Civil Engineering Systems Analysis Lecture VIII

Instructor: Prof. Naveen Eluru
Department of Civil Engineering and Applied Mechanics

Today's Learning Objectives

- Mid-term Exam
- Problem Set 1 Solutions
- Simplex Method

Mid-term Exam

- Date and Time
- October 18th 1.05 – 2.25
  - No extra time will be provided [so be on time]

- Rooms
  - Trottier 100 [Even number student ids]
  - ENGMC13 [Odd number student ids]

- TAs will be monitoring the rooms and I will be moving from one to the other

Problem Set 1

- Solutions

Simplex Theory

Simplex Theory

- In the previous discussions, we examined simplex method in terms of some constraints and equations.
- Now lets try to discuss the approach in terms of generalized expressions!
- Maximize $Z = c_1x_1 + c_2x_2 + \ldots c_nx_n$
  - subject to

$$a_{11}x_1 + a_{12}x_2 + ..a_{1n}x_n \leq b_1$$
$$a_{21}x_1 + a_{22}x_2 + ..a_{2n}x_n \leq b_2$$

…

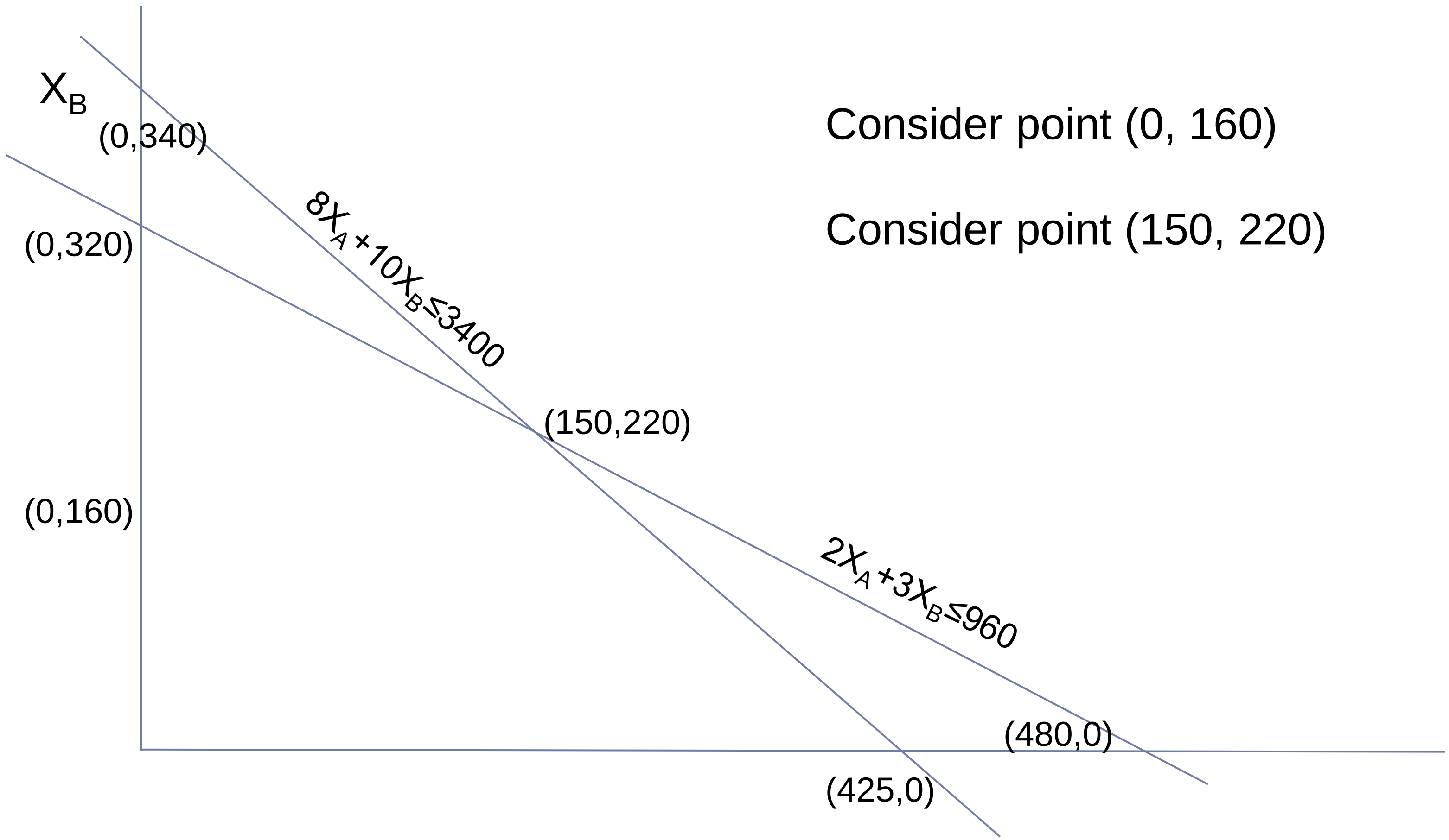
…

$$a_{m1}x_1 + a_{m2}x_2 + ..a_{mn}x_n \leq b_m$$
$$x_1 \geq 0,\ x_2 \geq 0,\ \ldots\ x_n \geq 0$$

Simplex Theory

- In 2-D each constraint is represented as a line, in 3-D it becomes a plane, in higher dimensions each constraint is represented as a hyper plane in n-dimensional space
  - The hyper plane forms the constraint boundary
- Corner point feasible solution
  - A feasible solution that does not lie on any segment connecting two other feasible solutions.

Illustration



Consider point (0, 160)

Consider point (150, 220)

