15.093J Optimization Methods

Lecture 4: The Simplex Method II

1 Outline

Slide 1

- Revised Simplex method
- The full tableau implementation
- Finding an initial BFS
- The complete algorithm
- The column geometry
- Computational efficiency

2 Revised Simplex

Slide 2

Initial data: $\boldsymbol{A}, \boldsymbol{b}, \boldsymbol{c}$

1. Start with basis $\boldsymbol{B} = [\boldsymbol{A}_{B(1)}, \ldots, \boldsymbol{A}_{B(m)}]$
   and $\boldsymbol{B}^{-1}$.
2. Compute $\boldsymbol{p}' = \boldsymbol{c}'_B \boldsymbol{B}^{-1}$
   $\overline{c}_j = c_j - \boldsymbol{p}' \boldsymbol{A}_j$
   - If $\overline{c}_j \geq 0$; x optimal; stop.
   - Else select $j : \overline{c}_j < 0$.

Slide 3

3. Compute $\boldsymbol{u} = \boldsymbol{B}^{-1} \boldsymbol{A}_j$.
   - If $\boldsymbol{u} \leq \boldsymbol{0} \Rightarrow$ cost unbounded; stop
   - Else
4. $\theta^* = \min\limits_{1 \leq i \leq m, u_i > 0} \dfrac{x_{B(i)}}{u_i} = \dfrac{u_{B(l)}}{u_l}$
5. Form a new basis $\overline{B}$ by replacing $\boldsymbol{A}_{B(l)}$ with $\boldsymbol{A}_j$.
6. $y_j = \theta^*$, $y_{B(i)} = x_{B(i)} - \theta^* u_i$

Slide 4

7. Form $[\boldsymbol{B}^{-1} | \boldsymbol{u}]$
8. Add to each one of its rows a multiple of the lth row in order to make the last column equal to the unit vector $\boldsymbol{e}_l$.
   The first m columns is $\overline{\boldsymbol{B}}^{-1}$.

2.1 Example

SLIDE 5

$$\begin{array}{lllll} \min & x_1+ & 5x_2 & -2x_3 & \\ \text{s.t.} & x_1+ & x_2+ & x_3 & \leq 4 \\ & x_1 & & & \leq 2 \\ & & & x_3 & \leq 3 \\ & & 3x_2+ & x_3 & \leq 6 \\ & x_1, & x_2, & x_3 & \geq 0 \end{array}$$

SLIDE 6

B $= \{\boldsymbol{A}_1, \boldsymbol{A}_3, \boldsymbol{A}_6, \boldsymbol{A}_7\}$, BFS: $\boldsymbol{x} = (2, 0, 2, 0, 0, 1, 4)'$
$\overline{\boldsymbol{c}}' = (0, 7, 0, 2, -3, 0, 0)$

$$\boldsymbol{B} = \begin{bmatrix} 1 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 \\ 0 & 1 & 0 & 1 \end{bmatrix}, \quad \boldsymbol{B}^{-1} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 1 & -1 & 0 & 0 \\ -1 & 1 & 1 & 0 \\ -1 & 1 & 0 & 1 \end{bmatrix}$$

$(u_1, u_3, u_6, u_7)' = \boldsymbol{B}^{-1}\boldsymbol{A}_5 = (1, -1, 1, 1)'$
$\theta^* = \min\left(\frac{2}{1}, \frac{1}{1}, \frac{4}{1}\right) = 1, \quad l = 6$

SLIDE 7

$l = 6$ ($\boldsymbol{A}_6$ exits the basis).

$$[\boldsymbol{B}^{-1}|\boldsymbol{u}] = \begin{bmatrix} 0 & 1 & 0 & 0 & 1 \\ 1 & -1 & 0 & 0 & -1 \\ -1 & 1 & 1 & 0 & 1 \\ -1 & 1 & 0 & 1 & 1 \end{bmatrix}$$

$$\Rightarrow \overline{\boldsymbol{B}}^{-1} = \begin{bmatrix} 1 & 0 & -1 & 0 \\ 0 & 0 & 1 & 0 \\ -1 & 1 & 1 & 0 \\ 0 & 0 & -1 & 1 \end{bmatrix}$$

