Partial Inverse Traveling Salesman Problems on the Line

Yerim Chung*, Myoung-Ju Park**

*Professor, School of Business, Yonsei University, Seoul, Korea
**Professor, Dept. of Industrial and Management Systems Engineering, Kyung Hee University, Youngin, Korea

[Abstract]

The partial inverse optimization problem is an interesting variant of the inverse optimization problem in which the given instance of an optimization problem need to be modified so that a prescribed partial solution can constitute a part of an optimal solution in the modified instance. In this paper, we consider the traveling salesman problem defined on the line (TSP on the line) which has many applications such as item delivery systems, the collection of objects from storage shelves, and so on. It is worth studying the partial inverse TSP on the line, defined as follows. We are given \( n \) requests on the line, and a sequence of \( k \) requests that need to be served consecutively. Each request has a specific position on the real line and should be served by the server traveling on the line. The task is to modify as little as possible the position vector associated with \( n \) requests so that the prescribed sequence can constitute a part of the optimal solution (minimum Hamiltonian cycle) of TSP on the line. In this paper, we show that the partial inverse TSP on the line and its variant can be solved in polynomial time when the server is equipped with a specific internal algorithm Forward Trip or with a general optimal algorithm.

Key words: Inverse Optimization, Partial Inverse Problem, Traveling Salesman Problem, Poly-time Algorithm, TSP on the Line

[요 약]

부분역최적화는 역최적화의 흥미로운 변형으로, 주어진 최적화문제와 그 문제의 부분해가 주어지면 이 부분해가 최적해에 포함되도록 문제를 최소한으로 수정하는 문제이다. 이 논문은 라인 위에서 정의되는 순환외판원문제(TSP)를 다루는데, 이는 배달시스템, 창고 선반에서 물건을 수집하는 것, 등의 많은 응용을 가진다. 라인 위에서 위치하는 \( n \)개의 일이 주어지고 이 중 연속적으로 처리해야하는 일 \( k \)개가 부분적으로 주어진다. 각각의 일은 라인 위의 특정 장소에 위치하고 라인을 움직이는 서버에 의해 처리되어야 한다. 우리의 임무는 \( k \)개의 일이 최적해에서 연속적으로 처리되도록 \( n \)개의 일의 위치를 라인 위에서 최소한으로 조정하는 것이다. 이 논문에서 이 문제와 이 문제의 다양한 변종을 다양한 내부에 푸는 알고리즘을 개발한다. 구체적으로, 서버가 특정한 Forward Trip이라는 특정한 내부 알고리즘을 사용하는 경우와 일반적인 최적 알고리즘을 사용하는 경우에 대한 부분역최적화를 다룬다.

주제어: 역최적화, 부분역최적화, 순환외판원문제, 다항알고리즘, 라인 위에서의 TSP

*First Author: Yerim Chung, Corresponding Author: Myoung-Ju Park
**Yerim Chung (yerimchung@yonsei.ac.kr), School of Business, Yonsei University
Myoung-Ju Park (pmj0684@khu.ac.kr), Dept. of Industrial and Management Systems Engineering, Kyung Hee University
Received: 2019. 10. 10, Revised: 2019. 11. 05, Accepted: 2019. 11. 05.
Copyright © 2019 The Korea Society of Computer and Information
http://www.ksci.re.kr pISSN:1598-849X | eISSN:2383-9945
I. Introduction

Given a problem, we usually aim to find the optimal solution. However, it is not trivial to find the best solution, especially in the case where the given problem is intrinsically hard to solve. The traveling salesman problem (TSP for short) is the one of such problems.

On the other hand, it is relatively easy to get a feasible solution, even for the hard optimization problems. The inverse optimization problem is defined with the goal of making the prescribed feasible solution optimal in the new instance rather than finding an optimal solution [1-3]. The inverse optimization has been studied for many combinatorial optimization problem such as the shortest path problem, the minimum spanning tree problem, the maximum independent set problem, the minimum graph coloring problem, TSP, and so on [4-13].

The partial inverse optimization problem is an interesting variant of the inverse optimization problem in which a partial solution is given instead of a whole feasible solution. The objective is to modify the least possible the instance of the problem so that the prescribed partial solution becomes a part of the optimal solution in the modified instance. The partial inverse problem has been studied for the assignment problem, minimum cut problem, the shortest path problem, TSP, and so on [14-16].

