ECEN 615
Methods of Electric Power Systems Analysis

Lecture 22: Linear Programming and Optimal Power Flow

Prof. Tom Overbye
Dept. of Electrical and Computer Engineering
Texas A&M University
overbye@tamu.edu

TEXAS A&M UNIVERSITY

Announcements

- Read Chapters 3 and 8 from the book
- Homework 5 is due on Thursday November 14
- Second exam is in class on Nov 21
  - Same format as with the first exam

Aside: Matrix Singular Value Decomposition (SVD)

- The SVD is a factorization of a matrix that generalizes the eigendecomposition to any m by n matrix to produce

$$\mathbf{Y} = \mathbf{U}\boldsymbol{\Sigma}\mathbf{V}^{\mathbf{T}}$$

The original concept is more than 100 years old, but has founds lots of recent applications

  where $\boldsymbol{\Sigma}$ is a diagonal matrix of the singular values
- The singular values are non-negative real numbers that can be used to indicate the major components of a matrix (the gist is they provide a way to decrease the rank of a matrix
- A key application is image compression

Aside: SVD Image Compression Example

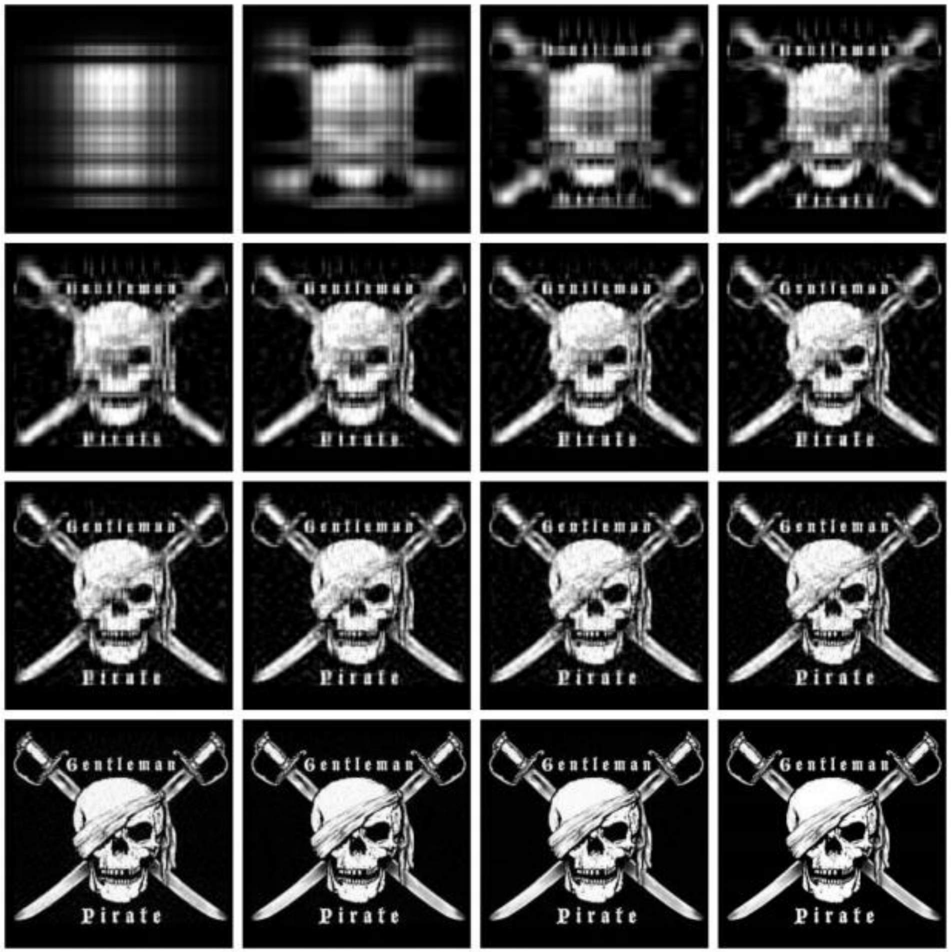


Figure 3.1: Image size 250x236 – modes used {{1,2,4,6},{8,10,12,14},{16,18,20,25},{50,75,100,original image}}

Images can be represented with matrices. When an SVD is applied and only the largest singular values are retained the image is compressed.

