OPTIMALLY TUNED MARKOV CHAIN SIMULATIONS OF BATTLES FOR REAL TIME DECISION MAKING

Russell Cheng

James Moffat

University of Southampton
Highfield
Southampton, SO17 1BJ, UNITED KINGDOM

DSTL
Portsdown West, Portsdown Hill Rd
Fareham, PO17 6AD, UNITED_KINGDOM

ABSTRACT

We show how a Markov chain provides a simple representation of the underlying character of a Blue versus Red battle engagement. Fixed time-step simulation provides a natural practical implementation of such a representation. We demonstrate how such an implementation can be optimally tuned to model and capture the most important aspects of a given battle whilst still enabling simulations to be carried out sufficiently fast to be useful in a real-time context. Thus such an approach could potentially be used by field commanders as an aid in real-time battlefield decision making. A realistic example is provided based on a real tactical conflict drawn from recent history.

1   INTRODUCTION

This report describes work carried out for the Defence Science and Technology Laboratory (Dstl) investigating simple stochastic combat simulation models. Dstl already has such model called SIMBAT which is capable of modelling Blue versus Red combat engagements in quite realistic detail (see for example Moffat (2011, Chapter 6). The aim of the work described here is to see if the main characteristics of engagements displayed in simulation runs of the SIMBAT model can be reproduced from runs of a model based on a simpler Markov chain representation, but which retains the main features of the SIMBAT model. We shall call these simpler models *Markov Chain SIMBAT-Like* (MCSL) models. These MCSL models are discussed in a series of reports produced for Dstl, namely Cheng (2009a,b,c), Cheng (2011) and build on earlier joint work with Dstl. The material in this present article is in part a summary of these earlier reports.

   Our main conclusion is that runs of such MCSL models are capable of producing results very similar to those obtained from runs of the SIMBAT model, but with run times that are about 40 times faster. Thus simulations of combats that might take minutes in SIMBAT only take seconds in MCSL.

   Such simpler models might possibly be used in real-time battlefield conditions for carrying out risk assessment by allowing multiple runs to be carried out to assess the likelihood of success of a proposed plan of action. The rapid evaluation also makes them potentially suitable as fitness functions for optimal search algorithms such as genetic algorithms.

2   BLUE V RED COMBATS

Both SIMBAT and MCSL models represent a Blue versus Red battle in essentially the same way using a *fixed time step* simulation. The Blue and Red forces each comprise a number of units, m_B and m_R say, and at the start of the battle each unit comprises a given number of similar *components*. Components are the smallest element of a force considered, each comprising a given number of personnel and weapon assets. The *strength* of a unit is measured by the number of active components. Each unit has a designated *initial*

position, and the task of each unit is to move along its own *designated path* and to reach a specific *destination* by a given time.

The way the simulation is organised is illustrated in Figure 1. At each time step and for each unit, a *sighting acquisitions algorithm* calculates the unit's hostile unit acquisitions (in effect which hostile units the unit can see). This then enables a *perceived combat power ratio* (PCPR) to be calculated by each unit which measures its strength relative to the hostile units it has acquired. A unit encountering hostile units will try to engage these in combat. Two types of combat are considered: (i) a *mini-battle* in which a unit exchanges fire at a distance against acquired hostile opponents, (ii) a unit may attempt to engage an opposing unit in *close combat* if it is sufficiently close. A unit's losses are calculated in terms of the number of its components knocked out (KO'd), and its strength is then updated. An active unit also has a *moving state* and a *defensive state*. There are three moving states: advancing, halted, withdrawing. When halted a unit takes on one of a number of different defensive states. The moving and defensive states of each unit, which depend on its PCPR, are also updated. Finally the positions of each unit are updated.

The above calculations are repeated at each time step until the simulation run end time is reached.
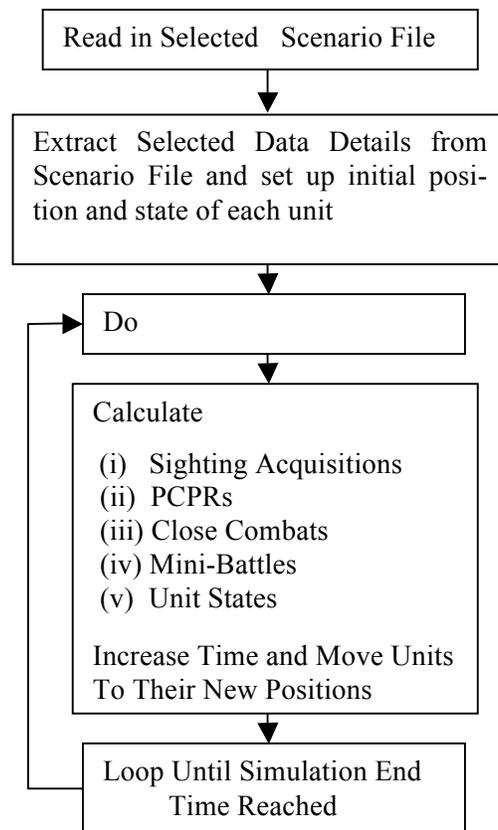
Figure 1: Structure of SIMBAT and MCSL models

The sighting and combat calculations involve probabilistic elements, reflecting uncertainty in acquisition of hostile units, and in damage inflicted during combat, so that if simulation runs are replicated with the *same initial conditions*, the battle will result in *different* outcomes. In risk analysis, it is the extent of this variation in the outcome and its measurement that is of principal interest.

