No. 52 (79-16)

A New Efficient Algorithm for Finding Shortest Paths in Networks with Arcs of Negative Length

by

Satoru FUJISHIGE

A NEW EFFICIENT ALGORITHM FOR FINDING SHORTEST PATHS IN NETWORKS WITH ARCS OF NEGATIVE LENGTH

Satoru FUJISHIGE

Institute of Socio-Economic Planning

University of Tsukuba

Sakura, Ibaraki 300-31

JAPAN

Keywords: shortest paths, the Dijkstra Method, networks with negative arcs.

1. Introduction

Let $N(V,A;\ell)$ be a network with a vertex set V, an arc set A and a length function $\ell: A \to R$, where R is the set of all reals. Consider the problem of finding shortest paths from a fixed vertex v_0 to all the other vertices, where the length function is not nonnegative but the length of any directed cycle is nonnegative.

We shall propose a new efficient algorithm which solves the original shortest-path problem by repeatedly solving $d+1$ shortest-path problems for modified networks to which the Dijkstra method [1] can be applied, where $d = \min(d^+,d^-)$ and d^+ (d^-) is the number of distinct initial (terminal) vertices of arcs of negative length. By the use of the Dijkstra method, this algorithm solves the shortest-path problem in an $O((d+1)|V|^2)$ running time. Note that $d \leqq |V|$ and usual shortest-path algorithms [2] require an $O(|V|^3)$ worst-case running time regardless of the value of d. Therefore, the proposed algorithm is more efficient than the $O(|V|^3)$ algorithms when d is small, compared with $|V|$.

2. Algorithm

We call an arc of negative length a negative arc. Also, we call a directed tree T with root v_0 an optimal directed tree with root v_0 relative to length function ℓ if for any vertex v in T the unique path, in T, from vertex v_0 to vertex v is a shortest path relative to ℓ and if for any vertex v in V but not in T, if any, there is no directed path from v_0 to v in N. Moreover, we denote by $V(T)$ the set of all the vertices in T.

Let us define:

$$\bar{A} = \{a \mid a \in A,\ \ell(a) < 0\}, \tag{2.1}$$

$$\bar{V}^+ = \{v \mid v = \partial^+ a,\ a \in \bar{A}\}, \tag{2.2}$$

$$\bar{V}^- = \{v \mid v = \partial^- a,\ a \in \bar{A}\}, \tag{2.3}$$

where $\partial^+ a$ ($\partial^- a$) denotes the initial (terminal) vertex of arc a. Note that $d^+ = |\bar{V}^+|$ and $d^- = |\bar{V}^-|$. Also, define:

$$\bar{A}^+(v) = \{a \mid a \in \bar{A}, \partial^+ a = v\} \qquad (v \in \bar{V}^+), \tag{2.4}$$

$$\bar{A}^-(v) = \{a \mid a \in \bar{A}, \partial^- a = v\} \qquad (v \in \bar{V}^-). \tag{2.5}$$

Now, the algorithm for solving the shortest-path problem is given as follows.

Algorithm for Solving the Shortest-Path Problem

Step I: For a given network $N(V,A;\ell)$, find the values $d^+ = |\bar{V}^+|$ and $d^- = |\bar{V}^-|$ defined by (2.1) - (2.3).

Step II: If $d^+ \leqq d^-$, then carry out Algorithm(+); or else, carry out Algorithm(-). Stop.

Algorithm(+)

Step 1: Put

$$\tilde{\ell}(a) := 0 \qquad (a \in \bar{A}),$$
$$:= \ell(a) \qquad (a \in A - \bar{A}).$$

Find an optimal directed tree T with root v_0 relative to the length function $\tilde{\ell}$ by use of the Dijkstra method. Put

$$\underline{\text{previous}}(v) = \partial^+ a \text{ for the unique arc } a \text{ in } T \text{ such that } \partial^- a = v \ (v \in V(T) - \{v_0\}), \tag{2,7}$$

$p(v) :=$ the length of the shortest path from v_0 to v relative to the length function $\tilde{\ell}$ $(v \in V)$. (2.8)

(Here, $p(v)$ is equal to ∞ if there is no directed path from v_0 to v and functions <u>previous</u> and p are determined in the course of carrying out the Dijkstra method.)

Also, put

$$\tilde{\ell}(a) := \tilde{\ell}(a) - p(\partial^- a) + p(\partial^+ a) \qquad (a \in A), \tag{2.9}$$

$$U := \overline{V}^+. \tag{2.10}$$

<u>Step 2</u>: If $|U| = 0$, then the algorithm terminates.

Otherwise, choose any $v \in U$ and put

$$U := U - \{v\}, \tag{2.11}$$

$$\tilde{\ell}(a) := \tilde{\ell}(a) + \ell(a) \qquad (a \in \overline{A}^+(v)). \tag{2.12}$$

If $\tilde{\ell}(a) \geqq 0$ $(a \in \overline{A}^+(v))$, then (T is an optimal directed tree with root v_0 relative to length function $\tilde{\ell}$ and) go back to the beginning of Step 2.