Computationally the SVD is order m^2n (with $n \leq m$)

Image Source: www.math.utah.edu/~goller/F15_M2270/BradyMathews_SVDImage.pdf

Aside: Pseudoinverse of a Matrix

- The pseudoinverse of a matrix generalizes concept of a matrix inverse to an m by n matrix, in which m >= n
  - Specifically this is a Moore-Penrose Matrix Inverse
- Notation for the pseudoinverse of **A** is $\mathbf{A}^{+}$
- Satisfies $\mathbf{A}\mathbf{A}^{+}\mathbf{A} = \mathbf{A}$
- If **A** is a square matrix, then $\mathbf{A}^{+} = \mathbf{A}^{-1}$
- Quite useful for solving the least squares problem since the least squares solution of $\mathbf{A}\mathbf{x} = \mathbf{b}$ is $\mathbf{x} = \mathbf{A}^{+}\mathbf{b}$
- Can be calculated using an SVD

$$\mathbf{A} = \mathbf{U}\boldsymbol{\Sigma}\mathbf{V}^{T}$$

$$\mathbf{A}^{+} = \mathbf{V}\boldsymbol{\Sigma}^{+}\mathbf{U}^{T}$$

Least Squares Matrix Pseudoinverse Example

- Assume we wish to fix a line ($mx + b = y$) to three data points: (1,1), (2,4), (6,4)
- Two unknowns, m and b; hence $\mathbf{x} = [m \;\; b]^T$
- Setup in form of $\mathbf{Ax} = \mathbf{b}$

$$\begin{bmatrix} 1 & 1 \\ 2 & 1 \\ 6 & 1 \end{bmatrix} \begin{bmatrix} m \\ b \end{bmatrix} = \begin{bmatrix} 1 \\ 4 \\ 4 \end{bmatrix} \quad \text{so} \;\; \mathbf{A} = \begin{bmatrix} 1 & 1 \\ 2 & 1 \\ 6 & 1 \end{bmatrix}$$

Least Squares Matrix Pseudoinverse Example, cont.

- Doing an economy SVD

$$\mathbf{A} = \mathbf{U}\boldsymbol{\Sigma}\mathbf{V}^T = \begin{bmatrix} -0.182 & -0.765 \\ -0.331 & -0.543 \\ -0.926 & 0.345 \end{bmatrix} \begin{bmatrix} 6.559 & 0 \\ 0 & 0.988 \end{bmatrix} \begin{bmatrix} -0.976 & -0.219 \\ 0.219 & -0.976 \end{bmatrix}$$

- Computing the pseudoinverse

$$\mathbf{A}^+ = \mathbf{V}\,\boldsymbol{\Sigma}^+\mathbf{U}^T = \begin{bmatrix} -0.976 & 0.219 \\ -0.219 & -0.976 \end{bmatrix} \begin{bmatrix} 0.152 & 0 \\ 0 & 1.012 \end{bmatrix} \begin{bmatrix} -0.182 & -0.331 & -0.926 \\ -0.765 & -0.543 & 0.345 \end{bmatrix}$$

$$\mathbf{A}^+ = \mathbf{V}\,\boldsymbol{\Sigma}^+\mathbf{U}^T = \begin{bmatrix} -0.143 & -0.071 & 0.214 \\ 0.762 & 0.548 & -0.310 \end{bmatrix}$$

In an economy SVD the $\boldsymbol{\Sigma}$ matrix has dimensions of m by m if m < n or n by n if n < m

Least Squares Matrix Pseudoinverse Example, cont.

- Computing $\mathbf{x} = [\text{m b}]^T$ gives

$$\mathbf{A}^{+}\mathbf{b} = \begin{bmatrix} -0.143 & -0.071 & 0.214 \\ 0.762 & 0.548 & -0.310 \end{bmatrix} \begin{bmatrix} 1 \\ 4 \\ 4 \end{bmatrix} = \begin{bmatrix} 0.429 \\ 1.71 \end{bmatrix}$$

- With the pseudoinverse approach we immediately see the sensitivity of the elements of **x** to the elements of **b**
  - New values of m and b can be readily calculated if **y** changes
- Computationally the SVD is order m^2n (with $n \leq m$)

