

WHO'S YOUR TIGER? USING SIMULATION TO OPTIMIZE THE LINEUP OF THE DETROIT TIGERS OFFENSE

Jared Davis
Barbara Fordyce
James Cicala
Matthew Cooper
Omer Tsimhoni

Department of Industrial and Operations Engineering
1205 Beal Avenue, University of Michigan
Ann Arbor, MI 48109-2117, USA

ABSTRACT

As part of an undergraduate engineering class project, a simulation of the Detroit Tigers offense was created to explore potential changes that would increase number of wins. More specifically, we seek to determine a lineup for an MLB team, the Detroit Tigers, that would maximize their potential runs. To answer our ultimate question of whether a manager actually maximizes runs scores, we compare our results to the Tigers' 2007 performance. We determine that though the Tigers did not use the ideal lineup as determined by our model, the lineup they did utilize was moderately robust, with ours winning 89.6 games versus the actual wins of 88 games. Additionally, we apply our model to a normative analysis of the Tigers ideal lineup for the 2008 season. This ideal lineup only changes the middle of the batting order (acceptable by management) and is predicted to win 99 games in the 2008 season.

1 INTRODUCTION

Modern baseball strategy has developed into subjective analysis of statistics based on deep-seeded traditions. These heuristics have served teams well in the past, but it is time that these traditions be questioned. The aspect of baseball most heavily influenced by tradition is batting lineup. Some traditions associated with batting lineup are (Baseball-reference, 2007):

- The leadoff hitter should be good at getting on base and should not be a power hitter.
- The third hitter should be the best all around batter, with a good average and power.
- The fourth hitter should be the best power hitter.
- Overall, the best hitters are clustered in the beginning of the lineup and the worst at the end.

The goal of a baseball manager during the regular season is obviously to maximize the number of games won. Pitching and defense are no doubt important factors in baseball, but scoring runs wins games. Throughout the history of baseball, batting has played more and more of a crucial role in determining success; ranging from the lowering of the pitchers mound, recruiting of power hitters over top defenders, and an increased number of runs scored per game (baseball-reference.com). These factors make pitching skill less and less relevant to a team's performance. Increasing the runs scored in a season by even a small amount may have significant effects on the number of games that a team wins. For instance, in 2007 the Philadelphia Phillies and the Texas Rangers allowed approximately the same number of runs on defense. However, the Phillies scored 76 more runs (892 to 816), a difference of less than .5 runs scored per game. The difference contributed to the Phillies winning 14 more games than the Rangers this year. The Phillies came in first place in their division, while the Rangers came in last (Baseball-Almanac 2007).

It is imperative that a manager maximizes total runs scored because of several financial and image implications. Teams earn revenues through ticket sales, concessions, merchandising, etc. Teams with higher win totals are generally able to generate greater revenues and make the playoffs, which increases revenues further. The financial implications of more wins may be on the order of tens of millions of dollars. In addition to monetary gains, more wins can have a significant influence on the status of the baseball team. A team that has a reputation for winning regularly is more attractive to potential free agents. For example, there have been instances where players have turned down large offers from losing teams to sign smaller contracts with a winning team.

Baseball statistics have been an integral part of the culture of baseball for many years. Traditionally, data such as runs batted in, runs scored, and player athleticism were considered in decision making. These, however, are not good predictors of a team's success. More recently, a new branch of analysis tools, termed Sabermetrics, has emphasized using only appropriate quantitative data to optimize the performance of baseball teams (Grabiner 2008). The strongest proponent of Sabermetrics, Bill James, has even devised a system to measure a player's contribution to his team's wins based on various personal statistics (James 2002).

Much has been done in the way of optimizing batting lineups through analytical optimization. These approaches use Markovian processes and rely on grouping batters into characteristic groups to optimize the lineup, which has been done successfully to near-optimal levels (Sokol 2003, Ovens 2006)

As part of an undergraduate engineering class project (see Tsimhoni and Wu 2005), we decided to apply simulation and optimization techniques to the analysis of the game of Baseball. We consider simulation to be an obvious expansion to the existing Sabermetrics and analytical approaches. Using simulation to better understand the successes of a baseball team will add to baseball's evaluation toolkit. Simulation is also a good technique for baseball evaluation because it can take into account all of a player's statistics without having to subjectively classify him as a certain type of batter, which is necessary in analytical approaches.

