

Hands-on Tutorial on Optimization

F. Eberle, R. Hoeksma, and N. Megow

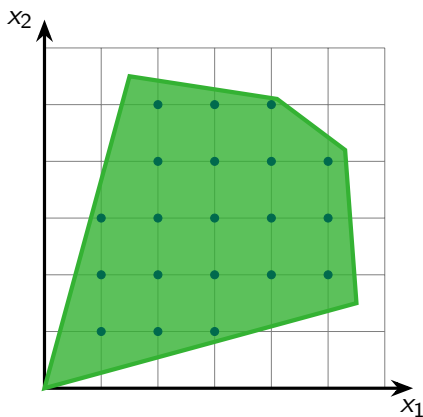
September 24, 2019

Cutting planes and dealing with symmetry

How to deal with integrality constraints

Canonical ILP:

$$\begin{array}{ll}\max & c^T x \\ \text{s.t.} & Ax \leq b \\ & x \in \mathbb{N}\end{array}$$

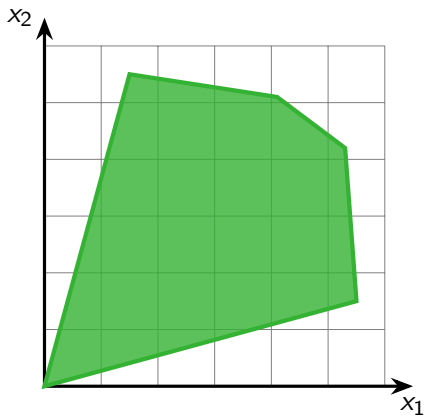


How to deal with integrality constraints

Canonical ILP:

$$\begin{array}{ll} \max & c^T x \\ \text{s.t.} & Ax \leq b \\ & x \geq 0 \end{array}$$

Valid when:



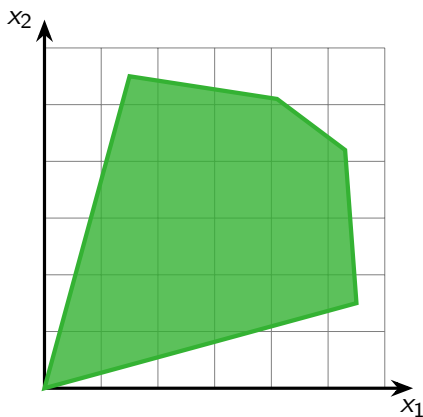
How to deal with integrality constraints

Canonical ILP:

$$\begin{array}{ll}\max & c^T x \\ \text{s.t.} & Ax \leq b \\ & x \geq 0\end{array}$$

Valid when:

- ▶ Your problem is **integral**



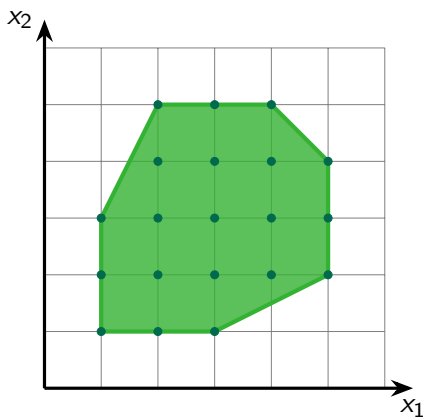
How to deal with integrality constraints

Canonical ILP:

$$\begin{array}{ll} \max & c^T x \\ \text{s.t.} & Ax \leq b \\ & x \geq 0 \end{array}$$

Valid when:

- ▶ Your problem is **integral**



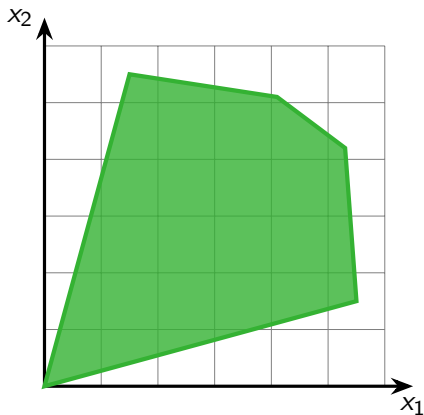
How to deal with integrality constraints

