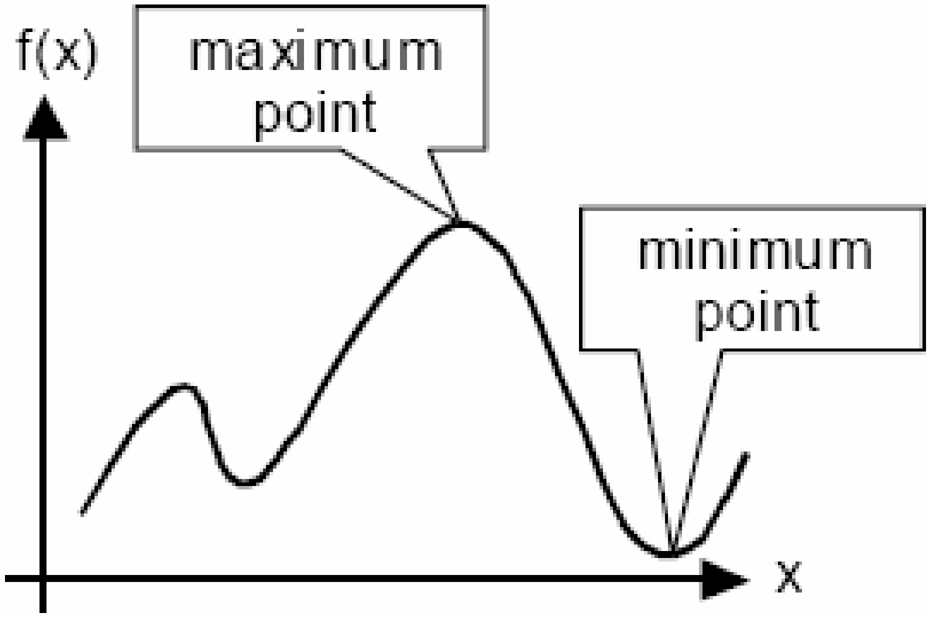
# Lessons of Linear Programming

Abu (Sayeem) Reaz
(with special thanks to Dr. Massimo Tornatore)
University of California, Davis, USA

Group Presentation
May 22, 2009

# Unconstrained Optimization

- Finding the maximum or minimum point on a function




**Simple unconstrained optimization.**

# Constrained Optimization

- Most popular form of constrained optimization: Linear Programming
  - Find the ‘Best Point’ on a function
  - Respect the constraints
  - Respect the bounds
  - The best solution may not occur at the top of a peak or bottom of a valley

- Linear Program (LP) has following components:
  - Variables
  - Objective Function
  - Constraints
  - Bounds

# An Example (1)

- Acme Bicycle Company
  - Profit for Mountain Bike = $15
  - Profit for Racer Bike = $10
  - 2 Mountain Bikes per day
  - 3 Racer Bikes per day
  - 4 metal finishing per day
- Linear Program (LP) to maximize profit Z

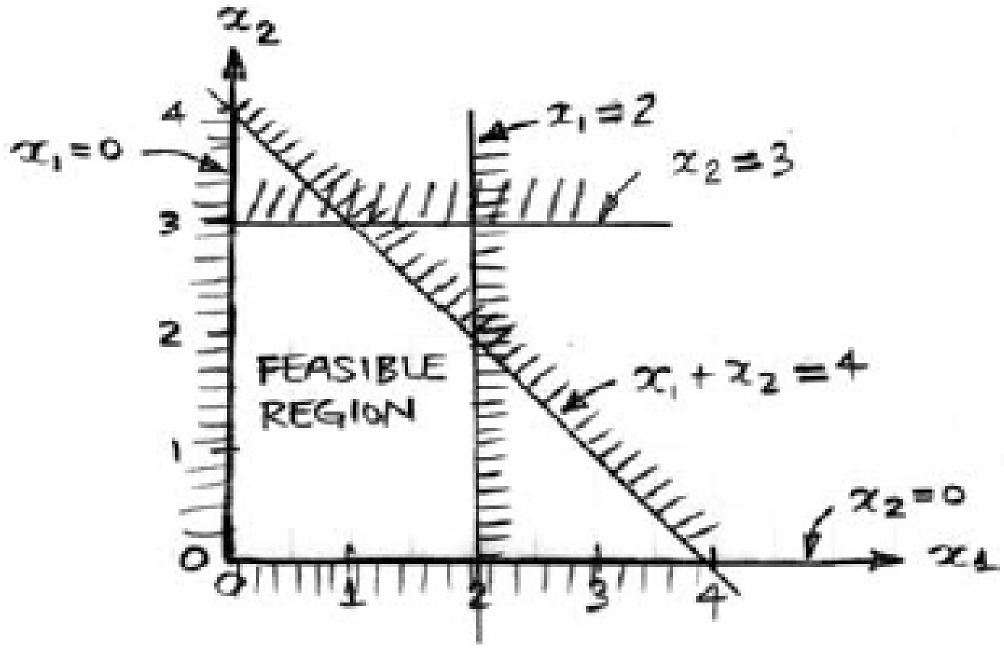
$$\text{maximize } Z = 15x_1 + 10x_2$$ ← Objective function

