A Recursive Algorithm to Generate Balanced Weekend Tournaments

Richard Hoshino
Quest University Canada,
Squamish, British Columbia, Canada

Abstract

In this paper, we construct a Balanced Weekend Tournament, motivated by the real-life problem of scheduling an $n$-team double round-robin season schedule for a Canadian university soccer league. In this 6-team league, games are only played on Saturdays and Sundays, with the condition that no team has two road games on any weekend.

The implemented regular-season schedule for $n = 6$ was best-possible, but failed to meet an important “compactness” criterion, as the 10-game tournament required more than five weekends to complete. The motivation for this paper was to determine whether an optimal season schedule, satisfying all of the league’s constraints on compact balanced play, could be constructed for sports leagues with $n > 6$ teams.

We present a simple recursive algorithm to answer this question for all even $n > 6$. As a corollary, our construction gives us an explicit solution to a challenging and well-known graph theory question, namely the problem of decomposing the complete directed graph $K_{2m}$ into $2m - 1$ directed Hamiltonian cycles of length $2m$.

Introduction

Sports scheduling has emerged as a growing field of AI research over the past two decades (Kendall et al. 2010), especially since the introduction of the Traveling Tournament Problem by the head schedulers of Major League Baseball (Easton, Nemhauser, and Trick 2001). Integer programming, constraint programming, metaheuristics, and hybrid methods have been successfully applied to solve complex problems in sports scheduling (Goerigk et al. 2014).

The majority of sports scheduling research deals with the organization and optimization of professional sports leagues, as these leagues generate worldwide audiences and are multi-billion dollar industries. However, a much larger percentage of individuals competing in sports leagues are students playing on varsity teams against other schools, with their games fixed in certain time slots (e.g. weekends).

The author is employed by a small Canadian university competing in the Pacific Western Athletic Association (PACWEST), consisting of six soccer teams playing a season-long double round-robin tournament such as this:

<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>
</tr>
</thead>
<tbody>
<tr>
<td>$t_1$</td>
<td>$t_6$</td>
<td>$t_5$</td>
<td>$t_4$</td>
<td>$t_2$</td>
<td>$t_3$</td>
<td>$t_5$</td>
<td>$t_2$</td>
<td>$t_4$</td>
<td>$t_1$</td>
<td>$t_3$</td>
</tr>
<tr>
<td>$t_2$</td>
<td>$t_5$</td>
<td>$t_6$</td>
<td>$t_4$</td>
<td>$t_1$</td>
<td>$t_5$</td>
<td>$t_6$</td>
<td>$t_4$</td>
<td>$t_1$</td>
<td>$t_5$</td>
<td>$t_2$</td>
</tr>
<tr>
<td>$t_3$</td>
<td>$t_4$</td>
<td>$t_1$</td>
<td>$t_6$</td>
<td>$t_5$</td>
<td>$t_2$</td>
<td>$t_4$</td>
<td>$t_1$</td>
<td>$t_6$</td>
<td>$t_5$</td>
<td>$t_2$</td>
</tr>
<tr>
<td>$t_4$</td>
<td>$t_3$</td>
<td>$t_5$</td>
<td>$t_2$</td>
<td>$t_6$</td>
<td>$t_3$</td>
<td>$t_5$</td>
<td>$t_2$</td>
<td>$t_6$</td>
<td>$t_3$</td>
<td>$t_5$</td>
</tr>
<tr>
<td>$t_5$</td>
<td>$t_2$</td>
<td>$t_4$</td>
<td>$t_3$</td>
<td>$t_1$</td>
<td>$t_2$</td>
<td>$t_4$</td>
<td>$t_3$</td>
<td>$t_1$</td>
<td>$t_2$</td>
<td>$t_4$</td>
</tr>
<tr>
<td>$t_6$</td>
<td>$t_1$</td>
<td>$t_2$</td>
<td>$t_3$</td>
<td>$t_4$</td>
<td>$t_5$</td>
<td>$t_1$</td>
<td>$t_2$</td>
<td>$t_3$</td>
<td>$t_4$</td>
<td>$t_5$</td>
</tr>
</tbody>
</table>

Table 1: A 5-weekend 6-team tournament

Table 1 illustrates an example of a double round-robin tournament, where each team plays in $2(n - 1)$ different time slots, with one home game and one road game against each of the other $n - 1$ teams. Home games are marked in bold, e.g. $t_1$ is the home team versus $t_6$ in slot #1.

In the PACWEST league, the student-athletes on the $n = 6$ teams complete their regular-season games over $n - 1 = 5$ weekends, due to academic priorities during the week that prevent weekday games from being scheduled.

The major weakness with this schedule is that four teams, namely $t_1$, $t_2$, $t_3$, $t_6$, have one weekend where they play two road games (e.g. $t_1$ in time slots #5 and #6). Playing back-to-back road games is problematic for the teams, given the academic and social commitments of the student-athletes on weekends. There is sometimes the additional cost of having to book a hotel room on the Saturday night, depending on the length of time it would take to make an additional back-and-forth trip in a single weekend.

To mitigate this issue, the PACWEST league approached the author, who had previously helped the Japanese professional baseball league reduce their carbon emissions by developing a regular-season schedule to minimize total travel distance (Hesse 2012).

The challenge for this sports scheduling problem was to have each team play $2(n - 1)$ games over $(n - 1)$ weekends where every team plays at most one road game in a single weekend, thus ensuring that the student-athletes could make a short trip on either the Saturday or Sunday, and return home immediately following the game. For obvious reasons, such a schedule is applicable to more than just an inter-university soccer league in British Columbia, Canada; given the popularity of soccer (i.e., football) throughout the world, such a schedule would be relevant to any $n$-team league playing back-to-back games on weekends.
The paper proceeds as follows: we first define the n-team Balanced Weekend Tournament Problem (BWTP). Though no solution exists for \( n \leq 6 \), we provide a recursive algorithm to generate a solution \( S_n \) to the BWTP for all even \( n \geq 8 \). As a corollary, we show that our construction provides an explicit solution to the Hamiltonian cycle decomposition problem of the complete directed graph \( K_n^* \), which works for all even \( n \geq 8 \).

