



SAPIENZA
UNIVERSITÀ DI ROMA

Integer Linear Programming

Exercises 3 — MIP solvers

Worked exercises on chapters 9–9, with complete solutions. Each exercise cites the catalogue problem it uses.

Fabio Furini

fabiofurini.github.io

Department of Computer, Control and Management Engineering — Sapienza University of Rome

Online version and downloadable material:
fabiofurini.github.io/integer-linear-programming/
github.com/fabiofurini/integer-linear-programming

Academic year 2026–2027

Contents

IV	MIP solvers	2
9	MIP solvers: what they do	3



Part IV

MIP solvers



MIP solvers: what they do

Exercise 9.1 — Presolve: bound tightening

Consider the following ILP problem with $n = 3$ integer variables, each with the bounds $0 \leq x_j \leq 10$:

$$\begin{aligned} \max \quad & 3x_1 + 2x_2 + 5x_3 \\ & 4x_1 + 3x_2 + 5x_3 \leq 20 \\ & -x_1 + x_2 \leq -1 \\ & -x_2 - x_3 \leq -2 \\ & x_1, x_2, x_3 \in \mathbb{Z}_{\geq 0} \end{aligned}$$

The constraints $x_1 - x_2 \geq 1$ and $x_2 + x_3 \geq 2$ are written in canonical form as the second and third constraints.

Questions

1. Apply bound tightening: for each constraint $\sum_j a_{ij} x_j \leq b_i$ and each variable x_k , use $a_{ik} x_k \leq b_i - \sum_{j \neq k} \min\{a_{ij} x_j\}$ (minimum over the current bounds), round to integers, and repeat until no bound changes.
2. Check the new bounds by enumeration. Are they the smallest possible ones?
3. Compare the optimal value of the LP relaxation before and after bound tightening.

Solution

1. The minimum of $a_{ij} x_j$ is a_{ij} times the lower bound if $a_{ij} > 0$, and times the upper bound if $a_{ij} < 0$. Within each round the bounds are updated as soon as they are found.

- **Round 1.**

- Constraint 1: $4x_1 \leq 20 - (0) \Rightarrow x_1 \leq 5$, so $x_1 \leq 5$ (was 10).
- Constraint 1: $3x_2 \leq 20 - (0) \Rightarrow x_2 \leq \frac{20}{3}$, so $x_2 \leq 6$ (was 10).
- Constraint 1: $5x_3 \leq 20 - (0) \Rightarrow x_3 \leq 4$, so $x_3 \leq 4$ (was 10).

- Constraint 2: $-x_1 \leq -1 - (0) \Rightarrow x_1 \geq 1$, so $x_1 \geq 1$ (was 0).
- Constraint 2: $x_2 \leq -1 - (-5) \Rightarrow x_2 \leq 4$, so $x_2 \leq 4$ (was 6).

• **Round 2.**

- Constraint 1: $5x_3 \leq 20 - (4) \Rightarrow x_3 \leq \frac{16}{5}$, so $x_3 \leq 3$ (was 4).

• **Round 3.** No bound changes: **STOP**.

The tightened bounds are

$$1 \leq x_1 \leq 5, \quad 0 \leq x_2 \leq 4, \quad 0 \leq x_3 \leq 3.$$

2. The ILP problem has 6 feasible solutions, and all satisfy the new bounds: no feasible solution is lost. The smallest and the largest value of each variable over the feasible solutions are

$$1 \leq x_1 \leq 3, \quad 0 \leq x_2 \leq 2, \quad 0 \leq x_3 \leq 3.$$

The propagated bounds are valid but not all of them are the smallest possible: for example $x_1 = 5$ requires $x_2 = x_3 = 0$ by the first constraint, which violates $x_2 + x_3 \geq 2$. Bound tightening looks at one constraint at a time.

3. The LP relaxation with the original bounds has optimal value 19, and with the tightened bounds $\frac{75}{4}$. The data are integer, so the upper bound goes from 19 to 18.

Bound tightening gives $1 \leq x_1 \leq 5, 0 \leq x_2 \leq 4, 0 \leq x_3 \leq 3$ without losing any feasible solution; the upper bound goes from 19 to 18, equal to $z(\text{ILP})$.

