Discrete Optimization

MA2827

Fondements de l'optimisation discrète

https://project.inria.fr/2015ma2827/

Material from P. Van Hentenryck's course

Outline

- Linear Programming
- Mixed Integer Program
- Examples (TSP, Knapsack)

What is a linear program?

$$\min c_1x_1 + \ldots + c_nx_n$$

subject to

$$a_{11}x_1 + \ldots + a_{1n}x_n \leq b_1$$

$$\ldots$$

$$a_{m1}x_1 + \ldots + a_{mn}x_n \leq b_m$$

$$x_i \geq 0 \quad (1 \leq i \leq n)$$

What is a linear program?

$$\min c_1x_1 + \ldots + c_nx_n$$

subject to

$$a_{11}x_1 + \ldots + a_{1n}x_n \le b_1$$

$$\ldots$$

$$a_{m1}x_1 + \ldots + a_{mn}x_n \le b_m$$

$$x_i \ge 0 \quad (1 \le i \le n)$$

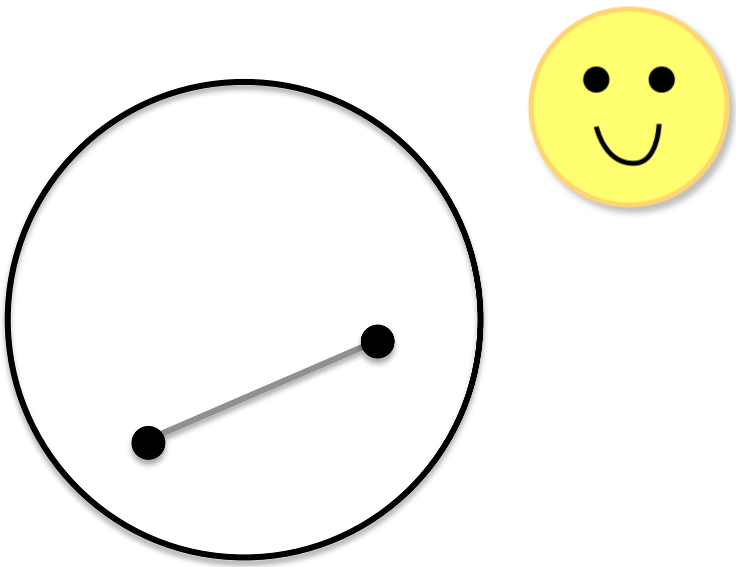
- n variables, m constraints
- Variables are non-negative
- Inequality constraints

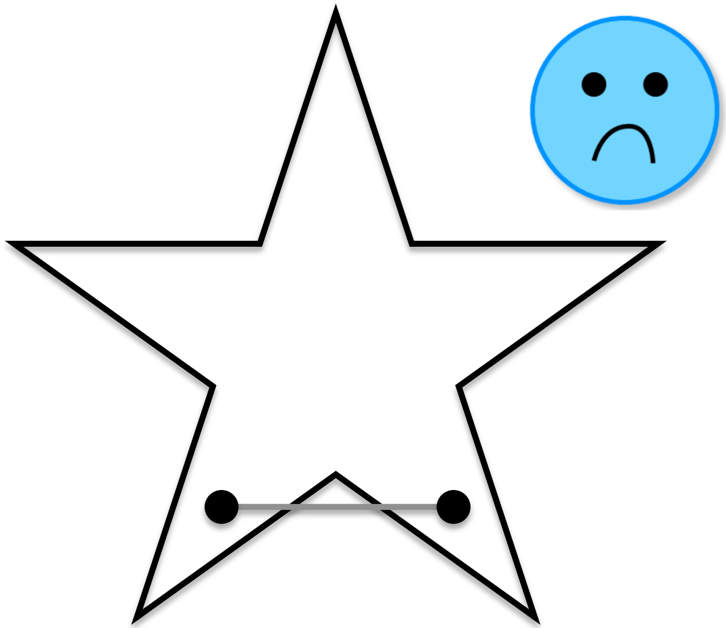
What is a linear program?

- What about maximization?
  - solve $\min -(c_1 x_1 + \ldots + c_n x_n)$
- What if a variable can take negative values?
  - replace x_i by $x_i^+ - x_i^-$
- Equality constraints?
  - use two inequalities
- Variables taking integer values?
  - mixed integer programming

Geometrical view

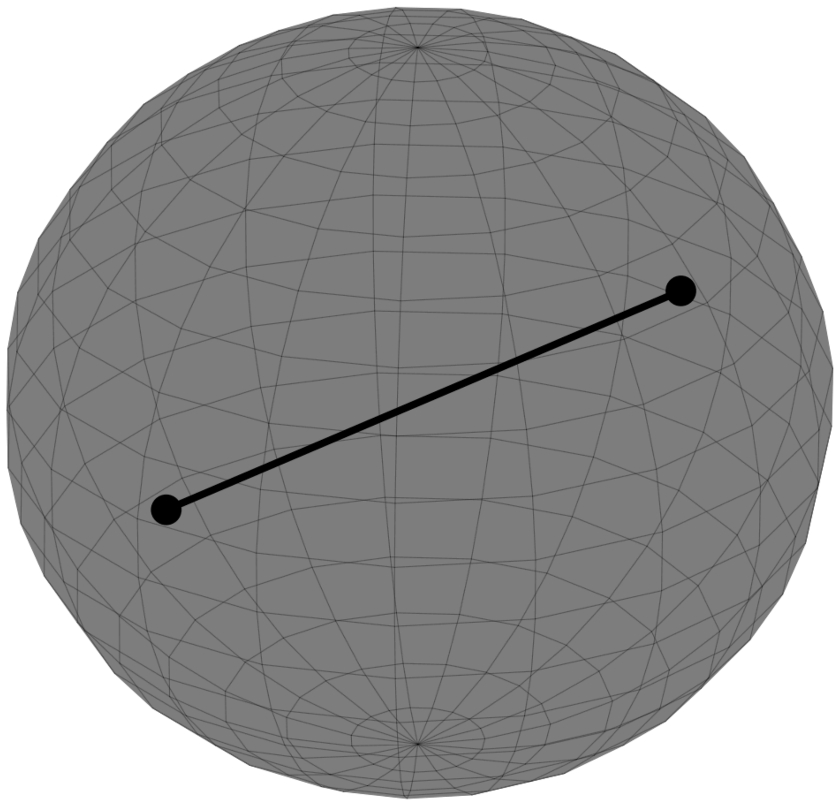
- First, convex sets

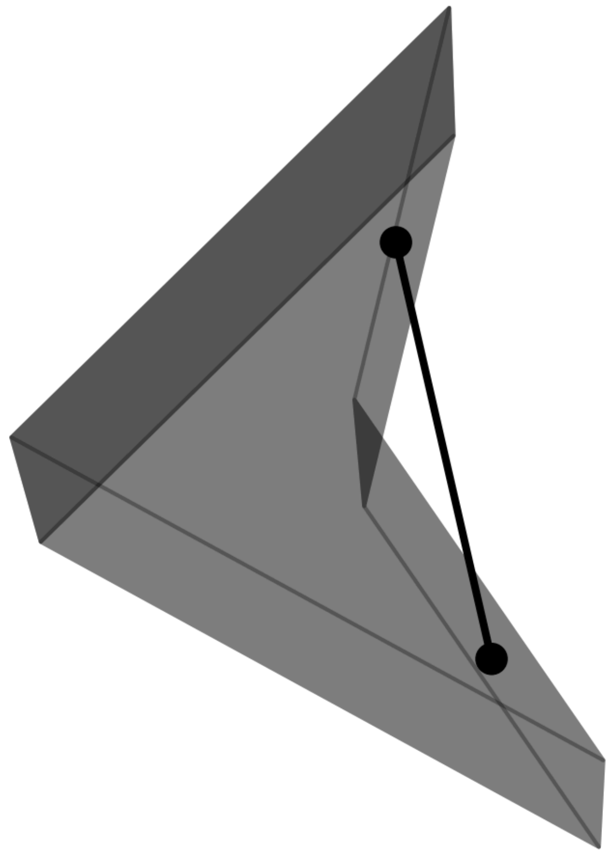




Geometrical view

- First, convex sets





Geometrical view

- Convex combinations

$\lambda_1 v_1 + \ldots + \lambda_n v_n$ is a convex combination of $v_1, \ldots, v_n$ if

$$\lambda_1 + \ldots + \lambda_n = 1$$

and

$$\lambda_i \geq 0 \quad (1 \leq i \leq n).$$

Geometrical view

- Convex sets
  - A set S in R^n is convex if it contains all the convex combinations of the points in S
- The intersection of convex sets is convex
  - Proof ?

Geometrical view

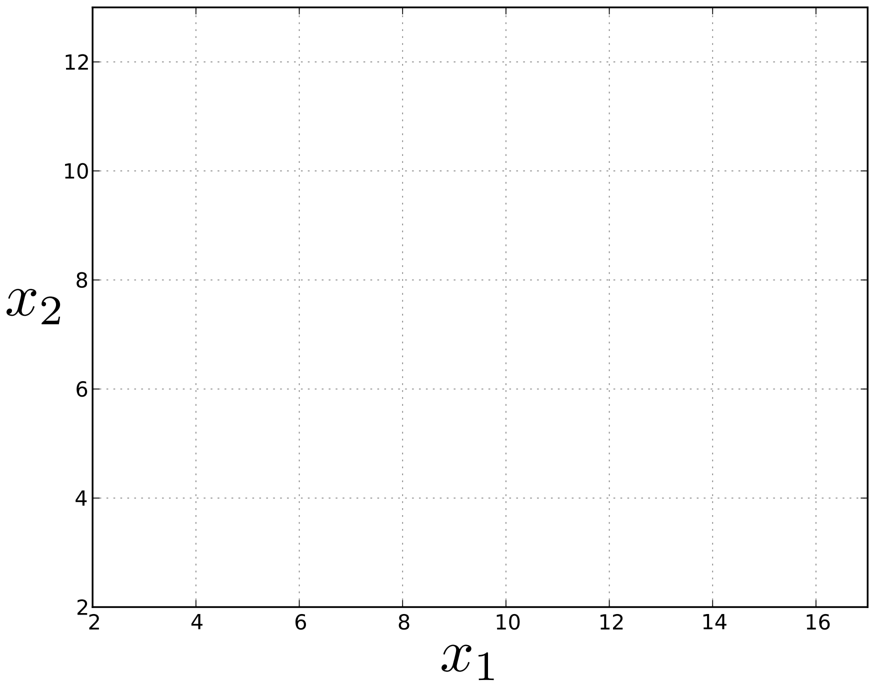
- A half space is a convex set

$$a_1 x_1 + \ldots + a_n x_n \leq b$$

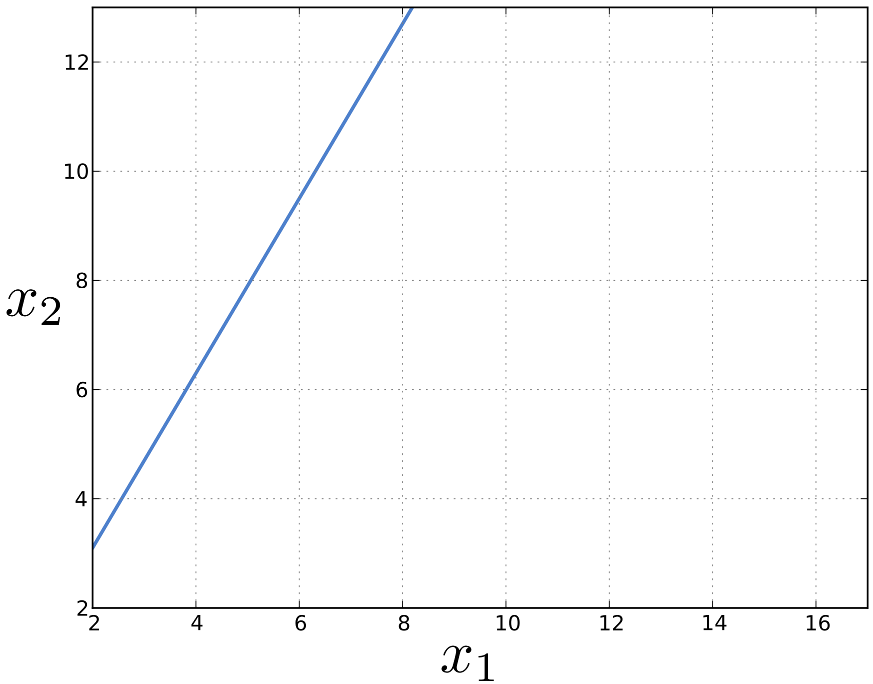
- Polyhedron: intersection of set of half spaces (also convex). If finite, polytope

$$a_{11} x_1 + \ldots + a_{1n} x_n \leq b_1$$
$$\ldots$$
$$a_{m1} x_1 + \ldots + a_{mn} x_n \leq b_m$$