Quick Coverage of Linear Programming

- LP is probably the most widely used mathematical programming technique
- It is used to solve linear, constrained minimization (or maximization) problems in which the objective function and the constraints can be written as linear functions

Example Problem 1

- Assume that you operate a lumber mill which makes both construction-grade and finish-grade boards from the logs it receives. Suppose it takes 2 hours to rough-saw and 3 hours to plane each 1000 board feet of construction-grade boards. Finish-grade boards take 2 hours to rough-saw and 5 hours to plane for each 1000 board feet. Assume that the saw is available 8 hours per day, while the plane is available 15 hours per day. If the profit per 1000 board feet is $100 for construction-grade and $120 for finish-grade, how many board feet of each should you make per day to maximize your profit?

Problem 1 Setup

Let x_1=amount of cg, x_2= amount of fg

$$
\begin{aligned}
&\text{Maximize} && 100x_1 + 120x_2 \\
&\text{s.t.} && 2x_1 + 2x_2 \le 8 \\
& && 3x_1 + 5x_2 \le 15 \\
& && x_1, x_2 \ge 0
\end{aligned}
$$

Notice that all of the equations are linear, but they are inequality, as opposed to equality, constraints; we are seeking to determine the values of x_1 and x_2

Example Problem 2

- A nutritionist is planning a meal with 2 foods: A and B. Each ounce of A costs $ 0.20, and has 2 units of fat, 1 of carbohydrate, and 4 of protein. Each ounce of B costs $0.25, and has 3 units of fat, 3 of carbohydrate, and 3 of protein. Provide the least cost meal which has no more than 20 units of fat, but with at least 12 units of carbohydrates and 24 units of protein.

Problem 2 Setup

Let x_1=ounces of A, x_2= ounces of B

Minimize $\quad 0.20x_1 + 0.25x_2$

s.t. $\quad 2x_1 + 3x_2 \le 20$

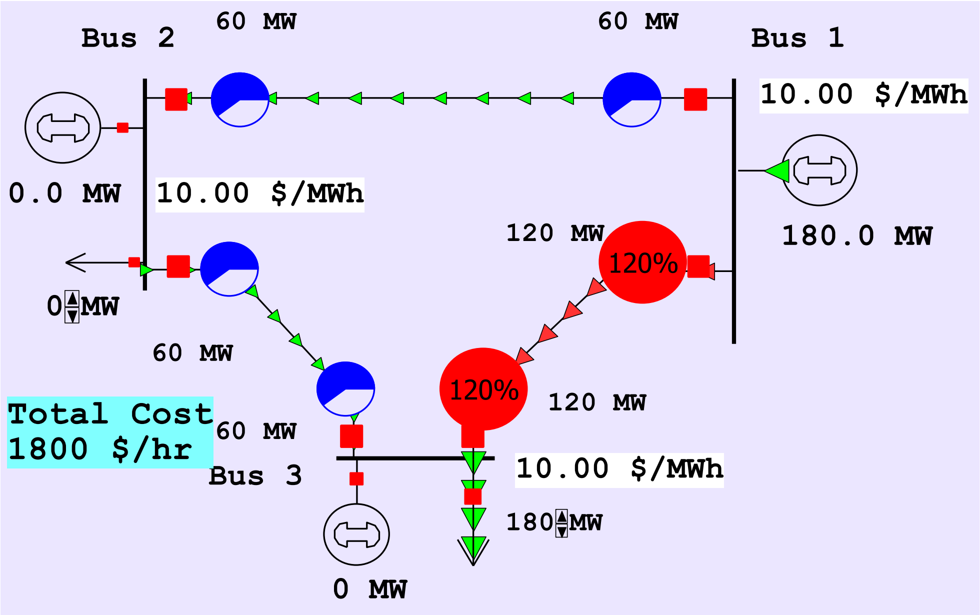
$$x_1 + 3x_2 \ge 12$$

$$4x_1 + 3x_2 \ge 24$$

Again all of the equations are linear, but
they are inequality, as opposed to equality, constraints;
we are again seeking to determine the values of x_1 and x_2;
notice there are also more constraints then solution variables

