**In the name of God**

# Network Flows

# 7. Multicommodity Flows Problems
# 7.2 Lagrangian Relaxation Approach

**Fall 2010**

*Instructor: Dr. Masoud Yaghini*

# Lagrangian Relaxation

- ***The multicommodity flow problem formulation:***

$$\text{Minimize} \quad \sum_{1 \le k \le K} c^k x^k$$

subject to

$$\sum_{1 \le k \le K} x_{ij}^k \le u_{ij} \qquad \text{for all } (i, j) \in A,$$

$$\mathcal{N} x^k = b^k \qquad \text{for } k = 1, 2, \ldots, K,$$

$$0 \le x_{ij}^k \le u_{ij}^k \qquad \text{for all } (i, j) \in A \text{ and all } k = 1, 2, \ldots, K.$$

# Lagrangian Relaxation

- We associate nonnegative Lagrange multipliers $w_{ij}$ with the bundle constraints, creating the following ***Lagrangian subproblem***:

$$L(w) = \min \sum_{1 \le k \le K} c^k x^k + \sum_{(i,j) \in A} w_{ij} \Big( \sum_{1 \le k \le K} x_{ij}^k - u_{ij} \Big)$$

- or, equivalently,

$$L(w) = \min \sum_{1 \le k \le K} \sum_{(i,j) \in A} (c_{ij}^k + w_{ij}) x_{ij}^k - \sum_{(i,j) \in A} w_{ij} u_{ij}$$

- subject to

$$\mathcal{N} x^k = b^k \qquad \text{for all } k = 1, \ldots, K,$$

$$x_{ij}^k \ge 0 \qquad \text{for all } (i, j) \in A \text{ and all } k = 1, 2, \ldots, K.$$

# Lagrangian Relaxation

- Since the term

$$-\sum_{(i,j)\in A} w_{ij}u_{ij}$$

- is a constant for any given choice of the Lagrange multipliers therefore we can ignore it.
- The resulting objective function has a cost

$$c_{ij}^{k} + w_{ij}$$

- associated with every flow variable $x_{ij}^{k}$

# Lagrangian Relaxation

- Since none of the constraints in this problem contains the flow variables for more than one of the commodities,
  - the problem decomposes into separate **minimum cost flow problems**, one for each commodity.
- Consequently, to apply the subgradient optimization procedure to this problem, we would alternately
  - (1) solve a set of minimum cost flow problems (for a fixed value of the Lagrange multipliers $\boldsymbol{w}$) with the cost coefficients $c_{ij}^{k} + w_{ij}$
  - (2) update the multipliers by the algorithmic procedures

# Lagrangian Relaxation

- If $y_{ij}^k$ denotes the optimal solution to the minimum cost flow subproblems when the Lagrange multipliers have the value $w_{ij}^q$ at the $q$th iteration, the subgradient update formula becomes:

$$w_{ij}^{q+1} = [w_{ij}^q + \theta_q (\sum_{1 \le k \le K} y_{ij}^k - u_{ij})]^+$$

- In this expression, the notation $[\alpha]^+$ denotes the positive part of $\alpha$, that is, $\max(\alpha, 0)$.
- The scalar $\theta_q$ is a step size specifying how far we move from the current solution $w_{ij}^q$

# Lagrangian Relaxation

- If the subproblem solutions $y_{ij}^k$ use more than the available capacity $u_{ij}$ of that arc, the update formula increases the multiplier $w_{ij}^q$ on arc $(i, j)$ by the amount:

$$\left(\sum_{1 \le k \le K} y_{ij}^k - u_{ij}\right)$$

- If the subproblem solutions $y_{ij}^k$ use less than the available capacity of that arc, the update formula reduces the Lagrange multiplier of arc $(i, j)$ by the amount:

$$\left(u_{ij} - \sum_{1 \le k \le K} y_{ij}^k\right)$$

# Lagrangian Relaxation

- The decrease would cause the multiplier $w_{ij}^{q+1}$ to become negative, we reduce the multiplier to value zero.
- We choose the step sizes $\theta_q$ for iterations $q = 1, 2, \ldots,$ in accordance with:

$$\theta_k = \frac{\lambda_k[\mathrm{UB} - L(\mu^k)]}{\| \mathcal{A}x^k - b \|^2}.$$

