

# Complexity of the Unconstrained Traveling Tournament Problem

Rishiraj Bhattacharyya

Indian Statistical Institute, Kolkata

rishiraj.bhattacharyya@gmail.com

## Abstract

The Traveling Tournament problem is a problem of scheduling round robin leagues which minimizes the total travel distance maintaining some constraints on consecutive home and away matches. The problem was proven NP-hard when the upper bound on any consecutive home or away stint is 3. In this paper, we prove that even without the constraints on the consecutive home or away matches, the problem remains NP-Hard.

**Keywords:** Scheduling, Traveling Tournament Problem, NP-Hard.

## 1 Introduction

The *Traveling Tournament Problem* (TTP) addresses the problem of minimizing the total travel distance in a double round robin league tournament schedule where each team can play at most  $U$  consecutive home and away matches [5]. Since its proposal in [5], TTP has been a notoriously difficult problem to solve. Numerous works using metaheuristic techniques [1, 2, 6] have been conducted to solve TTP in the past years. We refer the reader to [14, 11] for detail survey on these techniques. The complexity of TTP was also open for many years. Finally, in 2011, Thielen and Westphal proved that the Traveling Tournament problem with  $U = 3$  is NP-hard [15].

Given the hardness of the constrained TTP, it is natural to explore a simplified problem. One natural is TTP without any constraint on the consecutive home and away matches. Formally, the Unconstrained Traveling Tournament Problem (UTTP) can be defined as follows:

- **[Input]:**
  - The number of teams  $n$  ( $n$  is even).
  - A distance matrix  $D_{n \times n}$ .
- **[Output]:** A double round robin league schedule of  $n$  teams such that the total distance traveled by all the teams is minimized.

Recently, Imahori, Matsui, and Miyashiro have proposed a 2.75-approximation algorithm for the Unconstrained TTP [10].

## Our Result

In this paper, we prove that the Unconstrained Traveling Tournament Problem is NP-hard. Specifically, we show that if the teams are allowed to play any number of consecutive home and away matches, then there is a reduction from the  $(1, 2)$ -Traveling Salesman Problem; the problem of finding a TSP tour in a graph where the cost of every edge is either 1 or 2.

**Theorem 1.1** *The Unconstrained Traveling Tournament problem is NP-Hard.*

The rest of the paper is organized as follows. In the next section, we introduce the notations used in the paper and formally define the decision problems. In Section 3 we describe the constructions and the associated results needed for the reduction. Finally in Section 4 we describe the formal reduction from  $(1, 2)$ -TSP to UTTP.

## 2 Notation and Preliminaries

Throughout the paper we follow the following notation. If  $S$  is a set  $|S|$  denotes the cardinality of  $S$ .  $[n]$  denotes the set of first  $n$  natural numbers.  $\mathbb{Z}_n$  denotes the set  $\{0, 1, \dots, n-1\}$ .  $G = (V, E)$  is a complete weighted graph without self-loops or parallel edges.  $V$  is the set of vertices and  $E$  is the set of edges.

We recall the definitions and well known results on round robin tournaments.

**Definition 2.1 Single Round Robin Schedule** *Let  $z \in \mathbb{N}$  be an even integer. A tournament schedule on  $z$  teams is called a single round robin schedule, if each team plays with every other team once.*

We denote a schedule by a set of matches. Each match is represented by a 3-tuple,  $(a, b, r)$  where  $a$  plays at home against  $b$  in round  $r$ . The following theorem is folklore.

**Theorem 2.2**  $\forall z \in \mathbb{N}$ , there is a single round robin tournament of  $2z$  teams.

**Definition 2.3 Double Round Robin Schedule** *Let  $n \in \mathbb{N}$  be an even integer. A tournament schedule on  $n$  teams is called a double round robin league schedule, if each team plays every other team twice; once at home and once away.*

One can design a schedule of a double round robin tournament by repeating a single round robin tournament schedule with home-away reversed.

### Unconstrained Traveling Tournament Problem

We formulate the decision version of the Unconstrained Traveling Tournament Problem as follows:

- **Problem:** Traveling Tournament Problem
- **Instance:** An even integer  $n$ ; An  $n \times n$  matrix  $D$ ; An integer  $K$
- **Question:** Is there a double round robin league schedule of  $n$  teams located at venues represented by  $D$ , with total distance traveled by all the teams is at most  $K$ ?

We represent an instance of this problem by  $TTP(n, D, K)$ . Without loss of generality we can assume  $D$  is a metric as any real life distance matrix will be a metric. To show that  $UTTP$  is  $NP$ -hard, we show a reduction from a variant of Traveling Salesman Problem; called (1, 2)-Traveling Salesman Problem ((1, 2)-TSP). Then decision version of (1, 2)-TSP is defined as follows

- **Problem:** (1, 2)- Traveling Salesman Problem
- **Instance:** A set  $C = \{c_1, c_2, \dots, c_n\}$  of  $n$  cities; Distance  $d(c_i, c_j) \in \{1, 2\}$  for all pair  $c_i, c_j \in C, i \neq j$ ; An integer  $K$ .
- **Question:** Is there a cycle in  $C$  with cost at most  $K$ ? In other terms, is there a permutation  $\pi : [n] \rightarrow [n]$  such that

$$\sum_{i=1}^{n-1} d(c_{\pi(i)}, c_{\pi(i+1)}) + d(c_{\pi(n)}, c_{\pi(1)}) \leq K?$$

It is easy to prove the NP-hardness of the (1,2)-TSP using a reduction from the Hamiltonian Cycle problem. We refer the reader [8] for a detailed proof.