Inspired by the \( p \)-norm in functional analysis, we then define a similar \( p \)-norm to measure the effectiveness of a feasible solution, to show that some schedules are more “balanced” than others. We prove that our recursive construction generates an \( n \)-team schedule \( S_n \) for which \( \| S_n \|_1 < \frac{2}{n} n^2 \) and \( \| S_n \|_p < n^{1+1/p} \) for \( p \geq 2 \), upper bounds that we conjecture are close to optimal. Finally, at the end of each section of this paper, we conclude with an open problem.

**Problem Statement**

The schedulers of the PACWEST league requested a double round-robin schedule satisfying the following constraints. Here are the criteria, in decreasing order of importance.

(a) **Each-Venue**: Each pair of teams plays twice, once in each other’s home venue.
(b) **Each-Half**: Each pair of teams plays one game in the first half of the season, and one game in the second half.
(c) **No-Repeat**: A team cannot play against the same opponent in two consecutive games.
(d) **No-Two-Road**: Each team plays at most one road game each weekend.
(e) **Compactness**: The tournament takes place over \( n - 1 \) weekends, with two games each weekend.

We note that condition (e) requires \( n \) to be even, since every team must play one game in each time slot.

It is straightforward to create a schedule that satisfies all the conditions except for (c). To do this, we apply the well-known canonical schedule (de Werra 1981) to generate a single round-robin tournament for the first half of the season, and use the “English construction” (Kendall et al. 2010) to produce the second half. Here is the construction for \( n = 6 \).

\[
\begin{array}{cccccccc}
\text{Team} & 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 \\
\hline
1 & t_1 & t_5 & t_3 & t_7 & t_6 & t_5 & t_3 & t_7 \\
2 & t_2 & t_6 & t_0 & t_4 & t_3 & t_6 & t_0 & t_4 \\
3 & t_3 & t_1 & t_0 & t_3 & t_1 & t_0 & t_3 & t_1 \\
4 & t_4 & t_2 & t_6 & t_1 & t_3 & t_5 & t_2 & t_6 \\
5 & t_5 & t_4 & t_2 & t_6 & t_1 & t_3 & t_5 & t_2 \\
6 & t_6 & t_5 & t_4 & t_2 & t_6 & t_1 & t_3 & t_5 \\
\end{array}
\]

Table 2: A tournament violating the No-Repeat condition

A simple computer program proves that there exist no 6-team tournaments satisfying all five conditions.

Because of this, the PACWEST decided to adopt the almost-optimal 10-game 6-weekend schedule given in Table 3, which met all of the criteria except for (e). In this schedule, two weekends have only a Saturday game and no Sunday game, corresponding to slots 1 and 6.

\[
\begin{array}{cccccccc}
\text{Team} & 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 \\
\hline
1 & t_1 & t_4 & t_0 & t_3 & t_5 & t_6 & t_3 & t_2 \\
2 & t_2 & t_5 & t_6 & t_3 & t_1 & t_8 & t_5 & t_0 \\
3 & t_3 & t_1 & t_6 & t_5 & t_2 & t_4 & t_1 & t_8 \\
4 & t_4 & t_2 & t_1 & t_3 & t_5 & t_6 & t_4 & t_7 \\
5 & t_5 & t_3 & t_4 & t_1 & t_3 & t_6 & t_5 & t_2 \\
6 & t_6 & t_4 & t_5 & t_2 & t_1 & t_3 & t_6 & t_5 \\
\end{array}
\]

Table 3: The 2016-17 PACWEST Soccer Schedule

For each pair of teams \( t_i \) and \( t_j \), let \( h(i, j) \) be the slot in which \( t_i \) plays a home game against \( t_j \), and \( r(i, j) \) be the slot in which \( t_i \) plays a road game against \( t_j \). Notice that \( |h(i, j) - r(i, j)| = 5 \) for all \( 1 \leq i < j \leq 6 \). This is known as a mirrored schedule (Ribeiro and Urrutia 2004), and there has been much analysis done on the construction of \( n \)-team schedules that require \( |h(i, j) - r(i, j)| \) to always equal \( n - 1 \), for all \( (i, j) \). Later in the paper, we explain why a mirrored schedule cannot satisfy conditions (d) and (e).

While the PACWEST scheduling problem was solved and implemented, there was little satisfaction as no double round-robin schedule satisfied all five constraints. But inspired by this, we pose the Balanced Weekend Tournament Problem (BWTP), on whether there exists a feasible tournament satisfying these five conditions for \( n > 6 \).

Since there are \( n \) teams with \( \frac{n}{2} \) teams playing at home and \( \frac{n}{2} \) teams playing on the road in any given time slot, condition (d) implies that each team must play exactly one home game and one road game every weekend.

A computer search generated numerous solutions to the BWTP for the case \( n = 8 \), including the following:

\[
\begin{array}{cccccccc}
\text{Team} & 1 & 2 & 3 & 4 & 5 & 6 & 7 \ \\
\hline
1 & t_1 & t_6 & t_5 & t_3 & t_2 & t_4 & t_7 \\
2 & t_2 & t_3 & t_6 & t_4 & t_7 & t_5 & t_1 \\
3 & t_3 & t_5 & t_2 & t_4 & t_7 & t_6 & t_3 \\
4 & t_4 & t_6 & t_1 & t_2 & t_7 & t_3 & t_5 \\
5 & t_5 & t_7 & t_1 & t_2 & t_3 & t_6 & t_4 \\
6 & t_6 & t_2 & t_3 & t_7 & t_4 & t_1 & t_5 \\
7 & t_7 & t_1 & t_3 & t_6 & t_5 & t_4 & t_2 \\
8 & t_1 & t_2 & t_3 & t_4 & t_5 & t_6 & t_7 \\
\end{array}
\]

By inspection, we can check that all five criteria of the BWTP are satisfied in the above schedule. To produce this schedule, we enumerate all possible single round-robin tournaments (slots 1 to 7) and determine for each option all possible ways to assign home and away slots so that each team plays one home game and one road game each weekend. We then check whether there exists a complementary single round-robin schedule (slots 8 to 14) for which \( h(i, j) \leq 7 \) if and only if \( r(i, j) \geq 8 \) for all \( (i, j) \) with \( 1 \leq i < j \leq 8 \).

In the next two sections, we present recursive algorithms that generate a solution to the BWTP for the cases \( n = 2k \) and \( n = 2k - 2 \), given a feasible solution for \( n = k \). Combined with the demonstration that there exist feasible solutions for the cases \( n = 10 \) and \( n = 12 \), the above two algorithms solve the BWTP for all even values of \( n \geq 8 \).
Generating $S_{2k}$ from $S_k$

Let $n = k$ be even. Suppose that $S_k$ is a feasible solution to the BWTP, with teams $t_1, t_2, \ldots, t_k$. We now construct $S_{2k}$, a feasible solution to the BWTP for the case $n = 2k$.

