A Markov Chain Approach to Baseball

Article *in* Operations Research · February 1997

DOI: 10.1287/opre.45.1.14

Bruce Bukiet
New Jersey Institute of Technology

José Luis Palacios
University of New Mexico

A MARKOV CHAIN APPROACH TO BASEBALL

BRUCE BUKIET and ELLIOTTE RUSTY HAROLD

New Jersey Institute of Technology, Newark, New Jersey

JOSÉ LUIS PALACIOS

Universidad Simón Bolívar, Caracas, Venezuela

(Received August 1992; revisions received May 1993, November 1993, April 1994; accepted December 1994)

Most earlier mathematical studies of baseball required particular models for advancing runners based on a small set of offensive possibilities. Other efforts considered only teams with players of identical ability. We introduce a Markov chain method that considers teams made up of players with different abilities and which is not restricted to a given model for runner advancement. Our method is limited only by the available data and can use any reasonable deterministic model for runner advancement when sufficiently detailed data are not available. Furthermore, our approach may be adapted to include the effects of pitching and defensive ability in a straightforward way. We apply our method to find optimal batting orders, run distributions per half inning and per game, and the expected number of games a team should win. We also describe the application of our method to test whether a particular trade would benefit a team.

People have long sought better ways to compare baseball players and teams and have used voluminous statistics in this quest. Much work has focused on ranking individual player abilities by computing an index based on details of the player's performance. Examples of such ranking methods include batting average, slugging average, scoring index (D'Esopo and Lefkowitz 1960) and run production models (e.g., Cover and Keillers 1977, Lindsey 1977, Pankin 1978, Bennett and Fleuck 1983). Although such efforts are useful for comparing individual players, they do not give insight into the performance of a team. In this paper, we introduce a Markov chain method for evaluating the performance of baseball teams as well as the influence of a particular player on team performance. This approach enables us to study several problems that cannot be investigated using earlier identical player approaches. In particular, we study in a systematic way the influence of batting order on a team's performance. Previously, simulations of hundreds of thousands of baseball games were performed to study only a few batting orders. (Freeze 1974) We also apply the Markov chain approach to estimate the number of games a team can expect to win in a season, to calculate the distribution of runs scored in each inning, and to quantify the influence of a trade on the number of games a team wins.

At any time in a half-inning, a baseball game can be in one of 25 states. Twenty-four of these states correspond to zero, one, two, or three runners on the bases (eight possibilities) and zero, one, or two outs. The twenty-fifth state corresponds to the end of the inning when the third out occurs. For example, a runner on second base, no one on first or third, and one out is one state. A runner on second base, no one on first or third, and two outs is another. A 25×25 transition matrix is set up for each player. The entries of this matrix are the probabilities for that player, in one plate appearance, to change any state of the game to any other state. If the exact probabilities of a runner changing one state of a game to another are not known, then we can fill the transition matrix by using any of several simple models of how particular hits advance runners combined with easily available statistics about how often given batters get given types of hits. If extra information is available, our method can take advantage of it. For example, suppose that a given batter is a "clutch" hitter and is twice as likely to hit a home run when the bases are loaded as when they're not. Then we double the matrix elements representing transitions from the states with three runners on base to the states with no runners on base and the same number of outs. (Other entries would then be scaled so that the sum of the entries in each row is unity.) Pitching and defense statistics can influence batter transition matrices as well. The transition matrices for nine batters in a given order determine the run distribution produced by this lineup.

Run distributions for different batting orders can be used to determine the optimal batting order for a set of nine players. We define the optimal batting order as the one leading to the largest expected number of runs scored

Subject classifications: Probability Markov processes: Markov process to model baseball. Recreation and sports: Finding optimal batting orders in baseball.
Area of review: STOCHASTIC PROCESSES & THEIR APPLICATIONS.

0030-364X/97/4501-0014 $05.00

in a nine-inning game. This requires exhaustive computation for the 9! (or 362,880) possible batting orders for nine players. Our computations show that on over half of the National League teams tested, the team "slugger" (the player with the highest slugging average) should not bat fourth and the pitcher should not bat last. We also develop three other algorithms to find near-optimal and near-worst batting orders. These algorithms reduce the search space to 44, 140, and 987 batting orders, respectively, to find a batting order which can expect to come within one run per season (162 games) of the optimal order.

Run distributions also can be used to estimate the number of games a team should win with a given batting order. Here the expected number of runs does not yield enough information, since the team with the greatest average number of runs in a season does not win all of its games. By comparing the run distributions produced by our optimal and worst batting orders, we can quantify the effect of batting order on the expected number of games a team will win. Freeze stated that the effect of batting order is less than three wins per season. Our results show a typical batting order effect of about four wins per season with an extreme case of five and half wins. As a final application of the Markov chain approach, we evaluate the effect trading players has on the expected number of wins per season.

1. THE MARKOV PROCESS METHOD

A baseball game can be thought of as a set of transitions occurring due to each player's plate appearance. When a player comes up to bat, he finds himself in one of 24 possible situations. There are eight possibilities for the distribution of runners on base. There either is or isn't a runner on each of first, second, and third bases ($2^3 = 8$). Furthermore, there are zero, one, or two outs ($3 \times 8 = 24$). The half inning ends with the third out, which we treat as a twenty-fifth state.