In the next section we discuss how a Markov chain provides a natural stochastic model for representing the behavior of such a fixed-time step simulation.

3   MARKOV CHAIN MODEL

We now show that the fixed time step model described in the previous section can be viewed a Markov chain.

It will be clear from the form of both SIMBAT and MCSL models, that the overall state of the battle at any given time point, is completely determined by the overall state of each unit that is active at that time point, where the *overall state of a unit* comprises its position (on its designated path), strength, and moving and defensive states. Let $s_{iB}(t)$ be the overall state of the *i*th Blue unit at time *t* and $s_{jR}(t)$ that of the *j*th Red unit at time *t*. The overall state of the battle as a whole can then be written as $S(t) = (s_{iB}(t), \; i = 1, 2, ..., m_B; s_{jR}(t), j = 1, 2, ..., m_R)$.

$S(t)$ is a high multidimensional vector. For simplicity we shall assume that all the components of the vector specifying $S(t)$ are discretized. This includes each path along which a unit moves which is discretized into a fixed number of *waypoints* including the unit's initial and final-objective positions. Thus the totality of all possible different battle-states, though massive, is finite, and can be index by *i* = 1,2,…, *N*. In this discretized form we can simply use *i* to indicate a particular given state so that $S(t) = i$ means that at time point *t* the system is in state *i*. We also assume time is discretized into fixed equally spaced time points with index *t* = 0,1,2,…,*T*. An entire battle is thus fully described by the state at successive time points: $S(0)$, $S(1)$,…, $S(T)$, where $S(0)$ is the initial given state. The system changes randomly, reflecting uncertainty in sighting hostile units and in combat casualties inflicted. Succeeding states are therefore random variables, which can be viewed as drawn from a probability distribution that changes with time.

The battle just described, viewed as a stochastic process *S*, is a (homogeneous) *Markov chain* if the probability distribution of $S(t+1)$ depends only on the state $S(t) = i$ immediately preceding it, and not on *t* or any other previous states. The conditional probabilities

$$p_{ij} = \Pr(S(t+1) = j \mid S(t) = i) \quad\quad i,j = 1,2,..., N \tag{1}$$

are then all independent of *t* and give the probability of moving from any given state *i* of *S* to any other given state *j* of *S* in one time step. The p_{ij} are called the *single step transition probabilities*.

A simulation in effect calculates a *realization* of the battle yielding the overall state, $s(0)$, $s(1)$, $s(2)$, …, $s(T)$, of the battle at successive time points, with the state $s((t+1))$ calculated solely from the state $s(t)$ at the previous time point t. A lower case *s* is used here to indicate specific state values obtained in the realization. In terms of risk analysis we are interested not just in one realization, but in the *probability distribution*

$$\pi(t) = (\pi_1(t), \pi_2(t), ..., \pi_N(t)) \tag{2}$$

giving the probabilities of being in each of the states {1,2,…,*N*} at time *t*. In the Markov chain model this can in theory be studied via the *N×N single-step transition* matrix

$$P = (p_{ij}) \tag{3}$$

where the p_{ij} are as defined in (1); see Stewart (1994) for example. Specifically the matrix (3) can be used to calculate (2) recursively from the Chapman Kolmogorov (CK) equations :

$$\pi(0) \text{ given}, \quad \pi(t) = \pi(t-1)P, \quad t = 1,2,...,T \tag{4}$$

where $\pi(0)$ is the initial state distribution. Typically we are in a given fixed state $S(0) = j$ initially so that

$$\pi(0) = (0, 0, ... 0, 1, 0, ..., 0) \text{ with the 1 in position } j.$$

The probability distribution of the final outcome of the battle is given by

$$\pi = \lim_{t \to \infty} \pi(t). \tag{5}$$

This has an obvious form which we will discuss shortly.

A complete calculation of $\pi(t)$ or π is prevented by the *problem of dimensionality* which occurs because the state-size, N, grows geometrically with the number of units/paths. For example, assume there are just n_i distinct points on the path along which unit i can move, and for simplicity that each unit comprises just one component and that the state of unit i is simply its present position z_i ($z_i = 1,2, ..., n_i$) on its path if it is active, or $z_i = 0$ if it is KO'd. Thus unit i can be in one of just (n_i+1) states. The total number of possible states is therefore

$$N = \prod_{i=1}^{m_B + m_R} (n_i + 1).$$

This number grows intractably large even for 'toy' situations. For example, suppose that there are just 3 Blue units and 2 Red units, and that their respective paths comprise just 3, 3, 4, 3, 3 waypoints. Then $N = 4 \times 4 \times 5 \times 4 \times 4 = 1280$. The P matrix thus has $1280^2 = 1,638,400$ entries.

