60016 OPERATIONS RESEARCH

Finite Termination and Degeneracies

Last Lecture

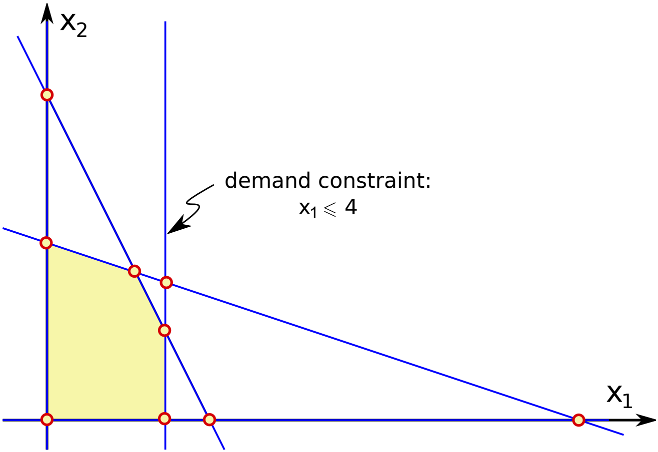
- Simplex tableau
- Pivoting
  - Pivoting equations
  - Pivot selection rules ensuring:
    - Non-inferiority
    - Feasibility

This Lecture

- Finite termination in the Simplex method
- Degenerate BS's
- Finite termination theorem
- Cycling

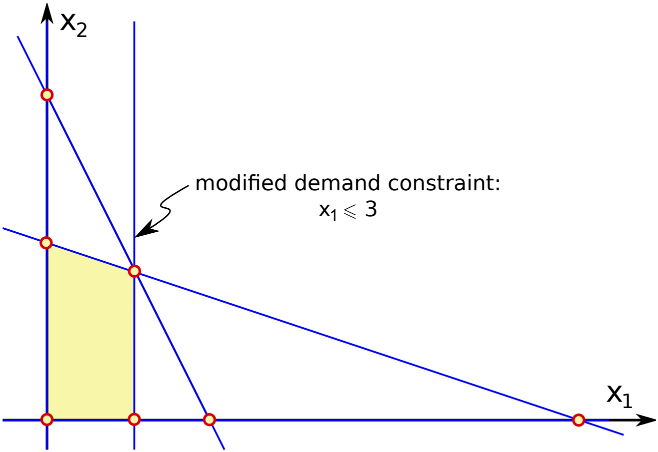
Degenerate BS's

Feasible set of Example 1:



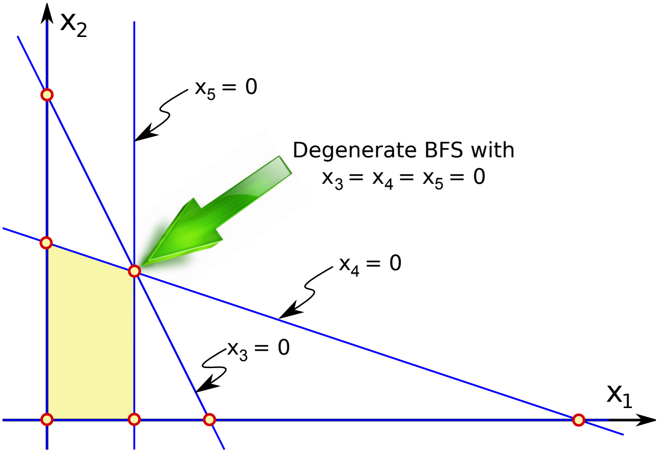
Degenerate BS's (cont)

Consider now a variant of Example 1:



Degenerate BS's (cont)

The highlighted BS corresponds to the index sets
$I = \{1, 2, 3\}$, $I = \{1, 2, 4\}$ and $I = \{1, 2, 5\}$.



Degenerate BFS's: Definition

Definition: A BS is called degenerate if one or more basic variables (BVs) are zero.

$\Rightarrow$ A degenerate BS has more than $n - m$ zero-valued variables.

$\Rightarrow$ If we look at the tableau, there exists at least a BV such that $i \in I$ and $y_{i0} = 0$.

Definition: A BS is called non-degenerate if all of its basic variables are different from zero.

Finite Termination

Theorem: If all BFS's are non-degenerate, then the simplex algorithm must terminate after a finite number of steps with

- either an optimal solution
- or a proof that the problem is unbounded.

