In the name of God

Part 4. Decomposition Algorithms

4.3. The Decomposition Algorithm for the Multicommodity Flow Problems

Spring 2010

Instructor: Dr. Masoud Yaghini

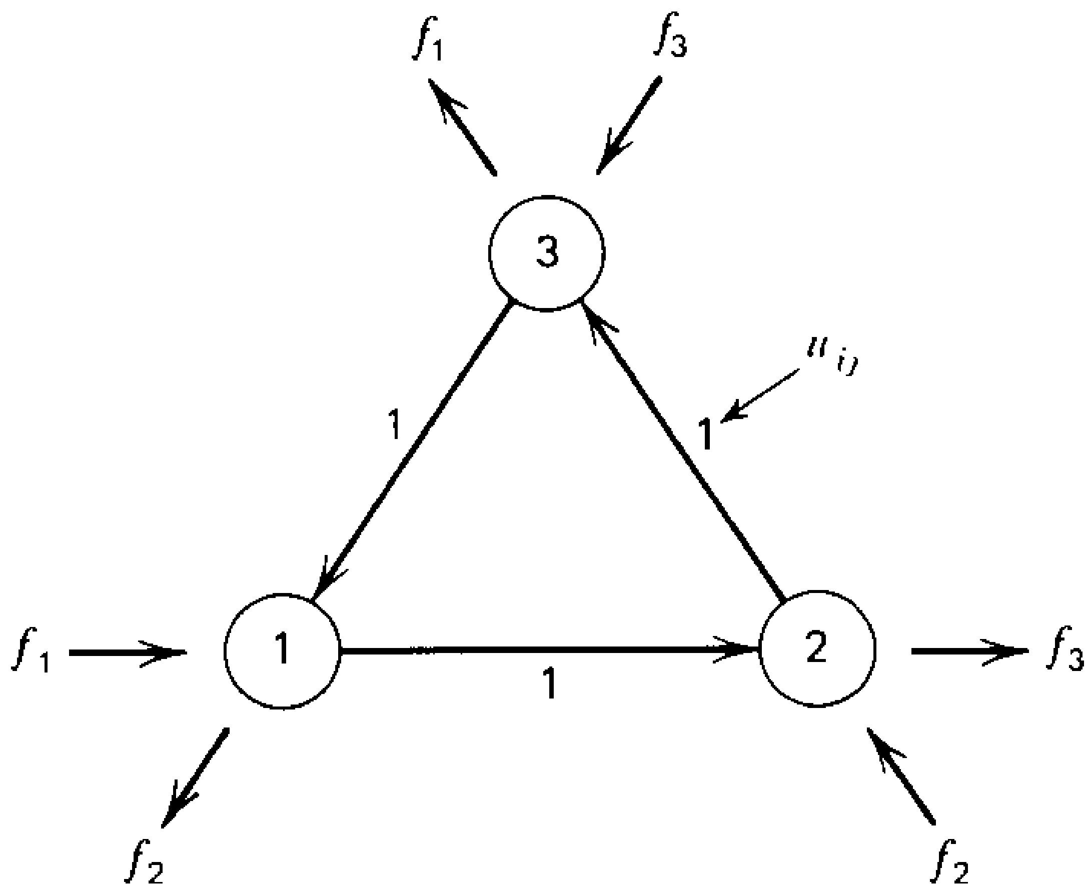
Multicommodity Flow Problem

Introduction

- **Single-commodity flow problems**
  - It is not necessary to distinguish among the units flowing in the network.
- **Multicommodity flow problems**
  - It is necessary to distinguish among the flows in the network.

Example

- An example of multicommodity flow problem:



Example

- **Example:**
  - Suppose that there are three commodities that flow through the network.
  - The source for commodity 1 is node 1, and the sink for commodity 1 is node 3. That is, commodity 1 must originate only at node 1 and terminate only at node 3.
  - Let the source and sink for commodity 2 be nodes 2 and 1 respectively.
  - Finally, the source and sink for commodity 3 are nodes 3 and 2 respectively.
  - With the restriction that the sum of all commodities flowing on an arc should not exceed the arc capacity $u_{ij} = 1$,
  - what is the maximal sum of commodity flows, $f_1 + f_2 + f_3$ possible in the network?

Example

- Finding the **maximal flow** for the three-commodity problem is relatively simple since there is only one path that each commodity can take on its way from its source to its sink.
- The paths for commodity 1, 2, and 3 respectively are

$$P_1 = \{(1, 2), (2, 3)\}$$

$$P_2 = \{(2, 3), (3, 1)\}$$

$$P_3 = \{(3, 1), (1, 2)\}$$

Example

- If we place a single unit of flow on any one of the paths, then the other paths are completely blocked (that is, must have zero flow) and thus the total flow would be 1.
- However, there is a better solution available if we do not require integer flows.
- Suppose that we place 1/2 unit of flow of commodity 1 on P_1, 1/2 unit of flow of commodity 2 on P_2, and 1/2 unit of flow of commodity 3 on P_3
- In this case none of the arc capacities are violated and the total flow of all commodities is 3/2.