Geometrical view

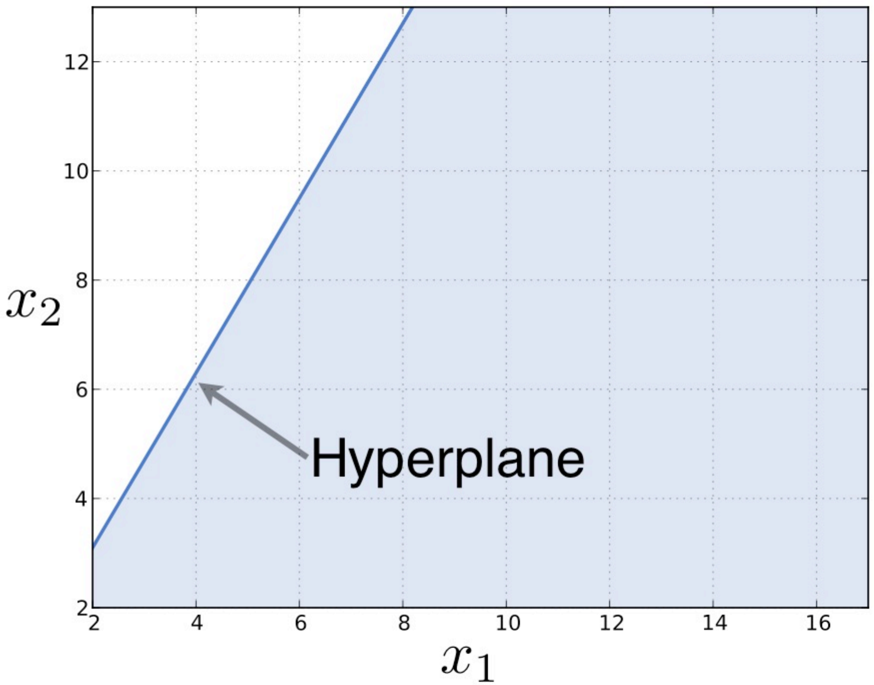


Geometrical view



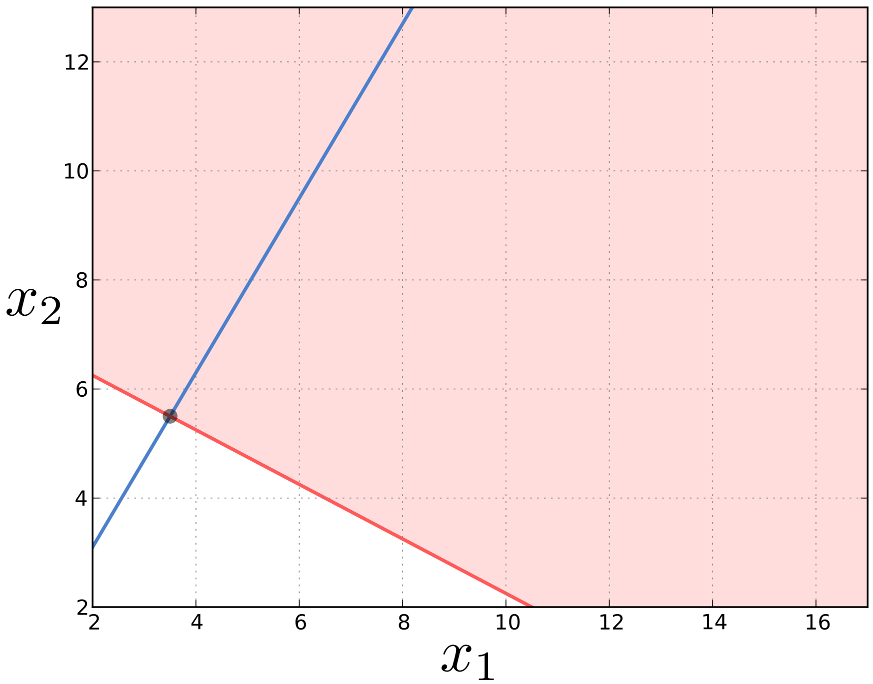
$$-4.0x_1 \quad + \quad 2.5x_2 \quad \leq \quad -0.25$$

Geometrical view



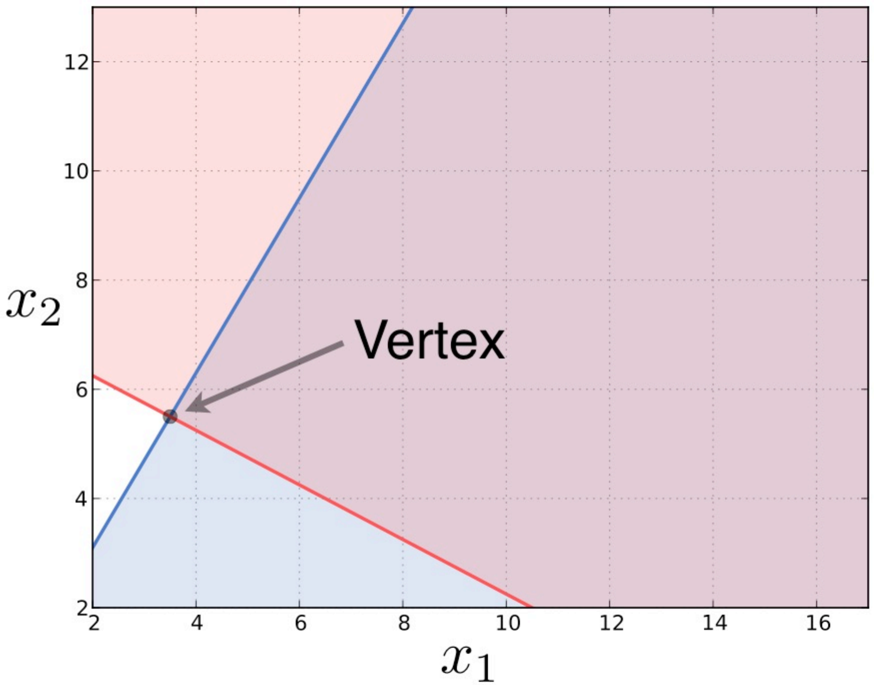
$$-4.0x_1 \quad + \quad 2.5x_2 \quad \leq \quad -0.25$$

Geometrical view



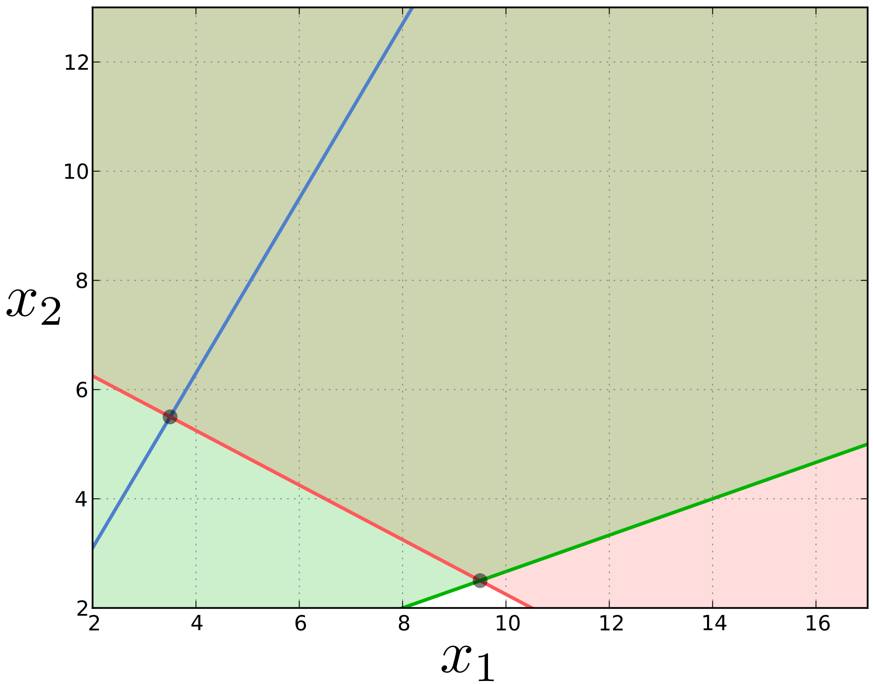
$$-3.0x_1 \quad - \quad 6.0x_2 \quad \leq \quad -43.50$$

Geometrical view



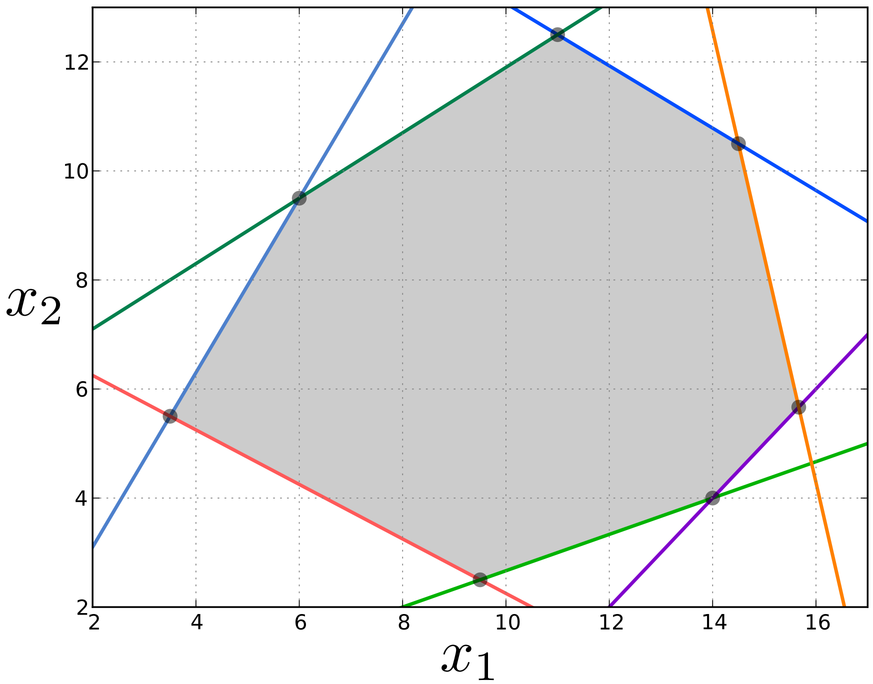
$$-3.0x_1 \quad - \quad 6.0x_2 \quad \leq \quad -43.50$$

Geometrical view



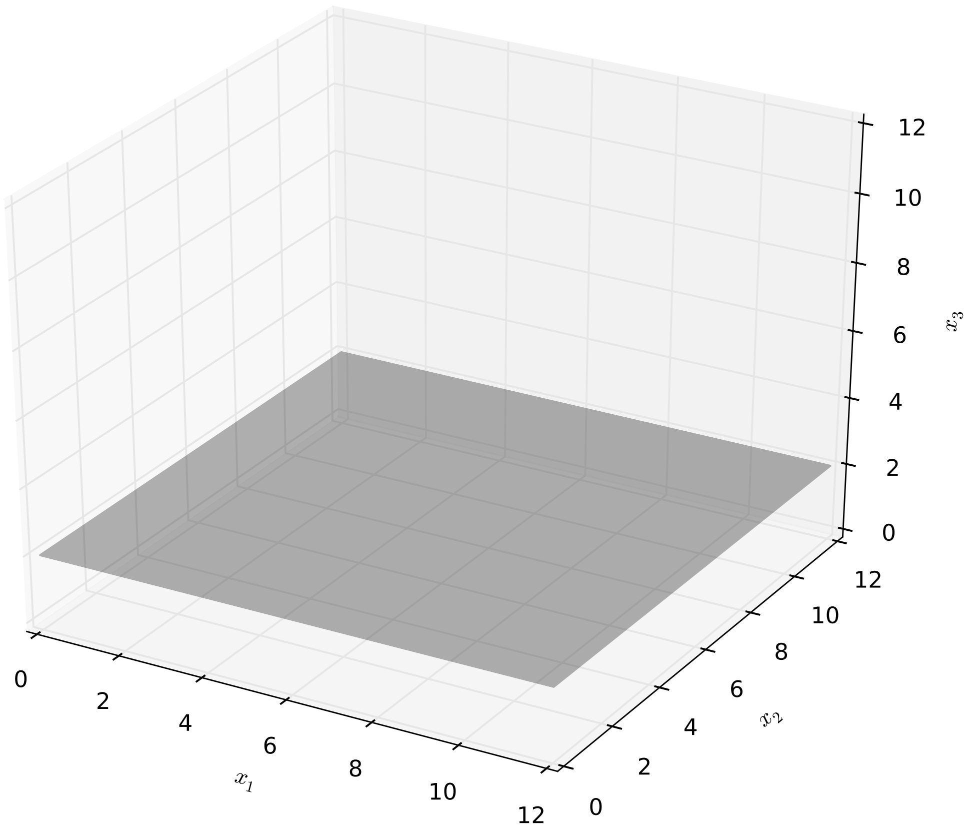
$$1.5x_1 \quad - \quad 4.5x_2 \quad \leq \quad 3.00$$

