22. Cutting planes and branch & bound

- Algorithms for solving MIPs
- Cutting plane methods
- Branch and bound methods

Laurent Lessard (www.laurentlessard.com)

MIP algorithms

We can't expect any algorithm for solving MIPs to be efficient in the worst case. Remember that we are solving NP-complete problems!

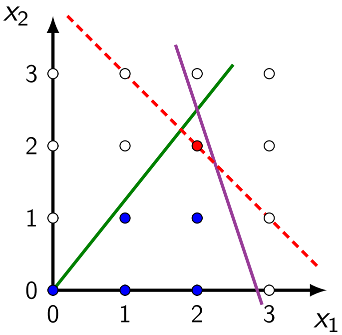
We will see two classes of algorithms:

1. *Cutting plane methods.* These can also be used to solve convex problems with integer constraints.

2. *Branch and bound methods.* These can also be used to solve nonliear problems with integer constraints (MINLP).

These are the most popular methods for solving MIP and combinatorial problems. Every modern solver uses variants of the above methods.

Review of MIPs

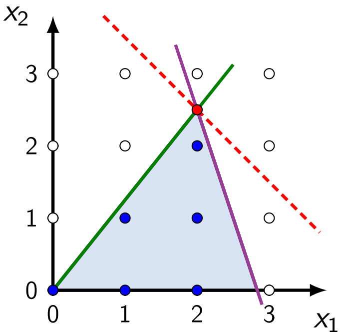


$$\max_x \quad x_1 + x_2$$
$$\text{s.t.} \quad -5x_1 + 4x_2 \leq 0$$
$$6x_1 + 2x_2 \leq 17$$
$$x_1, x_2 \geq 0 \quad \text{integer}$$

Optimal solution $= 4$

- ● Optimal solution
- ● Feasible points
- ○ Infeasible points

Review of MIPs



$$
\begin{aligned}
\max_x \quad & x_1 + x_2 \\
\text{s.t.} \quad & -5x_1 + 4x_2 \le 0 \\
& 6x_1 + 2x_2 \le 17 \\
& x_1, x_2 \ge 0
\end{aligned}
$$

Optimal solution $= 4.5$

- Remove integer constraint to obtain the LP relaxation.
- Optimal solution is an upper bound on the optimal cost.
- If solution is integral, it is optimal for the original problem.

Cutting plane method

Basic idea:

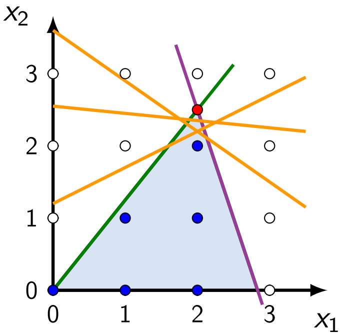
1. Solve LP relaxation.

2. If LP solution is integral, it is optimal for the original problem. We're done!

3. If LP solution is not integral, find a linear constraint that excludes the LP solution but does not exclude any integer points (always possible). This is called a **cut**.

4. Add the cut constraint to the problem. Return to step **1.**

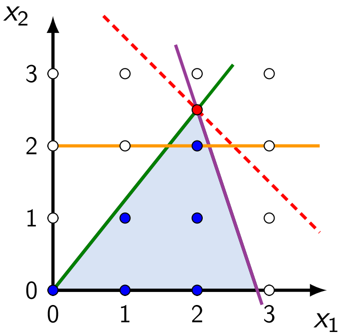
Cutting plane methods



$$
\begin{aligned}
\max_x \quad & x_1 + x_2 \\
\text{s.t.} \quad & -5x_1 + 4x_2 \leq 0 \\
& 6x_1 + 2x_2 \leq 17 \\
& \text{valid cuts} \\
& x_1, x_2 \geq 0
\end{aligned}
$$

- A cut must simultaneously exclude the LP solution while keeping all the feasible integer points.
- There always exists at least one valid cut.