2.2 Practical issues

SLIDE 8

- **Numerical Stability**
  $\boldsymbol{B}^{-1}$ needs to be computed from scratch once in a while, as errors accumulate
- **Sparsity**
  $\boldsymbol{B}^{-1}$ is represented in terms of sparse triangular matrices

3 Full tableau implementation

SLIDE 9

$-\boldsymbol{c}_B'\boldsymbol{B}^{-1}\boldsymbol{b}$	$\boldsymbol{c}' - \boldsymbol{c}_B'\boldsymbol{B}^{-1}\boldsymbol{A}$
$\boldsymbol{B}^{-1}\boldsymbol{b}$	$\boldsymbol{B}^{-1}\boldsymbol{A}$

or, in more detail,

$-\boldsymbol{c}'_B\boldsymbol{x}_B$	$\overline{c}_1$	$\ldots$	$\overline{c}_n$
$x_{B(1)}$	$\vert$		$\vert$
$\vdots$	$\boldsymbol{B}^{-1}\boldsymbol{A}_1$	$\ldots$	$\boldsymbol{B}^{-1}\boldsymbol{A}_n$
$x_{B(m)}$	$\vert$		$\vert$

3.1 Example

SLIDE 10

$$\begin{array}{llllllll} \min & -10x_1 & - & 12x_2 & - & 12x_3 & & \\ \text{s.t.} & x_1 & + & 2x_2 & + & 2x_3 & \leq & 20 \\ & 2x_1 & + & x_2 & + & 2x_3 & \leq & 20 \\ & 2x_1 & + & 2x_2 & + & x_3 & \leq & 20 \\ & x_1, x_2, x_3 \geq 0 & & & & & & \end{array}$$

$$\begin{array}{llllllllllll} \min & -10x_1 & - & 12x_2 & - & 12x_3 & & & & & & \\ \text{s.t.} & x_1 & + & 2x_2 & + & 2x_3 & + & x_4 & & & = & 20 \\ & 2x_1 & + & x_2 & + & 2x_3 & & & + x_5 & & = & 20 \\ & 2x_1 & + & 2x_2 & + & x_3 & & & & + x_6 & = & 20 \\ & x_1, \ldots, x_6 \geq 0 & & & & & & & & & & \end{array}$$

BFS: $\boldsymbol{x} = (0, 0, 0, 20, 20, 20)'$

B=$[\boldsymbol{A}_4, \boldsymbol{A}_5, \boldsymbol{A}_6]$

SLIDE 11

		x_1	x_2	x_3	x_4	x_5	x_6
	0	−10	−12	−12	0	0	0
$x_4 =$	20	1	2	2	1	0	0
$x_5 =$	20	2*	1	2	0	1	0
$x_6 =$	20	2	2	1	0	0	1

$\overline{\boldsymbol{c}}' = \boldsymbol{c}' - \boldsymbol{c}'_B\boldsymbol{B}^{-1}\boldsymbol{A} = \boldsymbol{c}' = (-10, -12, -12, 0, 0, 0)$

SLIDE 12

		x_1	x_2	x_3	x_4	x_5	x_6
	100	0	−7	−2	0	5	0
$x_4 =$	10	0	1.5	1*	1	−0.5	0
$x_1 =$	10	1	0.5	1	0	0.5	0
$x_6 =$	0	0	1	−1	0	−1	1

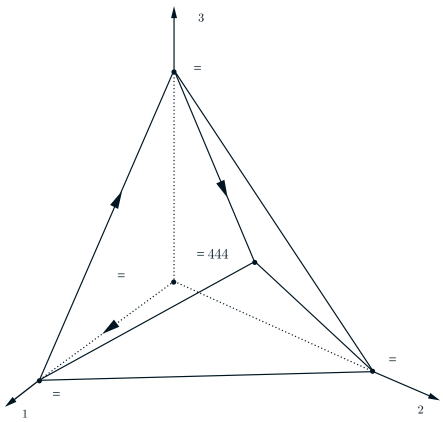
SLIDE 13

		x_1	x_2	x_3	x_4	x_5	x_6
	120	0	-4	0	2	4	0
$x_3 =$	10	0	1.5	1	1	-0.5	0
$x_1 =$	0	1	-1	0	-1	1	0
$x_6 =$	10	0	2.5*	0	1	-1.5	1

