# An Introduction to Branch-and-Cut
# Part I: Polyhedral Theory

Adam N. Letchford

Department of Management Science
Lancaster University

September 2011

# Outline

1. Integer Linear Programmes
2. Classical exact methods
3. Fundamentals of polyhedral theory
4. ‘Easy’ polyhedra (assignment, matching)
5. ‘Hard’ polyhedra (knapsack, traveling salesman)

# A word on notation

- $\mathbb{R}$ denotes the real numbers.
- $\mathbb{Z}$ denotes the integers.
- $\mathbb{R}_+$ denotes the *non-negative* real numbers, and similarly for $\mathbb{Z}_+$.
- $\mathbb{R}^n$ denotes the set of all vectors with $n$ real entries, and similarly for $\mathbb{Z}^n$.
- $\mathbb{R}^n$ can also be viewed as the Euclidean space in $n$ dimensions.

# Integer Linear Programmes

The majority of Combinatorial Optimisation Problems (COPs) can be formulated as *Integer Linear Programmes* (ILPs).

An ILP with $n$ variables and $m$ constraints takes the form

$$\min\left\{c^T x : Ax \geq b, x \in \mathbb{Z}^n_+\right\},$$

where $x \in \mathbb{Z}^n_+$ is a vector of integer decision variables, $c \in \mathbb{Z}^n$ is the objective function vector, $b \in \mathbb{Z}^m$ is the right hand side vector and $A \in \mathbb{Z}^{m \times n}$ is the matrix of constraint coefficients.

# Integer Linear Programmes (cont.)

- If we replace the condition $x \in \mathbb{Z}^n_+$ with $x \in \mathbb{R}^n_+$, we obtain the *LP relaxation* of the ILP.
- The LP relaxation is easy to solve (e.g., by simplex or interior-point methods).
- If the solution to the LP relaxation is integral, we are done.
- This happens with certain network problems (e.g., maximum flow, minimum cost flow, assignment).
- In general, however, it is unlikely to happen.
- So more sophisticated approaches are needed.

# Integer Linear Programmes (cont.)

**Remark**

Many $\mathcal{NP}$-hard COPs (e.g., knapsack, traveling salesman) can be formulated as ILPs. So Integer Linear Programming is $\mathcal{NP}$-hard.

Linear Programming, on the other hand, is solvable in polynomial time (Khachiyan, 1979).

# Classical exact methods

Several exact methods have been proposed to solve ILPs. For example:

- *dynamic programming* (Bellmann, 1957)
- *cutting planes* (Gomory, 1958)
- *branch-and-bound* (Land & Doig, 1961)
- *Lagrangian relaxation* (Geoffrion, 1974).

Of these, cutting planes and branch-and-bound are the most widely applicable.

# Classical exact methods (cont.)

Gomory's cutting-plane method has some serious drawbacks:

- A huge number of cutting planes are typically needed.
- Cuts tend to get weaker and weaker ('tailing-off').
- Numerical errors accumulate as cuts are added.
- No feasible integer solution is obtained until the very end.

On the other hand, it requires little memory (since at most $n$ constraints are binding at any given time).

# Classical exact methods (cont.)

Branch-and-bound usually performs better, but it has some drawbacks as well:

- A huge number of sub-problems may need to be solved.
- If we do ‘breadth-first’ search, we may run out of memory.
- If we do ‘depth-first’ search, we may waste time exploring a ‘dead-end’.

On the other hand, it usually yields feasible integer solutions quickly.

# Classical exact methods (cont.)

So what to do?

- In the 1970s, researchers starting looking for 'strong' cutting planes, that would perform better than those of Gomory.
- In the 1980s, researchers got good results by cutting first and branching second (now called *cut-and-branch*).
- In the 1990s, they got even better results by cutting at every node of the branch-and-bound tree (now called *branch-and-cut*).
- Branch-and-cut works so well, that it is now used by almost all ILP software packages!