# Lagrangian Relaxation

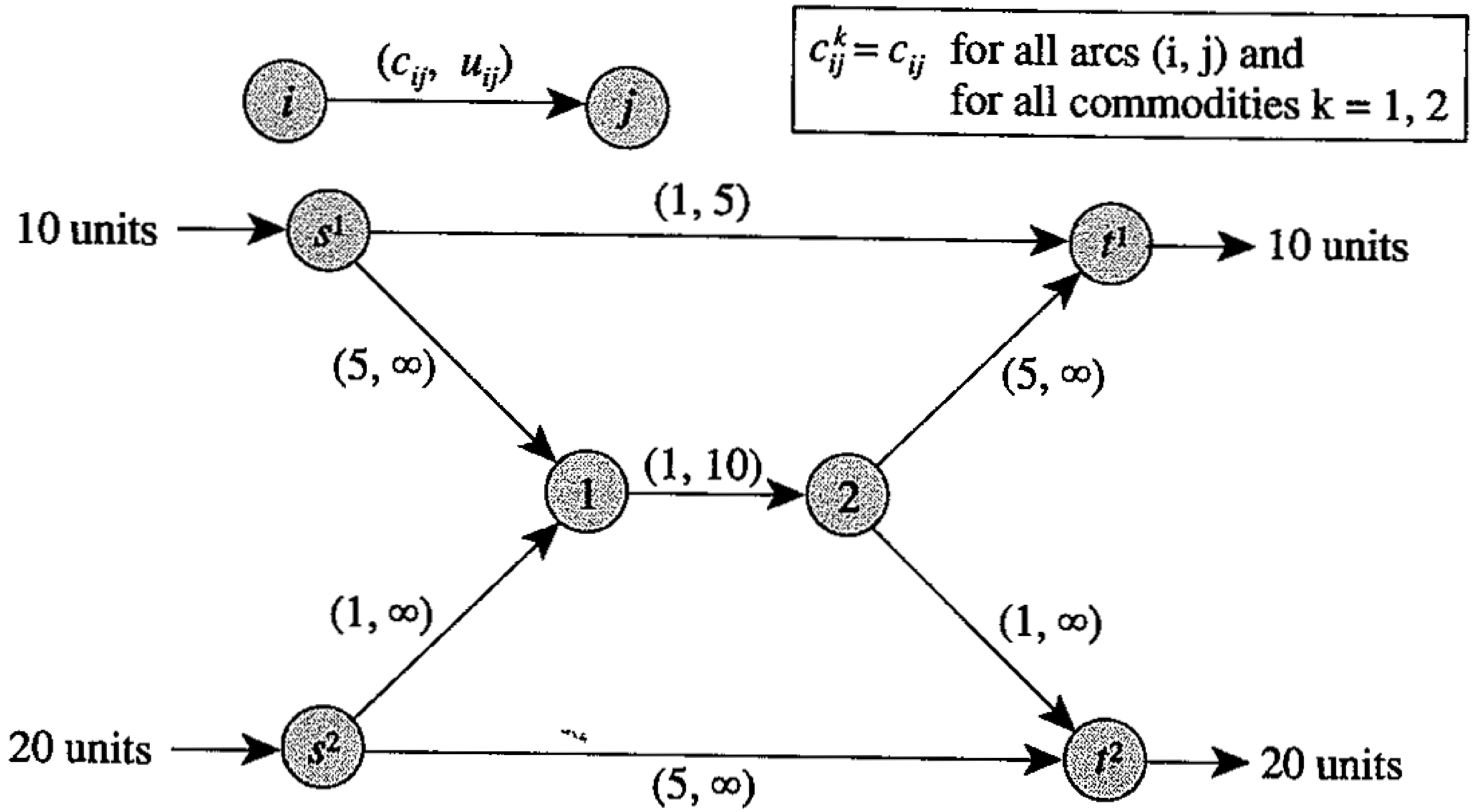
- Whenever we apply Lagrangian relaxation to any linear program, such as the multicommodity flow problem, the optimal value

$$L^* = \max_{w \geq 0} L(w)$$

- of the Lagrangian multiplier problem equals the optimal objective function value $z^*$ of the linear program.

