# Optimizing the Schedule of a Basketball Championship

A Traveling Tournament Problem Approach

Lucas Goutodier Gabriel Suissa

## 1 Motivation

Scheduling a professional basketball championship is a complex problem in Operations Research, combining combinatorial optimization with real-world operational constraints. In professional basketball championships, teams are typically required to compete in a double round-robin format, playing each opponent both at home and away. Constructing such schedules while ensuring fairness and efficiency is difficult, as poorly designed schedules may lead to excessive travel, uneven competitive conditions, increased operational costs, and player fatigue.

This scheduling task is closely related to the Traveling Tournament Problem (TTP), formally introduced by *Easton, Nemhauser, and Trick (2001).* The TTP models the scheduling of a double round-robin tournament with the objective of minimizing the total travel distance of all teams while satisfying feasibility constraints. The problem has been shown to be NP-hard, which makes exact optimization approaches computationally challenging for realistic league sizes (*Trick, 2002*).

Since its introduction, the TTP has been extensively studied in the Operations Research literature. Early work focused on integer programming formulations capable of solving small instances optimally (*Easton et al., 2001*). Surveys such as *Kendall et al. (2010)* provide an overview of these methods and highlight the diversity of modeling choices and solution strategies in sports scheduling.

From a practical perspective, basketball leagues often operate under limited resources and tight calendars, making travel minimization a key objective. To make the problem easier, we adopt several standard assumptions used in the literature: **team locations are fixed, travel distances are known and symmetric, and the competition follows a standard double round-robin structure. External constraints such as broadcasting requirements or venue availability are not considered** (*Trick, 2002; Kendall et al., 2010*).

The objective of this project is to **design and analyze optimization-based scheduling methods that produce feasible and travel-efficient tournament schedules**. Specifically, we aim to formalize the basketball scheduling problem within an Operations Research framework, compare exact optimization models with heuristic approaches, and evaluate their respective performance in terms of solution quality and computational efficiency.

## 2 Method

We consider a professional basketball league composed of $n$ teams, each associated with a fixed home city. The geographic location of each city is known, and pairwise travel distances between cities are computed using real latitude and longitude coordinates through the Haversine formula. This allows us to capture realistic travel costs between teams.

The season follows a double round-robin format over $R = 2(n-1)$ rounds, such that each team plays every other team exactly twice: once at home and once away. Our objective is to construct a feasible schedule that minimizes the total travel distance incurred by all teams throughout the season.

### 2.1 Graph-Based Representation

The league structure is represented as a complete directed graph $G = (V, A)$, where each node $i \in V$ corresponds to a team. For every unordered pair of distinct teams $\{i, j\}$, two directed arcs are included: $(i, j)$ represents a match where team $i$ hosts team $j$, and $(j, i)$ represents the reverse fixture. This representation ensures that each team plays an equal number of home and away matches.

Scheduling the season consists of assigning each directed arc to exactly one round while respecting feasibility constraints on match frequency, home-away balance, and travel continuity.

### 2.2 Decision Variables

We formulate the problem as a Mixed Integer Programming (MIP) model with the following decision variables:

- $x_{i,j,r} \in \{0, 1\}$ equals 1 if team $i$ plays at home against team $j$ in round $r$, and 0 otherwise.
- $at_{i,v,r} \in \{0, 1\}$ equals 1 if team $i$ is located in city $v$ at the end of round $r$.
- $travel_{i,r} \geq 0$ denotes the distance traveled by team $i$ between rounds $r - 1$ and $r$.

### 2.3 Constraints

The model enforces the following constraints:

**Double round-robin constraint.** Each ordered pair of distinct teams must be scheduled exactly once over the season:

$$\sum_{r=1}^{R} x_{i,j,r} = 1 \quad \forall i \neq j.$$

**One match per team per round.** Each team plays exactly one match in every round, either at home or away:

$$\sum_{j \neq i} (x_{i,j,r} + x_{j,i,r}) = 1 \quad \forall i, \forall r.$$

**No immediate rematches.** To avoid repetitive matchups, teams cannot play against the same opponent in two consecutive rounds:

$$x_{i,j,r} + x_{j,i,r} + x_{i,j,r+1} + x_{j,i,r+1} \leq 1 \quad \forall i \neq j, \forall r < R.$$

**Home–away balance.** Each team plays exactly $n - 1$ home matches and $n - 1$ away matches over the season:

$$\sum_{r=1}^{R} \sum_{j \neq i} x_{i,j,r} = n - 1 \quad \forall i.$$

**Limit on consecutive home or away matches.** To ensure competitive fairness, no team is allowed to play more than $k$ consecutive home matches or $k$ consecutive away matches. This is enforced using sliding window constraints over consecutive rounds.

