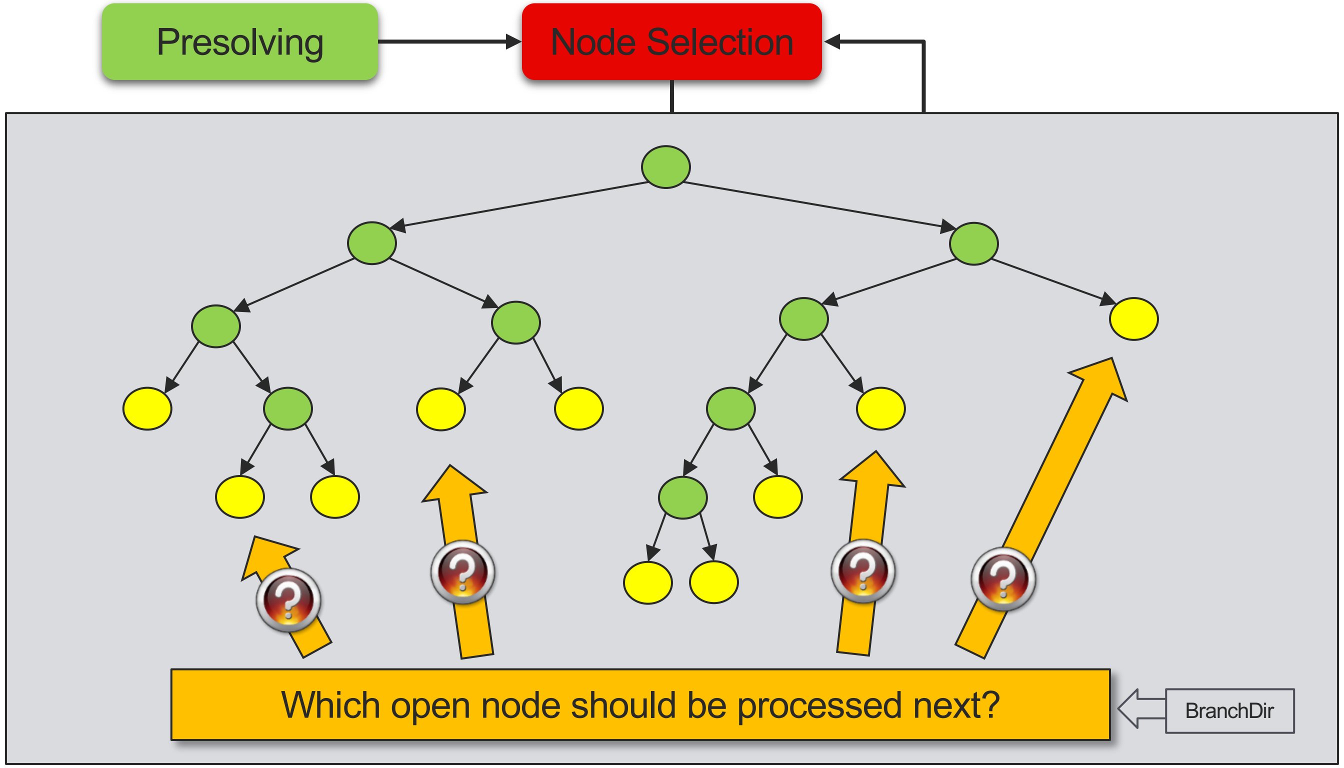
Presolving

Cutting Planes

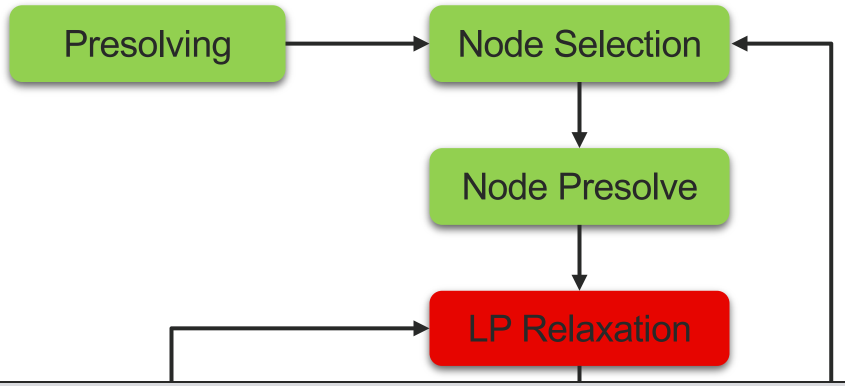
Branching

```
Gurobi Optimizer version 6.0.0 (linux64)
Copyright (c) 2014, Gurobi Optimization, Inc.

Read MPS format model from file /models/mip/roll3000.mps.bz2
Reading time = 0.03 seconds
roll3000: 2295 rows, 1166 columns, 29386 nonzeros
Optimize a model with 2295 rows, 1166 columns and 29386 nonzeros
Coefficient statistics:
  Matrix range    [2e-01, 3e+02]
  Objective range [1e+00, 1e+00]
  Bounds range    [1e+00, 1e+09]
  RHS range       [6e-01, 1e+03]
Presolve removed 1308 rows and 311 columns
Presolve time: 0.08s
Presolved: 987 rows, 855 columns, 19346 nonzeros
Variable types: 211 continuous, 644 integer (545 binary)

Root relaxation: objective 1.112003e+04, 1063 iterations, 0.03 seconds

    Nodes    |    Current Node    |     Objective Bounds      |     Work
 Expl Unexpl |  Obj  Depth IntInf | Incumbent    BestBd   Gap | It/Node Time

     0     0 11120.0279    0  154          - 11120.0279      -     -    0s
     0     0 11526.8918    0  207          - 11526.8918      -     -    0s
     0     0 11896.9710    0  190          - 11896.9710      -     -    0s
