DM545

Linear and Integer Programming

Lecture 6

Revised Simplex Method

Marco Chiarandini

Department of Mathematics & Computer Science
University of Southern Denmark

Outline

1. Geometric Interpretation

2. Farkas Lemma

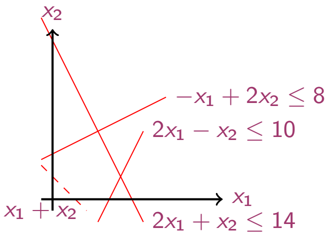
3. Revised Simplex Method

Outline

1. Geometric Interpretation
2. Farkas Lemma
3. Revised Simplex Method

Geometric Interpretation

$$\begin{aligned} \max \quad & x_1 + x_2 \\ & 2x_1 + x_2 \leq 14 \\ & -x_1 + 2x_2 \leq 8 \\ & 2x_1 - x_2 \leq 10 \\ & x_1, x_2 \geq 0 \end{aligned}$$



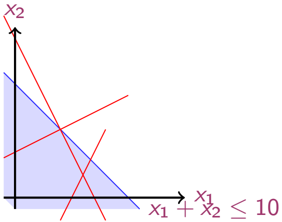
Opt $x^* = (4, 6)$, $z^* = 10$. To prove this we need to prove that $y^* = (3/5, 1/5, 0)$ is a feasible solution of D:

$$\begin{aligned} \min \; 14y_1 + 8y_2 + 10y_3 &= w \\ 2y_1 - y_2 + 2y_3 &\geq 1 \\ y_1 + 2y_2 - y_3 &\geq 1 \\ y_1, y_2, y_3 &\geq 0 \end{aligned}$$

and that $w^* = 10$

$$\begin{array}{rrcrl} \frac{3}{5}\cdot & 2x_1 & + & x_2 & \leq 14 \\ \frac{1}{5}\cdot & -x_1 & + & 2x_2 & \leq 8 \\ \hline & x_1 & + & x_2 & \leq 10 \end{array}$$

the feasibility region of P is a subset of the half plane $x_1 + x_2 \leq 10$



$(2v - w)x_1 + (v + 2w)x_2 \leq 14v + 8w$ set of half planes that contain the feasibility region of P and pass through $[4, 6]$

$$2v - w \geq 1$$

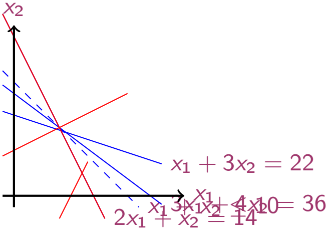
$$v + 2w \geq 1$$

Example of boundary lines among those allowed:

$$v = 1, w = 0 \implies 2x_1 + x_2 = 14$$

$$v = 1, w = 1 \implies x_1 + 3x_2 = 22$$

$$v = 2, w = 1 \implies 3x_1 + 4x_2 = 36$$



Outline

1. Geometric Interpretation

2. Farkas Lemma

3. Revised Simplex Method

Farkas Lemma

Theorem (Farkas Lemma)

Let $A \in \mathbb{R}^{m \times n}$ and $b \in \mathbb{R}^m$. Then,

either I. $\exists x \in \mathbb{R}^n : Ax = b$ and $x \geq 0$

or II. $\exists y \in \mathbb{R}^m : y^T A \geq 0^T$ and $y^T b < 0$

Easy to see that both I and II cannot occur together:

$$(0 \leq) \quad \underbrace{(y^T A)}_{\geq 0} \underbrace{x}_{\geq 0} = y^T b \quad (< 0)$$

In general:

	The system $A\mathbf{x} \leq \mathbf{b}$	The system $A\mathbf{x} = \mathbf{b}$
has a solution $\mathbf{x} \geq \mathbf{0}$ iff	$\mathbf{y} \geq \mathbf{0}, \mathbf{y}^T A \geq \mathbf{0} \Rightarrow \mathbf{y}^T\mathbf{b} \geq 0$	$\mathbf{y}^T A \geq \mathbf{0}^T \Rightarrow \mathbf{y}^T\mathbf{b} \geq 0$
has a solution $\mathbf{x} \in \mathbb{R}^n$ iff	$\mathbf{y} \geq \mathbf{0}, \mathbf{y}^T A = \mathbf{0} \Rightarrow \mathbf{y}^T\mathbf{b} \geq 0$	$\mathbf{y}^T A = \mathbf{0}^T \Rightarrow \mathbf{y}^T\mathbf{b} = 0$

Geometric interpretation of Farkas L.