Three Bus Case Formulation

- For the earlier three bus system given the initial condition of an overloaded transmission line, minimize the cost of generation such that the change in generation is zero, and the flow on the line between buses 1 and 3 is not violating its limit



- Can be setup considering the change in generation, $(\Delta P_{G1}, \Delta P_{G2}, \Delta P_{G3})$

Three Bus Case Problem Setup

Let $x_1 = \Delta P_{G1}$, $x_2 = \Delta P_{G2}$, $x_3 = \Delta P_{G3}$

Minimize $\quad 10x_1 + 12x_2 + 20x_3$

s.t. $\quad \frac{2}{3}x_1 + \frac{1}{3}x_2 \le -20$ Line flow constraint

$x_1 + x_2 + x_3 = 0$ Power balance constraint

enforcing limits on $x_1,\ x_2,\ x_3$

LP Standard Form

The standard form of the LP problem is

Minimize $\quad \mathbf{c}\,\mathbf{x}$

s.t. $\quad \mathbf{A}\mathbf{x} = \mathbf{b}$

$\quad \mathbf{x} \geq \mathbf{0}$

Maximum problems can be treated as minimizing the negative

where $\quad \mathbf{x} =$ n-dimensional column vector

$\mathbf{c} =$ n-dimensional row vector

$\mathbf{b} =$ m-dimensional column vector

$\mathbf{A} =$ m×n matrix

For the LP problem usually n>> m

The previous examples were not in this form!

Replacing Inequality Constraints with Equality Constraints

- The LP standard form does not allow inequality constraints
- Inequality constraints can be replaced with equality constraints through the introduction of slack variables, each of which must be greater than or equal to zero

$$\ldots \le b_i \rightarrow \ldots + y_i = b_i \quad \text{with } y_i \ge 0$$

$$\ldots \ge b_i \rightarrow \ldots - y_i = b_i \quad \text{with } y_i \ge 0$$

- Slack variables have no cost associated with them; they merely tell how far a constraint is from being binding, which will occur when its slack variable is zero

Lumber Mill Example with Slack Variables

- Let the slack variables be x_3 and x_4, so

Minimize the negative

$$\text{Minimize} \quad -(100x_1 + 120x_2)$$

$$\text{s.t.} \quad 2x_1 + 2x_2 + x_3 = 8$$

$$3x_1 + 5x_2 + x_4 = 15$$

$$x_1, x_2, x_3, x_4 \geq 0$$

LP Definitions

A vector $\mathbf{x}$ is said to be basic if

1. $\mathbf{Ax} = \mathbf{b}$
2. At most m components of $\mathbf{x}$ are non-zero; these are called the basic variables; the rest are non basic variables; if there as less than m non-zeros then $\mathbf{x}$ is called degenerate

$\mathbf{A}_\mathrm{B}$ is called the basis matrix

Define $\mathbf{x} = \begin{bmatrix} \mathbf{x}_\mathrm{B} \\ \mathbf{x}_\mathrm{N} \end{bmatrix}$ (with $\mathbf{x}_\mathrm{B}$ basic) and $\mathbf{A} = \begin{bmatrix} \mathbf{A}_B & \mathbf{A}_N \end{bmatrix}$

With $\begin{bmatrix} \mathbf{A}_B & \mathbf{A}_N \end{bmatrix} \begin{bmatrix} \mathbf{x}_\mathrm{B} \\ \mathbf{x}_\mathrm{N} \end{bmatrix} = \mathbf{b}$ so $\mathbf{x}_\mathrm{B} = \mathbf{A}_B^{-1}(\mathbf{b} - \mathbf{A}_N \mathbf{x}_\mathrm{N})$

Fundamental LP Theorem

- Given an LP in standard form with **A** of rank m then
  - If there is a feasible solution, there is a basic feasible solution
  - If there is an optimal, feasible solution, then there is an optimal, basic feasible solution
- Note, there could be a LARGE number of basic, feasible solutions
  - Simplex algorithm determines the optimal, basic feasible solution usually very quickly

LP Graphical Interpretation

- The LP constraints define a polyhedron in the solution space
  - This is a polytope if the polyhedron is bounded and nonempty
  - The basic, feasible solutions are vertices of this polyhedron
  - With the linear cost function the solution will be at one of vertices