For TSP, the inverse optimization and the partial inverse optimization problems have been studied by [8, 17]. In [8], the authors show that the inverse TSP against the closest neighbor algorithm can be solved in polynomial time, and the problem against 2-opt is NP-hard when the edge length are restricted to 1 or 2. In [17], the partial inverse TSP is shown to be NP-hard against both Closest Neighbor, and 2-opt.

In this paper, we study the partial inverse traveling salesman problem defined on a special metric space, the real line. While TSP is the one of the most well-known NP-hard problem [18], TSP on the line becomes trivial. For TSP on the line space, the inverse problem is studied by [8] and it is shown that the inverse traveling salesman problem can be solved in $O(n \log n)$ time. However, the partial inverse problem for TSP on the line has not been studied yet.

In practice, TSP on the line (TSPL, for short) has many applications such as robotic welding, item delivery systems, and the collection of objects from storage shelves, and so on. In practice, we may have some partial solution for TSPL. It can be a list of objects that need to be collected in order, or a list of items that need to be delivered consecutively. It is worth studying its partial inverse version where the given partial solution can become a part of an optimal solution for TSPL.

The partial inverse traveling salesman problem on the line is defined as follows and we denote it by $PI(TSPL)$.

We deal with two versions of partial inverse traveling salesman problem on the line. The one is defined with a partial solution of TSPL which is a subsequence of requests that need to be served consecutively. The other is defined with a subset of requests with a specific prescribed order (e.g., precedence conditions). We denote the former by $PI(TSPL)$ and the latter $PI(TSPLP)$.

Definition 1. $PI(TSPL)$.

Given $n$ requests on the horizontal line, and a sequence of $k$ requests, $PI(TSPL)$ is to modify as little as possible the positions of some requests on the line so that the given sequence of $k$ requests can be can be a part of the optimal TSPL-solution.
Definition 2. $PI(TSPL)$. 

Given $n$ requests on the horizontal line, and $k$ requests with precedence conditions in the prescribed order, $PI(TSPL)$ is to modify as little as possible the positions of some requests on the line so that the given $k$ requests can be served in the prescribed order by the server in the optimal $TSPL$-solution, while satisfying the precedence conditions between requests.

This paper is organized as follows: In Section II, we introduce the partial inverse traveling salesman problem on the line against a specific algorithm. In Section III, we discuss the polynomial time solvability of the partial inverse traveling salesman problem on the line against any optimal algorithm. We consider in Section IV the partial inverse traveling salesman problem on the line with precedence conditions, and discuss the relation with the minimum cost flow problem. We then conclude the paper.

II. Partial Inverse Traveling Salesman Problems on the Line against a Specific Algorithm

Let us consider the partial inverse traveling salesman problem on the line against a specific algorithm $A$ for $TSP$, denoted by $PI(TSPL)_A$. In [4], the authors introduced the inverse problem against a specific algorithm. The objective is to determine a new instance at a minimum cost (with minimum modification) such that the given feasible solution can be selected as a best solution when applying the algorithm.

In $PI(TSPL)_A$, we are given $n$ requests dispersed along a horizontal line space, and $k$ requests that need to be served consecutively by the server. The server decides the order to serve the requests by using its specific internal algorithm $A$ in order to minimize the makespan, i.e., the time until all requests have been served. It may often happen that the given $k$ requests cannot be visited in the prescribed sequential order in the optimal tour returned by the algorithm. The objective of $PI(TSPL)_A$ is to modify as little as possible the positions of some requests on the line in such a way that the given sequence $k$ requests can constitute a part of the best solution returned by $A$.

As a specific algorithm for solving $TSP$ on the line space, we consider the greedy algorithm, called Forward Trip ($FT$, for short). $FT$ starts the trip from the leftmost request on the line and satisfies all the requests in its forward trip: no request can be served on the way back. It is clear that $FT$ is an optimal algorithm for $TSP$ on the line ($TSPL$ for short).

Note that an instance of $TSPL$ can be described as a position vector $x = (x_1, \cdots, x_n)$ of $n$ requests on the real line space. Then $PI(TSPL)_{FT}$ can be defined as follows.