The Multicommodity Minimal Cost Flow Problem

The Multicommodity Minimal Cost Flow Problem

- Suppose:
  - G : a network with m nodes and n arcs in which there will flow t different commodities.
  - $\mathbf{u_i}$: the vector of upper limits on flow for commodity i in the arcs of the network.
  - u_{ipq} : the upper limit on flow of commodity i in arc (p, q).
  - $\mathbf{u}$: the vector of upper limits on the sum of all commodities flowing in the arcs of the network.
  - u_{pq} : the upper limit on the sum of all commodity flows in arc (p, q).
  - $\mathbf{c_i}$: the vector of arc costs in the network for commodity i.

The Multicommodity Minimal Cost Flow Problem

- Suppose (cont.)
  - c_{ipq} : the unit cost of commodity i on arc (p, q).
  - $\mathbf{b_i}$: the vector of supplies (or demands) of commodity i in the network.
  - b_{iq} is the supply (if $b_{iq} > 0$) or demand (if $b_{iq} < 0$) of commodity i at node q.
  - $\mathbf{x_i}$: the vector of flows of commodity i in the network
  - $\mathbf{A}$: the node-arc incidence matrix of the graph

The Multicommodity Minimal Cost Flow Problem

- The linear programming formulation for the **multicommodity minimal cost flow problem** is as follows:

$$\text{Minimize} \quad \sum_{i=1}^{t} \mathbf{c}_i \mathbf{x}_i$$

$$\text{Subject to} \quad \sum_{i=1}^{t} \mathbf{x}_i \leq \mathbf{u}$$

$$\mathbf{A}\mathbf{x}_i = \mathbf{b}_i \qquad i = 1, \ldots, t$$

$$\mathbf{0} \leq \mathbf{x}_i \leq \mathbf{u}_i \qquad i = 1, \ldots, t$$

The Multicommodity Minimal Cost Flow Problem

- This formulation is called the **node-arc formulation** since it uses the node-arc incidence matrix.
- The multicommodity minimal cost flow problem has
  - $(t + 1)n$ variables (including the slack variables for the coupling constraints and ignoring the nonnegativity)
  - $n + m \cdot t$ (including constraints and upper bound constraints $\mathbf{0} < \mathbf{x}_\mathbf{i} < \mathbf{u}_\mathbf{i}$).
- Thus, even for moderate-sized problems, the constraint matrix will be large.
  - For example, suppose that we have a problem with 100 nodes, 250 arcs, and 10 commodities. The problem will have 2750 variables and 1250 constraints.

The Decomposition Algorithm to the Multicommodity Minimal Cost Flow Problem

Decomposition Algorithm

- The multicommodity minimal cost flow problem possesses the **block diagonal structure**.
- Thus we may apply the block diagonal decomposition technique to the foregoing problem.

Decomposition Algorithm

- Let $X_i = \{\mathbf{x_i} : \mathbf{Ax_i} = \mathbf{b_i}, \mathbf{0} \leq \mathbf{x_i} \leq \mathbf{u_i}\}$.
- Assume that each component of $\mathbf{u_i}$ is finite so that X_i is bounded
- Then any $\mathbf{x_i}$ can be expressed as a convex combination of the extreme points of X_i as follows:

$$\mathbf{x}_i = \sum_{j=1}^{k_i} \lambda_{ij} \mathbf{x}_{ij}$$

- where

$$\sum_{j=1}^{k_i} \lambda_{ij} = 1$$

$$\lambda_{ij} \geqslant 0 \qquad j = 1, \ldots, k_i$$

Decomposition Algorithm

- and $\mathbf{x}_{i1}, \mathbf{x}_{i2}, \ldots, \mathbf{x}_{ik}$ are the extreme points of X_i
- Substituting for $\mathbf{x}_i$ in the problem and denoting the vector of slacks by $\mathbf{s}$, we get the following.

$$\text{Minimize} \quad \sum_{i=1}^{t} \sum_{j=1}^{k_i} (\mathbf{c}_i \mathbf{x}_{ij}) \lambda_{ij}$$

$$\text{Subject to} \quad \sum_{i=1}^{t} \sum_{j=1}^{k_i} \mathbf{x}_{ij} \lambda_{ij} + \mathbf{s} = \mathbf{u}$$

$$\sum_{j=1}^{k_i} \lambda_{ij} = 1 \qquad i = 1, \ldots, t$$

$$\lambda_{ij} \geqslant 0 \qquad j = 1, \ldots, k_i, \quad i = 1, \ldots, t$$

$$\mathbf{s} \geqslant \mathbf{0}$$

Decomposition Algorithm

- Suppose that we have a basic feasible solution to the multicommodity minimal cost flow problem in terms of the λ_{ij}
- Let $(\mathbf{w}, \alpha)$ be the vector of dual variables corresponding to the basic feasible solution
  - $\mathbf{w}$ has n components and α has t components
- Then dual feasibility is given by the following two conditions:
  - (i) $w_{pq} \leq 0$ corresponding to each s_{pq} , and
  - (ii) $\mathbf{w}\mathbf{x}_{\mathbf{ij}} + \alpha_{\mathbf{i}} - \mathbf{c}_{\mathbf{i}}\mathbf{x}_{\mathbf{ij}} \leq 0$ corresponding to each λ_{ij}