First we create a second copy of the $k$ teams, and label them $u_1, u_2, \ldots, u_k$.

In slots $1 \leq s \leq k$, $t_i$ plays against $u_{s+i-1}$, with all arithmetic calculated mod $k$. If $s$ is odd, then $t_i$ is the home team, and if $s$ is even, then $t_i$ is the away team.

Table 5 illustrates this construction for the case $k = 8$, where the first 8 slots are given for the schedule $S_{16}$.

<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>
</tr>
</thead>
<tbody>
<tr>
<td>$t_1$</td>
<td>$u_1$</td>
<td>$u_2$</td>
<td>$u_3$</td>
<td>$u_4$</td>
<td>$u_5$</td>
<td>$u_6$</td>
<td>$u_7$</td>
<td>$u_8$</td>
</tr>
<tr>
<td>$t_2$</td>
<td>$u_2$</td>
<td>$u_3$</td>
<td>$u_4$</td>
<td>$u_5$</td>
<td>$u_6$</td>
<td>$u_7$</td>
<td>$u_8$</td>
<td>$u_1$</td>
</tr>
<tr>
<td>$t_3$</td>
<td>$u_3$</td>
<td>$u_4$</td>
<td>$u_5$</td>
<td>$u_6$</td>
<td>$u_7$</td>
<td>$u_8$</td>
<td>$u_1$</td>
<td>$u_2$</td>
</tr>
<tr>
<td>$t_4$</td>
<td>$u_4$</td>
<td>$u_5$</td>
<td>$u_6$</td>
<td>$u_7$</td>
<td>$u_8$</td>
<td>$u_1$</td>
<td>$u_2$</td>
<td>$u_3$</td>
</tr>
<tr>
<td>$t_5$</td>
<td>$u_5$</td>
<td>$u_6$</td>
<td>$u_7$</td>
<td>$u_8$</td>
<td>$u_1$</td>
<td>$u_2$</td>
<td>$u_3$</td>
<td>$u_4$</td>
</tr>
<tr>
<td>$t_6$</td>
<td>$u_6$</td>
<td>$u_7$</td>
<td>$u_8$</td>
<td>$u_1$</td>
<td>$u_2$</td>
<td>$u_3$</td>
<td>$u_4$</td>
<td>$u_5$</td>
</tr>
<tr>
<td>$t_7$</td>
<td>$u_7$</td>
<td>$u_8$</td>
<td>$u_1$</td>
<td>$u_2$</td>
<td>$u_3$</td>
<td>$u_4$</td>
<td>$u_5$</td>
<td>$u_6$</td>
</tr>
<tr>
<td>$t_8$</td>
<td>$u_8$</td>
<td>$u_1$</td>
<td>$u_2$</td>
<td>$u_3$</td>
<td>$u_4$</td>
<td>$u_5$</td>
<td>$u_6$</td>
<td>$u_7$</td>
</tr>
</tbody>
</table>

Table 5: Slots 1 to 8 of $S_{16}$

The schedule for slots $2k + 1 \leq s \leq 3k$ is identical, except the home and away teams are switched. Table 6 illustrates this construction for the case $k = 8$.

<table>
<thead>
<tr>
<th>Team</th>
<th>17</th>
<th>18</th>
<th>19</th>
<th>20</th>
<th>21</th>
<th>22</th>
<th>23</th>
<th>24</th>
</tr>
</thead>
<tbody>
<tr>
<td>$t_1$</td>
<td>$u_1$</td>
<td>$u_2$</td>
<td>$u_3$</td>
<td>$u_4$</td>
<td>$u_5$</td>
<td>$u_6$</td>
<td>$u_7$</td>
<td>$u_8$</td>
</tr>
<tr>
<td>$t_2$</td>
<td>$u_2$</td>
<td>$u_3$</td>
<td>$u_4$</td>
<td>$u_5$</td>
<td>$u_6$</td>
<td>$u_7$</td>
<td>$u_8$</td>
<td>$u_1$</td>
</tr>
<tr>
<td>$t_3$</td>
<td>$u_3$</td>
<td>$u_4$</td>
<td>$u_5$</td>
<td>$u_6$</td>
<td>$u_7$</td>
<td>$u_8$</td>
<td>$u_1$</td>
<td>$u_2$</td>
</tr>
<tr>
<td>$t_4$</td>
<td>$u_4$</td>
<td>$u_5$</td>
<td>$u_6$</td>
<td>$u_7$</td>
<td>$u_8$</td>
<td>$u_1$</td>
<td>$u_2$</td>
<td>$u_3$</td>
</tr>
<tr>
<td>$t_5$</td>
<td>$u_5$</td>
<td>$u_6$</td>
<td>$u_7$</td>
<td>$u_8$</td>
<td>$u_1$</td>
<td>$u_2$</td>
<td>$u_3$</td>
<td>$u_4$</td>
</tr>
<tr>
<td>$t_6$</td>
<td>$u_6$</td>
<td>$u_7$</td>
<td>$u_8$</td>
<td>$u_1$</td>
<td>$u_2$</td>
<td>$u_3$</td>
<td>$u_4$</td>
<td>$u_5$</td>
</tr>
<tr>
<td>$t_7$</td>
<td>$u_7$</td>
<td>$u_8$</td>
<td>$u_1$</td>
<td>$u_2$</td>
<td>$u_3$</td>
<td>$u_4$</td>
<td>$u_5$</td>
<td>$u_6$</td>
</tr>
<tr>
<td>$t_8$</td>
<td>$u_8$</td>
<td>$u_1$</td>
<td>$u_2$</td>
<td>$u_3$</td>
<td>$u_4$</td>
<td>$u_5$</td>
<td>$u_6$</td>
<td>$u_7$</td>
</tr>
</tbody>
</table>

Table 6: Slots 17 to 24 of $S_{16}$

In slots $k + 1 \leq s \leq 2k$, we make two copies of the first $k$ columns of $S_k$. Specifically, if $t_i$ plays at $t_j$ in slot $s$, then in our schedule $S_{2k}$, $t_i$ plays at $t_j$ in slot $s + k$, and $u_i$ plays at $u_j$ in slot $s + k$.