APPENDIX 3B: LINEAR PROGRAMMING (LP) 11

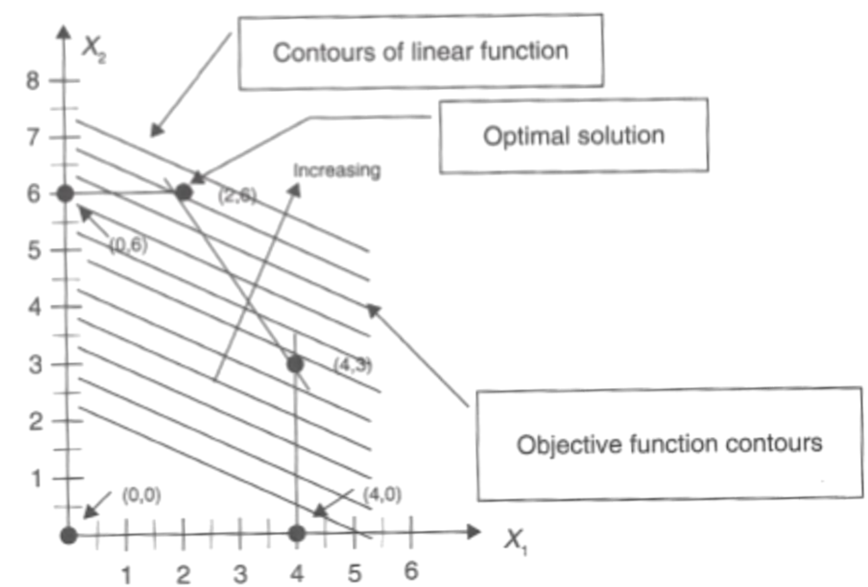


FIGURE 3.26 x_1, x_2 plane with cost contours and the optimal solution shown.

Image: Figure 3.26 from course text

Simplex Algorithm

- The key is to move intelligently from one basic feasible solution (i.e., a vertex) to another, with the goal of continually decreasing the cost function
- The algorithm does this by determining the “best” variable to bring into the basis; this requires that another variable exit the basis, while always retaining a basic, feasible solution
- This is called pivoting

Determination of Variable to Enter the Basis

- To determine which non-basic variable should enter the basis (i.e., one which currently 0), look at how the cost function changes w.r.t. to a change in a non-basic variable (i.e., one that is currently zero)

$$\text{Define } z = \mathbf{c}\mathbf{x} = [\mathbf{c}_B \quad \mathbf{c}_N]\begin{bmatrix}\mathbf{x}_B \\ \mathbf{x}_N\end{bmatrix}$$

$$\text{With } \mathbf{x}_\mathrm{B} = \mathbf{A}_B^{-1}\left(\mathbf{b} - \mathbf{A}_N\mathbf{x}_\mathrm{N}\right)$$

$$\text{Then } z = \mathbf{c}_B\mathbf{A}_B^{-1}\mathbf{b} + \left(\mathbf{c}_N - \mathbf{c}_B\mathbf{A}_B^{-1}\mathbf{A}_N\right)\mathbf{x}_\mathrm{N}$$

Elements of $\mathbf{x}_\mathrm{n}$ are all zero, but we are looking to change one to decrease the cost

Determination of Variable to Enter the Basis

- Define the reduced (or relative) cost coefficients as

$$\mathbf{r} = \mathbf{c}_N - \mathbf{c}_B \mathbf{A}_B^{-1} \mathbf{A}_N$$

r is an n-m dimensional row vector

- Elements of this vector tell how the cost function will change for a change in a currently non-basic variable
- The variable to enter the basis is usually the one with the most negative relative cost
- If all the relative costs are nonnegative then we are at an optimal solution

Determination of Variable to Exit Basis

- The new variable entering the basis, say a position j, causes the values of all the other basic variables to change. In order to retain a basic, feasible solution, we need to insure no basic variables become negative. The change in the basic variables is given by

$$\tilde{\mathbf{x}}_B = \mathbf{x}_B - \mathbf{A}_B^{-1}\mathbf{a}_j\,\varepsilon$$

where ε is the value of the variable entering the basis, and $\mathbf{a}_j$ is its associated column in $\mathbf{A}$

Determination of Variable to Exit Basis

We find the largest value ε such

$$\tilde{\mathbf{x}}_B = \mathbf{x}_B - \mathbf{A}_B^{-1}\mathbf{a}_j\varepsilon \geq \mathbf{0}$$

If no such ε exists then the problem is unbounded; otherwise at least one component of $\tilde{\mathbf{x}}_B$ equals zero. The associated variable exits the basis.

