Discrete Applied Mathematics 9 (1984) 41-46
North-Holland

ON THE WINDY POSTMAN PROBLEM

Meigu GUAN*
Department of Combinatorics and Optimization, University of Waterloo, Waterloo, Canada

Received 31 March 1983
Revised 12 August 1983

1. The problem

The so-called 'windy postman problem' was proposed by E. Minieka in [1]. The problem is: "Let $G=(V,E)$ be a connected undirected graph, such that for every $e\in E$, $e=(x,y)$, the distance from x to y can be different from the distance from y to x. Find a shortest postman route for graph G".

Sometimes, one direction of a street may be uphill and the other direction downhill, or one direction may be with the wind and the other direction against the wind. For these cases, using the windy postman problem (abbreviated WPP) to model the real-world problem is suitable.

We can restate the above problem as follows:

Construct a symmetric directed graph $G_1=(V,A)$, where the vertex set V is the same as that in G, and corresponding to each edge $e=(x,y)$ in E, there are two arcs $a'=(x,y)$ and $a''=(y,x)$ in A, both a' and a'' have non-negative length $l(a')$ and $l(a'')$. Now we want to find a shortest directed closed path p in G_1, such that: for every $e\in E$, at least one of the corresponding arcs $\{a',a''\}$ belong to p.

In the sequel, we shall always use a' and a'' to denote the arcs in G_1 corresponding to an edge e, and use C' and C'' to denote the directed cycles in G_1 corresponding to a cycle C in G.

2. The NP-completeness of WPP

Theorem 1. WPP *is NP-hard.*

Proof. It is proved in [1] that WPP can be reduced to the mixed postman problem. We now prove that the mixed postman problem can also be reduced to WPP.

In fact, let G be a mixed graph, for each directed arc $a=(x,y)$, we add an arc $\bar{a}=(y,x)$. The length of these added arcs is a large number M. The resulting graph

*On leave from Shandong Teachers' University.

$\bar{G}$ is a symmetric graph (considering the undirected edges in G as two arcs with different directions), so we can consider WPP on $\bar{G}$.

For each postman route in the new problem using no added arc, there is a postman route in the old problem having the same cost. And conversely, by the choice of M, solving WPP in the new problem solves the mixed postman problem, or shows that it has no feasible solution.

We have proved that the WPP is equivalent to the mixed postman problem, and since it is proved that the mixed postman problem is NP-hard in [2], so the WPP is NP-hard. □

3. A special class of WPP which can be solved by a polynomial bounded algorithm

Definition 1. We say that a graph G satisfies condition Q, if for every cycle C in G, the corresponding C' and C'' have the same length.

Now we give a polynomial bounded algorithm, which can be used to solve the WPP for graphs satisfying condition Q.

Algorithm A

Step 1: Define the weight of an edge e in G to be $w(e)=\frac{1}{2}\{l(a')+l(a'')\}$.

Step 2: Find a minimum Euler supergraph $\bar{G}$ of G by using any polynomial bounded algorithm ([3] or [4]).

Step 3: Arbitrarily find an Euler tour p for $\bar{G}$; then p is the required shortest postman route for the WPP.

Before proving the validity of Algorithm A, let us first give a transformation of the length of arcs of graph G_1. Assuming that we have associated a real number u_i with each vertex v_i of G, then, for any edge $e=(v_k, v_l)$ in G, we can transform the lengths of the two corresponding arcs as follows:

$$\begin{aligned} &\text{for } a'=(v_k, v_l), \text{ set } \quad \bar{l}(a') = l(a')+u_l-u_k, \\ &\text{for } a''=(v_l, v_k), \text{ set } \quad \bar{l}(a'') = l(a'')+u_k-u_l. \end{aligned}$$

(After such transformation, some arcs may have negative length).

Lemma 1. *Let C be a directed cycle in G_1, $l(C)$ and $\bar{l}(C)$ be the lengths of C w.r.t. to the length functions l and $\bar{l}$ respectively. Then $l(C)=\bar{l}(C)$.*

Proof. Trivial.

Lemma 2. *Let p be a postman route of G, then $l(p)=\bar{l}(p)$.*

Proof. Decompose p into the union of several directed cycles in G_1; then Lemma 2 follows from Lemma 1 directly.

