Linear Programming Applications

Software for Linear Programming

Objectives

- Use of software to solve LP problems
- MMO Software with example
  - Graphical Method
  - Simplex Method
- Simplex method using optimization toolbox of MATLAB

MMO Software
Mathematical Models for Optimization

An MS-DOS based software

Used to solve different optimization problems

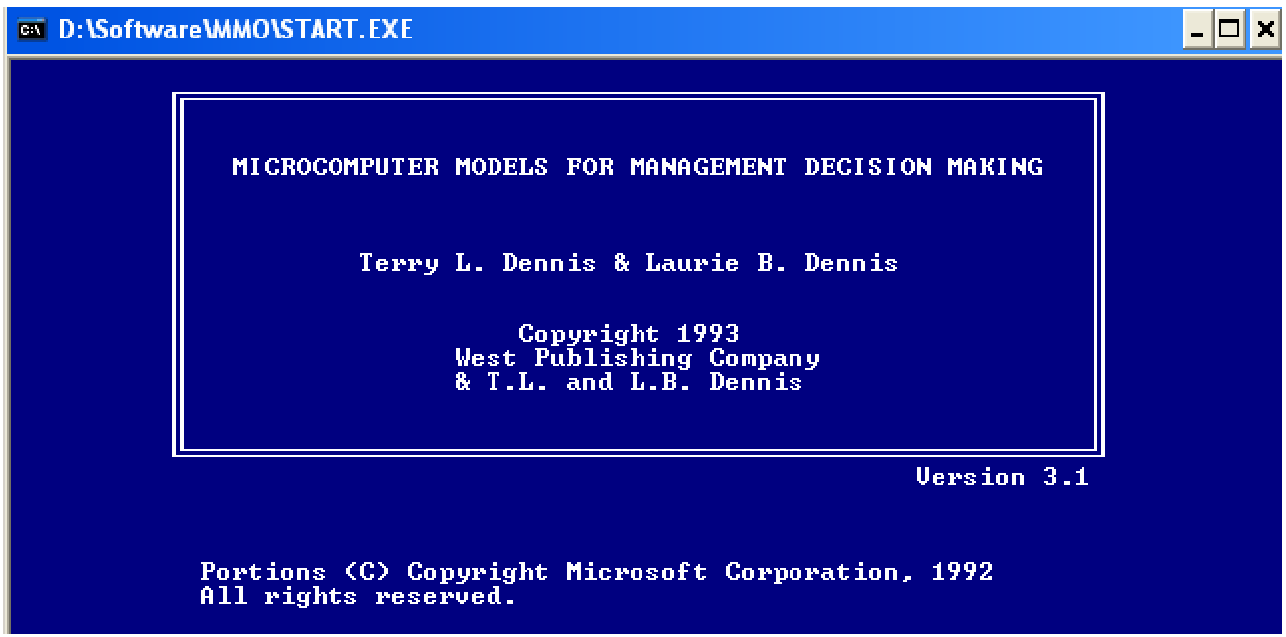
- Graphical method and Simplex method will be discussed.

Installation

- Download the file “MMO.ZIP” and unzip it in a folder in the PC
- Open this folder and double click on the application file named as “START”. It will open the MMO software