In a more realistic (but still small) engagement with say 20 units (10 on each side) and say 15 waypoints on each path, $N = (15+1)^{20}$. This is clearly intractably large for full analytic study.

There is one mitigating factor. The matrix P is sparse, with most of the entries p_{ij} being zero. Thus sparse matrix methods can be used to study examples which would otherwise be intractable. In the example considered above where the P matrix has 1,638,400 entries, only 138,873 entries (less than 10%) are non-zero.

Even allowing for sparseness, the problem of dimensionality means that we would not even be able to *store* let alone calculate P in realistic examples. Analysing P in detail is an even more remote possibility.

The general form of the matrix P is clear however. It will correspond to a *terminating* Markov chain where each state will either be *transient* or *absorbing*, but not recurrent. The absorbing states are of most interest as they correspond to the different final outcomes of the battle when one or other force has 'won'. With suitable ordering of the states the P matrix can be written in upper Hessenberg form with all the positive entries either in a set of square submatrices on the main diagonal or above these. Figure 2 is the P matrix for a two blue and one red unit battle with each designated path of each unit made up of just three way points including initial and target end points, so that $N = (3+1)^3 = 64$.
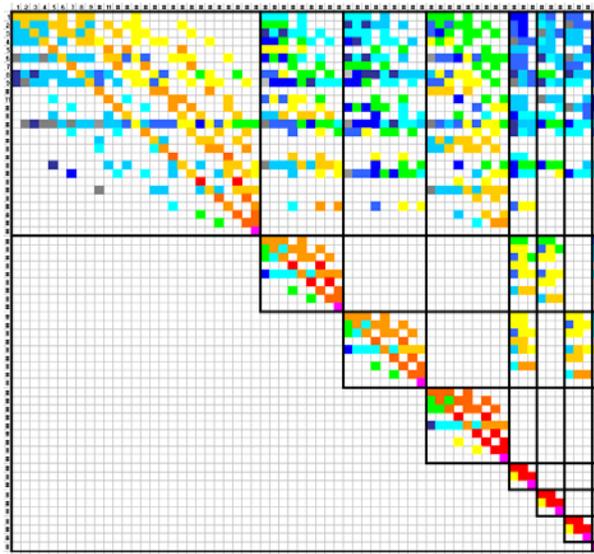
Figure 2: Upper Hessenberg form of P matrix in a 2 Blue 1 Red unit battle

Simple sighting and firing routines are not described here, but details are given in Cheng (2009a), These were used to calculate the results of combat between opposing units. The coloured squares indicate entries where p_{ij} is positive, with warmer colors, in the order given in Figure 3, corresponding to larger values. It is readily verified that all the states are transient except for the states corresponding to the last rows in each of the 8 square submatrices on the main diagonal. These 8 states comprise all the absorbing states, each of which represents a different final position where each of the four units has separately either reached its final target position or has been KO'd.

Figure 3: Colour coded scale giving transition probability magnitudes, increasing left to right

4 RISK ANALYSIS

4.1 State Probabilities Calculated from the CK Equations

The main benefit of being able to calculate the limiting state probability distribution π as given in (5) is that it includes the probabilities of all the different possible final outcomes of the battle under consideration, A simple way of obtaining π, at least approximately, is from the CK equations (4) using a large T. This is only computationally tractable in unrealistic small examples. Even so it is of interest to consider such an example if only to illustrate the form that the solution takes.

Figure 4 gives a visual impression of the form of $\pi(t)$ obtained from solving the CK equations (4) for a one Blue versus two Red units engagement with just 4 waypoints on each designated path, including the initial and final endpoints, so that $N = (4+1)^3 = 125$. In the main block, the probability distributions $\pi(t)$ are set out in columns and for reasons of space only the time points $t = 0,1,2,...15$ are depicted. The distribution for the final time point $T = 25$ is displayed in the three left-most columns where the individual unit states that apply in each row are listed. The gradual and systematic convergence and concentration of the probabilities to the absorbing states is typical.

The probabilities of having reached the 8 absorbing states by time step $T = 25$ are given in Figure 5. The overall probability of 0.9989 gives the probability that one of the 8 absorbing states will have been reached by time step 25. This shows that the battle will be over with this high probability by time step 25, as the battle is over once any one of these absorbing states is reached.

The most likely result is a win for Blue with probability 0.3549, with the single Blue unit having reached its destination way point (#4), and both Red units KO'd. The second most likely result is a win for Red, with probability 0.2061 that both Red units survive and reach their target destinations (both #4), and Blue KO'd. The relatively high probability of Blue knocking out both Red units comes about because of a higher KO capability used for Blue (twice that of a Red unit) in the combat calculation.