<u>Step 3</u> (A modified Dijkstra method):

(i) Put

$v^* :=$ the set of vertices lying on the path, in T, from vertex v_0 through vertex v, (2.13)

$$\tilde{p}(v) := 0 \qquad (v \in V^*),$$

$$:= \infty \qquad (v \in V - V^*), \tag{2.14}$$

$$U^* := V^*. \tag{2.15}$$

(ii) For each arc $a \in A$ such that $\partial^+ a \in U^*$ and $\partial^- a \in V - V^*$, if $\tilde{p}(\partial^- a) > \tilde{p}(\partial^+ a) + \tilde{\ell}(a)$, then put

$$\tilde{p}(\partial^- a) := \tilde{p}(\partial^+ a) + \tilde{\ell}(a), \tag{2.16}$$

$$\underline{\text{previous}}(\partial^{-}a) := \partial^{+}a. \tag{2.17}$$

(iii) If there is no vertex v in $V - V^*$ such that $\tilde{p}(v) < \infty$, then go to Step 4; or else, find a vertex $\hat{v}$ in $V - V^*$ such that

$$\tilde{p}(\hat{v}) = \min\{p(v) \mid v \in V - V^*,\ p(v) < \infty\}. \tag{2.18}$$

If $\tilde{p}(\hat{v}) = 0$, then go to Step 4; or else, put

$$U^* := \{v \mid v \text{ is a vertex in the directed subtree, of } T, \text{ with root } \hat{v}\}, \tag{2.19}$$

$$V^* := V^* \cup U^*, \tag{2.20}$$

$$\tilde{p}(v) := \tilde{p}(\hat{v}) \qquad (v \in U^*). \tag{2.21}$$

If $V^* = V$, then go to Step 4; or else, go back to (ii) of Step 3.

Step 4: Let us put

$$p(v) := p(v) + \tilde{p}(v) \qquad (v \in V^*), \tag{2.22}$$

$$\tilde{\ell}(a) := \tilde{\ell}(a) - \tilde{p}(\partial^{-}a) + \tilde{p}(\partial^{+}a) \qquad (a \in A), \tag{2.23}$$

$$T := \text{the directed tree with root } v_0 \text{ defined by the function } \underline{\text{previous}}. \tag{2.24}$$

Go back to Step 2.

Algorithm(−)

Step 1: (The same as Step 1 of Algorithm(+) except that $\overline{V}^{+}$ is replaced by $\overline{V}^{-}$.)

Step 2: (The same as Step 2 of Algorithm(+) except that $\overline{A}^{+}(v)$ is replaced by $\overline{A}^{-}(v)$.)

Step 3(−): Let a^* be an arc in $\overline{A}^{-}(v)$ satisfying

$$\tilde{\ell}(a^*) = \min\{\tilde{\ell}(a) \mid a \in \overline{A}^{-}(v)\} \tag{2.25}$$

and put

$$v := \partial^{+}a^{*}. \tag{2.26}$$

Step 3: (The same as Step 3 of Algorithm(+).)

Step 4: (The same as Step 4 of Algorithm(+).)

Remarks: When the algorithm terminates, $p(v)$ is the length of a shortest path from v_0 to v ($v \in V$) and T is an optimal directed tree with root v_0 relative to length function ℓ.

Note that Step 3 is an efficient implementation of the Dijkstra method to the network $N(V,A;\tilde{\ell})$ (cf. (2.18) - (2.21)), where use is made of the fact that the length of every arc of T obtained in Step 1 (Step 4) is equal to zero relative to the nonnegative length function $\tilde{\ell}$ given by (2.9)((2.23)).

3. Validity of the Algorithm

First, let us consider the validity of Algorithm(+). We may have only to show the validity of Step 3. At the beginning of Step 3, for any vertex u the length of a shortest path from v_0 to u relative to length function $\tilde{\ell}$ given by (2.12) can be negative only if such a shortest path contains arcs in $\overline{A}^{+}(V)$ and thus vertex v. Therefore, the unique path, in T, from v_0 to v must be a shortest path from v_0 to v relative to $\tilde{\ell}$, since there is no directed cycle of negative length in $N(V,A;\tilde{\ell})$. Consequently, the validity of the application of the Dijkstra method follows from the fact that negative arcs in $\overline{A}^{+}(v)$ are processed by (2.16) and (2.17) in the first application of (ii) of Step 3.

Next, we show the validity of Steps 3(-) and 3 of Algorithm(-). Since there is no directed cycle of negative length in $N(V,A;\tilde{\ell})$, any elementary shortest path P from v_0 to v relative to length fuction $\tilde{\ell}$ must contain one and only one arc, say $\hat{a}$, in $\overline{A}^-(v)$, where vertex v is the original one before it is replaced by $\partial^+ a^*$ in Step 3(-). Since the path, contained in P, from v_0 to $\partial^+\hat{a}$ does not contain arcs in $\overline{A}^-(v)$, by the way of determining a^* by (2.25) the path from v_0 to v composed of the path from v_0 to $\partial^+ a^*$ in T and arc a^* must be a shortest path from v_0 to v relative to $\tilde{\ell}$ as well. The Dijkstra method is applicable since all the negative arcs in $N(V,A;\tilde{\ell})$ have vertex $v = \partial^- a^*$ as their terminal vertices and there is no directed cycle of negative length in $N(V,A;\tilde{\ell})$.

References

[1] E. W. Dijkstra: A note on two problems in connexion with graphs. Numerische Mathematik, 1 (1959) 269-271.

[2] A. V. Aho, J. E. Hopcroft and J. D. Ullman: The Design and Analysis of Computer Algorithms. Addison-Wesley, Reading, Massachusetts, 1974.