### 3 Construction of UTTP instance

We shall prove a reduction from (1,2)-TSP to the unconstrained TTP. The reduction works in two parts. In the first part, we construct a modified instance of the metric TSP problem from the input instance. Next, we construct a UTTP instance based on the modified instance.

#### Modification of the TSP Instance

Consider an input instance of (1,2)-TSP on  $n$  cities, a complete weighted graph  $G = (V, E)$  with  $|V| = n$ .  $d(v_i, v_j) \in \{1, 2\}$  for all  $i, j \in [n]$ . Our objective is to construct a graph  $G'$  from  $G$  such that the optimal traveling salesman tour of  $G$  has cost  $K$  iff Traveling Salesman tour of  $G'$  is of cost  $nK$ . We construct such a graph using the well known windmill graph.

A windmill graph  $K_n^{(l)}$  is the graph consisting of  $l$  copies of the complete graph  $K_n$  with a vertex in common across the copies. We construct an extended windmill graph which is a complete graph on  $l(n-1) + 1$  vertices, and assign weights based on whether the endpoints belong to same copy of or not.

**Definition 3.1** Let  $G = (V, E)$  be a complete weighted graph on  $n$  vertices ( $n > 2$ ). Let  $d(i, j)$  denote the weight of each edge  $(i, j) \in E$ . The extended windmill graph  $W_n^l[G]$  is a weighted complete graph on  $l(n-1) + 1$  vertices constructed from  $G$  in the following way

- Construct the windmill graph  $K_n^l$  from  $G$  taking  $l$  copies of  $V$  with  $v$  as the shared vertex. Let  $v_i^k$  denote the vertex  $v_i$  of the  $k^{th}$  copy.
- For all  $k \in [l]$ ,  $d'(v_i^k, v) = d(v_i, v)$ .
- For all  $k \in [l]$ ,  $d'(v_i^k, v_j^k) = d(v_i, v_j)$ .

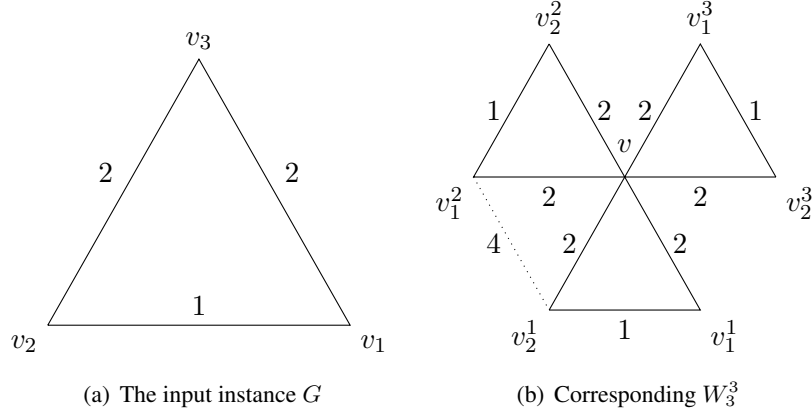


Figure 1: (a): The input instance of TSP, the graph  $G$ , (b): Construction of  $W_3^3[G]$

- $d'(v_i^{k_1}, v_j^{k_2}) = d(v_i, v) + d(v, v_j)$  where  $k_1 \neq k_2$ .

The vertex  $v$  is called the central vertex.

We note that the cost of each edge in  $W_n^l[G]$  is at most 4.

In Figure 1(b), we show the construction of  $W_3^3[G]$  from the input instance of Figure 1(a). We took three copies of  $G$  and contracted all the copies of  $v_3$  to one vertex  $v$ . We showed only one edge between different copies to keep the image neat.

**Proposition 3.2** *Let  $G = (V, E)$  be a complete weighted graph on  $n$  vertices. Let  $W_n^l[G]$  be the extended windmill graph. If triangle inequality holds in  $G$ , triangle inequality holds in  $W_n^l[G]$ .*

The proof of the triangle inequality for three arbitrary vertices  $a, b, c \in W_n^l[G]$  follows by an elementary case distinction into the three cases where all the three vertices are in one copy of  $G$ , in two different copies of  $G$ , or in three different copies of  $G$ .