# Lagrangian Relaxation

- Example:
  - Consider a two-commodity problem



# Lagrangian Relaxation

- For any choice of the Lagrange multipliers $\boldsymbol{w}_{12}$ and $w_{s^1t^1}$ for the two capacitated arcs ($s^l$, $t^l$) and (1, 2), the problem decomposes into two shortest path problems.
- If we start with the Lagrange multipliers

$$w^0_{12} = w^0_{s^1t^1} = 0$$

- then in the subproblem solutions, the shortest path:
  - the shortest path $\mathbf{s^1\text{-}t^l}$ carries 10 units of flow at a cost of 1(10) = 10 and,
  - the shortest path $\mathbf{s^2\text{-}1\text{-}2\text{-}t^2}$ carries 20 units of flow at a cost of 3(20) = 60
  - Therefore, ***L*(0) = 10 + 60 = 70** is a lower bound on the optimal objective function value for the problem.

# Lagrangian Relaxation

- Since

$$y^1_{s^1 t^1} + y^2_{s^1 t^1} - u_{s^1 t^1} = 5$$

$$y^1_{12} + y^2_{12} - u_{12} = 10$$

- The update formulas become:

$$w^{q+1}_{ij} = [w^q_{ij} + \theta_q ( \sum_{1 \le k \le K} y^k_{ij} - u_{ij})]^+$$

$$w^1_{s^1 t^1} = [0 + \theta_0 \cdot 5]^+$$

$$w^1_{12} = [0 + \theta_0 \cdot 10]^+$$

- If we choose $\theta_0 = 1$, then

$w^1_{s^1 t^1} = 5$ and $w^1_{12} = 10$.

# Lagrangian Relaxation

- The new shortest path solutions send:
  - 10 units on the path $s^1$-$t^1$ at a cost of (1 + 5) (10) = 60
  - 20 units on the path $s^2$-$t^2$ at a cost of 5(20) = 100
- The new lower bound obtained through:

$$L(w) = \min \sum_{1 \le k \le K} \sum_{(i,j) \in A} (c_{ij}^k + w_{ij}) x_{ij}^k - \sum_{(i,j) \in A} w_{ij} u_{ij}$$

$$60 + 100 - w_{12}^1 u_{12} - w_{s^1 t^1}^1 u_{s^1 t^1} =$$

$$160 - 10(10) - 5(5) = 35$$

- The value of the lower bound has decreased.

# Lagrangian Relaxation

- At this point:

$$y^1_{s^1t^1} + y^2_{s^1t^1} - u_{s^1t^1} = 5$$

$$y^1_{12} + y^2_{12} - u_{12} = \text{-10}$$

- so the update formulas become

$$w^{q+1}_{ij} = [w^q_{ij} + \theta_q ( \sum_{1 \le k \le K} y^k_{ij} - u_{ij})]^+$$

$$w^1_{s^1t^1} = [5 + \theta_0 \cdot 5]^+$$

$$w^1_{12} = [10 - \theta_0 \cdot 10]^+$$

- If we choose $\boldsymbol{\theta_0 = 1}$,

$$w^1_{s^1t^1} = 10$$

$$w^1_{12} = 0$$

# Lagrangian Relaxation

- The new shortest path solutions send:
  - 10 units on the path $\boldsymbol{s^1}$-$\boldsymbol{t^1}$ at a cost of (1 + 10) (10) = 110 and
  - 20 units on the path $\boldsymbol{s^2}$**-1-2-**$\boldsymbol{t^2}$ at a cost of 3(20) = 60
- If we continue by choosing the step sizes for the *k*th iteration as $\theta_k = 1/k$, we obtain the set of iterates:

