

Algorithm for solution of a routing problem

Tran Vu Thieu*, Pham Xuan Hinh

Institute of Mathematics, 18 Hoang Quoc Viet, Cau Giay, Hanoi, Vietnam

Received 15 November 2006; received in revised form 12 September 2007

Abstract. In this paper we consider a combinatorial optimization problem that is similar to the bottleneck traveling salesman problem. We show that an optimal tour for this problem is pyramidal tour (1, 3, 5, ..., n, ..., 6, 4, 2) or consists of some pyramidal subtours. The above methods can be extended to complete bipartite graphs.

1. Problem statement

It is well-known that the traveling salesman problem (TSP) is strongly NP-hard (cf. [1], p. 353). But for some special cases of the TSP can be solvable in polynomial time. This is the case where the distance matrix in the TSP fulfills certain additional conditions (e.g. the Monge property, Kalmanson matrices, the Demidenko conditions or the Supnick conditions), cf. [2-4].

In the sequel we will introduce another special case of the TSP which can easily be solvable and show that the optimal tour for this problem is pyramidal tour or consists of some (at most three) pyramidal subtours.

Consider a complete graph $G = (A, E)$ with vertex set $A = \{a_1, a_2, \dots, a_n\}$ and edge set $E = A \times A$. Each vertex $a_i \in A$ has a real number t_i ($i = 1, \dots, n$), called the altitude of vertex a_i . We specify vertices $a_b \in A$ (the source) and $a_c \in A$ (the sink) such that $b, c \in \{1, 2, \dots, n\}$ and $t_b \leq t_c$. Consider the following problem, called *Problem A* for short:

Problem A. Find a Hamiltonian path in the graph from a_b to a_c (which visits every vertex exactly once) so that to minimize the highest difference between altitudes of any two successive vertices in the path.

In other word, among permutations $\pi = \{i_1, i_2, \dots, i_n\}$ of the numbers 1, 2, ..., n with $i_1 = b$, $i_n = c$ find a permutation so that to minimize the function

$$f(\pi) = \max_{1 \leq k \leq n-1} |t_{i_k} - t_{i_{k+1}}| \rightarrow \min.$$

It is easy to see that such a permutation corresponds to a Hamiltonian path in the graph from a_b to a_c and that the total number of permutations is equal to $(n-2)!$ Let us denote by \mathcal{P} the set of all these permutations. Each permutation $\pi \in \mathcal{P}$ is called a *tour* from a_b to a_c and permutation $\pi^* = \operatorname{argmin}\{f(\pi) : \pi \in \mathcal{P}\}$ *optimal tour* from a_b to a_c .

Also consider a similar problem:

Problem B. Find a Hamiltonian cycle in the given graph so that to minimize the biggest difference between altitudes of any two successive vertices in the cycle. Each of such cycles is also

* Corresponding author. Tel.: 84-4-8351235
E-mail: trvuthieu@yahoo.com

referred to as an *optimal tour*. It is obvious that the number of Hamiltonian cycles in the graph is equal to $n!$

As before, this problem is also formulated as follows: Find a cyclic permutation $\tau = \{i_1, i_2, \dots, i_n\}$ of the numbers $1, 2, \dots, n$ so that

$$g(\tau) \equiv \max_{1 \leq k \leq n} |t_{i_k} - t_{i_{k+1}}| \rightarrow \min$$

with convention that $i_{n+1} = i_1$. The solution of the above stated problems does not change if we replace t_i by $t'_i = t_i + t$ with t being an arbitrary real number. So by taking $t > 0$ sufficiently large we can assume that $t_i > 0$ for all $i = 1, 2, \dots, n$. Furthermore, without loss of generality we can assume (by enumerating the vertices of the graph if needed)

$$0 < t_1 \leq t_2 \leq \dots \leq t_b \leq \dots \leq t_c \leq \dots \leq t_n. \tag{1}$$

The vertices that have the same value t_i can be arranged in an arbitrary order. Let in the sequence (1) there be q different values ($q \leq n$). We write these values as $h_1 < h_2 < \dots < h_q$. We divide the vertices of the graph into q classes, denoted by T_1, T_2, \dots, T_q such that vertex a_i belongs to class T_k if and only if $t_i = h_k$ ($i = 1, 2, \dots, n; k = 1, 2, \dots, q$). The vertices of class T_1 are first numbered then vertices of T_2 and finally vertices of T_q . The vertices of the same class are numbered in an arbitrary order.