**Lemma 3.3**  *$W_n^l[G]$  has a traveling salesman tour of cost at most  $lK$  iff  $G$  has a traveling salesman tour of cost at most  $K$ .*

**Proof.** *If part*

Suppose,  $\tau = \{v_1, v_2, \dots, v_n, v_1\}$  be a traveling salesman tour of the graph  $G$  of cost  $K$ . Construct the corresponding  $W_n^l[G]$  with  $v_1$  as the central vertex. The tour  $(v_1, v_2^1, \dots, v_n^1, v_2^2, \dots, v_n^2, \dots, v_n^l, v_1)$  admits the cost  $lK$ .

*Only if part*

We need to prove that, if there is a traveling salesman tour of  $W_n^l[G]$  of cost  $lK$ ,  $G$  has a traveling salesman tour of cost at most  $K$ . We prove it by induction on  $l$ .

**Case  $l = 1$ :** Obvious.

Let, the statement hold for the case  $l = t$  for some  $t \in \mathbb{N}$ .

**Case  $l = t + 1$ :** Recall the iterative construction of  $W_n^{t+1}[G]$ . First we construct  $W_n^t[G]$  with  $v$  as the

central vertex. We add  $(t + 1)^{th}$  copy of  $G$  with the vertex  $v$  shared with  $W_n^t[G]$ . In the rest of the proof,  $\overline{G}$  represents the  $(t + 1)^{th}$  copy of  $G$ .

**Proposition 3.4** *If  $W_n^{t+1}[G]$  has a traveling salesman tour of cost  $(t + 1)K$ , then either there exists a closed walk, visiting all the vertices in  $\overline{G}$  of cost at most  $K$  or there exists a closed walk, visiting all the vertices in  $W_n^t[G]$  of cost at most  $tK$ .*

The following proposition is an obvious fact for any graph with edges satisfying triangle inequality.

**Proposition 3.5** *Let  $G$  be a weighted complete graph where triangle inequality holds. Let  $\omega$  be a closed walk in  $G$ , that visits all the vertices. If the cost of  $\omega$  is  $K$ , then there exists a Hamiltonian cycle in  $G$  of cost at most  $K$ .*

By, Proposition 3.4 and Proposition 3.5, either there exists a Hamiltonian cycle in  $\overline{G}$  of cost at most  $K$ , proving the Lemma, or there exists a Hamiltonian cycle of  $W_n^t[G]$  from  $\tau_2$ , of cost at most  $tK$ . In the later case, by induction hypothesis, we get a Hamiltonian cycle in  $G$  of cost at most  $K$ . ■

#### Proof of Proposition 3.4

Consider the traveling salesman tour  $\tau_{opt}$  of  $W_n^l[G]$  of cost  $lK$ . Each edge  $(a, b)$  of  $\tau_{opt}$  is one of the following three types

- **Type I.** Both  $a, b \in \overline{G}$ . This includes the edges where one endpoint is  $v$  and the other is some vertex in  $\overline{G}$ .
- **Type II.** Both  $a, b \in W_n^t[G]$ . This includes the edges where one endpoint is  $v$  and the other is some vertex in  $W_n^t[G]$ .
- **Type III.**  $(a \in \overline{G} \wedge b \in W_n^t[G])$  or  $(a \in W_n^t[G] \wedge b \in \overline{G})$ . Notice that in type III edges,  $a, b \neq v$ .

Consider the walk along  $\tau_{opt}$  starting from the central vertex  $v$ . Since  $\tau_{opt}$  is a cycle and visits  $v$  only once, there is a non-empty sequence of type III edges between every sequence of type I and type II edges in  $\tau_{opt}$ .

For each type III edge  $(a, b)$  along the walk, replace  $(a, b)$  by the pair of edges  $(a, v), (v, b)$ . The resulting walk,  $\tau$ , is a closed walk in  $W_n^{t+1}[G]$  consisting only type I and type II edges.

$$\tau = \{v, \dots \text{type I edges} \dots, v, \dots \text{type II edges} \dots, v, \text{type I edges}, \dots, \dots, v\}$$

The subsequence  $\tau_1$ , consisting only type I edges in  $\tau$  defines a closed walk in  $\overline{G}$  visiting all the vertices. Similarly the subsequence of only type II edges,  $\tau_2$ , is a closed walk in  $W_n^t[G]$  visiting all the vertices.

By the definition of the extended windmill graph, for all type III edges  $(a, b)$

$$\text{cost}(a, b) = \text{cost}(a, v) + \text{cost}(v, b).$$

Hence,  $\text{cost}(\tau) = \text{cost}(\tau_{opt})$ .

$$\text{cost}(\tau_1) + \text{cost}(\tau_2) = \text{cost}(\tau) = \text{cost}(\tau_{opt}) \leq (t + 1)K.$$

Hence, either  $\text{cost}(\tau_1) \leq K$  or  $\text{cost}(\tau_2) \leq tK$ . ■

The following corollary is immediate from the proof of the *only if* part of Lemma 3.3

**Corollary 3.6** *If  $W_n^l[G]$  has a traveling salesman tour of cost less than  $lK$ ,  $G$  has a traveling salesman tour of cost less than  $K$ .*

For the reduction, we put  $l = n$ .