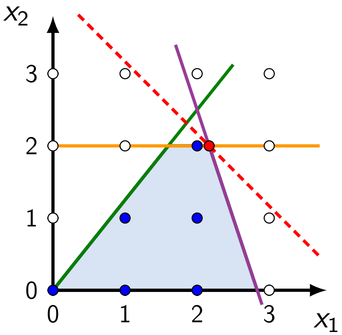
Cutting plane method



$$\begin{aligned} \max_x \quad & x_1 + x_2 \\ \text{s.t.} \quad & -5x_1 + 4x_2 \le 0 \\ & 6x_1 + 2x_2 \le 17 \\ & x_2 \le 2 \\ & x_1, x_2 \ge 0 \end{aligned}$$

- The constraint $x_2 \le 2$ is a valid cut because it excludes the optimal LP solution but doesn't exclude any integer points.
- Now solve the LP relaxation for this new problem...

Cutting plane method

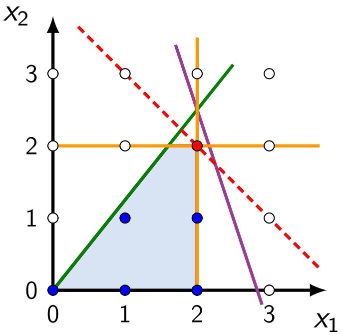


$$
\begin{aligned}
\max_{x} \quad & x_1 + x_2 \\
\text{s.t.} \quad & -5x_1 + 4x_2 \leq 0 \\
& 6x_1 + 2x_2 \leq 17 \\
& x_2 \leq 2 \\
& x_1, x_2 \geq 0
\end{aligned}
$$

Optimal solution $= 4.1667$

- Adding a cut reduces our upper bound because we are shrinking the feasible set (we added another constraint).
- Solution is still not an integer. Add another cut!

Cutting plane method

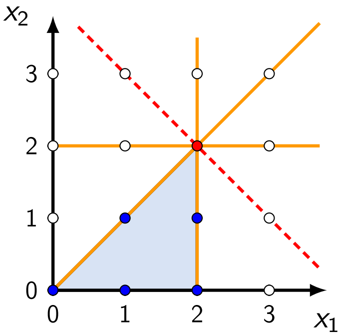


$$
\begin{aligned}
\max_{x} \quad & x_1 + x_2 \\
\text{s.t.} \quad & -5x_1 + 4x_2 \le 0 \\
& 6x_1 + 2x_2 \le 17 \\
& x_2 \le 2 \\
& x_1 \le 2 \\
& x_1, x_2 \ge 0
\end{aligned}
$$

Optimal solution $= 4$

- LP solution is integral, so it must also be optimal for the original integer problem.

Cutting plane method



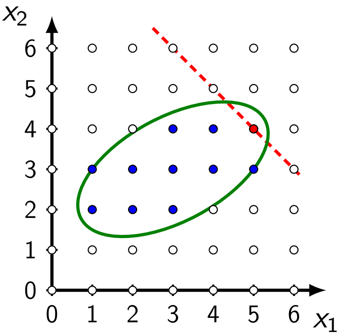
- If we add *all* the possible linear constraints that don't exclude feasible integral points, we obtain the **convex hull** of the feasible integral points.
- If we use the convex hull, then the LP relaxation always gives us the true optimal solution.

- The convex hull is generally very difficult to compute when we have a large number of decision variables.
- By using a cutting plane method, we can (hopefully) find the optimal point without computing the entire convex hull.

Gomory cut

- One famous method for creating valid cuts is called the **Gomory cut**, discovered by American mathematician Ralph Gomory (1950).
- Nice features of the Gomory cut:
  - Cuts are easy to compute; they can be computed as a byproduct of the simplex algorithm for solving LPs (this is why many LP solvers can also solve MIPs).
  - Cutting plane method using Gomory cuts is guaranteed to find the optimal solution using *finitely many* cuts.
- Of course, that finite number may be very very large...
- Gomory cuts or variants widely used in commercial solvers.

