CHAPTER 14

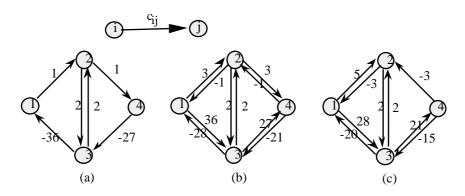
EXERCISE 14.1 (a) **Always convex**. $h(x_1 + (1-)x_2) = f(x_1 + (1-)x_2) + g(x_1 + (1-)x_2)$ ($f(x_1) + g(x_2) + (1-)(f(x_2) + g(x_2)) = h(x_1) + (1-)h(x_2)$.

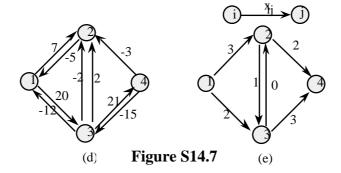
- (b) **Not necessarily convex**. Counterexample: The functions f(x) = x, and $g(x) = x^2$ are convex, but $h(x) = x x^2$ is concave.
- (c) Not necessarily convex. Counterexample: The functions $f(x) = -x^{1/4}$ is convex, but $h(x) = x^{1/2}$ is concave.
- (d) Not necessarily convex. Counterexample: The functions f(x) = x is convex, but $h(x) = x^{1/2}$ is concave.

EXERCISE 14.3. Let x^1 and x^2 be the flows when the supply/demand vector is set to b($_1$) and b($_2$), respectively. It is easy to see that the flow $x^r = rx^1 + (1-r)x^2$ will satisfy the mass balance constraints when the supply/demand vector is set to b($_1$ +(1-r) $_2$) for all 0 r 1. Also observe that for each (i, j) A, x^r_{ij} will be between x^1_{ij} and x^2_{ij} and, therefore would satisfy the flow bound constraints. Hence, x^r is a feasible flow to the problem with right-hand side equal to b($_1$ +(1-r) $_2$) and with an objective function value of r z($_1$)+(1-r)z($_2$), which would be an upper bound on the optimal solution value x($_1$) of this problem. Consequently, z($_1$) = z(r $_1$ +(1-r) $_2$) and is a convex function.

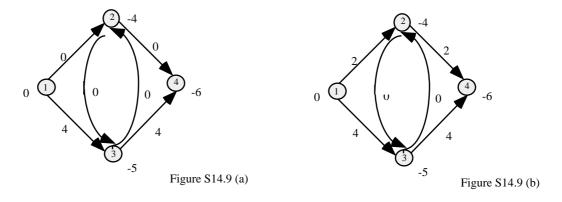
EXERCISE 14.5. Let G = (N, A) be the network in which the nearly feasible flow has to be found. Construct network $G^O = (N \ \{s\}, A \ A_O)$, where $A_O = \{(s, i) : i \ N\} \ \{(i, s) : i \ N\}$. Each arc in A_O is uncapacitated and has a cost function of $C_{ij}(x_{ij}) = x_{ij}^2$. The cost of each arc in A is zero. From a minimum cost flow in G^O , we can obtain a nearly feasible flow in G by deleting arcs in A_O .

EXERCISE 14.7. Since our cost function is a concise one, it is impossible to find an exactly optimal solution using the cycle canceling algorithm. Hence, we shall restrict ourselves to finding an integer optimal solution. The sequence of residual network is illustrated in Figure S14.7. The first two augmentations are of one unit, each along the cycle 1-2-4-3-1, and the third augmentation is also of one unit and is along the cycle 1-2-3-1. The optimal integer flow is given in Figure S14.7(e).

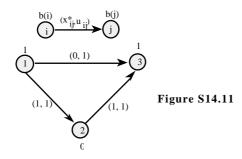




EXERCISE 14.9. Since U = 5, in the first scaling phase, we use only segments of size 4, while in the second scaling phase, segments of size 2 are used. In the first scaling phase, only one augmentation of 4 units is performed along the path 1-3-4. The optimal flow and node potentials at the end of the first scaling phase is illustrated in Figure S14.9(a). At the beginning of the second scaling phase, the arcs (2, 4) and (1, 2) violate their optimality conditions and the flow on arc (2, 4) and (1, 2) is increased by = 2 units. The flow and node potentials at the end of this preprocessing (for the second scaling phase) are illustrated in Figure S14.9(b). No augmentations are performed during the second scaling phase since S() = T() = 0. There are node imbalances at nodes 1 and 4, but they are less than 0. Hence the node potentials do not change either.