Finite Termination (Proof)

- At each step we have $y_{i0} > 0 \; \forall i \in I$ (non-degeneracy).
- Unless optimality or unboundedness is detected in STEP 1 or 2, we find $\beta_0' = \beta_0 - \frac{\beta_q}{y_{pq}} y_{p0} < \beta_0$.
- Thus, the sequence of objective values obtained by the algorithm is strictly decreasing.

$$\beta_0 > \beta_0' > \beta_0'' > \cdots$$

No basic solution will ever be repeated!

- There are $\leq \binom{n}{m}$ basic solutions, since $\binom{n}{m}$ is the number of ways of picking m columns out of n to form an index set I.
- Thus, the process cannot continue indefinitely and must terminate at STEP 1 or 2 after a finite number of iterations (even though possibly a very large one!).

□

Degeneracy

Lemma: Assume that, $\forall i = 1, \ldots, n$, $\exists$ BS $\hat{x}$ with $\hat{x}_i \neq 0$.
Then, a BS x is degenerate if and only if it is associated with more than one index set.

Proof: BS x degenerate $\Leftarrow$ BS x has more than one index set

- Suppose a BS x corresponds to index sets I_1 and I_2, $I_1 \neq I_2$.
- Then $x_i = 0$ for all NBVs x_i with either $i \notin I_1$ or $i \notin I_2$ or both.
- In particular, since $I_1 \neq I_2$, there will be a NBV x_i in I_1, that is a BV in I_2. Since the two index sets describe the same BS x, x_i must be zero also in I_2 where it is basic.

$\Rightarrow$ x is a degenerate BS.

- (The same holds for any x_i that is NBV in I_2 and BV in I_1.)

Degeneracy (cont)

Proof: BS x degenerate $\Rightarrow$ BS x has more than one index set

- Suppose x is a degenerate BS associated with some index set I; consider the corresponding simplex tableau.
- Due to degeneracy, $\exists p \in I$ with $y_{p0} = 0$.
- $\exists q \notin I$ such that $y_{pq} \neq 0$. Otherwise, it would be always $x_p = 0$ in all the feasible set which we assume impossible in the theorem statement.
- Pivoting on (p, q) gives a new basic solution which is identical to the current one since

$$y'_{q0} = \frac{y_{p0}}{y_{pq}} = 0 = y_{p0} \text{ and } y'_{i0} = y_{i0} - \frac{y_{iq}}{y_{pq}} y_{p0} = y_{i0} \ \forall i \in I \backslash \{p\}.$$

$\Rightarrow$ x corresponds to the index sets I and $(I \backslash \{p\}) \cup \{q\}$.

$\square$

Degeneracy and Simplex Algorithm

How does degeneracy affect the simplex algorithm?

- The index sets I and $(I \backslash \{p\}) \cup \{q\}$ produce the same BFS but different basic representations.
- If we pivot on (p, q) when $y_{p0} = 0$, then the new BFS is identical to the old one.
- In particular, we find

$$\beta_0' = \beta_0 - \frac{\beta_q}{y_{pq}} y_{p0} = \beta_0,$$

  and the finite termination theorem breaks down (no strict improvement of objective value).
- A pivot step (p, q) is called degenerate if $y_{p0} = 0$ and non-degenerate otherwise.

Degeneracy and Simplex Algorithm

The simplex algorithm can now be decomposed into:

$$\begin{bmatrix} \text{sequence of} \\ \text{degenerate} \\ \text{pivots} \end{bmatrix} \quad \begin{matrix} \text{non-} \\ \text{degenerate} \\ \text{pivot} \end{matrix} \quad \begin{bmatrix} \text{sequence of} \\ \text{degenerate} \\ \text{pivots} \end{bmatrix} \ldots$$

Note: Some or all of these sequences of degenerate pivots may be empty.

Geometrically, the current BFS remains unchanged throughout a sequence of degenerate pivots, and a non-degenerate pivot moves it to a different BFS.

Degeneracy and Simplex Algorithm

- We know that the number of index sets is $\leq \binom{n}{m}$.

$\Rightarrow$ Sequences of degenerate pivots are finite if no index set is repeated.

- However, on same rare instances pivoting can result in a cycling behaviour.
- In general, choosing degenerate pivots is a necessary condition, but not a sufficient one, for cycling.
- After a sequence of pivots, we return to the same index set and so the algorithm will cycle and never terminate!

Cycling Example

$$\min z = -\frac{3}{4}x_4 + 20x_5 - \frac{1}{2}x_6 + 6x_7$$

subject to:

$$\begin{array}{llllllll}
x_1 & & & +\frac{1}{4}x_4 & -8x_5 & -x_6 & +9x_7 & = 0 \\
& x_2 & & +\frac{1}{2}x_4 & -12x_5 & -\frac{1}{2}x_6 & +3x_7 & = 0 \\
& & x_3 & & & +x_6 & & = 1
\end{array}$$

Use standard pivoting conventions!

