The Inter-League Extension of the Traveling Tournament Problem and its Application to Sports Scheduling

Richard Hoshino and Ken-ichi Kawarabayashi
National Institute of Informatics
2-1-2 Hitotsubashi, Chiyoda-ku, Tokyo 101-8430, Japan

Abstract

With the recent inclusion of inter-league games to professional sports leagues, a natural question is to determine the “best possible” inter-league schedule that retains all of the league’s scheduling constraints to ensure competitive balance and fairness, while minimizing the total travel distance for both economic and environmental efficiency. To answer that question, this paper introduces the Bipartite Traveling Tournament Problem (BTTP), the inter-league extension of the well-studied Traveling Tournament Problem.

We prove that the 2n-team BTTP is NP-complete, but for small values of n, a distance-optimal inter-league schedule can be generated from an algorithm based on minimum-weight 4-cycle-covers. We apply our algorithm to the 12-team Nippon Professional Baseball (NPB) league in Japan, creating an inter-league tournament that reduces total team travel by 16% compared to the actual schedule played by these teams during the 2010 NPB season. We also analyze the problem of inter-league scheduling for the 30-team National Basketball Association (NBA), and develop a tournament schedule whose total inter-league travel distance is just 3.8% higher than the trivial theoretical lower bound.

Introduction

In many professional sports leagues, the teams are divided into “conferences” based on historical rivalry or geographic proximity. During the season, each team plays intra-league games against teams from their own conference, as well as inter-league games against teams from the other conference. Many professional sports leagues adopt a two-conference structure, including the “Big Four” leagues of North America: the National Basketball Association (NBA), the National Football League (NFL), the National Hockey League (NHL), and Major League Baseball (MLB).

As teams must travel long distances to play their games during the course of a season, finding a schedule that reduces travel distance is important, for both economic and environmental reasons. Since the majority of regular season games occur within one’s conference, much of the research in sports scheduling has focused on intra-league play, with the goal of minimizing the sum total of distances traveled by all teams. The challenge of creating a distance-optimal intra-league schedule motivated the Traveling Tournament Problem (TTP), in which every pair of teams plays twice, with one game at each team’s home stadium. The output is an optimal schedule that minimizes the sum total of distances traveled by the teams as they move from city to city, subject to several constraints that ensure competitive balance.

Since its introduction (Easton, Nemhauser, and Trick 2001), the TTP has emerged as a popular area of study (Kendall et al. 2010) within the operations research community due to its complexity and depth. Research on the TTP has led to the development of powerful techniques in integer programming, constraint programming, as well as heuristics such as simulated annealing and hill-climbing (Lim, Rodrigues, and Zhang 2006). The TTP has direct applications to scheduling optimization, and can aid professional sports leagues as they make their regular season schedules more efficient, saving time and money, as well as reducing greenhouse gas emissions.

In this paper, we extend the TTP to inter-league play, connecting the techniques and methods of the Traveling Tournament Problem to the theory of bipartite tournaments, thus producing new directions for research in scheduling optimization. Determining distance-optimal inter-league schedules is a natural next step in the field of sports scheduling, especially given the recent introduction of inter-league play to pro leagues. For example, in Major League Baseball, inter-league play began only in 1997, despite having first been proposed in the 1930s.

We introduce the Bipartite Traveling Tournament Problem (BTTP), the inter-league analogue of the TTP. We prove that BTTP on 2n teams is NP-complete by obtaining a reduction from 3-SAT, the well-known NP-complete problem on boolean satisfiability (Garey and Johnson 1979). Despite its computational intractability for general n, we present a powerful heuristic based on minimum-weight 4-cycle-covers and apply it to the 12-team Nippon Professional Baseball (NPB) league in Japan, as well as the 30-team National Basketball Association (NBA). We solve BTTP for the NPB, producing a distance-optimal schedule that represents a 16% reduction compared to the actual distance traveled by the teams during the 2010 NPB season. We conclude the paper by finding a nearly-optimal solution to BTTP for the NBA.
Definitions of BTTP and BTTP*

Let there be $2n$ teams, with $n$ teams in each league. Let $X$ and $Y$ be the two leagues, with $X = \{x_1, x_2, \ldots, x_n\}$ and $Y = \{y_1, y_2, \ldots, y_n\}$. Let $D$ be the $2n \times 2n$ distance matrix, where entry $D_{p,q}$ represents the home stadia of teams $p$ and $q$. By definition, $D_{p,q} = D_{q,p}$ for all $p, q \in X \cup Y$, and all diagonal entries $D_{p,p}$ are zero.

Similar to the original TTP, we require the following conditions: that each team play one game per day; that no team has a home stand or road trip lasting more than three games; that no team play against the same opponent in two consecutive games; and that for all $1 \leq i, j \leq n$, teams $x_i$ and $y_j$ play twice, once in each other’s home venue.

To illustrate, Table 1 provides two examples of a feasible tournament satisfying all of the above conditions for the case $n = 3$. In this table, as in all other schedules that will be subsequently presented, home games are marked in bold.

<table>
<thead>
<tr>
<th>Team</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
</tr>
</thead>
<tbody>
<tr>
<td>$x_1$</td>
<td>$y_1$</td>
<td>$y_2$</td>
<td>$y_3$</td>
<td>$y_1$</td>
<td>$y_2$</td>
<td>$y_3$</td>
</tr>
<tr>
<td>$x_2$</td>
<td>$y_2$</td>
<td>$y_3$</td>
<td>$y_1$</td>
<td>$y_2$</td>
<td>$y_3$</td>
<td>$y_1$</td>
</tr>
<tr>
<td>$x_3$</td>
<td>$y_3$</td>
<td>$y_1$</td>
<td>$y_2$</td>
<td>$y_3$</td>
<td>$y_1$</td>
<td>$y_2$</td>
</tr>
</tbody>
</table>

Table 1: Two feasible inter-league tournaments for $n = 3$.

Following the convention of the TTP, whenever a team is scheduled for a road trip consisting of multiple away games, the team doesn’t return to their home city but rather proceeds directly to their next away venue. Furthermore, we assume that every team begins the tournament at home and returns home after its last away game. For example, in Table 1, team $x_1$ would travel a distance of $D_{x_1,y_1} + D_{y_1,y_2} + D_{y_2,y_3} + D_{y_3,x_1}$, when playing the top schedule and a distance of $D_{x_1,y_1} + D_{y_3,y_2} + D_{y_2,y_1} + D_{x_1,y_3}$ when playing the bottom schedule. The desired solution to BTTP is the tournament schedule that minimizes the total distance traveled by all $2n$ teams subject to the given conditions.

Let $BTTP^*$ be the restriction of $BTTP$ to the set of tournament schedules where on any given day, the teams in each league either all play at home, or all play on the road. For example, the top schedule in Table 1 is a feasible solution of both $BTTP^*$ and $BTTP$. We say that such schedules are uniform. While this uniformity constraint significantly reduces the number of potential tournaments, it allows us to quickly generate an approximate solution to $BTTP$ from an algorithm based on minimum-weight 4-cycle-covers. We now prove that both $BTTP$ and $BTTP^*$ are NP-complete, by obtaining a reduction from 3-SAT.

