Discrete Optimization 14 (2014) 147–159

Contents lists available at ScienceDirect

Discrete Optimization

journal homepage: www.elsevier.com/locate/disopt

Four-point conditions for the TSP: The complete complexity classification

Vladimir G. Deineko [a,*], Bettina Klinz [b], Alexander Tiskin [c], Gerhard J. Woeginger [d]

[a] *Warwick Business School, The University of Warwick, Coventry CV4 7AL, United Kingdom*

[b] *Institut für Optimierung und Diskrete Mathematik, TU Graz, Steyrergasse 30, A-8010 Graz, Austria*

[c] *Department of Computer Science, The University of Warwick, Coventry CV47AL, United Kingdom*

[d] *Department of Mathematics and Computer Science, TU Eindhoven, P.O. Box 513, 5600 MB Eindhoven, Netherlands*

ARTICLE INFO

Article history:
Received 14 November 2013
Received in revised form 6 September 2014
Accepted 12 September 2014
Available online 4 October 2014

Keywords:
Traveling salesman problem
Four-point condition
Polynomially solvable case

ABSTRACT

The combinatorial optimization literature contains a multitude of polynomially solvable special cases of the traveling salesman problem (TSP) which result from imposing certain combinatorial restrictions on the underlying distance matrices. Many of these special cases have the form of so-called four-point conditions: inequalities that involve the distances between four arbitrary cities.

In this paper we classify *all possible* four-point conditions for the TSP with respect to computational complexity, and we determine for each of them whether the resulting special case of the TSP can be solved in polynomial time or whether it remains NP-hard.

© 2014 Elsevier B.V. All rights reserved.

1. Introduction

The traveling salesman problem (TSP) is a classical problem of combinatorial optimization. In the TSP, one is given a $n \times n$ distance matrix $C = (c_{ij})$ and looks for a cyclic permutation τ of the set $\{1, 2, \ldots, n\}$ that minimizes the function $c(\tau) = \sum_{i=1}^{n} c_{i\tau(i)}$. The value $c(\tau)$ is called the *length* of the permutation τ. The items in τ are usually called *cities* or *points* or *nodes*. In this paper we are only interested in the *symmetric* TSP, where $c_{ij} = c_{ji}$ holds for all i, j.

The TSP in its general form is NP-hard [1]. However, the optimization literature contains many highly-structured special cases that can be solved in polynomial time; see for instance [2–4] for surveys of such efficiently solvable cases of the TSP. Many well-known efficiently solvable cases of the TSP result from imposing conditions on the underlying distance matrix. An important subclass of these conditions is formed by the so-called *four-point* conditions. Consider four points i, j, k, l with $1 \le i < j < k < l \le n$. The (symmetric!) distance matrix contains six different entries for these four points, which correspond to the six edges connecting the points. There are three pairings of these points into two disjoint edges $\{(i, j), (k, l)\}$, $\{(i, k), (j, l)\}$, $\{(i, l), (j, k)\}$, and we denote the lengths of these pairs as follows:

$$\mathcal{A} = c_{ij} + c_{kl}, \qquad \mathcal{B} = c_{ik} + c_{jl}, \qquad \mathcal{C} = c_{il} + c_{jk}.$$

Now a *four-point* condition simply specifies one or two inequalities which rank the values $\mathcal{A}, \mathcal{B}, \mathcal{C}$; these inequalities have to be satisfied for all possible choices of i, j, k, l with $1 \le i < j < k < l \le n$.

* Corresponding author. Tel.: +44 2476 524 501; fax: +44 2476 524539.
E-mail addresses: Vladimir.Deineko@wbs.ac.uk (V.G. Deineko), klinz@opt.math.tu-graz.ac.at (B. Klinz), tiskin@dcs.warwick.ac.uk (A. Tiskin), gwoegi@win.tue.nl (G.J. Woeginger).

http://dx.doi.org/10.1016/j.disopt.2014.09.003
1572-5286/© 2014 Elsevier B.V. All rights reserved.

Let us illustrate these notions with two concrete examples. The story of four-point conditions for the TSP started more than fifty years ago, when Fred Supnick [5] investigated the following special case of the TSP. Supnick assumed that the distance matrix (c_{ij}) satisfies for all i, j, k, l with $1 \le i < j < k < l \le n$ the conditions

$$c_{ij} + c_{kl} \le c_{ik} + c_{jl} \quad \text{and} \quad c_{ik} + c_{jl} \le c_{il} + c_{jk}.$$

The results of Supnick yield a polynomial time solution for his special case. In our language, such a *Supnick distance matrix* is specified by the four-point conditions $\mathcal{A} \le \mathcal{B}$ and $\mathcal{B} \le \mathcal{C}$.

The second example concerns the work of Kalmanson [6] who back in 1975 considered the so-called quadrangle inequalities

$$c_{ij} + c_{kl} \le c_{ik} + c_{jl} \quad \text{and} \quad c_{ik} + c_{jl} \ge c_{il} + c_{jk}$$

for all i, j, k, l with $1 \le i < j < k < l \le n$. These inequalities capture certain properties of convex quadrangles, and essentially state that in a convex quadrangle the total length of the two diagonals is always greater or equal to the total length of two opposing sides. In our language, *Kalmanson's* conditions are stated as the four-point conditions $\mathcal{A} \le \mathcal{B}$, $\mathcal{B} \ge \mathcal{C}$.

