©Journal of Sports Science and Medicine (2006) **5,** 495-502
http://www.jssm.org

Research article

The 8th Australasian Conference on Mathematics and Computers in Sport, 3-5 July 2006, Queensland, Australia

A MATHEMATICAL MODELLING APPROACH TO ONE-DAY CRICKET BATTING ORDERS

Matthews Ovens [1] ✉ and Bruce Bukiet [2]

[1] School of Mathematical Sciences, Faculty of Science, Monash University, Australia
[2] Center for Applied Mathematics and Statistics, Department of Mathematical Sciences, New Jersey Institute of Technology, USA

Published (online): 15 December 2006

ABSTRACT
While scoring strategies and player performance in cricket have been studied, there has been little published work about the influence of batting order with respect to One-Day cricket. We apply a mathematical modelling approach to compute efficiently the expected performance (runs distribution) of a cricket batting order in an innings. Among other applications, our method enables one to solve for the probability of one team beating another or to find the optimal batting order for a set of 11 players. The influence of defence and bowling ability can be taken into account in a straightforward manner. In this presentation, we outline how we develop our Markov Chain approach to studying the progress of runs for a batting order of non-identical players along the lines of work in baseball modelling by Bukiet et al. (1997). We describe the issues that arise in applying such methods to cricket, discuss ideas for addressing these difficulties and note limitations on modelling batting order for One-Day cricket. By performing our analysis on a selected subset of the possible batting orders, we apply the model to quantify the influence of batting order in a game of One Day cricket using available real-world data for current players.

KEY WORDS: One-day cricket, batting orders, mathematical modelling

INTRODUCTION

Many cricket commentators will suggest that certain players perform best as "Number Three" in the batting line-up. Listen to almost any commentary team during the course of a One-Day game and you will hear statements based upon rules of thumb like, "Ricky Ponting is a genuine number three." This raises the question, is there really a "Batting Order Effect" and assuming there is, how would you test this? Suppose that a cricket coach decided to test every possible batting order for a team of 11 players, how many games would they have to play? With a team of 11 players, there are nearly 40 million possible line-ups, thus if they could play 1 game every day, it would take a little more than 109286 years (assuming players lived and could play for that long). However, if one has data for the ability of each of the batsmen in a cricket lineup, one can apply techniques of mathematical modelling to ascertain how well a set batting order (against a specified set of bowlers) should perform.

Previous Research

There has been little published work about the influence of batting order with respect to One-Day cricket. The earliest work on mathematical analysis of cricket was by Wood (1945) and Elderton (1945)

who studied whether cricketers' scores follow geometric progressions. Starting in the late 1980's and continuing up to the present, Clarke's and his group [e.g. Clarke (1988), Johnston et al. (1993) and Norman and Clarke (2004)] have studied cricket scoring strategies and player performance, among other cricket issues, by applying dynamic programming methods. Clarke (1988) addressed cricket strategies in terms of optimal run rates using dynamic programming techniques. Johnston et al. (1993) assessed player performance by dynamic programming and developed a ranking system to aid in assessing a player's performance. Cohen (2002) studied the probability of dismissing a team before 50 overs and the geometric nature of scoring strokes. Norman and Clarke (2004) investigated the effect of a sticky wicket and how a batting team should adjust its line-up in the longer form of the game. Although Swartz et al. (2006) did study the problem of finding optimal cricket batting orders; their work has been from a statistical simulation, rather than a mathematical modelling perspective. Swartz et al. (2006) devised a statistical method to compute the probability of each outcome for each batsman with corrections based on wickets and balls remaining. They then simulated games with various batting orders (10,000 games per batting order) and used simulated annealing to reduce number of choices of batting orders to consider.