Linear combination of a_i with nonnegative terms generates a convex cone:

$$\lambda_1 a_1 + \ldots + \lambda_n a_n, \lambda_1, \ldots, \lambda_n \geq 0$$

intersection of many $ax \leq 0$ polyhedral cone: $C = \{x \mid Ax \leq 0\}$
Convex hull of rays $p_i = \{\lambda_i a_i, \lambda_i \geq 0\}$

Either point b lies in convex cone C

or $\exists$ hyperplane h passing through point 0 $h = \{x \in \mathbb{R}^m : y^T x = 0\}$ for $y \in \mathbb{R}^m$ such that all vectors $a_1, \ldots, a_n$ (and thus C) lie on one side and b lies (strictly) on the other side (ie, $y^T a_i \geq 0, \forall i = 1 \ldots n$ and $y^T b < 0$).

Proof:
We prove: that the system $Ax \leq b$ has no solution iff there is a y such that:

$$\begin{aligned} A^T y &= 0^T \\ y &\geq 0 \\ b^T y &< 0 \end{aligned} \qquad (*)$$

$$\max 0 \qquad Ax \leq b$$

$$\min b^T y \qquad A^T y = 0 \qquad y \geq 0$$

Clearly dual is feasible (ie, $y = 0$).
Hence the primal is infeasible iff the dual is unbounded.
The dual is unbounded iff there exists a sol to (*).
Starting from $y = 0$ the simplex on the dual problem, would find an unbounded improvement Δy such that (*) is true.
Note that:

- There are other proofs for the Farkas Lemma that use analysis
- The Farkas Lemma can be used to prove the strong duality theorem

Certificate of Infeasibility

Farkas Lemma provides a way to certificate infeasibility.
Given a certificate y^* it is easy to check the conditions:

$$\begin{aligned} A^T y^* &\geq 0 \\ by^* &< 0 \end{aligned}$$

Proof: (by contradiction)
why y^* would be a certificate of infeasibility?
If $\exists$: $Ax^* = b$, $x^* \geq 0$, then:

$$A^T y^* \geq 0 \text{ and } x^* \geq 0 \qquad \Longrightarrow (y^*)^T A x^* \geq 0$$

$$(0 \leq) \quad (y^*)^T A x^* = (y^*)^T b \quad (< 0)$$

General form:

$$\begin{aligned} \max\ & c^T x \\ A_1 x &= b_1 \\ A_2 x &\leq b_2 \\ A_3 x &\geq b_3 \\ x &\geq 0 \end{aligned}$$

infeasible $\Leftrightarrow \exists y^*$

$$\begin{aligned} b_1^T y_1 + b_2^T y_2 + b_3^T y_3 &> 0 \\ A_1^T y_1 + A_2^T y_2 + A_3^T y_3 &\leq 0 \\ y_2 &\leq 0 \\ y_3 &\geq 0 \end{aligned}$$

Example:

$$\begin{aligned} \max\ & c^T x \\ x_1 &\leq 1 \\ x_1 &\geq 2 \end{aligned}$$

$$\begin{aligned} b_1^T y_1 + b_2^T y_2 &> 0 \\ A_1^T y_1 + A_2^T y_2 &\leq 0 \\ y_1 &\leq 0 \\ y_2 &\geq 0 \end{aligned}$$

$$\begin{aligned} y_1 + 2y_2 &> 0 \\ y_1 + y_2 &\leq 0 \\ y_1 &\leq 0 \\ y_2 &\geq 0 \end{aligned}$$

$y_1 = -1, y_2 = 1$ is a valid certificate.

- Observe that it is not unique!
- Note that it can always be reported in place of the dual solution.
- To repair infeasibility we should change the primal at least so much as that the certificate of infeasibility is no longer valid.
- Only constraints with $y_i \neq 0$ the infeasibility of the certificate

Outline

1. Geometric Interpretation

2. Farkas Lemma

3. Revised Simplex Method

Revised Simplex Method

Crucial: pivoting (ie, updating) the tableaux us the most costly part. Several ways to carry out this efficiently, requires matrix description of simplex.