The simulation we describe in this paper predicts the batting lineup which maximizes runs scored per season and correlates this number to games won. The simulation serves two purposes. First, it demonstrates the potential of a standard simulation and optimization methodology to address real world, sports-related, problems. Second, it answers the specific question: what is the optimal batting lineup for the Detroit Tigers in 2007. The results of the latter question could provide managers a more specific way of selecting a batting lineup to optimize their performance. Furthermore, as in other simulation projects, the simulation project lends itself well to making predictions. For example, in early December 2007, the Tigers acquired three new players that are expected to be highly utilized in the 2008 season. Using our validated simulation model, we predict the optimal lineup for the 2008 season that will incorporate these new players.

2 ASSUMPTIONS

2.1 High Level Assumptions

Three concessions were made in the model to adjust for the unpredictability of baseball. We assume that there are no

interaction effects between different players in the lineup or the position in which that player bats; the odds for success of each player-dependent batting event are equal no matter where that player is in the lineup. In the reality, this may not be entirely accurate as a player hitting directly in front of a power hitter is less likely to be walked, and a player batting towards the end of the order may face more predictable pitches. Nevertheless, these variances in pitching behavior are situational and should not alter our results. In the 2007 season, Detroit Tigers batters accounted for 4113 total outs, equivalent to 1371 innings. We use this inning total for all validation and prediction experiments. In order to determine the optimal lineup, we assume that the 9 players included in the lineup are able to play every inning for the entire length of the season. All other basic baseball rules apply.

2.2 Detailed Assumptions

We make several other assumptions in the model conceptualization and data collection stages:

- A batter reaching base on error is treated like a single for both the batter and base runner.
- No stolen bases or base runners caught stealing.
- No double plays at bases other than first.
- All base runners use equal probabilities for advancement regardless of player speed or aggressiveness.
- Fielder's choice outs are modeled as non-strikeout outs.
- No wild pitches or passed balls by opposing pitchers.
- No base runner advancement on strikeouts.
- No additional base runner advancement, other than forced, on base on balls.
- Player attitude and effort are not taken into account.

2.3 Stolen Bases Assumption

We chose to exclude stolen bases in the model to decrease the model's complexity without distorting results. A successfully stolen base leads to a higher run total. For example, a player at second is more likely to score than a player at first. However, a player thrown out stealing a base both increases the out total and removes a player from the base paths. To break even with respect to expected runs scores, a team must successfully steal bases at a 73% success rate, determined by equation 1.

$$\text{Success Rate (\%)} = \frac{\text{Steals}}{\text{Steals} + \text{Caught Stealing}} * 100\% \quad (1)$$

The Detroit Tigers starters in 2007 accounted for 87 steals and were caught 28 times for a success rate of 75.7%

(equation 1). This total is very close to the break-even success rate. A general rule states the marginal runs from steals in equation 2.

$$\text{Runs} = 0.24 * \text{Steals} - 0.03 * \text{Caught Stealing} \quad (2)$$

The Tigers, then would have added 2.68 additional runs from stealing bases (equation 2). This amount is quite small relative to runs scored in a season, and should be insignificant. Additionally, this number remains approximately constant among different lineup combinations and would not affect comparisons.

3 METHOD

3.1 Data Acquisition

For each player, 2007 season totals of plate appearances, singles, doubles, triples, homeruns, strikeouts, non-strikeout outs, bases on balls, hit by pitches, sacrifices, number of times reaching on error, and number of times grounding into double plays were collected from Baseball-reference.com.

Our model also calls for data on how a base runner advances within a given scenario. Base runner advancement statistics are not historically compiled; however play by play accounts of every game are available. We collected the play by play records from 12 high scoring games (high scoring games usually have more runner advancement activity) of the Detroit Tigers 2007 season. (Baseball-reference.com 2007)

3.2 Data Analysis

We translated the hitting statistics into less conventional ratios of specific batting events to plate appearances.