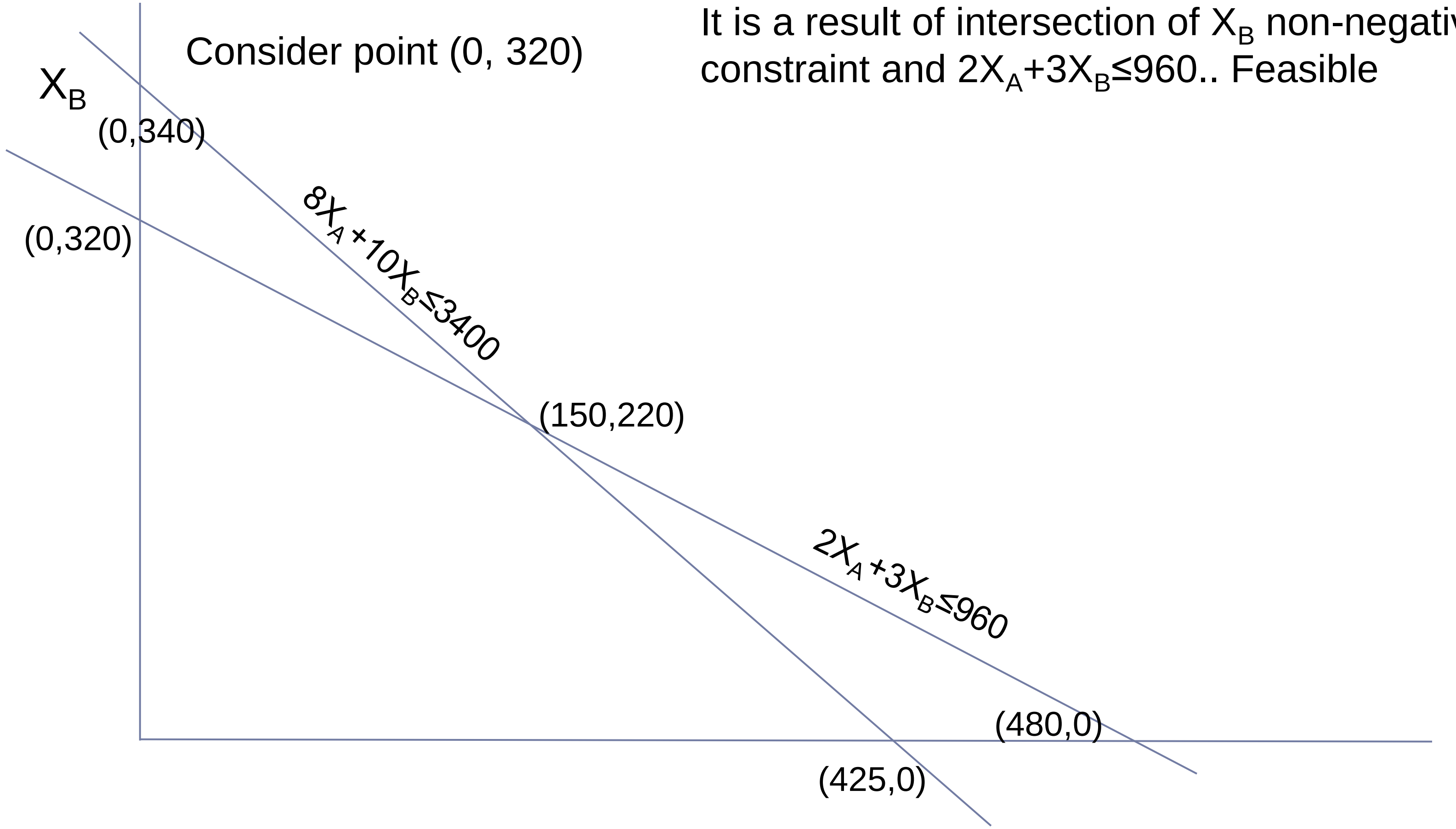
Simplex Theory

- For any LP with n decision variables, each CPF lies at the intersection of n constraint boundaries
  - It is solution to the n simultaneous equations
- However, this is not to say that every set of n constraints from n+m constraints (n: non-negativity and m: functional) yields a CPF. It is possible that some equations have no feasible solution
- To summarize, the system of 2 decision variables and 4 constraints have 6 solutions (Of these 4 are feasible and 2 are infeasible)

Illustration

Consider point (0, 320)

It is a result of intersection of X_B non-negativity constraint and $2X_A+3X_B \le 960$.. Feasible



Consider point (480, 0)

It is a result of intersection of X_A non-negativity and $2X_A+3X_B \le 960$… Infeasible

Simplex Theory

- Simplex moves from iteration to iteration through one CPF to another adjacent CPF
- Two CPF are adjacent if the line segment connecting them forms the edge of the feasible region
- From each CPF there are n edges emanating .. Each leading to n CPF solutions
- In the simplex, we move from one CPF to another by moving along one of the edges. When the simplex method chooses an entering basic variable, the geometric interpretation is that it is choosing one of the edges emanating from the current CPF solution to move along.
- Increasing this variable from zero (and simultaneously changing the values of the other basic variables accordingly) corresponds to moving along this edge. Having one of the basic variables (the leaving basic variable) decrease so far that it reaches zero corresponds to reaching the first new constraint boundary at the other end of this edge of the feasible region