### The graph $H$ : reduced graph for UTTP instance

In this section, we construct the graph for the unconstrained Traveling Tournament Problem. Let  $G = (V, E)$  ( $|V| = n$ ) be the input TSP instance. Let  $W_n^n[G] = (V', E')$  be the extended windmill graph with  $n$  copies of  $G$ . Let  $m = |V'| = n(n - 1) + 1$ . Notice that  $m$  is an odd number.

Construct the complete weighted graph  $H$  as follows.

- The vertex set is  $V' \cup \{u\}$ .
- For all  $v \in V'$ ;  $d'(u, v) = 1000(m + 1)^4$ .
- Rest of the edges have the same weight as in  $W_n^n[G]$ .

The graph  $H$  is the instance graph of the reduced TTP instance.

**Lemma 3.7** *If any traveling salesman tour of  $G$  has cost at least  $K$ , then any traveling salesman tour of  $H$  has cost at least  $2000(m + 1)^4 + nK - 4$ .*

**Proof.** Suppose,  $H$  has a TSP tour  $\tau$  of cost less than  $2000(m + 1)^4 + nK - 4$ . Let  $(u, a)$  and  $(u, b)$  be the two edges in  $\tau$ , incident on  $u$ . By replacing  $(u, a), (u, b)$  by  $(a, b)$  we get a TSP tour  $\tau'$  of  $W_n^n[G]$ . By the construction of the extended windmill graph,  $\text{cost}(a, b) \leq 4$ .

$$\begin{aligned} \text{cost}(\tau') &= \text{cost}(\tau) - 2 \times 1000(m + 1)^4 + \text{cost}(a, b) \\ &< 2000(m + 1)^4 + nK - 4 - 2000(m + 1)^4 + 4 \\ &= nK. \end{aligned}$$

By Corollary 3.6, we get a traveling salesman tour of  $G$  of cost less than  $K$ , which contradicts the condition of the Lemma. ■

### 3.1 Placing teams on $H$

The number of teams of the UTTP instance will be  $10(m + 1)^2$  where  $m$  is the number of vertices in  $W_n^n[G]$ . The teams are distributed as follows

- $m + 1$  teams are placed in  $W_n^n[G]$ . The central vertex has two teams, where all other vertices have one team each.
- $(m + 1)(10m + 9)$  teams are placed at vertex  $u$ .

## Partitioning the teams in groups

Let  $T$  be the set of teams of the UTTP instance.  $|T| = n'$ . Let  $\ell = 10(m + 1)$ . We partition  $T$  in  $10(m + 1)$  subsets  $\{T_0, T_1, \dots, T_{\ell-1}\}$  such that

- Each partition contains  $(m + 1)$  teams.
- $T_{\ell-1}$  contains the teams from  $W_n^n[G]$ .
- $T_0, T_1, \dots, T_{\ell-2}$  contain teams from the vertex  $u$ .

## Numbering of teams

We represent the teams with the following notation. The ordered pair  $(i, \alpha)$  denotes the  $i^{\text{th}}$  team of the partition  $T_\alpha$ , where  $i \in \mathbb{Z}_{(m+1)}$ , and  $\alpha \in \mathbb{Z}_\ell$ .

## 4 The Reduction

In this section, we prove the reduction from (1, 2)- Traveling Salesman problem. We assume  $G = (V, E)$  be the input (1, 2)-TSP instance,  $|V| = n$ . Let  $H$  be the graph constructed from  $G$  and let the teams be placed as described in the previous Section. First we show the equivalence of the Yes instances, if there exists a Traveling Salesman tour in  $G$  of cost  $K$ , then there exists a feasible TTP schedule of the constructed instance of cost at most  $\zeta(K, n)$

**Lemma 4.1** *If  $G$  has a traveling salesman tour of cost at most  $K$ , then there exists a double round robin schedule of the UTTP instance of cost at most  $\zeta(K, n) = 20000(m + 1)^6 + 10m(m + 1)(nK + 1)$ , where  $m = n(n - 1) + 1$ .*

In Section 4.4, we show the equivalence of the No instances.

### 4.1 Proof of Lemma 4.1

Let  $\tau$  be the traveling salesman tour in  $G$  with cost  $K$ . By Lemma 3.3,  $W_n^n[G]$  has a traveling salesman tour,  $\tau'$  of cost  $nK$ . We view  $\tau'$  as a permutation over  $[m]$ ; i.e.  $\tau'$  denotes the traveling salesman tour  $\{v_{\tau'(1)}, v_{\tau'(2)}, \dots, v_{\tau'(m)}, v_{\tau'(1)}\}$ . Without loss of generality, we assume  $v_{\tau'(1)}$  to be the central vertex in  $W_n^n[G]$ . Define a permutation  $\pi : \mathbb{Z}_{m+1} \rightarrow \mathbb{Z}_{m+1}$ ,

$$\pi(i) = \begin{cases} 0 & \text{if } i = 0 \\ \tau'(i) & \text{if } i \in [m] \end{cases} \quad (1)$$

Recall that the created instance has  $\ell(m + 1)$  teams where  $\ell = 10(m + 1)$ . Hence the double round robin schedule will have  $2(\ell(m + 1) - 1)$  rounds, each round consisting  $\ell(m + 1)/2$  matches. The idea is to design a schedule where for all the teams from  $u$ , the sequence of away matches in  $W_n^n[G]$  is defined by the traveling salesman tour  $\tau'$ . The challenge is to synchronize these tours and still maintain a particular upper bound of the incurred cost.