Base runner advancement depends on the activity of the current player at bat. For example, if a base runner is at second base and the batter hits a double, the base runner will advance to home base and score a run. On the other hand, if the batter is walked, the base runner will not advance but will remain on second base. In some of these scenarios it is obvious how the base runners will advance because there may be only one option. In contrast, there are many scenarios in baseball where base runner advancement is more variable. For instance, if a base runner is on second base and the batter hits a single, the base runner could advance to third or home depending on the location of the hit and the situation of the game. The frequencies of base runner advancement given these scenarios were then converted to percentages and used to dictate how the base runners advance. All base runners score when the batter either hits a triple or homerun regardless of their original position on the bases.

Figures 1-3 summarize the resulting percentages for batter dependent base-runner advancement.

		Advance (bases)				
		0	1	2	3	Out
Batter Activity	Single	0.000	1.000	0.000	0.000	0.000
	Double	0.000	0.000	0.433	0.548	0.019
	Non-SO Out	0.715	0.241	0.000	0.000	0.044

Figure 1: Base-runner advancement for first base

		Baserunner Advancement (bases)			
		0	1	2	Out
Batter Activity	Single	0.033	0.361	0.590	0.016
	Double	0.000	0.000	1.000	0.000
	Non-SO Out	0.565	0.435	0.000	0.000

Figure 2: Base-runner advancement for second base

		Advance (bases)		
		0	1	Out
Batter Activity	Single	0.000	1.000	0.000
	Double	0.000	1.000	0.000
	Non-SO Out	0.433	0.567	0.000

Figure 3: Base-runner advancement for third base

For example, a base-runner on second base has a 3.3% chance of staying at second base on a single, a 36.1% chance of advancing to third base, a 59.0% of scoring, and a 1.6% chance of getting out.

3.3 Input

For every at bat, we calculate the probability of a hit based on the cumulative probabilities from that player's hitting statistics. Base runner advancement is found in a similar way as above. The system will then react according to its corresponding batter activity.

Our model uses the players from the highest frequency lineup from the 2007 season, as follows:

1. Granderson
2. Polanco
3. Sheffield
4. Ordonez
5. Guillen
6. Rodriguez
7. Casey
8. Monroe
9. Inge

3.4 Tigers Offense as a ProModel Simulation

We use ProModel (version 7) to realize the simulation. Upon initialization, all relevant player statistics are loaded from an excel sheet into arrays that are associated with specific players. At every arrival, a player loads the values of these arrays into pertinent attributes that are used to determine hits. Locations that players are routed through are: home plate, three bases, a batter's box for arrivals, and a dugout for players who score, with path networks around the bases for the players to follow. Figure 4 shows a screen shot of the simulation.

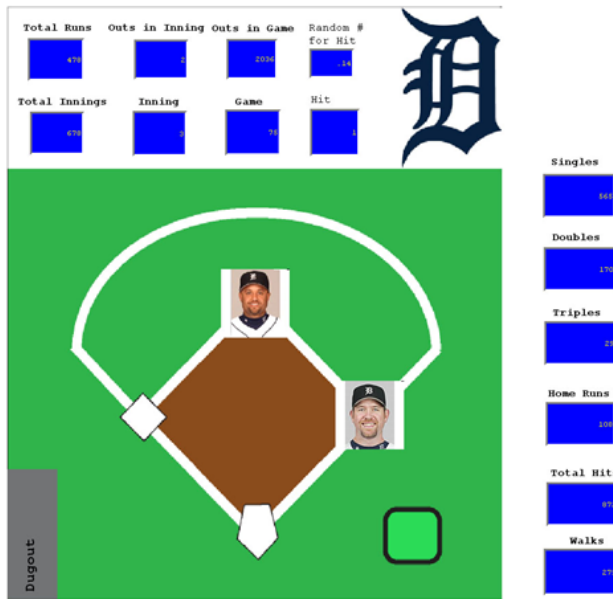


Figure 4: Sample screen from simulation

Our model is not a conventional ProModel simulation, in that it does not use time as its primary operational logic. Instead, we use time as a standard to allow the entity's to arrive in the system in the proper order. Consequently, we set the first arrival time (in minutes) of each player to their respective position in the lineup with a frequency of 10 minutes. The effect of this arrival setup is that each player in the lineup comes up to bat every minute. With entity speeds sufficiently fast, players move around the bases in between hits, and time remains unimportant to the rest of the model.