Canonical ILP:

$$\begin{array}{ll} \max & c^T x \\ \text{s.t.} & Ax \leq b \\ & x \geq 0 \end{array}$$

Valid when:

- ▶ Your problem is **integral**
- ▶ Your quantities are large enough that it doesn't matter



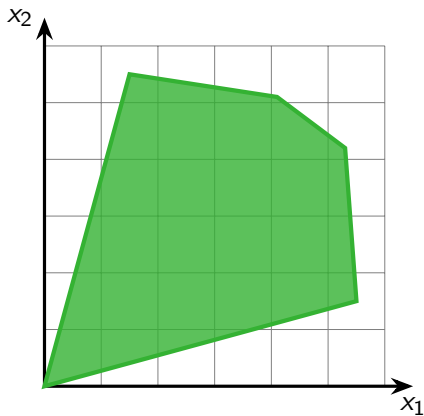
How to deal with integrality constraints

Canonical ILP:

$$\begin{aligned} \max \quad & c^T x \\ \text{s.t.} \quad & Ax \leq b \\ & x \geq 0 \end{aligned}$$

Valid when:

- ▶ Your problem is **integral**
- ▶ Your quantities are large enough that it doesn't matter
- ▶ You can safely round your non-integral solution



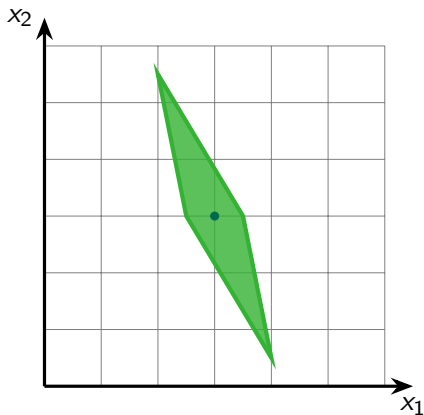
How to deal with integrality constraints

Canonical ILP:

$$\begin{aligned} \max \quad & c^T x \\ \text{s.t.} \quad & Ax \leq b \\ & x \geq 0 \end{aligned}$$

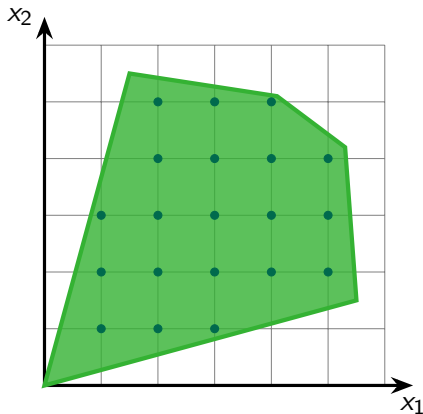
Valid when:

- ▶ Your problem is **integral**
- ▶ Your quantities are large enough that it doesn't matter
- ▶ You can safely round your non-integral solution



Cutting plane method

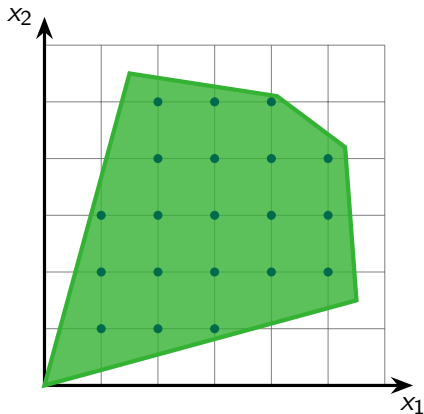
Cutting planes are extra constraints that tighten the description.



Cutting plane method

Cutting planes are extra constraints that tighten the description.