Canonical Form

- The Simplex Method works by having the problem in what is known as canonical form
- Canonical form is defined as having the m basic variables with the property that each appears in only one equation, its coefficient in that equation is unity, and none of the other basic variables appear in the same equation
- Sometime canonical form is readily apparent

$$\begin{aligned} &\text{Minimize} \quad -(100x_1 + 120x_2) \\ &\text{s.t.} \quad 2x_1 + 2x_2 + x_3 = 8 \\ &\qquad\quad 3x_1 + 5x_2 + x_4 = 15 \\ &\qquad\quad x_1, x_2, x_3, x_4 \geq 0 \end{aligned}$$

Note that with x_3 and x_4 as basic variables $\mathbf{A}_B$ is the identity matrix

Canonical Form

- Other times canonical form is achieved by initially adding artificial variables to get an initial solution
- Example of the nutrition problem in canonical form with slack and artificial variables (denoted as **y**) used to get an initial basic feasible solution

Let x_1=ounces of A, x_2= ounces of B

Minimize $y_1+y_2+y_3$

s.t.

$$2x_1+3x_2+x_3+y_1=20$$

$$x_1+3x_2-x_4+y_2=12$$

$$4x_1+3x_2-x_5+y_3=24$$

$$x_1,x_2,x_3,x_4,x_5,y_1,y_2,y_3\geq 0$$

Note that with y_1, y_2, and y_3 as basic variables $\mathbf{A}_B$ is the identity matrix

LP Tableau

- With the system in canonical form, the Simplex solution process can be illustrated by forming what is known as the LP tableau
  - Initially this corresponds to the **A** matrix, with a column appended to include the **b** vector, and a row added to give the relative cost coefficients; the last element is the negative of the cost function value
  - Define the tableau as **Y**, with elements $\mathbf{Y}_{ij}$
  - In canonical form the last column of the tableau gives the values of the basic variables
- During the solution the tableau is updated by pivoting

LP Tableau for the Nutrition Problem with Artificial Variables

- When in canonical form the relative costs vector is

$$\mathbf{r} = \mathbf{c}_N - \mathbf{c}_B \mathbf{A}_B^{-1} \mathbf{A}_N = \mathbf{c}_B \mathbf{A}_N$$

$$\mathbf{r} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}^T - \begin{bmatrix} 1 & 1 & 1 \end{bmatrix} \begin{bmatrix} 2 & 3 & 1 & 0 & 0 \\ 1 & 3 & 0 & -1 & 0 \\ 4 & 3 & 0 & 0 & -1 \end{bmatrix} = \begin{bmatrix} -7 \\ -9 \\ -1 \\ 1 \\ 1 \end{bmatrix}^T$$

- The initial tableau for the artificial problem is then

x_1	x_2	x_3	x_4	x_5	y_1	y_2	y_3	
2	3	1	0	0	1	0	0	20
1	3	0	–1	0	0	1	0	12
4	3	0	0	–1	0	0	1	24
–7	–9	–1	1	1	0	0	0	–56

Note the last column gives the values of the basic variables

LP Tableau Pivoting

- Pivoting is used to move from one basic feasible solution to another
  - Select the pivot column (i.e., the variable coming into the basis, say q) as the one with the most negative relative cost
  - Select the pivot row (i.e., the variable going out of the basis) as the one with the smallest ratio of x_i/Y_{iq} for $Y_{iq} > 0$; define this as row p (x_i is given in the last column)

That is, we find the largest value ε such

$$\tilde{\mathbf{x}}_B = \mathbf{x}_B - \mathbf{A}_B^{-1}\mathbf{a}_q\varepsilon \geq \mathbf{0}$$

If no such ε exists then the problem is unbounded;

otherwise at least one component of $\tilde{\mathbf{x}}_B$ equals zero.

The associated variable exits the basis.