Specially structured matrices introduced above are related to some other optimization problems. Many of these problems involve the so-called Monge matrices. An $n \times n$ matrix C is called a *Monge matrix* if

$$c_{ij} + c_{rs} \le c_{is} + c_{rj} \quad \text{for all } 1 \le i < r \le n,\ 1 \le j < s \le n.$$

The main diagonal elements are involved in the definition above, while they are irrelevant to the TSP. Notice that no diagonal elements are involved in the definitions of Supnick and Kalmanson matrices. It can be shown (see [2, Proposition 2.13, p. 508]) that for example in a Supnick matrix, the diagonal elements can always be specified so that the matrix becomes a special case of the Monge matrix. For more information on the importance of Monge matrices in discrete optimization, the reader is referred to the surveys [7,8] and some recent publications [9–14].

Kalmanson matrices, although not as popular as Monge matrices, are also well known to the Operational Research community. If involved, Kalmanson matrices allow one to find in polynomial time solutions to such problems as the prize-collecting TSP [15], the master tour problem [16], the Steiner tree problem [17], the three-dimensional matching problem [18], and the quadratic assignment problem [19,20].

Note that both Supnick matrices and Kalmanson matrices satisfy the condition $\mathcal{A} \le \mathcal{B}$. In general, a distance matrix satisfying this condition is known as a *Demidenko* matrix [21]. As first shown in [21], an optimal tour for the Demidenko TSP can be found in the set of pyramidal tours. A tour $\tau = \langle 1, i_1, i_2, \ldots, i_r, n, j_1, j_2, \ldots, j_{n-r-2}, 1\rangle$ is called *pyramidal*, if $i_1 < i_2 < \cdots < i_r$ and $j_1 > j_2 > \cdots > j_{n-r-2}$. Although the set of pyramidal tours contains an exponential number 2^{n-2} of tours, an optimal pyramidal tour can be computed through dynamic programming in $O(n^2)$ time; see [21] and also [2,3]. The set of pyramidal tours was probably one of the first well-known exponential neighborhoods in combinatorial optimization that could be searched in polynomial time. Studies of other exponential neighborhoods have been extensively reported in the literature for various combinatorial optimization problems; see [22–26], and the surveys [27,28].

Contribution of this paper: in this paper, we classify *all possible* four-point conditions for the TSP with respect to their computational complexity. We determine for each of them whether the resulting special case of the TSP can be solved in polynomial time or whether it remains NP-hard. Parts of our work have been reported in conference papers by Deineko [29], Deineko, Klinz and Woeginger [30] and Deineko and Tiskin [31].

Technical description of our approach: the typical approach to show that a certain set of four-point conditions yields a polynomially solvable special case of the TSP is as follows. As a first step, one identifies an appropriate *neighborhood* $\mathcal{S}$ which is a subset of highly-structured permutations. In the second step, one proves that the neighborhood always contains an optimal tour. And in the third step, one shows that optimization over $\mathcal{S}$ is easy and can be done in polynomial time.

The second step (showing that $\mathcal{S}$ contains an optimal tour) is usually done by the so-called *tour improvement technique*, and the underlying idea is as follows. One starts from an arbitrary tour τ, and constructs a corresponding sequence $\tau = \tau_1, \tau_2, \ldots, \tau_T$ of tours such that

$$c(\tau_1) \ge c(\tau_2) \ge \cdots \ge c(\tau_T).$$

The final tour τ_T lies in the neighborhood $\mathcal{S}$, and the four-point conditions are used to establish the inequalities $c(\tau_t) \ge c(\tau_{t+1})$ for $t = 1, \ldots, T$. In some lucky cases, the neighborhood contains only a polynomial number of tours (or even consists of a single tour), and then the third step is trivial.

For example, the tour improvement technique can be used to show that for the Supnick TSP [5] an optimal tour is given by $\sigma_{\text{Smin}} = \langle 1, 3, 5, 7, 9, \ldots, 8, 6, 4, 2, 1\rangle$; in other words the optimal solution first traverses the odd numbers in increasing order and then the even numbers in decreasing order. For the Kalmanson TSP [6], the tour improvement technique shows that an optimal tour is given by the identity permutation $\tau_{\text{Kmin}} = \langle 1, 2, 3, \ldots, n-1, n, 1\rangle$; see also [2].

Notation: the set of all permutations of $\{1, 2, \ldots, n\}$ is denoted by S_n. For $\tau \in S_n$, we denote by τ^{-1} the inverse of τ; hence $\tau^{-1}(i)$ is the pre-image of i in permutation τ, for $i = 1, \ldots, n$. For $k > 1$, we define $\tau^k(i)$ recursively as $\tau(\tau^{(k-1)}(i))$, and $\tau^{-k}(i)$ as $\tau^{-1}(\tau^{-(k-1)}(i))$. We also use a cyclic representation of a cyclic permutation of the form $\tau = \langle i, \tau(i), \tau^2(i), \ldots, \tau^{-2}(i),$

Download English Version:

https://daneshyari.com/en/article/1141739

Download Persian Version:

https://daneshyari.com/article/1141739

Daneshyari.com