```

Branch-and-cut



Branch-and-cut




```
Presolved: 987 rows, 855 columns, 19346 nonzeros
Variable types: 211 continuous, 644 integer (545 binary)

Root relaxation: objective 1.112003e+04, 1063 iterations, 0.03 seconds

    Nodes    |    Current Node    |     Objective Bounds      |     Work
 Expl Unexpl |  Obj  Depth IntInf | Incumbent    BestBd   Gap | It/Node Time

     0     0 11120.0279    0  154          - 11120.0279      -     -    0s
     0     0 11526.8918    0  207          - 11526.8918      -     -    0s
     0     0 11896.9710    0  190          - 11896.9710      -     -    0s
...
H  327   218                    13135.000000 12455.2162  5.18%  42.6    1s
H  380   264                    13093.000000 12455.2162  4.87%  41.6    1s
H  413   286                    13087.000000 12455.2162  4.83%  41.4    1s
  1066   702 12956.2676   31  192 13087.0000 12629.5426  3.50%  37.2    5s
  1097   724 12671.8285    8  147 13087.0000 12671.8285  3.17%  41.6   10s
H 1121   702                    12890.000000 12689.9964  1.55%  40.7   13s
  1135   710 12732.5601   32  126 12890.0000 12727.1362  1.26%  44.6   15s
```