4.2 State Probabilities Estimated from Simulation Runs of the Markov Chain

The limiting distribution π can be estimated without having to calculate the full P matrix simply by making Monte Carlo simulations of the Markov chain. Such a simulation in effect mimics the more detailed runtime calculations of the SIMBAT or MCSL model. Figure 6 depicts the results of 5000 independent Monte Carlo simulation runs of the 1 Blue 2 Red units example each with the initial state $S(0) = (1 \mid 1\ 1)$, where the 1's denote the fact that the Blue unit and two Red units are all at the initial point of their designated paths at the start of the run. The final state, when $T = 25$ time steps in our example, is recorded. Let n_i be the number of times state i is the final state in the N runs, so that

$$\sum_i n_i = N.$$

Then an estimate of the probability that i is the final outcome is

$$\hat{\pi}_i = \frac{n_i}{N}. \tag{6}$$
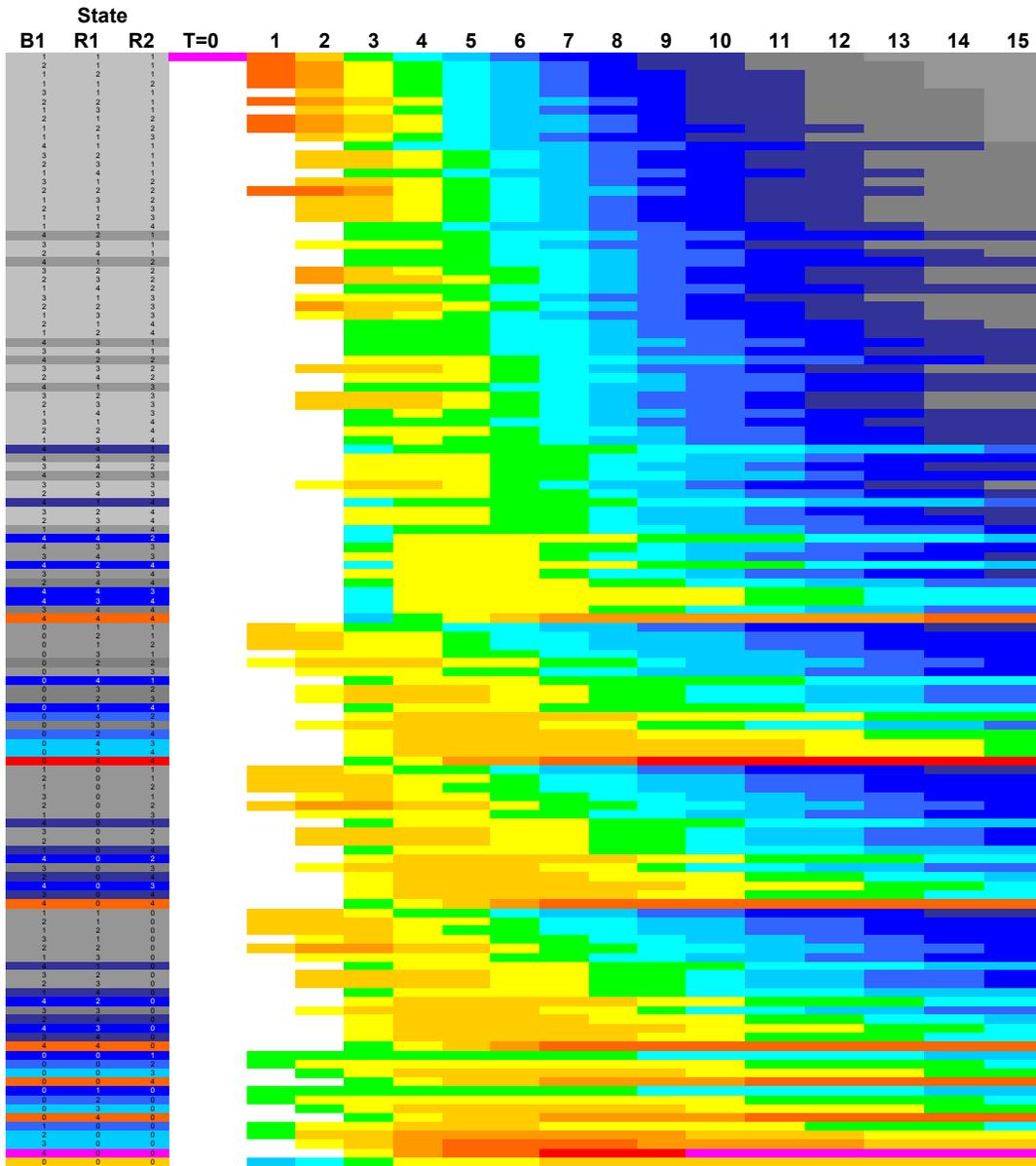
Figure 4: State probability distributions $\pi(t)$, $t = 0,1,..,15$, in the 1 Blue 2 Red example. Calculated from the CK equations and displayed in columns

Blue1	Red1	Red2	Prob
4	4	4	0.0595
0	4	4	0.2061
4	0	4	0.1081
4	4	0	0.1081
0	0	4	0.0751
0	4	0	0.0751
4	0	0	0.3549
0	0	0	0.0119
	Total Prob		0.9989