Definition. Given a position vector $x = (x_1, \cdots, x_n)$ of $n$ requests on the real line and a sequence $r_1 - r_2 - \cdots - r_k$ of $k$ requests, $PI(TSPL)_{FT}$ is to find a new position vector $y = (y_1, \cdots, y_n)$ such that the given sequence can be a part of the optimal tour returned by $FT$ in the new instance, and the total deviation $\|x - y\|_1$ between the original and new position vectors becomes minimum under the $L_1$-norm.

Focusing on the given $k$ requests that need to be served consecutively (without preemption), we consider the following sets of indices. We denote by $L = \{i \in \{k+1, \cdots, n\} | x_i \leq x_i \}$ the indices of the requests located on the left-hand side with respect to the request $r_1$, and denote by
\[ R = \{ i \in \{ k+1, \ldots, n \} \mid x_i \geq x_k \} \] the indices of the requests originally located on the right-hand side with respect to the request \( r_k \).

Then, one can rewrite \( \text{PIT}(\text{TSPL})_{FT} \) as follows.

\[
\begin{align*}
(P_1) \quad & \min \sum_{i=1}^{n} |x_i - y_i| \\
\text{subject to} \quad & \max_{i \in L} x_i \leq y_1 \leq y_2 \leq \cdots \leq y_k \leq \min_{j \in R} x_j
\end{align*}
\]

Proposition 1. \( \text{PIT}(\text{TSPL})_{FT} \) has an optimal solution such that \( y_i \leq y_j \) if \( x_i \leq x_j \) for any \( i, j \in \{ k+1, \ldots, n \} \).

Proof. Let \( y = (y_1, \ldots, y_n) \) be an optimal solution of \( \text{PIT}(\text{TSPL})_{FT} \). Assume that for some \( i, j \in \{ k+1, \ldots, n \} \), we have \( x_i \leq x_j \) and \( y_i \geq y_j \). Now we consider another feasible solution \( y' \) such that \( y'_i = y_j, y'_j = y_i, \) and \( y'_k = y_k \) for any \( k \neq i, j \). It is easy to see that \( y' \) is at least as good as the given optimal solution \( y \). By repeating this, we can obtain an optimal solution for \( \text{PIT}(\text{TSPL})_{FT} \) such that \( y_i \leq y_j \) if \( x_i \leq x_j \) for any \( i, j > k \).

Due to Proposition 1, one can assume without loss of generality that \( x_{k+1} \leq x_{k+2} \leq \cdots \leq x_n \). Then, we can rewrite \( \text{PIT}(\text{TSPL})_{FT} \) as the problem of solving the following \( n-k+1 \) subproblems:

\[
\begin{align*}
(P_2) \quad & \min \sum_{i=1}^{n} |x_i - y_i| \\
\text{subject to} \quad & y_1 \leq \cdots \leq y_k \leq y_{k+1} \leq \cdots \leq y_n \\
& \text{or} \\
& y_{k+1} \leq y_1 \leq \cdots \leq y_k \leq y_{k+2} \leq \cdots \leq y_n \\
& \text{or} \\
& y_{k+1} \leq y_{k+2} \leq y_1 \leq \cdots \leq y_k \leq y_{k+3} \leq \cdots \leq y_n \\
& \text{or} \\
& \cdots
\end{align*}
\]

It is straightforward to see that each of these subproblems is a special case of the isotonic median regression problem[19-21]. In [20], the PAV-algorithm (Pool Adjacent Violaters) is proposed for solving the isotonic median regression problem, and this algorithm solves each subproblem of \( \text{PIT}(\text{TSPL})_{FT} \) in time \( O(n \log n) \) in the worst case. This leads to the following result.

Proposition 2. \( \text{PIT}(\text{TSPL})_{FT} \) can be solved in polynomial time.

Proof. One can solve \( \text{PIT}(\text{TSPL})_{FT} \) by applying the PAV algorithm to the \( n-k+1 \) problems, and the new position vector (the optimal solution) which has the minimum deviation value.

III. Partial Inverse Traveling Salesman Problems on the Line against Any Optimal Algorithm

In this section, we consider the partial inverse traveling salesman problem on the line against any general optimal algorithm rather than a specific algorithm for \( \text{TSPL} \). We denote this problem by \( \text{PIT}(\text{TSPL})_{OPT} \).