Branch-and-cut

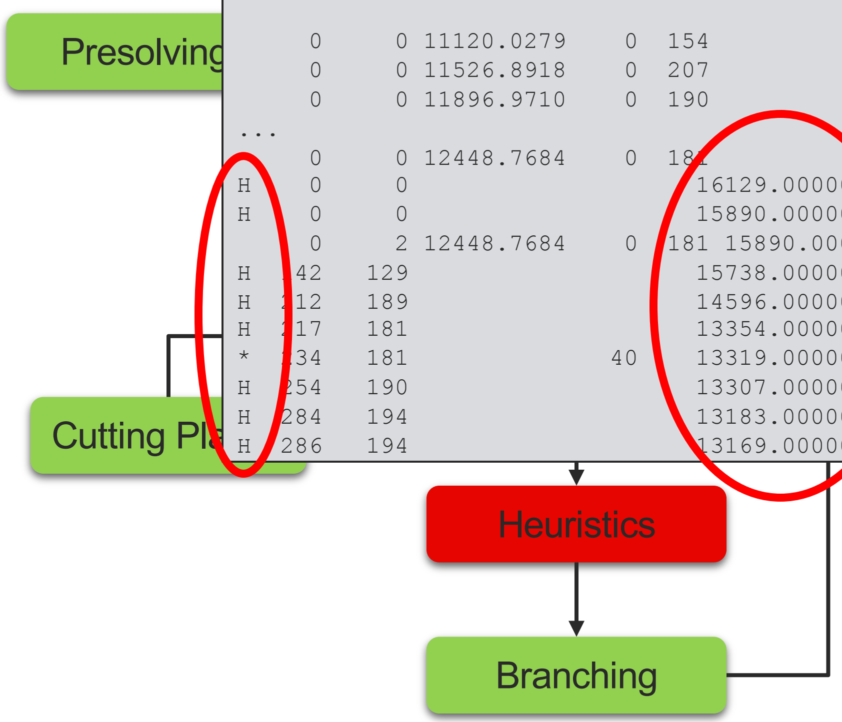
Presolving

Cutting Planes

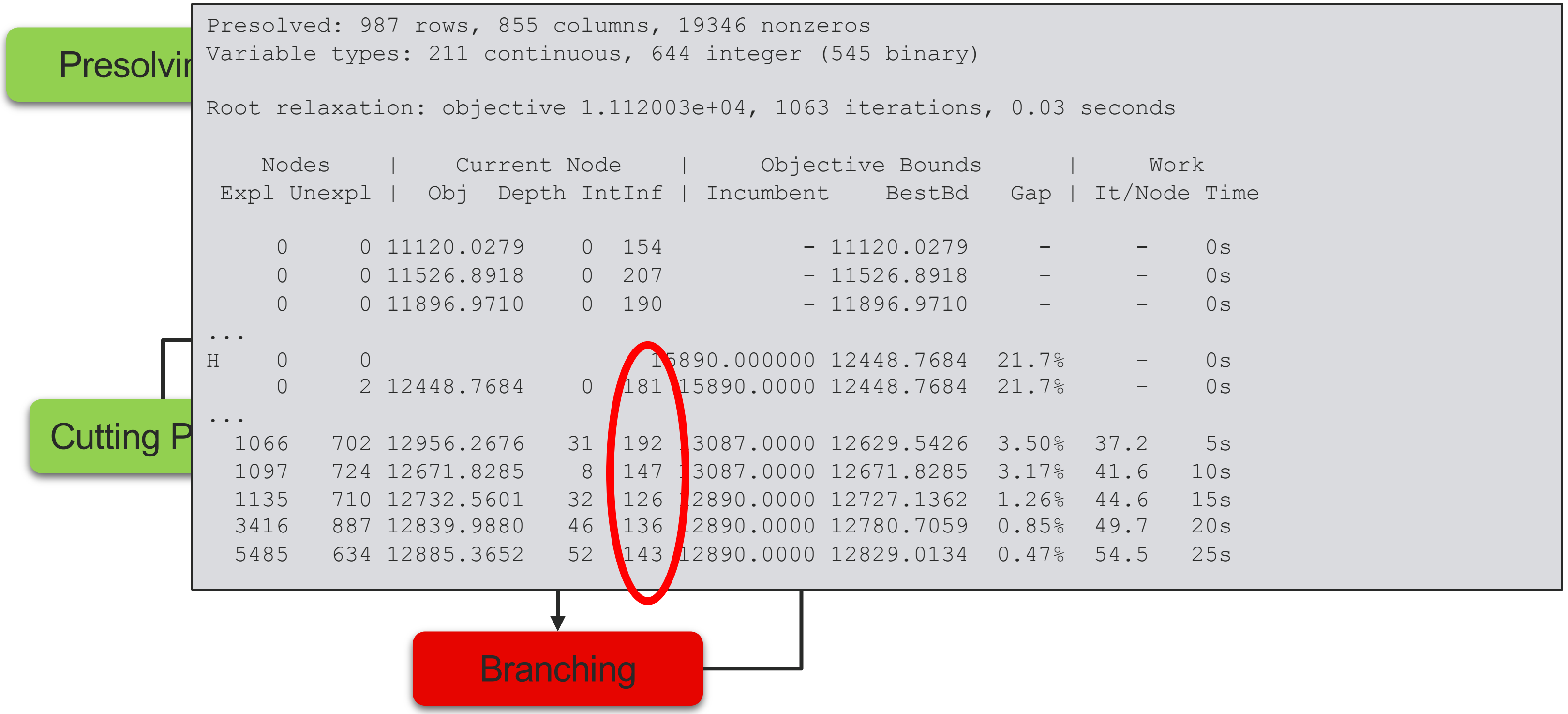
```
Presolved: 987 rows, 855 columns, 19346 nonzeros
Variable types: 211 continuous, 644 integer (545 binary)

Root relaxation: objective 1.112003e+04, 1063 iterations, 0.03 seconds

    Nodes    |    Current Node    |     Objective Bounds      |     Work
 Expl Unexpl |  Obj  Depth IntInf | Incumbent    BestBd   Gap | It/Node Time

     0     0 11120.0279    0  154          - 11120.0279      -     -    0s
     0     0 11526.8918    0  207          - 11526.8918      -     -    0s
     0     0 11896.9710    0  190          - 11896.9710      -     -    0s
     0     0 12151.4022    0  190          - 12151.4022      -     -    0s
     0     0 12278.3391    0  208          - 12278.3391      -     -    0s
...
  5485   634 12885.3652   52  143 12890.0000 12829.0134  0.47%  54.5   25s

Cutting planes:
  Learned: 4
  Gomory: 46
  Cover: 39
  Implied bound: 8
  Clique: 2
  MIR: 112
  Flow cover: 27
  GUB cover: 11
  Zero half: 91

Explored 6008 nodes (357915 simplex iterations) in 27.17 seconds
Thread count was 4 (of 8 available processors)
```