- $\max\{c^T x \mid Ax \leq b, x \geq 0\}$
- $B = \{1 \ldots m\}$
- $N = \{n+1 \ldots n+m\}$
- $A_B = [A_1 \ldots A_m]$
- $A_N = [A_{n+1} \ldots A_{n+m}]$

Standard form

$$\left[\begin{array}{c:c:c:c} A_N & A_B & 0 & b \\ \hdashline c_N & c_B & 1 & 0 \end{array}\right]$$

$$Ax = A_N x_N + A_B x_B = b$$
$$A_B x_B = b - A_N x_N$$
$$x_B = A_B^{-1} b - A_B^{-1} A_N x_N$$

basic feasible solution:

- $X_N = 0$
- A_B lin. indep.
- $X_B \geq 0$

$$z = cx = c_B(A_B^{-1} b - A_B^{-1} A_N x_N) + c_N x_N =$$
$$= c_B A_B^{-1} b + (c_N - c_B \underbrace{A_B^{-1} A_N}_{\bar{A}}) x_N$$

Canonical form

$$\left[\begin{array}{c:c:c:c} A_B^{-1} A_N & I & 0 & A_B^{-1} b \\ \hdashline c_N^T - C_B^T A_B^{-1} A_N & 0 & 1 & -c_B^T A_B^{-1} b \end{array}\right]$$

We do not need to compute all elements of $\bar{A}$

$$\begin{aligned} \max \quad & x_1 + x_2 \\ & -x_1 + x_2 \leq 1 \\ & x_1 \leq 3 \\ & x_2 \leq 2 \\ & x_1, x_2 \geq 0 \end{aligned}$$

$$\begin{aligned} \max \quad & x_1 + x_2 \\ & -x_1 + x_2 + x_3 = 1 \\ & x_1 + x_4 = 3 \\ & x_2 + x_5 = 2 \\ & x_1, x_2, x_3, x_4, x_5 \geq 0 \end{aligned}$$

```
| x1 | x2 | x3 | x4 | x5 | -z | b |
|----+----+----+----+----+----+---|
| -1 |  1 |  1 |  0 |  0 |  0 | 1 |
|  1 |  0 |  0 |  1 |  0 |  0 | 3 |
|  0 |  1 |  0 |  0 |  1 |  0 | 2 |
|----+----+----+----+----+----+---|
|  1 |  1 |  0 |  0 |  0 |  1 | 0 |
```

After two iterations

```
| x1 | x2 | x3 | x4 | x5 | -z | b |
|----+----+----+----+----+----+---|
|  1 |  0 |  1 |  0 | -1 |  0 | 1 |
|  0 |  1 |  0 |  0 | -1 |  0 | 2 |
|  0 |  0 | -1 |  1 |  1 |  0 | 2 |
|----+----+----+----+----+----+---|
|  0 |  0 |  1 |  0 | -2 |  1 | 3 |
```

- Basic variables x_1, x_2, x_4. Non basic: x_3, x_5

$$A_B = \begin{bmatrix} -1 & 1 & 0 \\ 1 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix} \quad A_N = \begin{bmatrix} 1 & 0 \\ 0 & 0 \\ 0 & 1 \end{bmatrix} \quad x_B = \begin{bmatrix} x_1 \\ x_2 \\ x_4 \end{bmatrix} \quad x_N = \begin{bmatrix} x_3 \\ x_5 \end{bmatrix}$$

$$c_B = \begin{bmatrix} 1 & 1 & 0 \end{bmatrix} \quad c_N = \begin{bmatrix} 0 & 0 \end{bmatrix}$$

- **Entering variable**:
  in std. we look at tableau, in revised we need to compute: $c_N - c_B A_B^{-1} A_N$
  1. find $y = c_B A_B^{-1}$ by solving $yA_B = c_B$ (the latter can be done more efficiently)
  2. calculate $c_N - y^T A_N$

Step 1:

$$\begin{bmatrix} y_1 & y_2 & y_3 \end{bmatrix} \begin{bmatrix} -1 & 1 & 0 \\ 1 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix} = \begin{bmatrix} 1 & 1 & 0 \end{bmatrix}$$

$$\begin{bmatrix} -1 & 0 & 1 \\ 0 & 0 & 1 \\ 1 & 1 & -1 \end{bmatrix} \begin{bmatrix} 1 & 1 & 0 \end{bmatrix} = \begin{bmatrix} -1 \\ 0 \\ 2 \end{bmatrix}$$