In the course of a plate appearance, runners may advance, and the number of outs may increase or remain the same. No matter how the batter performs, the team will find itself in one of the 25 situations described above on the completion of the plate appearance. If the probability of the player changing the state of the game from any situation to any other situation is known, a transition matrix can be set up which will have the following form:

$$\mathbf{P} = \begin{pmatrix} A_0 & B_0 & C_0 & D_0 \\ 0 & A_1 & B_1 & E_1 \\ 0 & 0 & A_2 & F_2 \\ 0 & 0 & 0 & 1 \end{pmatrix}, \tag{1}$$

where the As, Bs, and Cs are 8×8 block matrices; and D_0, E_1, and F_2 are 8×1 column vectors. The zeroes in the middle two rows of $\mathbf{P}$ represent 8×8 blocks of zeroes. The last row contains 24 zeroes and a one. The subscripts represent the number of outs at the start of the plate appearance.

The columns of the 8×8 blocks represent transitions to the states no runner on base, runner on first base, runner on second base, runner on third base, runners on first and second, runners on first and third, runners on second and third, and bases loaded, respectively. The A blocks represent events which do not increase the number of outs in the inning. The B blocks represent events that increase the number of outs by one but do not end with three outs, i.e., events which lead from no outs to one out and from one out to two outs. The C block represents events which increase the number of outs from zero to two. The vectors D_0, E_1 and F_2 represent events which lead to three outs. D_0 represents the probability of going from zero outs to three outs, E_1 represents the transition from one out to three outs, and F_2 represents the transition from two outs to three outs. The 8×8 blocks of zeroes represent transitions which decrease the number of outs in the inning and thus have probability zero.

Another way to look at $\mathbf{P}$ is that the first eight columns of the matrix represent transitions to states with no outs. The next eight columns represent transitions to one-out states. The 17th through 24th columns represent transitions to states with two outs, while the 25th column represents transitions to the absorbing three-outs state. Rows represent transition probabilities *from* these states, respectively. $\mathbf{P}$ may be constructed for any batter if the relevant statistics about his performance are known. Even if not all the desired statistics are available, reasonable values for the entries of $\mathbf{P}$ can be found using simple models and whatever statistical information is available about the player.

Certain transitions result in the scoring of runs. The number of runs scored in each transition is simple to calculate. For example, moving from the state with zero outs and men on first and second base to the state with zero outs and a man on second base in a single plate appearance requires that two runners score. The number of runs scored via each transition is stored and used to calculate the distribution of runs for the team.

To study the influence of batting order on the number of runs scored in a game, a transition matrix $\mathbf{P}$ is calculated for each of the nine players. In our calculations, we assume that the same nine players play the entire game. To find the distribution of runs in a game, we need to keep track of the probability of scoring any number of runs until the current at bat.

We describe the calculation for one inning of play and then describe how to extend the method to study a nine-inning baseball game. The inning begins with no outs and no one on base, which is represented by the 1×25 vector, $\mathbf{u}_0$, with first entry unity and all other entries zero. We call the vector representing the current state of the system $\mathbf{u}_n$, where the subscript n indicates that n batters on the team have already taken their turns at bat. If we perform the multiplication $\mathbf{u}_n \times \mathbf{P}_{n+1}$ (where $\mathbf{P}_{n+1}$ is the current batter's transition matrix) we obtain the probability distribution of states in the inning after $n + 1$ batters. Since we need to keep track of the number of runs scored up to this point in the inning, we need vectors representing these runs. Thus, in our computations, we start with a matrix $\mathbf{U}_0$ with 21 rows (representing zero to 20 runs) and 25 columns. Each row represents the number of runs that have scored thus far while the columns represent the current state (bases with runners and number of outs) in the inning. The probability of a team obtaining more than 20 runs in an inning is negligible. The record for most runs by one team in an inning in a Major League game is 18 by Chicago in 1883 (*Sporting News* 1992). In high school and Little League games where teams are occasionally mismatched enough to make 20-run innings possible, the 10-run rule exists to keep the losing team from being too humiliated. The combination of high star salaries, finite owner wallets, and the draft keep Major League games from being this mismatched. Later, we will limit the maximum number of runs a team may accumulate in a nine-inning game to 20.

Each time a transition occurs which causes a run to score, the probability of this occurrence is translated down one row of $\mathbf{U}_{n+1}$ to account for the scoring of this run. In our computations, we actually decompose each of the **P** matrices into five matrices—**P0**, **P1**, **P2**, **P3**, and **P4**—where the values in each matrix correspond to transition probabilities leading to the scoring of zero runs, one run, two runs, three runs, and four runs, respectively, in a single plate appearance. For example, the matrix leading to four runs, **P4**, has at most one nonzero entry in any 8×8 block. It is the probability of obtaining a home run (or a four-base error) when the bases are loaded. Summing the results of multiplying $\mathbf{U}_n$ by each of these five matrices yields $\mathbf{U}_{n+1}$.

$$\begin{aligned}\mathbf{U}_{n+1}(\text{row } j) &= \mathbf{U}_n(\text{row } j)\mathbf{P0} + \mathbf{U}_n(\text{row } j-1)\mathbf{P1} \\ &+ \mathbf{U}_n(\text{row } j-2)\mathbf{P2} + \mathbf{U}_n(\text{row } j-3)\mathbf{P3} \\ &+ \mathbf{U}_n(\text{row } j-4)\mathbf{P4}. \end{aligned} \tag{2}$$

After many multiplications and translations, we find that virtually all entries of $\mathbf{U}_n$ outside the three-out state go to zero. (Innings in which more than 20 runs score are counted as 20-run innings.) The 25th column of the final matrix $\mathbf{U}_\infty$ (after the stop criterion is reached) gives the distribution of runs in an inning for the team being considered.