SLIDE 14

		x_1	x_2	x_3	x_4	x_5	x_6
	136	0	0	0	3.6	1.6	1.6
$x_3 =$	4	0	0	1	0.4	0.4	-0.6
$x_1 =$	4	1	0	0	-0.6	0.4	0.4
$x_2 =$	4	0	1	0	0.4	-0.6	0.4

SLIDE 15



4 Comparison of implementations

SLIDE 16

	Full tableau	**Revised simplex**
Memory	$O(mn)$	$O(m^2)$
Worst-case time	$O(mn)$	$O(mn)$
Best-case time	$O(mn)$	$O(m^2)$

5 Finding an initial BFS

SLIDE 17

- **Goal:** Obtain a BFS of $\boldsymbol{Ax} = \boldsymbol{b}, \quad \boldsymbol{x} \geq \boldsymbol{0}$
  or decide that LOP is infeasible.

- Special case: $\boldsymbol{b} \geq \boldsymbol{0}$

$$\boldsymbol{Ax} \leq \boldsymbol{b}, \quad \boldsymbol{x} \geq \boldsymbol{0}$$

$$\Rightarrow \boldsymbol{Ax} + \boldsymbol{s} = \boldsymbol{b}, \quad \boldsymbol{x}, \boldsymbol{s} \geq \boldsymbol{0}$$

$$\boldsymbol{s} = \boldsymbol{b}, \quad \boldsymbol{x} = \boldsymbol{0}$$

5.1 Artificial variables

SLIDE 18

$$\boldsymbol{Ax} = \boldsymbol{b}, \quad \boldsymbol{x} \geq \boldsymbol{0}$$

1. Multiply rows with -1 to get $\boldsymbol{b} \geq \boldsymbol{0}$.

2. Introduce artificial variables $\boldsymbol{y}$, start with initial BFS $\boldsymbol{y} = \boldsymbol{b}, \ \ \boldsymbol{x} = \boldsymbol{0}$, and apply simplex to auxiliary problem

$$\begin{array}{rl} \min & y_1 + y_2 + \ldots + y_m \\ \text{s.t.} & \boldsymbol{Ax} + \boldsymbol{y} = \boldsymbol{b} \\ & \boldsymbol{x}, \boldsymbol{y} \geq \boldsymbol{0} \end{array}$$

SLIDE 19

3. If cost $> 0 \ \Rightarrow$ **LOP infeasible**; stop.

4. If cost $= 0$ and no artificial variable is in the basis, then a BFS was found.

5. Else, all $y_i^* = 0$, but some are still in the basis. Say we have $\boldsymbol{A}_{B(1)}, \ldots, \boldsymbol{A}_{B(k)}$ in basis $k < m$. There are $m - k$ additional columns of $\boldsymbol{A}$ to form a basis.

SLIDE 20

6. Drive artificial variables out of the basis: If lth basic variable is artificial examine lth row of $\boldsymbol{B}^{-1}\boldsymbol{A}$. If all elements $= 0 \Rightarrow$ row redundant. Otherwise pivot with $\neq 0$ element.

6 A complete Algorithm for LO

SLIDE 21

Phase I:

1. By multiplying some of the constraints by -1, change the problem so that $\boldsymbol{b} \geq \boldsymbol{0}$.

2. Introduce $y_1, \ldots, y_m$, if necessary, and apply the simplex method to min $\sum_{i=1}^{m} y_i$.

3. If cost> 0, original problem is infeasible; STOP.

4. If cost$= 0$, a feasible solution to the original problem has been found.
5. Drive artificial variables out of the basis, potentially eliminating redundant rows.