Players arrive to a batter's box and are then routed to home plate where the majority of the operational logic takes place. Here, using a random number taken from a uniform distribution and player statistics, a hit value is assigned to the batter. Before routing occurs for this batter, all base runners are assigned how many bases to advance or exit. These advances are determined by the hit value, what other bases are occupied, another uniform random number, and the historical statistics on base running behav-

ior found in Figures 1-3. Once all base runners have been designated to a new base position or to exit, the original hitter is routed to his new base. Next, the base runners are routed to new positions based on the previously determined operational logic. All players scheduled to score a run are routed through home and into the dugout where they exit and increment the run counter.

After every new hitting or base runner out, a macro determines the state of the simulation and which parameters need to be reset. If there have been three outs in an inning, all base runners exit and a new inning begins at the point left off in the lineup. Additionally, nine innings signal the end of the game and the batting order resets by having players exit directly from the batter's box, without hitting or incrementing outs, until the lead-off hitter arrives. This resetting of the batting order is important because over the course of the season players at the top of the batting order have significantly more hits than those at the bottom.

Our model finds the predicted total number of runs scored in a season by the Detroit Tigers for a specific lineup, as well as assorted other significant seasonal statistics.

3.5 Validation

We made three validations of our model; data validation, computerized validation, and operational validation (Sargent 2007). Data and computerized validation were fairly straight forward because of the nature of the statistics used and the debugging features of ProModel. The hitting data was validated by confirming player's statistics across multiple baseball-statistics websites (Baseball-reference.com 2007; Baseballalmanac.com 2007; Yahoo Sports 2007). The debugging features of ProModel allowed us to step through the simulation and confirm its programming validity.

To determine the operational validity of the model, we compared the critical simulation output – runs scored – to the runs scored by the Tigers in the 2007 season. The Tigers, in 2007, scored 887 runs. The total of 887 runs is just 6 more than what they scored in the simulation (N = 30) and well within a 95% confidence interval. However, there is a great deal of variation in batting orders over the course of a 162 game season. Consequently, from the 25 games in which the highest frequency lineup played, we extrapolated the runs scored for the entire 2007 season. This extrapolation led to an estimated 868 runs scored, which is also within the confidence interval. To ensure accuracy, we measured several other variables as shown in Figure 5.

Variable	2007	Avg	95% CI Lo	95% CI Hi
Hits	1652	1712.2	1693.1	1731.3
Doubles	352	344.3	336	352.6
Triples	50	47.4	44.6	50.2
Home Runs	177	172.3	167.2	177.4
Walks+HBP	530	549.1	538.7	559.5
Runs	887	881.6	866.1	897.1

Figure 5: 2007 Results accurately modeled

We looked at total hits, doubles, triples, home runs, and walks to determine whether the model outputted at these figures accurately as well. All these totals accurately reflected 2007 actual values except for hits. However, this may be explained by our inclusion of times reached on error in this number, despite the fact that this number is not included in actual baseball bookkeeping. Inclusion of these 46 events brings the 2007 total to 1698 hits, now within the lower bound of the 95% CI.

To further ensure model accuracy, we compiled the 2007 statistics for three other baseball teams: the Toronto Blue Jays, Boston Red Sox, and New York Yankees. These teams scored 753, 867, and 968 runs, respectively. These teams were chosen to reflect a wide range of runs scored. A simulation that accurately reflected their runs scored for the season would serve to increase confidence in the model. Results of the simulation are summarized below in Figure 6.