In \( \text{PIT}(\text{TSPL})_{OPT} \), the requests on the line can be served during both forward and backward trips, while in \( \text{PIT}(\text{TSPL})_{FT} \), the requests can be served only on the forward trip. It is easy to see that the result of Proposition 1 still holds for \( \text{PIT}(\text{TSPL})_{OPT} \).
We assume that the server starts the trip by visiting the request \( r_1 \) and finishes the trip by returning to that request. Let \( r_s \) and \( r_l \) be the leftmost and rightmost request on the line with \( 1 \leq s, l \leq n \).

Then, one can observe that any instance of \( PI(TSPL)_{OPT} \) admits an optimal Hamiltonian cycle \( r_1 - r_2 - \cdots - r_n - r_1 \) if the associated position vector \( x = (x_1, \cdots, x_n) \) satisfies the following conditions (a) and (b):

(a) if \( 1 \leq s < t \leq n \),
\[
x_s \leq x_{s+1} \leq \cdots \leq x_t
\]
and
\[
x_s \leq x_{s-1} \leq \cdots \leq x_1 \leq x_n \leq x_{n-1} \leq \cdots \leq x_{t+1} \leq x_t
\]

(b) if \( 1 \leq t < s \leq n \),
\[
x_s \leq x_{s+1} \leq \cdots \leq x_t
\]
and
\[
x_s \leq x_{s-1} \leq \cdots \leq x_2 \leq x_1 \leq x_2 \leq x_{t-1} \leq x_t
\]

Based on these results, we devise an optimal algorithm for \( PI(TSPL)_{OPT} \), using the PAV-algorithm for solving the special case of the isotonic median regression.

When an original position vector \( x = (x_1, \cdots, x_n) \) and a sequence \( r_1 - r_2 - \cdots - r_k \) of \( k \) requests in the increasing order are given, our algorithm computes a new vector \( y_{(i_0,j_0)} = (y_1, \cdots, y_{i_0}, \cdots, y_{j_0}, \cdots, y_n) \) for each \( (i_0,j_0) \) with \( 1 \leq i_0, j_0 \leq k \) that minimizes the quantity \( \sum_{i=1}^{n} |x_i - y_i| \), while satisfying the following:

(i) \( \forall i \in \{k+1, \cdots, n\}, y_i = x_i \),

(ii) By using the PAV-algorithm, determine \( y_i \) \( \forall i \in \{1, \cdots, k\} \) by solving the following problems.

In the case \( 1 \leq i_0 < j_0 \leq k \),

\[
(P_3) \quad \min \sum_{i=1}^{k} |x_i - y_i|
\]

subject to
\[
y_{i_0} \leq y_{i_0+1} \leq \cdots \leq y_{j_0},
y_{i_0} \leq y_{i_0-1} \leq \cdots \leq y_{i_1} \leq y_{i_1} \leq y_{j_1} \leq \cdots \leq y_{j_0}
\]

In the case \( 1 \leq j_0 < i_0 \leq k \),

\[
(P_4) \quad \min \sum_{i=1}^{k} |x_i - y_i|
\]

subject to
\[
y_{i_0} \leq y_{i_0-1} \leq \cdots \leq y_{j_0},
y_{i_0} \leq y_{i_0+1} \leq \cdots \leq y_{j_0}, \text{ and}
y_{j_0} \leq y_{j_0+1} \leq \cdots \leq y_{i_0} \leq y_{i_0}
\]

For each \( (i_0, j_0) \) with \( 1 \leq i_0, j_0 \leq k \), the PAV-algorithm [16] solve this problem in time \( O(k \log k) \). Hence, our algorithm computes the best solution for \( PI(TSPL)_{OPT} \) with the minimum deviation with respect to the \( L_1 \)-norm in time \( O(k^3 \log k) \).