Thus, the algorithm consists simply of multiplying the matrix $\mathbf{U}_0$ by the transition matrix $\mathbf{P}_1$ for the first batter in the lineup, then multiplying the result by the transition matrix $\mathbf{P}_2$ for the second batter, and so on until nine batters have been up and we return to the first batter. All the while, we keep track of the runs scored by performing the necessary translations. If we denote the transition matrix for each batter with the subscript of the position where he hits in the batting order, the distribution of runs after many batters becomes

$$\mathbf{U}_0\mathbf{P}_1\mathbf{P}_2\mathbf{P}_3 \ldots \mathbf{P}_9\mathbf{P}_1\mathbf{P}_2 \ldots . \tag{3}$$

In order to consider a nine-inning game, we simply give $\mathbf{U}_n$ nine times as many (or 189) rows with each set of 21 rows representing an inning. When the three-out state in one inning is reached, the results are moved to the zero-out state of the following inning with the same number of runs scored, i.e., 21 rows down and 24 columns to the left. We perform the computation until the probability that there are 27 outs is greater than 0.999. The result is the distribution of runs in a nine-inning game for a particular lineup of nine players. This implementation of our algorithm allows for a total of 20 runs in a nine-inning game for a team. This limit can be changed by adjusting a few values in the computer code. There will be a small error introduced due to games in which both teams obtain 20 or more runs. The error in truncating at 20 runs is reasonable, since only twice this century have both teams in a Major League game scored at least 20 runs. The Cubs defeated the Phillies 26–23 on August 25, 1922 (*New York Times* 1922), and the Phillies defeated the Cubs 23–22 in ten innings on May 17, 1979 (UPI 1979). Both games took place at Wrigley Field.

2. CALCULATION OF THE SCORING INDEX

We will need a means of ranking individual players' offensive abilities to develop efficient algorithms for computing near-optimal batting orders for a baseball lineup. Of the available rankings, we have chosen the scoring index of D'Esopo and Lefkowitz because it is most closely related to our method of determining run production. Scoring index is defined as the expected number of runs per inning for a team composed of nine copies of the batter being ranked using a particular deterministic model of runner advancement. It can be calculated by a one-inning version of the the Markov chain method described in the previous section. Since scoring index uses a special case of our Markov process transition matrix, while calculating it we will demonstrate one means of filling a player's transition matrix. We will also describe the results that even such a simple model for the dynamics of baseball can yield (as well as some of the model's shortcomings).

In the scoring index model the following assumptions for advancing players on the basepaths are made based on five allowable offensive possibilities:

Out: base runners do not advance. Outs increase by one.

Walk: runners advance only if forced.

Single: runner on first moves to second base. Other runners score.

Double: runner on first advances to third base. Other runners score.

Triple: all base runners score.

Home Run: all base runners and batter score.

A batter who gets on base is credited with an out if a runner already on base is forced out on the same play. This is quite natural for our model since it does not consider which runners are on base, merely which bases are occupied. In baseball statistics, this is reflected in lower batting, slugging, and on-base averages for batters who reach base while their teammates are forced out. Only minor changes in the following analysis would be needed to study similar deterministic models. This would be accomplished merely by changing the appropriate entries in the transition matrix.

For this model, the 25×25 transition matrix is given by

$$\mathbf{P} = \begin{pmatrix} A & B & 0 & 0 \\ 0 & A & B & 0 \\ 0 & 0 & A & F \\ 0 & 0 & 0 & 1 \end{pmatrix}, \tag{4}$$

where A and B are blocks with

$$A = \begin{pmatrix} P_H & P_S + P_W & P_D & P_T & 0 & 0 & 0 & 0 \\ P_H & 0 & 0 & P_T & P_S + P_W & 0 & P_D & 0 \\ P_H & P_S & P_D & P_T & P_W & 0 & 0 & 0 \\ P_H & P_S & P_D & P_T & 0 & P_W & 0 & 0 \\ P_H & 0 & 0 & P_T & P_S & 0 & P_D & P_W \\ P_H & 0 & 0 & P_T & P_S & 0 & P_D & P_W \\ P_H & P_S & P_D & P_T & 0 & 0 & 0 & P_W \\ P_H & 0 & 0 & P_T & P_S & 0 & P_D & P_W \end{pmatrix}, \tag{5}$$

and

$$B = P_{\text{out}} \mathbf{I}. \tag{6}$$

P_W, P_S, P_D, P_T, P_H, and P_{out} are the probabilities of a player getting a walk, single, double, triple, home run, or out, respectively. I is the 8×8 identity matrix, and

$$F = (P_{\text{out}}, \ldots, \mathrm{P}_{\text{out}})^T, \tag{7}$$

is an 8×1 vector. Notice that the blocks C_0, D_0 and E_0 are zero since this model does not include double and triple plays. Off-diagonal elements of the B blocks are zero since this model does not allow runners to advance on an out. These inaccuracies somewhat offset each other.

Since all batters are identical, we can use the theory of absorbing chains (see, for example, Seneta 1980) to find the time to absorption (the three-out state) from any state in the system by deleting the 25th row and 25th column of $\mathbf{P}$ to get the 24×24 matrix Q. Now the "fundamental matrix" $(I - Q)^{-1}$ provides all the information: its (i, j) entry, $(I - Q)_{ij}^{-1}$, gives the expected number of visits starting from state i to state j before absorption into the three-out state. Also, the expected absorption time, $E_i T$, into the three-out state starting from state i is given by the sum:

$$E_i T = \sum_{j=1}^{24} (I - Q)_{ij}^{-1}. \tag{8}$$

For the D'Esopo and Lefkowitz model:

$$Q = \begin{pmatrix} A & B & 0 \\ 0 & A & B \\ 0 & 0 & A \end{pmatrix}, \tag{9}$$

and

$$(I - Q)^{-1} = \begin{pmatrix} R & RBR & RBRBR \\ 0 & R & RBR \\ 0 & 0 & R \end{pmatrix}, \tag{10}$$

where $R = (I - A)^{-1}$.