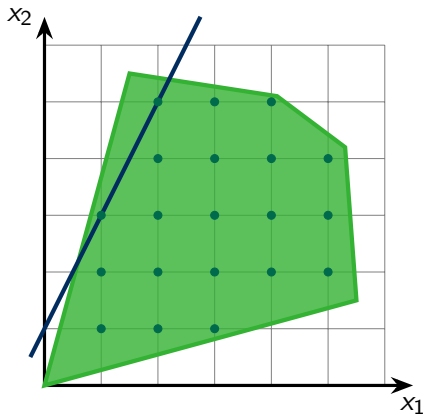
A **Cutting plane method** iteratively adds cutting planes to the LP-relaxation until the optimal LP solution is integral.



Cutting plane method

Cutting planes are extra constraints that tighten the description.

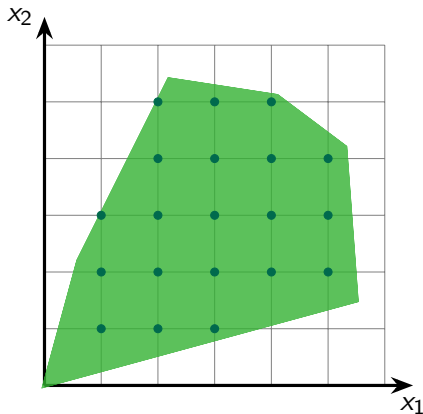
A **Cutting plane method** iteratively adds cutting planes to the LP-relaxation until the optimal LP solution is integral.



Cutting plane method

Cutting planes are extra constraints that tighten the description.

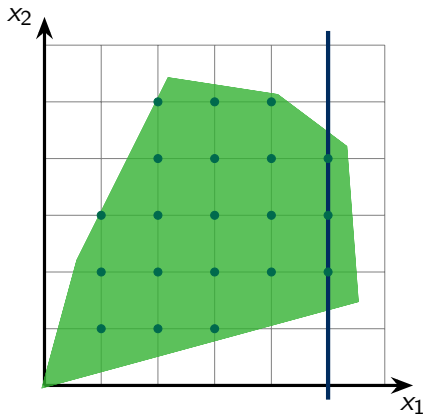
A **Cutting plane method** iteratively adds cutting planes to the LP-relaxation until the optimal LP solution is integral.



Cutting plane method

Cutting planes are extra constraints that tighten the description.

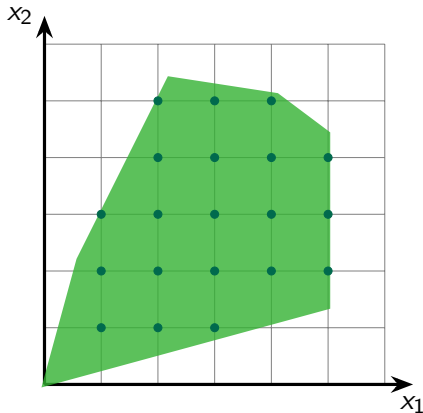
A **Cutting plane method** iteratively adds cutting planes to the LP-relaxation until the optimal LP solution is integral.



Cutting plane method

Cutting planes are extra constraints that tighten the description.

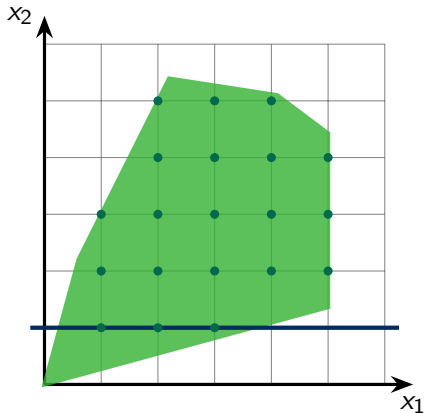
A **Cutting plane method** iteratively adds cutting planes to the LP-relaxation until the optimal LP solution is integral.