**Proposition 3.** The overall computational complexity for \( PI(TSPL)_{OPT} \) is \( O(k^3 \log k) \).

**IV. Partial Inverse Traveling Salesman Problems on the Line with Precedence Constraints**

In the previous sections, we have considered the partial inverse traveling salesman problem on the line, defined with the prescribed partial solution. In this section, we consider an interesting variant of \( PI(TSPL) \), where instead of a sequence of requests, the precedence conditions \( r_i < r_j \) are imposed on some requests. We mean by \( r_i < r_j \) that the request \( r_i \) needs to be served before \( r_j \), however the other requests can be served between \( r_i \) and \( r_j \) by the server. We assume that the precedence conditions are imposed on \( k \) requests, i.e., \( r_1 < r_2 < \cdots < r_k \). We call this problem the
partial inverse traveling salesman problem on the line with the precedence conditions, and denote it by $PI(TSPLP)_A$.

Let us consider $PI(TSPLP)_A$ against $FT$. Let $\Omega = \{(i,j) | r_i < r_j\}$ be the set of the pairs of requests with precedence conditions. Then, the problem $PI(TSPLP)_{FT}$ can be formulated as follows:

\[(P_5) \quad \min \sum_{i=1}^{n} |x_i - y_i| \]
subject to \[-y_i + y_j \geq 0, \ \forall \ (i,j) \in \Omega\]

By linearization, we obtain the following model.

\[(P_6) \quad \min \sum_{i=1}^{n} (d_i^+ - d_i^-) \]
subject to \[y_i + d_i^+ - d_i^- = x_i, \ \forall \ i = 1, \cdots, n \]
\[-y_i + y_j \geq 0, \ \forall \ (i,j) \in \Omega \]
\[d_i^+, d_i^- \geq 0, \ \forall \ i = 1, \cdots, n \]

Let $\mu_i$ and $\lambda_{i,j}$ be the dual variables corresponding to the first and second constraints of the problem $(P_2)$, respectively. Then, the dual of the problem $(P_2)$ can be written as follows.

\[(P_7) \quad \max \sum_{i=1}^{n} x_i \mu_i \]
subject to \[\sum_{j \in (i,j) \in \Omega} \lambda_{i,j} - \sum_{j \in (j,i) \in \Omega} \lambda_{j,i} = \mu_i, \ \forall \ i = 1, \cdots, n \]
\[-1 \leq \mu_i \leq 1, \ \forall \ i = 1, \cdots, n \]
\[\lambda_{i,j} \geq 0, \ \forall \ (i,j) \in \Omega \]

or equivalently,

\[(P_8) \quad \min - \sum_{i=1}^{n} x_i \mu_i \]
subject to \[\sum_{j \in (i,j) \in \Omega} \lambda_{i,j} - \sum_{j \in (j,i) \in \Omega} \lambda_{j,i} = \mu_i, \ \forall \ i = 1, \cdots, n \]
\[-1 \leq \mu_i \leq 1, \ \forall \ i = 1, \cdots, n \]
\[\lambda_{i,j} \geq 0, \ \forall \ (i,j) \in \Omega \]

Using this formulation, we will show that $PI(TSPLP)_{FT}$ can be reduced to the minimum cost flow problem ($MCF$, for short). We construct an instance of $MCF$ as follows:

- **Graph structure:**
  \[G = (V, E), \ \text{with} \ V = \{1, \cdots, n\} \cup \{s, t\} \ \text{and} \ E = \Omega \cup \{(s, i), (i, t) | i = 1, \cdots, n\} \cup \{(s, t)\}.\]

- **Arc capacity:**
  \[u_{i,j} = \infty, \ \text{if} \ \forall \ (i,j) \in \Omega, \]
  \[u_{s,i} = u_{i,t} = 1, \ \forall \ i = 1, \cdots, n \]
  and \[u_{s,t} = n\]

- **Arc cost:**
  \[c_{i,j} = 0, \ \text{if} \ \forall \ (i,j) \in \Omega, \]
  \[c_{s,i} = -x_i, \ c_{i,t} = x_i, \ \forall \ i = 1, \cdots, n \]
  and \[c_{s,t} = 0\]

- **Node balance:**
  \[b(s) = n, b(t) = -n \]
  and \[b(i) = 0, \ \forall \ i = 1, \cdots, n\]

Let be \(f_{i,j}\) the flow on each arc \((i,j) \in \Omega\).

**Proposition 4.** $PI(TSPLP)_{FT}$ can be reduced to the minimum cost flow problem and thus can be solved in polynomial time.