D'Esopo and Lefkowitz calculated analytically the distribution of runs in an inning using this simple deterministic model with identical players. They also calculated the expected number of runs per game. The model predicts the scoring of 6.7 percent fewer runs than the actual results for their test cases. We validated our Markov chain method against the analytical formulas of D'Esopo and Lefkowitz for several identical batter cases and obtained the same results.

3. MORE COMPLEX MODELS FOR BATTER TRANSITION MATRICES

The batter transition matrices $\mathbf{P}$ have great flexibility for describing batter behavior. In this section, we discuss how to use the Markov chain approach while allowing for more complex possibilities than in the D'Esopo and Lefkowitz model. We provide several examples as well as a discussion of the limitations of our method.

We begin with some examples of how the inclusion of more complete baseball statistics would affect a batter's transition matrix. If a runner advances on an out or lead runners are thrown out on a hit, entries in the B blocks would change. If double plays are included in the model, there would be nonzero entries in C and E. Triple plays would lead to nonzero entries in D. If a player performs better when there is one out than when there are no outs, entries in rows 9 through 16 of $\mathbf{P}$ change. If a batter performs better (or worse) with runners in scoring position (i.e., on second or third base) than without runners on base, rows 5–8, 13–16, and 21–24 would change. The above are examples of how more of the events that occur in baseball can be included using the Markov chain approach.

The Markov chain approach is not restricted to transition matrices which remain static throughout a game or season. If the data are available, variations can be made in the values of the entries of $\mathbf{P}$ during a game or season. For example, if a batter's performance improves during the course of a game, the transition matrix entries can be made functions of the number of at-bats or the current inning. Modeling the tiring of the pitcher could be included in exactly the same way; just vary the entries in the opposing batters' transition matrices. A batter's transition matrix also can be made to vary based on his position in the lineup or the position in the field he's playing that day. One can take this idea too far since not enough empirical data are available to set up a transition matrix for all possible occasions. For example, it is unlikely there are much data on the offensive performance of a given batter in cold weather, during daytime, on grass, facing Dwight Gooden, in the sixth inning, with a runner on second base and one out. Filling a realistic transition matrix with empirical player data for all situations would require complete batter-by-batter records of ballgames such as the database maintained by STATS. (Dewan and Zminda 1992). Although such data were not used in this study, they have been compiled for the last several seasons and are available for a price. This work and earlier work of others (e.g., D'Esopo and Lefkowitz) show that simple models can give accurate results.

It now should be clear that almost any batting data can be incorporated in our Markov chain approach in a relatively straightforward way. The only offensive information that cannot be included is which runners are on which bases (rather than simply which bases are occupied) and performance based on the current score of the game—for example, a transition matrix describing a batter's performance in games when his team is behind by one run could not be included. Neither could bunting or pinch hitting based on the current score of the game be incorporated into our approach. However, pitching and defense may be included by making the opposing batter's transition matrix entries functions of the pitchers and fielders they are facing that day. We give a simple example in Section 6.1.

4. WINNING THE GAME

We now describe how to use the distribution of runs in a nine-inning game to predict the outcome of the game, including the possibility of a victory in extra innings. Once the distribution of runs in a game is known, we can calculate the expected number of wins for a team in a series of games between two teams.

A team wins a game by scoring more runs than the opposing team. The probability of the first team winning a game in nine innings is

$$\sum_{i=1}^{20} \left[S(\text{Team 1})_i \sum_{j=0}^{i-1} S(\text{Team 2})_j \right], \tag{11}$$

where $S(\text{Team 1})_i$ is the probability that Team 1 scores i runs in a nine-inning game. The probability of a tie sending the game into extra innings is

$$\sum_{i=1}^{20} S(\text{Team 1})_i S(\text{Team 2})_i. \tag{12}$$

In our implementation of the Markov chain method, we only allow for 20 runs per team per nine innings. There will be a small error introduced due to games in which both teams obtain 20 or more runs. (As mentioned earlier, this error is small since only twice this century have both teams in a Major League game scored at least 20 runs.)

In order to estimate the probability of each team winning in extra innings, we consider increments of one inning and follow along the lines of Dubner (1988). We approximate one-inning run distributions by extrapolating the nine inning distribution backward to a one-inning distribution. We denote the probability that a team scores n runs in an inning by R_n and then let

$$R_0 = S_0^{1/9}, \tag{13}$$

$$R_1 = \frac{S_1}{9R_0^8}, \tag{14}$$

$$R_2 = \frac{S_2 - \frac{9!}{7!} R_1^2 R_0^7}{9R_0^8}, \tag{15}$$

etc.

Thus the probability of Team 1 winning in the first extra inning (given that the game was tied after nine innings) is

$$\sum_{i=1}^{20} R(\text{Team 1})_i \left[\sum_{j=0}^{i-1} R(\text{Team 2})_j \right], \tag{16}$$

and the probability of a tie after one extra inning is

$$\sum_{i=1}^{20} R(\text{Team 1})_i R(\text{Team 2})_i. \tag{17}$$

The probability of Team 1 winning an extra inning game equals the probability that Team 1 wins in the first extra inning, plus the product of the probability that the extra inning game is tied after the first extra inning and the probability that Team 1 wins an extra inning game that has at least two extra innings. Since we assume the extra innings have identical distributions, using expressions (16) and (17) we have

$$\begin{aligned} &\text{P[Team 1 wins an extra inning game]} \\ &\quad = \sum_{i=1}^{20} R(\text{Team 1})_i \left[\sum_{j=0}^{i-1} R(\text{Team 2})_j \right] \\ &\qquad + \sum_{i=1}^{20} R(\text{Team 1})_i R(\text{Team 2})_i \\ &\qquad \times \text{P[Team 1 wins an extra inning game]}. \end{aligned} \tag{18}$$