Exercise 9.2 — Presolve: coefficient tightening

Consider the constraint with 4 binary variables

$$6x_1 + 5x_2 + 4x_3 + 3x_4 \leq 10, \quad x_1, \dots, x_4 \in \{0, 1\},$$

and the objective function $\max x_1 + x_2 + x_3 + x_4$.

Questions

1. Apply coefficient tightening to the coefficients $a_1, \dots, a_4$, one at a time: the coefficient a_j can be raised to $b - M_j$, where M_j is the largest value of $\sum_{k \neq j} a_k x_k$ over the binary points with $x_j = 1$ that satisfy the (current) constraint.
2. Check that the two constraints have the same binary solutions.
3. Compare the optimal values of the LP relaxations ($0 \leq x_j \leq 1$) of the two constraints with the optimal value of the binary problem.

Solution

1. Raising a_j to $b - M_j$ keeps every binary solution with $x_j = 1$ (its left-hand side is at most $b - M_j + M_j = b$) and does not change the points with $x_j = 0$.
 - a_1 : with $x_1 = 1$ the best point is $(1, 0, 1, 0)$, so $M_1 = 4$ and $b - M_1 = 6$: $a_1 = 6$ does not change.
 - a_2 : with $x_2 = 1$ the best point is $(0, 1, 1, 0)$, so $M_2 = 4$ and a_2 is raised from 5 to $10 - 4 = 6$.
 - a_3 : with $x_3 = 1$ the best point is $(1, 0, 1, 0)$, so $M_3 = 6$ and $b - M_3 = 4$: $a_3 = 4$ does not change.
 - a_4 : with $x_4 = 1$ the best point is $(1, 0, 0, 1)$, so $M_4 = 6$ and a_4 is raised from 3 to $10 - 6 = 4$.

The tightened constraint is

$$6x_1 + 6x_2 + 4x_3 + 4x_4 \leq 10.$$

2. Both constraints are satisfied by the same 10 binary points (of the $2^4 = 16$), namely $(0, 0, 0, 0)$, $(0, 0, 0, 1)$, $(0, 0, 1, 0)$, $(0, 0, 1, 1)$, $(0, 1, 0, 0)$, $(0, 1, 0, 1)$, $(0, 1, 1, 0)$, $(1, 0, 0, 0)$, $(1, 0, 0, 1)$, $(1, 0, 1, 0)$.
3. The LP relaxation of the original constraint has optimal solution $(0, \frac{3}{5}, 1, 1)$ with value $\frac{13}{5}$; the LP relaxation of the tightened constraint has optimal solution $(\frac{1}{3}, 0, 1, 1)$ with value $\frac{7}{3}$. The binary problem has optimal value 2.

Coefficient tightening turns $6x_1 + 5x_2 + 4x_3 + 3x_4 \leq 10$ into $6x_1 + 6x_2 + 4x_3 + 4x_4 \leq 10$: same binary solutions, same size, and the LP bound improves from $\frac{13}{5}$ to $\frac{7}{3}$ (the optimal value is 2).

Exercise 9.3 — Reading a solver log

Gurobi solves a generic ILP problem with $n = 10$ integer variables $x_j \in \{0, \dots, 3\}$ and $m = 3$ constraints, $\max \mathbf{c}'\mathbf{x}$ subject to $\mathbf{A}\mathbf{x} \leq \mathbf{b}$, with integer pseudo-random data (Python `random.Random(4)`): a_{ij} uniform in $\{1, \dots, 20\}$, c_j uniform in $\{5, \dots, 25\}$, and $b_i = \lfloor 3 \sum_j a_{ij} / 3 \rfloor$, with `Presolve = 0`, `Cuts = 0` and `Heuristics = 0` (`Threads = 1`, `Seed = 0`). The log is:

Gurobi log: generic ILP problem, presolve, cuts and heuristics off

```

01| Gurobi Optimizer version 13.0.3 build v13.0.3rc0 (linux64 - "Ubuntu 26.04 LTS")
02|
03| CPU model: Intel(R) Core(TM) i5-14500T, instruction set [SSE2|AVX|AVX2]
04| Thread count: 20 physical cores, 20 logical processors, using up to 1 threads
05|
06| Non-default parameters:
07| Heuristics 0
08| Cuts 0
09| LogToConsole 0
10| Presolve 0
11| Threads 1
12|
13| Optimize a model with 3 rows, 10 columns and 30 nonzeros (Max)
14| Model fingerprint: 0x92726b84
15| Model has 10 linear objective coefficients
16| Variable types: 0 continuous, 10 integer (0 binary)
17| Coefficient statistics:
18|   Matrix range    [1e+00, 2e+01]
19|   Objective range [7e+00, 2e+01]
20|   Bounds range    [3e+00, 3e+00]
21|   RHS range       [8e+01, 1e+02]
22|
23| Variable types: 0 continuous, 10 integer (0 binary)
24|
25| Root relaxation: objective 2.296103e+02, 4 iterations, 0.00 seconds (0.00 work units)
26|
27|      Nodes |      Current Node |      Objective Bounds |      Work
28|  Expl Unexpl |  Obj  Depth IntInf | Incumbent    BestBd   Gap | It/Node Time
29|
30|      0      0  229.61033   0   3         -  229.61033       -   -   0s
31|      0      0  229.30837   0   3         -  229.30837       -   -   0s
32|      0      2  229.30837   0   3         -  229.30837       -   -   0s
33| *  12      6           9  220.0000000  224.56863  2.08%  1.2   0s
34| *  22      3           6  221.0000000  224.53846  1.60%  1.0   0s
35| *  23      1           5  223.0000000  224.53846  0.69%  1.0   0s
36|
37| Explored 25 nodes (30 simplex iterations) in 0.00 seconds (0.00 work units)
38| Thread count was 1 (of 20 available processors)
39|
40| Solution count 3: 223 221 220
41|
42| Optimal solution found (tolerance 1.00e-04)
43| Best objective 2.230000000000e+02, best bound 2.230000000000e+02, gap 0.0000%

```

Questions

1. Is the problem a maximization or a minimization problem? How many variables and constraints does it have?

2. Which line gives the bound of the LP relaxation of the root node? Which value of UB does it give?
3. Which lines report new incumbent solutions, and how are they found? Give the lower bound LB after each of them.
4. Recompute the gap of line 33.
5. How many nodes are explored? Is the final solution certified optimal?

Solution

1. Line 13: (**Max**), so it is a **maximization** problem, with 3 rows (constraints) and 10 columns (variables), all integer and not binary.
2. Line 25: the root relaxation has optimal value 229.61033 (exactly $z(\text{LP}) = \frac{48907}{213}$).
The data are integer, so $\text{UB} = \lfloor \frac{48907}{213} \rfloor = 229$. The column **BestBd** of line 30 reports the same value; at that point no incumbent exists (the dash in **Incumbent**).
3.
 - Line 33 (node 12): $\text{LB} = 220$.
 - Line 34 (node 22): $\text{LB} = 221$.
 - Line 35 (node 23): $\text{LB} = 223$.

The three lines start with *****: each incumbent solution is the integer optimal solution of the LP problem of a node (pruning by integrality); the heuristics are off, so no line starts with **H**.

4. In line 33 we have $\text{Incumbent} = 220$ and $\text{BestBd} = 224.569$, so

$$\text{gap} = \frac{|224.569 - 220|}{220} \approx 2.08\%,$$

as in the column **Gap** (2.08%).

5. Line 37: 25 nodes are explored, with 30 simplex iterations. Line 42: the solution of value 223 is optimal with the default tolerance 10^{-4} ; the last line reports best bound 223 and gap 0.

LB grows from 220 to 223 with integer node solutions, while UB decreases from 229 to 223 by branching: after 25 nodes $\text{LB} = \text{UB} = 223$ and the solution is **certified optimal**.