Cutting planes in general

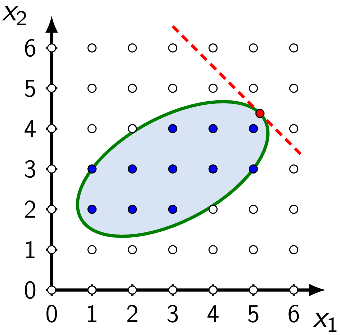


$$
\begin{aligned}
\max_{x} \quad & x_1 + x_2 \\
\text{s.t.} \quad & 2x_1^2 + 4x_2^2 - 3x_1x_2 \\
& \qquad - 3x_1 - 15x_2 + 19 \leq 0 \\
& x_1, x_2 \geq 0 \quad \text{integer}
\end{aligned}
$$

Optimal solution $= 9$

- The cutting plane idea still works for more general convex problems subject to integer constraints.
- Begin by solving the relaxation...

Cutting planes in general

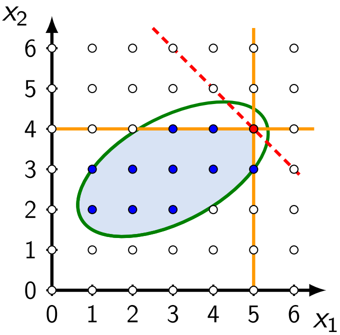


$$
\begin{aligned}
\max_{x} \quad & x_1 + x_2 \\
\text{s.t.} \quad & 2x_1^2 + 4x_2^2 - 3x_1x_2 \\
& \qquad - 3x_1 - 15x_2 + 19 \le 0 \\
& x_1, x_2 \ge 0
\end{aligned}
$$

Optimal solution $= 9.5386$

- This is a convex QCQP, and the optimal objective value is an upper bound on the optimal integer objective value.
- Add a cuts, as before...

Cutting planes in general



$$
\begin{aligned}
\max_{x} \quad & x_1 + x_2 \\
\text{s.t.} \quad & 2x_1^2 + 4x_2^2 - 3x_1x_2 \\
& \quad - 3x_1 - 15x_2 + 19 \le 0 \\
& x_1 \le 5, \quad x_2 \le 4 \\
& x_1, x_2 \ge 0
\end{aligned}
$$

Optimal solution $= 9$

- Since the cuts never exclude feasible integer points, once we obtain an integral solution to the relaxation we know we found an optimal point to the original problem.

Cutting planes recap

- Sequentially add linear constraints (cuts) and solve the relaxed version of the integer program.
- Cuts exclude the (non-integer) solution of the relaxed problem while preserving all the integral points in the feasible region
- It is always possible to find a cut whenever the relaxed problem is convex. This is because any two non-intersecting convex sets can be separated by a hyperplane, and this hyperplane can serve as a cut.
- The Gomory cut for MIPs is easy to compute and guaranteed to find the optimal solution after a finite number of cuts (though that number may be large).

Branch and bound methods

- **Basic idea:** it's a tree-based search heuristic to help us search the very large space of possible variable values.
- By keeping track of upper and lower bounds on the optimal solution, we can prune branches of the tree so we don't have to search every possibility (if we're lucky).
- We need two basic facts (assume a maximization MIP)
  - Removing a constraint makes the feasible set larger, so the new solution will be an upper bound to the optimal solution.
  - Adding a constraint makes the feasible set smaller, so the new solution will be an lower bound to the optimal solution.

Branch and bound methods

1. **Lower bounds:** keep track of the best current *lower bound*. This is a feasible (integer) point, so it provides a lower bound to the optimal cost. Update this lower bound if we come across a better one.

2. **Upper bounds:** solve several relaxed problems (subject to varying assumptions). These are easy to solve and since they involve relaxing constraints, they provide *upper bounds* on the optimal solution subject to those assumptions.

3. **Pruning:** if an upper bound turns out to be worse than our best lower bound, then the assumptions made in that case were incorrect and we can discard them.

Branch and bound example

$$\begin{array}{rl} \max & 15x_1 + 12x_2 + 4x_3 + 2x_4 \\ \text{s.t.} & 8x_1 + 5x_2 + 3x_3 + 2x_4 \le 10 \\ & x_i \in \{0, 1\} \end{array}$$

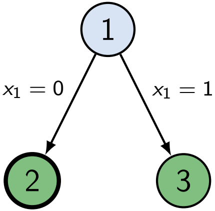
Best lower bound (feasible):
$z_\star = 0, \quad x_\star = \begin{bmatrix} 0 & 0 & 0 & 0 \end{bmatrix}$

Solve the LP relaxation of (1):
$z_{\mathrm{LP}}^{(1)} = 21.38, \quad x_{\mathrm{LP}}^{(1)} = \begin{bmatrix} 0.63 & 1 & 0 & 0 \end{bmatrix}$

LP solution is superior to $z_\star$, but not integral. Branch on the fractional variable x_1. Mark descendants active.