NP-completeness of BTTP and BTTP*

Let $S = C_1 \land C_2 \land \ldots \land C_m$ be the conjunction of $m$ clauses with three literals on the variables $u_1, u_2, \ldots, u_4$. From $S$, we will define the sets $X_S$ and $Y_S$ representing the teams in leagues $X$ and $Y$. From this set of $|X_S| + |Y_S|$ vertices, we will construct a complete graph and assign edge weights to produce the distance matrix $D_S$. We then prove the existence of an integer $T = T(m)$ for which the solutions to $BTTP$ and $BTTP^*$ have total travel distance $\leq T$ if $S$ is satisfiable. This will establish the desired results.

We can assume that literals $u_i$ and $\overline{u_i}$ occur equally often in $S$ for each $1 \leq i \leq l$. To see why, assume without loss that $u_i$ occurs less frequently than $\overline{u_i}$. By repeated addition of the tautological clause $(u_i \lor u_{i+1} \lor \overline{u_{i+1}})$, which does not affect the satisfiability of $S$, we can ensure that the number of occurrences of $u_i$ in $S$ matches that of $\overline{u_i}$.

Let $r(i)$ denote the number of occurrences of $u_i$ in $S$. In Figure 1, we present a “gadget” for each variable $u_i$, where the vertices $u_{i,r}$ and $\overline{u_i,r}$ correspond respectively to the $r^{th}$ occurrence of $u_i$ and $\overline{u_i}$ in $S$, vertex $u_{i,r}$ is adjacent to $u_{i,r-1}$ and $u_{i,r}$, and vertex $b_{i,r}$ is adjacent to $u_{i,r}$ and $\overline{u_i,r}$. (Note: we set $\overline{u_i,0} := \overline{u_i,r(i)}$ for all $i$.)

This gadget was used to establish the NP-completeness of deciding whether an undirected graph contains a given number of vertex-disjoint $s$-$t$ paths of a specified length (Iitai, Perl, and Shiloach 1982) and to prove that the original TTP is NP-complete (Thielen and Westphal 2010). There are $l$ gadgets, one for each $u_i$, $i = 1, 2, \ldots, l$. Now we define the graph $G_S$. We create vertices $c_j$ and $d_j$ for $1 \leq j \leq m$, one pair for each clause in $S$. Join $c_j$ to $d_j$. Now connect $c_j$ to vertex $u_{i,r}$ iff clause $C_j$ contains the $r^{th}$ occurrence of $u_i$ in $S$. Similarly, connect $c_j$ to vertex $\overline{u_i,r}$ iff clause $C_j$ contains the $r^{th}$ occurrence of $\overline{u_i}$ in $S$.

To illustrate, let $S = C_1 \land C_2 \land C_3 \land C_4 \land C_5 \land C_6 \land C_7 \land C_8$, where $C_1 = (u_1 \lor u_2 \lor u_3)$, $C_2 = (\overline{u_1} \lor \overline{u_2} \lor \overline{u_3})$, $C_3 = (u_1 \lor \overline{u_2} \lor u_4)$, $C_4 = (u_2 \lor \overline{u_3} \lor u_4)$, $C_5 = (\overline{u_1} \lor u_3 \lor u_4)$, $C_6 = (u_1 \lor \overline{u_2} \lor \overline{u_4})$, $C_7 = (u_2 \lor \overline{u_3} \lor \overline{u_4})$, and $C_8 = (\overline{u_1} \lor u_3 \lor \overline{u_4})$. By definition, $S$ is an instance of 3-SAT. The gadget graph $G_S$ is given in Figure 2.

Since each literal occurs as often as its negation, and each clause has three literals, the number of clauses in $S$ must be even. Hence, $m = 2k$ for some integer $k \geq 1$. From the instance $S$, we will define a set $X_S$ with $18k$ vertices corresponding to the teams in league $X$. We will then define another set $Y_S$, with just 3 vertices (labeled $p$, $q$, and $r$), and place $6k$ teams at each of these three vertices. This will create a $36k$-team league, with $18k$ teams in both $X$ and $Y$. The
weight of each edge will just correspond to the distance between the teams located at those vertices. Using the gadget graph $G_S$, we will define the edge weights in such a way that $S$ is satisfiable if and only if the solutions to $BTP$ and $BTTP*$ have total distance at most $T = T(k) = 96k^2(2900k^3 + 375k + 11)$. This will establish the desired reduction from 3-SAT.

We first define $X_S$. Let $C = \{c_1, c_2, \ldots, c_{2k}\}$ and $D = \{d_1, d_2, \ldots, d_{2k}\}$, which are the same set of vertices from the corresponding gadget graph $G_S$. Let $U$ be the set of $6k$ vertices of the form $u_{i,r}$ or $\overline{u}_{i,r}$ that appear in $G_S$, and let $A$ and $B$ be respectively the set of vertices of the form $a_{i,r}$ and $b_{i,r}$ that appear in $G_S$. Finally, we present two additional sets, $E = \{e_1, e_2, \ldots, e_k\}$ and $F = \{f_1, f_2, \ldots, f_k\}$, which will be matched up to the vertices of $U$ in our cycle cover.

We define $X_S = A \cup B \cup C \cup D \cup E \cup F \cup U$. Hence, $|X_S| = |A| + |B| + |C| + |D| + |E| + |F| + |U| = 3k + 3k + 6k + 2k + 2k + k + 6k = 18k$.

Having defined $X_S$, we now define the edge weights connecting each pair of vertices in $X_S$, thus producing a complete graph on $18k$ vertices. The weight of each edge will be a function of $k$. For readability, we will express each weight as a function of $z$, where $z := 20k + 1$. To each edge in this complete graph, we assign a weight from the set $\{z^2, z^2 + z, 2z^2 - 1\}$ as follows:

1. A weight of $z^2$ is given to every edge that appears in the gadget graph $G_S$, the $6k^2$ edges from $U$ to $E$, and the $k$ edges connecting $c_i$ to $f_i$ (for each $1 \leq i \leq k$).
2. A weight of $z^2 + z$ is given to the $6k^2$ edges from $U$ to $F$, the $6k$ edges connecting $A$ to $B$ through a common neighbour in $U$, and the $6k$ edges connecting $D$ to $U$ through a common neighbour in $C$.
3. A weight of $2z^2 - 1$ is given to every other edge.

For example, the edge from $c_i$ to $p$ is given a weight of $2z^2 - 1$, for all $i = 1, 2, \ldots, 2k$. We repeat the same process for each of the $7 \times 3 = 21$ pairs connecting a vertex in $X_S = A \cup B \cup C \cup D \cup E \cup F \cup U$ to a vertex in $Y_S = \{p, q, r\}$.