Solving for the probability that Team 1 wins an extra inning game yields

$$\begin{aligned} &\text{P[Team 1 wins an extra inning game]} \\ &\quad = \frac{\sum_{i=1}^{20} R(\text{Team 1})_i [\sum_{j=0}^{i-1} R(\text{Team 2})_j]}{1 - \sum_{i=1}^{20} R(\text{Team 1})_i R(\text{Team 2})_i}. \end{aligned} \tag{19}$$

Using expressions (11), (12), and the right-hand side of (19) gives the probability that Team 1 wins a game.

$$\begin{aligned} &\sum_{i=1}^{20} [S(\text{Team 1})_i \ \sum_{j=0}^{i-1} S(\text{Team 2})_j] \\ &\quad + \sum_{i=1}^{20} S(\text{Team 1})_i S(\text{Team 2})_i \\ &\quad \times \frac{\sum_{i=1}^{20} R(\text{Team 1})_i [\sum_{j=0}^{i-1} R(\text{Team 2})_j]}{1 - \sum_{i=1}^{20} R(\text{Team 1})_i R(\text{Team 2})_i}. \end{aligned} \tag{20}$$

Multiplying this expression by the number of games the two teams play against each other yields the expected number of wins for Team 1 in the series.

5. OPTIMAL BATTING ORDERS

We study the effect of batting order on team performance. We assume a player's batting ability does not vary with his position in the batting order, although given the appropriate statistical data it is easy to include the data in the player's transition matrix **P**. In what follows, all batters are ranked according to scoring index as described in Section 2; e.g., the "best" batter is the one with the highest scoring index. Since a team of nine batters has $9! = 362{,}880$ possible batting orders, we present three efficient algorithms for finding a near-optimal batting order as well as results found by performing the computations for all 9! permutations.

Because of our belief that the designated hitter rule is a blotch on the grand sport of baseball equalled only by the Black Sox scandal of 1919, we consider only National League teams in what follows. Since pitchers don't bat in the American League, the absence of a hitter who is demonstrably worse than all other players on the team might lead to different batting orders in that league than we recommend here.

In various experiments, we have found that the batting order that produces the highest expected number of runs in a game is also the order that produces the greatest expected number of wins for that team. In order to find the batting order for a set of nine players which maximizes the expected number of runs produced, one must test all $9! = 362{,}880$ possible lineups. Our highly optimized C code for finding a nine-inning run distribution for one lineup using the D'Esopo and Lefkowitz runner advancement model takes about 1.3 seconds on a SPARCstation2 after compilation with gcc 2.1 with maximum optimization (−O2). The matrix is moderately sparse, and it is important to avoid doing any unnecessary multiplications. The evaluation of 9! possible orders takes about five and a half CPU days per team. After testing several algorithms, we found the following procedures give very good results in optimizing the expected number of runs.

Since we expect the placement of the best and worst batters to make the greatest difference in team run production, we search for the batting order that produces the highest expected number of runs for a team made up of the best and worst batters and seven equal players, each of whom is an average of the seven unused players in the lineup. The position of the best and worst players which maximizes expected number of runs per game for this semi-averaged team is then set in this near-optimal order. We thus test $9 \times 8 = 72$ batting orders to place the first two players. Then we consider the players with the second best and second worst scoring indexes and average the five remaining players. We consider only batting orders in which the best and worst performers are left in their near-optimal spots in the lineup and the two players now being considered are placed in all combinations of the seven remaining spots, while the five averaged players fill out the lineup. We continue this double-placement process until all positions are filled. Thus, rather than having to test over 360,000 lineups, we test just $(9 \times 8) + (7 \times 6) + (5 \times 4) + (3 \times 2) = 140$ lineups; the ninth batter's position is given by default. The calculation of our near-optimal lineup takes about three CPU minutes.

We also investigated a quicker, single-placement approach in which we placed the players in order of the distance of their scoring indexes from the team mean one at a time while averaging the remaining players. The calculation of this near-optimal lineup tests only nine batting orders to place the first batter, eight orders to place the second, and so on. Since this requires only 44 total tests, a near-optimal lineup can be found in one minute of CPU time.

These two approaches often disagreed on the near-optimal order, but neither consistently produced better teams than the other. For a given team, both approaches can be tried in only four minutes of CPU time. The expected number of runs per game each approach produced typically differed by less than 0.01 runs per game, less than two runs per season.

Using these models but instead placing the batters in the positions which *minimize* the expected number of runs gives near-worst orders. We can compare the near-optimal and near-worst teams and find how many games per season each could expect to win against the other. This gives a measure of the significance of batting order in team performance. One might also consider how many games each of these two teams would win in a season against a reference team or a reference run distribution. The results would be similar if the reference team were at all comparable to the team being studied. However, since Freeze calculated his batting order effect by comparing a best order against a worst order, we do the same.

To find the probability that a team wins a game against another team, we calculate the nine-inning run distributions for each team. For a given batting order, the Markov chain approach produces a run distribution for a nine-inning game. We use expression (20) in conjunction with the run distributions to find the exact probability of each team winning the game.

We present an example of our procedures to find the optimal batting order for the set of players in Table I. All player statistics were taken from Neft and Cohen (1990). The players were chosen from the 1989 Atlanta Braves; for simplicity, we used the D'Esopo and Lefkowitz runner advancement model to set up the players' transition matrices. We applied the single- and double-placement algorithms, minimizing expected number of runs to calculate near-worst batting orders. For comparison we also list exact best and worst orders found by testing all 9! possible orders.