Cycling Example (cont)

BV	x_1	x_2	x_3	x_4	x_5	x_6	x_7	RHS
z	0	0	0	$\frac{3}{4}$	-20	$\frac{1}{2}$	-6	0
x_1	1	0	0	$\frac{1}{4}$	-8	-1	9	0
x_2	0	1	0	$\frac{1}{2}$	-12	$-\frac{1}{2}$	3	0
x_3	0	0	1	0	0	1	0	1

Cycling Example (cont)

BV	x_1	x_2	x_3	x_4	x_5	x_6	x_7	RHS
z	0	0	0	$\frac{3}{4}$	-20	$\frac{1}{2}$	-6	0
x_1	1	0	0	$\frac{1}{4}$	-8	-1	9	0
x_2	0	1	0	$\frac{1}{2}$	-12	$-\frac{1}{2}$	3	0
x_3	0	0	1	0	0	1	0	1
z	-3	0	0	0	4	$\frac{7}{2}$	-33	0
x_4	4	0	0	1	-32	-4	36	0
x_2	-2	1	0	0	4	$\frac{3}{2}$	-15	0
x_3	0	0	1	0	0	1	0	1

Cycling Example (cont)

BV	x_1	x_2	x_3	x_4	x_5	x_6	x_7	RHS
z	-3	0	0	0	4	$\frac{7}{2}$	-33	0
x_4	4	0	0	1	-32	-4	36	0
x_2	-2	1	0	0	4	$\frac{3}{2}$	-15	0
x_3	0	0	1	0	0	1	0	1

Cycling Example (cont)

BV	x_1	x_2	x_3	x_4	x_5	x_6	x_7	RHS
z	-3	0	0	0	4	$\frac{7}{2}$	-33	0
x_4	4	0	0	1	-32	-4	36	0
x_2	-2	1	0	0	4	$\frac{3}{2}$	-15	0
x_3	0	0	1	0	0	1	0	1
z	-1	-1	0	0	0	2	-18	0
x_4	-12	8	0	1	0	8	-84	0
x_5	$-\frac{1}{2}$	$\frac{1}{4}$	0	0	1	$\frac{3}{8}$	$-\frac{15}{4}$	0
x_3	0	0	1	0	0	1	0	1

Cycling Example (cont)

BV	x_1	x_2	x_3	x_4	x_5	x_6	x_7	RHS
z	-1	-1	0	0	0	2	-18	0
x_4	-12	8	0	1	0	8	-84	0
x_5	$-\frac{1}{2}$	$\frac{1}{4}$	0	0	1	$\frac{3}{8}$	$-\frac{15}{4}$	0
x_3	0	0	1	0	0	1	0	1

Cycling Example (cont)

BV	x_1	x_2	x_3	x_4	x_5	x_6	x_7	RHS
z	-1	-1	0	0	0	2	-18	0
x_4	-12	8	0	1	0	8	-84	0
x_5	$-\frac{1}{2}$	$\frac{1}{4}$	0	0	1	$\frac{3}{8}$	$-\frac{15}{4}$	0
x_3	0	0	1	0	0	1	0	1
z	2	-3	0	$-\frac{1}{4}$	0	0	3	0
x_6	$-\frac{3}{2}$	1	0	$\frac{1}{8}$	0	1	$-\frac{21}{2}$	0
x_5	$\frac{1}{16}$	$-\frac{1}{8}$	0	$\frac{3}{64}$	1	0	$\frac{3}{16}$	0
x_3	$\frac{3}{2}$	-1	1	$-\frac{1}{8}$	0	0	$\frac{21}{2}$	1

Cycling Example (cont)

BV	x_1	x_2	x_3	x_4	x_5	x_6	x_7	RHS
z	2	-3	0	$-\frac{1}{4}$	0	0	3	0
x_6	$-\frac{3}{2}$	1	0	$\frac{1}{8}$	0	1	$-\frac{21}{2}$	0
x_5	$\frac{1}{16}$	$-\frac{1}{8}$	0	$\frac{3}{64}$	1	0	$\frac{3}{16}$	0
x_3	$\frac{3}{2}$	-1	1	$-\frac{1}{8}$	0	0	$\frac{21}{2}$	1

Cycling Example (cont)