Step 2:

$$\begin{bmatrix} 0 & 0 \end{bmatrix} - \begin{bmatrix} -1 & 0 & 2 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 0 & 0 \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} 1 & 2 \end{bmatrix}$$

(Note that they can be computed individually: $c_j - ya_{ij} > 0$)
Let's take the first we encounter x_3

- **Leaving variable**
  we increase variable by largest feasible amount θ

$$\begin{aligned} \text{I: } & x_1 + x_3 - x_5 = 1 \\ \text{II: } & -x_3 + x_4 + x_5 = 2 \end{aligned} \qquad \begin{aligned} x_1 &= 1 - x_3 \\ x_4 &= 2 + x_3 \end{aligned}$$

$$x_B = x_B^* - A_B^{-1} A_N x_N$$
$$x_B = x_B^* - d\theta$$

d is the column of $A_B^{-1} A_N$ that corresponds to the entering variable, ie, $d = A_B^{-1} a$ where a is the entering column

3. Find θ such that x_B stays positive:
   Find $d = A_B^{-1} a$ by solving $A_B d = a$

Step 3:

$$\begin{bmatrix} d_1 \\ d_2 \\ d_3 \end{bmatrix} = \begin{bmatrix} -1 & 0 & 1 \\ 0 & 0 & 1 \\ 1 & 1 & -1 \end{bmatrix} \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} \implies d = \begin{bmatrix} -1 \\ 0 \\ 1 \end{bmatrix} \implies x_B = \begin{bmatrix} 1 \\ 2 \\ 2 \end{bmatrix} - \begin{bmatrix} -1 \\ 0 \\ 1 \end{bmatrix} \theta \geq 0$$

$2 - \theta \geq 0 \implies \theta \leq 2 \rightsquigarrow x_4$ leaves

- So far we have done computations, but now we save the pivoting update. The update of A_B is done by replacing the leaving column by the entering column.

$$x_B^* = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} 3 \\ 2 \\ 2 \end{bmatrix} \begin{matrix} x_1 - d_1\theta \\ x_2 - d_2\theta \\ \theta \end{matrix} \qquad A_B = \begin{bmatrix} -1 & 1 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}$$

- Many implementations depending how $yA_B = c_B$ and $A_B d = a$ are solved. They are in fact solved from scratch.

- many operations saved especially if many variables!

- special ways to call the matrix A from memory

- better control over numerical issues since A_B^{-1} can be recomputed.

Solving system of equations
LU-factorization

- $Bx = b$ solved without computing B^{-1}
- it can be shown that $B = L'D^{-1}U$, where L' has elements $\neq 0$ only below the diagonal, D is a diagonal matrix and U has elements $\neq 0$ only above the diagonal matrix
- it can be rewritten as $B = LU$, L lower triangular matrix, U upper triangular matrix, ie, LU-factorization
- $LUx = b$: setting $y = Ux$ then
  1. $Ly = b$ can be solved easily by forward substitution
  2. $Ux = y$ can be solved easily by backward substitution

Eta Factorization of the Basis

Let $A_B = B$, kth iteration
B_k be the matrix with col p differing from B_{k-1}
Column p is the a column appearing in $B_{k-1}d = a$ solved at 3)
Hence:

$$B_k = B_{k-1}E_k$$

E_k is the eta matrix differing from id. matrix in only one column
(insert example)
No matter how we solve $yB_{k-1} = c_B$ and $B_{k-1}d = a$, their update always relays on $B_k = B_{k-1}E_k$ with E_k available.
Plus when initial basis by slack variable $B_0 = I$ and $B_1 = E_1, B_2 = E_1E_2 \cdots$:

$$B_k = E_1E_2 \ldots E_k \qquad \text{eta factorization}$$

$$((((yE_1)E_2)E_3) \cdots)E_k = c_b \quad uE_4 = c_B \; v = E_3 = u \; wE_2 = v \; yE_1 = w$$
$$(E_1(E_2 \cdots E_k d)) = a \quad E_1 u = a \; E_2 v = u \; E_3 w = v \; E_4 d = w$$

Resume

1. Geometric Interpretation
2. Farkas Lemma
3. Revised Simplex Method

Summary

Further topics in Linear Programming

- Ellipsoid method: cannot compete in practice but polynomial time (Khachyian, 1979)
- Interior point method(s) comptetitve with simplex and polynomial in some versions
- Lagrangian relaxation
- Decomposition methods:
  - Dantzig Wolfe decomposition
  - Benders decomposition