Decomposition Algorithm

- If any of these conditions is violated, the corresponding variable (s_{pq} or λ_{ij}) is a candidate to enter the master basis.
- Here s_{pq} is a candidate to enter the basis if $w_{pq} > 0$.
- For a given commodity i, a nonbasic variable among the λ_{ij} could enter the basis if the optimal objective of the following subproblem is positive.

$$\text{Maximize} \quad (\mathbf{w} - \mathbf{c}_i)\mathbf{x}_i + \alpha_i$$

$$\text{Subject to} \qquad \mathbf{A}\mathbf{x}_i = \mathbf{b}_i$$

$$\mathbf{0} \leqslant \mathbf{x}_i \leqslant \mathbf{u}_i$$

Decomposition Algorithm

- Since **A** is a node-arc incidence matrix, this is simply a **single-commodity flow problem**.
- Thus it may be solved by one of the efficient techniques for solving single-commodity network flow problems (network simplex or out-of- kilter).

Decomposition Algorithm

- We now summarize the decomposition algorithm to the multicommodity minimal cost flow problem.

INITIALIZATION STEP

- **INITIALIZATION STEP**
  - Begin with a basic feasible solution to the master problem.
  - Store $\mathbf{B}^{-1}$, $\bar{\mathbf{b}} = \mathbf{B}^{-1}\begin{pmatrix}\mathbf{u}\\ \mathbf{1}\end{pmatrix}$ and $(\mathbf{w}, \alpha) = \hat{\mathbf{c}}_{\mathbf{B}}\mathbf{B}^{-1}$,
  - Where $\hat{c}_{ij} = \mathbf{c_i x_{ij}}$

MAIN STEP

- **Step 1:**
  - Let $(\mathbf{w}, \alpha)$ be the vector of dual variables corresponding to the current basic feasible solution to the master problem.
  - If any $w_{pq} > 0$, then the corresponding s_{pq} is a candidate to enter the master basis.
  - If $w_{pq} \leq 0$ for each arc, consider the following *i*th subproblem.

$$\begin{aligned} \text{Maximize} \quad & (\mathbf{w} - \mathbf{c}_i)\mathbf{x}_i + \alpha_i \\ \text{Subject to} \quad & \mathbf{A}\mathbf{x}_i = \mathbf{b}_i \\ & \mathbf{0} \leqslant \mathbf{x}_i \leqslant \mathbf{u}_i \end{aligned}$$

MAIN STEP

- This is a single-commodity flow problem.
- If the solution $\mathbf{x_{ik}}$ to this problem has

$$z_{ik} - c_{ik} = (\mathbf{w} - \mathbf{c_i})\ \mathbf{x_{ik}} + \alpha_\mathrm{i} > 0$$

- Then λ_{ik} is a candidate to enter the master basis.
- Go to step 2.

MAIN STEP

- **Step 2:**
  - If there is no candidate to enter the master basis, then stop; the optimal solution is at hand.
  - Otherwise, select a candidate variable, update its column accordingly,