In any double round robin league schedule of the constructed instance, there are two types of matches; *intra partition matches*, where a team plays with another team of the same partition, and *interpartition matches*, where a team plays with some team from other partitions. Accordingly our schedule is divided in two parts,

- **Intrapartition League:** For the first  $2m$  rounds, each team plays against the teams from the same partition. We construct a double round robin league schedule of  $m + 1$  teams (number of teams in one partition), and apply the schedule to all the partitions parallelly. Hence, each round consists of  $(m + 1)/2$  matches for each of the  $\ell$  partitions, amounting to  $\ell(m + 1)/2$  matches. The algorithm INTRAPARTITION describes the schedule for one partition.
- **Interpartition Tours:** In this phase, we schedule the matches between teams of different partitions. The algorithm TOUR, using the traveling salesman tour  $\tau'$ , gives a schedule where all the teams from one partition (say  $\alpha$ ) travels to play against the teams of some other partition (say  $\beta$ ). Such a schedule, implements  $m + 1$  rounds, each containing  $m + 1$  away matches of the teams of partition  $\alpha$  against the teams of partition  $\beta$ . To synchronize the tours, we use the algorithm INTRAPARTITION. We identify each partition as a team, and create a double round robin league schedule of  $\ell$  (number of partitions) teams. Each match  $(\alpha, \beta, .)$  of this schedule is then replaced by  $\text{Tour}(\alpha, \beta)$  creating a schedule for the  $2(\ell - 1)$  rounds of tours, equivalently scheduling  $2(\ell - 1)(m + 1)$  rounds of matches. Moreover, for each round the  $\ell/2$  many tours are scheduled parallelly. Hence, in the final schedule each round has  $\ell(m + 1)/2$  many matches.

These two phases implement the required  $2m + 2(\ell - 1)(m + 1) = 2(\ell(m + 1) - 1)$  rounds, each consisting  $\ell(m + 1)/2$  matches. At every round each team plays exactly one match. Moreover, any pair of teams play against each other twice; during IntraPartition league if both of them are in the same partition, or during the tours  $\text{Tour}(\alpha, \beta)$  and  $\text{Tour}(\beta, \alpha)$  if they are from partitions  $\alpha$  and  $\beta$  respectively.

---

**Algorithm 1: A SCHEDULE OF THE UTTP INSTANCE**

---

```

Input:  $T = \{T_0, T_1, \dots, T_{\ell-1}\}$ 
Output: A schedule of a double round robin tournaments for  $T$ 
1  $S = \emptyset$ 
2  $rnd = 0$ 
3 for  $\alpha = 0$  to  $\ell - 1$  do
4    $S = S \cup \text{INTRAPARTITION}(\alpha, m + 1, rnd)$  /* Scheduling the IntraPartition leagues parallelly */
5  $rnd = 2m$ 
6  $S = S \cup \text{TOURSCHEDULE}(\ell, rnd)$  /* Schedule the tours between the partitions */
7 return  $S$ 

```

---

The algorithms INTRAPARTITION and TOURSCHEDULE are described in the following sections.

## 4.2 Matches between teams from same Partition

Now we describe the algorithm INTRAPARTITION, which creates a double round robin league of  $m + 1$  teams, all from the same partition. We use the standard algorithm using Berger Table (See [4]), to create



a double round robin league schedule. Algorithm 2, creates a schedule for the intrapartition league.  $S_\alpha$  denotes the intrapartition league schedule for the partition  $\alpha$ .

---

**Algorithm 2:** INTRAPARTITION: to schedule the matches played between teams from the same partition

---

**Input:**  $\alpha$ : partition number,  $m + 1$ : number of teams in the partition,  $rnd$ : the current round number  
**Output:** A schedule of the intrapartition league in partition  $\alpha$

```

1  $S_\alpha = \emptyset$ 
2  $z = \frac{m+1}{2}$ 
3 for  $r = 0$  to  $2z - 2$  do
4   for all  $x \in [2z - 2]$  do
5      $a = (x + r \bmod (2z - 1), \alpha)$ 
6      $b = (2z - 1 - x + r \bmod (2z - 1), \alpha)$ 
7     if  $a < b$  then
8        $S_\alpha = S_\alpha \cup \{(a, b, rnd + r), (b, a, rnd + 2z - 1 + r)\}$ 
9     else
10       $S_\alpha = S_\alpha \cup \{(b, a, rnd + r), (a, b, rnd + 2z - 1 + r)\}$ 
11    $S_\alpha = S_\alpha \cup \{(r, 2z - 1, rnd + r), (2z - 1, r, rnd + 2z - 1 + r)\}$ 
12 return  $S_\alpha$ .
```