# Fundamentals of polyhedral theory

- I said that researchers looked for ‘strong’ cutting planes.
- But how can we decide if a cutting plane is ‘strong’ or ‘weak’?
- The answer is to look at certain *integer polyhedra* associated with the ILP in question.
- The idea arose in early work by Balinski (assignment polyhedra), Gomory (corner polyhedra) and Edmonds (matching and matroid polyhedra).
- Later, Chvátal, Padberg, Balas, Wolsey and others developed and popularised the concept.
- It is now a field in its own right: polyhedral combinatorics.

# Fundamentals of polyhedral theory (cont.)

The key concept is that of a *polyhedron*:

**Definition**

A set $P \subset \mathbb{R}^n$ is called a *convex polyhedron* (or just *polyhedron*) if there exist an integer $m \in \mathbb{Z}_+$, a matrix $A \in \mathbb{R}^{m \times n}$ and a vector $b \in \mathbb{R}^n$ such that:

$$P = \{x \in \mathbb{R}^n : Ax \geq b\}.$$

# Fundamentals of polyhedral theory (cont.)

- Cubes, pyramids, prisms, etc. are 3-dimensional polyhedra.
- Polygons (triangles, squares, etc.) are 2-dimensional polyhedra.
- In more than 3 dimensions, polyhedra are hard to visualise!

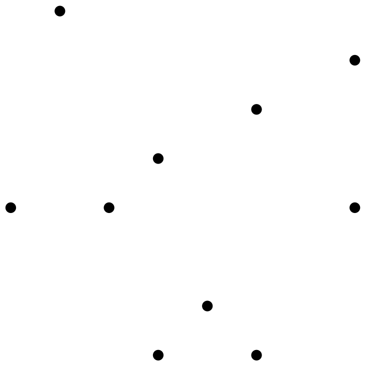
# Fundamentals of polyhedral theory (cont.)

Another important concept is that of a *convex hull*:

**Definition**

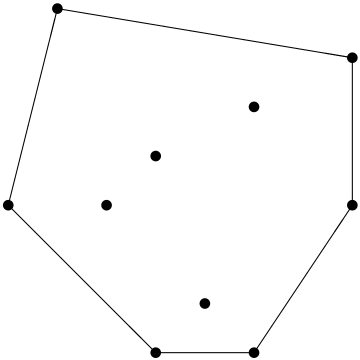
The *convex hull* of a set of points is the smallest polyhedron that contains all of the points in the set.

# Fundamentals of polyhedral theory (cont.)



Ten random points in $\mathbb{R}^2$.

# Fundamentals of polyhedral theory (cont.)



Ten random points in $\mathbb{R}^2$
and their convex hull.

# Fundamentals of polyhedral theory (cont.)

The LP relaxation of our ILP is:

$$\min\left\{c^T x : Ax \geq b,\ x \in \mathbb{R}^n_+\right\}.$$

Its feasible region is the polyhedron:

$$P = \left\{x \in \mathbb{R}^n_+ : Ax \geq b\right\}.$$

The set of integer solutions is:

$$\left\{x \in \mathbb{Z}^n_+ : Ax \geq b\right\}.$$

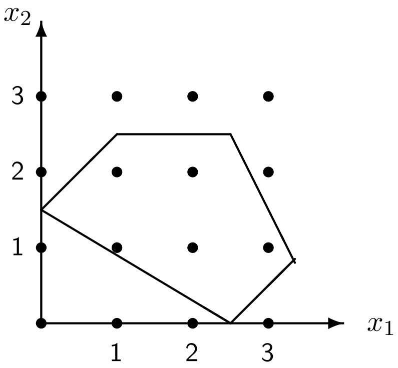
The *integral hull* is the convex hull of the set of integer solutions, i.e., the polyhedron:

$$P_I = \text{conv}\left\{x \in \mathbb{Z}^n_+ : Ax \geq b\right\}.$$

# Fundamentals of polyhedral theory (cont.)

- By definition, we have $P_I \subseteq P$.
- Usually, $P_I$ is strictly contained in $P$.
- A cutting plane is nothing but a linear inequality that is *valid for* $P_I$ *but not for* $P$.
- The strongest possible cutting planes are those that define *facets* of $P_I$.