Simplex Theory

- Properties of CPF solutions
- (1a) If there is exactly one optimal solution, then it must be a CPF solution.
  - Lets prove this by contradiction i.e. the optimal solution x^* (optimal obj. function Z^*) is not a CPF
  - So, $x^* = \alpha\, x1 + (1-\alpha)x2$ for $0 < \alpha < 1$
  - Thus $Z^* = \alpha Z1 + (1-\alpha)Z2$
  - => three possibilities
    - $Z^*=Z1=Z2$; $Z1<Z^*<Z2$; $Z1>Z^*>Z2$
    - $Z^*=Z1=Z2$ contradicts that there is only one solution
    - The second and third contradict the optimality of Z^* itself!
    - Hence our assumption is wrong and so if there is only one optimal solution it has to be a CPF

Simplex Theory

- Properties of CPF solutions
- (1b) If there are multiple optimal solutions (and a bounded feasible region), then at least two must be adjacent CPF solutions
  - In 2-D this would mean the obj. func is parallel to one of the constraints. So, the obj. func will increase until we hit the constraint and create a part of the line which is optimal (with two CPF solns at the ends)
  - In n-D it will just end up being a hyper plane
- What 1(a) and 1(b) allow us is the freedom to only examine CPFs

Simplex Theory

- Properties of CPF solutions
- (2) There are only a finite CPF solutions
- In our example with 2 decision variables and 4 constraints we had 6 possible systems of equations. Of these only 4 yielded CPF but 6 is the upper bound
- For n decision variables and n+m constraints the upper bound on the CPF is (n+m)Cn = $(n+m)!/[(n!)(m!)]$