Finally, let the weights of edges $pq$, $pr$, and $qr$ all be $2z^2 - 1$. As a result, we now have created a complete graph on the vertex set $X_S \cup Y_S$, and assigned a weight to each edge. Moreover, the weight of each edge appears in the set $\{z^2, z^2 + z, 2z^2 - 1\}$, where $z = 20k + 1$. As most versions of the TTP require the teams to be located at points satisfying the Triangle Inequality, we have chosen the weights in our inter-league variant to ensure that the Triangle Inequality holds for any triplet of points in $X_S \cup Y_S$.

We now partition the $18k$ vertices of $X_S$ into groups of cardinality at most three and attach them to each $y \in \{p, q, r\} = Y_S$ to produce a union of cycles of length at most 4. More formally, we define the following:

**Definition 1** For each $y \in Y_S$, a $y$-rooted 4-cycle-cover is a union of cycles of length at most 4, where every cycle contains $y$, no cycle contains a vertex from $Y_S \setminus \{y\}$, and every vertex of $X_S$ appears in exactly one cycle.

Figure 3 gives a $p$-rooted 4-cycle-cover with $|X_S| = 18$.

This definition is motivated by our tournament construction, where we will show that the total travel distance is minimized by creating a uniform schedule where each team takes the maximum number of three-game road trips to play their $18k$ away games. In the case of the $6k$ teams of $Y_S$ located at vertex $p$, their $6k$ three-game road trips will correspond to the $6k$ 4-cycles in the minimum weight $p$-rooted 4-cycle-cover. For example, if $p - u_{1.1} - c_5 - d_5 - p$ appears as one of the $6k$ cycles, then each team in $Y_S$ located at vertex $p$
will play three consecutive road games during the tournament against the teams of $X_S$ located at $v_{1,1}$, $c_5$, and $d_5$.

So the total distance traveled by each team at $y \in Y_S$ is bounded below by the sum of the edge weights of the minimum weight $g$-rooted 4-cycle-cover.

**Definition 2** We define three special types of cycles that may appear in a $p$-rooted 4-cycle-cover.

1. A $(p, a, u, b, p)$-cycle is a 4-cycle with vertices $p$, $a$, $u$, $b$ in that order, where $p \in Y_S$, $a \in A$, $u \in U$, $b \in B$, where $au$ and $ub$ are both edges in the gadget graph $G_S$.

2. A $(p, u, c, d, p)$-cycle is a 4-cycle with vertices $p$, $u$, $c$, $d$ in that order, where $p \in Y_S$, $u \in U$, $c \in C$, $d \in D$, where $uc$ and $cd$ are both edges in the gadget graph $G_S$.

3. A $(p, u, e, f, p)$-cycle is a 4-cycle with vertices $p$, $u$, $e$, $f$ in that order, where $p \in Y_S$, $u \in U$, $e \in E$, $f \in F$, where $e$ and $f$ have the same index (i.e., $e_i$ and $f_i$ for some $1 \leq i \leq k$).

For example, for our instance $S$ whose gadget graph was illustrated in Figure 2, $p-a_1,2-\pi_1,1-1,1-1$ is a $(p, a, u, b, p)$-cycle, but $p-a_1,2,1,1-1,1-1$ is not. Similarly, $p-\pi_1,3,3-\pi_8,8-8$ is a $(p, u, c, d, p)$-cycle, but $p-\pi_1,3,3-\pi_8,8-8$ is not a $(p, u, e, f, p)$-cycle.

Following the convention of the TTP (Easton, Nemhauser, and Trick 2001), we define $ILB_T$ to be the individual lower bound for team $t$. This value represents the minimum possible distance that can be traveled by team $t$ in order to complete all of their games under the constraints of BTTP, independent of the other teams’ schedules. By definition, for each team $t$ located at $y \in Y_S$, the value of $ILB_T$ is the minimum weight of a $g$-rooted 4-cycle-cover.

Similarly, we define the league lower bound $LLB_T$ to be the minimum possible distance traveled by all of the teams $t$ in league $T$, and the tournament lower bound $TLB_T$ to be the minimum possible distance traveled by all the teams in both leagues. We note the following trivial inequalities:

$$TLB \geq LLB_X + LLB_Y$$

$$LLB_X \geq \sum_{i \in X} ILB_i, \quad LLB_Y \geq \sum_{i \in Y} ILB_i.$$

By definition, the solution to BTTP is a tournament schedule whose total travel distance is $TLB_T$.

We now have all of the definitions we need to complete the proof of the NP-completeness of BTTP and BTTP*.

We will create an inter-league tournament between the 18k teams of $X_S$ and the 18k teams of $Y_S$ (with one-third of the teams at each vertex of $Y_S$), and show that there exists a distance-optimal (uniform) tournament with total distance $T(k) = 96k^2(2900k^2 + 3775k + 11)$ iff $S$ is satisfiable. The desired result will follow from the next four lemmas.

In each lemma, we let $K_S$ be the complete graph on the 18k + 3 vertices of $X_S \cup Y_S$, with edge weights as described in our construction.

**Lemma 1** The following statements are equivalent:

(i) $S = C_1 \land C_2 \land \ldots \land C_{2k}$ is satisfiable.

(ii) There exists a $p$-rooted 4-cycle-cover of $K_S$ with exactly $3k$ $(p, u, a, b, p)$-cycles, $2k$ $(p, u, c, d, p)$-cycles, and $k$ $(p, u, e, f, p)$-cycles.

**Proof** First, we prove $(i) \rightarrow (ii)$. If $S$ is satisfiable, then there exists a valid truth assignment, i.e., a function $\phi$ for which $\phi(u_i) \in \{\text{TRUE}, \text{FALSE}\}$ for every $1 \leq i \leq l$ that ensures that each clause $C_i$ evaluates to TRUE for all $1 \leq j \leq 2k$. From $\phi$, we build a $p$-rooted 4-cycle-cover of $K_S$ with exactly $3k$ $(p, a, u, b, p)$-cycles, $2k$ $(p, u, c, d, p)$-cycles, and $k$ $(p, u, e, f, p)$-cycles.

We first identify the $3k$ $(p, a, u, b, p)$-cycles. For each $1 \leq i \leq l$, if $\phi(u_i)$ is FALSE, then select all 4-cycles of the form $p-a_i,1-r_i,1-r_i-b_i,r_i-p$ for $r = 1, 2, \ldots, r(i)$. And if $\phi(u_i)$ is TRUE, then select all 4-cycles of the form $p-a_i,1-r_i,1-r_i-b_i,r_i-p$, where $a_i,r(i)+1 := a_{i,1}$. Repeating this construction for each $i$, we produce $3k$ $(p, a, u, b, p)$-cycles, covering the 6k vertices of $A \cup B$, as well as $3k$ vertices of $U$.

Now consider any clause $C_j$. Since $\phi$ is a valid truth assignment, at least one of the three literals in $C_j$ evaluates to TRUE. Thus, there exists some index $i$ for which $u_i \in C_j$ and $\phi(u_i)$ is TRUE, or $\exists i \in C_j$ and $\phi(u_i)$ is FALSE.