$$x_1 \le 2$$
$$x_2 \le 3$$
$$x_1 + x_2 \le 4$$

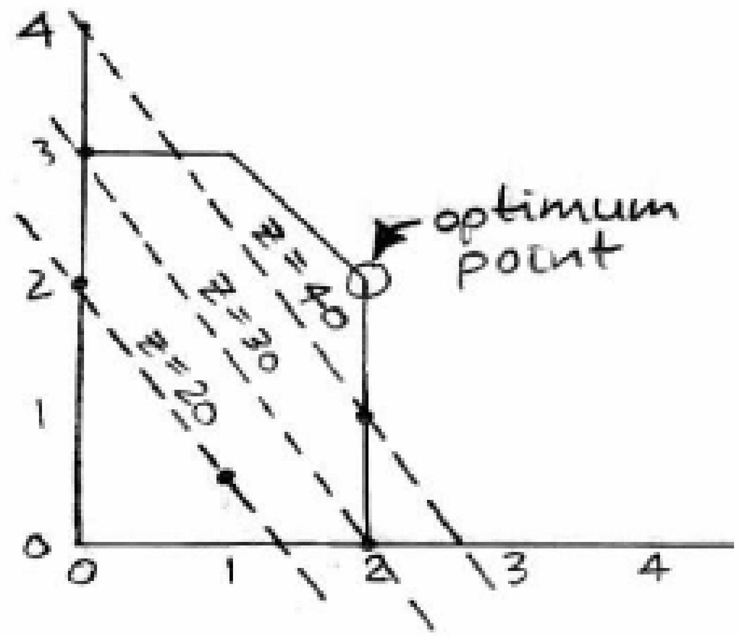
} Constraints

$$x_1 \ge 0,\ x_2 \ge 0$$ ← Bounds

# An Example (2)




**The feasible region for the Acme Bicycle Company problem.**




Best value of the objective.

# Some Techniques

- Sometimes we think of a problem, just don't know how to express it in LP
- Such as
  - If-Else
  - AND
  - OR
  - Finding Range
  - Connecting multiple variables
  - Finding the value of a variable

  - List goes on….

# Connecting two variables (1)

- If both x and y are true, then z is true

$$z \geq x + y - 1$$

- What if x and y varies?
    - For example, x and y has indices

- Still z will take the value 1 if for any index, both x and y are 1

# If

- We are using binary variables here
- If two variables share the same value

$$x = y$$

  - If y = 0, then x = 0
  - If y = 1, then x = 1

- If they may have different values

$$x \geq y$$

  - If y = 1, then x = 1
  - Otherwise x can take either 1 or 0

# OR

- Big M Constraint
- “Binarization” of variables
  - We can create a ‘flag’ that may work as a set of variables

- If any of the y is true, x is true

$$x \geq \frac{\sum_i y_i}{M}$$

  - Here, M is a large value, usually with at least one magnitude higher

# AND

- z = x AND y

- If both x and y is true, z is true

$$z \leq x;$$
$$z \leq y;$$
$$z \geq x + y - 1;$$

  - If either x or y is 0, then z  must be 0
  - The last equation enforces that if both of them are 1, z must be 1

# Ranges

- x and y both are integers, z is binary
- We want to find out if x falls within a range defined by y

- If x >= y, z is true

$$z \geq \frac{x - y + 1}{M};$$

- If x <= y, z is true

$$z \geq \frac{y - x + 1}{M};$$

If the value is 0 or negative, z is 0, else z is 1

# Finding a value

- Just combine the two ranges
- A, B, C are binary variables

- If x = y, $C_y$ is true

$$A \geq \frac{x - y + 1}{M};$$
$$B \geq \frac{y - x + 1}{M};$$
$$C_y = A \wedge B;$$