# Lagrangian Relaxation

| Iteration number $q$ | $w_{12}^q$ | $w_{s^1t^1}^q$ | Shortest paths | Shortest path costs | $10w_{12}^q + 5w_{s^1t^1}^q$ | Lower bound $L(w^q)$ | $\theta_q$ |
|---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | $s^1$–$t^1$<br>$s^2$–1–2–$t^2$ | 10(1)<br>20(1 + 1 + 1) | 0 | 70 | 1 |
| 1 | 10 | 5 | $s^1$–$t^1$<br>$s^2$–$t^2$ | 10(1 + 5)<br>20(5) | 125 | 35 | 1 |
| 2 | 0 | 10 | $s^1$–$t^1$<br>$s^2$–1–2–$t^2$ | 10(1 + 10)<br>20(1 + 1 + 1) | 50 | 120 | 0.5 |
| 3 | 5 | 12.5 | $s^1$–$t^1$<br>$s^2$–$t^2$ | 10(1 + 12.5)<br>20(5) | 112.5 | 122.5 | 0.333 |
| 4 | 1.67 | 14.17 | $s^1$–1–2–$t^1$<br>$s^2$–1–2–$t^2$ | 10(5 + 2.67 + 5)<br>20(1 + 2.67 + 1) | 87.55 | 132.5 | 0.25 |
| 5 | 6.67 | 12.92 | $s^1$–$t^1$<br>$s^2$–$t^2$ | 10(1 + 12.92)<br>20(5) | 131.3 | 107.9 | 0.2 |
| 6 | 4.67 | 13.92 | $s^1$–$t^1$<br>$s^2$–$t^2$ | 10(1 + 13.92)<br>20(5) | 116.3 | 132.9 | 0.167 |
| 7 | 3 | 14.75 | $s^1$–1–2–$t^1$<br>$s^2$–$t^2$ | 10(5 + 4 + 5)<br>20(5) | 103.8 | 136.3 | 0.143 |

# Lagrangian Relaxation

| Iteration number $q$ | $w^q_{12}$ | $w^q_{s^1t^1}$ | Shortest paths | Shortest path costs | $10w^q_{12} + 5w^q_{s^1t^1}$ | Lower bound $L(w^q)$ | $\theta_q$ |
|---|---|---|---|---|---|---|---|
| 8 | 3 | 14.04 | $s^1$–1–2–$t^1$<br>$s^2$–$t^2$ | 10(5 + 4 + 5)<br>20(5) | 100.2 | 139.8 | 0.125 |
| 9 | 3 | 13.41 | $s^1$–1–2–$t^1$<br>$s^2$–$t^2$ | 10(5 + 4 + 5)<br>20(5) | 97.05 | 143.0 | 0.111 |
| 10 | 3 | 12.86 | $s^1$–$t^1$<br>$s^2$–$t^2$ | 10(1 + 12.86)<br>20(5) | 94.3 | 144.3 | 0.1 |
| 11 | 2 | 13.36 | $s^1$–1–2–$t^1$<br>$s^2$–$t^2$ | 10(5 + 3 + 5)<br>20(5) | 86.8 | 143.2 | 0.091 |
| 12 | 2 | 12.9 | $s^1$–1–2–$t^1$<br>$s^2$–$t^2$ | 10(5 + 3 + 5)<br>20(5) | 84.5 | 145.5 | 0.083 |
| 13 | 2 | 12.48 | $s^1$–1–2–$t^1$<br>$s^2$–$t^2$ | 10(5 + 3 + 5)<br>20(5) | 82.2 | 147.6 | 0.077 |
| 14 | 2 | 12.09 | $s^1$–1–2–$t^1$<br>$s^2$–$t^2$ | 10(5 + 3 + 5)<br>20(5) | 80.25 | 149.5 | 0.071 |