LP Tableau Pivoting for Nutrition Problem

- Starting at

x_1	x_2	x_3	x_4	x_5	y_1	y_2	y_3	
2	3	1	0	0	1	0	0	20
1	3	0	–1	0	0	1	0	12
4	3	0	0	–1	0	0	1	24
–7	–9	–1	1	1	0	0	0	–56

- Pivot on column q=2; for row get minimum of {20/3, 12/3, 24/3), which is row p=2

LP Tableau Pivoting

- Pivoting on element Y_{pq} is done by
  - First dividing row p by Y_{pq} to change the pivot element to unity.
  - Then subtracting from the k^{th} row Y_{kq}/Y_{pq} times the p^{th} row for all rows with Y_{kq} <> 0

x_1	x_2	x_3	x_4	x_5	y_1	y_2	y_3	
2	3	1	0	0	1	0	0	20
1	**3**	0	–1	0	0	1	0	12
4	3	0	0	–1	0	0	1	24
–7	–9	–1	1	1	0	0	0	–56

I'm only showing fractions with two ROD digits

	x_1	x_2	x_3	x_4	x_5	y_1	y_2	y_3	
	1	0	1	1	0	1	–1	0	8
Pivoting gives	0.33	1	0	–0.33	0	0	0.33	0	4
	3	0	0	1	–1	0	–1	1	12
	–4	0	–1	–2	1	0	3	0	–20

LP Tableau Pivoting, Example, cont.

- Next pivot on column 1, row 3

x_1	x_2	x_3	x_4	x_5	y_1	y_2	y_3	
1	0	1	1	0	1	–1	0	8
0.33	1	0	–0.33	0	0	0.33	0	4
3	0	0	1	–1	0	–1	1	12
–4	0	–1	–2	1	0	3	0	–20

- Which gives

x_1	x_2	x_3	x_4	x_5	y_1	y_2	y_3	
0	0	1	0.67	0.33	1	–0.67	–0.33	4
0	1	0	–0.44	0.11	0	0.44	–0.11	2.67
1	0	0	0.33	–0.33	0	–0.33	0.33	4.0
0	0	–1	–0.67	–0.33	0	1.67	1.33	–4

LP Tableau Pivoting, Example, cont.

- Next pivot on column 3, row 1

x_1	x_2	x_3	x_4	x_5	y_1	y_2	y_3	
0	0	**1**	0.67	0.33	1	–0.67	–0.33	4
0	1	0	–0.44	0.11	0	0.44	–0.11	2.67
1	0	0	0.33	–0.33	0	–0.33	0.33	4
0	0	–1	–0.67	–0.33	0	1.67	1.33	–4

- Which gives

x_1	x_2	x_3	x_4	x_5	y_1	y_2	y_3	
0	0	1	0.67	0.33	1	–0.67	–0.33	4
0	1	0	–0.44	0.11	0	0.44	–0.11	2.67
1	0	0	0.33	–0.33	0	–0.33	0.33	4
0	0	0	0	0	1	1	1	0

Since there are no negative relative costs we are done with getting a starting solution

LP Tableau Full Problem

- The tableau from the end of the artificial problem is used as the starting point for the actual solution
  - Remove the artificial variables
  - Update the relative costs with the costs from the original problem and update the bottom right-hand corner value

$$\mathbf{c} = [0.2 \quad 0.25 \quad 0 \quad 0 \quad 0]$$

$$\mathbf{r} = \mathbf{c}_N - \mathbf{c}_B \mathbf{A}_B^{-1} \mathbf{A}_N = \mathbf{c}_B \mathbf{A}_N$$

$$\mathbf{r} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}^T - [0 \quad 0.25 \quad 0.2] \begin{bmatrix} 0.67 & 0.33 \\ -0.44 & 0.11 \\ 0.33 & -0.33 \end{bmatrix} = \begin{bmatrix} 0.04 \\ 0.04 \end{bmatrix}^T$$

- Since none of the relative costs are negative we are done with x_1=4, x_2=2.7 and x_3=4

Marginal Costs of Constraint Enforcement

If we would like to determine how the cost function will change for changes in **b**, assuming the set of basic variables does not change then we need to calculate

$$\frac{\partial z}{\partial \mathbf{b}} = \frac{\partial(\mathbf{c}_B \mathbf{x}_B)}{\partial \mathbf{b}} = \frac{\partial(\mathbf{c}_B \mathbf{A}_B^{-1}\mathbf{b})}{\partial \mathbf{b}} = \mathbf{c}_B \mathbf{A}_B^{-1} = \boldsymbol{\lambda}$$

So the values of $\boldsymbol{\lambda}$ tell the marginal cost of enforcing each constraint.