Geometrical view

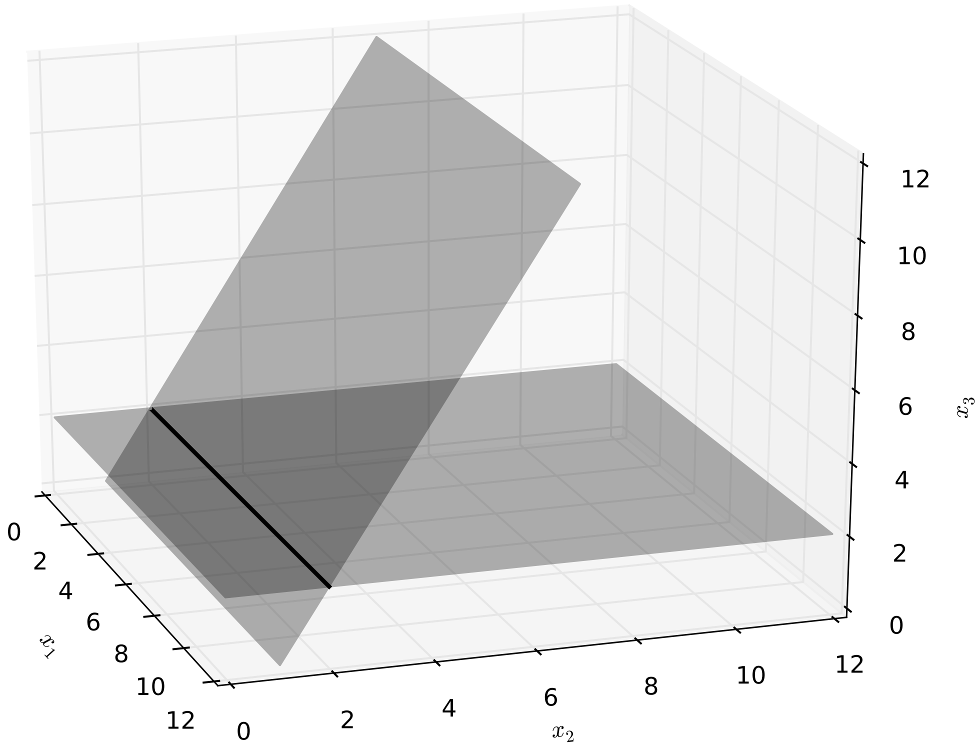


$$
\begin{array}{rcrcr}
-4.0x_1 & + & 2.5x_2 & \leq & -0.25 \\
-3.0x_1 & - & 6.0x_2 & \leq & -43.50 \\
1.5x_1 & - & 4.5x_2 & \leq & 3.00 \\
1.7x_1 & - & 1.7x_2 & \leq & 16.67 \\
4.8x_1 & + & 1.2x_2 & \leq & 82.33 \\
2.0x_1 & + & 3.5x_2 & \leq & 65.75 \\
-3.0x_1 & + & 5.0x_2 & \leq & 29.50
\end{array}
$$