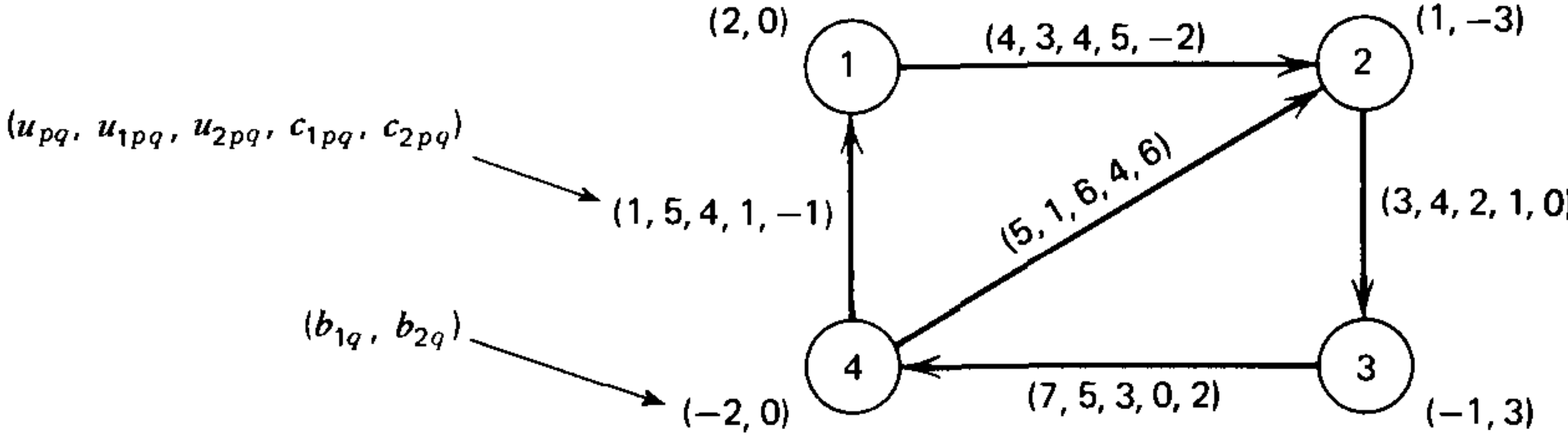
$$\mathbf{B}^{-1}\begin{pmatrix} \mathbf{e}_{pq} \\ \mathbf{0} \end{pmatrix} \text{ for } s_{pq} \text{ and } \mathbf{B}^{-1}\begin{pmatrix} \mathbf{x}_{ik} \\ \mathbf{e}_i \end{pmatrix} \text{ for } \lambda_{ik},$$

  - Note that **e** is a unit vector with the 1 in the row associated with arc (p, q).
  - This updates the basis inverse, the dual variables, and the right-hand side.
  - Return to step 1.

An Example of the Multicommodity Minimal Cost Flow Algorithm

An Example

- Consider the **two-commodity** minimal cost, flow problem whose data are given:



An Example

- The constraint matrix and the right-hand side are:

	FIRST COMMODITY VARIABLES						SECOND COMMODITY VARIABLES						SLACK VARIABLES					
	x_{112}	x_{123}	x_{134}	x_{141}	x_{142}	x_1	x_{212}	x_{223}	x_{234}	x_{241}	x_{242}	x_2	s_{12}	s_{23}	s_{34}	s_{41}	s_{42}	RHS
	−5	−1	0	−1	−4	0	2	0	−2	1	−6	0	0	0	0	0	0	0
Coupling Constraints	1	0	0	0	0	0	1	0	0	0	0	0	1	0	0	0	0	4
	0	1	0	0	0	0	0	1	0	0	0	0	0	1	0	0	0	3
	0	0	1	0	0	0	0	0	1	0	0	0	0	0	1	0	0	7
	0	0	0	1	0	0	0	0	0	1	0	0	0	0	0	1	0	1
	0	0	0	0	1	0	0	0	0	0	1	0	0	0	0	0	1	5
Node-arc incidence matrix for subproblem 1	1	0	0	−1	0	0	0	0	0	0	0	0	0	0	0	0	0	2
	−1	1	0	0	−1	0	0	0	0	0	0	0	0	0	0	0	0	1
	0	−1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	−1
	0	0	−1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	−2
Node-arc incidence matrix for subproblem 2	0	0	0	0	0	0	1	0	0	−1	0	0	0	0	0	0	0	0
	0	0	0	0	0	0	−1	1	0	0	−1	0	0	0	0	0	0	−3
	0	0	0	0	0	0	0	−1	1	0	0	0	0	0	0	0	0	3
	0	0	0	0	0	0	0	0	−1	1	1	1	0	0	0	0	0	0

An Example

- The lower and upper bound constraints $\mathbf{0} \leq \mathbf{x_1} \leq \mathbf{u_1}$, and $\mathbf{0} \leq \mathbf{x_2} < \mathbf{u_2}$ are not displayed.
- Notice the structure of the coupling constraints and the special structured block diagonal constraints.
- Also note that x_1, and x_2 represent the **artificial variables** for the two commodities.

Initialization

- Suppose that we begin with the following feasible solutions.

$$\mathbf{x}_{11} = \begin{bmatrix} x_{112} \\ x_{123} \\ x_{134} \\ x_{141} \\ x_{142} \end{bmatrix} = \begin{bmatrix} 2 \\ 3 \\ 2 \\ 0 \\ 0 \end{bmatrix} \quad \text{and} \quad \mathbf{x}_{21} = \begin{bmatrix} x_{212} \\ x_{223} \\ x_{234} \\ x_{241} \\ x_{242} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 3 \\ 0 \\ 3 \end{bmatrix}$$

- Note that the master basis (in the space of the slack variables and the λ_{ij}'s) consists of all the slacks, λ_{11} and λ_{21}