Exercise 9.4 — Reading a solver log: a minimization problem

Gurobi solves a generic ILP problem with $n = 12$ integer variables $x_j \in \{0, \dots, 3\}$ and $m = 4$ constraints, $\min \mathbf{c}'\mathbf{x}$ subject to $\mathbf{A}\mathbf{x} \geq \mathbf{b}$, with integer pseudo-random data (Python `random.Random(3)`): a_{ij} uniform in $\{1, \dots, 20\}$, c_j uniform in $\{5, \dots, 25\}$, and $b_i = \lfloor \frac{2}{5} \cdot 3 \sum_j a_{ij}/3 \rfloor$, with the default parameters (`Threads = 1`, `Seed = 0`). The log is:

Gurobi log: generic minimization problem, default parameters

```

01| Gurobi Optimizer version 13.0.3 build v13.0.3rc0 (linux64 - "Ubuntu 26.04 LTS")
02|
03| CPU model: Intel(R) Core(TM) i5-14500T, instruction set [SSE2|AVX|AVX2]
04| Thread count: 20 physical cores, 20 logical processors, using up to 1 threads
05|
06| Non-default parameters:
07| LogToConsole 0
08| Threads 1
09|
10| Optimize a model with 4 rows, 12 columns and 48 nonzeros (Min)
11| Model fingerprint: 0x33d6ab84
12| Model has 12 linear objective coefficients
13| Variable types: 0 continuous, 12 integer (0 binary)
14| Coefficient statistics:
15|   Matrix range      [1e+00, 2e+01]
16|   Objective range   [1e+01, 2e+01]
17|   Bounds range      [3e+00, 3e+00]
18|   RHS range         [5e+01, 6e+01]
19|
20| Found heuristic solution: objective 161.0000000
21| Presolve time: 0.00s
22| Presolved: 4 rows, 12 columns, 48 nonzeros
23| Variable types: 0 continuous, 12 integer (0 binary)
24|
25| Root relaxation: objective 6.192991e+01, 5 iterations, 0.00 seconds (0.00 work units)
26|
27|      Nodes      |      Current Node      |      Objective Bounds      |      Work
28|  Expl Unexpl |  Obj  Depth IntInf | Incumbent    BestBd    Gap | It/Node Time
29|
30|      0       0   61.92991    0    3  161.00000   61.92991   61.5%    -    0s
31| H      0       0                   90.0000000   61.92991   31.2%    -    0s
32| H      0       0                   83.0000000   61.92991   25.4%    -    0s
33| H      0       0                   67.0000000   61.92991    7.57%    -    0s
34|      0       0      cutoff    0          67.00000   67.00000   0.00%    -    0s
35|
36| Cutting planes:
37|   Gomory: 1
38|
39| Explored 1 nodes (9 simplex iterations) in 0.00 seconds (0.00 work units)
40| Thread count was 1 (of 20 available processors)
41|
42| Solution count 4: 67 83 90 161
43|
44| Optimal solution found (tolerance 1.00e-04)
45| Best objective 6.7000000000000e+01, best bound 6.7000000000000e+01, gap 0.0000%
```

Questions

1. In this minimization problem, which column gives the upper bound and which one the lower bound?

2. What happens in lines 20, 22 and 25? Give the bounds after line 25, using that the data are integer.
3. Recompute the gap of line 33.
4. Explain lines 34 and 37. Why is the problem solved at the root node?

Solution

1. In a minimization problem the roles are exchanged: **Incumbent** is the value of the best feasible solution, an **upper bound** UB on the optimal value, and **BestBd** is a **lower bound** LB.
2.
 - Line 20: a heuristic finds a feasible solution of value 161 before the presolve, so $UB = 161$.
 - Line 22: the presolve does not reduce the model (4 rows, 12 columns).
 - Line 25: the LP relaxation of the root node has value 61.92991 (exactly $\frac{13253}{214}$).
The objective function takes integer values, so $LB = \lceil \frac{13253}{214} \rceil = 62$, and $62 \leq z(ILP) \leq 161$.
3. In line 33, $Incumbent = 67$ and $BestBd = 61.9299$:

$$gap = \frac{|61.9299 - 67|}{67} \approx 7.57\%,$$

as in the column **Gap**. The gap is divided by the incumbent value, as in the definition of the optimality gap.

4. The lines starting with H are new incumbent solutions found by heuristics at the root, 90, 83 and 67. In line 34 the root node is closed (**cutoff**): after the cut loop the best bound reaches 67, the value of the incumbent. Line 37: one Gomory cut is added.

Heuristics lower UB from 161 to 67, the LP relaxation and one Gomory cut raise LB from 62 to 67: $LB = UB = 67$ at the root, and the solver explores a single node.