Cutting plane method

Cutting planes are extra constraints that tighten the description.

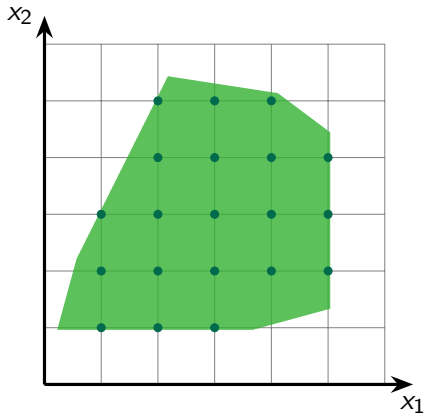
A **Cutting plane method** iteratively adds cutting planes to the LP-relaxation until the optimal LP solution is integral.



Cutting plane method

Cutting planes are extra constraints that tighten the description.

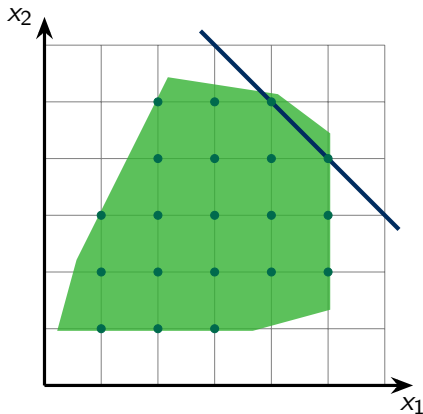
A **Cutting plane method** iteratively adds cutting planes to the LP-relaxation until the optimal LP solution is integral.



Cutting plane method

Cutting planes are extra constraints that tighten the description.

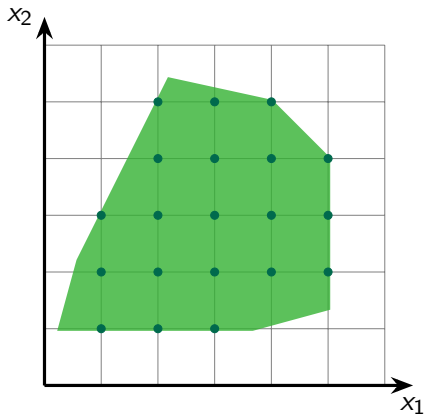
A **Cutting plane method** iteratively adds cutting planes to the LP-relaxation until the optimal LP solution is integral.



Cutting plane method

Cutting planes are extra constraints that tighten the description.

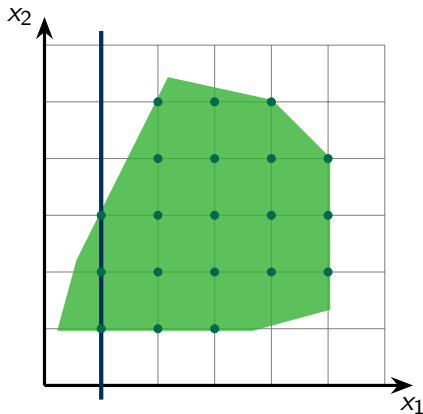
A **Cutting plane method** iteratively adds cutting planes to the LP-relaxation until the optimal LP solution is integral.



Cutting plane method

Cutting planes are extra constraints that tighten the description.

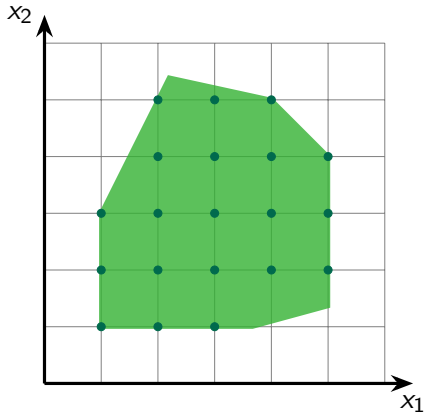
A **Cutting plane method** iteratively adds cutting planes to the LP-relaxation until the optimal LP solution is integral.