Figure 5: CK probabilities at *T* = 25 of the 8 absorbing states for the 1 Blue 2 Red example
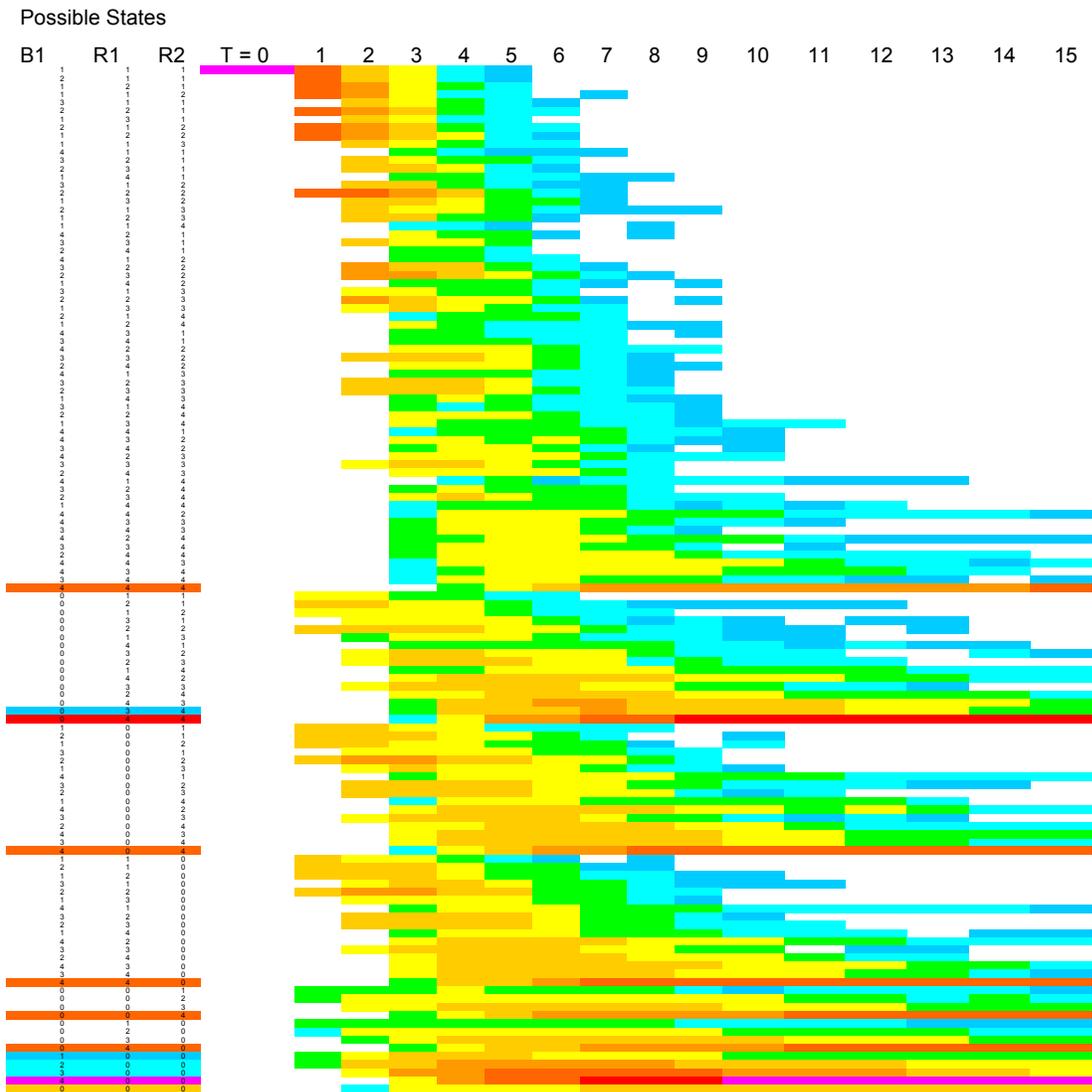
Figure 6: State probability distributions $\pi(t)$, *t* = 0,1,..,15, in the 1 Blue 2 Red example. Calculated from 5000 simulation runs of the Markov chain and displayed in columns

A comparison of Figures 4 and 6 reveals a broad agreement in the way the state probability distribution changes with time.

Figure 7 shows the state probability distribution at time step 25 for the 8 absorbing states as obtained from equation (6). The Blue and Red columns denote final unit positions. It will be seen that the probabilities are very similar to those obtained from the CK calculations given in Figure 5.

Blue1	Red1	Red2	Prob
4	4	4	0.0588
0	4	4	0.2042
4	0	4	0.1062
4	4	0	0.1068
0	0	4	0.0760
0	4	0	0.0750
4	0	0	0.3596
0	0	0	0.0122
		Total Prob	0.9989

Figure 7: Simulation probabilities at *T* = 25 of the 8 absorbing states for the 1 Blue 2 Red example

The calculations required in the Markov chain simulation approach do not require storage of the elements of the *P*. During a simulation run probabilities needed to decide the next transition are calculated in only as and when they are required. The simulation approach is thus computationally much less intensive than the CK equation approach.