Team	N	Avg Runs	95% CI Lo	95% CI Hi
Blue Jays	30	725	704.2	745.8
Red Sox	30	838.8	820	857.6
Yankees	30	982	947	1017.4

Figure 6: Other AL team simulation results

These results lend credence to the validity of our model. For the Blue Jays and Red Sox, our model only slightly underestimates their total runs scored. The upper 95% confidence limit is just 7.2 runs and 9.4 runs less, respectively. For the Yankees, the simulation average was just 14 runs away from the actual total, which was well within the bounds of the confidence interval. Though two of the run totals fell outside the confidence interval, we deemed this to be reasonable. First, the many model inputs were specific to the Tigers, meaning that the results for other teams may be slightly skewed. Additionally, the lineups modeled for these teams, though the mode, accounted for less than 10% of the games in the season. This would introduce greater variability.

3.6 Optimization Software

A key deliverable of our program is to determine the batting lineup which results in the most runs scored per season. To assist with executing this deliverable, we used the optimization software, SimRunner. SimRunner evaluates an existing ProModel simulation model, performs a variety of “what-if” scenarios on the model parameters, and uses optimization algorithms to optimize a single output factor.

The inputs required to use SimRunner were an optimization equation, single value RTI macros for manipulation, weighted value of each variable, the number of replications per run, the selection of an optimization setting, and a validated model in ProModel.

3.7 Macros Formulation

SimRunner allowed us to enforce upper bound and lower bound values for each input variable, but it did not allow for the use of the complex constraint equations. Therefore, additional programming in ProModel was required to ensure that each position in the batting lineup only had one player and that each player was assigned to only one position in the lineup. The solution to this problem was the use of macros in ProModel to determine the batting lineup, using the variables manipulated in SimRunner as the input variables for the macros.

During optimization, the lineup order is determined as a function of eight single-value macros, whose available values are shown below in Figure 7.

Macro Name	Possible Values
Mac1	1, 2, 3, 4, 5, 6, 7, 8, 9
Mac2	1, 2, 3, 4, 5, 6, 7, 8
Mac3	1, 2, 3, 4, 5, 6, 7
Mac4	1, 2, 3, 4, 5, 6
Mac5	1, 2, 3, 4, 5
Mac6	1, 2, 3, 4
Mac7	1, 2, 3
Mac8	1, 2

Figure 7: Macro Ranges

During the initialization logic, the macros are called, processed through an algorithm, and used to assign lineup positions to players via global variables representing each player. The algorithm used to assign the lineup positions is as follows. First, an array representing available positions in the batting lineup is initialized with all spots in the batting lineup as available (represented as 0).

The value of Mac1 represents the spot in the lineup Player1 is going to bat, so that spot in the array is marked as unavailable (represented by a 1) and the variable Player1 is set to the value of Mac1. At that point, there are only eight available spots in the lineup for Player2, which is

why Mac2 only has eight possible values as seen in Figure 7. If $Mac2 = n$, then Player2 is assigned the n th available spot in the batting lineup starting from one, and the n th available spot is marked as unavailable. For example, if $Mac1 = 2$ and $Mac2 = 5$, then Player1 would bat second in the lineup and Player2 would bat 6th. This visually demonstrated in Figure 8.

Macro Values: $Mac1 = 2, Mac2 = 5$

Resulting Variables: $Player1 = 2, Player2 = 6$

Line-up	1	2	3	4	5	6	7	8	9
Array value	0	1	0	0	0	1	0	0	0

↑ ↑ ↑ ↑ ↑
 1st available 2st 3st 4st 5st

Figure 8: Example macro to batting lineup assignment

This algorithm of assigning the macro value to the n th available position in the batting lineup is repeated through macro 8. When Player8 is assigned the $Mac8 = n$ th available position in the lineup, there will only be one more spot available and Player9 is assigned that position in the lineup. Once all players were assigned to their spot in the lineup, the first arrival time of each player was off-set by the number of minutes equal to their spot in the lineup. For instance, if Player5 was 8th in the lineup, his first arrival time would be at 8 minutes.