2. Basic properties of optimal tours

We now consider the problem of finding an optimal tour from a_b to a_e in G . We temporarily assume that all t_i 's are different, that is we enumerated the vertices of the graph so that ($1 \leq b < e \leq n$)

$$0 < t_1 < t_2 < t_3 < \dots < t_{n-1} < t_n. \tag{2}$$

In the sequel we shall remove this assumption. Some useful properties of optimal tours are given in the below stated propositions. Their proofs make use of the tour improvement technique as used in [1] for proving the existence of pyramidal tours.

Proposition 1. Let $\pi = (i_1 = b, i_2, i_3, \dots, i_{n-1}, i_n = e)$ be an arbitrary optimal tour from b to e . If $b > 1$ and $1 = i_k$ with $1 < k < n$ then we have

a) From b to 1 tour π passes through vertices with decreasing indexes:

$$b = i_1 > i_2 > \dots > i_{k-1} > i_k = 1.$$

b) Tour π cannot pass through three successive vertices $i - 1, i, i + 1$ or in inverse direction $i + 1, i, i - 1$ with $1 \leq i - 1$ and $i + 1 \leq b$.

Proof. a) Conversely, suppose that from b to 1 tour π passes through a vertex with index greater than b . Then on the path from b to 1 there is two successive vertices r and s such that $s < b < r$. Also on the path from 1 to n there exists two successive vertices u and v such that $u < b < v$, i.e. tour π is of the form:

$$\pi = (b, \dots, r, s, \dots, 1, \dots, u, v, \dots, n, \dots, e) \text{ with } s < b < r \text{ and } u < b < v.$$

We construct a new tour π' as indicated in Fig. 1:

$$\pi' = (b, \dots, s, \dots, 1, \dots, u, v, \dots, r, \dots, n, \dots, e) \text{ if } v < r \text{ or}$$

$$\pi' = (b, \dots, s, \dots, 1, \dots, u, r, \dots, v, \dots, n, \dots, e) \text{ if } r < v.$$

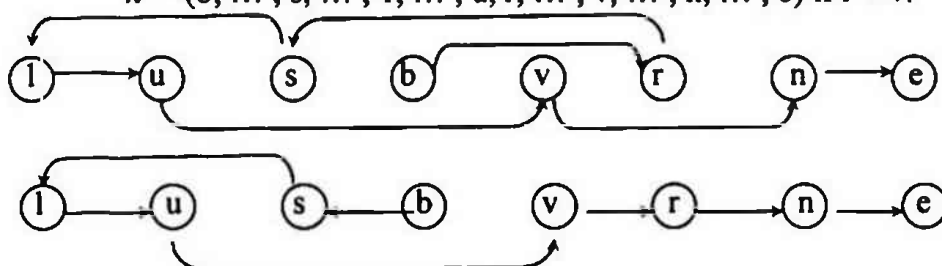


Fig. 1. Optimal tour π and modified tour π' .

It is easily verified that $f(\pi') < f(\pi)$, which contradicts the optimality of π .

b) The proof is in similar way. □

Proposition 2. Let $\pi = (i_1 = b, i_2, i_3, \dots, i_{n-1}, i_n = e)$ be an arbitrary optimal tour from b to e . If $e < n$ and $n = i_r$ with $1 < r < n$ then we have

a) From n to e tour π passes through vertices with decreasing indexes:

$$n = i_r > i_{r+1} > \dots > i_{n-1} > i_n = e.$$

b) Tour π cannot pass through three successive vertices $i - 1, i, i + 1$ or in inverse direction $i + 1, i, i - 1$ with $e \leq i - 1$ and $i + 1 \leq n$.

