Proceedings of the 2001 Winter Simulation Conference
B. A. Peters, J. S. Smith, D. J. Medeiros, and M. W. Rohrer, eds.

STUDY OF AN ERGODICITY PITFALL IN MULTITRAJECTORY SIMULATION

John B. Gilmer, Jr.
Frederick J. Sullivan

Wilkes University
Post Office Box 111
Wilkes-Barre, PA 18766, U.S.A.

ABSTRACT

Multitrajectory Simulation allows random events in a simulation to generate multiple trajectories. Management techniques have been developed to manage the choices of trajectories to be continued as combinatorial explosion and limited resources prevents continuing all of them. One of the seemingly most promising methods used trajectory probability as a criterion, so that higher probability trajectories were preferentially continued, resulting in a more even distribution of (surviving) trajectory probabilities, and better than stochastic approximation to a reference outcome. It was also found that this management technique introduced a failed ergodicity assumption. The higher and lower probability trajectories behave differently to a significant extent. The effect is to limit the number of trajectories which can usefully be applied to the problem, such that additional runs would fail to converge further toward the definitive reference outcome set. This may be a useful model for understanding other simulation modeling limitations.

1 BACKGROUND

The goal of multitrajectory simulation is to explore the outcome space of a simulation, that is, the set of all possible outcomes, more systematically and less expensively (for a given quality of understanding) than can be achieved with conventional stochastic simulation. (Gilmer and Sullivan 1998). The heart of the proposed method is to explicitly track each possible simulation trajectory, as illustrated in Figure 1. When an event that would normally be stochastic occurs, instead of one outcome, multiple outcomes are generated, each constituting a trajectory having its own state. Because the trajectory bifurcates, this is also referred to as “splitting”, with “cloning” of the state. In concept, such a multiple trajectory simulation is integrated with its support system in such a way that its use provides outcomes with probabilities associated with each, and an accounting for the key events or circumstances leading to the differences.

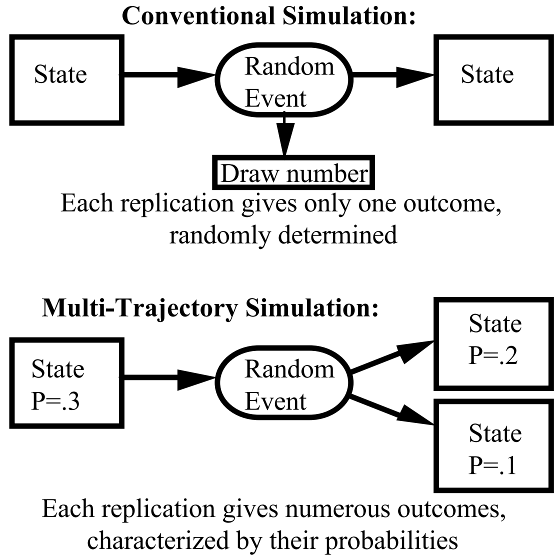

Figure 1: Concept of Explicitly Tracking Trajectories

The peril in this approach is combinatorial explosion as the number of events becomes large. We have assumed a simulation context in which only relatively important events are treated in this manner, perhaps hundreds in a typical trajectory. However, even then, there must be pruning of the tree of possible trajectories. The method which has been used is the “truncation” of trajectories. Event management policy implemented by a control method explicitly decides, for each event in some trajectory, whether to resolve the event in multitrajectory fashion (resulting in the creation of a new trajectory) or to instead resolve it deterministically or stochastically. The case of stochastic resolution with only one continuing state corresponds to the “Russian Roulette” used in conjunction with “splitting” as a variance reduction technique.

This approach has been explored in the context of force on force military simulations. In many scenarios, multitrajectory methods which managed the events did quite well, better than runs having significantly larger quite well, better than runs having significantly larger numbers of stochastic replications. However, the multitrajectory runs also seemed to reach a plateau of performance (described later), where increasing numbers of trajectories yielded no benefit. The problem tended to occur most often in the simpler, smaller scenarios, and was noticed most in the "two division" scenario (referred to henceforth as "scenario 2"), shown Figure 2. The problem remained unexplained until the discovery of the ergodicity pitfall reported in this paper.