Table 7 illustrates this construction for the case $k = 8$, where we construct slots 9 to 16 of $S_{16}$ using the schedule $S_8$ given in Table 4.

<table>
<thead>
<tr>
<th>Team</th>
<th>9</th>
<th>10</th>
<th>11</th>
<th>12</th>
<th>13</th>
<th>14</th>
<th>15</th>
<th>16</th>
</tr>
</thead>
<tbody>
<tr>
<td>$t_1$</td>
<td>$t_6$</td>
<td>$t_4$</td>
<td>$t_8$</td>
<td>$t_2$</td>
<td>$t_4$</td>
<td>$t_7$</td>
<td>$t_5$</td>
<td>$t_3$</td>
</tr>
<tr>
<td>$t_2$</td>
<td>$t_3$</td>
<td>$t_7$</td>
<td>$t_5$</td>
<td>$t_4$</td>
<td>$t_3$</td>
<td>$t_6$</td>
<td>$t_5$</td>
<td>$t_8$</td>
</tr>
<tr>
<td>$t_3$</td>
<td>$t_5$</td>
<td>$t_7$</td>
<td>$t_1$</td>
<td>$t_2$</td>
<td>$t_4$</td>
<td>$t_7$</td>
<td>$t_6$</td>
<td>$t_1$</td>
</tr>
<tr>
<td>$t_4$</td>
<td>$t_7$</td>
<td>$t_2$</td>
<td>$t_5$</td>
<td>$t_3$</td>
<td>$t_6$</td>
<td>$t_8$</td>
<td>$t_2$</td>
<td>$t_1$</td>
</tr>
<tr>
<td>$t_5$</td>
<td>$t_3$</td>
<td>$t_7$</td>
<td>$t_4$</td>
<td>$t_6$</td>
<td>$t_3$</td>
<td>$t_7$</td>
<td>$t_5$</td>
<td>$t_4$</td>
</tr>
<tr>
<td>$t_6$</td>
<td>$t_1$</td>
<td>$t_8$</td>
<td>$t_7$</td>
<td>$t_5$</td>
<td>$t_2$</td>
<td>$t_4$</td>
<td>$t_3$</td>
<td>$t_7$</td>
</tr>
<tr>
<td>$t_7$</td>
<td>$t_4$</td>
<td>$t_5$</td>
<td>$t_6$</td>
<td>$t_8$</td>
<td>$t_3$</td>
<td>$t_1$</td>
<td>$t_2$</td>
<td>$t_6$</td>
</tr>
<tr>
<td>$t_8$</td>
<td>$t_2$</td>
<td>$t_6$</td>
<td>$t_1$</td>
<td>$t_7$</td>
<td>$t_5$</td>
<td>$t_4$</td>
<td>$t_2$</td>
<td>$t_1$</td>
</tr>
</tbody>
</table>

Table 7: Slots 9 to 16 of $S_{16}$

Finally, in slots $3k + 1 \leq s \leq 4k - 2$, we make two copies of the last $k - 2$ columns of $S_k$. Specifically, if $t_i$ plays at $t_j$ in slot $s$, then in our schedule $S_{2k}$, $t_i$ plays at $t_j$ in slot $s + 2k$, and $u_i$ plays at $u_j$ in slot $s + 2k$. This is illustrated in Table 8 below, once again for the case $k = 8$.

<table>
<thead>
<tr>
<th>Team</th>
<th>25</th>
<th>26</th>
<th>27</th>
<th>28</th>
<th>29</th>
<th>30</th>
</tr>
</thead>
<tbody>
<tr>
<td>$t_1$</td>
<td>$t_6$</td>
<td>$t_4$</td>
<td>$t_8$</td>
<td>$t_2$</td>
<td>$t_7$</td>
<td>$t_5$</td>
</tr>
<tr>
<td>$t_2$</td>
<td>$t_3$</td>
<td>$t_7$</td>
<td>$t_5$</td>
<td>$t_4$</td>
<td>$t_6$</td>
<td>$t_7$</td>
</tr>
<tr>
<td>$t_3$</td>
<td>$t_5$</td>
<td>$t_2$</td>
<td>$t_4$</td>
<td>$t_7$</td>
<td>$t_6$</td>
<td>$t_8$</td>
</tr>
<tr>
<td>$t_4$</td>
<td>$t_2$</td>
<td>$t_1$</td>
<td>$t_4$</td>
<td>$t_5$</td>
<td>$t_3$</td>
<td>$t_6$</td>
</tr>
<tr>
<td>$t_5$</td>
<td>$t_3$</td>
<td>$t_7$</td>
<td>$t_2$</td>
<td>$t_1$</td>
<td>$t_6$</td>
<td>$t_8$</td>
</tr>
<tr>
<td>$t_6$</td>
<td>$t_1$</td>
<td>$t_8$</td>
<td>$t_3$</td>
<td>$t_5$</td>
<td>$t_2$</td>
<td>$t_4$</td>
</tr>
<tr>
<td>$t_7$</td>
<td>$t_7$</td>
<td>$t_6$</td>
<td>$t_8$</td>
<td>$t_1$</td>
<td>$t_2$</td>
<td>$t_4$</td>
</tr>
<tr>
<td>$t_8$</td>
<td>$t_8$</td>
<td>$t_7$</td>
<td>$t_6$</td>
<td>$t_5$</td>
<td>$t_4$</td>
<td>$t_2$</td>
</tr>
</tbody>
</table>

Table 8: Slots 25 to 30 of $S_{16}$

From this construction, we have generated a tournament with $4k - 2$ slots, with each team having a different opponent in each time slot.

We complete the construction by replacing each $u_i$ with $t_{k+i}$, for each $1 \leq i \leq k$. This gives us our solution $S_{2k}$.

In the next section, we recursively construct our solution $S_{2k-2}$ using the known solution $S_k$. 

6197
Generating $S_{2k-2}$ from $S_k$

Label the teams $t_1, t_2, \ldots, t_{k-1}, u_1, u_2, \ldots, u_{k-1}$. Table 9 illustrates this construction for the case $k = 8$, with $S_{14}$ recursively generated from our schedule $S_8$ in Table 4.