The remainder of the paper considers the simulation based MCSL approach in more detail.

5   THE MCSL MODEL

Figure 1 shows how SIMBAT and MCSL models are both organised. The main difference between the two models is that at various points the MCSL model replaces the more detailed calculations of the SIMBAT model by an *adjustment factor*. Seven adjustment factors are currently employed in MCSL. Each operates in a similar way as a multiplicative factor, typically with a value in the range (0,1), that is used to scale down the effect of the quantity on which it operates. The seven factors are: Speed, Fire Rate, Single Shot Kill Probability, Line of Sight, Time of Day, Indirect Fire Kill Probability, Indirect Fire Obscuration and Suppression. These adjustment factors provide significant flexibility in the possible behaviour of the MCSL model.

These factors are fully discussed in Cheng (2011) and details will not be repeated here. However we do describe just one to illustrate the approach used for all of them. In SIMBAT the Sighting Algorithm is very detailed. It calculates observability using a nonlinear viewing distance measure which also takes account of terrain unevenness and the availability of observers with appropriate equipment to scan different sectors of the overall horizontal visual arc. MCSL also uses a nonlinear viewing distance measure but this is fixed except for multiplication by the line of sight adjustment factor to allow for the combined effect of all other viewing aspects.

The present method of fixing the values of the factors is to compare the output from MCSL simulation runs for a given battle scenario with the output from corresponding runs of the SIMBAT model using a *discrepancy measure*. The problem is treated as one of search optimization where the objective is to find that combination of adjustment factor values which minimizes this discrepancy measure. The discrepancy measure used is most easily explained after viewing some actual results obtained from an example.

We have considered a real tactical conflict drawn from recent history for comparison with our approach. Figure 8 depicts the initial position of the 25 Blue units and 26 Red units making up the two forces. The simulations were of 20 hours of real time over which the battle took place, with a time-step of 90 seconds.

Map for Example showing Initial Positions of Blue and Red Units
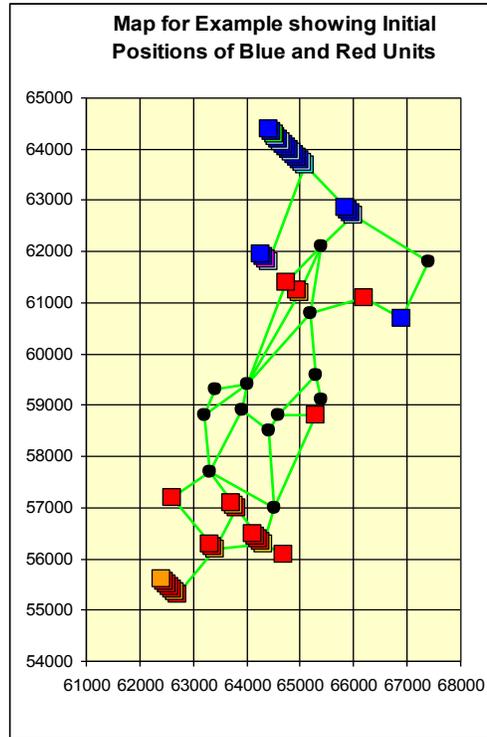
Figure 8: Scenario of the battle example showing the waypoints and initial positions of the 25 Blue and 26 Red units

We describe the initial experimentation which was focused on finding values for the seven adjustment factors in the MCSL model which would produce MCSL simulation run results that give a good match to those of SIMBAT runs. A set of 50 runs of the SIMBAT model was therefore made of the given real conflict scenario with which subsequent runs of the MCSL model could be compared. At the end of each run of either model three *final outputs* were recorded for each unit in each force: (i) the final unit strength as measured by the number of components of the unit that are active, (ii) the total amount of ammunition used by all the components of the unit, and (iii) the distance travelled by the unit along its designated path. Each of these outputs was measured on the integer scale of 0 through 8. The 50 SIMBAT runs produced 50 observations of each of the three output values for each unit. We consider just one of the final outputs, the final strength. Figure 9 includes histograms showing the frequency distribution of final strength for each of the units. The Figure also shows the corresponding histograms from 50 runs of the MCSL model using a particular combination of adjustment factor level settings obtained by an elementary search procedure. It will be seen that the combination used is good in that the final strength histograms obtained from the MCSL model runs are quite well matched to the histograms obtained from the SIMBAT runs across all units.

Corresponding results for the other two final outputs are not discussed here but were similar to the strength output results in that the histograms for these other two outputs obtained from MCSL runs matched those from the SIMBAT runs quite well across all units.

The search procedure used to identify a good combination of adjustment factor values was the elementary one of varying one, or at most two, factors at a time. Each step of the search involved a comparison of 50 observations of each of the three final output values for each unit obtained from the 50 MCSL runs with the corresponding 50 observations obtained from the SIMBAT runs. The comparison was made using a *difference measure* so that a good combination of adjustment factor values corresponded to a small value of this difference measure. We discuss the form of this measure in the next section.