# Lagrangian Relaxation

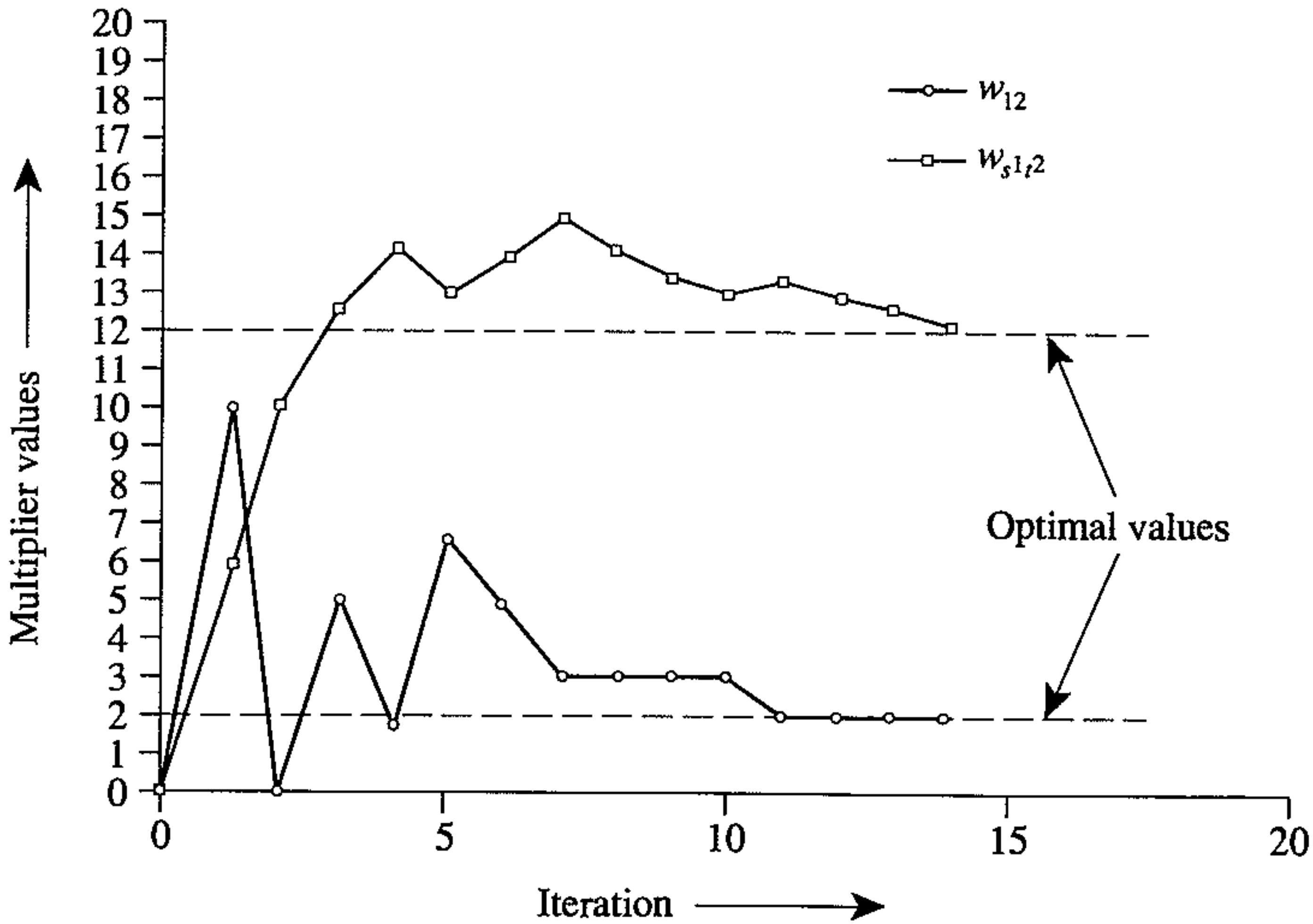
- From iteration 14 on, the values of the Lagrange multipliers oscillate about, and converge to, their optimal values:

$$w_{12} = 2$$
$$w_{s^1 t^1} = 12$$

- The optimal lower bound oscillates about its optimal value 150, which equals the optimal objective function value of the multicommodity flow problem.