---

**Lemma 4.2**  $\sum_{\alpha=0}^{\ell-1} \text{cost}(S_\alpha) \leq 8m(m + 1)$

**Proof.** The partition  $T_{\ell-1}$  contains teams from  $W_n^n[G]$ . Rest of the partitions,  $T_0, T_1, \dots, T_{\ell-2}$ , contain teams only from the vertex  $u$ . So, during the intrapartition league the teams from  $T_0, T_1, \dots, T_{\ell-2}$  do not travel.

$$\text{cost}(S_\alpha) = 0 \quad \forall \alpha \in \{0, 1, \dots, \ell - 2\}. \quad (2)$$

Let  $(i, \ell - 1)$  be any team in partition  $T_{\ell-1}$  placed at vertex  $v_i$  in  $W_n^n[G]$ . By triangle inequality, the total traveling cost of  $(i, \ell - 1)$  is at most twice the sum of the weight of the edges incident on  $v_i$ . As the cost of each edge is at most 4, and there are exactly  $m$  other teams to play, the traveling cost of  $(i, \ell - 1)$  is at most  $8m$ . Taking sum over all  $i \in \mathbb{Z}_{m+1}$ , we get,  $\text{cost}(S_{\ell-1}) \leq 8m(m + 1)$ . This completes the proof. ■

### 4.3 Matches between teams from different partitions

Now we describe the algorithm TOURSCHEDULE. As discussed, to schedule the matches between teams of different partitions, we create a double round robin league schedule of  $\ell = 10(m + 1)$  teams, representing the partitions. Each match  $(a, b, \cdot)$  of that schedule will be expanded into a minileague, a schedule of  $m + 1$  (number of teams in each partition) rounds, each round consisting  $m + 1$  matches played between the teams of partitions  $T_a$  and  $T_b$ . The minileagues are scheduled using the subroutine Tour (Algorithm 4).

**Definition 4.3** Let  $T_\alpha$  and  $T_\beta$  be two disjoint subsets of  $T$ . A mini-league of  $T_\beta$  with  $T_\alpha$ , denoted by  $\text{Tour}(\alpha, \beta)$ , is the set of away matches played by the teams of  $T_\beta$  against the teams of  $T_\alpha$ .

$$\text{Tour}(\alpha, \beta) = \{(a, b, \cdot) | a \in T_\alpha, b \in T_\beta\}.$$

---

**Algorithm 3: TOURSCHEDULE:** schedule the interpartition matches
 

---

**Input:**  $\ell$ : the number of partitions,  $rnd$ : current round number  
**Output:** A schedule of tours

```

1  $z = \frac{\ell}{2}$ 
2 for  $r = 0$  to  $2z - 2$  do
3   for all  $x \in [2z - 2]$  do
4      $\alpha = x + r \pmod{2z - 1}$ 
5      $\beta = 2n - 1 - x + r \pmod{2z - 1}$ 
6     if  $a < b$  then
7        $S = S \cup \text{TOUR}(\alpha, \beta, rnd + r(m + 1))$ 
8        $S = S \cup \text{TOUR}(\beta, \alpha, rnd + (2z - 1 + r)(m + 1))$ 
9     else
10       $S = S \cup \text{TOUR}(\beta, \alpha, rnd + r(m + 1))$ 
11       $S = S \cup \text{TOUR}(\alpha, \beta, rnd + (2z - 1 + r)(m + 1))$ 
12    $S = S \cup \{\text{TOUR}(2z - 1, r, rnd + r(m + 1))\}$ 
13    $S = S \cup \{\text{TOUR}(r, 2z - 1, rnd + (2z - 1 + r)(m + 1))\}$ 
14 return  $S$ .
```

---



---

**Algorithm 4: TOUR:** to schedule the mini-leagues
 

---

**Input:** Two partition number  $\alpha, \beta \in \mathbb{Z}_\ell$ ,  $rnd$ : the current round number  
**Output:** A schedule for the mini-league  $\text{tour}(\alpha, \beta)$

```

1  $S = \emptyset$ ;
2 for  $r = 0$  to  $m$  do
3   for  $i = 0$  to  $m$  do
4      $b = (i, \beta)$ ;
5      $a = (\pi(i + r \pmod{m + 1}), \alpha)$ ;
6      $S = S \cup \{(a, b, rnd + r)\}$ 
7 return  $S$ 
```

---

The specific mini-leagues which are important to us are  $\text{Tour}(\ell - 1, \alpha)$ ,  $\alpha \in \mathbb{Z}_{\ell-1}$ ; where teams in  $u$  visit the teams in  $W_n^n[G]$ . We wish to upper bound the cost of our schedule of  $\text{Tour}(\ell - 1, \alpha)$ , when  $G$  has a traveling salesman tour of cost at most  $K$ . To this end, we need to define a tour used by the teams in  $u$  to visit the cities in  $W_n^n[G]$ .