Optimization procedures based on more elaborate and formal statistical design of experiment methodology were also tried but proved rather unsuccessful. From pilot studies it was clear that this was due to the highly irregular effect of adjustment factors. For example, one could not assume a convex quadratic response in order to identify a minimum of the difference measure.

MCSL Blue	SIMBAT Blue	MCSL Red	SIMBAT Red
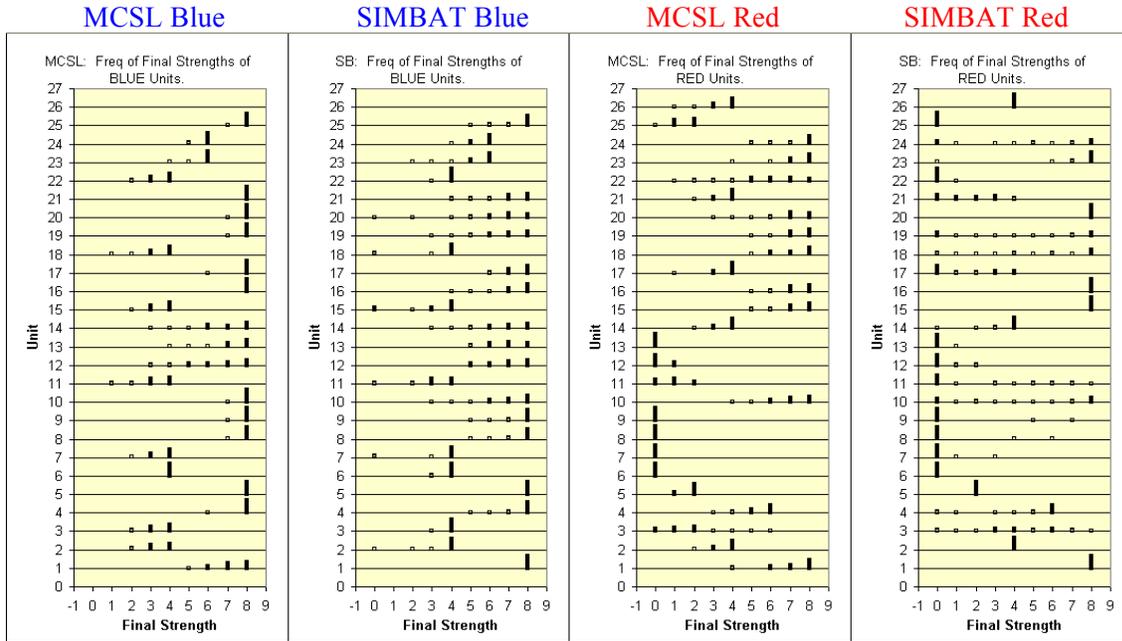
Figure 9. Histograms of unit final strength from 50 MCSL and 50 SIMBAT runs. Adjustment factor settings obtained by random search.

6   CRAMÈR VON MISES DIFFERENCE MEASURE

A good way of measuring the difference between two samples is to compare their *empirical distribution functions* (EDF). The EDF of a sample of final strengths of a given unit, say, $\{M_1, M_2, ... M_{50}\}$ obtained from 50 MCSL runs is

$$\widetilde{F}_M(y) = \frac{\# \ of \ M_i's \leq y}{n}, \text{all } y. \tag{7}$$

The two sample Cramèr-von Mises statistic, see Anderson (1962), is a measure of the difference between two such EDF's. In the situation that we have where the only possible sample values are the integers between 0 and 8 inclusive it can be defined as

$$CvM = K \sum_{y=0}^{8} [\widetilde{F}_M(y) - \widetilde{F}_S(y)]^2 \tag{8}$$

where $\widetilde{F}_M(y)$ is the EDF, (7), of a sample of MCSL observations and $\widetilde{F}_S(y)$ is the EDF of a corresponding sample of SIMBAT observations, and $K = 1/8$ is a constant set to make the maximum value that CvM can take equal to 1.0. A geometric interpretation of the comparison made by CvM is depicted in Figure 10.

The CvM statistic of (8) gauges only the difference between two samples. Taken over all m_B+m_R units of the two forces and all three final outputs: strength, ammunition used and speed, there are $3(m_B+m_R)$ CvM statistics to consider. We therefore used a simple average, denoted by $CvM2$, of all these individual

CvM statistcs as our *overall difference measure* of the difference between the MCSL and SIMBAT outputs.

It is of interest to consider the likely values that *CvM2* can take. Like the individual *CvM* statistics, *CvM*2 must have a value between 0 and 1. However, for the case where one set of EDFs being compared were to comprise EDFs each formed from 50 observations uniformly distributed over the integers from 0 to 8 whilst the other set of EDFs are all formed from 50 observations all precisely equal to 0 - so that there is a big difference in the EDFs being compared - then the *CvM*2 value turns out to be about 0.32 typically. For the case where one set of EDFs comprises EDFs each formed from uniformly distributed samples whilst the other set is formed from observations obtained from 50 SIMBAT runs then the difference in EDFs being compared is still large but less so. The typical *CVM2* value in this case is about 0.22. If both sets of EDFs are formed from uniformly distributed samples so that the samples being compared all come from the same distribution, then the typical CVM2 value is about 0.01.