In the current work, we apply a mathematical modelling approach to compute efficiently the runs distribution of a cricket batting order in an innings. The approach, which will be described in the following section, uses Markov Chains and is based on the method developed by Bukiet et al. (1997) that has been used over a number of years for the modelling of the run production in baseball. Among other applications, our method enables one to solve for the expected number of runs a batting order should score and the probability of one team beating another. By considering all 11! or 39,916,800 batting orders, one could potentially find the optimal batting order for a set of 11 players, i.e., the batting order that can be expected to attain the most runs. The model is set up such that the influence of defence and bowling ability can be taken into account in a straightforward manner. As we note in later sections of this paper, the time it takes to analyse a single lineup using our technique is such that evaluating all possible lineups (full enumeration) would take a prohibitively long amount of time. Thus, presently, the model is mainly for theoretical purposes in terms of finding the best batting order, but it could be applied to address some questions. For example, by considering a selected subset of the possible batting orders for 11 Australian players and applying our model, we demonstrate that there is a difference between best and worst batting orders and that this difference is significant.

METHODS

In this paper, we outline how we develop our Markov Chain approach to studying the progress of runs for a batting order of non-identical players along the lines of work in baseball modelling by Bukiet et al. (1997). We describe the issues that arise in applying such methods to cricket and how we have addressed the difficulties particular to cricket.

In a Markov process, it is not important to know how a given situation arose, just that you are in a particular situation. The probability of going from one situation to any other is known. There are a finite number of situations.

In the context of cricket, the dynamics of run production depends mainly on the interaction of the bowler and the batsman. So the game can be modelled as a sequence of one-on-one interactions. A batsman takes a turn and then we stop and have a new situation. The probability of any occurrence depends only on the current situation (who is the facing batsman, who is the bowler, who is the batsman at the other wicket (the non-striker), how many balls are left) and possibly only a small subset of that. For the most part, there are only 7 states to which a given situation will commonly transition on a single bowl of the ball.

- A batsman is dismissed and no runs score with probability P_d
- Zero runs score (no dismissal) with probability P_0
- One run is scored (no dismissal) with probability P_1
- Two runs are scored (no dismissal) with probability P_2
- Three runs are scored (no dismissal) with probability P_3
- Four runs are scored (no dismissal) with probability P_4
- Six runs are scored (no dismissal) with probability P_6

Making the situation slightly more complex is the effect of the batsman switching places. If an odd number of runs are scored, the batsman will have switched ends. Similarly, if a multiple of six balls have been bowled, the bowler has finished the over and a new bowler will begin bowling from the other end, resulting in the non-striker becoming the facing batsman. (We note that it is possible, but uncommon

to score 5 runs. Similarly, it is possible to score runs when a batsman is runout. We disregard these events, other rare offensive possibilities as well as rules concerning fielding restrictions. Our method could handle most of these at the cost of greatly increased computational time. It appears, at least in the case of modelling baseball that ignoring many rare situations makes little difference in the results as there is much cancellation between including positive events (e.g. fives) and negative events (e.g., runouts on run scoring balls)).

Let the multidimensional Matrix M have entries M (b,r,w,b_1,b_2) represent the number of balls bowled, runs scored and wickets down, the next batsman and the batsman at the other wicket, respectively. For each number of balls bowled, we can compute the probability of being in a given situation by multiplying the (multidimensional) matrix representing the set of probabilities after the b-1 balls by the probability of each of the events listed above occurring. For example, the game begins with 0 balls bowled, 0 runs scored, 0 wickets gone and batsman number 1 about to hit, with batsman number 2 at the other wicket. Thus, M (0,0,0,1,2) = 1 and all other entries of M (0,r,w,b_1,b_2) are zero. After the first ball, M (1,0,1,3,2) = P_d, M(1,0,0,1,2) = P_0, M (1,1,0,2,1) = P_1, M (1,2,0,1,2) = P_2 and so forth. These values are obtained in the general case (b balls) by multiplying each non-zero entry of M (b-1,r,w, b_1,b_2) by each of the probabilities P_d, P_0-P_6 and placing the result in the appropriate location in the M (b,r,w,b_1,b_2). After the computation has considered 300 balls (with 10 wickets down causing no future balls to be bowled) we end up with the probability of any given number of runs having been scored (the runs distribution). The computation is actually simplified by looping through the number of balls and saving only the situation after b-1 balls to compute the situation after b balls. Also, one need not keep track of wickets dismissed since the batsmen currently in the game provide that information (if batsman number 6 in the order is in the game, but 7-11 are not, then 4 batsmen have been dismissed). Thus, an 11 X 11 X 1800 (batsmen X batsmen X runs) matrix needs be maintained and updated. We note that the method automatically takes into account that the batsmen early in the lineup, if they are the best batters will face more balls than the later batsmen. Summing the product of each possible number of runs and its probability of being the result in the game gives the expected number of runs for the batting order considered.