# Example I



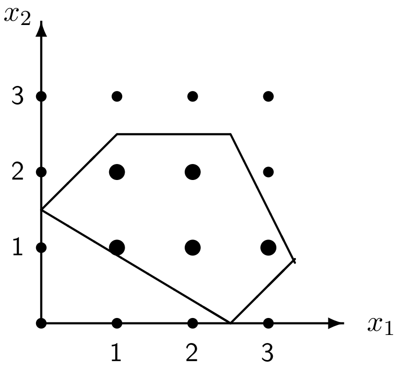

$P$ is defined by five inequalities:

$2x_2 \leq 5$, $6x_1 + 10x_2 \geq 15$

$2x_1 - 2x_2 \leq 5$, $2x_1 - 2x_2 \geq -3$

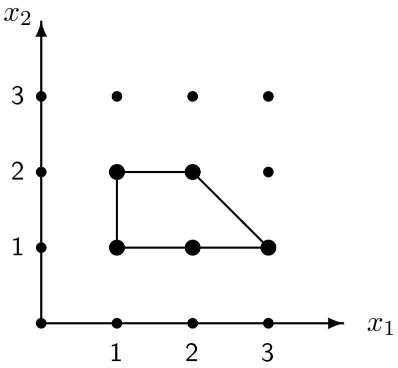
$4x_1 + 2x_2 \leq 15$

# Example I (cont.)




$P$ contains 5 integer points

# Example I (cont.)

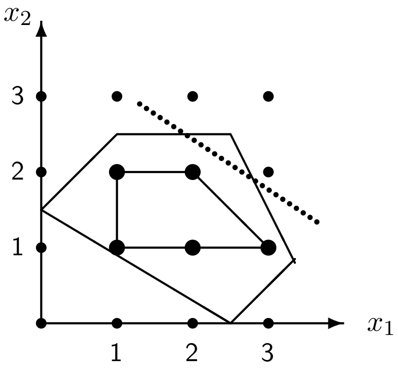



$P_I$ has four facets:

$x_1 \geq 1$, $x_2 \geq 1$

$x_2 \leq 2$, $x_1 + x_2 \leq 4$

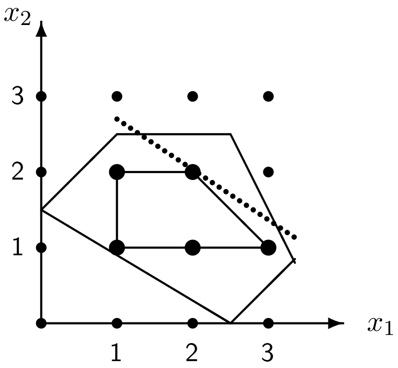
# Example I (cont.)




A weak cutting plane:

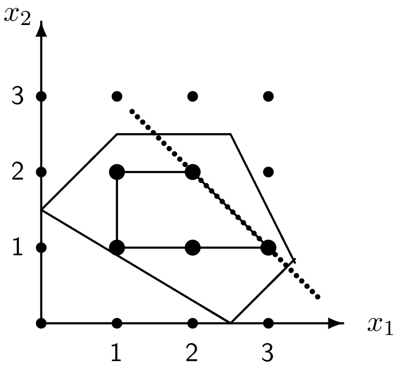
doesn't even touch $P_I$

# Example I (cont.)




A stronger cutting plane: touches $P_I$

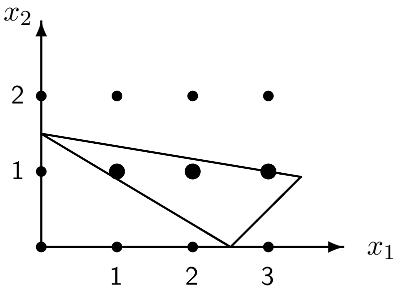
# Example I (cont.)