The marginal costs will be used to determine the OPF locational marginal costs (LMPs)

Nutrition Problem Marginal Costs

- In this problem we had basic variables 1, 2, 3; nonbasic variables of 4 and 5

$$\mathbf{x}_B = \mathbf{A}_B^{-1}\left(\mathbf{b} - \mathbf{A}_N \mathbf{x}_N\right) = \begin{bmatrix} 2 & 3 & 1 \\ 1 & 3 & 0 \\ 4 & 3 & 0 \end{bmatrix}^{-1} \begin{bmatrix} 20 \\ 12 \\ 24 \end{bmatrix} = \begin{bmatrix} 4 \\ 2.67 \\ 4 \end{bmatrix}$$

$$\boldsymbol{\lambda} = \mathbf{c}_B \mathbf{A}_B^{-1} = \begin{bmatrix} 0.2 & 0.25 & 0 \end{bmatrix} \begin{bmatrix} 2 & 3 & 1 \\ 1 & 3 & 0 \\ 4 & 3 & 0 \end{bmatrix}^{-1} = \begin{bmatrix} 0 \\ 0.044 \\ 0.039 \end{bmatrix}$$

There is no marginal cost with the first constraint since it is not binding; values tell how cost changes if the **b** values were changed

Lumber Mill Example Solution

Minimize $-(100x_1 + 120x_2)$

s.t. $2x_1 + 2x_2 + x_3 = 8$

$3x_1 + 5x_2 + x_4 = 15$

$x_1, x_2, x_3, x_4 \geq 0$

An initial basic feasible solution is $x_1 = 0, x_2 = 0, x_3 = 8, x_4 = 15$

The solution is $x_1 = 2.5, x_2 = 1.5, x_3 = 0, x_4 = 0$

Then $\boldsymbol{\lambda} = \begin{bmatrix} 100 & 120 \end{bmatrix} \begin{bmatrix} 2 & 2 \\ 3 & 5 \end{bmatrix}^{-1} = \begin{bmatrix} 35 \\ 10 \end{bmatrix}$

Economic interpretation of λ is the profit is increased by 35 for every hour we up the first constraint (the saw) and by 10 for every hour we up the second constraint (plane)

Complications

- Often variables are not limited to being ≥ 0
  - Variables with just a single limit can be handled by substitution; for example if x ≥ 5 then x-5=z ≥ 0
  - Bounded variables, high ≥ x ≥ 0 can be handled with a slack variable so x + y = high, and x,y ≥ 0
- Unbounded conditions need to be detected (i.e., unable to pivot); also the solution set could be null

Minimize $x_1 - x_2$ s.t. $x_1 + x_2 \geq 8$

$\rightarrow x_1 + x_2 - y_1 = 8 \rightarrow x_2 = 8$ is a basic feasible solution

x_1	x_2	y_1	
1	1	–1	8
2	0	–1	8

Complications

- Degenerate Solutions
  - Occur when there are less than m basic variables > 0
  - When this occurs the variable entering the basis could also have a value of zero; it is possible to cycle, anti-cycling techniques could be used
- Nonlinear cost functions
  - Nonlinear cost functions could be approximated by assuming a piecewise linear cost function
- Integer variables
  - Sometimes some variables must be integers; known as integer programming; we'll discuss after some power examples

LP Optimal Power Flow

- LP OPF was introduced in
  - B. Stott, E. Hobson, "Power System Security Control Calculations using Linear Programming," (Parts 1 and 2) *IEEE Trans. Power App and Syst.*, Sept/Oct 1978
  - O. Alsac, J. Bright, M. Prais, B. Stott, "Further Developments in LP-based Optimal Power Flow," *IEEE Trans. Power Systems*, August 1990
- It is a widely used technique, particularly for real power optimization; it is the technique used in PowerWorld