Example borrowed from MIT 15.053/8

Branch and bound example



$$\begin{aligned} \max \quad & 15x_1 + 12x_2 + 4x_3 + 2x_4 \\ \text{s.t.} \quad & 8x_1 + 5x_2 + 3x_3 + 2x_4 \le 10 \\ & x_i \in \{0, 1\} \end{aligned}$$

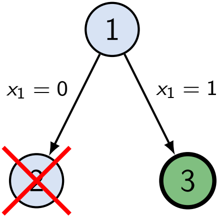
Best lower bound (feasible):
$z_\star = 0, \quad x_\star = \begin{bmatrix} 0 & 0 & 0 & 0 \end{bmatrix}$

Solve the LP relaxation of (2):
$z_{\text{LP}}^{(2)} = 18, \quad x_{\text{LP}}^{(2)} = \begin{bmatrix} 0 & 1 & 1 & 1 \end{bmatrix}$

It's integral and superior to $z_\star$, so it becomes our new lower bound. No need to branch any further. Prune the node and move on to the next active node.

Example borrowed from MIT 15.053/8

Branch and bound example



$$\begin{array}{ll} \max & 15x_1 + 12x_2 + 4x_3 + 2x_4 \\ \text{s.t.} & 8x_1 + 5x_2 + 3x_3 + 2x_4 \leq 10 \\ & x_i \in \{0, 1\} \end{array}$$

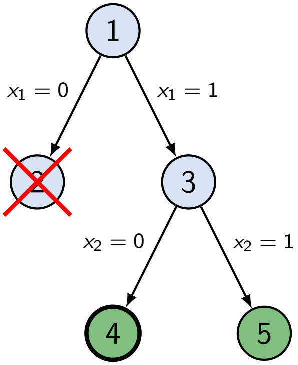
Best lower bound (feasible):
$z_\star = 18, \quad x_\star = \begin{bmatrix} 0 & 1 & 1 & 1 \end{bmatrix}$

Solve the LP relaxation of (3):
$z_{\text{LP}}^{(3)} = 19.8, \quad x_{\text{LP}}^{(3)} = \begin{bmatrix} 1 & 0.4 & 0 & 0 \end{bmatrix}$

LP solution is superior to $z_\star$, but not integral. Branch on the fractional variable x_2. Mark descendants as active.

Example borrowed from MIT 15.053/8

Branch and bound example



$$\begin{array}{ll} \max & 15x_1 + 12x_2 + 4x_3 + 2x_4 \\ \text{s.t.} & 8x_1 + 5x_2 + 3x_3 + 2x_4 \leq 10 \\ & x_i \in \{0, 1\} \end{array}$$

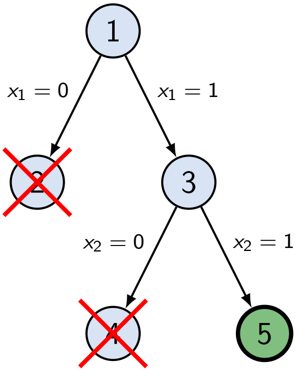
Best lower bound (feasible):
$z_\star = 18, \quad x_\star = \begin{bmatrix} 0 & 1 & 1 & 1 \end{bmatrix}$

Solve the LP relaxation of (4):
$z_{\text{LP}}^{(4)} = 17.67, \quad x_{\text{LP}}^{(4)} = \begin{bmatrix} 1 & 0 & 0.67 & 0 \end{bmatrix}$

LP solution is inferior to $z_\star$. No need to branch any further. Prune the node and move on to the next active node.