Simplex Theory

- Properties of CPF solutions
- (3) If a CPF solution has no adjacent CPF solutions that are better (as measured by Z), then there are no better CPF solutions anywhere.
- This comes from Property 1b… if there were another CPF solution then it would have been adjacent to this..
- So far we are examining only the LP as it is (without augmentation)

Simplex Theory

- Once we start augmenting, we start defining the basis
- Basic solutions are augmented corner point solutions
- Basic feasible solutions are augmented CPF solutions
- Each basic solution has m basic variables (same as constraints) and rest of the variables are non-basic and set to 0
- Basic Feasible solution is a basic solution where all m basic variables are non-negative
- BF solution is degenerate if any of the basic variables has a value of 0

Other Solution Approaches

Other Solution Approaches to LP problem

- A discovery was made in 1984 by a young mathematician at AT&T Bell Laboratories, Narendra Karmarkar
- The algorithm is also an iterative procedure. We start with a feasible *trial solution.* Then we proceed from current to the next feasible solution. The difference lies in the nature of these solutions
- For the simplex method, the trial solutions are *CPF solutions* (or BF solutions after augmenting), so all movement is along edges on the *boundary* of the feasible region.
- For Karmarkar's algorithm, the trial solutions are **interior points,** i.e., points *inside* the boundary of the feasible region. For this reason, Karmarkar's algorithm and its variants are referred to as **interior-point algorithms**

Comparison with Simplex Method

- Simplex is an exponential time algorithm while the Karmarkar method is proven to have a polynomial bound – an indicator of worst-case performance
- However, the bound does not provide information on average performance
- Average performance is a function of no. of iterations and time per iteration.
- Interior point methods are more complicated and hence take longer per iteration.
- For fairly small problems, the numbers of iterations needed by an interior-point algorithm and by the simplex method tend to be somewhat comparable.

Comparison with Simplex Method

- For example, on a problem with 10 functional constraints, roughly 20 iterations would be typical for either kind of algorithm. Consequently, on problems of similar size, the total computer time for an interior-point algorithm will tend to be many times longer than that for the simplex method.
- On the other hand, a key advantage of interior-point algorithms is that large problems do not require many more iterations than small problems.
- For example, a problem with 10,000 functional constraints probably will require well under 100 iterations. Even considering the very substantial computer time per iteration needed for a problem of this size, such a small number of iterations makes the problem quite tractable.
- By contrast, the simplex method might need 20,000 iterations and so might not finish within a reasonable amount of computer time. Therefore, interior-point algorithms often are faster than the simplex method for such huge problems.
- An issue with the Karmarkar method is that the approach is not suited for undertaking the post-optimality analysis. That's a big drawback

Heuristics

- Sometimes the LP methods are not adequate to solve optimization problems
  - Approximate algorithms are used in this case

- Examples
  - Genetic algorithms
  - Simulated annealing techniques
  - Ant colony optimization methods

Matrix Implementation

Simplex Method in Matrix Form

- Prior to starting this discussion
- To help you distinguish between matrices, vectors, and scalars, we consistently use **BOLDFACE CAPITAL** letters to represent matrices, **boldface lowercase** letters to represent vectors, and *italicized* letters in ordinary print to represent scalars.
- We also use a boldface zero (**0**) to denote a *null vector* (a vector whose elements all are zero) in either column or row form, whereas a zero in ordinary print (0) continues to represent the number zero

Simplex Method in Matrix Form

- Maximize Z = **cx**
- subject to **Ax** ≤ **b** and **x** ≥ **0,** where **c** is the row vector
- **c** = $[c_1, c_2, \ldots c_n]$
- where **x**, **b**, and **0** are the column vectors, **A** is matrix
- $\mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \\ . \\ . \\ . \\ x_n \end{bmatrix}$ and $\mathbf{b} = \begin{bmatrix} b_1 \\ b_2 \\ . \\ . \\ . \\ b_m \end{bmatrix}$ and $\mathbf{0} = \begin{bmatrix} 0 \\ 0 \\ . \\ . \\ . \\ 0 \end{bmatrix}$ $\mathbf{A} = \begin{bmatrix} a_{11} & a_{12} & \ldots\ a_{1n} \\ a_{21} & a_{22} & \ldots\ a_{2n} \\ \ldots & \ldots & \ldots\ \ \ldots \\ a_{m1} & a_{m2} & \ldots\ a_{mn} \end{bmatrix}$