<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>
</tr>
</thead>
<tbody>
<tr>
<td>$t_1$</td>
<td>$u_2$</td>
<td>$u_3$</td>
<td>$u_4$</td>
<td>$u_5$</td>
<td>$u_6$</td>
<td>$u_7$</td>
<td>$u_8$</td>
<td>$t_1$</td>
<td>$t_2$</td>
<td>$t_3$</td>
<td>$t_4$</td>
<td>$t_5$</td>
<td>$t_6$</td>
<td>$t_7$</td>
</tr>
<tr>
<td>$t_2$</td>
<td>$u_3$</td>
<td>$u_4$</td>
<td>$u_5$</td>
<td>$u_6$</td>
<td>$u_7$</td>
<td>$u_8$</td>
<td>$t_2$</td>
<td>$t_1$</td>
<td>$t_3$</td>
<td>$t_4$</td>
<td>$t_5$</td>
<td>$t_6$</td>
<td>$t_7$</td>
<td>$t_8$</td>
</tr>
<tr>
<td>$t_3$</td>
<td>$u_4$</td>
<td>$u_5$</td>
<td>$u_6$</td>
<td>$u_7$</td>
<td>$u_8$</td>
<td>$t_3$</td>
<td>$t_2$</td>
<td>$t_1$</td>
<td>$t_4$</td>
<td>$t_5$</td>
<td>$t_6$</td>
<td>$t_7$</td>
<td>$t_8$</td>
<td>$t_9$</td>
</tr>
<tr>
<td>$t_4$</td>
<td>$u_5$</td>
<td>$u_6$</td>
<td>$u_7$</td>
<td>$u_8$</td>
<td>$t_4$</td>
<td>$t_3$</td>
<td>$t_2$</td>
<td>$t_1$</td>
<td>$t_5$</td>
<td>$t_6$</td>
<td>$t_7$</td>
<td>$t_8$</td>
<td>$t_9$</td>
<td>$t_{10}$</td>
</tr>
<tr>
<td>$t_5$</td>
<td>$u_6$</td>
<td>$u_7$</td>
<td>$u_8$</td>
<td>$t_5$</td>
<td>$t_4$</td>
<td>$t_3$</td>
<td>$t_2$</td>
<td>$t_1$</td>
<td>$t_6$</td>
<td>$t_7$</td>
<td>$t_8$</td>
<td>$t_9$</td>
<td>$t_{10}$</td>
<td>$t_{11}$</td>
</tr>
<tr>
<td>$t_6$</td>
<td>$u_7$</td>
<td>$u_8$</td>
<td>$t_6$</td>
<td>$t_5$</td>
<td>$t_4$</td>
<td>$t_3$</td>
<td>$t_2$</td>
<td>$t_1$</td>
<td>$t_7$</td>
<td>$t_8$</td>
<td>$t_9$</td>
<td>$t_{10}$</td>
<td>$t_{11}$</td>
<td>$t_{12}$</td>
</tr>
<tr>
<td>$t_7$</td>
<td>$u_8$</td>
<td>$t_7$</td>
<td>$t_6$</td>
<td>$t_5$</td>
<td>$t_4$</td>
<td>$t_3$</td>
<td>$t_2$</td>
<td>$t_1$</td>
<td>$t_8$</td>
<td>$t_9$</td>
<td>$t_{10}$</td>
<td>$t_{11}$</td>
<td>$t_{12}$</td>
<td>$t_{13}$</td>
</tr>
</tbody>
</table>

Table 9: $S_{14}$, a solution to the BWTP for $n = 14$

Proof of Correctness

We now justify that the two constructions given in the previous section are indeed feasible, i.e., that $S_{2k-2}$ and $S_{2k}$ are solutions to the Balanced Weekend Tournament Problem.

Consider $S_k$, a solution to the BWTP for the case $n = k$ satisfying the five conditions given in Section 2. For each $1 \leq i < j \leq k$, let $h(i,j)$ be the slot in which $t_i$ plays a home game against $t_j$, and $r(i,j)$ be the slot in which $t_i$ plays a road game against $t_j$.

As $S_k$ is a feasible schedule, we know that the Each-Venue condition is satisfied, i.e., $h(i,j)$ and $r(i,j)$ exist for all pairs $(i,j)$. Furthermore, the Each-Half condition necessitates that $1 \leq h(i,j) \leq k - 1$ if and only if $k \leq r(i,j) \leq 2k - 2$, and the No-Repeat condition implies that $|h(i,j) - r(i,j)| \geq 2$ for all $i$ and $j$.

We first notice that in both $S_{2k-2}$ and $S_{2k}$, the Compactness condition is trivially satisfied. For the No-Two-Road condition, we notice that every team plays exactly one home game and one road game each weekend, in each of our four “blocks”. For example, in $S_{2k-2}$, the four blocks are $s \in \{1, k-2\}$, $s \in \{k-1, 2k-2\}$, $s \in \{2k-1, 3k-4\}$, and $s \in \{3k-3, 4k-6\}$. The first and third blocks are balanced as we used parity to determine the home and away assignments, while the second and fourth blocks are guaranteed to be balanced as they were recursively constructed from $S_k$.

We have now dealt with conditions (d) and (e). To justify that conditions (a), (b), (c) are satisfied in $S_{2k}$, we define $h'(i,j)$ and $r'(i,j)$ for each $1 \leq i < j \leq 2k$.

From our construction for the first and third blocks of $S_{2k}$, for all $1 \leq i < j \leq k$ and $k+1 \leq j \leq 2k$, we have $h'(i,j) = (j - i \mod k) + 1$ and $r'(i,j) = (j - i \mod k) + 2k + 1$, or $r'(i,j) = (j - i \mod k) + 1$ and $h'(i,j) = (j - i \mod k) + 2k + 1$. In both cases, conditions (a), (b), (c) are satisfied, since $|h'(i,j) - r'(i,j)| = 2k$.

From our construction for the second and fourth blocks of $S_{2k}$, we have for all $1 \leq i < j \leq k$, $h'(i,j) = h(i,j) + k$ if $1 \leq h(i,j) \leq k$ and $h'(i,j) = h(i,j) + 2k$ if $k+1 \leq h(i,j) \leq 2k - 2$. We also have the same equations relating $r'(i,j)$ to $r(i,j)$. As $S_k$ satisfies condition (b), if $h'(i,j) = h(i,j) + k$, then we must have $r'(i,j) = r(i,j) + 2k$, and vice-versa.