SLIDE 22

Phase II:

1. Let the final basis and tableau obtained from Phase I be the initial basis and tableau for Phase II.
2. Compute the reduced costs of all variables for this initial basis, using the cost coefficients of the original problem.
3. Apply the simplex method to the original problem.

6.1 Possible outcomes

SLIDE 23

1. Infeasible: Detected at Phase I.
2. $\boldsymbol{A}$ has linearly dependent rows: Detected at Phase I, eliminate redundant rows.
3. Unbounded (cost$= -\infty$): detected at Phase II.
4. Optimal solution: Terminate at Phase II in optimality check.

7 The big-M method

SLIDE 24

$$\begin{array}{ll} \min & \sum_{j=1}^{n} c_j x_j + M \sum_{i=1}^{m} y_i \\ \text{s.t.} & \boldsymbol{Ax} + \boldsymbol{y} = \boldsymbol{b} \\ & \boldsymbol{x}, \boldsymbol{y} \geq \boldsymbol{0} \end{array}$$

8 The Column Geometry

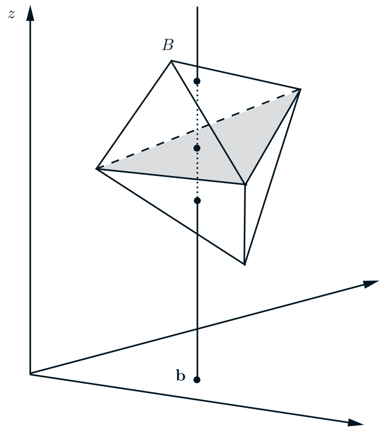
SLIDE 25

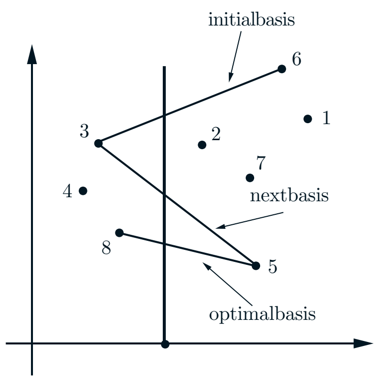
$$\begin{array}{llcl} \min & \boldsymbol{c}'\boldsymbol{x} & & \\ \text{s.t.} & \boldsymbol{Ax} & = & \boldsymbol{b} \\ & \boldsymbol{e}'\boldsymbol{x} & = & 1 \\ & \boldsymbol{x} & \geq & \boldsymbol{0} \end{array}$$

$$x_1 \begin{bmatrix} \boldsymbol{A}_1 \\ c_1 \end{bmatrix} + x_2 \begin{bmatrix} \boldsymbol{A}_2 \\ c_2 \end{bmatrix} + \cdots + x_n \begin{bmatrix} \boldsymbol{A}_n \\ c_n \end{bmatrix} = \begin{bmatrix} \boldsymbol{b} \\ z \end{bmatrix}$$

SLIDE 26

SLIDE 27


z
B
b


initialbasis
6
1
3
2
7
4
nextbasis
8
5
optimalbasis

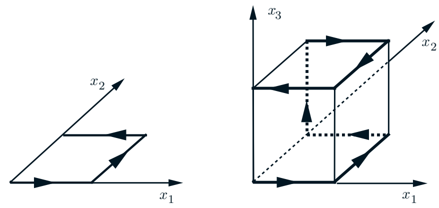
9 Computational efficiency

Slide 28

Exceptional practical behavior: linear in n
Worst case

$$\begin{array}{rl} \max & x_n \\ \text{s.t.} & \epsilon \leq x_1 \leq 1 \\ & \epsilon x_{i-1} \leq x_i \leq 1 - \epsilon x_{i-1}, \qquad i = 2, \ldots, n \end{array}$$

Slide 29



(a) (b)

Slide 30

Theorem

- The feasible set has 2^n vertices
- The vertices can be ordered so that each one is adjacent to and has lower cost than the previous one.
- There exists a pivoting rule under which the simplex method requires $2^n - 1$ changes of basis before it terminates.

15.093J / 6.255J Optimization Methods
Fall 2009