<table>
<thead>
<tr>
<th>Title</th>
<th>On the Importance of Each Edge Using Its Traffic along Shortest Paths in a Network</th>
</tr>
</thead>
<tbody>
<tr>
<td>Author(s)</td>
<td>CHENG, Peng; MASUYAMA, Shigeru</td>
</tr>
<tr>
<td>Citation</td>
<td>数理解析研究所講究録 (1994), 871: 100-104</td>
</tr>
<tr>
<td>Issue Date</td>
<td>1994-05</td>
</tr>
<tr>
<td>URL</td>
<td><a href="http://hdl.handle.net/2433/84050">http://hdl.handle.net/2433/84050</a></td>
</tr>
<tr>
<td>Right</td>
<td></td>
</tr>
<tr>
<td>Type</td>
<td>Departmental Bulletin Paper</td>
</tr>
<tr>
<td>Textversion</td>
<td>publisher</td>
</tr>
</tbody>
</table>

Kyoto University
On the Importance of Each Edge Using Its Traffic along Shortest Paths in a Network

Introduction

In our daily life, we may usually select some shortest path from A to B in order to travel from a place A to a place B in a road network. It implies that the traffic passing through each road interval based on some rule of selecting shortest paths is considered as a measure of the importance of each road interval for a road network.

We formalize a road network as a digraph, namely, directed graph $G = (V, E)$ with two specified source $s$ and sink $t$, where each edge $e$ has a positive real length $l(e)$, namely $l(e) > 0$. As there may be a large number of shortest paths between two vertices in a road network, a user will select one of them to suit his own convenience. To describe the user's preference among shortest paths, we define a distribution function $\alpha_v$: $f(E_v^-) \mapsto f(E_v^+)$ at each vertex $v$, where $f(e)$ denotes the traffic passing through an edge $e$ with respect to source $s$ and sink $t$, $E_v^- (E_v^+)$ represents the set of edges entering vertex $v$ (leaving vertex $v$), and $f(E_v^-) (f(E_v^+))$ represents an $|E_v^-| (|E_v^+|)$ dimensional vector consisting of traffics $f(e)$'s passing through all edges $e$'s in $E_v^- (E_v^+)$. Furthermore, we assume that the distribution function $\alpha_v$ at each vertex $v$ is computed in $O(\beta_v)$ time where $\beta_v$ is a function of the input size, that $\alpha_v$ satisfies the conservation constraint (namely, for each vertex $v$ in a digraph $G$, the amount of the traffics entering vertex $v$ is equal to the amount of the traffics leaving vertex $v$), and that the traffic $f(e)$ passing through each edge $e$ is treated as a single commodity flow with respect to source $s$ and sink $t$. Thus, we define the problem IETSP as follows.

Input: A digraph $G = (V, E)$ with source $s$ and sink $t$ where each edge $e$ has a positive real length $l(e) (> 0)$ and each vertex $v$ has a distribution function $\alpha_v$, and a required traffic $F_{st}$ from source $s$ to sink $t$.

Output: The traffic $f(e)$ passing through each edge $e$ in order to move the required traffic $F_{st}$ along only shortest paths from source $s$ to sink $t$ by using the distribution function $\alpha_v$ at each vertex $v$.

Related measures of the importance of each edge, e.g., the number of shortest paths
passing through each edge [1, 4] and that of minimum spanning trees containing each edge
[2], etc., in a network have been investigated.

In this paper, we propose a polynomial time algorithm of solving the problem IETSP,
based on the property of the topological sort of vertices of an acyclic digraph.

2 An Algorithm for IETSP

The basic idea of the algorithm described in this section for solving problem IETSP is
that we construct an acyclic subdigraph $G_{st}$ from a digraph $G$ by deleting redundant edges,
namely, edges not contained in any shortest path from $s$ to $t$, and assign the traffic $f(e)$
passing through each edge $e$ based on a topological sort of vertices in the acyclic subgraph
$G_{st}$.

For a digraph $G = (V, E)$, let $d(u, v)(\forall u, v \in V)$ denote the length of the shortest path
from $u$ to $v$, where we assume that $d(u, u) = 0$ for any $u \in V$ and that if there is no path from
$u$ to $v$ then $d(u, v) = \infty$. Let $L(\pi)$ denote the length of a path $\pi$ from $u$ to $v$, namely, the
sum of lengths of edges in a path $\pi$. It is well-known that the lengths $d(u, v)$’s ($\forall u, v \in V$)
of shortest paths for all pairs of vertices over $V$ are found in polynomial time (see e.g. [5]).
Furthermore, we prove the following Lemma 1.

**Lemma 1.** In a digraph $G = (V, E)$ with source $s$ and sink $t$, there is a shortest path from $s$
to $t$ containing an edge $(u, v) \in E$ if, and only if, $d(s, t) \neq \infty$ and $d(s, u) + l((u, v)) + d(v, t) =
d(s, t)$ hold.