The Braves' best order could be expected to win 86.5 games of a 162-game season against the same players using the worst batting order. The effect of batting order is 5.5 games in this case (86.5–162/2). Our experiments with the teams in the 1989 National League show that batting order has an average influence of about four games per season.

Table I
Best and Worst Orders for the 1989 Atlanta Braves (as calculated by the single and double-placement models and by full enumeration)

Players	Scoring Index	Best Orders: Single Placement	Best Orders: Double Placement	Best Orders: True	Worst Order: Single Placement	Worst Order: Double Placement	Worst Order: True
Smith	0.887	Smith	Murphy	Perry	Pitcher	Pitcher	Pitcher
McDowell	0.649	McDowell	Perry	Smith	Murphy	Murphy	Treadway
Blauser	0.484	Perry	Treadway	McDowell	Treadway	Treadway	Perry
Treadway	0.430	Blauser	Smith	Blauser	Perry	Perry	Davis
Perry	0.416	Treadway	McDowell	Treadway	Davis	Davis	Thomas
Murphy	0.411	Murphy	Blauser	Murphy	Thomas	Thomas	Blauser
Thomas	0.241	Thomas	Thomas	Thomas	Blauser	Blauser	McDowell
Davis	0.198	Davis	Davis	Pitcher	McDowell	McDowell	Murphy
Pitcher	0.078	Pitcher	Pitcher	Davis	Smith	Smith	Smith
Expected Runs Per Game		3.523	3.539	3.557	3.264	3.264	3.260
Win-Loss		85.9–76.1	86.1–75.9	86.5–75.5	76.1–85.9	75.9–86.1	75.5–86.5

Namely, if the best order for a team played against the worst order, it would win 84.2–86.5 games in a 162-game series, i.e., the effect of batting order is 3.2–5.5 games for the 1989 National League. Freeze tested 13 lineups in over 200,000 games by performing simulations using a *Sports Illustrated* model of the game. He concluded that the effect of lineup is less than three wins per season. The bulk of his work involved playing lineups made up of the same players against each other.

We also performed full enumeration testing of batting orders for the teams in the 1989 National League to find the genuine optimal order for given a set of nine frequently used players for each team. Since individual pitchers play far fewer games than other players, we averaged the offensive data of the pitchers. Although our approximate approaches found the optimal order only once, the best of the two near-optimal orders for a given team was never off by more than 0.022 runs per game, and on average it was off by only 0.012 runs per game, a difference of less than two runs per year. In our computations, the optimal lineup obtained 30 to 50 more runs per season than the worst. Thus, we found near-optimal batting orders by testing less than 200 lineups instead of over 360,000 for each team.

More interestingly we found that in the optimal order the best hitter (the player with highest scoring index) on a team most often (seven teams out of twelve) should have batted second and should have batted third on two of the remaining teams. On only three teams should the slugger (the player with the highest slugging average) have batted in the traditional fourth position; and on only two teams should the player with the best chance of hitting a home run have batted fourth. We found that the pitcher (i.e., the player with the worst scoring index), who almost always bats last in actual games, should in fact have batted either seventh or eighth on all but one team, the Astros. (See Table II.) We hypothesize that it is more important to give the slugger more at bats by putting him earlier in the order than it is to have him drive in more first-inning runs by having him bat cleanup. We also suspect that it is important to keep the pitcher away from the slugger and the other good batters so that the pitcher neither bats cleanup for the slugger nor has the slugger batting cleanup for him. This explains why the pitcher should in general not bat ninth when the slugger bats earlier than fourth.

Once we had optimal orders for each team we viewed the lineups according to scoring index. (See Table II.) Analysis of this table suggests several rules of thumb for producing optimal orders. Of the possible criteria for batter placement we selected the following:

(1) The best batter (by scoring index) should bat second, third, or fourth.

(2) The second best batter should bat somewhere in the first through fifth positions.

(3) The third and fourth best batters should bat somewhere in the first through sixth positions.

(4) The fifth best batter should come up first or second, or fifth through seventh.

(5) The sixth best batter should bat in any position except eighth or ninth.

Table II
Optimal Batting Order, 1989 National League

Team	Optimal Batting Order Listed by Scoring Index
Astros	5 6 3 1 2 4 7 8 9
Braves	5 1 2 3 4 6 7 9 8
Cardinals	2 1 6 3 5 4 9 7 8
Cubs	6 1 2 3 5 4 8 9 7
Dodgers	2 5 4 1 3 7 6 9 8
Expos	5 1 2 6 4 3 8 9 7
Giants	7 3 2 1 4 6 5 9 8
Mets	2 1 3 4 5 6 9 7 8
Padres	3 1 2 4 5 7 6 9 8
Phillies	4 2 1 3 5 6 9 7 8
Pirates	2 4 1 3 6 5 9 7 8
Reds	2 1 3 4 5 6 9 8 7

(6) The seventh best batter can bat either first or sixth through ninth.

(7) The eighth and ninth best batters bat in the last three positions.

(8) Either the second or third best batter must be placed immediately before or immediately after the best batter.

(9) The worst batter must be placed four through six positions after the best batter.

(10) The second worst batter must be placed four through seven positions after the best batter.

We used these criteria to create a code that tests the 987 batting orders meeting these conditions. This near-optimal, criteria-based code takes about 20 minutes of CPU time to run. We note that those batters whose statistics place them near the top or bottom of the team in scoring index can more easily be localized in the batting order than the other players. One reason for this may be that the differences between the scoring indexes of mid-ranked hitters (third through seventh in scoring index) are relatively small on most teams. Our placement algorithms are somewhat dependent on strong differences in the individual players' abilities.