In the former case, where $u_i \in C_j$ and $\phi(u_i)$ is TRUE, there exists some index $r$ for which $u_i,r-e_j$ is an edge of the gadget graph $G_S$. Then $p-a_i,1-r_i,1-r_i-b_i,r_i-p$ is a $(p, u, c, d, p)$-cycle. Note that $u_i,r$ has not been previously selected in a $(p, u, b, p)$-cycle since $\phi(u_i)$ is TRUE (and so only the vertices $\pi_{1,1}, \pi_{1,2}, \ldots, \pi_{i,r}(i)$ were covered earlier).

In the latter case, where $\pi_i \in C_j$ and $\phi(u_i)$ is FALSE, there exists some index $r$ for which $\pi_i,r-e_j$ is an edge of the gadget graph $G_S$. Then $p-\pi_{i,1},r,e_j,1-r_i-b_i,r_i$ is a $(p, u, c, d, p)$-cycle. Note that $\pi_i,e_j$ has not been previously selected in a $(p, u, b, p)$-cycle since $\phi(u_i)$ is FALSE (and so only the vertices $u_i, \pi_{1,1}, \pi_{1,2}, \ldots, \pi_{i,r}(i)$ were covered earlier).

Repeating this construction for each $j$, we produce $2k$ $(p, u, c, d, p)$-cycles, covering the 4k vertices of $C \cup D$. Note that no $u \in U$ can be chosen twice since each vertex in $U$ is adjacent to only one vertex in $C$. Thus, these 2k cycles cover a set of 6k vertices in $X_S$, completely disjoint from the 9k vertices covered by the previously-constructed 3k $(p, u, b, p)$-cycles. As a result, we are left with 3k vertices in $X_S$ still to be covered, specifically $k$ vertices in each of $U$, $E$, and $F$. These vertices can be trivially partitioned into $k$ $(p, u, e, f, p)$-cycles by just ensuring that $e_i$ and $f_i$ belong to the same cycle for each $1 \leq i \leq k$. When this process is complete, our $p$-rooted 4-cycle-cover of $K_S$ will contain exactly $3k$ $(p, a, u, b, p)$-cycles, $2k$ $(p, u, c, d, p)$-cycles, and $k$ $(p, u, e, f, p)$-cycles.

We now prove $(ii) \rightarrow (i)$. Consider a $p$-rooted 4-cycle-cover of $K_S$ containing exactly $3k$ $(p, a, u, b, p)$-cycles, $2k$ $(p, u, c, d, p)$-cycles, and $k$ $(p, u, e, f, p)$-cycles. We prove there exists a function $\phi$ that is a satisfying truth assignment for $S$, where $\phi(u_i) \in \{\text{TRUE}, \text{FALSE}\}$ for each $1 \leq i \leq l$.

Define an a-b path to be any path on three vertices whose endpoints are $a_{i,j}$ and $b_{i,k}$, for some indices $i, j, k$. Consider the problem of maximizing the number of vertex-disjoint a-b paths in the $i^{th}$ gadget. A maximum packing of a-b paths occurs iff the $r(i)$ paths are chosen by taking all paths of the form $a_{i,r},i,r,b_{i,r}$ for each $1 \leq r \leq r(i)$, or all paths of the form $a_{i,r+1},i,r,b_{i,r}$ for each $1 \leq r \leq r(i)$.

The former case corresponds to selecting our a-b paths vertically; the latter, diagonally. These are the only two ways to achieve a maximum packing. Thus, in our $p$-rooted
4-cycle-cover containing $3k$ $(p, a, u, b, p)$-cycles, one of the following scenarios must hold true in the $i$th gadget:

(1) For each $r = 1, 2, \ldots, r(i)$, vertex $u_{i,r}$ appears in some $(p, a, u, b, p)$-cycle, while no vertex $u_{i,r}$ appears in any $(p, a, u, b, p)$-cycle.

(2) For each $r = 1, 2, \ldots, r(i)$, vertex $u_{i,r}$ appears in some $(p, a, u, b, p)$-cycle, while no vertex $u_{i,r}$ appears in any $(p, a, u, b, p)$-cycle.

In our given $p$-rooted 4-cycle-cover of $K_S$, for each $i$ define $\phi(u_i) = \text{FALSE}$ in scenario (1) and define $\phi(u_i) = \text{TRUE}$ in scenario (2). We claim that this is our desired function $\phi$. To prove this, consider the $2k (p, u, c, d, p)$-cycles in our 4-cycle-cover. For each $1 \leq j \leq 2k$, the $(p, u, c, d, p)$-cycle containing $c_j$ also contains some other vertex in $U$. This vertex is either $u_{i,r}$ or $u_{i,r}$, for some indices $i$ and $r$.

In the former case, $u_{i,r}$ and $c_j$ appear in the same $(p, u, c, d, p)$-cycle, implying that $u_{i,r}c_j$ is an edge of the gadget graph $G_s$, and that $u_{i,r}$ is a literal in clause $C_j$. Since $u_{i,r}$ appears in this $(p, u, c, d, p)$-cycle and therefore not in any $(p, a, u, b, p)$-cycle, this implies scenario (2). Since $\phi(u_i) = \text{TRUE}$ and $u_i \in C_j$, clause $C_j$ evaluates to $\text{TRUE}$.

In the latter case, $u_{i,r}$ and $c_j$ appear in the same $(p, u, c, d, p)$-cycle, implying that $u_{i,r}c_j$ is an edge of the gadget graph $G_s$, and that $u_{i,r}$ is a literal in clause $C_j$. Since $u_{i,r}$ appears in this $(p, u, c, d, p)$-cycle and therefore not in any $(p, a, u, b, p)$-cycle, this implies scenario (1). Since $\phi(u_i) = \text{FALSE}$ and $u_i \in C_j$, clause $C_j$ evaluates to $\text{TRUE}$.

Since $C_j$ evaluates to $\text{TRUE}$ for all $1 \leq j \leq 2k$, this implies that $\phi$ is a valid truth assignment. We conclude that $S = C_1 \land C_2 \land \ldots \land C_{2k}$ is satisfiable.

We illustrate Lemma 1 with an example. We can show that the instance $S$, whose gadget graph was illustrated in Figure 2, is not satisfiable. Therefore, there is no $p$-rooted 4-cycle-cover of $K_S$ with 12 $(p, a, u, b, p)$-cycles, 8 $(p, u, c, d, p)$-cycles, and 4 $(p, u, e, f, p)$-cycles.

**Lemma 2** The following statements are equivalent:

(i) A $p$-rooted 4-cycle-cover of $K_S$ has exactly 3k $(p, a, u, b, p)$-cycles, 2k $(p, u, c, d, p)$-cycles, and $k (p, u, e, f, p)$-cycles.

(ii) A $p$-rooted 4-cycle-cover of $K_S$ has total edge weight $k(24z^2 + 3z)$.