Cutting plane method

Cutting planes are extra constraints that tighten the description.

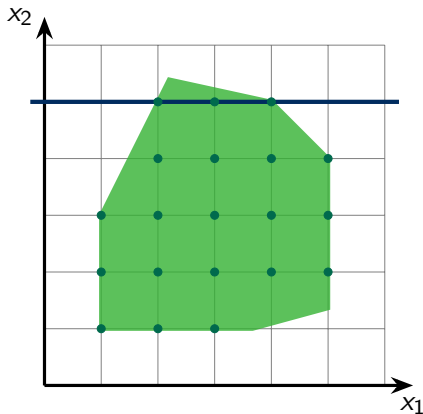
A **Cutting plane method** iteratively adds cutting planes to the LP-relaxation until the optimal LP solution is integral.



Cutting plane method

Cutting planes are extra constraints that tighten the description.

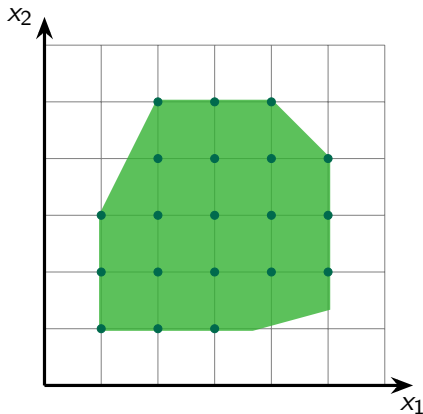
A **Cutting plane method** iteratively adds cutting planes to the LP-relaxation until the optimal LP solution is integral.



Cutting plane method

Cutting planes are extra constraints that tighten the description.

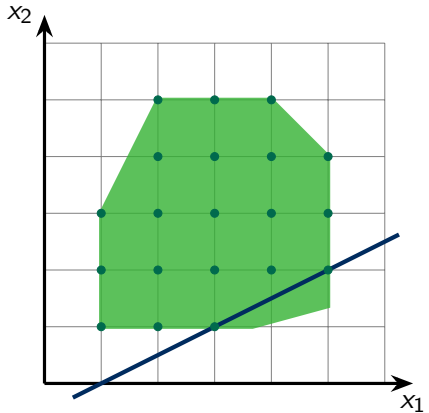
A **Cutting plane method** iteratively adds cutting planes to the LP-relaxation until the optimal LP solution is integral.



Cutting plane method

Cutting planes are extra constraints that tighten the description.

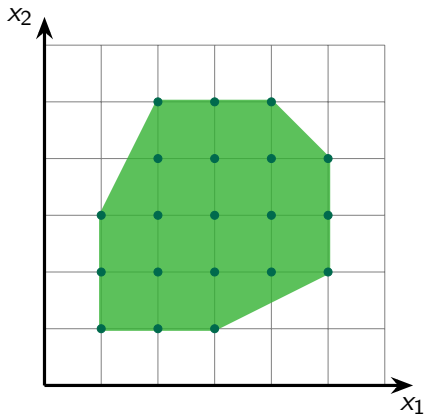
A **Cutting plane method** iteratively adds cutting planes to the LP-relaxation until the optimal LP solution is integral.



Cutting plane method

Cutting planes are extra constraints that tighten the description.

A **Cutting plane method** iteratively adds cutting planes to the LP-relaxation until the optimal LP solution is integral.

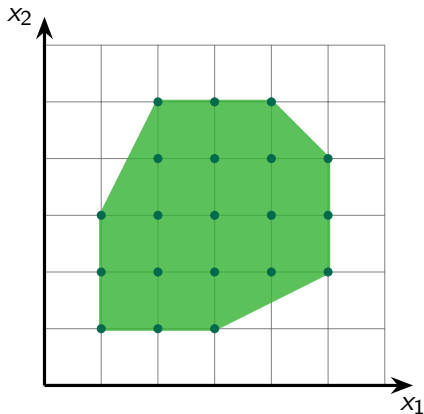


Cutting plane method

Cutting planes are extra constraints that tighten the description.