Exercise 9.5 — The gap as a stopping criterion

Gurobi solves the generic ILP problem of the exercise «Reading a solver log» (a generic ILP problem with $n = 10$ integer variables $x_j \in \{0, \dots, 3\}$ and $m = 3$ constraints, $\max \mathbf{c}'\mathbf{x}$ subject to $\mathbf{A}\mathbf{x} \leq \mathbf{b}$, with integer pseudo-random data (Python `random.Random(4)`): a_{ij} uniform in $\{1, \dots, 20\}$, c_j uniform in $\{5, \dots, 25\}$, and $b_i = \lfloor 3 \sum_j a_{ij}/3 \rfloor$) with the same parameters and, in addition, $\text{MIPGap} = 0.02$. The log is:

Gurobi log: the same instance with $\text{MIPGap}=0.02$

```

01| Gurobi Optimizer version 13.0.3 build v13.0.3rc0 (linux64 - "Ubuntu 26.04 LTS")
02|
03| CPU model: Intel(R) Core(TM) i5-14500T, instruction set [SSE2|AVX|AVX2]
04| Thread count: 20 physical cores, 20 logical processors, using up to 1 threads
05|
06| Non-default parameters:
07| MIPGap 0.02
08| Heuristics 0
09| Cuts 0
10| LogToConsole 0
11| Presolve 0
12| Threads 1
13|
14| Optimize a model with 3 rows, 10 columns and 30 nonzeros (Max)
15| Model fingerprint: 0x92726b84
16| Model has 10 linear objective coefficients
17| Variable types: 0 continuous, 10 integer (0 binary)
18| Coefficient statistics:
19|   Matrix range    [1e+00, 2e+01]
20|   Objective range [7e+00, 2e+01]
21|   Bounds range    [3e+00, 3e+00]
22|   RHS range       [8e+01, 1e+02]
23|
24| Variable types: 0 continuous, 10 integer (0 binary)
25|
26| Root relaxation: objective 2.296103e+02, 4 iterations, 0.00 seconds (0.00 work units)
27|
28|      Nodes      |      Current Node      |      Objective Bounds      |      Work
29|  Expl Unexpl |  Obj  Depth IntInf | Incumbent    BestBd   Gap | It/Node Time
30|
31|      0       0  229.61033   0   3         -  229.61033       -   -   0s
32|      0       0  229.30837   0   3         -  229.30837       -   -   0s
33|      0       2  229.30837   0   3         -  229.30837       -   -   0s
34| *  12       6                9  220.0000000  224.56863  2.08%  1.2   0s
35|
36| Explored 13 nodes (18 simplex iterations) in 0.00 seconds (0.00 work units)
37| Thread count was 1 (of 20 available processors)
38|
39| Solution count 1: 220
40|
41| Optimal solution found (tolerance 2.00e-02)
42| Best objective 2.200000000000e+02, best bound 2.240000000000e+02, gap 1.8182%

```

Questions

1. Why does the solver stop? Recompute the final gap from the last line.
2. What does the sentence of line 41 guarantee about the solution of value 220?

3. With the default **MIPGap** the optimal value of this instance is 223. Is this consistent with the log?

Solution

1. Line 42: best objective 220 and best bound 224. The value 224.569 of **BestBd** in line 34 is rounded down by the solver, since all the data are integer. The final gap is

$$\frac{224 - 220}{220} \approx 1.82\% < 2\%,$$

below **MIPGap**: the solver stops after 13 nodes (line 36) instead of 25.

2. It guarantees only that $220 \leq z(\text{ILP}) \leq 224$: the solution is within 2% of the optimum, **not** that it is optimal. The solver writes «Optimal solution found» with the tolerance in use, here $2 \cdot 10^{-2}$.
3. Yes: 223 lies in the interval $[220, 224]$ given by the two bounds. The incumbent solution is 3 units, about 1.35%, below the optimum, and the remaining nodes would be needed to find it and prove it.

With **MIPGap** = 0.02 the solver stops with LB = 220 and UB = 224: the solution is certified only within 2%, and in fact it is **not optimal** ($z(\text{ILP}) = 223$).