This strategy is the same in philosophy as that of Bukiet et al. (1997) only the details are modified. The strategy involves mathematically only addition, multiplication and some logic. The method makes sense only because cricket has the following properties:

- Dynamics of run production depends mainly on interaction of bowler and batsman so the game can be modelled as a sequence of one-on-one interactions
- There are a finite number of states in the game (batsman, outs, runs)
- The probability of an occurrence depends only on current situation (to a reasonable approximation) (One can also implement run or score dependence).

Some aspects of One-Day Cricket make it more complicated and lengthy to model than for baseball.

- There are 11 players who bat on a team in an innings. Thus there are over 39 million batting orders to consider in a full enumeration.
- Up to 300 balls are bowled in an innings, in groups of 6 (an over). After each over has been completed, a different bowler bowls the ball from the opposite side of the field. (In baseball one can consider the situations as batter by batter and there are only about 50 batters up in a game for each team).
- An odd number of runs scored on a ball results in the other active batsman batting next (unless this is the end of an over).
- Bowlers switch sides at the end of an over and can only bowl a maximum of 10 overs in a match. This makes some of the logic and bookkeeping more complicated.
- Typical one-day cricket matches result in each team scoring 200-300 runs (much more than the 0-10 runs common in baseball, with extreme cases running up to 20 runs for a team). In cricket, in theory a team could get up to 1800 runs, although the present record is 438 runs (*ODI #2349 South Africa v Australia at New Wanderers Stadium, Johannesburg on 12 March 2006, South Africa scored 9/438 in reply to Australia's 4/434. Previous record was 5/398 scored by Sri Lanka in ODI #1074 Sri Lanka v Kenya at Asgiriya Stadium, Kandy during the 1995/96 Wills World Cup. Source: CricInfo*).
- This increases run time and storage requirements.
- P_d is very small for most batsmen, typically about 0.03 or less, which means outs are very rare when compared to baseball.
- By comparison to baseball, players play in fewer games per year. As a result, small errors

can have a larger effect on an individual player's probabilities.

- When the innings is nearing its end, batsmen begin taking more risks and score at a higher rate. Players who come in to bat with very few overs remaining will usually score at a higher rate than they normally would if they came in earlier in the innings. Only the first batsman has a non-zero probability of facing all 300 balls in a match. This could potentially skew the probability distribution (P_0-P_6) obtained from the data set for a later order batsman.
- For one lineup a simply written code for 1-day cricket takes ~15 seconds on an average PC (up to 600 runs considered). To evaluate 11! ~ 40,000,000 line-ups such a code would take about 1000 days.

The large computational time involved using this straightforward approach (referred to later on as "the straightforward method" makes it unattractive as a planning tool for coaches, however some streamlining improves the performance. Instead of considering each batsman and each ball individually, we consider (in what we call our "streamlined method") each pair of batsmen (11 x 10 pairs) and each over individually (50 overs). As a further simplification, we assume that a maximum of 1 wicket can fall in any given over; the result is about a 1 second improvement in processing time per line-up studied.