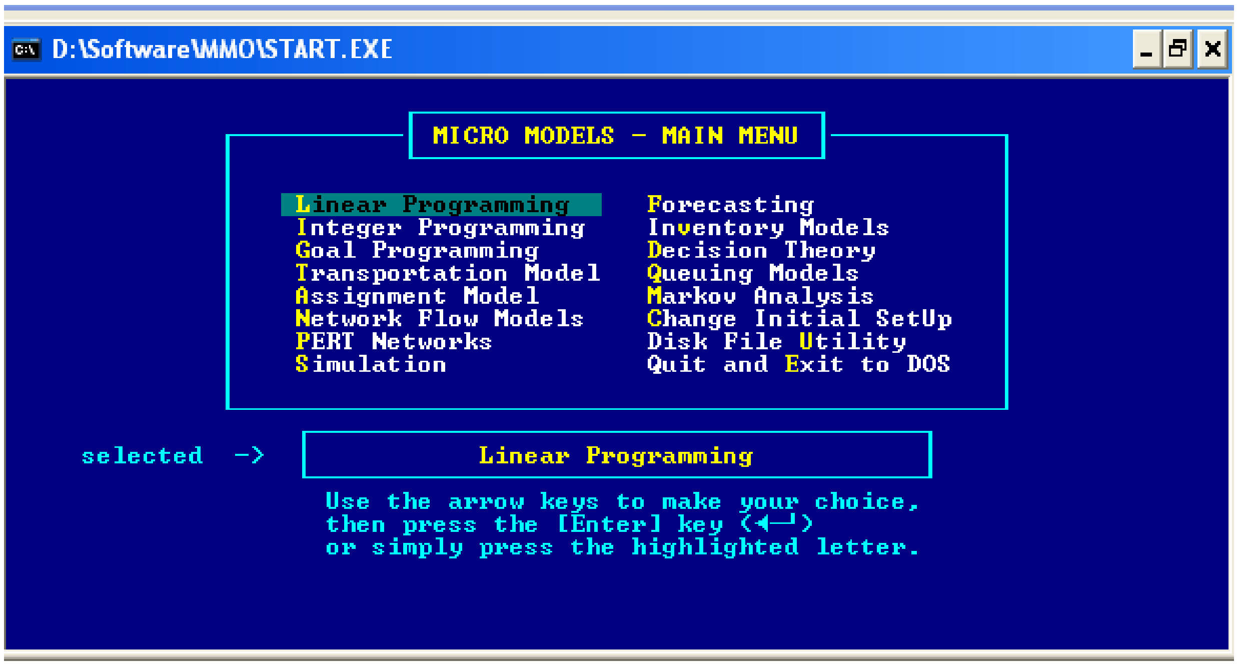
Working with MMO

Opening Screen



Working with MMO

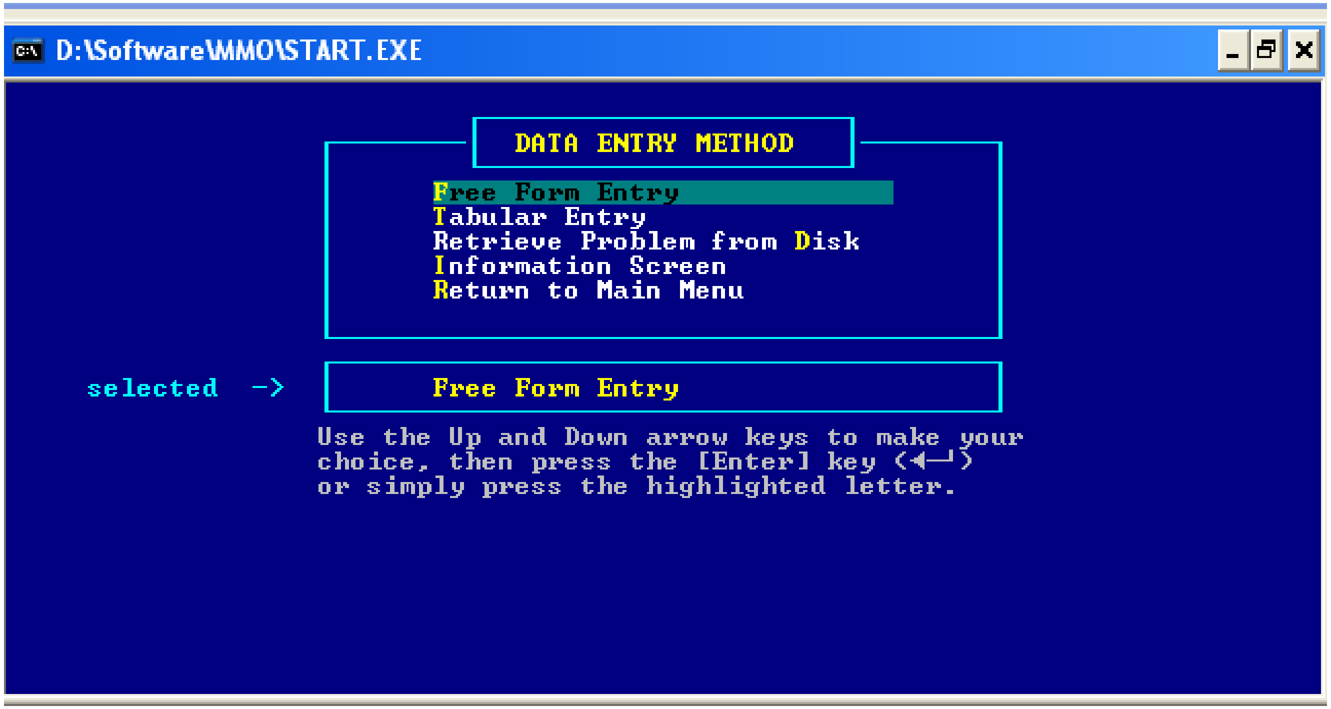
Starting Screen



SOLUTION METHOD: GRAPHIC/ SIMPLEX

Graphical Method

Data Entry



Data Entry: Few Notes

- Free Form Entry: Write the equation at the prompted input.
- Tabular Entry: Spreadsheet style. Only the coefficients are to be entered, not the variables.
- All variables must appear in the objective function (even those with a 0 coefficient)
- Constraints can be entered in any order; variables with 0 coefficients do not have to be entered