**Proof.** Necessity: Assume that $G$ has a shortest path $\pi$ from $s$ to $t$ containing edge $(u, v)$,
and let $\pi: \pi_{su}, (u, v), \pi_{vt}$, where $\pi_{su}$ and $\pi_{vt}$ are paths from $s$ to $u$ and from $v$ to $t$,
respectively. Clearly, $d(s, u) \neq \infty$ and $d(v, t) \neq \infty$ hold. As $d(s, t) = L(\pi_{st}) = L(\pi_{su}) +
\pi_{vt}) + L(\pi_{vt})$ holds, we have $L(\pi_{su}) = d(s, u)$ and $L(\pi_{vt}) = d(v, t)$. Otherwise, $d(s, t) >
L(\pi_{st})$ holds, contradicting the assumption.

Sufficiency is obvious. \hfill \Box

By Lemma 1, we can delete all redundant edges with respect to $s$, $t$ from a digraph $G$,
based on the lengths $d(u, v)$’s of shortest paths in $G$, and obtain the following subdigraph
$G_{st} = (V_{st}, E_{st})$ of $G$ having no redundant edge.

$$E_{st} = \{ (u, v) \in E \mid d(s, t) \neq \infty \text{ and } d(s, u) + l((u, v)) + d(v, t) = d(s, t) \},$$

$$V_{st} = \{ v \in V \mid (u, v) \text{ or } (v, u) \in E_{st} \}.$$

It is clear that, for a digraph $G = (V, E)$ with source $s$ and sink $t$, if $d(u, v)$’s ($\forall u, v \in V$)
are known, then $G_{st}$ is obtained in $O(|E|)$ time. Furthermore, by the definition of $G_{st}$, for
any edge \((u, v)\) of the subdigraph \(G_{st}\) of \(G\), there is a shortest path from \(s\) to \(t\) containing edge \((u, v)\) in \(G\). The following Lemma 2 is also proved.

**Lemma 2.** In a digraph \(G = (V, E)\) with source \(s\) and sink \(t\), each shortest path from \(s\) to \(t\) in \(G\) corresponds one-to-one to each path from source \(s\) to sink \(t\) in the subdigraph \(G_{st}\) obtained from \(G\).

**Proof.** Let a shortest path from \(s\) to \(t\) in \(G\) be \(\pi_{st}\):

\[s, (s, v_1), (v_1, v_2), v_2, \cdots, (v_k, t), t.\]

Then any subpath \(\pi_{sv_i}(1 \leq i \leq k)\) of \(\pi_{sv}\) is a shortest path from \(s\) to \(v_i\) in \(G\), as, otherwise, \(\pi_{st}\) is not a shortest path. Thus, by the definition of \(G_{st}\), all edges on \(\pi_{st}\) must be in \(E_s\).

On the other hand, let a path from \(s\) to \(t\) in \(G_{st}\) be \(\pi_{st}\):

\[s, (s, v_1), (v_1, v_2), v_2, \cdots, (v_k, t), t.\]

By the definition of \(G_{st}\),

\[d(s, s) + l((s, v_1)) = d(s, v_1),\]
\[d(s, v_1) + l((v_1, v_2)) = d(s, v_2),\]
\[\vdots\]
\[d(s, v_k) + l((v_k, t)) = d(s, t)\]

hold, where \(d(s, s) = 0\). Hence, we have

\[L(\pi_{st}) = l((s, v_1)) + l((v_1, v_2)) + \cdots + l((v_k, t)) = d(s, t).\]

This means that \(\pi_{st}\) is a shortest path in \(G\).

**Lemma 3.** For a digraph \(G = (V, E)\) with source \(s\) and sink \(t\) where each edge \(e\) has a positive real length \(l(e)\), the subdigraph \(G_{st}\) obtained from \(G\) has no cycle, namely, is acyclic.

**Proof.** Assume that \(G_{st}\) has a cycle \(C\). Let \(v\) be a vertex on \(C\). Consider a shortest path \(\pi_{st}\) from \(s\) to \(t\) passing through vertex \(v\), namely, \(L(\pi_{st}) = d(s, t)\). Let \(\pi'_{st}\) be a path from \(s\) to \(t\) obtained by concatenating subpath \(\pi_{sv}\) of \(\pi_{st}\), cycle \(C\) and subpath \(\pi_{vt}\) of \(\pi_{st}\), where \(C\) is treated as a path from \(v\) to \(v\). As \(L(\pi'_{st}) = L(\pi_{sv}) + L(C) + L(\pi_{vt}) \leq d(s, t) = L(\pi_{st})\) holds, we have \(L(C) = 0\), which, however, contradicts the assumption that each edge \(e\) of \(G\) has a positive real length \(l(e)\)(> 0).