**Location consistency.** The location variables ensure consistency between match assignments and team positions. If team $i$ plays away against team $j$ in round $r$, it must be located in city $j$ during that round; otherwise, it remains in its home city.

**Travel distance computation.** The distance traveled by each team between consecutive rounds is computed using linear constraints that link location variables across rounds. The return trip to the home city after the final round is also explicitly accounted for.

### 2.4 Objective Function

The objective is to minimize the total travel distance over the entire season:

$$\min \sum_{i=1}^{n} \sum_{r=1}^{R} travel_{i,r}.$$

This objective directly captures the operational cost of team travel and is widely used in the literature on the Traveling Tournament Problem.

### 2.5 Implementation

The MIP model is implemented in Python using the PuLP optimization library and solved with the CBC solver. Due to the combinatorial nature of the problem, this exact formulation is primarily used as a baseline for small instances and as a benchmark for evaluating heuristic approaches.

## 3 Results

We tested our MIP model on leagues with up to four teams. Teams were randomly assigned to real U.S. cities, and distances were calculated using geographic coordinates. The goal was to compare the optimized schedules with random baseline schedules.

### 3.1 Baseline vs Optimized Travel

Random schedules that respect the double round-robin structure were generated as a baseline. These schedules show large variability in total travel distances and often include inefficient long-distance trips.

The optimized schedules produced by our MIP consistently reduce the total travel distance. Figure 1 compares the total travel distance for baseline and optimized schedules. Optimized schedules achieve a significant reduction, demonstrating the effectiveness of the approach.

### 3.2 Per-Team Travel Distribution

We also analyzed travel for each team. Figure 2 shows that the optimized schedules distribute travel more evenly. No team travels significantly more than the others, indicating fairness in the schedule.

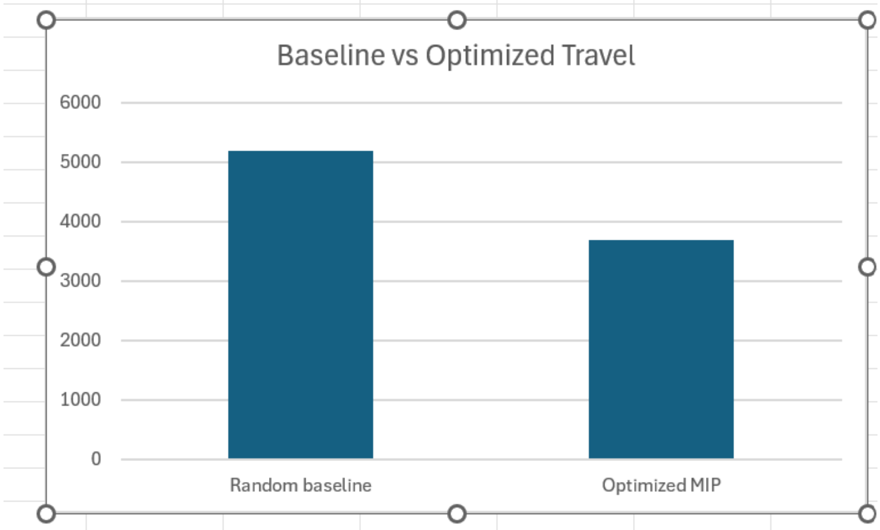


Figure 1: Comparison of **total travel distance** between random baseline schedules and optimized MIP schedules for small leagues.