- Constraints may not have negative right-hand-sides (multiply by -1 to convert them before entering)
- When entering inequalities using < or >, it is not necessary to add the equal sign (=)
- Non-negativity constraints are assumed and need not be entered

Example

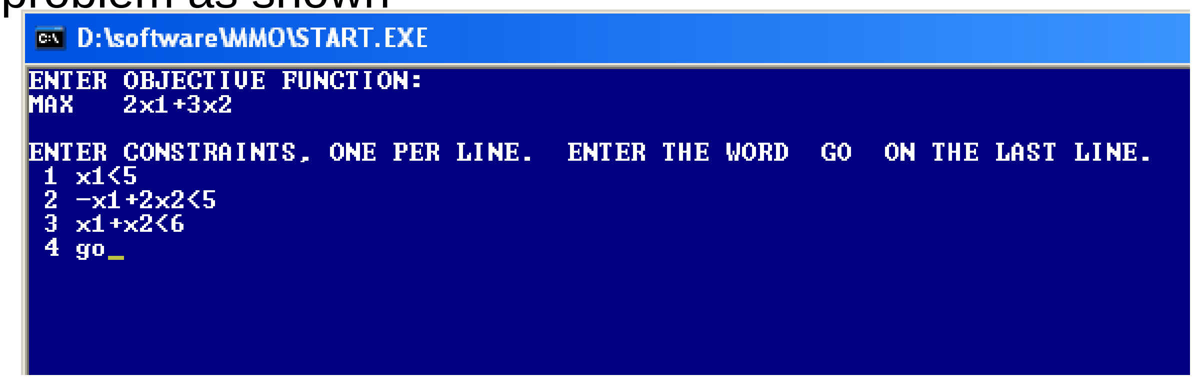
Let us consider the following problem

$$\begin{aligned} &Maximize \quad && Z = 2x_1 + 3x_2 \\ &Subject\ to && x_1 \leq 5, \\ & && x_1 - 2x_2 \geq -5, \\ & && x_1 + x_2 \leq 6 \\ & && x_1, x_2 \geq 0 \end{aligned}$$