Initialization

- The basis and its inverse are

$$\mathbf{B} = \begin{array}{c} \begin{matrix} s_{12} & s_{23} & s_{34} & s_{41} & s_{42} & \lambda_{11} & \lambda_{21} \end{matrix} \\ \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 2 & 0 \\ 0 & 1 & 0 & 0 & 0 & 3 & 0 \\ 0 & 0 & 1 & 0 & 0 & 2 & 3 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 3 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \end{array} \quad \mathbf{B}^{-1} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & -2 & 0 \\ 0 & 1 & 0 & 0 & 0 & -3 & 0 \\ 0 & 0 & 1 & 0 & 0 & -2 & -3 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & -3 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

- Here $\mathbf{c}_1\mathbf{x}_{11} = 13$ and $\mathbf{c}_2\mathbf{x}_{21} = 24$.
- Denoting $\begin{bmatrix} \mathbf{u} \\ \mathbf{1} \end{bmatrix}$ by $\hat{\mathbf{b}}$, we have:

$$(\mathbf{w}, \boldsymbol{\alpha}) = \hat{\mathbf{c}}_B\mathbf{B}^{-1} = (0, 0, 0, 0, 0, 13, 24)\mathbf{B}^{-1} = (0, 0, 0, 0, 0, 13, 24)$$

Initialization

$$\mathbf{x}_B = \mathbf{B}^{-1}\hat{\mathbf{b}} = \mathbf{B}^{-1}\begin{bmatrix} \mathbf{u} \\ 1 \\ 1 \end{bmatrix} = \mathbf{B}^{-1}\begin{bmatrix} 4 \\ 3 \\ 7 \\ 1 \\ 5 \\ 1 \\ 1 \end{bmatrix} = \begin{bmatrix} 2 \\ 0 \\ 2 \\ 1 \\ 2 \\ 1 \\ 1 \end{bmatrix}$$

$$z = \hat{\mathbf{c}}_B \mathbf{B}^{-1}\hat{\mathbf{b}} = 37$$

- Setting up the revised simplex array

$(\mathbf{w}, \alpha)$	$\hat{\mathbf{c}}_B \mathbf{B}^{-1}\hat{\mathbf{b}}$
$\mathbf{B}^{-1}$	$\mathbf{B}^{-1}\hat{\mathbf{b}}$

Initialization

- for the master problem, we get the following:

	w_{12}	w_{23}	w_{34}	w_{41}	w_{42}	α_1	α_2	RHS
z	0	0	0	0	0	13	24	37
s_{12}	1	0	0	0	0	−2	0	2
s_{23}	0	1	0	0	0	−3	0	0
s_{34}	0	0	1	0	0	−2	−3	2
s_{41}	0	0	0	1	0	0	0	1
s_{42}	0	0	0	0	1	0	−3	2
λ_{11}	0	0	0	0	0	1	0	1
λ_{12}	0	0	0	0	0	0	1	1