To include bowling and defensive performance, one could scale the offensive characteristics in an appropriate way. For example, if a given bowler has performance level, say, 2% worse, than the average bowler, by some measure, then opposing players would have their offensive performance (P_1-P_6) increased by 2% and P_0 decreased accordingly. Ideally, one would like to have enough data on how well each bowler performs against each batsman (and vice versa), but that is not likely to be the case. Another method of scaling batsman performance might take into account his "handedness", that of the bowler, and/or the type of bowler (e.g., a spin bowler) bowling. One of the authors has looked into various methods of considering pitcher ability in baseball and found that considering such complications did not lead to improved results.

RESULTS

Using data gathered from the CricInfo website and that kindly supplied by Champion Data, we were able to find estimates (for various players) of the probability of scoring 0, 1, 2, 3, 4 or 6 runs. Treating being not out at the end of a game as the same as being run out on the last ball, we can find an estimate for the probability of being dismissed. Table 1 shows these estimates for 11 Australian players. These players have enough experience such that the data used takes into account at least 100 balls being bowled to each of the players and each player's performance in at least 15 matches (except for Michael Slater and Glen McGrath as shown in Table 2). The data collected did not have many matches including Michael Slater, although he has played 42 matches for Australia. Glen McGrath normally is the last batsman in the Australian batting lineup. This limits the amount of data available on his performance.

The probabilities P_0-P_6 above sum to approximately 1 and are the distribution of 0-6 runs scored by the players whilst not being dismissed. The probability of dismissal, P_d, is calculated as the number of innings played divided by the number of balls faced. Whilst this does not account for innings in which the player does not get dismissed, this error in the data is small when a player has played many innings and unlikely to be as large as the error caused by using small data sets (e.g. a single innings). The data from Table 1 can also be used to determine the expected runs if the team were made up of only 1 player occupying all 11 places, which

Table 1. Estimated player probabilities.

P_0	P_1	P_2	P_3	P_4	P_6	P_d	Player
.513	.311	.071	.006	.083	.012	.027	Symonds
.561	.230	.055	.013	.128	.011	.027	Gilchrist
.546	.297	.067	.014	.070	.004	.019	Waugh, M.
.563	.273	.058	.012	.080	.011	.021	Hayden
.545	.295	.059	.012	.072	.013	.022	Ponting
.512	.294	.115	.000	.038	.038	.025	Slater
.533	.319	.060	.004	.060	.020	.051	Bichel
.526	.332	.071	.005	.026	.032	.083	Lee
.515	.387	.054	.018	.018	.006	.109	Gillespie
.687	.234	.046	.000	.031	.000	.250	McGrath
.555	.324	.072	.003	.039	.004	.051	Warne

Table 2. Expected runs and player rankings.

Player	Runs per innings	Rank	Balls	Innings
Symonds	244.04	3	2096	57
Gilchrist	263.89	1	4203	117
Waugh, M	202.15	5	2477	49
Hayden	216.39	4	3401	73
Ponting	177.64	6	5056	112
Slater	254.57	2	78	2
Bichel	148.66	7	444	23
Lee	61.31	9	334	28
Gillespie	53.95	10	165	18
McGrath	13.60	11	64	16
Warne	120.13	8	635	33

then enables us to rank the players. This ranking, computed when limiting runs to 600, is shown in Table 2.

Using this ranking, we compute the expected runs using the "straightforward method" when the players are ordered in Best-Worst ranked order (Gilchrist bats first, Slater second, Symonds third and so forth), the given order and Worst-Best ranked order. These expectations are shown in Table 3.