Note: The second constraint is to be multiplied by -1 while entering, i.e. $-x_1 + 2x_2 \leq 5$

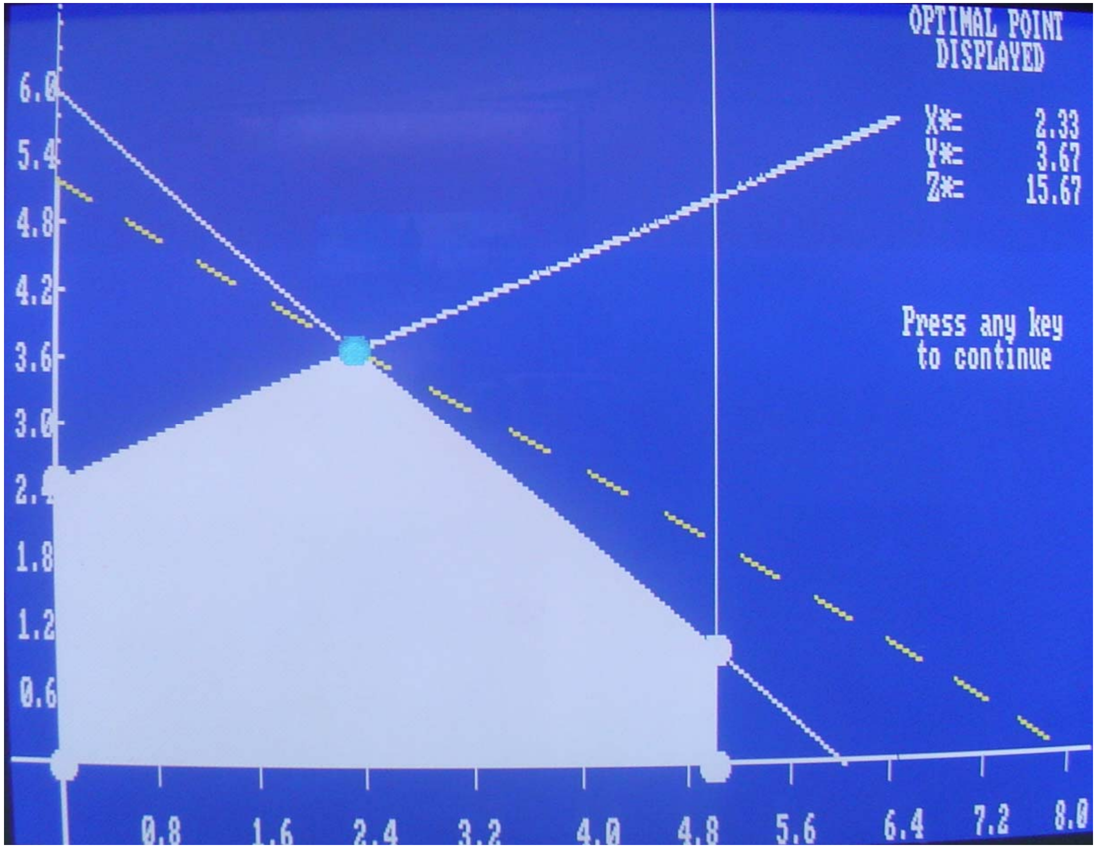
Steps in MMO Software

- Select ‘Free Form Entry’ and Select ‘TYPE OF PROBLEM’ as ‘MAX’
- Enter the problem as shown



- Write ‘go’ at the last line of the constraints
- Press enter
- Checking the proper entry of the problem
  - If any mistake is found, select ‘NO’ and correct the mistake
  - If everything is ok, select ‘YES’ and press the enter key