Therefore, for each pair $(i,j)$ with $1 \leq i < j \leq k$, we have $|h'(i,j) - r'(i,j)| = |h(i,j) - r(i,j)| + k$, with the exception of the $k$ pairs $(i,j)$ for which $h(i,j) = k$ or $r(i,j) = k$. In these cases, we have $h'(i,j) = h(i,j) + k$ and $r'(i,j) = r(i,j) + k$, implying that $|h'(i,j) - r'(i,j)| = |h(i,j) - r(i,j)|$.

And if $k+1 \leq i < j \leq 2k$, the same argument shows that $|h'(i,j) - r'(i,j)| = |h(i-k, j-k) - r(i-k, j-k)|$ or $|h'(i,j) - r'(i,j)| = |h(i-k, j-k) - r(i-k, j-k) + k|$, since the indices $t_{k+1}, \ldots, t_{2k}$ are simply $u_1, \ldots, u_k$, and reduces to the case in the previous two paragraphs.

From these equations, we see that conditions (a), (b), (c) are satisfied for all $1 \leq i < j \leq 2k$.

Similarly, we can prove that the three conditions (a), (b), (c) are satisfied in the schedule $S_{2k-2}$, for all $1 \leq i < j \leq 2k - 2$. Since the case analysis is virtually identical to the $S_{2k}$ case, we omit the details.
To complete the proof that the BWTP has a solution for all $n \geq 8$, it suffices to find a solution for the base cases $n = 8$, $n = 10$, and $n = 12$. The case $n = 8$ was found in Table 4. The next two tables demonstrate computer-generated solutions for $n = 10$ and $n = 12$, thus completing our proof.

**Table 10:** $S_{10}$, a solution to the BWTP for $n = 10$

<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>
</tr>
</thead>
<tbody>
<tr>
<td>$t_1$</td>
<td>$t_2$</td>
<td>$t_3$</td>
<td>$t_4$</td>
<td>$t_5$</td>
<td>$t_6$</td>
<td>$t_7$</td>
<td>$t_8$</td>
<td>$t_9$</td>
<td>$t_{10}$</td>
<td>$t_{11}$</td>
<td>$t_{12}$</td>
<td></td>
</tr>
<tr>
<td>$t_{10}$</td>
<td>$t_{11}$</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

(Note: for improved readability, we replaced $t_{10}$ with $t_0$.)

**Table 11:** $S_{12}$, a solution to the BWTP for $n = 12$

<table>
<thead>
<tr>
<th>Team</th>
<th>13</th>
<th>14</th>
<th>15</th>
<th>16</th>
<th>17</th>
<th>18</th>
<th>19</th>
<th>20</th>
<th>21</th>
<th>22</th>
</tr>
</thead>
<tbody>
<tr>
<td>$t_1$</td>
<td>$t_{13}$</td>
<td>$t_{14}$</td>
<td>$t_{15}$</td>
<td>$t_{16}$</td>
<td>$t_{17}$</td>
<td>$t_{18}$</td>
<td>$t_{19}$</td>
<td>$t_{20}$</td>
<td>$t_{21}$</td>
<td></td>
</tr>
<tr>
<td>$t_{14}$</td>
<td>$t_{15}$</td>
<td>$t_{16}$</td>
<td>$t_{17}$</td>
<td>$t_{18}$</td>
<td>$t_{19}$</td>
<td>$t_{20}$</td>
<td>$t_{21}$</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Although this recursive construction generates a valid solution $S_n$ for all $n \geq 8$, perhaps there is a more elegant solution that can be determined combinatorially or geometrically. This motivates the first open problem of this paper.

**Problem #1:** Determine a “canonical” schedule $S_n$ for which a simple function $f(i,j)$ determines the slot in which $t_i$ plays a home game against $t_j$, where the resulting schedule $S_n$ is a valid solution to the $n$-team BWTP.

---

**Hamiltonian Decompositions**

Let $G$ be a graph with $n$ vertices. We say that $G$ is Hamiltonian if it contains a Hamilton cycle, i.e., a cycle passing through each of the $n$ vertices of $G$. Hamiltonicity is one of the most well-studied concepts in graph theory, due to its applications to the Traveling Salesman Problem. Deciding whether a given graph $G$ is Hamiltonian is one of Karp’s 21 celebrated NP-complete problems (Karp 1972).

A generalization of Hamiltonicity is to decide whether a given graph $G$ has a Hamiltonian decomposition, i.e., a partition of the edge set of $G$ into disjoint Hamilton cycles.

For example, the complete graph $K_5$ has a Hamiltonian decomposition, since $K_5$ can be decomposed into the cycles $1-2-3-4-5$ and $1-3-5-2-4$. To make our notation easier, we will represent these cycles as $((12345))$ and $((35241))$, respectively.

Clearly, a Hamiltonian decomposition is only possible when every vertex of $G$ has degree $n$, for some even $n$. Since the complete graph $K_n$ has degree $n-1$, a natural question is to determine the odd values of $n$ for which $K_n$ has a Hamiltonian decomposition. In 1890, Walecki demonstrated an explicit construction for all odd $n$ (Alsipch 2008).

We can generalize this problem by considering Hamiltonian decompositions of directed complete graphs $K^*_n$, where all $n(n-1)$ directed edges, or arcs, are drawn. For all odd $n$, $K^*_n$ must have a Hamiltonian decomposition by Walecki’s Theorem; for example, in the case $n = 5$ above, we have the decomposition $(12345), (15432), (13524), (14253)$, which we get by turning each of the $n$-cycles in our undirected $K_n$ decomposition into two directed $n$-cycles.

For even $n$, the problem of whether $K^*_n$ has a Hamiltonian decomposition was a long-standing open problem in graph theory. A partial solution showed that a Hamiltonian decomposition does not exist for $n = 4$ or $n = 6$ but does for each $8 \leq n \leq 18$ (Bermont and Faber 1976).

By considering four possible cases for $n$ (i.e., their remainder mod 8), Tillson used difference sequences to prove the existence of a Hamiltonian decomposition of $K^*_n$ for all even values of $n \geq 8$ (Tillson 1980).

As the Tillson construction is both lengthy and complex, a natural question is whether this Hamiltonian decomposition of $K^*_n$ could be generated more simply. We answer this question in the affirmative, as a corollary of our recursive construction to the Balanced Weekend Tournament Problem.