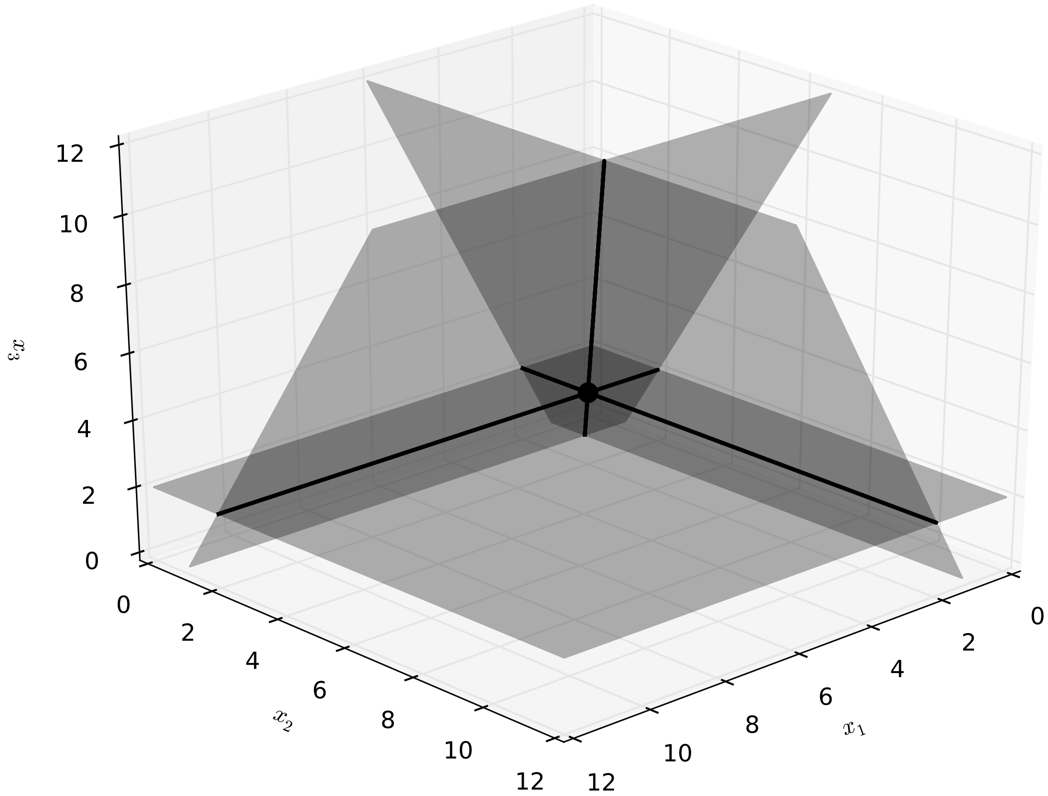
Geometrical view



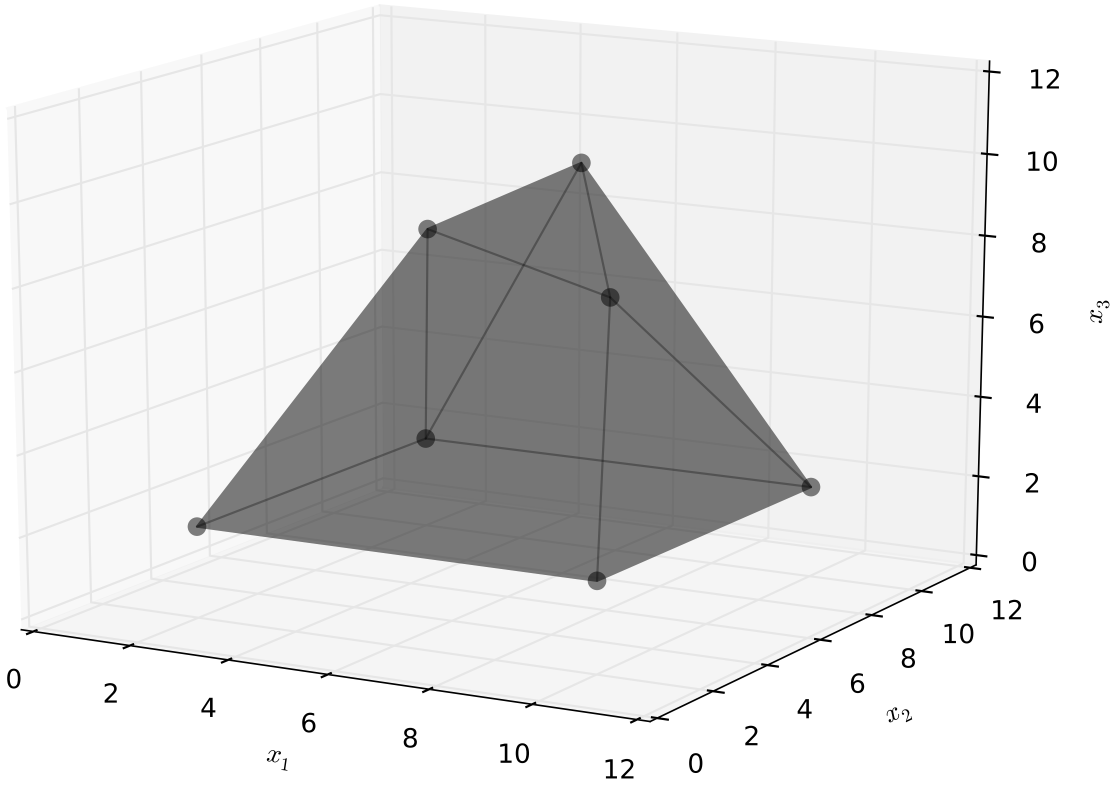
Geometrical view



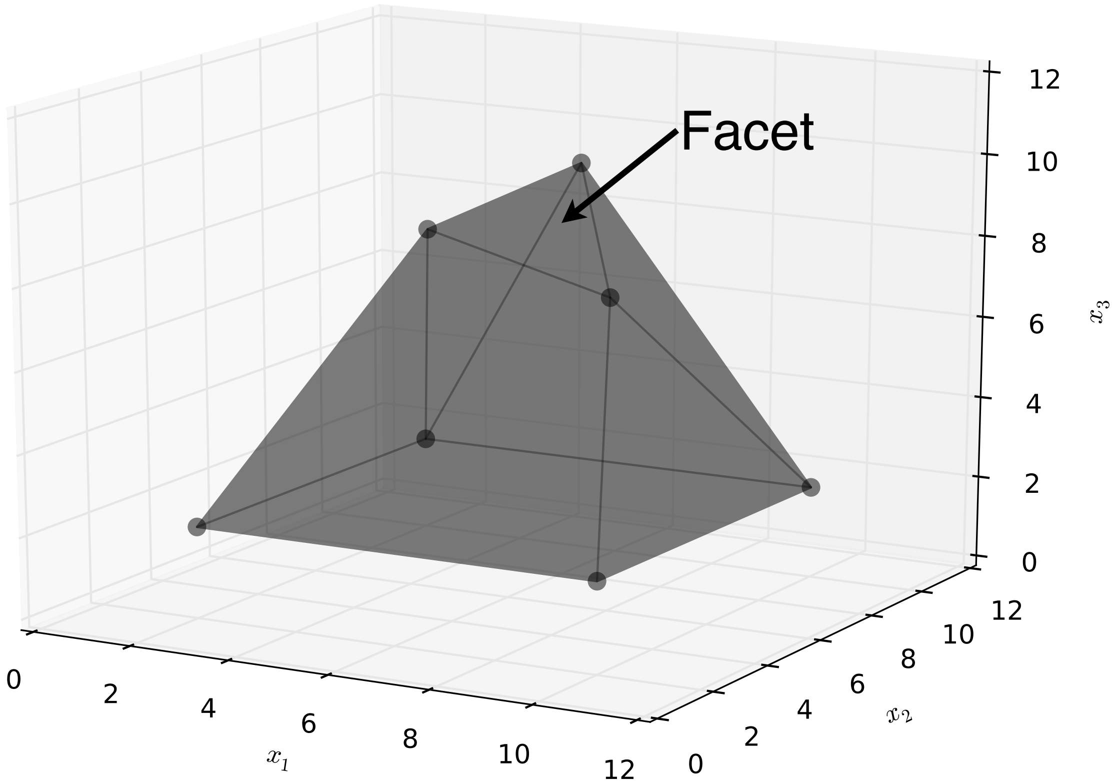
Geometrical view



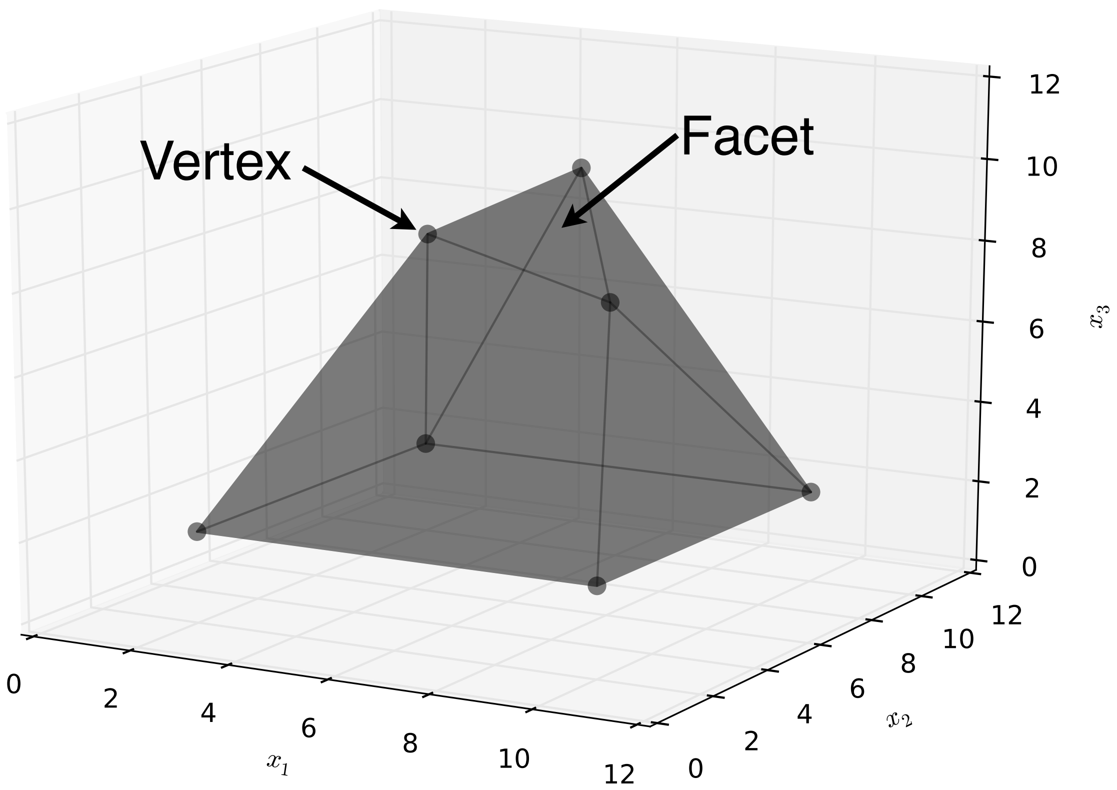
Geometrical view



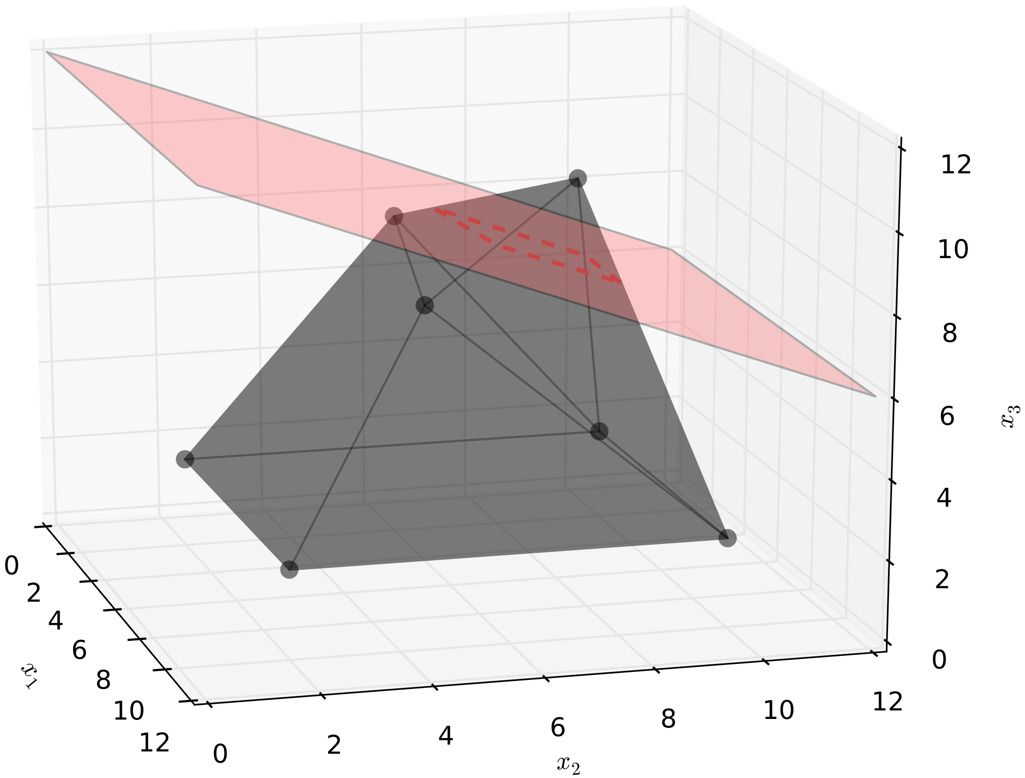
Geometrical view



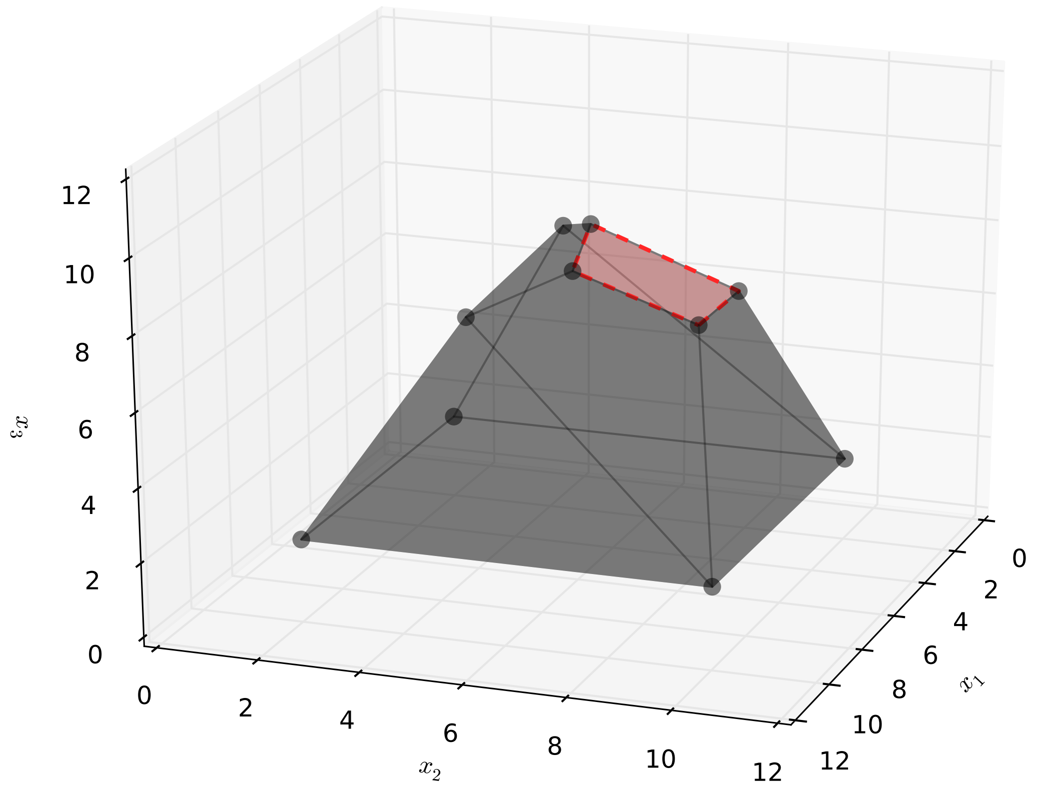
Geometrical view



Geometrical view



Geometrical view

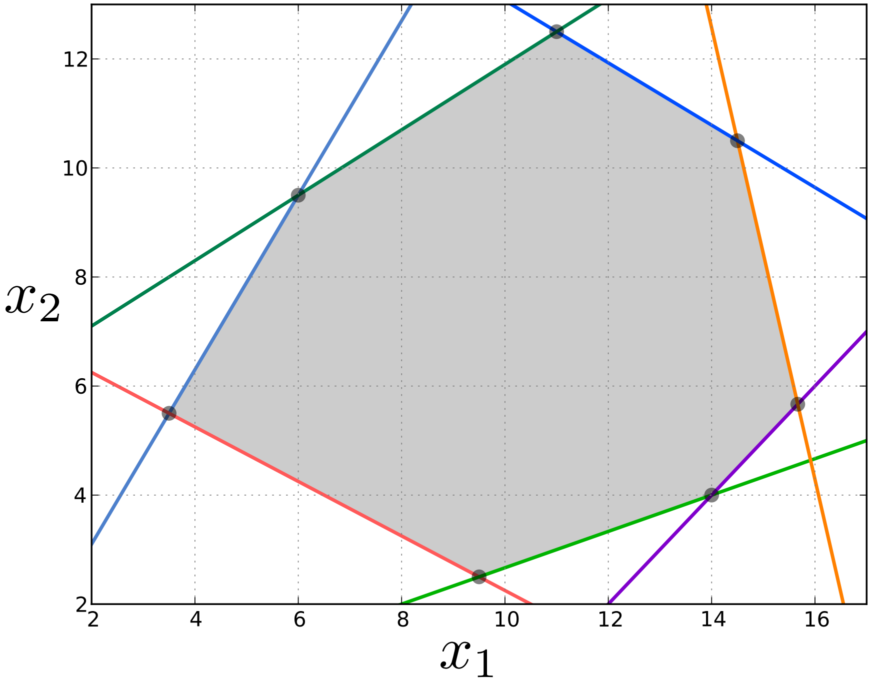


Geometrical view

- Every point in a polytope is a convex combination of its vertices

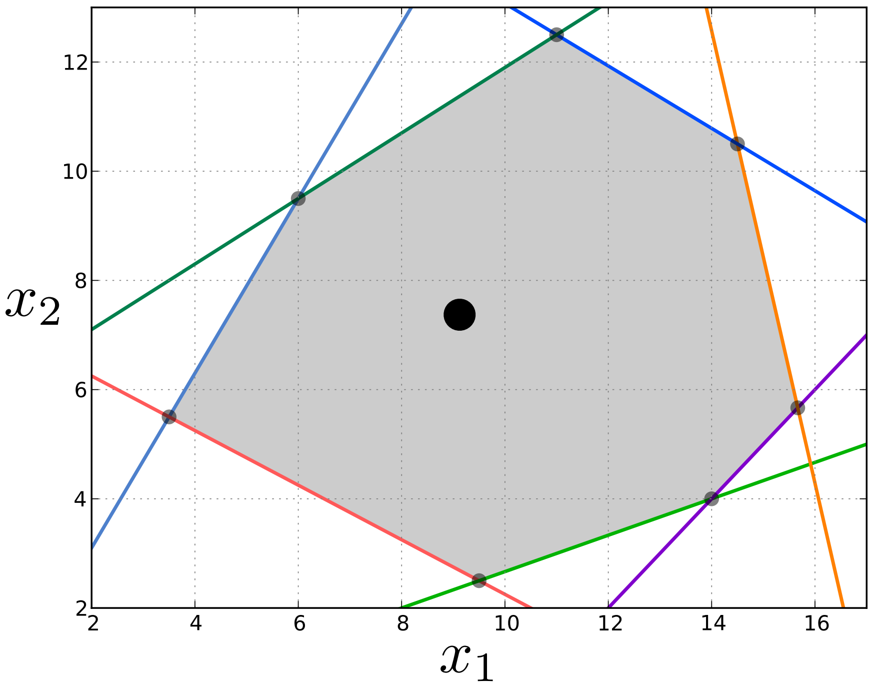
Geometrical view

- Every point in a polytope is a convex combination of its vertices



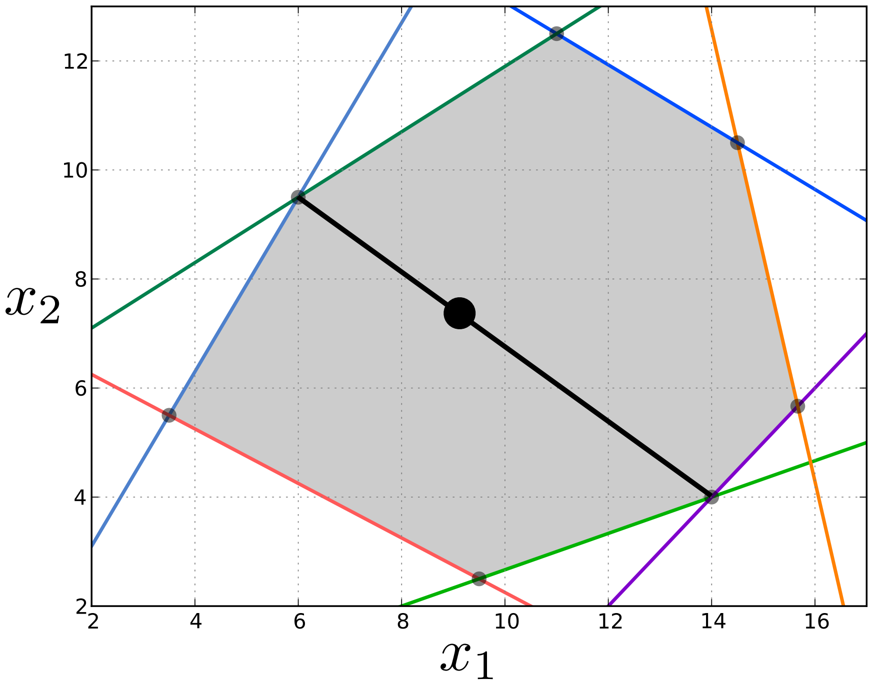
Geometrical view

- Every point in a polytope is a convex combination of its vertices