Strongest possible cutting plane:

defines a facet of $P_I$

# Example II

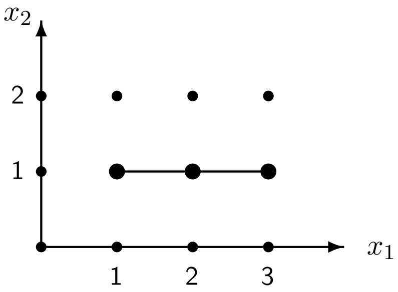



$P$ is defined by three inequalities:

$2x_1 - 2x_2 \leq 5$, $2x_1 - 2x_2 \geq -3$

$x_1 + 6x_2 \leq 9$

# Example II (cont.)




$P_I$ is defined by the equation:

$x_2 = 1$

and the inequalities:

$1 \leq x_1 \leq 3$

# Fundamentals of polyhedral theory (cont.)

The bad news:

- The number of facets can grow very rapidly with $n$ (worse than exponential).
- We can't compute them all — it would be easier to solve the ILP!
- For general ILPs, testing if an inequality defines a facet is itself $\mathcal{NP}$-hard (Karp & Papadimitriou, 1980)!

# Fundamentals of polyhedral theory (cont.)

The good news:

- For most polynomially-solvable COPs (e.g., assignment, matching, spanning tree), we completely understand the facets.
- For many $\mathcal{NP}$-hard COPs (e.g., knapsack, traveling salesman), we know and understand *some* of the facets.
- We don't need to know all of the facets to solve an instance.
- Even if an inequality does not define a facet, it may still be a useful cutting plane.

# Easy polyhedra I: assignment polyhedra

The assignment problem can be formulated as the ILP:

$$\begin{array}{lllr} \min & \sum_{i=1}^{n} \sum_{j=1}^{n} c_{ij} x_{ij} & & \\ \text{s.t.} & \sum_{j=1}^{n} x_{ij} = 1 & (i = 1, \ldots, n) & (1) \\ & \sum_{i=1}^{n} x_{ij} = 1 & (j = 1, \ldots, n) & (2) \\ & x_{ij} \in \mathbb{Z}_+ & (i = 1, \ldots, n; j = 1, \ldots, n). & \end{array}$$

We have $P = \left\{x \in \mathbb{R}_+^{n \times n} : (1), (2) \text{ hold}\right\}$.

Birkhoff (1946) proved that, in this case, $P = P_I$. No cutting planes are necessary.

The non-negativity inequalities $x_{ij} \geq 0$ define all the facets.

# Easy polyhedra II: perfect matching polyhedra

- The assignment problem can be viewed as a perfect matching problem on a bipartite graph.
- Now consider the perfect matching problem on a *general* graph $G = (V, E)$.
- For any vertex set $S \subset V$, we let $\delta(S)$ denote the set of edges with one end-vertex in $S$.
- Then, $\delta(\{i\})$ denotes the set of edges incident on the vertex $i$.

# Easy polyhedra II: perfect matching polyhedra (cont.)

The perfect matching problem can now be formulated as the ILP:

$$\begin{array}{lll} \min & \sum_{e \in E} c_e x_e & \\ \text{s.t.} & \sum_{e \in \delta(\{i\})} x_e = 1 & (\forall i \in V) \\ & x_e \in \mathbb{Z}_+ & (\forall e \in E). \end{array} \quad (3)$$

We have $P = \left\{ x \in \mathbb{R}_+^{|E|} : (3) \text{ hold} \right\}$.

When $G$ is not bipartite, $P_I$ is *not* equal to $P$.