Lemma 3. *Let T be a spanning tree of G. Then there exists a set of $\{u_i\}$ such that, for every edge $e \in T$, $\bar{l}(a') = \bar{l}(a'')$.*

Proof. Choose any vertex v_j and set $u_j = 0$. Now suppose that $e = (v_k, v_l)$ is an edge of T, such that u_k has been defined, but u_l has not. Set

$$u_l = u_k + \tfrac{1}{2}\{l(v_k, v_l) - l(v_l, v_k)\}.$$

Obviously, we can obtain a u_i for every v_i in this way. And it is not difficult to check that after the length transformation, $\bar{l}(a') = \bar{l}(a'')$ holds for every edge e in T.

Lemma 4. *Suppose that condition Q is satisifed for graph G. Then there exists a set of $\{u_i\}$, such that, for every edge e of G, $\bar{l}(a') = \bar{l}(a'')$.*

Proof. Arbitrarily choose a spanning tree T of G, and determine a set of u_i according to Lemma 3. For any non-tree edge e_i, denote the unique cycle in $T + e_i$ by C_i, then

$$\bar{l}(C_i') = l(C_i'), \qquad \bar{l}(C_i'') = l(C_i'')$$

by Lemma 1. Since condition Q is satisfied, we have $l(C_i') = l(C_i'')$, so

$$\bar{l}(C_i') = \bar{l}(C_i''). \tag{1}$$

But the following equalities are valid for all the edges e in C_i except e_i:

$$\bar{l}(a') = \bar{l}(a''). \tag{2}$$

Combining (1) and (2) we have $\bar{l}(a_i') = \bar{l}(a_i'')$.

Theorem 2. *If condition Q is satisfied for graph G, then the postman route obtained by Algorithm A is a shortest postman route for the WPP on G.*

Proof. Choose a set of $\{u_i\}$ by Lemma 4. After the length transformation, we have

$$\bar{l}(a') = \bar{l}(a'') \quad \text{for every } e \in E.$$

Obviously,

$$\bar{l}(a') = \bar{l}(a'') = \tfrac{1}{2}\{l(a') + l(a'')\}.$$

It is easy to see that the postman route p obtained by Algorithm A is the shortest postman route for graph G with $\bar{l}$ as length function. So, according to Lemma 2, it is also the shortest postman route for the original WPP on G. □

According to Definition 1, if we want to know whether condition Q is satisfied or not, we must check all the cycles of a graph G. But it is easy to prove the following Theorem 3, which makes the checking procedure much simpler.

Theorem 3. *If $C_1, C_2, \ldots, C_s$ (s is the cyclomatic number of graph G) is any cycle basis of graph G, and for each C_i, $l(C_i') = l(C_i'')$, then condition* Q *is satisfied.*

4. Approximation algorithm for WPP when condition Q is 'nearly' satisfied

First of all, let us consider the following problem: Under what conditions would the optimal solution of WPP transverse each edge no more than twice?

Without loss of generality, we may assume that the graph G has no vertices of degree two.

We give a condition, called condition Q_1, as follows: If for any cycle C in G and any edge $e \in C$,

$$|l(C') - (C'')| < l(a') + l(a''),$$

then we say that the graph G satisfies condition Q_1.

Theorem 4. *If a graph G satisfies condition Q_1, then any shortest postman route of* WPP *on G traverses each edge no more than twice.*

Proof. Let p be a shortest postman route of WPP on G, and suppose that p traverses an edge e more than twice. Obviously, p must traverse e in the same direction.

It is easy to see that e is not a bridge. Now, decompose p into some directed cycles; at least one of these directed cycles, say C', contains e. Set

$$p_1 = (p \setminus C') \cup C'' \setminus \{a', a''\}.$$

Obviously, p_1 is also a postman route of WPP on G. But, we have

$$\begin{aligned} l(p_1) &= l(p) - l(C') + l(C'') - l(a') - l(a'') \\ &\le l(p) + |l(C') - l(C'')| - (l(a') + l(a'')). \end{aligned}$$

since

$$|l(C') - (C'')| < l(a') + l(a'')$$

by condition Q_1. So we have $l(p_1) < l(p)$, contradicting the fact that p is a shortest postman route of WPP on G. □

The following theorem gives a sufficient condition for condition Q_1.