Table 3. Expected runs

Best-Worst Ranked Order	210.66 runs
Given Order	208.75 runs
Worst-Best Ranked Order	194.16 runs

To enable us to consider a greater number of batting orders, we used our "streamlined method" to evaluate about 10,000 line-ups, permuting only the last 8 players. We find that the minimum number of expected runs is approximately 219 compared with a maximum of almost 229. These results are higher than those achieved before the streamlining and are most likely due to the restriction on the number of wickets than can be lost per over. Table 4 shows a comparison of results achieved using the straightforward and streamlined methods. The line-up used in generating tables 1-4 uses players (like Mark Waugh) whom no longer play for Australia, but each player has played at least 10 games at the international level. Suppose, however, that we wish to replace 4 players with current or new players, like Hussey and Hodge. Table 5 shows the estimated probability distributions using the same source data. (Here, each player's data only includes at least 2 games and 50 balls bowled to him, to allow for newer players, as shown in Table 6 except for Stuart MacGill). Stuart MacGill normally bats after Glen McGrath when both are playing for Australia. This also limits the amount of data available on his performance.

Table 4. Expected runs using both methods.

Player	Runs/Inn Straightforward	Runs/Inn Streamlined
Symonds	244.04	249.29
Gilchrist	263.89	269.91
Waugh, M	202.15	203.06
Hayden	216.39	217.92
Ponting	177.64	178.88
Slater	254.57	258.88
Bichel	148.66	165.98
Lee	61.31	69.63
Gillespie	53.95	68.96
McGrath	13.60	23.68
Warne	120.13	134.19

Table 5. Estimated player probabilities

P_0	P_1	P_2	P_3	P_4	P_6	P_d	Name
.553	.313	.048	.010	.075	.003	.015	JL Langer
.563	.273	.059	.012	.080	.011	.021	ML Hayden
.545	.295	.060	.012	.073	.013	.022	RT Ponting
.412	.471	.039	.039	.039	.000	.039	BJ Hodge
.522	.317	.064	.016	.074	.006	.022	MEK Hussey
.514	.312	.072	.007	.083	.013	.027	A Symonds
.561	.231	.055	.014	.128	.011	.028	AC Gilchrist
.556	.324	.072	.003	.039	.005	.052	SK Warne
.527	.332	.072	.006	.027	.033	.084	B Lee
.714	.286	.000	.000	.000	.000	.286	SCG MacGill
.688	.234	.047	.000	.031	.000	.250	GD McGrath

Table 6. Expected runs and player rankings

Name	Exp_Runs	Rank	Balls	Innings
AC Gilchrist	281.34	1	4203	117
A Symonds	259.20	2	2096	57
MEK Hussey	242.79	3	312	7
RT Ponting	241.75	4	5056	112
ML Hayden	240.17	5	3401	73
BJ Hodge	239.58	6	51	2
B Lee	232.00	7	334	28
JL Langer	222.43	8	400	6
SK Warne	192.78	9	635	33
GD McGrath	128.37	10	64	16
SCG MacGill	80.06	11	7	2

Table 6 shows, like Table 2, the expected runs if these players made up the entire line-up (using the streamlined method with a maximum of 1800 runs allowed).

Using this data we compute the expected runs from a convenient subset (163,724 samples at this time) of batting line-ups, namely, those line-ups with the openers and some with the third batsman already determined. Table 7 shows a brief analysis of the data. It is interesting to note that the mean is almost exactly 235, which is also the current value of the G50 constant in the Duckworth/Lewis method for target resetting. Figure 1 shows a histogram of these results. We note that the distribution is unimodal, has small variance, but is highly skewed with large number of outliers. The line-ups that produced the minimum and maximum number of expected runs (among the 163,724 lineups studied) are shown in Table 8.

Table 7. Descriptive statistics

Mean	235.1
Sample Variance	95.2
Mode	209.8
Count	163724
Minimum	187.6
First Quartile	234.1
Median	236.9
Third Quartile	239. 5
Maximum	257.6
Range	70.0
IQR	5.4

DISCUSSION

Cricket is a game with many variables affecting the outcome; the weather, the pitch, the players and even the spectators at the match. Attempting to