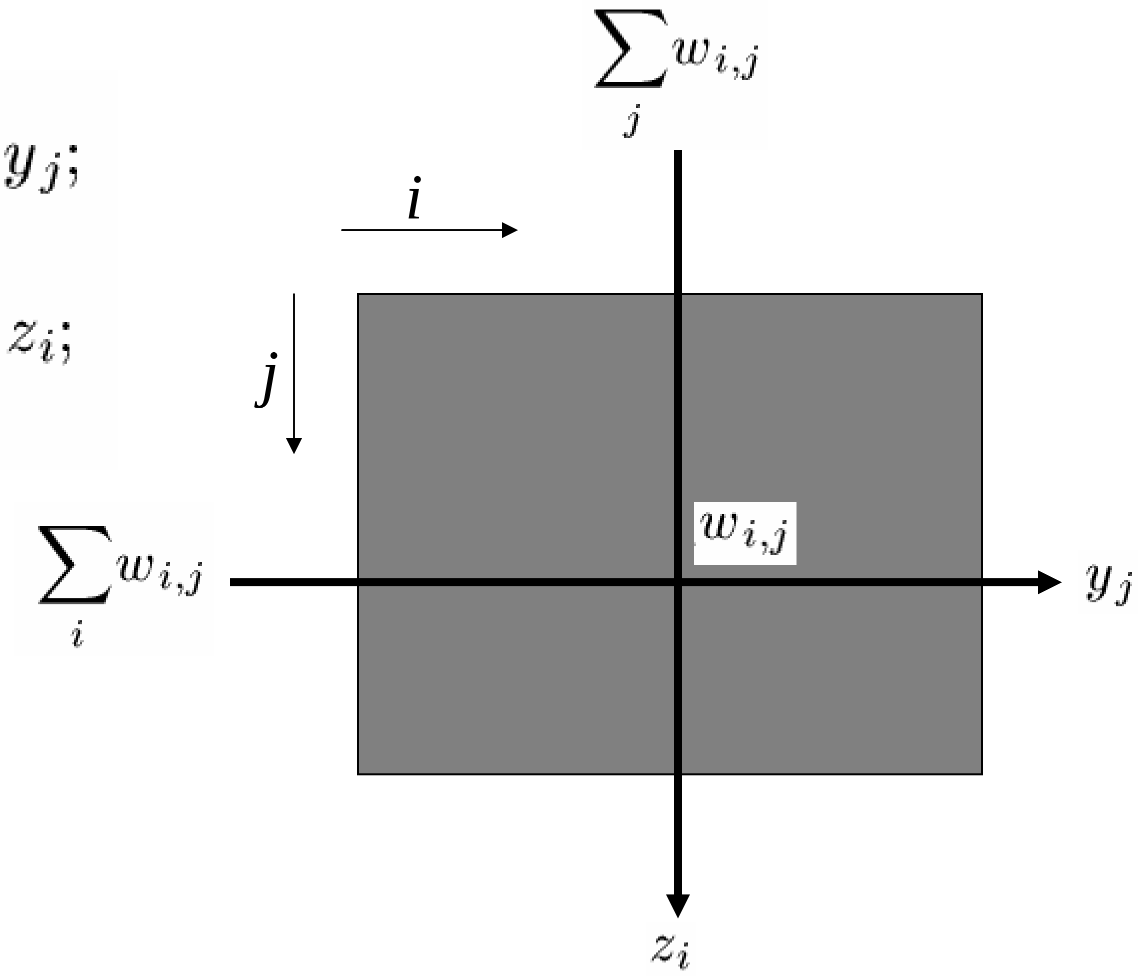
  - x takes the value of y if both the ranges are true

# Connecting two variables (2)

- If x varies with j, y varies with j

$$\sum_i w_{i,j} = y_j;$$

$$\sum_j w_{i,j} = z_i;$$



# If destination is not known…

- u is the node in consideration
- v is a neighbour of u
- $\gamma_i^k$ is the traffic for flow *k* from node *i*

$$\sum_v \lambda_{vu}^{k,i} - \sum_v \lambda_{uv}^{k,i} = \gamma_i^k * w_u^{k,i};$$

← Binary indicator if node *u* is the destination

$$w_u^{k,i} \leq x_u^k;$$

← Binary indicator if node *u* can be a destination for flow *k*

$$\sum_u w_u^{k,i} = 1;$$

← There is only 1 destination