**Proof** First, we prove (i) $\implies$ (ii). In a $(p, a, u, b, p)$-cycle, the edges $au$ and $ab$ appear in the gadget graph $G_s$. Therefore, the edge weights of $au$ and $ab$ are both $2z^2$. From Table 2, a $(p, a, u, b, p)$-cycle has edge weight $2z^2 + z^2 + 2z^2 + 2z^2 = 4z^2$. Similarly, a $(p, u, c, d, p)$-cycle has edge weight $(x^2 + z) + (x^2 + z) + (x^2 + z) = 4x^2 + z$, and a $(p, u, e, f, p)$-cycle has edge weight $(x^2 + z) + (x^2 + z) = 4x^2 + z$. So if a $p$-rooted 4-cycle-cover of $K_S$ has exactly 3k $(p, a, u, b, p)$-cycles, 2k $(p, u, c, d, p)$-cycles, and $k (p, u, e, f, p)$-cycles, then its total edge weight is exactly $3k(4x^2) + 2k(4x^2 + z) + k(4x^2 + z) = k(24z^2 + 3z)$.

We now prove (ii) $\implies$ (i). Let $R$ be a $p$-rooted 4-cycle-cover of $K_S$ which is the union of $r$ cycles, with total edge weight $k(24z^2 + 3z)$. Since each of the 18k vertices of $X_S$ is covered by exactly one cycle of $R$, the number of edges in $R$ is $|X_S| + r = 18k + r$. Since no cycle has length greater than 4, we have $r \geq \frac{18k}{4} = 6k$. Now suppose $r \geq 6k + 1$. Then there are at least $24k + 1$ edges in $R$, all of which have weight at least $z^2$ given the construction of our complete graph $K_S$. Hence, the total edge weight of $R$ is at least $(24k + 1)z^2 = 24kz^2 + z^2 = 24kz^2 + z(20k + 1) > 24kz^2 + 3z = k(24z^2 + 3z)$, a contradiction.

It follows that $r = 6k$, and that $R$ must be the union of $6k$ cycles of length 4. Recall that the weight of each edge appears in the set \{(z^2, z^2 + z, z^2 + 2z, 2z^2 + z)^+\}. Suppose that one of these $24k$ edges has weight $2z^2 - 1$. Then the total edge weight of $R$ is at least $(24k - 1)z^2 + (2z^2 - 1) > k(24z^2 + 3z)$, a contradiction. Hence, all edges of $R$ must have weight $z^2$, $z^2 + z$, or $z^2 + 2z$.

From Table 2, we see that no edges $p-c$ and $p-e$ can appear in our 4-cycle-cover $R$, since all edges from $p$ to $C \cup E$ have weight $2z^2 - 1$. Thus, there must exist $2k$ 4-cycles of the form $p-?c_j-?p$ and $k 4$-cycles of the form $p-?c_j-?p$, with each of these $2k + 2 = 3k 4$-cycles containing a unique element from $C \cup E$. Each blank space (denoted by a question mark) can only be filled with a vertex from $D, F$, or $U$, as the weights of edges $ca, cb, ea, eb$ are all $2z^2 - 1$ for all $a \in A, b \in B, c \in C$, and $e \in E$.

Since edge $p-u$ has weight $z^2 + z$, if some vertex $u \in U$ is chosen to appear in one of these $3k$ 4-cycles, then this adds edge weight $z^2 + z$, producing a 4-cycle of weight at least $4z^2 + z$. But if no vertices $u \in U$ are chosen to replace these blank spaces, then the cycles must be of the form $p-d_j-c_j-d_k-p$ or $p-f_j-c_j-f_k-p$, both of which lead to the addition of at least one edge of weight $2z^2 - 1$ (since we cannot simultaneously have $i = j, i = k,$ and $j \neq k$). It follows that these $2k + 2 = 3k 4$-cycles containing the vertices of $C \cup E$ must each have weight at least $4z^2 + z$, thus contributing at least $k(12z^2 + 3z)$ to the total distance of $R$.

Since the given 4-cycle-cover $R$ has weight $k(24z^2 + z)$, this implies that the rest of the $3k$ 4-cycles must each have weight exactly $4z^2$, and that in each of the $2k$ cycles of the form $p-?c_j-?p$ and $k$ cycles of the form $p-?c_j-?p$, the total edge weight must be exactly $4z^2 + z$ to ensure that the total edge weight of $R$ does not exceed $k(24z^2 + 3z)$. This implies that in these two scenarios, we cannot replace the two blank spaces with two distinct vertices from $U$, as that would create a cycle of weight $4z^2 + 2z$. It follows that $R$ must have $2k (p, u, c, d, p)$-cycles and $k (p, u, e, f, p)$-cycles.

We are now left with $3k$ vertices from each of $A, B,$ and $U$ to form our remaining $3k$ 4-cycles. For the total edge weight of $R$ to not exceed $k(24z^2 + 3z) = 3k(4z^2 + z) + 12kz^2$, each of the remaining $12k$ edges must have weight $z^2$. Since edge $p-u$ has weight $z^2 + z$ for all $u \in U$, the $3k$ remaining vertices in $U$ must each appear in a unique 4-cycle, none adjacent to the root vertex $p$. Thus, the remaining $3k$ 4-cycles of $R$ must all be $(p, a, u, b, p)$-cycles.

**Lemma 3** Let $ILB_y$ be the minimum total edge weight of a $y$-rooted 4-cycle-cover of $K_S$. Then

$$ILB_y = \begin{cases} k(24z^2 + 3z) & \text{if } y = p \\ k(24z^2 + 20z) & \text{if } y = q \\ k(24z^2 + 19z) & \text{if } y = r \end{cases}$$
Proof From the proof of Lemma 2, we see that $ILB_p = k(24z^2 + 3z)$. The other two cases $(y = q$ and $y = r)$ follow similarly, and so we omit the details.

Just as we defined special 4-cycles rooted at $p$ (e.g. $(p, a, u, b, p)$-cycles), we can similarly define 4-cycles rooted at $q$ and $r$. In the above lemma, the lower bound $ILB_{2p}$ occurs when the $q$-rooted 4-cycle-cover consists of $3k(q, u, b, a, q)$-cycles, $2k(q, e, d, u, q)$-cycles, and $k(q, f, u, e, q)$-cycles, with total edge weight $3k(4z^2 + 4z) + 2k(4z^2 + 3z) + k(4z^2 + 2z) = k(24z^2 + 20z)$. The lower bound $ILB_{2q}$ occurs when the $r$-rooted 4-cycle-cover consists of $3k(r, b, a, u, r)$-cycles, $2k(r, d, u, c, r)$-cycles, and $k(r, e, f, u, r)$-cycles, with total edge weight $3k(4z^2 + 4z) + 2k(4z^2 + 2z) + k(4z^2 + 3z) = k(24z^2 + 19z)$.

Recall that $LLB_{2y}$ is the league lower bound for the 18k teams in $Y_S = \{p, q, r\}$. By Lemma 3, it follows that $LLB_{2y} \geq 6k(ILB_p + ILB_q + ILB_r) = 6k^2(24z^2 + 3z) + 6k^2(24z^2 + 20z) + 6k^2(24z^2 + 19z) = 6k^2(72z^2 + 42z)$. We now construct a uniform double round-robin bipartite tournament so that this value of $LLB_{2y}$ is achieved whenever $S$ is satisfiable. In this schedule, each team in $Y_S$ plays three consecutive road games followed by three consecutive home games and repeats that pattern until the end of the tournament. Thus, each team has 6k three-game road trips.