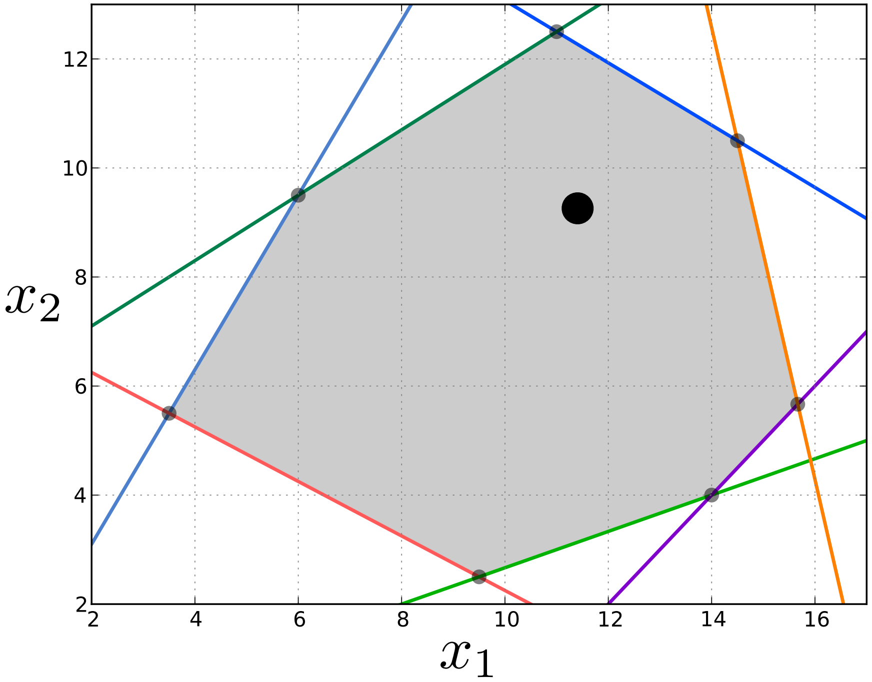
Geometrical view

- Every point in a polytope is a convex combination of its vertices



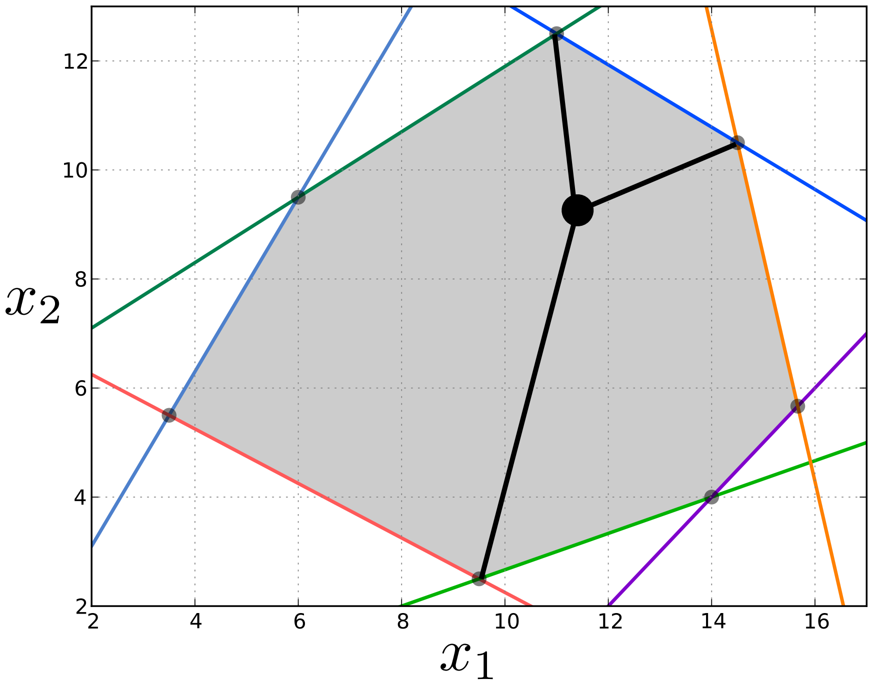
Geometrical view

- Every point in a polytope is a convex combination of its vertices



Geometrical view

- Every point in a polytope is a convex combination of its vertices

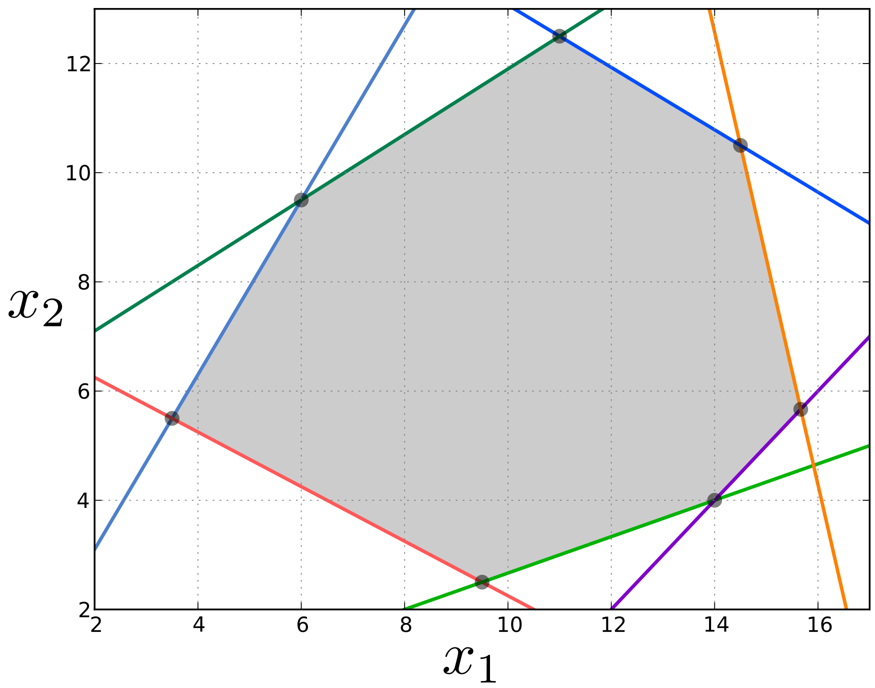


Geometrical view

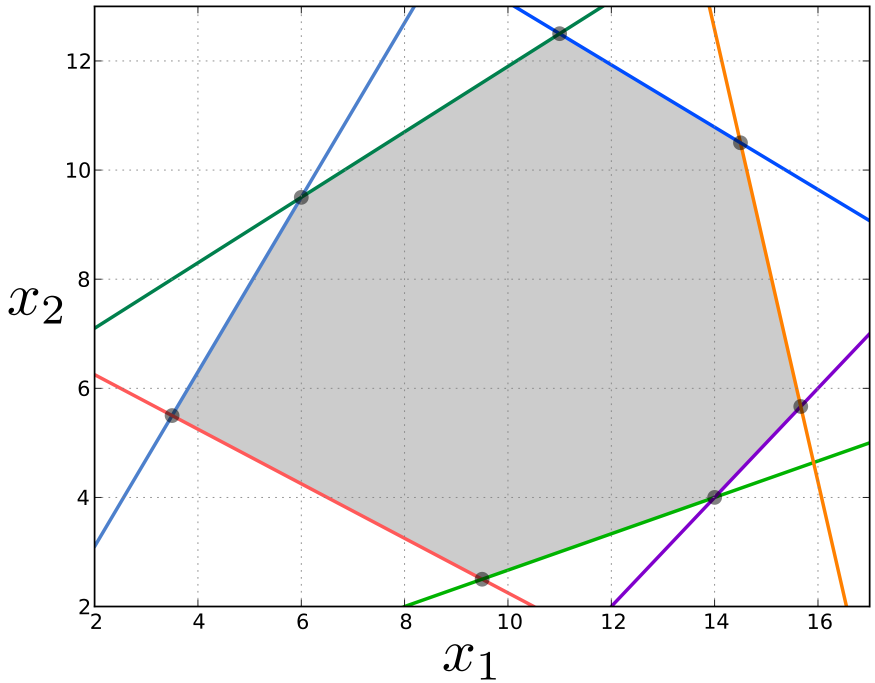
$$
\begin{aligned}
&\min c_1x_1 + \ldots + c_nx_n \\
&\text{subject to} \\
&\quad a_{11}x_1 + \ldots + a_{1n}x_n \leq b_1 \\
&\quad \ldots \\
&\quad a_{m1}x_1 + \ldots + a_{mn}x_n \leq b_m \\
&\\
&\quad x_i \geq 0 \quad (1 \leq i \leq n)
\end{aligned}
$$

- Theorem: At least one of the points where the objective value is minimal is a vertex.