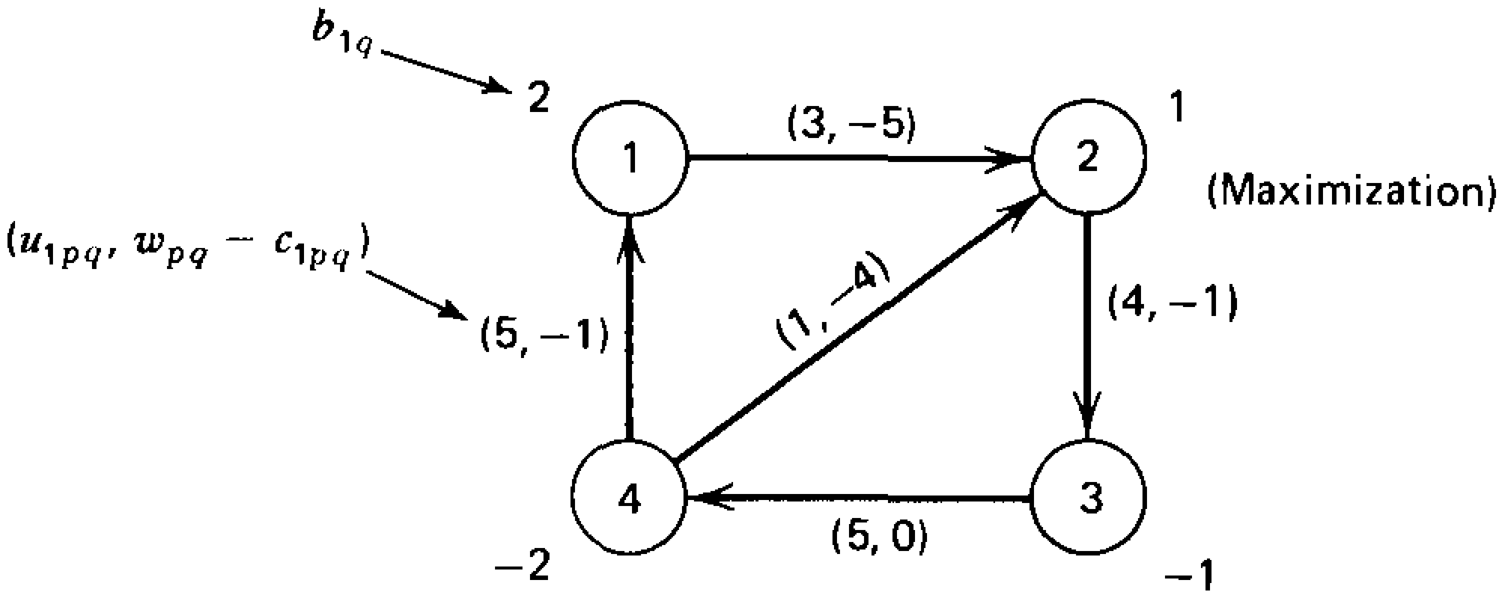
Iteration 1

- First, all $w_{pq} \leq 0$.
- Next we check whether a candidate from either subproblem (or commodity) is eligible to enter the master basis.

Iteration 1: SUBPROBLEM 1

$$\mathbf{w} - \mathbf{c}_1 = \mathbf{0} - \mathbf{c}_1 = (-5, -1, 0, -1, -4)$$

- Subproblem 1 is the single-commodity flow problem defined in:



Iteration 1: SUBPROBLEM 1

- The optimal (maximal cost) solution is

  $\mathbf{x}_{12} = (2, 3, 2, 0, 0)^t$

- and the value of the subproblem 1 objective is:

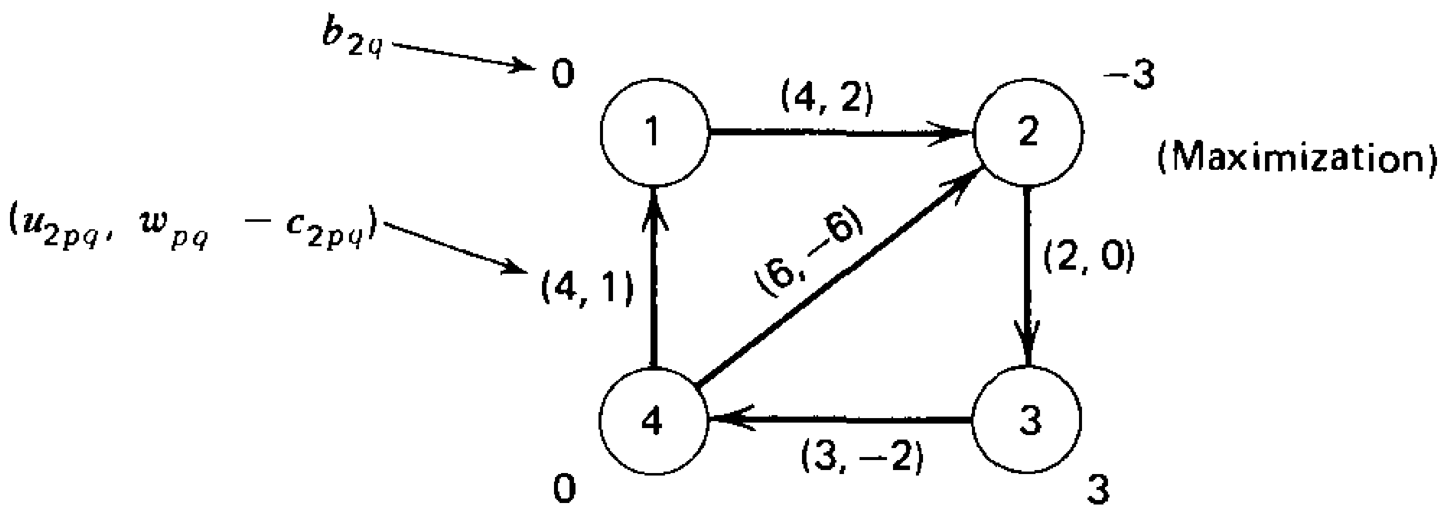
$$z_{12} - c_{12} = (\mathbf{w} - \mathbf{c}_1)\mathbf{x}_{12} + \alpha_1 = -13 + 13 = 0$$

- Thus there is no candidate from subproblem 1.