To find the optimal batting line up, we ran the simulation using SimRunner. The objective function was to maximize the total number of runs scored and the input variables were the eight single-value macros.

4 RESULTS

4.1 Optimization Process

In SimRunner, we ran the optimization multiple times with 30 replications per experiment on the “Aggressive” setting to maximize runs scored and ensure that results were accurate. These optimization trials were seeded with different initial macro values to ensure that the optimization software would not take the same path while optimizing. Each trial of the simulation yielded 109 experiments. An example trial is seen in Figure 9. The upward trend in runs scored, evident in the figure, indicates the search process among various combinations of lineups.

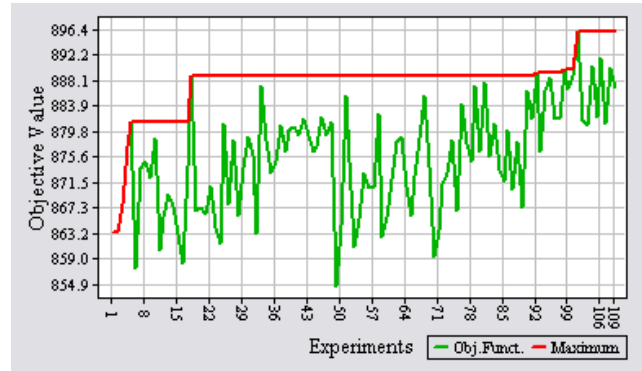


Figure 9: Optimization plot

4.2 Optimal Lineup

Our simulation converged to the same optimal lineup for the multiple replications. The optimal lineup was determined to be:

1. Sheffield
2. Ordonez
3. Guillen
4. Polanco
5. Granderson
6. Rodriguez
7. Monroe
8. Inge
9. Casey

The top 10 best lineups were all very similar to this lineup. Our optimal lineup was much different than traditions suggest. For instance, Gary Sheffield was identified as the leadoff hitter in all trials. In traditional baseball thinking, he would bat in the 3, 4, or 5 spots because of his power and lack of speed. However, our simulation de-emphasizes these batting characteristics and focuses more on Sheffield’s high on-base percentage, which would characterize him more as a lead-off hitter.

The ideal lineup from our model recorded 895.7 runs on average, at a 95% confidence interval of (878.5-912.8). If the Detroit Tigers most frequently used lineup had been able to play together for an entire season, they would have scored only 868 runs as calculated earlier in section 3.5. The optimized line-up provides a statistically significant improvement over this total. It is noted that the actual value of runs scored by the Tigers was 887, which is within the range of the confidence interval.

4.3 Win-Loss Improvement

The principle goal in baseball is not to score runs, but to win games. However, scoring more runs tends to lead to more wins in a season. To accurately estimate wins in a season, the Pythagorean Winning Percentage (PWP) equa-

tion may be applied. The equation takes runs scored and runs allowed as its only inputs to predict the winning percentage of a team (Davenport, 1999).

$$Win\% = \frac{(Runs\ Scored)^{1.83}}{(Runs\ Scored)^{1.83} + (Runs\ Allowed)^{1.83}} * 100\% \quad (3)$$

Multiplying equation 3 by the number of games in a baseball season (162) provides an estimate for the number of wins in a season. In 2007, the Tigers won 88 games. The PWP equation predicts that they would have won $(162 * 887^{1.83} / (887^{1.83} + 797^{1.83})) = 88.9$ games, very near their actual total. The most frequently used Tigers lineup data, which provides the best comparison versus the simulation, would have won $(162 * 868^{1.83} / (868^{1.83} + 797^{1.83})) = 87.3$ wins.

The most relevant comparison, then, may be a measure of expected wins as a result of our simulations. To ensure a proper comparison of just the offenses, we assume runs allowed will stay constant. The ideal lineup determined for 2007 would have produced $(162 * 896^{1.83} / (896^{1.83} + 797^{1.83})) = 89.6$ wins. This win total is very near the 2007 actual and starting lineup extrapolated totals. It is evident that the Detroit Tigers do a pretty good job of maximizing runs produced when developing lineups. Since most teams tend to follow similar tenets and traditions when developing lineups, a corollary to this statement would be that most baseball teams closely maximize runs scored.