Solution



Z=15.67

x_1=2.33

x_2=3.67

F1: Redraw

F2: Rescale

F3: Move Objective Function Line

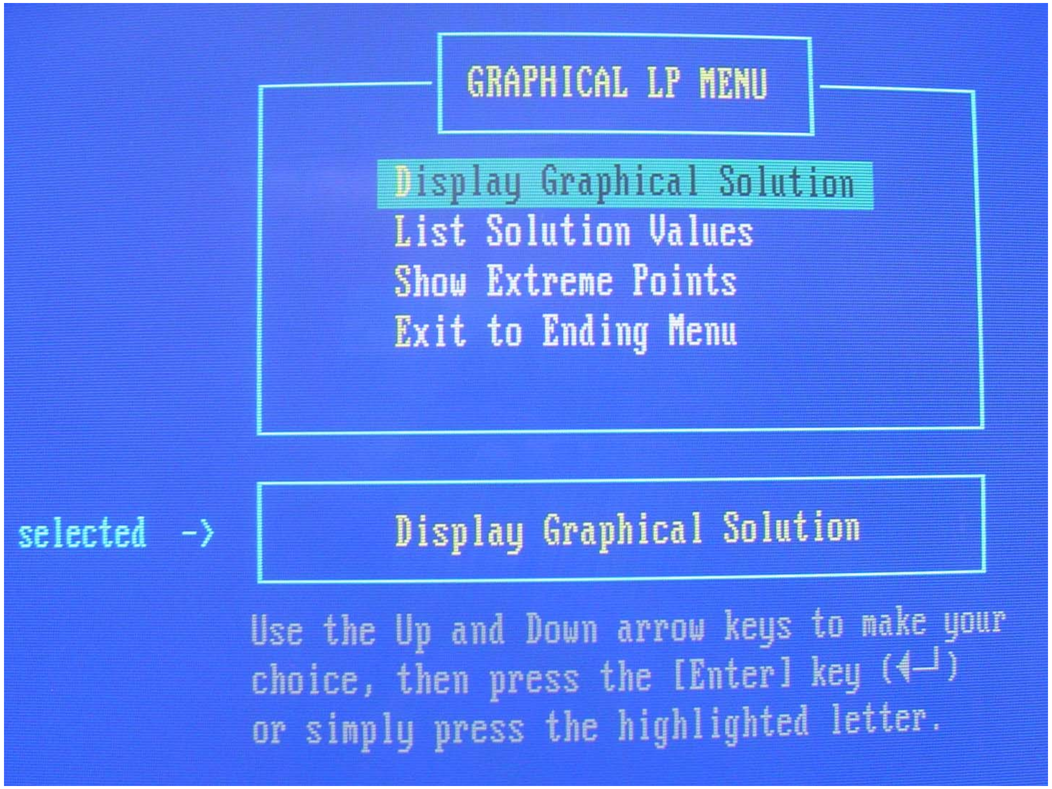
F4: Shade Feasible Region

F5: Show Feasible Points

F6: Show Optimal Solution Point

F10: Show Graphical LP Menu (GPL)

Graphical LP Menu



Extreme points and feasible extreme points

EXTREME POINTS:

	X1	X2
1	5.00	0.00
2	5.00	1.00
3	5.00	5.00
4	-5.00	0.00
5	0.00	2.50
6	2.33	3.67
7	6.00	0.00
8	0.00	6.00
9	0.00	0.00

FEASIBLE EXTREME POINTS:

	X1	X2	OBJ FUNCT VALUE
1	5.00	0.00	10.00
2	5.00	1.00	13.00
3	0.00	2.50	7.50
4	2.33	3.67	15.66
5	0.00	0.00	0.00

Simplex Method using MMO

- Simplex method can be used for any number of variables
- Select SIMPLEX and press enter.
- As before, screen for “data entry method” will appear
- The data entry is exactly same as discussed before.