Proof. The proof is similar to that of Proposition 1. □

Proposition 3. For $n > 3$ an optimal tour of Problem B cannot pass through three successive vertices $i - 1, i, i + 1$ or in inverse direction $i + 1, i, i - 1$ with $1 \leq i - 1 < i + 1 \leq n$.

Proof. Let $\pi = (i_1, i_2, i_3, \dots, i_{n-1}, i_n)$ be an arbitrary optimal tour of Problem B. On the contrary, suppose that π passes through three successive vertices, e.g. $i - 1, i, i + 1$. We shall show a contradiction by constructing a new tour π' such that $f(\pi') < f(\pi)$, which contradicts the optimality of π .

Indeed, let us assume that three successive vertices $i - 1, i, i + 1$ lie on the path from 1 to n . Since $i + 1 \leq n$, on the path from 1 to n there is a vertex r with $i + 1 \leq r \leq n$, whose successor s satisfies $1 \leq s \leq i - 1$. In addition, we have either $i + 1 < r$ or $s < i - 1$, that is

$$\pi = (1, \dots, i - 1, i, i + 1, \dots, n, \dots, r, s, \dots).$$

We construct a new tour π' by going directly from $i - 1$ to $i + 1$ and inserting i between the vertices r and s (see Fig. 2):

$$\pi' = (1, \dots, i - 1, i + 1, \dots, n, \dots, r, i, s, \dots).$$

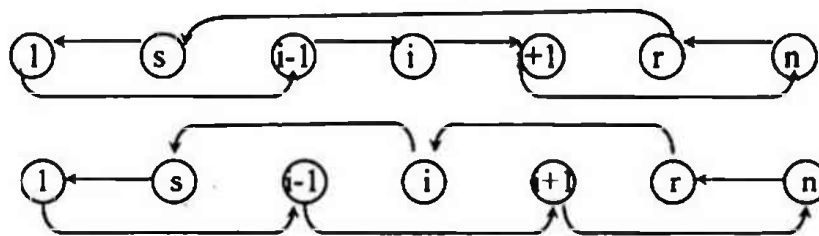


Fig. 2. Optimal tour π and modified tour π' .

Let us denote $U = \{(i-1, i), (i, i+1), (r, s)\}$. Then

$$f(\pi) = \max \{u, \max_{(k,l) \in U} |t_k - t_l|\} \text{ and } f(\pi') = \max \{v, \max_{(k,l) \in U} |t_k - t_l|\}, \tag{3}$$

where

$$u = \max_{(k,l) \in U} |t_k - t_l| = \max \{t_i - t_{i-1}, t_{i+1} - t_i, t_r - t_s\} = t_r - t_s,$$

$$v = \max \{t_{i+1} - t_i, t_r - t_i, t_i - t_s\} < t_r - t_s = u.$$

Hence, from (3) it follows that $f(\pi') < f(\pi)$, which is impossible. □

3. Pyramidal tour and piecewise pyramidal tour

From the above results we can find optimal tours for Problem A and Problem B under condition (1) as follows.

Problem A. The optimal tour π of Problem A consists of three parts, one or two parts of them may be empty depending on the relative location between b, e and n . Namely, $\pi = (\pi_1, \pi_2, \pi_3)$ with

π_1 - optimal subtour from b to $b-1$ passing through all vertices $i: 1 \leq i \leq b$.

π_2 - optimal subtour from $b+1$ to $e-1$ passing through all vertices i with $b+1 \leq i \leq e-1$.

π_3 - optimal subtour from $e+1$ to e passing through all vertices $i: e \leq i \leq n$.