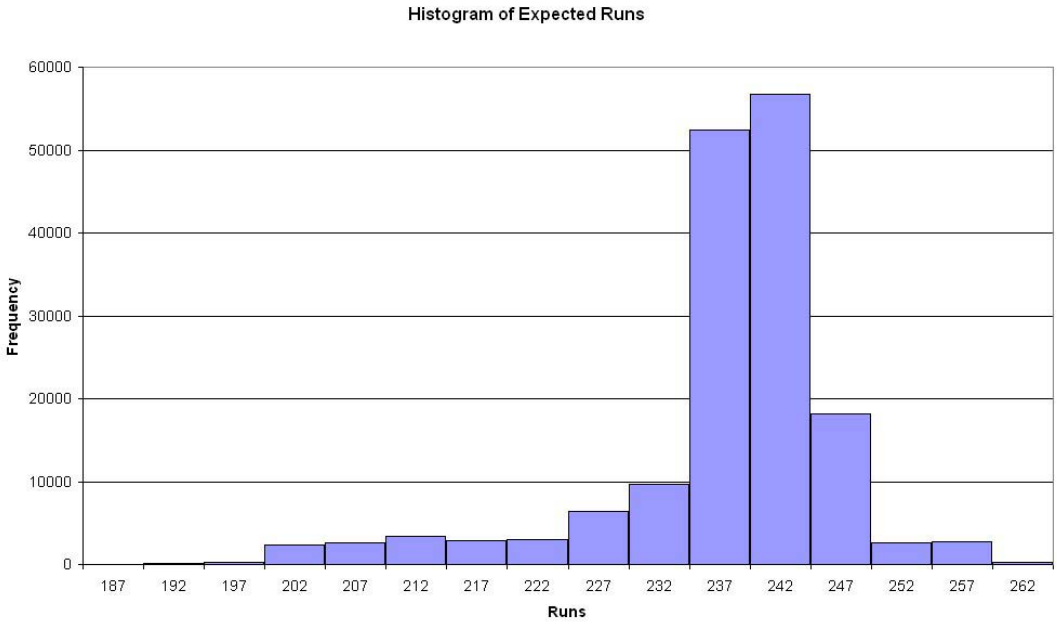

Figure 1. Histogram of expected number of runs for 164,724 line-ups taken from the players listed in Table 6.

model a cricket match requires reducing these many variables down to a manageable and quantifiable subset. In this paper we have attempted to simplify these many variables down to the batsman's average offensive ability versus every bowler they have faced (within the available data set). We have developed a straightforward and a streamlined approach for evaluating the distribution and thus, the expected number of runs a line-up should produce against average bowling. The large number of possible line-ups makes a "straightforward" approach to finding the optimal batting order virtually impossible to achieve in a reasonable time frame, however smaller subsets could be calculated rather quickly. This means that the most practical use of our work would be in determining the order of three or four batsmen with the rest of the line-up fixed. Our work could also be used to quantify the effect of the "super-sub" under the new laws of the One-Day game.

Table 8. Minimum/maximum run batting orders

Batsman	Minimum	Maximum
1	JL Langer	JL Langer
2	ML Hayden	ML Hayden
3	BJ Hodge	A Symonds
4	GD McGrath	AC Gilchrist
5	SK Warne	SK Warne
6	SCG MacGill	MEK Hussey
7	AC Gilchrist	B Lee
8	B Lee	SCG MacGill
9	MEK Hussey	GD McGrath
10	RT Ponting	RT Ponting
11	A Symonds	BJ Hodge

We find it interesting that the results of our model show a mean expected number of runs scored of almost exactly 235. Since we were not able to study all 40 million line-ups, we have shown for the set of players considered that the best batting order can expect to produce at least 70 runs more than the worst possible line-up. Figure 1 suggests that it is easier to find a very poor line-up than it is to find a very good one. We expect the result of allowing for slight variations in player ability will have a similar effect to such variations in baseball player ability as studied by Sokol (2003). That is, that while our technique will find (given enough computation time) the line-up with the greatest expected number of runs, slight variations in player performance ability would result in a different line-up being "better". In other words, the best line-up is not robust. However, any of a set of nearly optimal line-ups would be indistinguishable within the limit of accuracy of the input probabilities.