Note that for the subdigraph \(G_{st}\) obtained from a digraph \(G = (V, E)\), the in-degree of source \(s\) is zero and out-degree of sink \(t\) is zero. A topological sort of vertices [3] in the acyclic subdigraph \(G_{st}\) must start from source \(s\) and end at sink \(t\), namely,

\[v_1(= s), v_2, v_3, \cdots, v_{|V_{c}|-1}, v_{|V_{c}|}(= t).\]
Lemma 4[3]. For a digraph $G = (V, E)$ with source $s$ and sink $t$, any topological sort of vertices: $v_1(= s), v_2, \cdots, v_{|V_{st}|} = t$ in the subdigraph $G_{st}$ obtained from $G$ satisfies
(i) For any $i(1 \leq i \leq |V_{st}|)$, the tail $v_j$ of any edge with head $v_i$ is to the left of $v_i$, namely, for any edge $(v_j, v_i)$, $j < i$ holds, and
(ii) Any path from $v_1(= s)$ to $v_i$ can contain only some vertices of $(s =) v_1, v_2, \cdots, v_i$. □

Based on the above discussions, we describe the following algorithm for problem IETSP.

**Algorithm IETSP**

**Input**: A digraph $G = (V, E)$ with source $s$ and sink $t$ where each edge $e$ has a positive real length $l(e)(> 0)$, a distribution function $\alpha_v$ at each vertex $v$ and a required traffic $\mathcal{F}_{st}$ from source $s$ to sink $t$.

**Output**: The traffic $f(e)$ passing through each edge $e$ of $E$ in order to assign the traffic $\mathcal{F}_{st}$ along only shortest paths from source $s$ to sink $t$ by the given distribution function $\alpha_v$ at each vertex $v$.

**Begin**

A1. For each edge $(u, v)$ of $E$, $f((u, v)) := 0$.

A2. Compute the lengths $d(u, v)$'s of shortest paths for all vertex pairs $u, v (\in V)$ by applying the algorithm shown in [5].

A3. Construct $G_{st} = (V_{st}, E_{st})$ by deleting redundant edges based on the known values $d(u, v)$'s obtained in A2.

A4. Obtain a topological sort of vertices $v_1(= s), v_2, \cdots, v_{|V_{st}|} = t$ in $G_{st}$ by executing the algorithm shown in e.g., [3].

A5. For $i = 1$ to $|V_{st}|$, obtain $f(E_{v_i}^+)$ by computing $f_{v_i}(E_{v_i}^-)$, and output $f(E_{v_i}^+)$. □

**End**

The correctness of Algorithm IETSP is easily derived by the above lemmas. Now, we analyze the time complexity of Algorithm IETSP. The time complexity of executing A1, A3, and A4 in Algorithm IETSP is $O(|E|)$ by the above lemmas. A2 is executed in $O(|V|^3(\log \log |V|/ \log |V|)^{1/2})$ time by the algorithm shown in [5]. The time complexity of executing A5 is $O(|V| \max_{v \in V} O(\beta_v))$ as $f_{v_i}(E_{v_i}^-)$ is computed in $O(\beta_v)$ time where $\beta_v$ is a function of the input size. By the above analysis of the time complexity, we obtain the following Theorem 1.

**Theorem 1.** Given a digraph $G = (V, E)$ with source $s$ and sink $t$ where each edge $e$ has a positive real length $l(e)(> 0)$, and each vertex $v$ has a distribution function $\alpha_v$, and a required traffic $\mathcal{F}_{st}$ from source $s$ to sink $t$. Then, we can compute the traffic $f(e)$ passing each edge $e$ in $O(\max\{|V| \max_{v \in V} O(\beta_v), |V|^3(\log \log |V|/ \log |V|)^{1/2}\})$ time by Algorithm
IETSP, in order to move the required traffic $\mathcal{F}_{st}$ along only shortest paths from $s$ to $t$ by the distribution function $\alpha_v$ at each vertex $v$.

It is clear that if a distribution function $\alpha_v$ at each vertex $v$ is computed in polynomial time for the input size, Algorithm IETSP is polynomial. Note that the assumption that a distribution function $\alpha_v$ at each vertex $v$ is computable in polynomial time, is not strong.

3 Conclusion

Based on the property of the topological sort of vertices of an acyclic digraph $G_{st}$ obtained by removing redundant edges from $G$, we obtain a polynomial time algorithm for solving the problem IETSP. It is easy to see that the results obtained in this paper also hold if a digraph $G$ contains no cycle of a negative length or a zero length.

It is obvious that the problem with respect to all vertex pairs similar to the problem IETSP with respect to one vertex pair $\{s,t\}$ can be solved in polynomial time by applying Algorithm IETSP $|V|^2$ times.

References