Geometrical view

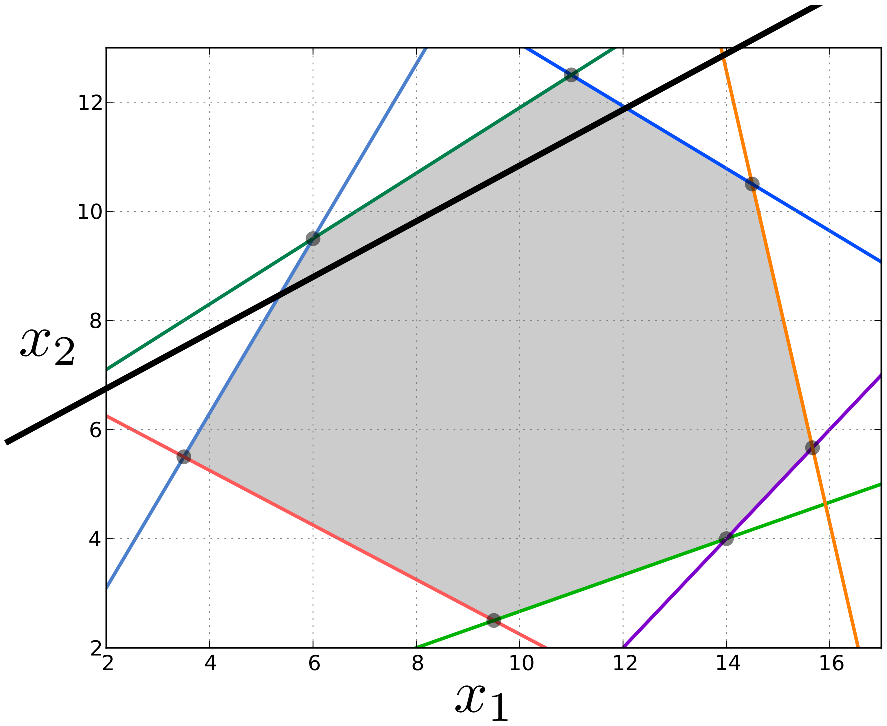


Geometrical view



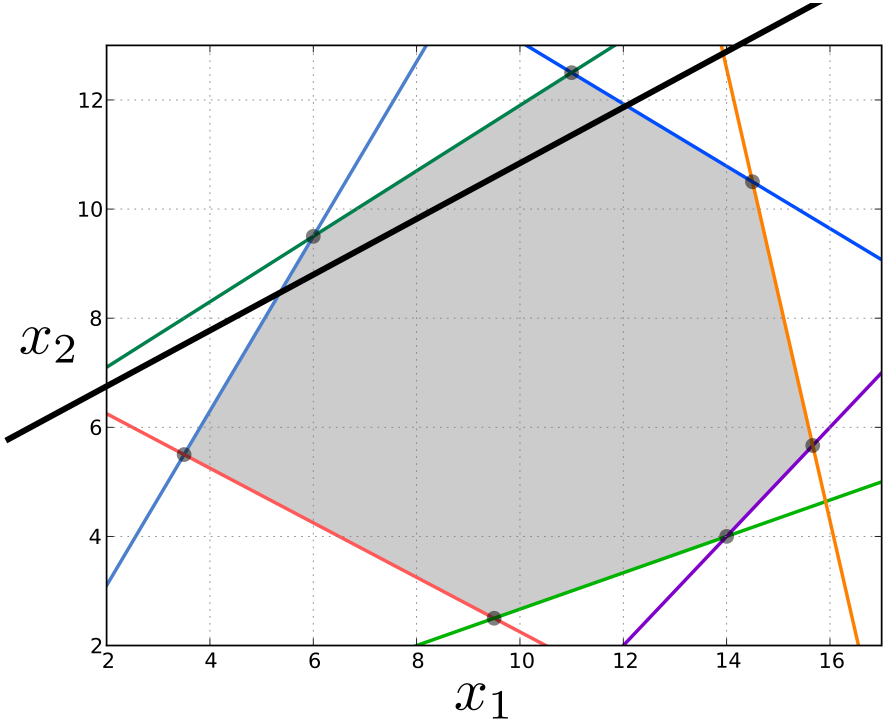
$$c_1 x_1 + \ldots + c_n x_n = b$$

Geometrical view



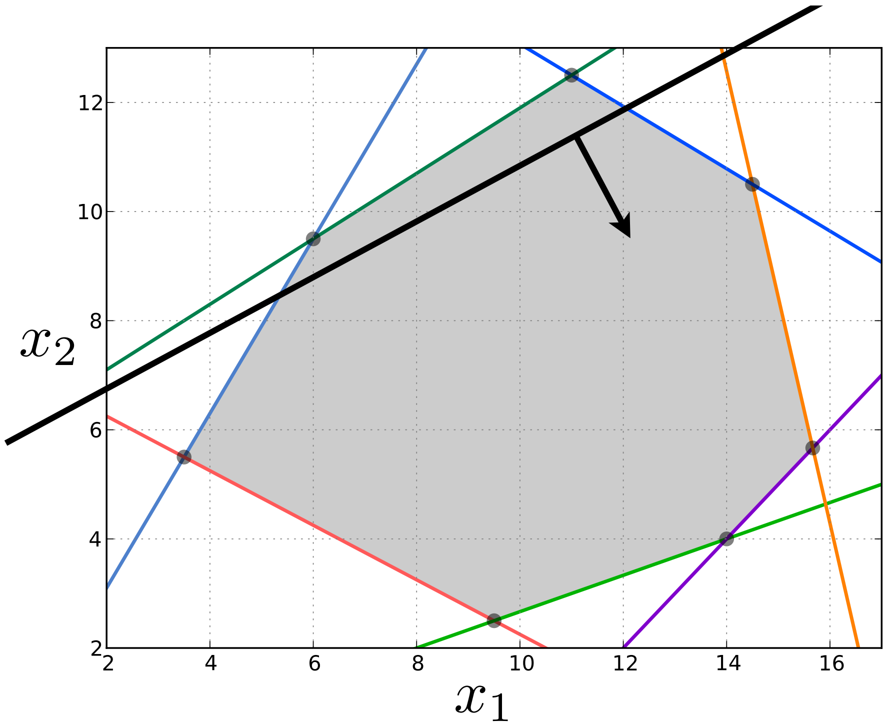
$$c_1 x_1 + \ldots + c_n x_n = b$$

Geometrical view



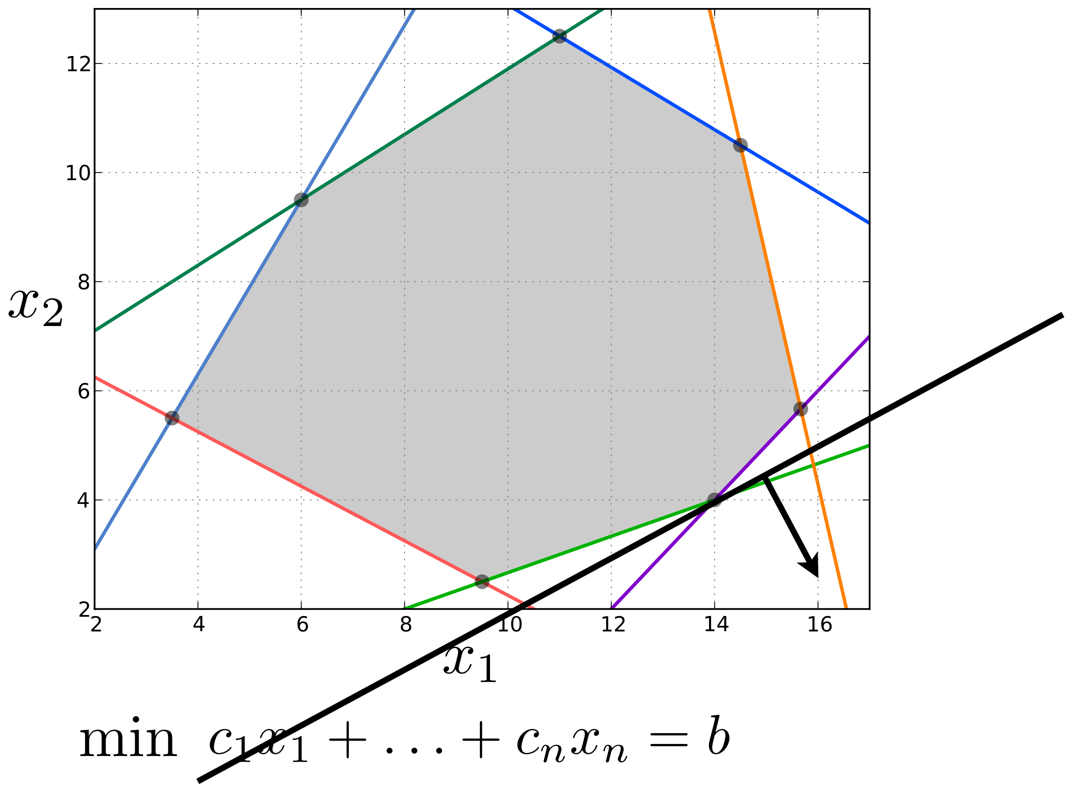
$$\min \; c_1x_1 + \ldots + c_nx_n = b$$

Geometrical view



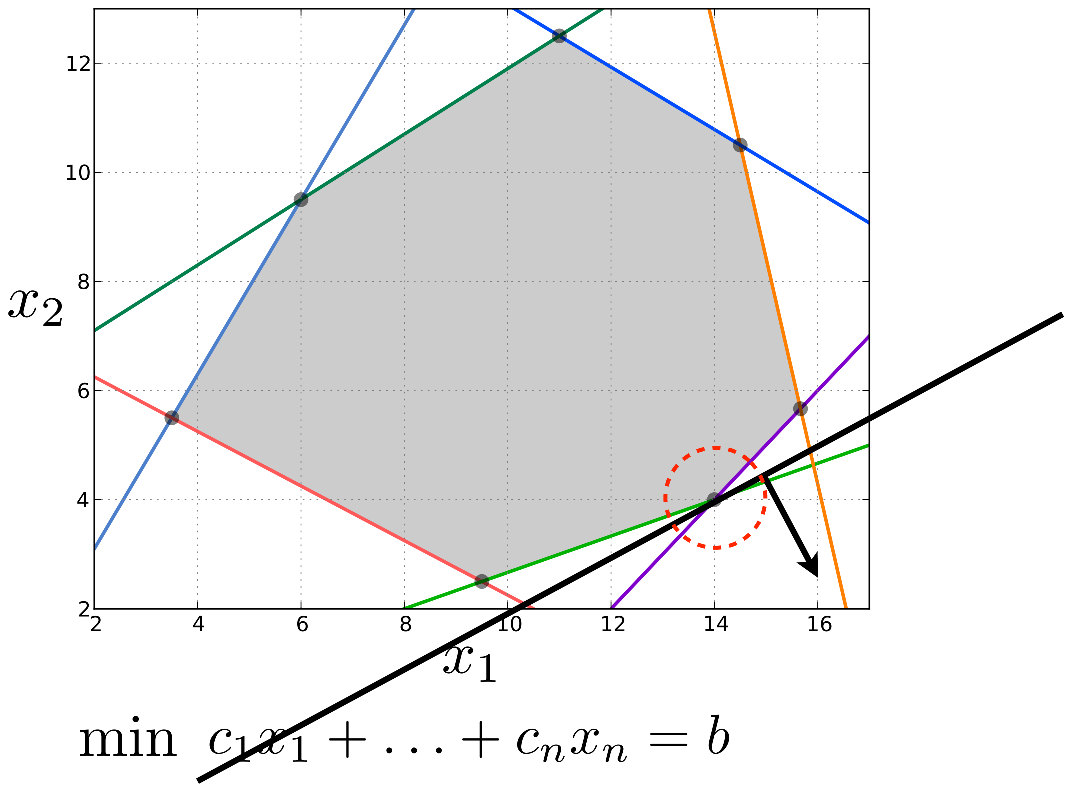
$$\min \;\; c_1 x_1 + \ldots + c_n x_n = b$$

Geometrical view



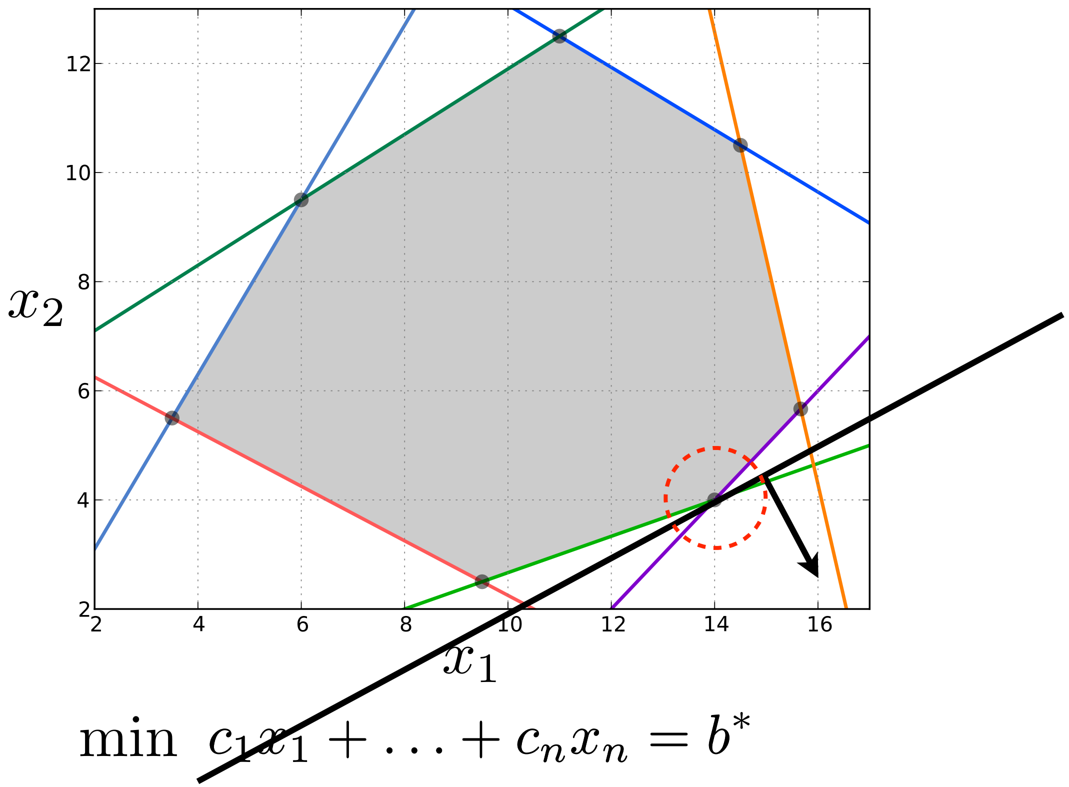
$$\min \; c_1x_1 + \ldots + c_nx_n = b$$

Geometrical view



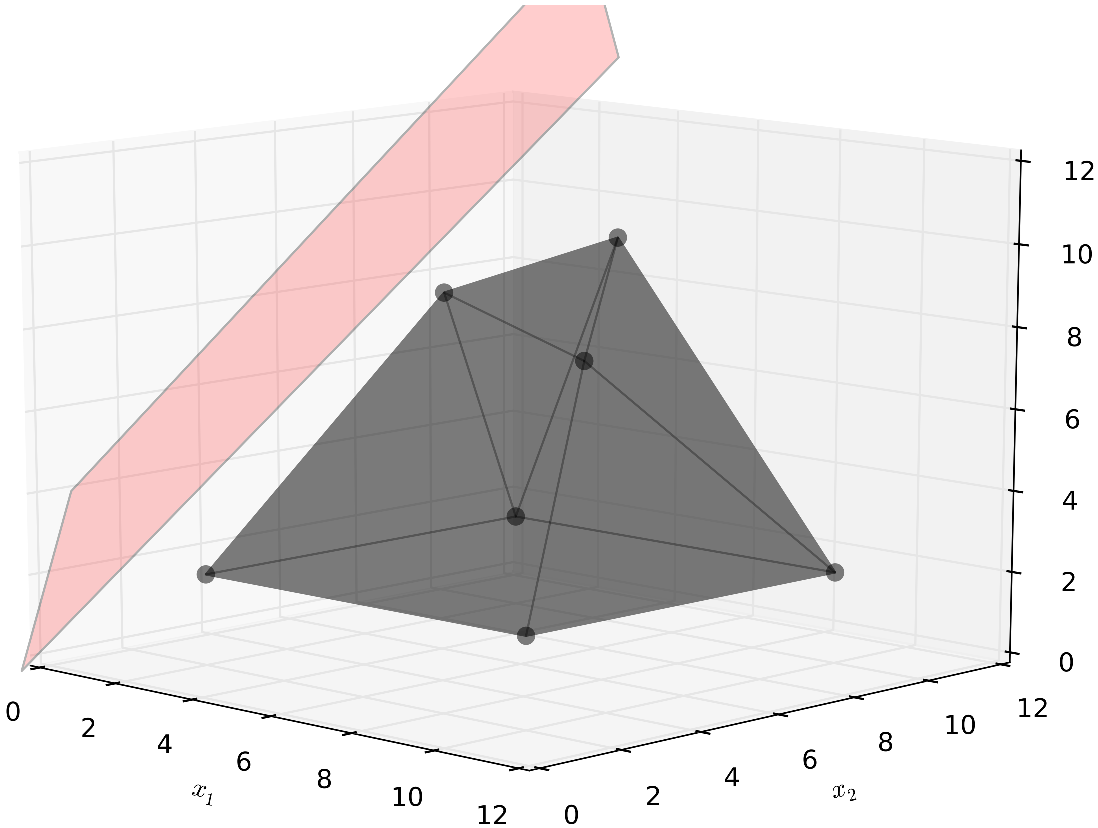
$$\min \; c_1 x_1 + \ldots + c_n x_n = b$$