CONCLUSIONS

Whilst we have done a large number of calculations and streamlined the code, we see that it is unlikely that the calculations could be finished (on a single computer) within a reasonable timeframe. However, we see other opportunities for future work and applications, for example: 1) We could study optimal batting order and impact of batting order on probability of winning a game. Although we may not perform a full enumeration, there are ideas which would allow us to study likely subsets (e.g. Swartz); 2) The data collection in One-Day Cricket is more difficult than baseball. We would like to obtain more data and more team information, but perhaps we could investigate ways of interpolating using available data; 3) We could expand our model to include bowling ability (defence) as was done for baseball (e.g. Bukiet, 1997); 4) A further extension of the model would be to compute the probability of winning a game and the effect of using one player instead of another (e.g. the "rotation" policy effect).

ACKNOWLEDGEMENTS

The authors would like to thank Champion Data and Mr. David McKenzie for their assistance in gathering the data needed. Additionally, the authors would like to thank the anonymous reviewer for their insightful and helpful comments.

REFERENCES

Bukiet, B., Harold, E. and Palacios, J. (1997) A Markov chain approach to baseball. *Operations Research,* **45**, 14-23.

Clarke, S.R. (1988) Dynamic programming in One-Day Cricket - optimal scoring rates. *Journal of Operational Research Society* **39**, 331-337.

Cohen, G.L. (2002) Cricketing chances. In: *Proceedings of the 6th Australasian Conference on Mathematics and Computers in Sport*. Eds: Cohen, G. and Langtry, T. Bond University, Queensland, Australia. 1-13.

Elderton, W.E. (1945) Cricket scores and some skew correlation distribution. *Journal of the Royal Statistical Society Series A* **108,** 1-11.

Johnston, M.I., Clarke, S.R. and Noble, D.H. (1993) Assessing player performance in One Day Cricket using dynamic programming *Asia-Pacific Journal of Operational Research* **10,** 45-55.

Norman, J. and Clarke, S.R. (2004) Dynamic programming in cricket: batting on a sticky wicket. In: *Proceedings of the 7th Australasian Conference on Mathematics and Computers in Sport.* Eds: Morton, H. and Ganesalingam, S. Massey

University, Palmerston North, New Zealand. 226-232.

Sokol, J. (2003) A robust heuristic for batting order optimization under uncertainty. *Journal of Heuristics* **9,** 353-370.

Swartz, T.B., Gill, P.S., Beaudoin, D. and de Silva, B. M. (2006) Optimal batting orders in One-Day Cricket. *Computers and Operations Research* **33,** 1939-1950.

Wood, G.H. (1945) Cricket scores and geometric progression. *Journal of the Royal Statistical Society Series A* **108,** 12-22.

KEY POINTS

- Batting order does effect the expected runs distribution in one-day cricket.
- One-day cricket has fewer data points than baseball, thus extreme values have greater effect on estimated probabilities.
- Dismissals rare and probabilities very small by comparison to baseball.
- Probability distribution for lower order batsmen is potentially skewed due to increased risk taking.
- Full enumeration of all possible line-ups is impractical using a single average computer.

AUTHORS BIOGRAPHY

Bruce BUKIET
Employment
Assoc. Professor of Mathematical Sciences and Associate Dean of the College of Science and Liberal Arts at NJIT.
Degree
PhD
Research interests
Mathematical modelling of biological systems, detonation dynamics and sports.
E-mail: bukiet@m.njit.edu

Matthew OVENS
Employment
LMS Training Officer, Monash University
Degree
B.Sc(Hons)
Research interests
Mathematics in cricket, mathematics education and online learning management systems.
E-mail:
matthew.ovens@its.monash.edu.au

✉ **Matthew Ovens**
Flexible Learning and Teaching Program, Information Technology Services Division, Monash University, Australia.