These subtours are of the form:

- If $b = 1$ then $\pi_1 = \emptyset$. If $b > 1$ then

$$\pi_1 = (b, b-2, \dots, 4, 2, 1, 3, \dots, b-3, b-1) \text{ when } b \text{ is even,}$$

that is π_1 passes first through even vertices with indices decreasing from b to 2 , then through odd vertices with indices increasing from 1 to $b-1$, and

$$\pi_1 = (b, b-2, \dots, 3, 1, 2, 4, \dots, b-3, b-1) \text{ when } b \text{ is odd,}$$

i.e. π_1 passes first through odd vertices with indices decreasing from b to 1 , then through even vertices with indices increasing from 2 to $b-1$.

- If $b+1 = e$ then $\pi_2 = \emptyset$. If $b+1 < e$ then

$$\pi_2 = (b+1, b+2, \dots, e-2, e-1),$$

passing through vertices with indices increasing from $b+1$ to $e-1$.

- If $e = n$ then $\pi_3 = \emptyset$. If $e < n$ then

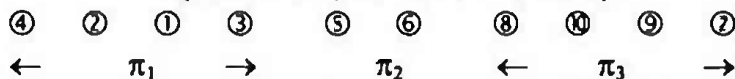
$$\pi_3 = (e+1, e+3, \dots, n-3, n-1, n, n-2, \dots, e+2, e) \text{ when the sum of } e+n \text{ is even,}$$

$$\pi_3 = (e+1, e+3, \dots, n-2, n, n-1, n-3, \dots, e+2, e) \text{ when the sum of } e \text{ and } n \text{ is odd,}$$

i.e. π_3 passes first through even (odd resp.) vertices with indices increasing from $e+1$ to n , then through odd (even resp.) vertices with indices decreasing from n to e .

Example 1. Under condition (1), when $n = 10, b = 4$ and $e = 7$ an optimal tour from 4 to 7 is

$$\pi = (4, 2, 1, 3, 5, 6, 8, 10, 9, 7).$$



Problem B. By Proposition 3 an optimal tour of Problem B cannot pass through three successive vertices $i-1, i, i+1$ or in inverse direction $i+1, i, i-1$ with $1 \leq i-1 < i+1 \leq n$. On the other hand, a tour which has two successive vertices i and j with $|i-j| \geq 3$ cannot be optimal. Consequently an optimal tour must be of form $\textcircled{1} \textcircled{3} \textcircled{5} \dots n \dots \textcircled{6} \textcircled{4} \textcircled{2}$, which is pyramidal.

The optimal value is $f_{\text{opt}} = t_2 - t_1$ if $n = 2$, and $f_{\text{opt}} = \max_{1 \leq i \leq n-2} \{t_{i+2} - t_i\}$ if $n \geq 3$.

Example 2. Here are two optimal tours for Problem B, under condition (1), with even n and odd n :

With $n = 9$: $\textcircled{1} \textcircled{3} \textcircled{5} \textcircled{7} \textcircled{9} \textcircled{8} \textcircled{6} \textcircled{4} \textcircled{2}$

With $n = 10$: $\textcircled{1} \textcircled{3} \textcircled{5} \textcircled{7} \textcircled{9} \textcircled{10} \textcircled{8} \textcircled{6} \textcircled{4} \textcircled{2}$

Remark \diamond As was described, if there are many equal values t_i then we divide the vertices of the graph into classes so that vertices having the same altitude belong to the same class. Each class is considered as a new vertex, called *aggregate vertex*, whose altitude is the altitude of its vertices. Each pair of aggregate vertices is linked by an edge. Applying the above solution we get an optimal tour for the new graph.

To obtain an optimal tour for the initial graph, we extend the aggregate optimal tour by replacing each aggregate vertex by any Hamiltonian path passing through all vertices (each exactly once) in the class corresponding to this aggregate vertex.