Geometrical view

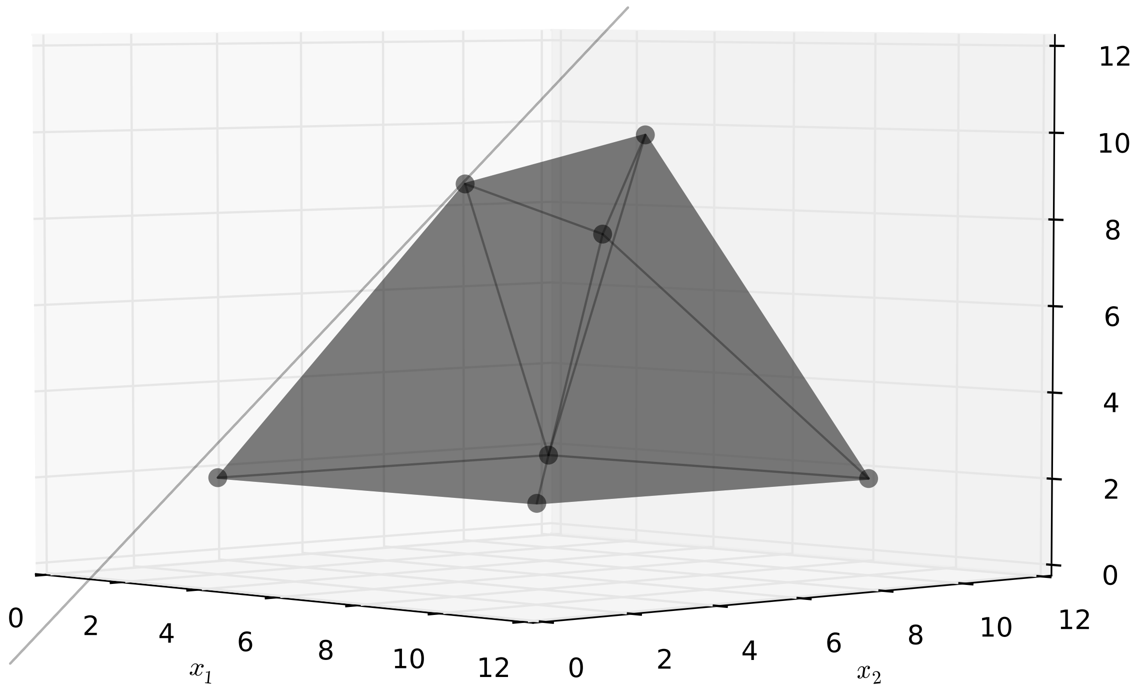


$$\min\ c_1x_1 + \ldots + c_nx_n = b^*$$

Geometrical view



Geometrical view



Geometrical view

- Theorem: At least one of the points where the objective value is minimal is a vertex
- Proof ?

Geometrical view

Let x^* be the minimum. Since each point in a polytope is a convex combination of the vertices $v_1, \ldots, v_t$, we have

$$x^* = \lambda_1 v_1 + \ldots + \lambda_t v_t$$

and the objective value at optimality can be expressed as

$$cx^* = \lambda_1 * (cv_1) + \ldots + \lambda_t (cv_t).$$

Assume that the minimum is not at a vertex, i.e.,

$$cx^* < cv_i \quad \forall i : 1 \leq i \leq t.$$

It follows that

$$\begin{array}{rcl} cx^* & = & \lambda_1 * (cv_1) + \ldots + \lambda_t (cv_t) \\ & > & \lambda_1 * (cx^*) + \ldots + \lambda_t (cx^*) \\ & > & (\lambda_1 + \ldots + \lambda_t)(cx^*) \\ & > & cx^*. \end{array}$$

Hence, it must be the case that $x^* = v_i$ for some $1 \leq i \leq t$.

Geometrical view

- How to solve a linear program?
  - Enumerate all the vertices
  - Select the one with the smallest objective value

Algebraic view

- The simplex algorithm
  - A more intelligent way of exploring the vertices
- Invented by G. Dantzig
- Exponential worst-case, but works well in practice

Simplex algorithm

- Outline
  1. An optimal solution is at a vertex
  2. A vertex is a basic feasible solution (BFS)
  3. You can move from one BFS to a neighboring BFS
  4. You can detect whether a BFS is optimal
  5. From any BFS, you can move to a BFS with better a cost

Basic feasible solution (BFS)

$$
\begin{aligned}
&\min c_1x_1 + \ldots + c_nx_n \\
&\text{subject to} \\
&\quad a_{11}x_1 + \ldots + a_{1n}x_n \le b_1 \\
&\quad \ldots \\
&\quad a_{m1}x_1 + \ldots + a_{mn}x_n \le b_m \\
&\\
&\quad x_i \ge 0 \quad (1 \le i \le n)
\end{aligned}
$$

Basic feasible solution (BFS)

Goal: How to find solutions to linear systems

$$a_{11}x_1 + \ldots + a_{1n}x_n = b_1$$

$$\ldots$$

$$a_{m1}x_1 + \ldots + a_{mn}x_n = b_m$$

$$x_i \geq 0 \quad (1 \leq i \leq n)$$

Basic feasible solution (BFS)

Basic Solution

$$x_1 = b_1 + \sum_{i=m+1}^{n} a_{1i}\, x_i$$

$$\ldots$$

$$x_m = b_m + \sum_{i=m+1}^{n} a_{mi}\, x_i$$

Basic feasible solution (BFS)

$\boxed{\text{Basic Solution}}$

$$
\begin{aligned}
x_1 &= b_1 + \sum_{i=m+1}^{n} a_{1i}\, x_i \\
&\cdots \\
x_m &= b_m + \sum_{i=m+1}^{n} a_{mi}\, x_i
\end{aligned}
$$

Basic Variables

Basic feasible solution (BFS)

Basic Solution

$$x_1 = b_1 + \sum_{i=m+1}^{n} a_{1i}\, x_i$$

$$\cdots$$

$$x_m = b_m + \sum_{i=m+1}^{n} a_{mi}\, x_i$$

Basic Variables

Non Basic Variables

Basic feasible solution (BFS)

Basic Solution

$$x_1 = b_1 + \sum_{i=m+1}^{n} a_{1i}\, x_i$$

$$\cdots$$

$$x_m = b_m + \sum_{i=m+1}^{n} a_{mi}\, x_i$$

Assign to the b's

Assign to zero

Basic feasible solution (BFS)

Basic Solution

$$x_1 = b_1 + \sum_{i=m+1}^{n} a_{1i}\, x_i$$

$$\ldots$$

$$x_m = b_m + \sum_{i=m+1}^{n} a_{mi}\, x_i$$

$$\{x_i = b_i | 1 \le i \le m\} \cup \{x_i = 0 | m+1 \le i \le n\}$$

Basic feasible solution (BFS)

Goal: How to find solutions to linear systems

$$a_{11}x_1 + \ldots + a_{1n}x_n = b_1$$

$$\ldots$$

$$a_{m1}x_1 + \ldots + a_{mn}x_n = b_m$$

$$x_i \geq 0 \quad (1 \leq i \leq n)$$

Basic feasible solution (BFS)

Basic Solution

$$x_1 = b_1 + \sum_{i=m+1}^{n} a_{1i}\, x_i$$

$$\dots$$

$$x_m = b_m + \sum_{i=m+1}^{n} a_{mi}\, x_i$$

Basic Variables

Non Basic Variables

Basic feasible solution (BFS)

Basic Feasible Solution

$$x_1 = \boxed{b_1} + \sum_{i=m+1}^{n} a_{1i}\, x_i$$

$$\dots$$

$$x_m = \boxed{b_m} + \sum_{i=m+1}^{n} a_{mi}\, x_i$$

Feasible if $\forall i \in 1..m : b_i \geq 0$

Basic feasible solution (BFS)

- Select m variables
  - Will be the basic variables
- Re-write them using only non-basic variables
  - Gaussian elimination
- If all the b's are non-negative
  - We have a BFS

Basic feasible solution (BFS)

- But we do not have equalities

$$
\begin{aligned}
&\mathrm{a}_{11}\ x_1 + \ldots + a_{1n}\ x_n \leq b_1 \\
&\ldots \\
&a_{m1}x_1 + \ldots + a_{mn}x_n \leq b_m
\end{aligned}
$$

Basic feasible solution (BFS)

- But we do not have equalities

$$
\begin{aligned}
&a_{11}\ x_1 + \ldots + a_{1n}\ x_n \leq b_1 \\
&\ldots \\
&a_{m1}x_1 + \ldots + a_{mn}x_n \leq b_m
\end{aligned}
$$

$$\downarrow$$

$$
\begin{aligned}
&a_{11}\ x_1 + \ldots + a_{1n}\ x_n + \boxed{s_1} = b_1 \\
&\ldots \\
&a_{m1}x_1 + \ldots + a_{mn}x_n + \boxed{s_n} = b_m
\end{aligned}
$$

Slack variables

$$s_1, \ldots, s_m \geq 0$$

Basic feasible solution (BFS)

- Re-write the constraints as equalities
  - With slack variables
- Select m variables
  - Will be the basic variables
- Re-write them using only non-basic variables
  - Gaussian elimination
- If all the b's are non-negative
  - We have a BFS

Naïve algorithm

- Generate all basic feasible solutions
  - i.e., select m basic variables, perform Gaussian elim.
  - test whether it is feasible
- Select the BFS with the best cost

- But, $\dfrac{n!}{m!(n-m)!}$

Can we explore this space more efficiently?

Simplex algorithm

- Outline
  1. An optimal solution is at a vertex
  2. A vertex is a basic feasible solution (BFS)
  3. You can move from one BFS to a neighboring BFS
  4. You can detect whether a BFS is optimal
  5. From any BFS, you can move to a BFS with better a cost

Simplex algorithm

- Local search algorithm
- Move from BFS to BFS
- Guaranteed to find a global optimum
  - Due to convexity

Key idea: How do we do this move operation?

Simplex algorithm

$$
\begin{array}{rcrcrcrcrcrcr}
3x_1 & - & 2x_2 & + & x_3 & & & & & & & = & 1 \\
2x_1 & & & & & + & x_4 & & & + & x_6 & = & 2 \\
x_1 & & & & & & & + & x_5 & + & x_6 & = & 3
\end{array}
$$

$$
\boxed{
\begin{array}{rcrcrcrcr}
x_3 & = & 1 & - & 3x_1 & - & 2x_2 & & \\
x_4 & = & 2 & - & 2x_1 & & & + & x_6 \\
x_5 & = & 3 & - & x_1 & & & + & x_6
\end{array}
}
$$

Simplex algorithm

$$
\begin{array}{lclclclcl}
x_3 & = & 1 & - & 3x_1 & - & 2x_2 & & \\
x_4 & = & 2 & - & 2x_1 & & & + & x_6 \\
x_5 & = & 3 & - & x_1 & & & + & x_6
\end{array}
$$

- Move to another BFS (local search)
  - Select a non-basic variable with negative coeff. (entering variable)
  - Replace a basic variable with this selection (leaving variable)
  - Perform Gaussian elimination

Simplex algorithm

$$\begin{array}{lclclclcl} x_3 & = & 1 & - & 3x_1 & \boxed{- \quad 2x_2} & & \\ x_4 & = & 2 & - & 2x_1 & & + & x_6 \\ x_5 & = & 3 & - & x_1 & & + & x_6 \end{array}$$

$$\downarrow$$

$$\begin{array}{lclclclclcl} \boxed{x_2 = \frac{1}{2} - \frac{3}{2}x_1 - \frac{1}{2}x_3} & & & & & & & & & & \\ x_4 & = & 2 & - & 2x_1 & & & + & x_6 \\ x_5 & = & 3 & - & x_1 & & & + & x_6 \end{array}$$

Simplex algorithm

$$\begin{aligned}
x_3 &= 1 \boxed{- 3x_1} - 2x_2 \\
x_4 &= 2 \boxed{- 2x_1} + x_6 \\
x_5 &= 3 \boxed{- x_1} + x_6
\end{aligned}$$

Simplex algorithm

$$
\begin{aligned}
x_3 &= 1 - 3x_1 - 2x_2 \\
x_4 &= 2 - 2x_1 + x_6 \\
x_5 &= 3 - x_1 + x_6
\end{aligned}
$$

Simplex algorithm

$$
\begin{aligned}
x_3 &= 1 - 3x_1 - 2x_2 \\
x_4 &= 2 - 2x_1 + x_6 \\
x_5 &= 3 - x_1 + x_6
\end{aligned}
$$

$$\Downarrow$$

$$
\begin{aligned}
x_3 &= -8 - 2x_2 + 3x_5 - 3x_6 \\
x_4 &= -4 + 2x_5 - x_6 \\
x_1 &= 3 - x_5 + x_6
\end{aligned}
$$

Simplex algorithm

$$
\begin{array}{rcrrrrrrr}
x_3 & = & 1 & - & 3x_1 & - & 2x_2 & & \\
x_4 & = & 2 & - & 2x_1 & & & + & x_6 \\
x_5 & = & 3 & - & x_1 & & & + & x_6
\end{array}
$$

$$\downarrow$$

$$
\begin{array}{rcrrrrrrr}
x_3 & = & -8 & - & 2x_2 & + & 3x_5 & - & 3x_6 \\
x_4 & = & -4 & & & + & 2x_5 & - & x_6 \\
x_1 & = & 3 & & & - & x_5 & + & x_6
\end{array}
$$

Not a BFS: I cannot select the leaving variable arbitrarily!