EXERCISE 14.11. The claim is not correct. A counterexample is illustrated in Figure S4.11. Assume that $C_{ij}(x_{ij}) = 0$ for each arc in Figure S14.11. Observe that the flow x_{ij}^* shown in the figure is an optimal solution to the problem. However, when a constant of k = 1 is added to each arc, then this flow does not remain optimal.



EXERCISE 14.13. Exercise 11.3#. Instead of stating that "Formulate this problem as a minimum cost flow problem," we should say "Show how this problem can be solved using network flow techniques." This exercise has a fairly simple but clever formulation, and uses the observation that the project assignment problem can be

decomposed into two independent problems. The first problem matches the students to projects (see Figure S11.3(a) and the second problem matches the projects to supervisions (see Figure S11.3 (b)).

EXERCISE 14.15. If x^* is an optimal solution of the convex cost flow problem, then $G(x^*)$ certainly does not contain a negative cycle, otherwise the cost of the flow can be improved by augmenting unit flow along such a cycle. Conversely, if the network $G(x^*)$ does not contain a negative cycle, then for any alternate flow x^O , $(x^O - x^*)$ is a circulation which can be decomposed into cycle flows along at most m cycles in $G(x^*)$. Let the cycles be W_i , ..., W_k . Then, $C(x^O) - C(x^*)$ $i=1 \\ C(W_i)$ $f(W_i)$. (Note that the convexity of the flow is used to establish the preceding inequality.) Hence, $C(x^O) - C(x^*)$ $i=1 \\ C(W_i)$ $f(W_i)$ $f(W_i)$

EXERCISE 14.17. Suppose that at the beginning of the -scaling phase, $c_{ij} < 0$ as well as $c_{ij} < 0$. Then,

$$C_{ii}(x_{ii} +) - C_{ii}(x_{ii}) - (i) + (j) < 0,$$
 (i)

and

$$C_{ij}(x_{ij}^{-}) - C_{ij}(x_{ij}) - (j) + (i) < 0.$$
 (ii)

Adding (i) and (ii), we obtain the following relationship:

$$C_{ij}(x_{ij}) \ > (C_{ij}(x_{ij} \ - \) + C_{ij}(x_{ij} \ + \)),$$

which violates our assumption that $C_{ij}(x_{ij})$ is a convex function of x_{ij} . Therefore, both c_{ij} and c_{ji} can not be negative.

EXERCISE 14.19. Replace each arc (i, j) in the network G by two parallel arcs $(i, j)^1$ and $(i, j)^2$. The arc $(i, j)^1$ has a capacity of u_{ij} and a cost of 0 units. The arc $(i, j)^2$ has a cost of u_{ij} and is uncapacitated. Now, we apply the parametric network simplex algorithm to this network and terminate when the cost of flow exactly equals D.

EXERCISE 14.21. We maintain a spanning tree at each step and an integer flow. For each nontree arc (i, j), we define two reduced costs: (i) the cost of increasing the flow on arc (i, j), which is essentially the cost of sending unit flow in the fundamental cycle W created by the arc (i, j) in the orientation of arc (i, j); and (ii) the cost of sending unit flow in W opposite to the orientation of arc (i, j). We can compute these costs by traversng the fundamental cycle W. (There exist clever methods to identify reduced costs of arcs, but they are left for the readers.) If no reduced cost is negative, then the algorithm terminates. Otherwise, we select an arc with the negative reduced cost, and augment unit flow in the appropriate direction. In fact, we may do binary search to augment more flow, as described in the adaptation of the cycle canceling algorithm in Setion 14.4.

EXERCISE 14.23. Notice that when we define the cost of an arc equal to its -incremental cost, then it means that the concise function $C_{ij}(x_{ij})$ has been approximated by a piecewise linear convex function in which the length of each linear segment is units. It is clear that if the solution x is optimal, then G(x) will contain no negative cycle with a negative -incremental cost. The converse result can also be easily established by choosing arbitrarily close to zero and then using the proof given in Theorem 3.8.

An algorithm to identify a real-valued solution within a desired accuracy can be suggested along the following lines. Select some value of and apply the negative cycle canceling algorithm with respect to the -incremental

costs. When no negative remains, then decrease by a factor of 2, and repeat the above described process. Continue until is sufficiently small so that the answers are within the desired degree of accuracy.