Iteration 1: SUBPROBLEM 2

$$\mathbf{w} - \mathbf{c}_2 = \mathbf{0} - \mathbf{c}_2 = (2, 0, -2, 1, -6)$$

- Subproblem 2 is the single-commodity flow problem defined in:



-

Iteration 1: SUBPROBLEM 2

- The optimal (maximal cost) solution is

  $\mathbf{x}_{22} = (3, 0, 3, 3, 0)^t$

- and the value of the subproblem 2 objective is:

$$z_{22} - c_{22} = (\mathbf{w} - \mathbf{c}_2)\mathbf{x}_{22} + \alpha_2 = 3 + 24 = 27$$

- Thus λ_{22} is a candidate to enter the basis.
- The updated column for λ_{22} (exclusive of $z_{22} - c_{22}$) is:

$$\mathbf{B}^{-1}\begin{bmatrix} \mathbf{x}_{22} \\ 0 \\ 1 \end{bmatrix} = (3, 0, 0, 3, -3, 0, 1)^t$$

Iteration 1: SUBPROBLEM 2

- The pivoting process is as follows:

	w_{12}	w_{23}	w_{34}	w_{41}	w_{42}	α_1	α_2	RHS	λ_{22}
z	0	0	0	0	0	13	24	37	27
s_{12}	1	0	0	0	0	−2	0	2	3
s_{23}	0	1	0	0	0	−3	0	0	0
s_{34}	0	0	1	0	0	−2	−3	2	0
s_{41}	0	0	0	1	0	0	0	1	(3)
s_{42}	0	0	0	0	1	0	−3	2	−3
λ_{11}	0	0	0	0	0	1	0	1	0
λ_{21}	0	0	0	0	0	0	1	1	1

Iteration 1: SUBPROBLEM 2

	w_{12}	w_{23}	w_{34}	w_{41}	w_{42}	α_1	α_2	RHS
z	0	0	0	-9	0	13	24	28
s_{12}	1	0	0	-1	0	-2	0	1
s_{23}	0	1	0	0	0	-3	0	0
s_{34}	0	0	1	0	0	-2	-3	2
λ_{22}	0	0	0	$\frac{1}{3}$	0	0	0	$\frac{1}{3}$
s_{42}	0	0	0	1	1	0	-3	3
λ_{11}	0	0	0	0	0	1	0	1
λ_{21}	0	0	0	$-\frac{1}{3}$	0	0	1	$\frac{2}{3}$

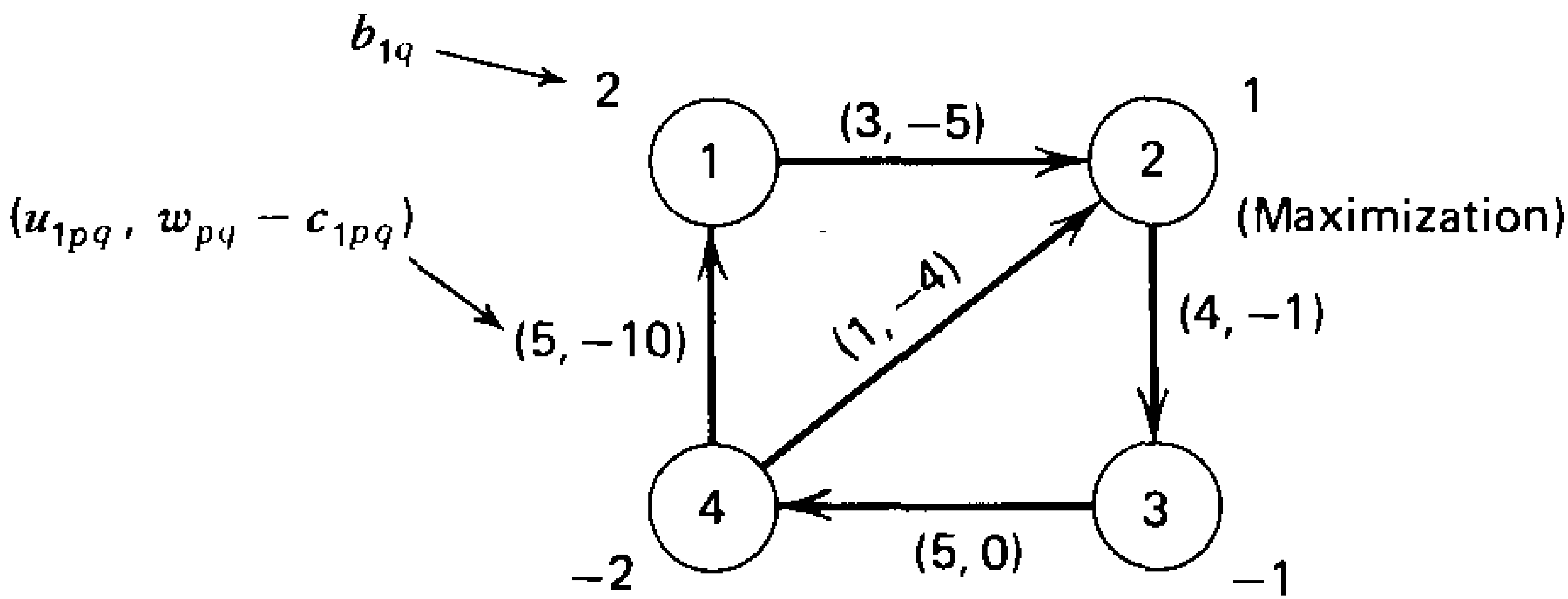
Iteration 2

- Again all $w_{pq} \leq 0$
  - so no s_{pq} is a candidate to enter the master basis