### Schedule of a mini-league

We start with scheduling the mini-league  $\text{Tour}(\ell - 1, \alpha)$ ;  $\alpha < \ell - 1$ . Let  $\pi$  be the permutation as defined in Equation (1). Let  $(\pi(0), \ell - 1)$  and  $(\pi(1), \ell - 1)$  be the two teams at the central vertex of  $W_n^n[G]$ , i.e.  $v_{\pi(0)} = v_{\pi(1)}$ . For other vertices in  $W_n^n[G]$ ,  $(\pi(i), \ell - 1)$  denotes the team at vertex  $v_{\pi(i)}$ ,  $i \in \{2, 3, \dots, m\}$ . The schedule of  $\text{Tour}(\ell - 1, \alpha)$  is defined as follows. In round  $r \in [m + 1]$ , team  $(i, \alpha)$  plays with  $(\pi(i + r \pmod{m + 1}), \ell - 1)$ .

For other mini-leagues, either the teams in  $W_n^n[G]$  visit  $u$ , or all the involved teams are at vertex  $u$  itself. In these mini-leagues the teams do not travel between the rounds, albeit they may need to travel before and after a mini-league. We use the schedule for  $\text{Tour}(\ell - 1, \alpha)$  for other mini-leagues as well.

**Lemma 4.4** *Let the traveling salesman tour  $\tau$  of  $G$  have cost  $K$ . Assume, at the start, all the teams were at their home. At the end of the schedule, the teams return to their home cities. For all  $\alpha \in \mathbb{Z}_{\ell-1}$ , cost of the schedule generated by  $\text{TOUR}(\ell - 1, \alpha)$  is  $2000(m + 1)^5 + mnK$ .*

**Proof.** Let  $\tau'$ , constructed using Lemma 3.3, be the traveling salesman tour of  $W_n^n[G]$  of cost  $nK$ . Let  $\pi : \mathbb{Z}_{m+1} \rightarrow \mathbb{Z}_{m+1}$  be defined as in Equation (1).

Consider a schedule  $S$  generated by  $\text{TOUR}(\ell - 1, \alpha)$ . Let  $(i, \alpha)$  be any team of the partition  $\alpha$ . The path traveled by  $(i, \alpha)$  is  $\{(u, v_{\pi(i)}), (v_{\pi(i)}, v_{\pi(i+1)}), \dots, (v_{\pi(i-2)}, v_{\pi(i-1)}), (v_{\pi(i-1)}, u)\}$ . Cost of this path is  $2000(m + 1)^4 + nK - \text{cost}(\pi(i - 1), \pi(i))$ . Taking sum over all  $i \in \mathbb{Z}_{m+1}$ , the total traveling cost in  $S$  is  $(m + 1)(2000(m + 1)^4 + nK) - nK$  which is  $2000(m + 1)^5 + mnK$ . ■

## Bounding the Travel Cost

**Lemma 4.5** *If  $G$  has a traveling salesman tour of cost  $K$ , then the cost of the schedule generated by Algorithm 1 is less than  $\zeta = 20000(m+1)^6 + 10m(m+1)(nK+1)$ , where  $m = n(n-1) + 1$ .*

**Proof.** Let  $\ell = 10(m+1)$ . The first phase of the schedule is the first  $2m$  rounds, where the IntraPartition matches takes place. By, Lemma 4.2, the total cost of the first phase is at most  $8m(m+1)$ . For the rest of the rounds, total traveling costs for the teams in each partition can be bounded as follows.

1.  $\alpha \in \mathbb{Z}_{\ell-1}$ : All the teams of  $T_\alpha$  are placed in  $u$ . The only traveling cost of these teams is during  $\text{Tour}(\ell-1, \alpha)$ . Using Lemma 4.4, for each  $\alpha$ , the total traveling cost for the teams in  $T_\alpha$  is at most  $20000(m+1)^5 + mnK$ .
2.  $\alpha = \ell-1$ : The teams of  $T_{\ell-1}$  are placed in  $W_n^n[G]$ . The cost incurred by each team is due to the travel to  $u$  and return. By Algorithm 3, the teams of  $T_{\ell-1}$  play away matches at  $u$  only in the last  $(\ell-1)(m+1)$  rounds. The total traveling cost of all the teams is  $20000(m+1)^5$ .

So total traveling cost of the schedule is

$$\begin{aligned}
& 8m(m+1) + 2000(m+1)^5 + \sum_{\alpha=0}^{\ell-2} (20000(m+1)^5 + mnK) \\
&= 8m(m+1) + 2000(m+1)^5 + 2000(m+1)^5(10m+9) + (10m+9)mnK \\
&= 8m(m+1) + 10 \times (m+1) \times (2000 \times (m+1)^5) + (10m+9)mnK \\
&< 20000(m+1)^6 + 10m(m+1)nK + 8m(m+1) \\
&< 20000(m+1)^6 + 10m(m+1)(nK+1).
\end{aligned}$$

■

## 4.4 Lower bound on the traveling cost of any feasible solution

Now we show a lower bound of the cost of any feasible solution of UTTP instance.