Label the $n$ vertices of $K^*_n$ as $t_1, t_2, \ldots, t_n$, and let $S_n$ be a solution to the $n$-team BWTP. Consider the $\frac{n}{2} + \frac{n}{2} = n$ matches played on any weekend, i.e., time slots $2s-1$ and $2s$ for each $1 \leq s \leq n-1$. If $t_i$ plays at the home of $t_j$, this corresponds to the directed edge $t_i \rightarrow t_j$ in $K^*_n$. 

6199
Thus, in Table 4, the games in Slot #1 are represented by the edges $t_2 \to t_8$, $t_3 \to t_5$, $t_6 \to t_1$, and $t_7 \to t_4$, and the games in Slot #2 are represented by the edges $t_1 \to t_3$, $t_4 \to t_2$, $t_5 \to t_7$, and $t_8 \to t_6$.

By the definition of an $n$-team Balanced Weekend Tournament, every team plays one home game and one road game each weekend, i.e., the $n$ edges in slots #1 and #2 must combine to form a directed graph where each vertex has indegree 1 and outdegree 1. In the example above, the resulting digraph is a Hamiltonian cycle.

$$t_1 \to t_3 \to t_5 \to t_7 \to t_4 \to t_2 \to t_8 \to t_6 \to t_1$$

Using our earlier notation, we represent this cycle as $(13574286)$, $(12345678)$, $(14625837)$, and $(15734628)$. Using our earlier notation, we represent this cycle as $(13574286)$.

We might conjecture that the problem of finding a Hamiltonian decomposition to $K^*_n$ is isomorphic to the problem of determining a solution to the k-team BWTP. Unfortunately, this is not the case, as many BWTP solutions do not yield Hamiltonian decompositions.

Nevertheless, we can use our recursive construction to show that if the $k - 1$ weekends of $S_k$ generate a Hamiltonian decomposition, then so do the $2k - 3$ weekends of $S^*_{2k-2}$. In other words, our recursive construction produces, as a corollary, a Hamiltonian decomposition of $K^*_n$.

To establish this result, for each $1 \leq i, j \leq k - 1$, define an intra-league game to be $t_i \to t_j$ or $u_i \to u_j$, and an inter-league game to be $t_i \to u_j$ or $u_j \to t_i$. From our construction (see Table 9), we see there are $k - 2$ weekends consisting of exclusively inter-league games, while the other $k - 1$ weekends have all intra-league games with the exception of two inter-league games (e.g. $t_2 \to u_2$ and $u_6 \to t_6$ in slots #7 and #8).

In Table 9, the $2k - 2$ = 14 edges of games of slots #1 and #2 trace out a directed cycle with $2k - 2 = 14$ edges.

$$u_3 \leftarrow u_4 \uparrow u_5 \leftarrow u_6 \uparrow u_7 \leftarrow u_1 \uparrow u_2 \leftarrow u_3 \uparrow u_4 \leftarrow u_5 \uparrow u_7 \leftarrow u_4 \uparrow u_2 \leftarrow u_3 \uparrow u_4 \leftarrow u_5 \uparrow u_7 \leftarrow u_4 \uparrow u_2$$

Because of how these inter-league games are scheduled (i.e., $t_i$ plays $u_{s+i}$ in slot $s$, with all arithmetic calculated mod $k - 1$), there is no way for these $2k - 2$ edges to form anything other than a Hamiltonian cycle; specifically, it is impossible for these $2k - 2$ edges to split into two directed $k - 1$ cycles, or any other non-Hamiltonian graph with every vertex having indegree 1 and outdegree 1. If the cycle starts at $t_1$, it must return to $t_1$ after exactly $2k - 2$ steps.

In Table 9, the $2k - 2 = 14$ games of slots #7 and #8 trace out a directed cycle with $2k - 2 = 14$ edges.

$$t_6 \to t_1 \to t_3 \to t_5 \to t_7 \to t_4 \to t_2 \uparrow$$
$$\downarrow$$
$$u_6 \leftarrow u_1 \leftarrow u_3 \leftarrow u_5 \leftarrow u_7 \leftarrow u_4 \leftarrow u_2$$

By our induction hypothesis, the weekend games of $S_k$ decompose into Hamiltonian cycles. For example, the first weekend of our schedule $S_k$ is represented by the 8-cycle $(13574286)$, which, when we remove $t_8$, turns into the 7-path $[6, 1, 3, 5, 7, 4, 2]$, precisely the sequence described above in both the $t$ and $u$ sequences.

In general, if $[p_1, p_2, \ldots, p_{k-1}]$ is the $(k - 1)$-path that arises by removing vertex $t_k$ from a Hamiltonian cycle of $S_k$, then the corresponding weekend of $S_{2k-2}$ becomes $t_{p_1} \to t_{p_2} \to \ldots \to t_{p_{k-1}} \to u_{p_{k-1}} \to \ldots \to u_{p_2} \to u_{p_1} \to t_{p_1}$, i.e., a Hamiltonian cycle with 2$k - 2$ vertices.

In our recursive construction of $S_{2k-2}$ from $S_k$, we ensured that there exist two distinct indices $p_1$ and $p_{k-1}$ for which $t_{p_{k-1}} \to u_{p_{k-1}}$ and $u_{p_1} \to t_{p_1}$, which shows that the weekends of $S_{2k-2}$ consisting of all but two intra-league games forms a Hamiltonian cycle with $2k - 2$ vertices.

Thus, we have found an explicit solution to the problem of decomposing the complete directed graph $K_{2k-2}$ into $2k - 3$ directed Hamiltonian cycles. For example, the columns of Table 9 yield a Hamiltonian decomposition of $K_{14}$.

Unfortunately, a similar argument does not work in our recursive construction of $S_{2k}$, as this schedule splits into $k$ weekends of only inter-league games and $k - 1$ weekends of only intra-league games. Though the former decompose into Hamiltonian cycles of length $2k$, the latter do not; it is easy to see that each of the $k - 1$ intra-league weekends (e.g. slots #9 and #10 in Table 7) decompose into two $k$-cycles, via our induction hypothesis that the $k - 1$ weekends of $S_k$ decompose into Hamiltonian cycles of length $k$.

To fix this, we employ the “cocktail graph” construction (Tillson 1980) where we turn two $k$-cycles into one $2k$-cycle by identifying edge $u_1 \to v_1$ from cycle $C_1$ and edge $u_2 \to v_2$ from cycle $C_2$, and replacing them with the edges $u_1 \to v_2$ and $v_2 \to v_1$. The Tillson construction demonstrates how the edges can be partitioned so that the $2k - 1$ subgraphs form a Hamiltonian decomposition of $S_{2k}$.