Example borrowed from MIT 15.053/8

Branch and bound example



$$\begin{array}{rl} \max & 15x_1 + 12x_2 + 4x_3 + 2x_4 \\ \text{s.t.} & 8x_1 + 5x_2 + 3x_3 + 2x_4 \le 10 \\ & x_i \in \{0, 1\} \end{array}$$

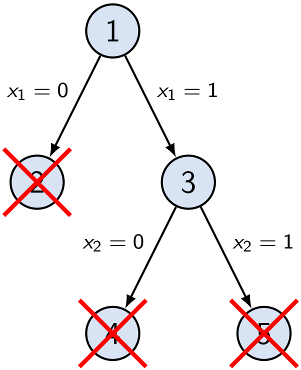
Best lower bound (feasible):
$z_\star = 18, \quad x_\star = \begin{bmatrix} 0 & 1 & 1 & 1 \end{bmatrix}$

Solve the LP relaxation of (5):
$z_{\text{LP}}^{(5)} = -\infty, \quad x_{\text{LP}}^{(5)} = \text{infeasible}$

LP is infeasible (inferior to $z_\star$). No need to branch any further. Prune the node. No more active nodes; we are done!

Example borrowed from MIT 15.053/8

Branch and bound example



$$\begin{aligned} \max \quad & 15x_1 + 12x_2 + 4x_3 + 2x_4 \\ \text{s.t.} \quad & 8x_1 + 5x_2 + 3x_3 + 2x_4 \le 10 \\ & x_i \in \{0, 1\} \end{aligned}$$

Optimal solution:
$z_\star = 18, \quad x_\star = \begin{bmatrix} 0 & 1 & 1 & 1 \end{bmatrix}$

Because we kept track of our bounds, we didn't need to search the entire space to find the optimal solution.

Example borrowed from MIT 15.053/8

Generic branch and bound

Let $(z_\star, x_\star)$ be a feasible point of the primal problem.

1. Let the primal problem be node (1) and mark it as active.
2. While there are active nodes remaining, select an active node (i) and mark it as inactive.
3. Solve the relaxation of node (i). Call it $(z_{\text{LP}}^{(i)}, x_{\text{LP}}^{(i)})$.
   - if $z_\star \geq z_{\text{LP}}^{(i)}$, prune node (i).
   - if $z_\star < z_{\text{LP}}^{(i)}$ and $x_{\text{LP}}^{(i)}$ is integral, then prune node (i) and replace $(z_\star, x_\star)$ with $(z_{\text{LP}}^{(i)}, x_{\text{LP}}^{(i)})$.
   - if $z_\star < z_{\text{LP}}^{(i)}$ and $x_{\text{LP}}^{(i)}$ is not integral, then branch on a non-integral variable. Mark the descendants as active.
4. Return to step **2.**

Flavors of branch and bound

- Integer variables:
  - If $1 \leq x \leq 10$, we can branch: $1 \leq x \leq 5$ and $6 \leq x \leq 10$.
  - Can also branch into more than two branches.
- Branching preference:
  - If there are many fractional variables in the LP solution, which one should we branch on? e.g. can pick the one with fractional part closest to 0.5.
  - Some solvers allow you to pick variable order for branching.
  - Can also branch on *constraints*.
- Alternate bounding methods:
  - Aside from LP relaxation, we can also simply remove a constraint, or use any other upper-bounding method.

Branch and cut

Branch and cut is a type of branch and bound method that uses cutting planes in addition to LP relaxation.

- In the bounding step, use cutting planes to improve the bounds found via the LP relaxation.
  - Can use just one cutting plane or many.
  - Cutting planes can be designed to provide local bounds (only valid for current branch) or global bounds.
- In the branching step, use the same branch and bound heuristic as before.

Any of these methods can work — there are many possible choices! Customized branch and bound algorithms can be designed and tailored for solving specific types of MIPs.

Even more generality

- Branch and bound is very general. All it requires is:
  - A *branching procedure* that partitions the feasible set into two or more sets (split the problem into smaller problems).
  - A *bounding procedure* that provides an upper bound for the objective value (must be relatively efficient).
- Versions of branch and bound can be used to find global optima for pretty much any optimization problem. This includes NLPs and MINLPs.