A **Cutting plane method** iteratively adds cutting planes to the LP-relaxation until the optimal LP solution is integral.

There exist general cutting plane methods → slow & impractical



Cutting plane method

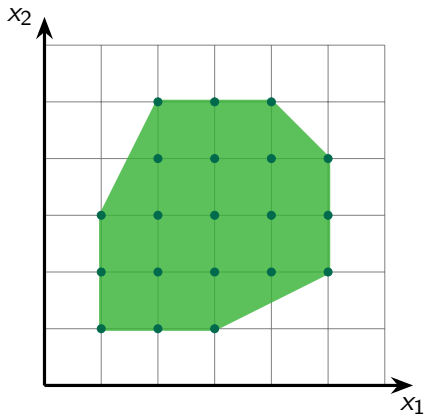
Cutting planes are extra constraints that tighten the description.

A **Cutting plane method** iteratively adds cutting planes to the LP-relaxation until the optimal LP solution is integral.

There exist general cutting plane methods \rightarrow slow & impractical

For specific problems very effective cutting plane methods exist (e.g. TSP, Knapsack).

ILP solvers implement cutting plane methods as well.



Adding constraints

Adding extra (redundant) constraints can slow the solver down...

Adding constraints

Adding extra (redundant) constraints can slow the solver down...
...or speed the solver up.

Adding constraints

Adding extra (redundant) constraints can slow the solver down...
...or speed the solver up.

Important case in which it can speed up the solver: removing symmetries.

Adding constraints

Adding extra (redundant) constraints can slow the solver down...
...or speed the solver up.

Important case in which it can speed up the solver: removing symmetries.

- ▶ Solvers have troubles dealing with symmetries.
- ▶ Thinking about the symmetries in your problem and adding smart extra constraints can help solving the problem faster.

Example - symmetries

Model: placing boxes in containers.

- ▶ The model only cares about volume.
- ▶ Many boxes with different sizes.
- ▶ Two containers.
- ▶ Objective: Fill the containers such that the maximum space left is minimized.

Example - symmetries

Model: placing boxes in containers.

- ▶ The model only cares about volume.
- ▶ Many boxes with different sizes.
- ▶ Two containers.
- ▶ Objective: Fill the containers such that the maximum space left is minimized.
- ▶ What symmetries exist?

Example - symmetries

Model: placing boxes in containers.

- ▶ The model only cares about volume.
- ▶ Many boxes with different sizes.
- ▶ Two containers.
- ▶ Objective: Fill the containers such that the maximum space left is minimized.
- ▶ What symmetries exist?
- ▶ Can we reduce symmetries?

Example - symmetries

Model: placing boxes in containers.

- ▶ The model only cares about volume.
- ▶ Many boxes with different sizes.
- ▶ **Many** containers.
- ▶ Objective: Store the boxes in as little as possible containers.

Example - symmetries

Model: placing boxes in containers.

- ▶ The model only cares about volume.
- ▶ Many boxes with different sizes.
- ▶ **Many** containers.
- ▶ Objective: Store the boxes in as little as possible containers.
- ▶ What symmetries exist?

Example - symmetries

Model: placing boxes in containers.

- ▶ The model only cares about volume.
- ▶ Many boxes with different sizes.
- ▶ **Many** containers.
- ▶ Objective: Store the boxes in as little as possible containers.
- ▶ What symmetries exist?
- ▶ Can we reduce symmetries?

Example - symmetries

Eight queens puzzle: place the eight queens on the chess board such that no two threaten each other.

Example - symmetries

Eight queens puzzle: place the eight queens on the chess board such that no two threaten each other.

- ▶ What symmetries exist?
- ▶ Can we reduce symmetries?