Simplex algorithm

$$\begin{array}{rcrrrrrr} x_3 & = & 1 & - & 3x_1 & - & 2x_2 & & \\ x_4 & = & 2 & - & 2x_1 & & & + & x_6 \\ x_5 & = & 3 & - & x_1 & & & + & x_6 \end{array} \qquad \begin{array}{c} \frac{1}{3} \\ 1 \\ 3 \end{array}$$

- How to choose the leaving variable?

– we must maintain feasibility

$$l = \underset{i:a_{ie}<0}{\text{arg-min}} \frac{b_i}{-a_{ie}}$$

Simplex algorithm

$$\begin{array}{rcrrrrrrr} x_3 & = & 1 & - & 3x_1 & - & 2x_2 & & \\ x_4 & = & 2 & - & 2x_1 & & & + & x_6 \\ x_5 & = & 3 & - & x_1 & & & + & x_6 \end{array} \qquad \begin{array}{c} \frac{1}{3} \\ 1 \\ 3 \end{array}$$

$$\begin{array}{rcrrrrrrr} x_1 & = & \frac{1}{3} & - & \frac{2}{3}x_2 & - & \frac{1}{3}x_3 & & \\ x_4 & = & \frac{4}{3} & + & \frac{4}{3}x_2 & + & \frac{2}{3}x_3 & + & x_6 \\ x_5 & = & \frac{8}{3} & + & \frac{2}{3}x_2 & + & \frac{1}{3}x_3 & + & x_6 \end{array}$$

Simplex algorithm

Move to another BFS:

- Select the entering variable x_e
  - Non-basic variable with negative coeff.
- Select the leaving variable x_l to maintain feasibility
- Apply Gaussian elimination
  - i.e., eliminate x_e from the the right hand side
- Pivot(e,l)

Simplex algorithm

- Outline
  1. An optimal solution is at a vertex
  2. A vertex is a basic feasible solution (BFS)
  3. You can move from one BFS to a neighboring BFS
  4. You can detect whether a BFS is optimal
  5. From any BFS, you can move to a BFS with better a cost

Simplex algorithm

- A BFS is optimal if the objective function, after eliminating all the basic variables is

$$c_0 + c_1 x_1 + \ldots + c_n x_n$$

$$c_i \geq 0 \quad (1 \leq i \leq n).$$

Simplex algorithm

- Example

$$
\begin{array}{lrcrcrcrcrcl}
\text{min} & x_1 & + & x_2 & + & x_3 & + & x_4 & + & x_5 & & \\
\text{subject to} & & & & & & & & & & & \\
& 3x_1 & + & 2x_2 & + & x_3 & & & & & = & 1 \\
& 5x_1 & + & x_2 & + & x_3 & + & x_4 & & & = & 3 \\
& 2x_1 & + & 5x_2 & + & x_3 & & & + & x_5 & = & 4
\end{array}
$$

Simplex algorithm

- Example

$$
\begin{array}{lrcrcrcrcrcl}
\text{min} & x_1 & + & x_2 & + & x_3 & + & x_4 & + & x_5 & & \\
\text{subject to} & & & & & & & & & & & \\
& 3x_1 & + & 2x_2 & + & x_3 & & & & & = & 1 \\
& 5x_1 & + & x_2 & + & x_3 & + & x_4 & & & = & 3 \\
& 2x_1 & + & 5x_2 & + & x_3 & & & + & x_5 & = & 4
\end{array}
$$

$$\downarrow$$

$$
\begin{array}{lrcrcrcr}
\text{min} & & & \boxed{6 - 3x_1 - 3x_2} & & & & \\
\text{subject to} & & & & & & & \\
& x_3 & = & 1 & - & 3x_1 & - & 2x_2 \\
& x_4 & = & 2 & - & 2x_1 & + & x_2 \\
& x_5 & = & 3 & + & x_1 & - & 3x_2
\end{array}
$$

Simplex algorithm

$$
\begin{array}{lrcrcrcr}
\text{min} & & & 6 & - & 3x_1 & - & 3x_2 \\
\text{subject to} & & & & & & & \\
& x_3 & = & 1 & - & 3x_1 & - & 2x_2 \\
& x_4 & = & 2 & - & 2x_1 & + & x_2 \\
& x_5 & = & 3 & + & x_1 & - & 3x_2
\end{array}
$$

Simplex algorithm

$$
\begin{array}{llrcrcr}
\text{min} & & 6 & - & 3x_1 & - & 3x_2 \\
\text{subject to} & & & & & & \\
& x_3 = & 1 & - & 3x_1 & - & 2x_2 \\
& x_4 = & 2 & - & 2x_1 & + & x_2 \\
& x_5 = & 3 & + & x_1 & - & 3x_2
\end{array}
$$

Simplex algorithm

$$
\begin{array}{lrcrcrcr}
\text{min} & & & 6 & - & 3x_1 & - & 3x_2 \\
\text{subject to} & & & & & & & \\
& x_3 & = & 1 & - & 3x_1 & - & 2x_2 \\
& x_4 & = & 2 & - & 2x_1 & + & x_2 \\
& x_5 & = & 3 & + & x_1 & - & 3x_2
\end{array}
$$

Simplex algorithm

$$
\begin{array}{lrcrcrcr}
\text{min} & & & 6 & - & 3x_1 & - & 3x_2 \\
\text{subject to} & & & & & & & \\
& x_3 & = & 1 & - & 3x_1 & - & 2x_2 \\
& x_4 & = & 2 & - & 2x_1 & + & x_2 \\
& x_5 & = & 3 & + & x_1 & - & 3x_2
\end{array}
$$

$$\downarrow$$

$$
\begin{array}{lrcrcrcr}
\text{min} & & & \frac{9}{2} & + & \frac{3}{2}x_1 & + & \frac{3}{2}x_3 \\
\text{subject to} & & & & & & & \\
& x_2 & = & \frac{1}{2} & - & \frac{3}{2}x_1 & - & \frac{1}{2}x_3 \\
& x_4 & = & \frac{5}{2} & - & \frac{7}{2}x_1 & - & \frac{1}{2}x_3 \\
& x_5 & = & \frac{3}{2} & + & \frac{11}{2}x_1 & + & \frac{3}{2}x_3
\end{array}
$$

Simplex algorithm

$$\begin{array}{lrcrcrcr}
\text{min} & & & 6 & - & 3x_1 & - & 3x_2 \\
\text{subject to} & & & & & & & \\
& x_3 & = & 1 & - & 3x_1 & - & 2x_2 \\
& x_4 & = & 2 & - & 2x_1 & + & x_2 \\
& x_5 & = & 3 & + & x_1 & - & 3x_2
\end{array}$$

$$\begin{array}{lrcrcrcr}
\text{min} & & & \boxed{\frac{9}{2} + \frac{3}{2}x_1 + \frac{3}{2}x_3} & & & & \\
\text{subject to} & & & & & & & \\
& x_2 & = & \frac{1}{2} & - & \frac{3}{2}x_1 & - & \frac{1}{2}x_3 \\
& x_4 & = & \frac{5}{2} & - & \frac{7}{2}x_1 & - & \frac{1}{2}x_3 \\
& x_5 & = & \frac{3}{2} & + & \frac{11}{2}x_1 & + & \frac{3}{2}x_3
\end{array}$$

Simplex algorithm

- Outline
  1. An optimal solution is at a vertex
  2. A vertex is a basic feasible solution (BFS)
  3. You can move from one BFS to a neighboring BFS
  4. You can detect whether a BFS is optimal
  5. From any BFS, you can move to a BFS with better a cost

Simplex algorithm

- Local move improves the objective value
- In a BFS, assume that
  - $b_1 > 0, b_2 > 0, \ldots, b_m > 0$
  - there exists an entering variable e with $c_e < 0$
- Then, the move pivot(e,l) is improving

Simplex algorithm

$$
\begin{aligned}
&\text{while } \exists 1 \le i \le n : c_i < 0 \text{ do} \\
&\quad \text{choose } e \text{ such that } c_e < 0; \\
&\quad l = \operatorname*{arg-min}_{i:a_{ie}<0} \frac{b_i}{-a_{ie}}; \\
&\quad \text{pivot}(e,l);
\end{aligned}
$$

- If
  - $b_1, b_2, \ldots, b_m$ are strictly positive
  - objective function is bounded below
- Algorithm terminates with an optimal solution

Simplex algorithm

- Selecting the leaving variable

$$l = \underset{i:a_{ie}<0}{\text{arg-min}} \frac{b_i}{-a_{ie}}$$

- No leaving variable!

$$\begin{array}{lrclcrcr}
\text{min} & & & 6 & - & 3x_1 & - & 3x_2 \\
\text{subject to} & & & & & & & \\
& x_3 & = & 1 & + & 3x_1 & - & 2x_2 \\
& x_4 & = & 2 & + & 2x_1 & + & x_2 \\
& x_5 & = & 3 & + & x_1 & - & 3x_2
\end{array}$$

Simplex algorithm

$$
\begin{array}{llcrcrcr}
\text{min} & & & 6 & - & 3x_1 & - & 3x_2 \\
\text{subject to} & & & & & & & \\
& x_3 & = & 1 & + & 3x_1 & - & 2x_2 \\
& x_4 & = & 2 & + & 2x_1 & + & x_2 \\
& x_5 & = & 3 & + & x_1 & - & 3x_2
\end{array}
$$

- Basic solution

$$\{x_1 = 0; x_2 = 0; x_3 = 1; x_4 = 2; x_5 = 3\}$$

- What if I increase the value of x_1
  - Solution remains feasible, but
  - Value of objective function decreases

Simplex algorithm

- Another issue: What if some b_i becomes zero?

$$\begin{array}{lrcrcrcr}
\min & & & 5 & - & 3x_1 & - & 3x_2 \\
\text{subject to} & & & & & & & \\
& x_3 & = & 0 & - & 3x_1 & - & 2x_2 \\
& x_4 & = & 2 & - & 2x_1 & + & x_2 \\
& x_5 & = & 3 & + & x_1 & - & 3x_2
\end{array}$$

- Leaving variable: x_2

$$\begin{array}{lrcrcrcr}
\min & & & \boxed{5} & + & \frac{3}{2}x_1 & + & \frac{3}{2}x_3 \\
\text{subject to} & & & & & & & \\
& x_2 & = & 0 & - & \frac{3}{2}x_1 & - & \frac{1}{2}x_3 \\
& x_4 & = & 2 & - & \frac{7}{2}x_1 & - & \frac{1}{2}x_3 \\
& x_5 & = & 3 & + & \frac{11}{2}x_1 & + & \frac{3}{2}x_3
\end{array}$$

Simplex algorithm

- Outline
  1. An optimal solution is at a vertex
  2. A vertex is a basic feasible solution (BFS)
  3. You can move from one BFS to a neighboring BFS
  4. You can detect whether a BFS is optimal
  5. From any BFS, you can move to a BFS with better a cost

Need to find a new way to guarantee termination

Simplex algorithm

- Termination
  - Bland rule: select always the first entering variable lexicographically
  - Lexicographic pivoting rule: break ties when selecting the leaving variable
  - Pertubation methods

Simplex algorithm

- How do I find my first BFS ?

$$
\begin{array}{lrcccccc}
\text{min} & c_1x_1 & + & \dots & + & c_nx_n & & \\
\text{subject to} & & & & & & & \\
 & a_{11}x_1 & + & \dots & + & a_{1n}x_n & = & b_1 \\
 & \dots & & & & & & \\
 & a_{m1}x_1 & + & \dots & + & a_{mn}x_n & = & b_m
\end{array}
$$

- Introduce artificial variables

Simplex algorithm

$$
\begin{array}{lrcccccccccc}
\min & c_1 x_1 & + & \ldots & + & c_n x_n & & & & & & \\
\text{subject to} & & & & & & & & & & & \\
& a_{11} x_1 & + & \ldots & + & a_{1n} x_n & + & y_1 & & & = & b_1 \\
& \ldots & & & & & & & & & & \\
& a_{i1} x_1 & + & \ldots & + & a_{in} x_n & & & + \; y_i & & = & b_i \\
& \ldots & & & & & & & & & & \\
& a_{m1} x_1 & + & \ldots & + & a_{mn} x_n & & & & + \; y_m & = & b_m
\end{array}
$$

- Easy BFS 

- But wrt another problem 

Simplex algorithm

- Two-phase strategy
  1. First find a BFS (if one exists)

  2. Then, find optimal BFS

Simplex algorithm

1. Find a BFS

$$
\begin{array}{lccccccccccccc}
\boxed{\min & & & & & & y_1 & + & \dots & + & y_m} & & \\
\text{subject to} & & & & & & & & & & & & \\
& a_{11}x_1 & + & \dots & + & a_{1n}x_n & + & y_1 & & & & = & b_1 \\
& \dots & & & & & & & & & & & \\
& a_{i1}x_1 & + & \dots & + & a_{in}x_n & & & + & y_i & & = & b_i \\
& \dots & & & & & & & & & & & \\
& a_{m1}x_1 & + & \dots & + & a_{mn}x_n & & & & & + & y_m = & b_m
\end{array}
$$

- Feasible if the objective value is 0
  - i.e., all y_i's are 0, and there is a BFS without y_i (almost always)

Simplex algorithm