\diamond In addition, we emphasize that before solving Problem A or B we have to make sure that the condition (1) is fulfilled (by renumbering the vertices of the graph if necessary). This can be done by using an algorithm for sorting a sequence of real numbers in non-decreasing order. Some well known such algorithms are available (e.g. merge sort, heap sort or quick sort, which require the same time $O(n \log n)$).

To close the paper, we give another example for solving Problem A and B.

Example 3. Find an optimal tour for Problem A and B with the following input data (the source a_4 and the sink a_{11} for Problem A):

vertices	a_1	a_2	a_3	a_4	a_5	a_6	a_7	a_8	a_9	a_{10}	a_{11}	a_{12}	a_{13}	a_{14}	a_{15}	a_{16}	a_{17}
altitude	9	3	8	5	3	8	8	9	1	16	11	4	15	11	4	16	11

We first renumber the vertices in increasing of altitudes. We see that $a_4 \leftrightarrow 6$, $a_{11} \leftrightarrow 12$, so we have $b = 6$ and $e = 12$.

vertices	a_9	a_2	a_5	a_{12}	a_{15}	a_4	a_3	a_6	a_7	a_1	a_8	a_{11}	a_{14}	a_{17}	a_{13}	a_{10}	a_{16}
altitude	1	3	3	4	4	5	8	8	8	9	9	11	11	11	15	16	16
renum.	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17

• An optimal tour from 6 to 12 for Problem A is $\pi = (\pi_1, \pi_2, \pi_3)$ with
 $\underbrace{6 - 4 - 2 - 1 - 3 - 5}_{\pi_1} - \underbrace{7 - 8 - 9 - 10 - 11}_{\pi_2} - \underbrace{13 - 15 - 17 - 16 - 14 - 12}_{\pi_3}$

or in the original vertices is $\pi = (\pi_1, \pi_2, \pi_3)$ with
 $\underbrace{a_4 - a_{12} - a_2 - a_9 - a_5 - a_{15}}_{\pi_1} - \underbrace{a_3 - a_6 - a_7 - a_1 - a_8}_{\pi_2} - \underbrace{a_{14} - a_{13} - a_{16} - a_{10} - a_{17} - a_{11}}_{\pi_3}$

with the optimal function value is $f_{opt} = 5 (= t_{16} - t_{14})$.

• An optimal tour for Problem B is
 $1 - 3 - 5 - 7 - 9 - 11 - 13 - 15 - 17 - 16 - 14 - 12 - 10 - 8 - 6 - 4 - 2 - 1$
 or in the original vertices is

$a_9 - a_5 - a_{15} - a_3 - a_7 - a_8 - a_{14} - a_{13} - a_{16} - a_{10} - a_{17} - a_{11} - a_1 - a_6 - a_4 - a_{12} - a_2 - a_9$
 with the optimal function value is $f_{opt} = 5 (= t_{16} - t_{14})$.

♦ The above methods can also be extended to complete bipartite graphs.

Acknowledgements. This paper is based on the talk given at the Conference on Mathematics, Mechanics, and Informatics, Hanoi, 7/10/2006, on the occasion of 50th Anniversary of Department of Mathematics, Mechanics and Informatics, Vietnam National University Hanoi.

References

[1] B. Korte, J. Vygen, *Combinatorial Optimization: Theory and Algorithms*. Second Edition. Springer, 2002.
 [2] R. E. Burkard, Efficiently Solvable Special Cases of Hard Combinatorial Optimization Problems, *Mathematical Programming* 79 (1997) 55.
 [3] V.G. Deineko, R. Rudolf, J.A.A. van der Veen, G.J. Woeginger, *Three Easy Special Cases of the Euclidian Traveling Salesman Problem*. Bericht Nr. 17, Janner 1995.
 [4] S. Khuller, Y.J. Sussman, W. Gasarch, *Advanced Algorithms, Lectures CMSC 858K*, 1997.