Since the criteria-based code was rigged to produce the twelve optimal orders for the 1989 National League, we tested it instead on the 1969 National League. It found a better order for each of the twelve teams than either the single- or double-placement approximation found. These criteria-based batting orders averaged about two runs per season per team better than the single- or double-placement orders.

We then ran full enumerations for the 1969 National League teams. Our criteria-based code found the optimal orders for five of these teams, and on four other teams the orders found came within 0.001 runs per game of the optimal value. On the other three teams the results were within 0.01 runs per game. On average, the optimal order was only 0.3 runs per season better than the order chosen by the criteria-based code.

We also found the worst order for each team in the 1989 National League. On 11 out of 12 of these lineups the pitcher batted first. The better batters (by scoring index) tended to come up later in the worst performing orders, with the best hitter never appearing earlier than seventh in the lineup but usually appearing in either eighth or ninth position.

We note that other authors have considered baseball as a Markov process. Trueman (1977), for example, developed a large set of recursive equations using transition probabilities mostly to study baseball strategies (bunting and stealing). His method also could be used for comparing batting orders, but it is not as flexible as the approach described here. Bellman (1977) considered baseball as having 2593 possible states in an inning in a theoretical work concerned with managerial decision making.

6. OTHER APPLICATIONS

6.1. The Pennant Race

Our Markov chain approach can be used to estimate the win–loss record of each team in the league. We approximated the National League of 1989 by choosing as batters the most frequently used players at each position on each team and averaging the offensive performance of the pitchers. We further assumed that managers would use the near-optimal batting order calculated by the double-placement approximation for these nine players. We picked this approximation because its suggested orders most closely matched those used in actual games (i.e., batter with highest slugging average bats fourth and pitcher bats last). A substantial savings in run time could be made at the expense of a substantial increase in time spent gathering and entering data if instead of calculating optimal batting orders we merely entered the individual orders actually used by each team during each game. We used the D'Esopo and Lefkowitz model to advance runners. To include pitching and defense in our model, we scaled the offensive production of each variable (specifically walks, singles, doubles, triples, and home runs) for a team by the ratio of the on-base average of the league against the opposing team's pitchers to the mean on-base average for pitchers in the league, and changed the probability of obtaining an out so that the sum of each row of the transition matrices remained one. For each pair of teams, we found the near-optimal batting orders and the expected number of wins for each season series. The results are presented in Table III.

Differences from the actual values of wins in the season for each team may stem in part from the same nine "regulars" playing every game of the season in our model. The offensive records of the other players (who made over one-third of total plate appearances) were discarded. Also neglected were the effects of pinch hitting, double, and triple

Table III

Standings Predicted by a Non-identical Batter Model for the 1989 National League

National League East		
Team	Actual Games Won	Model Games Won
Cubs	93	86.9
Mets	87	88.6
Cardinals	86	87.4
Expos	81	82.6
Pirates	74	78.4
Phillies	67	67.9

National League West		
Team	Actual Games Won	Model Games Won
Giants	92	90.7
Padres	89	84.0
Astros	86	72.8
Dodgers	77	73.7
Reds	75	79.5
Braves	73	79.6

Table IV
Expected Number of Runs Scored in Each Inning

Inning	1	2	3	4	5	6	7	8	9
Avg. Runs Model (NL 1989)	0.52	0.35	0.42	0.39	0.40	0.40	0.40	0.40	0.40
Lindsey's Data (1961)	0.53	0.38	0.53	0.44	0.45	0.52	0.46	0.50	0.45

plays and bases reached on errors. Furthermore, the distribution of hits against the pitchers of each team was not considered but could be if data were available. For instance, if the pitchers on a particular team intentionally walked home-run hitters very often and thus gave up fewer home runs but more walks than the league average, it would not appear in the above calculation.

The average number of runs scored per game in our calculation is 3.67. This is about seven percent less than the actual value of 3.95 for 1989 National League teams. This is the same result D'Esopo and Lefkowitz found in their experiments. It is most likely the result of a conservative runner advancement model where, for example, a runner on first base does not score on a double.

6.2. Runs per Inning

Lindsey (1961) noted that the average number of runs scored in an inning depends on the inning. More runs are scored in the first inning than in any other, and the second inning has the lowest number of runs scored. He found this by studying about 1800 major league games. Of course models that treat a batting order as having identical players find all innings to be equal in average run production. However, our Markov chain method for nonidentical players can be used to find the average number of runs in each inning. All that is needed is to save the run distribution data each time we move from one inning to the next. For the double-placement near-optimal teams found in the previous section, we present the expected number of runs in each inning along with Lindsey's real-life data in Table IV.

Lindsey's data from the early 1960s suggests that these numbers are reasonable, at least for the first few innings. In the first inning the best hitters usually step up to the plate, and more runs can be expected to score. On the other hand, in the second inning the bottom of the lineup often comes up to bat and thus produces the lowest scoring inning. As our model game progresses, the probability of each batter being the first one up equalizes. This is why the variation in number of runs scored per inning decreases as the game goes on.

In the third inning we agree qualitatively with Lindsey but not quantitatively. We are similarly low in the later innings. The additional jump in the third inning that Lindsey saw is probably due not only to the likelihood of the good batters at the top of the order coming up again (which explains the increase predicted by our model) but also to these batters getting a second chance to face the pitcher. The jump in the sixth inning that Lindsey's real-life data show, but we do not, is doubtless due to the tiring of the pitcher and to pinch-hitting.