The *CVM2* value comparing the fairly well matched MCSL and SIMBAT samples of Figures 9 was 0.06.
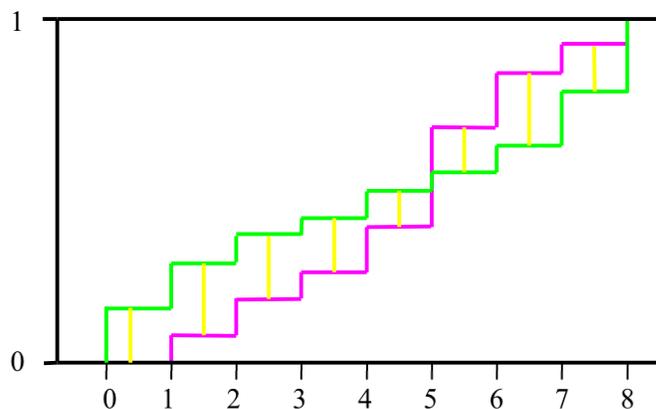
Figure 10: CvM = Sum of squares of yellow heights. This compares EDF1 with EDF2

7   FINAL COMMENTS

The work carried out so far on the MCSL model indicates that it may have the potential to provide fixed time step simulation modelling of realistic Blue versus Red battles that is sufficiently fast and accurate to be used in a practical real-time environment.

However there are still issues that need to be examined in more detail. Foremost is the robustness of the adjustment factor value settings. It seems possible, even using the crude search technique mentioned in this paper, to select these factors settings by comparing the outputs of MCSL runs with those of SIMBAT runs so that the MCSL model behaviour closely matches that of the SIMBAT model. But so far this has only been done for a specific scenario.

More work is needed to see if a MCSL model set up using one scenario can then be reliably used for other scenarios, and if not, to see whether it is possible to identify the conditions that other scenario have to satisfy before a MCSL model set up with a given scenario can be used. Initial further work indicates that the adjustment factors may be fairly robust to such changes in scenario.

REFERENCES

Anderson, T. W. 1962. "On the Distribution of the Two-Sample Cramer-von Mises Criterion." *Annals of Mathematical Statistics* 33: 1148-1159.

Cheng, R.C.H. 2009a. "Stochastic 'SIMBAT-like' Models of Combat as Markov Chains.", Unpublished Report #1, produced for Dstl, March 2009.Cheng, R. C. H. 2009b. "Stochastic 'SIMBAT-like' Models of Combat as Markov Chains." Unpublished Report #2, produced for Dstl, April 2009.

Cheng, R.C.H. 2009c. "Stochastic 'SIMBAT-like' Models of Combat as Markov Chains." Unpublished Report #3, produced for Dstl, June 2009.

Cheng, R. C. H. 2011. "Markov Chain SIMBAT Models" Unpublished Final Report produced for Dstl, Jan 2011.

Moffat J. 2011. *Adapting Modelling and Simulation for Network Enabled Operations* US DoD Command and Control Research Programme, Washington DC, USA.

Stewart, W. J. 1994. *Introduction to the Numerical Solution of Markov Chains*. Princeton: Princeton University Press.

AUTHOR BIOGRAPHIES

RUSSELL C. H. CHENG is Emeritus Professor of Operational Research at the University of Southampton. He has an M.A. and the Diploma in Mathematical Statistics from Cambridge University, England. He obtained his Ph.D. from Bath University. He is a former Chairman of the U.K. Simulation Society, a Fellow of the Royal Statistical Society and Fellow of the Institute of Mathematics and Its Applications. His research interests include: design and analysis of simulation experiments and parametric estimation methods. He was a Joint Editor of the *IMA Journal of Management Mathematics*. His email and web addresses are <R.C.H.Cheng@soton.ac.uk> and <www.personal.soton.ac.uk/rchc> .

JAMES MOFFAT is a Senior Fellow of the Defence Science and Technology Laboratory, UK, a Fellow of the Operational Research Society, a Fellow of the Institute of Mathematics and its Applications and a Fellow of the Royal Aeronautical Society. He is a visiting Professor at Cranfield University and has worked for the past 30 years or so on defence related operational analysis, systems engineering, and aerospace technology research. He has a BSc (Summa Cum Laude) in mathematics, and a PhD in Quantum Operator Theory. His current research interests are in building tools, models and theories which capture the key effects of human decision -making and other aspects of Information Age conflict. He has published extensively in the peer reviewed open literature. His most recent works include the books 'Command and Control in the Information Age; Representing its Impact' (The Stationery Office, London, UK, 2002) 'Complexity Theory and Network Centric Warfare' (CCRP, Dept of Defense, USA, 2003) 'The Agile Organisation' (CCRP, Dept of Defense, USA, 2005) and 'Adapting Modelling and Simulation for Network Enabled Operations' (CCRP, Dept of Defense, USA, 2011).