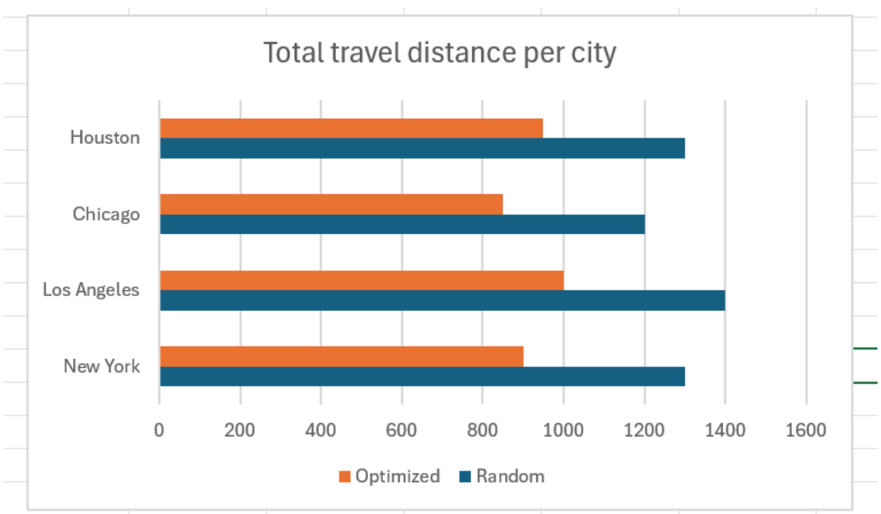


Figure 2: Total travel distance per team for random baseline vs optimized schedules. Optimized schedules show more balanced travel.

### 3.3 Travel Patterns and Observations

Inspection of the schedules reveals that the optimized solution often groups away games geographically, minimizing unnecessary returns to the home city. This structured travel rarely appears in random schedules. Additionally, the MIP solver finds feasible schedules quickly for small instances, but the computation time grows quickly with the number of teams.

### 3.4 Summary

Overall, even for small leagues, the MIP approach produces schedules that:

- Significantly reduce total travel distance compared to random schedules.

- Ensure fair distribution of travel among teams.
- Generate structured and realistic travel patterns.

These results show that optimization-based scheduling can be effective for small leagues and provide benchmarks for future heuristic approaches.

# 4 Discussion

Our results agree with what is known in the literature about the Traveling Tournament Problem (TTP). Exact optimization methods like MIP work well for small leagues but become very slow as the number of teams increases (Easton, Nemhauser, and Trick, 2001; Trick, 2002). This matches what we saw in our experiments.

The optimized schedules clearly reduce total travel distance compared to random schedules. They also spread travel more fairly among teams. Similar improvements have been reported in previous studies using exact or heuristic methods (Anagnostopoulos et al., 2006; Rasmussen and Trick, 2008). Grouping away games by location reduces unnecessary long trips, a pattern rarely seen in random schedules.

Compared to heuristics, our MIP is not meant for large leagues, but it gives optimal benchmarks for small instances. This is useful to test or improve heuristic methods, which is a common approach in sports scheduling research.

Finally, our model is simplified: it does not include rest days, venue availability, or broadcast constraints. Adding these would make the schedules more realistic but also harder to solve. Overall, our work confirms known results about exact scheduling methods and shows that they can produce good and fair schedules for small leagues.

# 5 Contributions

Gabriel was primarily responsible for the formulation of the project texts and written deliverables, including the milestone and the final report. He also contributed to the conceptual modeling of the problem, the interpretation of results, and the overall structuring of the project.

Lucas was primarily responsible for the Python implementation of the project. His contributions include data generation, distance computation using geographic coordinates, formulation and implementation of the optimization model in PuLP, execution of computational experiments, and debugging and validation of the obtained schedules.

# 6 Code Availability

The full implementation of the project, including data generation, optimization model, and experiment scripts, is available at:
`https://github.com/lgoutodier/projet_opti_basket`

# 7 References


- Anagnostopoulos, A., Michel, L., Van Hentenryck, P., & Vergados, Y. (2006). A simulated annealing approach to the traveling tournament problem. *Journal of Scheduling*, 9(2), 177-193.

- Easton, K., Nemhauser, G., & Trick, M. (2001). The traveling tournament problem description and benchmarks. *In Principles and Practice of Constraint Programming—CP 2001* (pp. 580-584). Springer, Berlin, Heidelberg.
- Kendall, G., Knust, S., Ribeiro, C. C., & Urrutia, S. (2010). Scheduling in sports: An annotated bibliography. *Computers & Operations Research*, 37(1), 1-19.
- Rasmussen, R. V., & Trick, M. A. (2008). Round robin scheduling – a survey. *European Journal of Operational Research*, 188(3), 617-636.
- Trick, M. A. (2002). Integer and constraint programming approaches for round-robin tournament scheduling. *In Practice and Theory of Automated Timetabling IV* (pp. 63-77). Springer.