Thus far in this report, we have found predicted and actual run totals from very good lineups. Therefore, the resulting variance in runs scored between these good lineups has been small, and could lead some to believe that lineup order is not significant. We demonstrate below that the order in which the players bat *does* make a difference in total runs scored. To demonstrate this point, we optimize the lineup with the objective of now *minimizing* runs scored. This optimization yielded a lineup order of:

1. Inge
2. Polanco
3. Monroe
4. Guillen
5. Sheffield
6. Rodriguez
7. Ordonez
8. Casey
9. Granderson

Upon initial inspection, this is an inferior lineup. This lineup places players with excellent statistics such as Ordonez and Granderson at the end of the lineup where they will get fewer at-bats while placing lower quality players such as Inge and Monroe at the beginning. The results yielded an average 838.5 runs per season with a 95% confidence interval (822.7, 854.3).

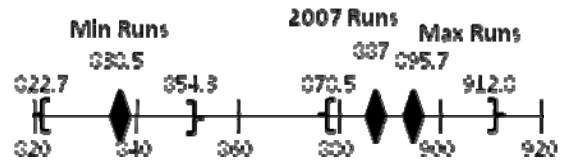


Figure 10: 2007 Improvement v. minimum

The results of the lineup optimized for minimum runs scored yields a statistically significant lower total than the optimization for maximum runs and the 2007 actual runs scored. If the Tigers had used the lineup above beginning with Inge, the team would have won only 84.7 wins (equation 3), a sufficiently lower number of runs to likely miss the postseason. Clearly, this demonstrates that lineup order is a significant factor in the number of runs scored in a season. Also, this shows that the current managers are doing a good job, but can still improve.

5 CONCLUSIONS

5.1 Practical Significance

Our simulation shows that the Detroit Tigers' 2007 lineup that uses traditional lineup methods is not quite optimal, but is marginally close. The optimal lineup from our simulation defies several traditions, such as putting power hitters in the middle of the lineup, but the difference between the optimal and actual lineups is only one to two wins per season. While this is statistically significant, it should not make a tremendous practical significance given the large number of lineups used over the course of a 162 game season. By the time this paper is reviewed, the Detroit Tigers season will be well underway, nevertheless we stand by our model.

5.2 Applications

After an active offseason, in which the Detroit Tigers acquired Jacque Jones, Miguel Cabrera, and Edgar Renteria to replace Craig Monroe, Brandon Inge, and Sean Casey, the coaching staff is now faced with an uncertainty with regards to the batting order for the 2008 season. Detroit Tigers' manager Jim Leyland was quoted as saying, "The early part of the lineup is very much set, and I have a good idea of my seven, eight, nine hitters. I have been intrigued by one of the lineups I have seen from [Tigers' director of operations] Mike Smith with regards to the three through six hitters." Baseball experts have interpreted this to mean Granderson and Polanco will bat in the 1 and 2 positions and Renteria, Jones, and Rodriguez will fill the 7,8,9 spots respectively leaving any combination of Sheffield, Ordonez, Guillen, and Cabrera to fill the middle of the lineup spots.

We extend our analysis to provide a solution to Leyland's uncertainty as to the order of Sheffield, Ordonez, Guillen, and Cabrera in the middle of the lineup. The simulation for the optimal 2008 lineup constrained by Leyland's quote above suggests the following lineup:

1. Granderson
2. Polanco
3. Ordonez
4. Sheffield
5. Guillen
6. Cabrera
7. Renteria
8. Jones
9. Rodriguez

This order of players yielded an estimated 1040 runs per season with a 95% confidence interval of (1021,1059). To be conservative, we used the lower limit of the confidence interval to calculate the expected wins. The number of expected wins using this batting order, if held constant across the entire season, is expected to win $(162 \cdot 1021^{1.83} / (1021^{1.83} + 797^{1.83})) = 99$ wins. In 2007, this team would have gone from missing the playoffs by 6 games to safely holding the best record in baseball, justifying the expenditure on new talent. No team since the advent of the Wild Card in 1995 has missed the playoffs with so many wins. Our simulation provides a valid method for selecting a batting lineup that will maximize runs scored, and with its use, a more successful season will result.