Example

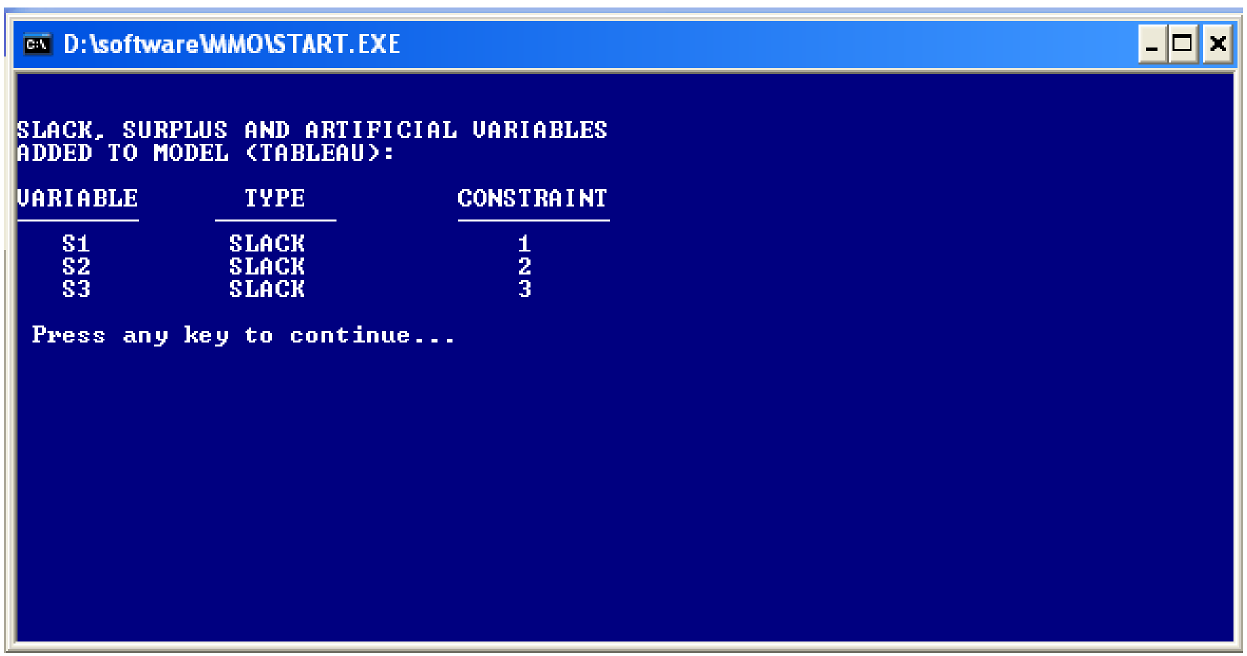
Let us consider the same problem.

(However, a problem with more than two decision variables can also be taken)

$$\begin{aligned} &\textit{Maximize} \quad && Z = 2x_1 + 3x_2 \\ &\textit{Subject to} \quad && x_1 \le 5, \\ & && x_1 - 2x_2 \ge -5, \\ & && x_1 + x_2 \le 6 \\ & && x_1, x_2 \ge 0 \end{aligned}$$

Slack, surplus and artificial variables

- There are three additional slack variables



Different options for Simplex tableau

- No Tableau: Shows direct solutions
- All Tableau: Shows all simplex tableau one by one
- Final Tableau: Shows only the final simplex tableau directly

Final Simplex tableau and solution

D:\software\MMO\START.EXE

TABLEAU NUMBER 3

C(j)		2	3	0	0	0	
BASIC	VAR	X1	X2	S1	S2	S3	RHS
0	S1	0	0	1	.333	-.667	2.667
3	X2	0	1	0	.333	.333	3.667
2	X1	1	0	0	-.333	.667	2.333
	Z	2	3	0	.333	2.333	15.667
	C-Z	0	0	0	-.333	-2.333	

Press any key to continue...