Iteration 2: SUBPROBLEM 1

$$(\mathbf{w} - \mathbf{c}_1) = (-5, -1, 0, -10, -4)$$

- Subproblem 1 is the single-commodity flow problem defined in:



Iteration 2: SUBPROBLEM 1

- The optimal solution is $\mathbf{x}_{13} = (2, 3, 2, 0, 0)^t$
- with the value of the subproblem 1 objective is:

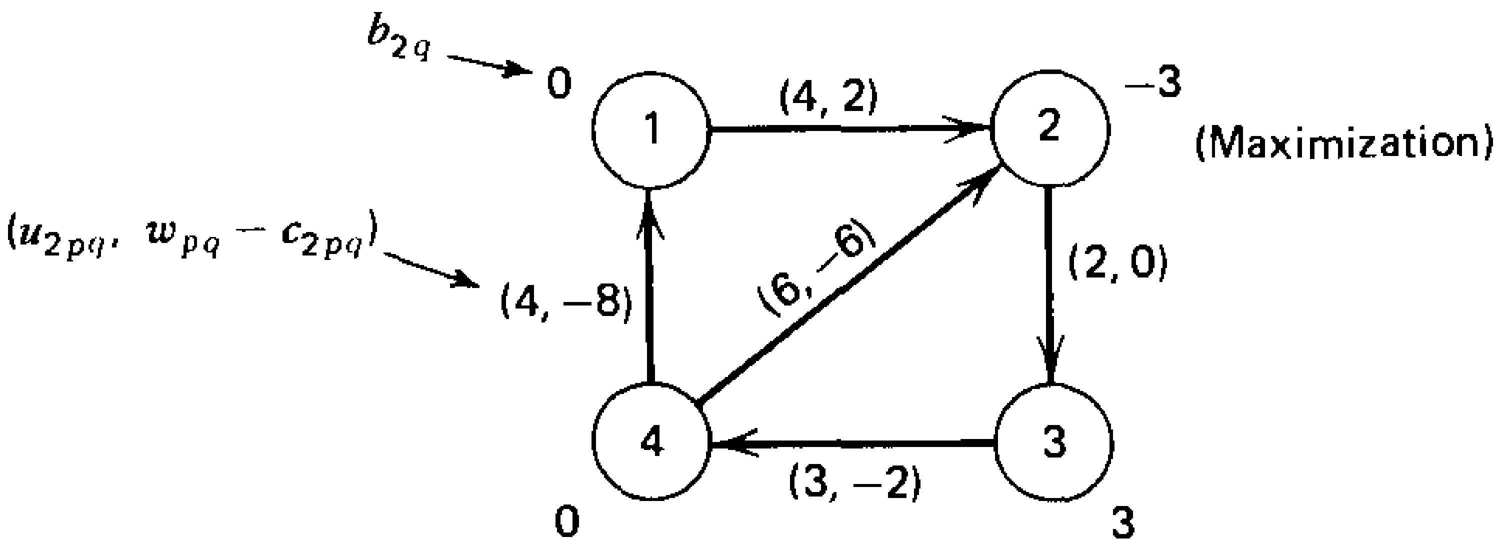
$$z_{13} - c_{13} = (\mathbf{w} - \mathbf{c}_1)\mathbf{x}_{13} + \alpha_1 = -13 + 13 = 0$$

- Thus there is no candidate from subproblem 1.

Iteration 2: SUBPROBLEM 2

$$(\mathbf{w} - \mathbf{c}_2) = (2, 0, -2, -8, -6)$$

- Subproblem 2 is the single-commodity flow problem defined in:



Iteration 2: SUBPROBLEM 2

- An optimal solution is

  $\mathbf{x}_{23} = (3, 0, 3, 3, 0)^t$

- with the value of the subproblem 2 objective is:

$$z_{23} - c_{23} = (\mathbf{w} - \mathbf{c}_2)\mathbf{x}_{23} + \alpha_2 = -24 + 24 = 0$$

- Thus there is no candidate from subproblem 2.

Iteration 2: SUBPROBLEM 2

- Therefore we already have the optimal solution as follows:

$$z^* = 28$$

$$\mathbf{x}_1^* = \lambda_{11}\mathbf{x}_{11} = (2, 3, 2, 0, 0)^t$$

$$\mathbf{x}_2^* = \lambda_{21}\mathbf{x}_{21} + \lambda_{22}\mathbf{x}_{22}$$

$$= \tfrac{2}{3}(0, 0, 3, 0, 3)^t + \tfrac{1}{3}(3, 0, 3, 3, 0)^t$$

$$= (1, 0, 3, 1, 2)^t$$

References

References

- M.S. Bazaraa, J.J. Jarvis, H.D. Sherali, **Linear Programming and Network Flows**, Wiley, 1990. (Chapter 12)

The End