**Lemma 4.6**  *$G = (V, E)$  be the input  $(1, 2)$ -TSP instance with  $|V| = n > 5$ . If  $G$  has no traveling salesman tour of cost less than or equal to  $K$ , then the constructed UTTP instance on  $H$  has no feasible schedule of cost less than or equal to  $20000(m+1)^6 + 10m(m+1)(nK+1)$ .*

**Proof.** For any schedule of a double round robin league, traveling cost of each team is at least the cost of the traveling salesman tour of the cities [5]. Let the optimal traveling salesman tour of  $G$  has the cost of  $K+k$ ,  $k \geq 1$ . By Lemma 3.7, any traveling salesman tour of  $H$  has cost at least  $2000(m+1)^4 + n(K+k) - 4$ .

$$\begin{aligned}
\text{Cost of any feasible schedule} &\geq \text{number of teams} \times \text{Cost of the optimal TSP tour of H} \\
&\geq 10(m+1)^2 \times (2000(m+1)^4 + n(K+k) - 4) \\
&> 10(m+1)^2 \times (2000(m+1)^4 + nK + 1) \\
&> 20000(m+1)^6 + 10m(m+1)(nK+1).
\end{aligned}$$

The third inequality follows from the condition of  $n > 5$  and  $k \geq 1$ . ■  
Using Lemma 4.5 and Lemma 4.6 we get Theorem 1.1.

## References

- [1] Aris Anagnostopoulos, Laurent Michel, Pascal Van Hentenryck, and Yannis Vergados. A simulated annealing approach to the traveling tournament problem. *J. Scheduling*, 9(2):177–193, 2006.
- [2] Fabrício N. Costa, Sebastián Urrutia, and Celso C. Ribeiro. An ILS heuristic for the traveling tournament problem with predefined venues. *Annals OR*, 194(1):137–150, 2012.
- [3] D. de Werra. Scheduling in sports. In *Annals of Discrete Mathematics (11) Studies on Graphs and Discrete Programming*, volume 59 of *North-Holland Mathematics Studies*, pages 381 – 395. North-Holland, 1981.
- [4] Jeffrey H. Dinitz. Designing schedules for leagues and tournaments, 2004.
- [5] Kelly Easton, George L. Nemhauser, and Michael A. Trick. The traveling tournament problem description and benchmarks. In *Principles and Practice of Constraint Programming - CP 2001, 7th International Conference, CP 2001, Paphos, Cyprus, November 26 - December 1, 2001, Proceedings*, pages 580–584, 2001.
- [6] Kelly Easton, George L. Nemhauser, and Michael A. Trick. Solving the travelling tournament problem: A combined integer programming and constraint programming approach. In *Practice and Theory of Automated Timetabling IV, 4th International Conference, PATAT 2002, Gent, Belgium, August 21-23, 2002, Selected Revised Papers*, pages 100–112, 2002.
- [7] Nobutomo Fujiwara, Shinji Imahori, Tomomi Matsui, and Ryuhei Miyashiro. Constructive algorithms for the constant distance traveling tournament problem. In *Practice and Theory of Automated Timetabling VI, 6th International Conference, PATAT 2006, Brno, Czech Republic, August 30 - September 1, 2006, Revised Selected Papers*, pages 135–146, 2006.
- [8] M. R. Garey and David S. Johnson. *Computers and Intractability: A Guide to the Theory of NP-Completeness*. W. H. Freeman, 1979.
- [9] Martin Henz. Scheduling a major college basketball conference - revisited. *Operations Research*, 49(1):163–168, 2001.
- [10] Shinji Imahori, Tomomi Matsui, and Ryuhei Miyashiro. A 2.75-approximation algorithm for the unconstrained traveling tournament problem. *Annals OR*, 218(1):237–247, 2014.
- [11] Graham Kendall, Sigrid Knust, Celso C. Ribeiro, and Sebastián Urrutia. Scheduling in sports: An annotated bibliography. *Computers & OR*, 37(1):1–19, 2010.
- [12] Thomas Kirkman. On a problem in combinatorics. *Cambridge and Dublin Mathematical Journal*, 2:191–204, 1847.

- [13] Joseph Leung, Laurie Kelly, and James H. Anderson. *Handbook of Scheduling: Algorithms, Models, and Performance Analysis*. CRC Press, Inc., Boca Raton, FL, USA, 2004.
- [14] Rasmus V. Rasmussen and Michael A. Trick. Round robin scheduling - a survey. *European Journal of Operational Research*, 188(3):617–636, 2008.
- [15] Clemens Thielen and Stephan Westphal. Complexity of the traveling tournament problem. *Theor. Comput. Sci.*, 412(4-5):345–351, 2011.
- [16] Sebastián Urrutia and Celso C. Ribeiro. Maximizing breaks and bounding solutions to the mirrored traveling tournament problem. *Discrete Applied Mathematics*, 154(13):1932–1938, 2006.