Theorem 5. *If there exists a real number $\varepsilon \ge 0$ and a spanning tree T of G, such that*

(a) *for every fundamental cycle C_i, $|l(C_i') - l(C_i'')| \le \varepsilon / s$ (s is the cyclomatic number of G),*

(b) *for every edge e of G, $l(a') + l(a'') > \varepsilon$,*

then condition Q_1 is satisfied.

Proof. Choose a set of $\{u_i\}$ as in Lemma 3, such that for every edge $e \in T$, $\bar{l}(a') = \bar{l}(a'')$. Now let e_i be a non-tree edge, and C_i be its corresponding fundamental cycle. Then

$$|\bar{l}(a_i') - \bar{l}(a_i'')| = |\bar{l}(C_i') - \bar{l}(C_i'')| = |l(C_i') - l(C_i'')| \le \varepsilon/s.$$

Now, let C be any cycle in G, and $e_1, e_2, \ldots, e_r$ be the non-tree edges of C. Then

$$|l(C') - l(C'')| = |\bar{l}(C') - \bar{l}(C'')| \le \sum_{i=1}^{r} |\bar{l}(a_i') - \bar{l}(a_i'')| \le s \cdot \varepsilon/s = \varepsilon.$$

And for every edge $e \in C$, $\varepsilon < l(a') + l(a'')$. So

$$|l(C') - l(C'')| < l(a') + l(a''). \qquad \square$$

Now, we consider a graph satisfying the conditions (a) and (b) of Theorem 5, for these graphs. We can use the following variation of Algorithm A to find an approximation solution for the WPP on graph G.

Algorithm $\bar{\text{A}}$

Step 1: Same as in Algorithm A.

Step 2: Same as in Algorithm A.

Step 3: Decompose the Euler supergraph $\bar{G}$ into the union of cycles, $C_1, C_2, \ldots, C_t$. For each C_i, choose the shorter one from C_i' and C_i''. Without loss of generality, assume C_i' is the shorter one.

Step 4: Combine C_i' to a directed Euler cycle p; then p is the required solution.

Theorem 6. *Let p_1 be the shortest postman route of the* WPP *on G, and p be the postman route obtained by Algorithm* $\bar{\text{A}}$. *Then* $|l(p_1) - l(p)| < s \cdot \varepsilon$.

Proof. By Theorems 4 and 5, p_1 traverses each edge of G at most twice. Let $\bar{G}'$ be the Euler super graph corresponding to p_1; then it is easy to see that $l(p_1) \le w(\bar{G}')$, and we have also $l(p) \le w(\bar{G})$.

Since p_1 is the shortest postman route of WPP on G, we have $l(p_1) \le l(p)$, and since $\bar{G}$ is a minimum Euler supergraph of G, so $w(\bar{G}) \le w(\bar{G}')$. Therefore

$$l(p_1) \le l(p) \le w(\bar{G}) \le w(\bar{G}').$$

Now, decompose $\bar{G}'$ into the union of cycles. It is easy to see that we can get a union of no more than $2s$ cycles. On each cycle C

$$|l(C') - l(C'')| \le \varepsilon \quad \text{and} \quad |l(C') - w(C)| \le \varepsilon/2.$$

Therefore

$$|w(\bar{G}') - l(p_1)| \le 2s \cdot \varepsilon/2 = s\varepsilon$$

and

$$|l(p_1) - l(p)| \le |l(p_1) - w(\bar{G}')| \le s \cdot \varepsilon. \qquad \square$$

For many real-world problems, it is reasonable to assume that condition Q is 'nearly' satisfied. This means there exist a small number $\varepsilon \geq 0$ such that (a) and (b) of Theorem 5 are satisfied. According to Theorem 6, we can use Algorithm $\bar{\mathrm{A}}$ to find an approximation solution for the problem.

References

[1] E. Minieka, The Chinese postman problem for mixed networks, Management Science 25 (1979) 643–648.

[2] C. Papadimitriou, On the complexity of edge traversing, J. Assoc. Comput. Mach. 23 (1976) 544–554.

[3] J. Edmonds and E. Johnson, Matching, Euler tours and the Chinese postman, Math. Programming 5 (1973) 88–124.

[4] J. Edmonds, The Chinese postman problem, Operations Research, Vol. 13, Supplement I (1965) B73.