Baseball Game and Umpire Crew Scheduling Optimization

Christopher Arena, Coco Cai, Logan Donaldson, James Ingram, Eli Katz*, Benjamin Keever, Justin Nam, Sam Oberly, Devon Osgood, Chloe Warren*, Emily Zhai

The Johns Hopkins University | Whiting School of Engineering | Baltimore, MD
Design Day 2021

Introduction

The Johns Hopkins University Baseball Scheduling Optimization Research Group was launched in 2011 by Anton Dahbura and Donniell Fishkind. The group uses combinatorial optimization and combinatorial design, as well as state-of-the-art software and computing resources, to create schedules for professional baseball. We have created season schedules and umpire crew schedules for the majority of the leagues in Minor League Baseball (MiLB) at all levels. Before we came on the scene, leagues in MiLB utilized very suboptimal, by-hand schedules which took weeks to create. Our group has become well-known throughout professional baseball for pioneering mathematical optimization in scheduling for leagues in MiLB.

Objectives

Create optimized schedules for MiLB using mathematical modeling, combinatorial optimization and design, advanced software, and supercomputing resources.

Materials and Methods

1) We meet with league scheduling committees and leadership to determine the league’s scheduling rules and priorities. Part of this task is to help the leagues themselves quantify a good schedule and resolve the competing interests and goals into a single objective function.

2) An appropriate skeleton ("template") is created, which includes gross features such as season midway point, off-day spread, holiday patterns, and division of the days into series units, etc.

3) We model the league constraints using appropriate variables and constraints. Quadratic constraints are converted into linear ones. The goal is the formation of a linear integer program whose objective function reflects the "badness" of a feasible schedule. Matrices and vectors with the specific problem parameters are created by building and running a computer program.

4) The matrices and vectors with the specific parameters of the binary integer linear program representing the schedule optimization are fed into a state-of-the-art solver, running on a 160-core computer until an optimal solution is produced.

5) The output is distilled into VBA spreadsheets that clearly summarize the descriptive statistics of the created schedule, and highlight important features.

6) We meet with the league scheduling committee and leadership to tune the model, and we adjust the priorities if this is useful.

Our computers:

Ziggy is an SGI UV2000 System with 8x Intel E5-4650V2 processors (10 cores each @ 2.4GHz, 256MB cache) and 256GB DDR3 1866MT/s Memory.

Chilly Willy is a custom Penguin Computing system with 4x AMD Opteron 6378 processors (16 cores each @ 3.9GHz, 16MB cache) and 129GB DDR3 1600MT/s Memory.

Results

Umpire Crew Scheduling –

Umpires are responsible for enforcing the rules of the game on the field and adjudicating all aspects of the game. There are half as many umpire crews as teams in a league. Umpire crew scheduling is built on the season schedule for the league, and the requirements and priorities of the crew schedule vary somewhat from league to league. Standard requirements include some of the following:

• Crew travel should be minimized;

• Differences between total travel of different crews should not be extreme;

• Crews cannot travel more than 500 miles without an off day;

• Crew travel patterns should be as sensible as possible to minimize "ping-ponging" and other negative patterns specified by the league;

The Johns Hopkins University Baseball Scheduling Optimization Research Group was launched in 2011 by Anton Dahbura and Donniell Fishkind. The group uses combinatorial optimization and combinatorial design, as well as state-of-the-art software and computing resources, to create schedules for professional baseball. We have created season schedules and umpire crew schedules for the majority of the leagues in Minor League Baseball (MiLB) at all levels. Before we came on the scene, leagues in MiLB utilized very suboptimal, by-hand schedules which took weeks to create. Our group has become well-known throughout professional baseball for pioneering mathematical optimization in scheduling for leagues in MiLB.

Figure 1 – Optimization Research Group

Figure 2 – Diamond Dollars for Season Scheduling

In the Diamond Dollars concept, each team is given a fixed number of tokens, or chips, to assign to days of the season that they would prefer to be home or away. The number of chips a team places on a day reflects the importance of that request. A team could in theory place all of its chips on a given day of the season, or they could spread the chips around the days of the season. Diamond Dollars supplement the other league and team constraints and are not intended to supersede or interfere with them. The group came up with a way of adjusting the priority of Diamond Dollars relative to the other constraints for the season. Surprisingly to many, the optimization was able to satisfy over 75% of the team Diamond Dollar preferences without disrupting the overall quality of the league schedule. The system was especially effective at identifying team requests that were complementary to the rest of the constraints. The Carolina League adopted the Stl Buck/Diamond Dollars system for use in one of their recent seasons.

Figure 3a,b– Constraint modeling, very simplified examples

The decision variables are constructed as follows:

\[ X_{i,j,k} = \begin{cases} 1 & \text{if team } j \text{ is playing home at stadium } k \text{ during series slot } i \\ 0 & \text{otherwise} \end{cases} \]

Constraints are designed to enforce the league requirements and priorities. Artificial variables are created as needed to allow penalized violations. This is important, as there is no perfect schedule, and there are always competing goals that need to be optimized.

3a) Every team must have a specified number of home games. For each team, the number of games played at home is n, with an allowance of deviation by one game at a penalty, enforced through an artificial variable.

3b) Every game is home versus visitor. We iterate through teams within stadiums within slots. We add to the iterative count of req because we are adding an equality constraint. Within our iteration, users are seeing that if one team is at all its stadium, then there should be another team there as well and that if it is not, there should be no teams there.

Conclusion

Our methods have proven to be effective and highly adaptable. Future directions include advancing the state of the art in season scheduling and umpire crew scheduling for professional baseball leagues.