REFERENCES

- Banks, J., J. S. Carson, B. L. Nelson and D. M. Nicol. 2001. *Discrete Event Simulation*. Third Edition. Prentice Hall, Inc. Upper Saddle River, New Jersey.
- Baseball-Almanac. 2007. *Baseball Almanac - The Official Baseball History Site*. Retrieved December 13, 2007 from the World Wide Web: <http://www.baseball-almanac.com/>.
- Baseball-reference.com. 2007. *Detroit Tigers Schedule, Box Scores and Splits*. Retrieved December 14, 2007 from World Wide Web: http://www.baseball-reference.com/teams/DET/2007_sched.shtml.
- Davenport, Clay, Woolner, Kieth. 1999 *Revisiting the Pythagorean Theorem*. Retrieved December 15, 2007 from World Wide Web: <http://www.baseballprospectus.com/article.php?articleid=342>.
- Grabner, D. 2008. *The Sabermetric Manifesto*. Retrieved March 28, 2008 from the World Wide Web: <http://www.baseball11.com/bb-data/grabner/manifesto.html>.
- James, B. 2000. Win Shares. *STATS, Inc.* Northbrook, IL.
- Lindsey, G.R. 1961. Progress of score during a baseball Game. *Journal of The American Statistical Association* 56(295): 703-728.
- Ovens, M. and B. Bukiet. 2006. A mathematical modelling approach to one-day cricket batting orders. *Journal of Sports Science and Medicine*, 5(4): 495-502.
- Pankin, M.D. 1991. Evaluating Offensive Performance in Baseball. *Operations Research* 26: 610-619.
- Tsimhoni, O. and C. Wu C. (2005). Learning simulation through team projects. In Proceedings of the 2005 Winter Simulation Conference, eds. M. E. Kuhl, N. M. Steiger, F. B. Armstrong, and J. A. Joines. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers, pp. 2613-2618.
- Sargent, Robert (2007) Verificatoin and Validation of Simulation Models. *Proceeding of the 2007 Winter Simulation Conference*.
- Sokol, J.S. (2003). A robust heuristic for batting order optimization under uncertainty. *Journal of Heuristics*, 9(4), 353
- Yahoo Sports *Detroit Tigers 2007 Season Stats*. Retrieved December 13, 2007 from World Wide Web: http://sports.yahoo.com/mlb/teams/det/stats/bycategory?cat=Batting&type=0&sort=22&qualified=0&year=season_2007.

AUTHOR BIOGRAPHIES

JARED DAVIS is a senior at the University of Michigan studying Industrial and Operations Engineering and Economics. He will continue in the Engineering Global Leadership Honors Program by completing his Masters in Industrial and Operations Engineering in 2010. jaredm@umich.edu.

BARBARA FORDYCE is a senior at the University of Michigan studying Industrial and Operations Engineering. She will continue in the Engineering Global Leadership Honors Program by completing her Masters in Industrial and Operations Engineering in 2009. bfordyce@umich.edu.

JAMES CICALA graduated in April 2008 from the University of Michigan with a Bachelor's degree in Industrial and Operations Engineering. He will continue to pursue interests in the Detroit Tigers. He will graduate in 2008. jcicala@umich.edu.

MATTHEW COOPER graduated in April 2008 from the University of Michigan with a Bachelor's degree in Industrial and Operations Engineering and Economics. He will enter the workforce as a management consultant in August of that year. mjcoop@umich.edu.

OMER TSIMHONI is an Adjunct Assistant Professor in the Industrial and Operations Engineering Department at the University of Michigan, where he teaches simulation to undergraduate and graduate students. He is an Assistant Research Scientist at the University of Michigan Transportation Research Institute (UMTRI), where he studies the human factors of driving. His research interest is in the application of simulation and optimization tools to cognitive and physical modeling of human performance.

omert@umich.edu.