Let $\phi$ be a satisfying truth assignment for $S$. By the proof of Lemma 1, $\phi$ generates $3k(p, a, u, b, p)$-cycles, $2k(p, u, c, d, p)$-cycles and $k(p, u, e, f, p)$-cycles. These 6k cycles form a p-rooted 4-cycle-cover of $K_S$ with total edge weight $ILB_p = k(24z^2 + 3z)$.

We first schedule the 6k three-game road trips for each team at $p \in Y_S$ according to this minimum weight p-rooted 4-cycle-cover, ensuring that every team $p$ plays a unique opponent in each of the 18k time slots. For each $1 \leq i \leq 6k$, we use Table 3 below to generate the schedules for teams $q_i$ and $r_i$, based on the schedule of team $p_i$.

<table>
<thead>
<tr>
<th>$p_i$</th>
<th>$a^* u$</th>
<th>$a$</th>
<th>$b^*$</th>
<th>$u$</th>
<th>$c^*$</th>
<th>$d$</th>
<th>$e$</th>
<th>$f^*$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$q_i$</td>
<td>$u$</td>
<td>$b$</td>
<td>$a^*$</td>
<td>$c$</td>
<td>$d^*$</td>
<td>$u^*$</td>
<td>$e$</td>
<td>$f^*$</td>
</tr>
<tr>
<td>$r_i$</td>
<td>$b^*$</td>
<td>$a$</td>
<td>$u^*$</td>
<td>$d$</td>
<td>$u^*$</td>
<td>$c^*$</td>
<td>$e$</td>
<td>$f^*$</td>
</tr>
</tbody>
</table>

Table 3: Construction of road games for teams in $Y_S$.

For example, suppose that some team $p_i$ plays against teams $a^* \in A$, $u^* \in U$, and $b^* \in B$ in slots 1, 2, and 3. Then team $q_i$ will play against $u^*$ in slot 1, $b^*$ in slot 2, and $a^*$ in slot 3. It is easy to see that this construction generates one-half of a uniform schedule where each team $y \in Y_S = \{p, q, r\}$ has total travel distance $ILB_y$, corresponding to the minimum-weight $y$-rooted 4-cycle-cover.

We now determine the value of $ILB_t$ for each $t \in X_S = A \cup B \cup C \cup D \cup E \cup F \cup U$. Every team $t \in X_S$ plays a road game against each of the 18k teams in $Y_S$, with 6k teams located at points $p$, $q$, and $r$. Team $t$ must make at least $4t = 2k$ trips to each of $p$, $q$, and $r$, since the maximum length of a road trip is three games. Therefore, $ILB_t \geq 2k(D_{tp} + D_{tq} + D_{tr})$, where $D_{tx}$ is the distance from $t \in X_S$ to $y \in Y_S$ for all choices of $t$ and $y$. Note that equality can occur, specifically when the road trips of team $t$ are scheduled in the most efficient way, with each trip consisting of three consecutive games against three teams located at the same point.

From Table 2, $ILB_t = 2k(D_{tp} + D_{tq} + D_{tr}) = 4k(4z^2 + z - 1)$ for all $t \in A \cup B \cup C \cup E$. Similarly, $ILB_t = 4k(4z^2 - 1)$ for all $t \in D \cup F$, and $ILB_t = 4k(3z^2 + 5z)$ for all $t \in U$. Thus, $LLB_{X_S} \geq 4k(4z^2 + z - 1)(|A| + |B| + |C| + |E|) + 4k(4z^2 - 1)(|D| + |F|) + 4k(3z^2 + 5z)(|U|) = k^2(264z^2 + 156z - 48).

Therefore, $TLB \geq LLB_{X_S} + LLB_{Y_S} \geq \sum_{t \in X_S} ILB_t = k^2(264z^2 + 156z - 48) + 6k^2(72z^2 + 42z) = k^2(696z^2 + 408z - 48) = 96k^2(2900k^2 + 375k + 11).

We can quickly construct the other half of our uniform schedule, where the teams in $X_S$ play on the road, ensuring that each team $t \in X_S$ has total travel distance $ILB_t$. All that is required when putting the schedules together is to ensure the no-repeat rule, which is a simple matter given all of the flexibility we have in constructing this half of the tournament schedule. Therefore, we have proven the following lemma:

Lemma 4 If $S$ is satisfiable, then there exists a uniform schedule (i.e., a solution to BTTP and BTTP*) whose total travel distance is $\sum_{t \in X_S} ILB_t = 96k^2(2900k^2 + 375k + 11)$.

Having provided all of the lemmas, we can now prove the main theorem of this paper.

Theorem 1 BTTP and BTTP are NP-complete.

Proof Let $S$ be an instance of 3-SAT with 2k clauses, and create sets $X_S$ and $Y_S$, with edge weights as described in our construction. Consider an inter-league tournament between the 18k teams in $X_S$ and the 18k teams at $Y_S$ (with one-third of the teams at each vertex of $Y_S$).

By Lemma 4, if $S$ is satisfiable, then there exists a tournament with total distance at most $96k^2(2900k^2 + 375k + 11)$. Since this tournament is uniform, it is a feasible solution to BTTP and BTTP*. We now prove the converse statement. Let $T(k) = 96k^2(2900k^2 + 375k + 11)$. Consider an inter-league tournament between these 36k teams with total travel distance at most $T(k)$. By Lemma 4, $T(k) = \sum_{t \in X_S} ILB_t$. Hence, every team $t \in X_S \cup Y_S$ must travel the shortest possible distance of $ILB_t$ to play all of their games. By Lemma 3, this implies that every team located at $p \in Y_S$ must travel a distance of $ILB_p = k(24z^2 + 3z)$.

By Lemma 2, if each team $p \in Y_S$ travels a distance of $k(24z^2 + 3z)$, then the graph $K_S$ contains a $p$-rooted 4-cycle-cover with exactly 3k $(p, u, b, p)$-cycles, $2k(p, u, c, d, p)$-cycles, and $k(p, u, e, f, p)$-cycles. And by Lemma 1, this implies iff $S$ is satisfiable.

Therefore, we have constructed a uniform inter-league tournament $K_S$ on 36k teams with distance matrix $D_S$ for which the solutions to BTTP and BTTP have total distance $\leq T(k)$ iff the instance $S$ with 2k clauses is satisfiable. This establishes the desired reduction from 3-SAT, proving the NP-hardness of BTTP and BTTP*. Finally, we note that both problems are clearly in NP, since the distance traveled by the teams in $X_S$ is polynomial time. Hence, we conclude that BTTP and BTTP are NP-complete.
Application 1: Japanese Baseball ($n = 6$)

Nippon Professional Baseball (NPB) is Japan’s largest professional sports league. In the NPB, the teams are split into two leagues (i.e., conferences) of six teams, with each team playing 120 intra-league and 24 inter-league games in the regular season. The intra-league problem was analyzed recently (Hoshino and Kawarabayashi 2011b), where the authors developed a multi-round generalization of the TTP based on Dijkstra’s shortest path algorithm and applied it to produce a distance-optimal schedule reducing the total travel distance by over 60000 kilometres (a 25% reduction) as compared to the 2010 NPB intra-league schedule.