Branch-and-cut

```
Presolved: 987 rows, 855 columns, 19346 nonzeros
Variable types: 211 continuous, 644 integer (545 binary)

Root relaxation: objective 1.112003e+04, 1063 iterations, 0.03 seconds

    Nodes    |    Current Node    |     Objective Bounds      |     Work
 Expl Unexpl |  Obj  Depth IntInf | Incumbent    BestBd   Gap | It/Node Time

     0     0 11120.0279    0  154          - 11120.0279      -     -    0s
     0     0 11526.8918    0  207          - 11526.8918      -     -    0s
     0     0 11896.9710    0  190          - 11896.9710      -     -    0s
...
     0     0 12448.7684    0  181          - 12448.7684      -     -    0s
H    0     0                    16129.000000 12448.7684  22.8%     -    0s
H    0     0                    15890.000000 12448.7684  21.7%     -    0s
     0     2 12448.7684    0  181 15890.0000 12448.7684  21.7%     -    0s
H  142   129                    15738.000000 12450.7195  20.9%  43.8    1s
H  212   189                    14596.000000 12453.8870  14.7%  42.3    1s
H  217   181                    13354.000000 12453.8870  6.74%  42.6    1s
*  234   181              40    13319.000000 12453.8870  6.50%  42.1    1s
H  254   190                    13307.000000 12453.8870  6.41%  41.3    1s
H  284   194                    13183.000000 12453.8870  5.53%  42.6    1s
H  286   194                    13169.000000 12453.8870  5.43%  42.7    1s
```



Branch-and-cut




```
Presolved: 987 rows, 855 columns, 19346 nonzeros
Variable types: 211 continuous, 644 integer (545 binary)

Root relaxation: objective 1.112003e+04, 1063 iterations, 0.03 seconds

    Nodes    |    Current Node    |     Objective Bounds      |     Work
 Expl Unexpl |  Obj  Depth IntInf | Incumbent    BestBd   Gap | It/Node Time

     0     0 11120.0279    0  154          - 11120.0279      -     -    0s
     0     0 11526.8918    0  207          - 11526.8918      -     -    0s
     0     0 11896.9710    0  190          - 11896.9710      -     -    0s
...
H    0     0                    15890.000000 12448.7684  21.7%     -    0s
     0     2 12448.7684    0  181 15890.0000 12448.7684  21.7%     -    0s
...
  1066   702 12956.2676   31  192 13087.0000 12629.5426  3.50%  37.2    5s
  1097   724 12671.8285    8  147 13087.0000 12671.8285  3.17%  41.6   10s
  1135   710 12732.5601   32  126 12890.0000 12727.1362  1.26%  44.6   15s
  3416   887 12839.9880   46  136 12890.0000 12780.7059  0.85%  49.7   20s
  5485   634 12885.3652   52  143 12890.0000 12829.0134  0.47%  54.5   25s
```

Outlook: Algorithmic Performance

- **Solving optimization problems is never fast enough**

- **Great solver performance is the sum of many parts**
  - Model construction speed
  - Selection and configuration of the underlying algorithms
  - Hardware
  - …and a lot more

- **The default settings of Gurobi are carefully selected to perform best on a large variety of model instance**

- **However, there is usually some tuning potential left, e.g.**
  - The model formulation itself
  - More efficient use of the API
  - Parameter changes
  - …and a lot more

Thank you!

GUROBI OPTIMIZATION

The World's Fastest Solver

Your Next Steps

- Try Gurobi 9.0 Now!
  - www.gurobi.com/free-trial
  - Get a 30-day commercial trial license of Gurobi at academic and research licenses are free!
- For questions about Gurobi pricing, please contact sales@gurobi.com or sales@gurobi.de
- **Need help leveraging Optimization for your business?**
  - Contact us at info@gurobi.com
- A recording of this webinar, including the slides, will be available in roughly one week.

This Week!

- **Wednesday, June 24th**
  - Compute Server and Cloud Demo
  - Introduction to Algorithms
  - Advanced Algorithms
  - Ask the Experts: R&D Team
- **Thursday, June 25th**
  - Customer Case Study: KPMG
  - Customer Case Study: NFL
  - Ask the Experts: Fireside chat with Michael North, VP of Planning & Scheduling at NFL