Final Solution

$Z=15.67$

$x_1=2.33$

$x_2=3.67$

MATLAB Toolbox for Linear Programming

- Very popular and efficient
- Includes different types of optimization techniques
- To use the simplex method
  - set the option as

    options = optimset ('LargeScale', 'off', 'Simplex', 'on')

  - then a function called 'linprog' is to be used

MATLAB Toolbox for Linear Programming

linprog

Solve a linear programming problem

$$\min_{x} f^T x \quad \text{such that} \quad \begin{aligned} A \cdot x &\le b \\ Aeq \cdot x &= beq \\ lb \le x &\le ub \end{aligned}$$

where *f*, *x*, *b*, *beq*, *lb*, and *ub* are vectors and *A* and *Aeq* are matrices.

Syntax

```
x = linprog(f,A,b,Aeq,beq)
x = linprog(f,A,b,Aeq,beq,lb,ub)
x = linprog(f,A,b,Aeq,beq,lb,ub,x0)
x = linprog(f,A,b,Aeq,beq,lb,ub,x0,options)
[x,fval] = linprog(...)
[x,fval,exitflag] = linprog(...)
[x,fval,exitflag,output] = linprog(...)
[x,fval,exitflag,output,lambda] = linprog(...)
```

MATLAB Toolbox for Linear Programming

Description

`linprog` solves linear programming problems.

`x = linprog(f,A,b)` solves min `f'*x` such that `A*x <= b`.

`x = linprog(f,A,b,Aeq,beq)` solves the problem above while additionally satisfying the equality constraints `Aeq*x = beq`. Set `A=[]` and `b=[]` if no inequalities exist.

`x = linprog(f,A,b,Aeq,beq,lb,ub)` defines a set of lower and upper bounds on the design variables, `x`, so that the solution is always in the range `lb <= x <= ub`. Set `Aeq=[]` and `beq=[]` if no equalities exist.

`x = linprog(f,A,b,Aeq,beq,lb,ub,x0)` sets the starting point to `x0`. This option is only available with the medium-scale algorithm (the `LargeScale` option is set to `'off'` using `optimset`). The default large-scale algorithm and the simplex algorithm ignore any starting point.

`x = linprog(f,A,b,Aeq,beq,lb,ub,x0,options)` minimizes with the optimization options specified in the structure `options`. Use `optimset` to set these options.

`[x,fval] = linprog(...)` returns the value of the objective function `fun` at the solution `x`: `fval = f'*x`.

`[x,lambda,exitflag] = linprog(...)` returns a value `exitflag` that describes the exit condition.

`[x,lambda,exitflag,output] = linprog(...)` returns a structure `output` that contains information about the optimization.

Example

Let us consider the same problem as before

$$\begin{aligned} &\textit{Maximize} \quad && Z = 2x_1 + 3x_2 \\ &\textit{Subject to} && x_1 \le 5, \\ & && x_1 - 2x_2 \ge -5, \\ & && x_1 + x_2 \le 6 \\ & && x_1, x_2 \ge 0 \end{aligned}$$

Note: The maximization problem should be converted to minimization problem in MATLAB

Example… contd.

Thus,

$$f = [-2 \quad -3] \qquad \text{\% Cost coefficients}$$

$$A = \begin{bmatrix} 1 & 0 \\ -1 & 2 \\ 1 & 1 \end{bmatrix} \qquad \text{\% Coefficients of constraints}$$

$$b = [5 \quad 5 \quad 6] \qquad \text{\% Right hand side of constraints}$$

$$lb = [0 \quad 0] \qquad \text{\% Lowerbounds of decision variables}$$

Example… contd.

MATLAB code

```
clear all
f=[-2 -3];              %Converted to minimization problem
A=[1 0;-1 2;1 1];
b=[5 5 6];
lb=[0 0];
options = optimset ('LargeScale', 'off', 'Simplex', 'on');
[x , fval]=linprog (f , A , b , [ ] , [ ] , lb );
Z = -fval               %Multiplied by -1
x
```

Solution

Z = 15.667 **with** $x_1 = 2.333$ **and** $x_2 = 3.667$

Thank You