BV	x_1	x_2	x_3	x_4	x_5	x_6	x_7	RHS
z	2	-3	0	$-\frac{1}{4}$	0	0	3	0
x_6	$-\frac{3}{2}$	1	0	$\frac{1}{8}$	0	1	$-\frac{21}{2}$	0
x_5	$\frac{1}{16}$	$-\frac{1}{8}$	0	$\frac{3}{64}$	1	0	$\frac{3}{16}$	0
x_3	$\frac{3}{2}$	-1	1	$-\frac{1}{8}$	0	0	$\frac{21}{2}$	1
z	1	-1	0	$\frac{1}{2}$	-16	0	0	0
x_6	2	-6	0	$-\frac{5}{2}$	56	1	0	0
x_7	$\frac{1}{3}$	$-\frac{2}{3}$	0	$-\frac{1}{4}$	$\frac{16}{3}$	0	1	0
x_3	-2	6	1	$\frac{5}{2}$	-56	0	0	1

Cycling Example (cont)

BV	x_1	x_2	x_3	x_4	x_5	x_6	x_7	RHS
z	1	-1	0	$\frac{1}{2}$	-16	0	0	0
x_6	2	-6	0	$-\frac{5}{2}$	56	1	0	0
x_7	$\frac{1}{3}$	$-\frac{2}{3}$	0	$-\frac{1}{4}$	$\frac{16}{3}$	0	1	0
x_3	-2	6	1	$\frac{5}{2}$	-56	0	0	1

Cycling Example (cont)

BV	x_1	x_2	x_3	x_4	x_5	x_6	x_7	RHS
z	1	-1	0	$\frac{1}{2}$	-16	0	0	0
x_6	2	-6	0	$-\frac{5}{2}$	56	1	0	0
x_7	$\frac{1}{3}$	$-\frac{2}{3}$	0	$-\frac{1}{4}$	$\frac{16}{3}$	0	1	0
x_3	-2	6	1	$\frac{5}{2}$	-56	0	0	1
z	0	2	0	$\frac{7}{4}$	-44	$-\frac{1}{2}$	0	0
x_1	1	-3	0	$-\frac{5}{4}$	28	$\frac{1}{2}$	0	0
x_7	0	$\frac{1}{3}$	0	$\frac{1}{6}$	-4	$-\frac{1}{6}$	1	0
x_3	0	0	1	0	0	1	0	1

Cycling Example (cont)

BV	x_1	x_2	x_3	x_4	x_5	x_6	x_7	RHS
z	0	2	0	$\frac{7}{4}$	-44	$-\frac{1}{2}$	0	0
x_1	1	-3	0	$-\frac{5}{4}$	28	$\frac{1}{2}$	0	0
x_7	0	$\frac{1}{3}$	0	$\frac{1}{6}$	-4	$-\frac{1}{6}$	1	0
x_3	0	0	1	0	0	1	0	1

Cycling Example (cont)

BV	x_1	x_2	x_3	x_4	x_5	x_6	x_7	RHS
z	0	2	0	$\frac{7}{4}$	-44	$-\frac{1}{2}$	0	0
x_1	1	-3	0	$-\frac{5}{4}$	28	$\frac{1}{2}$	0	0
x_7	0	$\frac{1}{3}$	0	$\frac{1}{6}$	-4	$-\frac{1}{6}$	1	0
x_3	0	0	1	0	0	1	0	1
z	0	0	0	$\frac{3}{4}$	-20	$\frac{1}{2}$	-6	0
x_1	1	0	0	$\frac{1}{4}$	-8	-1	9	0
x_2	0	1	0	$\frac{1}{2}$	-12	$-\frac{1}{2}$	3	0
x_3	0	0	1	0	0	1	0	1

This is the initial basic representation for $I = \{1, 2, 3\}$!

Bland's Rule

We can avoid cycling by amending the pivoting conventions.

Bland's Rule:

(i) Choose the lowest-numbered (leftmost) nonbasic column q with a positive cost.

$$q = \min\{j \neq 0 \mid \beta_j > 0\}$$

(ii) Denote as p the row with minimal $\overline{x}_{iq}$, in case of ties pick the row with the smallest index (same as standard conventions).

Theorem: With Bland's rule the simplex algorithm cannot cycle and hence is finite.

Degeneracy in Practice

- Cycling was thought to occur in contrived examples. For a long time it has therefore been ignored in commercial solvers.
- More recent experience with larger and larger problems indicates that cycling occurs, but it is still a rare event.
- Rigorous remedies such as Bland's rule are not satisfactory as they increase the number of iterations also in problems where cycling does not occur.
- In practice it is acceptable to replace a $y_{i0} = 0$ by $y_{i0} = \epsilon > 0$ (e.g., $\epsilon < 10^{-3}$) and then continue.