# Easy polyhedra II: perfect matching polyhedra (cont.)

Edmonds (1965) showed that $P_I$ is defined by the equations (3), the non-negativity inequalities $x_e \geq 0$, and the following *odd-cut* inequalities:

$$\sum_{e \in \delta(S)} x_e \geq 1 \qquad (\forall S \subset V : |S| \text{ odd}).$$

Note that there are an exponential number of odd-cut inequalities.

(Fortunately, Edmonds' blossom algorithm does not work by listing all of them!)

# Hard polyhedra I: knapsack polyhedra

The 0-1 knapsack problem can be formulated as the ILP:

$$\begin{aligned} \max \quad & \textstyle\sum_{i=1}^n p_i x_i \\ \text{s.t.} \quad & \textstyle\sum_{i=1}^n w_i x_i \le c && && (4) \\ & x_i \le 1 && (i = 1, \ldots, n) && (5) \\ & x_i \in \mathbb{Z}_+ && (i = 1, \ldots, n). \end{aligned}$$

We have $P = \left\{ x \in \mathbb{R}^n_+ : (4), (5) \text{ hold} \right\}$.

$P_I$ is almost never equal to $P$.

# Hard polyhedra I: knapsack polyhedra (cont.)

- Since the 0-1 knapsack problem is $\mathcal{NP}$-hard, we cannot hope to characterize *all* facets for general $n$.
- But some useful cutting planes known.
- E.g., Balas (1975) and Wolsey (1975) independently discovered the 'cover' inequalities.
- A cover is a set of items whose total weight exceeds the knapsack capacity.
- If $C$ is a cover, then the cover inequality $\sum_{i \in C} x_i \leq |C| - 1$ is valid for $P_I$.
- (The number of covers can be exponential in $n$.)

# Hard polyhedra I: knapsack polyhedra (cont.)

- Cover inequalities do not often define facets.
- They can be strengthened in various ways.
- E.g., Balas (1975) had the following idea.
- Let $j$ be the heaviest item in $C$.
- Let $E(C)$ (the 'extension' of $C$) contain all items not in $C$ that are at least as heavy as item $j$.
- The extended cover inequality is:

$$\sum_{i \in C \cup E(C)} x_i \leq |C| - 1.$$

- (Again: exponential in number!)

# Hard polyhedra II: traveling salesman polyhedra

The traveling salesman problem can be formulated as the ILP:

$$\begin{array}{lll} \min & \sum_{e \in E} c_e x_e & \\ \text{s.t.} & \sum_{e \in \delta(\{i\})} x_e = 2 & (\forall i \in V) \quad (6) \\ & \sum_{e \in \delta(S)} x_e \geq 2 & (\forall S \subset V) \quad (7) \\ & x_e \in \{0,1\} & (\forall e \in E). \end{array}$$

We have $P = \{x \in [0,1]^n : (6), (7) \text{ hold}\}$.

Note: now even $P$ has an exponential number of facets!

# Hard polyhedra II: traveling salesman polyhedra (cont.)

- Again, since the TSP is $\mathcal{NP}$-hard, we cannot hope to characterize *all* facets for general $n$.
- Padberg & Grötschel (1979) showed that the subtour elimination constraints (7) and the bounds $0 \le x_e \le 1$ define facets.
- They introduced some other facet-defining inequalities called *comb* inequalities.
- *Lots* of other inequalities are known: clique-tree, chain, ladder, crown, path, star, etc.
- Full description known only for $n$ up to 9.

# SUMMARY

- Most COPs can be formulated as ILPs.
- To solve ILPs to proven optimality, strong cutting planes are desirable.
- The strongest possible cutting planes are those that define *facets of the integral hull*.
- For many polynomially-solvable COPs, we understand all of the facets.
- For $\mathcal{NP}$-hard COPs, we understand only some of them.
- But even knowing only some of the facets can be extremely useful...