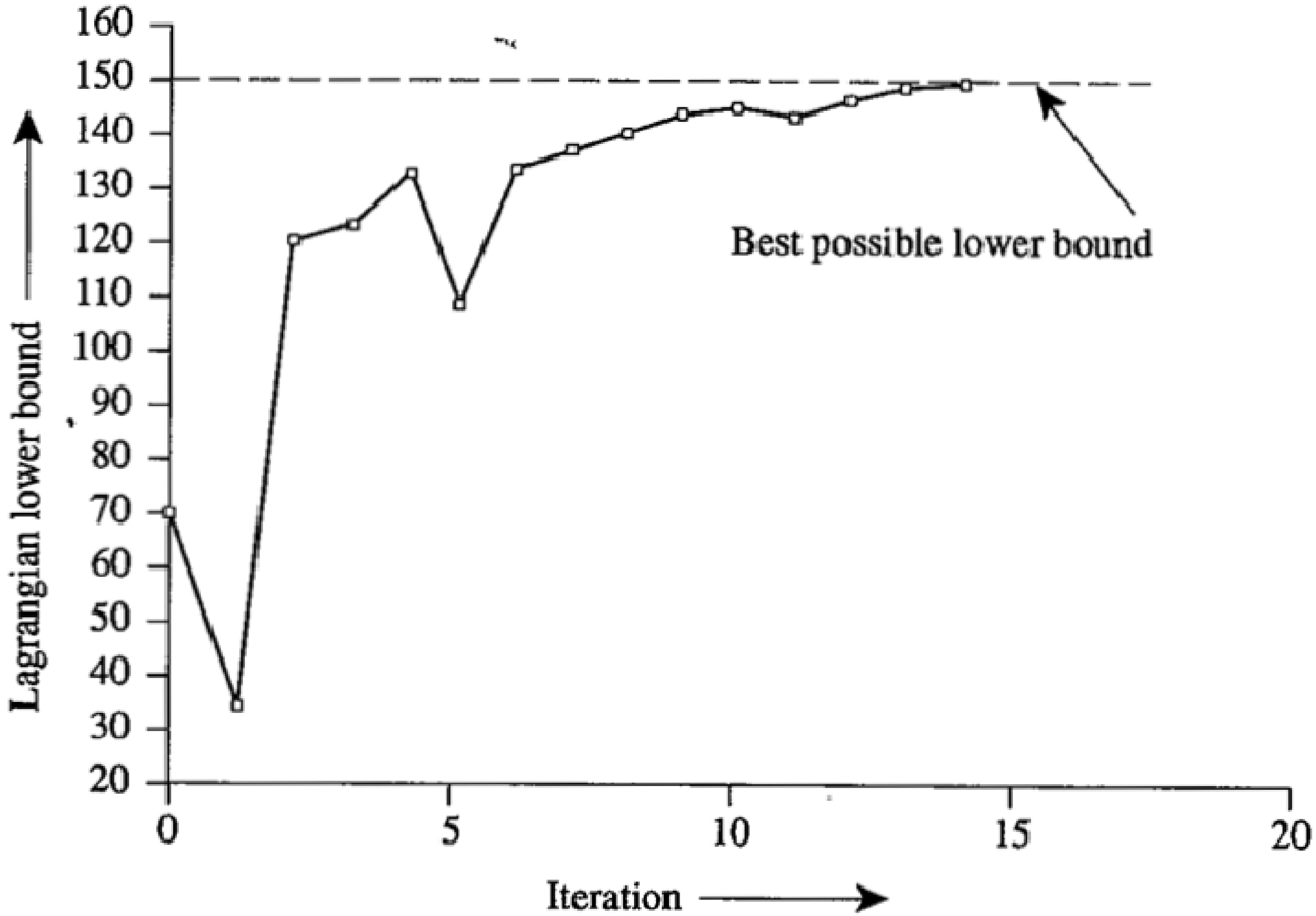
# Lagrangian Relaxation

- This figure shows how the values of the Lagrange multipliers vary during the execution of the algorithm



# Lagrangian Relaxation

- This figure shows how the values of Lagrangian lower bounds vary during the execution of the algorithm



# Lagrangian Relaxation

- Advantages of subgradient optimization for solving the Lagrangian multiplier problem:
  - (1) This solution approach permits us to exploit the underlying network flow structure.
  - (2) The formulas for updating the Lagrange multipliers $w_{ij}$ are rather small computationally and very easy to encode in a computer program.

# Lagrangian Relaxation

- Limitations of subgradient optimization:
  - (1) To ensure convergence we need to take small step sizes; as a result, the method does not converge very fast.
  - (2) the optimal flows $y_{ij}^k$ solve the Lagrangian subproblem, these subproblems might also have other optimal solutions that do not satisfy the bundle constramts.
    - For example for the problem we have just solved, with the optimal Lagrange multipliers $w_{12} = 2$ and $w_{s^1 t^1} = 12$, the shortest paths subproblems have solutions with 10 units on the path $s^1$-$t^1$ and 20 units on the path $s^2$-$t^2$ .
    - This solution violates the capacity of the arc $(s^1, t^1)$.
    - In general, to obtain optimal flows, even after we have solved the Lagrangian multiplier problem, requires additional work.

The End