We consider the inter-league problem, where the six teams in the NPB Pacific League ($\{p_i : 1 \leq i \leq 6\}$) each play four games against all six teams in the NPB Central League ($\{c_i : 1 \leq i \leq 6\}$), with one two-game set played at the home of the Pacific League team, and the other two-game set played at the home of the Central League team. All inter-league games take place during a five-week stretch between mid-May and mid-June, with no intra-league games occurring during that period. Thus, the NPB inter-league scheduling problem is precisely $BTTP$, for the case $n = 6$.

We determine the $12 \times 12$ distance matrix, representing the distances between each pair of teams from $\{p_1, \ldots, p_6, c_1, \ldots, c_6\}$. The locations of each team’s home stadium is marked in Figure 4. For the actual distance matrix, we refer the reader to our journal paper (Hoshino and Kawarabayashi 2011c). In the 2010 NPB inter-league schedule, the teams traveled a total of 51134 kilometres.

![Figure 4: Location of the 12 teams in the NPB.](image-url)

First, we determine $ILB_{p_i}$ and $ILB_{c_j}$ for each team in the two leagues. By the Triangle Inequality, if some team $t$ plays a single road set sandwiched between two home sets (e.g. HH-RR-HH-R-RR-R), then the time slots can be re-ordered to reduce the total travel distance. Hence, the value of $ILB_t$ occurs when team $t$ plays its six road sets in two blocks of three (e.g. HHH-RRR-HHH-RRR) or three blocks of two (e.g. HHH-RR-H-RR-H).

For each of these two scenarios, we consider all $6! = 720$ ways of permuting the six road games, from which we can determine all possible distances that can be traveled by that team for the given home-road pattern. We check all $2 \times 720$ cases and the minimum distance corresponds to $ILB_t$. We find that in all twelve cases, the value of $ILB_t$ occurs when team $t$ plays its six road sets in two blocks of three, i.e., the minimum-weight $t$-rooted 4-cycle-cover consists of just two cycles. We determine that $\sum ILB_t = \sum ILB_{p_i} + \sum ILB_{c_j} = 16863 + 26077 = 42763$.

Therefore, the optimal solution to $BTTP$ for the NPB distance matrix is a tournament requiring at least 42763 kilometres of total team travel. In Table 4 below, we present a feasible solution with total distance 42950 kilometres.

<table>
<thead>
<tr>
<th></th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
<th>11</th>
<th>12</th>
</tr>
</thead>
<tbody>
<tr>
<td>$p_1$</td>
<td>$c_3$</td>
<td>$c_5$</td>
<td>$c_1$</td>
<td>$c_2$</td>
<td>$c_4$</td>
<td>$c_6$</td>
<td>$c_2$</td>
<td>$c_4$</td>
<td>$c_5$</td>
<td>$c_6$</td>
<td>$c_4$</td>
<td>$c_2$</td>
</tr>
<tr>
<td>$p_2$</td>
<td>$c_5$</td>
<td>$c_3$</td>
<td>$c_1$</td>
<td>$c_2$</td>
<td>$c_4$</td>
<td>$c_6$</td>
<td>$c_2$</td>
<td>$c_4$</td>
<td>$c_5$</td>
<td>$c_6$</td>
<td>$c_4$</td>
<td>$c_2$</td>
</tr>
<tr>
<td>$c_1$</td>
<td>$c_2$</td>
<td>$c_6$</td>
<td>$c_4$</td>
<td>$c_5$</td>
<td>$c_3$</td>
<td>$c_1$</td>
<td>$c_2$</td>
<td>$c_6$</td>
<td>$c_4$</td>
<td>$c_5$</td>
<td>$c_3$</td>
<td>$c_1$</td>
</tr>
<tr>
<td>$p_3$</td>
<td>$c_1$</td>
<td>$c_2$</td>
<td>$c_6$</td>
<td>$c_4$</td>
<td>$c_5$</td>
<td>$c_3$</td>
<td>$c_1$</td>
<td>$c_2$</td>
<td>$c_6$</td>
<td>$c_4$</td>
<td>$c_5$</td>
<td>$c_3$</td>
</tr>
<tr>
<td>$c_6$</td>
<td>$c_4$</td>
<td>$c_2$</td>
<td>$c_1$</td>
<td>$c_2$</td>
<td>$c_6$</td>
<td>$c_4$</td>
<td>$c_2$</td>
<td>$c_1$</td>
<td>$c_2$</td>
<td>$c_6$</td>
<td>$c_4$</td>
<td>$c_2$</td>
</tr>
<tr>
<td>$p_4$</td>
<td>$c_1$</td>
<td>$c_2$</td>
<td>$c_6$</td>
<td>$c_4$</td>
<td>$c_5$</td>
<td>$c_3$</td>
<td>$c_1$</td>
<td>$c_2$</td>
<td>$c_6$</td>
<td>$c_4$</td>
<td>$c_5$</td>
<td>$c_3$</td>
</tr>
<tr>
<td>$c_5$</td>
<td>$c_3$</td>
<td>$c_2$</td>
<td>$c_1$</td>
<td>$c_2$</td>
<td>$c_6$</td>
<td>$c_4$</td>
<td>$c_2$</td>
<td>$c_1$</td>
<td>$c_2$</td>
<td>$c_6$</td>
<td>$c_4$</td>
<td>$c_2$</td>
</tr>
<tr>
<td>$p_5$</td>
<td>$c_1$</td>
<td>$c_2$</td>
<td>$c_6$</td>
<td>$c_4$</td>
<td>$c_5$</td>
<td>$c_3$</td>
<td>$c_1$</td>
<td>$c_2$</td>
<td>$c_6$</td>
<td>$c_4$</td>
<td>$c_5$</td>
<td>$c_3$</td>
</tr>
<tr>
<td>$c_6$</td>
<td>$c_4$</td>
<td>$c_2$</td>
<td>$c_1$</td>
<td>$c_2$</td>
<td>$c_6$</td>
<td>$c_4$</td>
<td>$c_2$</td>
<td>$c_1$</td>
<td>$c_2$</td>
<td>$c_6$</td>
<td>$c_4$</td>
<td>$c_2$</td>
</tr>
</tbody>
</table>

Table 4: Solution to $BTTP$ with total distance 42950 km.

We claim that this solution is optimal. Let $M = 42950 - 42763 = 187$, and let $\Phi$ be a distance-optimal solution to $BTTP$. For each team $t$, let $S^*_t$ be the set of team schedules whose total distance is at most $ILB_t + M$. Note that team $t$’s schedule must appear in $S^*_t$ as otherwise the total distance traveled would exceed $\sum ILB_t + M = 42950$.