**Proof.** Given an instance $(P_1)$ of $PI(TSPLP)_{FT}$, we have just constructed an instance of $MCF$. It remains to us to show that a feasible solution of $PI(TSPLP)_{FT}$ is feasible for the constructed
instance of $MCF$, with the same objective value, and vice versa.

Given a feasible solution $(\lambda_{i,j}, \mu_i)$ of $(P_4)$, we can find a feasible solution for $MCF$ by setting $f_{i,j} = \lambda_{i,j} \forall (i,j) \in \Omega$, $f_{s,i} = \mu_i, f_{i,t} = 0$, if $\mu_i \geq 0$, and $f_{s,i} = 0, f_{i,t} = -\mu_i$ if $\mu_i \leq 0$, and by setting $f_{s,t} = n - \sum_{i=1}^{n} \max\{0, \mu_i\}$. Clearly, this is a feasible solution for $MCF$.

We can also find a feasible solution $(\lambda_{i,j}, \mu_i)$ for $\text{PI}(\text{TSPL})_{FT}$ using a feasible solution $f_{i,j}$ for $MCF$ by setting $\lambda_{i,j} = f_{i,j} \forall (i,j) \in \Omega$ and $\mu_i = f_{s,i} - f_{i,t}$ for any $i = 1, \ldots, n$.

Hence, $\text{PI}(\text{TSPL})_{FT}$ can be reduced to the minimum cost flow problem and can be solved in polynomial time.

\section{V. Conclusions}

In this paper, we have studied the partial inverse traveling salesman problem on the line against any optimal algorithm as well as against a specific algorithm for $\text{TSPL}$. When a sequence of some requests is given as a partial solution, $\text{PI}(\text{TSPL})_{FT}$ and $\text{PI}(\text{TSPL})_{OPT}$ turn out to be polynomially solvable by using the PAV-algorithm devised for the isotonic median regression problem. When precedence conditions are given instead of a sequence of requests, we show that $\text{PI}(\text{TSPL})_{FT}$ is reducible to the minimum cost flow problem, one of the most well-known network optimization problem. The obtained results are summarized in the following table.

<table>
<thead>
<tr>
<th>Problems</th>
<th>Complexity Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\text{PI}(\text{TSPL})_{FT}$</td>
<td>poly-time solvable</td>
</tr>
<tr>
<td>$\text{PI}(\text{TSPL})_{OPT}$</td>
<td>poly-time solvable</td>
</tr>
<tr>
<td>$\text{PI}(\text{TSPL})_{FT}$</td>
<td>poly-time solvable</td>
</tr>
<tr>
<td>$\text{PI}(\text{TSPL})_{OPT}$</td>
<td>unknown</td>
</tr>
</tbody>
</table>

Our paper contributes to the inverse optimization literature by investigating the computational complexity of the partial inverse traveling salesman problem on the line, which has not yet been studied to the best of our knowledge.

This study has also some limitations. First, the complexity status for $\text{PI}(\text{TSPL})_{OPT}$ has not been investigated yet. Second, even though the algorithms we proposed for solving $\text{PI}(\text{TSPL})_{FT}$ and $\text{PI}(\text{TSPL})_{OPT}$ are theoretically correct, they have not yet been used for solving the real examples. The complexity status for $\text{PI}(\text{TSPL})_{OPT}$ and the computation part remain as a future work. In addition, it would be an interesting future research to consider the partial inverse traveling salesman problem on some other metric space such as trees and lattices.

\section{REFERENCES}


Authors

Yerin Chung received the B.S. degree in Business Administration from Yonsei University, Korea, in 2000. She received the M.S. and Ph.D. degree in Applied Mathematics and Computer Science from Paris 1 University, France, in 2004 and 2010, respectively. Dr. Chung joined the faculty of Business School at Yonsei University, Seoul, Korea, in 2011. She is interested in inverse optimization and network optimization, and their many application problems.

Myoung-Ju Park received his PhD in Industrial Engineering from Seoul National University in 2012. He is an associate Professor in the Department of Industrial and Management Systems Engineering, Kyung Hee University, Yongin, Korea. He is interested in combinatorial optimization, approximation algorithms, and scheduling.