Table V
Expected Standings in the National League East in 1989 if Darryl Strawberry of the Mets Were Traded for Milt Thompson of the Cardinals

National League East		
Team	Pre-Trade Model Games Won	Post-Trade Model Games Won
Cubs	86.9	87.0
Mets	88.6	86.9
Cardinals	87.4	88.9
Expos	82.6	82.6
Pirates	78.4	78.4
Phillies	67.9	67.9

6.3. Evaluation of Trades

Finally, we discuss the evaluation of trades using our method. If we assume that past performance is indicative of future performance and that the performance of a player does not depend on which team he plays for, then we can calculate the expected increase or decrease in number of team wins if one player is traded for another. If the players do not play the same position, one has to consider who takes the place of the player traded at that position and who is removed from the lineup to make room for the new player. We calculate a particularly simple example in which both players are outfielders.

We chose Milt Thompson of the St. Louis Cardinals, who was a solid singles hitter on a team without a particularly good home run hitter, and Darryl Strawberry of the Mets, a team that had two other home run hitters and might have benefited from an additional solid singles hitter. The Mets finished second in their division and six games out in 1989, and the Cardinals finished right behind them in third and seven games out. Milt Thompson of the Cardinals batted 0.290 in 1989 with only four home runs and a 0.393 slugging average, while Strawberry, only a 0.225 hitter that year, had 29 home runs and a 0.446 slugging average. We used the technique of Section 6.1 to find that this trade would have increased the Cardinals' expected number of runs per game by 0.07 from 3.81 to 3.88, or about 11 runs per year. It would have *decreased* the Mets' expected number of runs per game by 0.08 from 3.78 to 3.70, or 13 runs per year. In our computations, this resulted in the Cardinals winning 1.5 more games during the season and the Mets winning 1.7 fewer games. As we see from Table V our model results moved the Cardinals from second to first, while the Mets dropped from first to third. The Cubs moved up to second, and all other teams retained their rankings. Obviously these changes in win–loss records are well within our margin of uncertainty. Nonetheless it does show that it is not a very good idea to trade home-run hitters for singles hitters, even when the singles hitter gets on base a lot more often and a team has more than two home-run hitters.

7. CONCLUSIONS

We have presented a Markov chain approach to the study of baseball which goes beyond earlier studies to allow consideration of real teams made up of nonidentical players. The method is flexible and extendable. It can include as much detailed statistical information as is available and make up for what is not available with simple, deterministic models for runner advancement. Pitching and defense can be included in a straightforward way. Information on how a batter performs based on the particular runners on base or the current score of the game cannot be included. However, information based on the number of innings played so far or on the number of hitters who have come up to bat can be included easily.

The method has been shown to reproduce earlier results (for a specific deterministic model) for run distributions in baseball games. It has also been used to solve for the variation in the expected number of runs in different innings of a game. Other applications of the model have been discussed including its use to compute efficiently near-optimal batting orders for a team of nine players. We have shown that optimal batting orders can expect to win approximately four more games per 162-game season than worst orders. Testing optimal orders has demonstrated that the traditional lineup of a slugger batting cleanup while the "weakest batter comes up last" is usually not optimal. Finally, we have shown how our method could be used to evaluate trades.

ACKNOWLEDGMENTS

The authors thank Tom Carmichael for helping with some of the programming needed in this project. The authors also acknowledge support from the New Jersey Separate Budget for Research through the New Jersey Institute of Technology. The authors also thank the referees for their helpful comments.

REFERENCES

BELLMAN, R. 1977. Dynamic Programming and Markovian Decision Processes, with Application to Baseball. In *Optimal Strategies in Sports*, S. P. Ladany and R. E. Machol (Eds.). Elsevier-North Holland, New York.

BENNETT, J. M. AND J. A. FLUECK. 1983. An Evaluation of Major League Offensive Performance Models. *The American Statistician,* **37,** 76–82.

COVER, T. M. AND C. W. KEILERS. 1977. An Offensive Earned-Run Average for Baseball. *Opns. Res.* **25,** 729–740.

D'ESOPO, D. A. AND B. LEFKOWITZ. 1960. The Distribution of Runs in the Game of Baseball. SRI Internal Report.

DEWAN, J. AND D. ZMINDA. 1992. *STATS 1992 Baseball Scoreboard*. Sports Team Analysis, Chicago, IL.

DUBNER, H. 1988. Tennis Odds, Anyone? *J. Recreational Math.* **20,** 182–189.

FREEZE, R. A. 1974. An Analysis of Baseball Batting Order by Monte Carlo Simulation. *Opns. Res.* **22,** 728–735.

LINDSEY, G. R. 1961. The Progress of the Score During a Baseball Game. *Amer. Stat. Assoc. J.* **56,** 703–728.

LINDSEY, G. R. 1977. A Scientific Approach to Strategy in Baseball. In *Optimal Strategies in Sports*. S. P. Ladany and R. E. Machol (eds.). Elsevier-North Holland, New York.

NEFT, D. S. AND R. M. COHEN. 1990. *The Sports Encyclopedia: Baseball (Tenth Edition)*. St. Martin's Press, New York.

THE NEW YORK TIMES. 1922. August 26, 7.

PANKIN, M. D. 1978. Evaluating Offensive Performance in Baseball. *Opns. Res.* **26,** 610–619.

SENETA, E. 1980. *Non-negative matrices and Markov chains*. Springer-Verlag, New York.

SPORTING NEWS STAFF. 1992. *Complete Baseball Record Book, 1992*. Sporting News Publishing Company, St. Louis, MO.

TRUEMAN, R. E. 1977. Analysis of Baseball as a Markov Process. In *Optimal Strategies in Sports*. S. P. Ladany and R. E. Machol (eds.). Elsevier-North Holland, New York.

UPI. 1979. *The New York Times*. May 18, p. A19.