Therefore, we have shown how to generate a Hamiltonian decomposition of $K^*_n$ for all even values of $n \geq 8$, although our recursive construction requires edge-switching when $n \equiv 0 (mod 4)$.

It appears that our two constructions for $S_{2k}$ and $S_{2k-2}$ have different “topologies”, as one requires edge-switching to generate a Hamiltonian decomposition while the other does not. This inspires the following open problem.

**Problem #2:** Determine a recursive construction of $S_{2k}$ from $S_k$ for which each of the $2k - 1$ weekends directly forms a Hamiltonian cycle, without any switching of edges.

**The $p$-norm**

Let $S_n$ be a solution to the BWTP. Recall that $h(i, j)$ is the slot in which $t_i$ plays a home game against $t_j$, and $r(i, j)$ is the slot in which $t_i$ plays a road game against $t_j$.

In Table 4, the schedule for $n = 8$ has seven pairs ($i, j$) with $1 \leq i < j \leq n$ for which $|h(i, j) - r(i, j)| = n - 1$, the ideal distance between these pairs of games. As mentioned earlier, the most balanced tournament is mirrored, where $|h(i, j) - r(i, j)| = n - 1$ for all pairs ($i, j$).
Let us define the $p$-norm of an $n$-team schedule $S_n$ as

$$||S_n||_p := \left( \sum_{1 \leq i < j \leq n} ||h(i, j) - r(i, j)|| - (n - 1)^p \right)^{1/p}.$$  

This function, inspired by the $p$-norm in $L^p$ space, equals 0 if and only if $S_n$ is mirrored. We can check by inspection that in the schedule given in Table 4,

$$||S_8||_p = (7 \cdot 0^p + 12 \cdot 1^p + 7 \cdot 2^p + 2 \cdot 3^p)^{1/p}.$$  

This implies that $||S_8||_1 = 32$, $||S_8||_2 = \sqrt{58}$, and $||S_8||_p \to 3$ as $p \to \infty$.

Suppose there exists a schedule $S_n$ that satisfies all the conditions of the BWTP, for which $||S_n||_p = 0$. Let $t_i$ be a team that plays at home in slot #1. Then by the condition that each team plays exactly one home each weekend, as well as the mirroring requirement, $t_i$ plays on the road in slot #2, at home in slot #4, on the road in slot #6, at home in slot #3, on the road in slot #4, at home in slot #6, etc.

Continuing this process, we see that each team that plays at home in slot #1 must play at home in all odd slots and on the road in all even slots. Therefore, if teams $t_i$ and $t_j$ start at home, then they can never play against each other, which is a contradiction. Hence, no schedule can satisfy $||S_n||_p = 0$, i.e., no solution to the BWTP is perfectly balanced. A natural question is to find the $S_n$ for which $||S_n||_p$ is minimized.

Based on our recursive construction, we determine a formula for $||S_{2k}||_p$ and $||S_{2k-2}||_p$ as a function of $||S_k||_p$, and show that our schedule has a small $p$-norm.

In $S_{2k}$, define $h'(i, j)$ and $r'(i, j)$ for each $1 \leq i < j \leq 2k$. From our construction, we know that if $1 \leq i < k$ and $k + 1 \leq j \leq 2k$, then $|h'(i, j) - r'(i, j)| = 2k$. Also, if $1 \leq i < j \leq k$ or $k + 1 \leq i < j \leq 2k$, then $|h'(i, j) - r'(i, j)| = |h(i, j) - r(i, j)| + k$, except for $k$ pairs $(i, j)$ for which $|h'(i, j) - r'(i, j)| = |h(i, j) - r(i, j)| \geq 2$. We have:

$$\sum_{i < j \leq 2k} |h'(i, j) - r'(i, j)| - (2k - 1)^p$$

$$= k^2 \cdot 1^p + 2 \sum_{i < j \leq k} |h(i, j) - r(i, j)| - (2k - 1)^p$$

$$= k^2 + 2 \sum_{i < j \leq k} |h(i, j) - r(i, j)| - (k - 1)^p + k \cdot (\{2k - 1\})^p$$

Therefore, we have shown that $(||S_{2k}||_p)^p < k^2 + 2 (||S_k||_p)^p + k \cdot (2k - 3)^p$. Similarly, we can prove that $(||S_{2k-2}||_p)^p < (k - 1)(k - 2) + 2 (||S_k||_p)^p + (k - 1) \cdot (2k - 5)^p$.

To derive the latter inequality, notice that in $S_{2k-2}$, we have $|h'(i, j) - r'(i, j)| = 2k - 2$ for all $1 \leq i \leq k - 1$ and $k \leq j \leq 2k - 2$, provided $j - i \neq k - 1$. And for all other pairs $(i, j)$, $|h'(i, j) - r'(i, j)| = |h(i, j) - r(i, j)| + (k - 2)$, except for the $(k - 1)$ pairs for which $|h'(i, j) - r'(i, j)| = |h(i, j) - r(i, j)| \geq 2$.

From these two inequalities, we use induction to prove that for all $n \geq 8$, $||S_n||_1 < 4/3n^2$ and $||S_n||_p < n^{1+p}$ for $p \geq 2$. The base cases $n = 8, 10, 12$ follow by inspection, using the schedules given in Tables 4, 10, and 11.

Therefore, our recursive construction produces an infinite set of schedules $S_n$ for which $||S_n||_1$ is bounded by a quadratic function. Given that there are $\binom{n}{2} \sim n^2/2$ pairs of teams, we see each $S_n$ has the property that, on average, each pair of teams plays their two games less than $1/2$ slots apart from the “optimally-balanced” solution that is impossible to attain. But can we do better than a quadratic function? This motivates our final open problem.

**Problem #3:** Determine whether there exists a function $f(n) = c \cdot n^r$, with $r < 2$, for which some valid schedule $S_n$ satisfies $||S_n||_1 < f(n)$ for all $n \geq 8$.

**Acknowledgements**

The author thanks the Pacific Western Athletic Association (PACWEST) for the opportunity to consult on the project that inspired this paper, and thanks the AAAI-18 reviewers for their insightful comments that significantly improved this final product.

**References**