Simplex Method in Matrix Form

- To obtain the augmented form of the problem introduce the column of slack variables
- $\mathbf{x_s} = \begin{bmatrix} x_{n+1} \\ x_{n+2} \\ \cdot \\ \cdot \\ \cdot \\ x_{n+m} \end{bmatrix}$
- so that the constraints become
- $[\mathbf{A} \;\; \mathbf{I}] \begin{bmatrix} \mathbf{x} \\ \mathbf{x_s} \end{bmatrix} = \mathrm{b}$ and $\begin{bmatrix} \mathbf{x} \\ \mathbf{x_s} \end{bmatrix} \geq \mathbf{0},$
- where I is the m x m identity matrix and 0 – null vector with $n+m$ elements

Simplex Method in Matrix Form

- Solving for a basic feasible solution
- n non-basic variables from n+m elements of $\begin{bmatrix} \mathbf{X} \\ \mathbf{X_s} \end{bmatrix}$ are set to 0
- Eliminating these *n* variables by equating them to zero leaves a set of *m* equations and *m basic variables*
- Denoted as
- $\mathbf{Bx_B} = \mathbf{b}$
- where vector of basic variables
- $\mathbf{x_B} = \begin{bmatrix} x_{B1} \\ x_{B2} \\ . \\ . \\ . \\ x_{Bm} \end{bmatrix}$ and basis matrix B = $\begin{bmatrix} B_{11} & B_{12} & \dots B_{1m} \\ B_{21} & B_{22} & \dots B_{2m} \\ \dots & \dots & \dots \quad \dots \\ B_{m1} & B_{m2} & \dots B_{mm} \end{bmatrix}$ is

obtained by eliminating the columns corresponding to coefficients of non-basic variables from [**A I**]

Simplex Method in Matrix Form

- The simplex method introduces only basic variables such that $\mathbf{B}$ is *nonsingular*, so that $\mathbf{B}^{-1}$ always will exist
- Therefore, to solve $\mathbf{B}\mathbf{x}_{\mathbf{B}} = \mathbf{b}$, both sides are premultiplied by $\mathbf{B}^{-1}$
- $\mathbf{B}^{-1}\mathbf{B}\mathbf{x}_{\mathbf{B}} = \mathbf{B}^{-1}\mathbf{b}$.
- $\mathbf{x}_{\mathbf{B}} = \mathbf{B}^{-1}\mathbf{b}$.

- Let $\mathbf{c}_{\mathbf{B}}$ represent the coefficient vector corresponding to Basic variables ($\mathbf{x}_{\mathbf{B}}$) – including 0's for slack variables
- The objective function is $Z = \mathbf{c}_{\mathbf{B}}\mathbf{x}_{\mathbf{B}} = \mathbf{c}_{\mathbf{B}}\mathbf{B}^{-1}\mathbf{b}$

Simplex Method in Matrix Form

- Example: Lets get back to our classical example
  - Maximize 22 X_A+28X_B
    - $8X_A+10X_B \leq 3400$
    - $2X_A+3X_B \leq 960$
    - $X_A, X_B \geq 0$
- c = [22, 28]
- $[A\ I] = \begin{bmatrix} 8 & 10 & 1 & 0 \\ 2 & 3 & 0 & 1 \end{bmatrix}$
- $b = \begin{bmatrix} 3400 \\ 960 \end{bmatrix}$
- $x = \begin{bmatrix} x_A \\ x_B \end{bmatrix}$, $x_s = \begin{bmatrix} y_A \\ y_B \end{bmatrix}$
- <u>Iteration 0</u>
- $x_B = \begin{bmatrix} y_A \\ y_B \end{bmatrix}$; $B = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} = B^{-1}$
- so $\begin{bmatrix} y_A \\ y_B \end{bmatrix} = \begin{bmatrix} 3400 \\ 960 \end{bmatrix}$
- $c_B = [\ 0, 0]$, $Z = [0, 0] \begin{bmatrix} 3400 \\ 960 \end{bmatrix} = 0$

References

- Hillier F.S and G. J. Lieberman. Introduction to Operations Research, Ninth Edition, McGraw-Hill, 2010