As teams $p_5$ and $p_6$ are located quite far away from the other ten teams (see Figure 4), we find that $S^*_{p_5}$ and $S^*_{p_6}$ only consist of schedules where the six road sets are played in two blocks of three. Furthermore, each Central League team $c_j$ must play their road sets against $p_5$ and $p_6$ in two consecutive time slots, as otherwise that team would travel a distance exceeding $ILB_{c_j} + M$, contradicting the optimality of $\Phi$.

Based on these two observations, a simple lemma (Hoshino and Kawarabayashi 2011a) shows that the team schedules for $p_5$ and $p_6$ must have the pattern HH-RR-R-RRR-RH-R, with the six home sets having the following structure for some permutation $(a, b, c, d, e, f)$ of $\{1, 2, 3, 4, 5, 6\}$:

<table>
<thead>
<tr>
<th></th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
<th>11</th>
<th>12</th>
</tr>
</thead>
<tbody>
<tr>
<td>$p_5$</td>
<td>$c_a$</td>
<td>$c_b$</td>
<td>$c_c$</td>
<td>$c_d$</td>
<td>$c_e$</td>
<td>$c_f$</td>
<td>$c_a$</td>
<td>$c_b$</td>
<td>$c_c$</td>
<td>$c_d$</td>
<td>$c_e$</td>
<td>$c_f$</td>
</tr>
<tr>
<td>$p_6$</td>
<td>$c_a$</td>
<td>$c_b$</td>
<td>$c_c$</td>
<td>$c_d$</td>
<td>$c_e$</td>
<td>$c_f$</td>
<td>$c_a$</td>
<td>$c_b$</td>
<td>$c_c$</td>
<td>$c_d$</td>
<td>$c_e$</td>
<td>$c_f$</td>
</tr>
</tbody>
</table>

The forced structure of $p_5$ and $p_6$ significantly reduces the search space. We include this constraint in a simple Maple-software program that generates all possible bipartite tournaments satisfying the conditions of $BTTP$, where the schedule of each team $t$ appears in $S^*_t$ (Hoshino and Kawarabayashi 2011a). Using a Toshiba laptop under Windows with a single 2.10 GHz processor and 2.75 GB RAM, we find 28 optimal solutions in 34716 seconds (just under 10 hours). Each optimal solution (e.g. Table 4) has feasible value 42950, representing a reduction of 8184 kilometres, or 16%, compared to the actual distance traveled by the teams in the 2010 NPB season. Thus, we have solved $BTTP$ for the NPB.

Our algorithm requires 10 hours of processing time. We conjecture that a much faster solution could be generated using a IP, CP, or some hybrid combination of the two.
Application 2: American Basketball \((n = 15)\)

The National Basketball Association (NBA) is one of the world’s most lucrative sports leagues, with over four billion dollars in annual revenue, and an average franchise value of $400 million dollars. There are 15 teams in the Western Conference and 15 teams in the Eastern Conference. Every NBA team plays 82 regular-season games, of which 30 are inter-league (with one home game and one away game against each of the 15 teams from the other conference).

Given that NBA teams play inter-league games, we consider BTTP for this league, where we attempt to find a distance-optimal inter-league tournament. In this theoretical problem, we will assume that all inter-league games take place during a consecutive stretch in the regular season, as is done currently in the Japanese NPB. We will also enforce all the constraints of BTTP, including no team having a home stand or road trip lasting more than 3 games. We note that these strict conditions are not part of the NBA scheduling requirement, as evidenced by the San Antonio Spurs playing 6 consecutive home games followed immediately by 8 consecutive road games during the 2009-10 regular season.

<table>
<thead>
<tr>
<th>Team</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
<th>11</th>
<th>12</th>
<th>13</th>
<th>14</th>
<th>15</th>
</tr>
</thead>
<tbody>
<tr>
<td>BC</td>
<td>BTTP*</td>
<td>15</td>
<td>1</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>6</td>
<td>7</td>
<td>8</td>
<td>9</td>
<td>10</td>
<td>11</td>
<td>12</td>
<td>13</td>
<td>14</td>
</tr>
<tr>
<td>IP</td>
<td>15</td>
<td>BTTP*</td>
<td>1</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>6</td>
<td>7</td>
<td>8</td>
<td>9</td>
<td>10</td>
<td>11</td>
<td>12</td>
<td>13</td>
<td>14</td>
</tr>
<tr>
<td>WW</td>
<td>15</td>
<td>BTTP*</td>
<td>1</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>6</td>
<td>7</td>
<td>8</td>
<td>9</td>
<td>10</td>
<td>11</td>
<td>12</td>
<td>13</td>
<td>14</td>
</tr>
<tr>
<td>BW</td>
<td>15</td>
<td>BTTP*</td>
<td>1</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>6</td>
<td>7</td>
<td>8</td>
<td>9</td>
<td>10</td>
<td>11</td>
<td>12</td>
<td>13</td>
<td>14</td>
</tr>
<tr>
<td>PS</td>
<td>15</td>
<td>BTTP*</td>
<td>1</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>6</td>
<td>7</td>
<td>8</td>
<td>9</td>
<td>10</td>
<td>11</td>
<td>12</td>
<td>13</td>
<td>14</td>
</tr>
</tbody>
</table>

Table 5: A feasible solution for the NBA inter-league problem.

We determine the \(30 \times 30\) NBA distance matrix from an online website\(^1\) that lists the flight distance (in statute miles) between each pair of major cities in North America. From this, we calculate \(ILB_t\) for each team \(t\), giving \(LLB_W = \sum_{t \in W} ILB_t = 251795\), \(LLB_E = \sum_{t \in E} ILB_t = 266137\), and \(TLB = LLB_W + LLB_E = 517932\).

Unlike the 12-team NPB where we could solve BTTP, it appears highly unlikely that we can solve this problem for the 30-team NBA. Nonetheless, we can generate a uniform inter-league tournament (i.e., a solution to BTTP*) whose total distance is close to the trivial lower bound of \(\sum ILB_t\), by determining for each team \(t\) the set of \(t\)-rooted 4-cycle covers containing exactly 5 cycles whose weights are close to \(ILB_t\). Table 5 presents a uniform tournament schedule found by grouping the fifteen teams in each conference into five groups of three, and matching triplets from opposing leagues. This tournament has total distance 537791, which is just 3.8% more than the trivial lower bound of \(\sum ILB_t\).

Our journal paper (Hoshino and Kawarabayashi 2011c) explains all the details, as well as providing the 30 \(\times 30\) distance matrix and the labeling of all 30 teams (e.g., \(PT =\) Portland Trailblazers, \(MB =\) Milwaukee Bucks, etc.).

While we are certain that the trivial lower bound of \(\sum ILB_t\) cannot be achieved for either the BTTP or BTTP*, we conjecture that the 3.8% figure can be reduced using more sophisticated techniques. But how close can we get? We leave this as a challenge for the interested reader.

Acknowledgements

This research has been partially supported by the Japan Society for the Promotion of Science (Grant-in-Aid for Scientific Research), the C & C Foundation, the Kayamori Foundation, and the Inoue Research Award for Young Scientists.

References


\(^1\)http://www.savvy-discounts.com/discount-travel/JavaAirportCalc.html