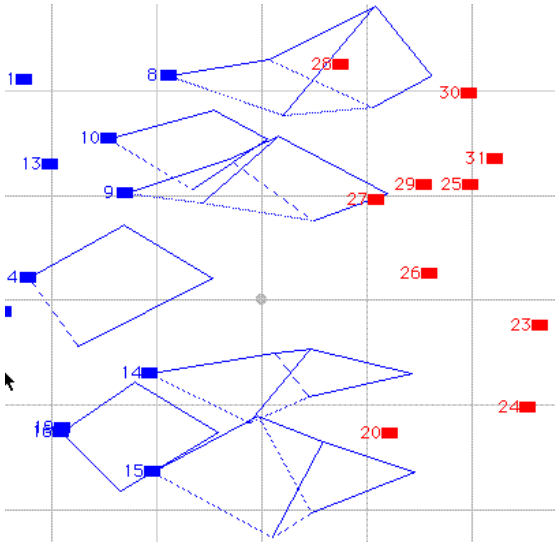

Figure 2: Two Division Scenario

2 THE MULTITRAJECTORY SIMULATION

This research has been conducted using a simplified, unclassified surrogate for the military simulations of interest. The simulation "eaglet" was designed to resemble the Corps level simulation "Eagle" in important respects, but to be of manageable simplicity. It includes Lanchester square law combat, movement by nominally battalion sized units along routes with multiple paths, decisionmaking, and artillery support. Figure 3 illustrates the smallest scenario we have used with "eaglet"; it is the one in which the problem reported here was most clearly manifest. In addition to the planned routes shown, units have some alternative routes (not shown). This scenario was originally constructed for illustrative purposes to show a situation where perverse results can occur in a simulation. A Blue reconnaissance unit encounters a Red unit in defense, then withdraws. A larger Blue force then attacks. The results of the reconnaissance encounter, if it leads to an ineffective Red force, is that a stronger Red unit moves up to the defensive position. Thus, initial success by Blue can cause a later attack to be made under disadvantageous conditions. As a result, this scenario has a very wide variety of possible outcomes, and shows a more pronounced nonmonotonicity (or "chaos") in its response than any of the others examined.

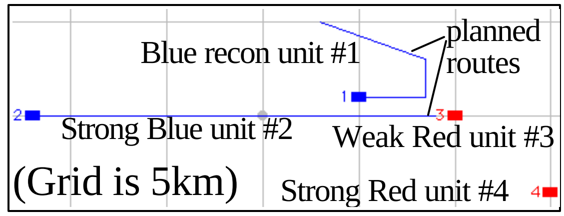

Figure 3: Simplest "eaglet" Scenario

3 THE CONVERGENCE ISSUE

The earlier project to assess multitrajectory performance focused on two particular Measures of Effectiveness, Blue Losses and Loss Exchange ratio, and compared how well multitrajectory techniques do relative to stochastic simulation at approaching a definitive reference. The methodology is outlined in Figure 4 below. The reference is generated by making a huge number of stochastic replications (5M originally, and later as many as 50M).

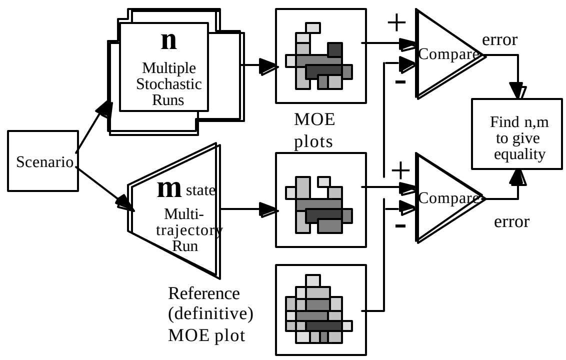

Figure 4: The Analysis Process Used to Compare Multitrajectory and Stochastic Runs

In the earlier project, it was found that the simpler multitrajectory methods did not converge on the stochastic reference as rapidly as traditional stochastic simulation. This was particularly true for the policy identified as "mt4" where all choices are multitrajectory (with :"splitting") up to a given state limit, and all further choices are stochastic (random). The plots appeared "grainy". Figures 5, 6, and 7 illustrate for Scenario 2 the MOE plots generated with 5M stochastic trajectories, 10K stochastic trajectories, and 10K mt4 trajectories.

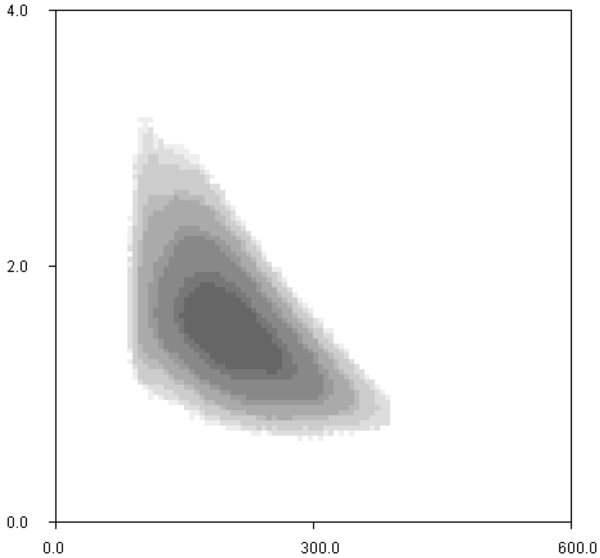

Figure 5: Scenario 2 Stochastic 5M Trajectory Reference

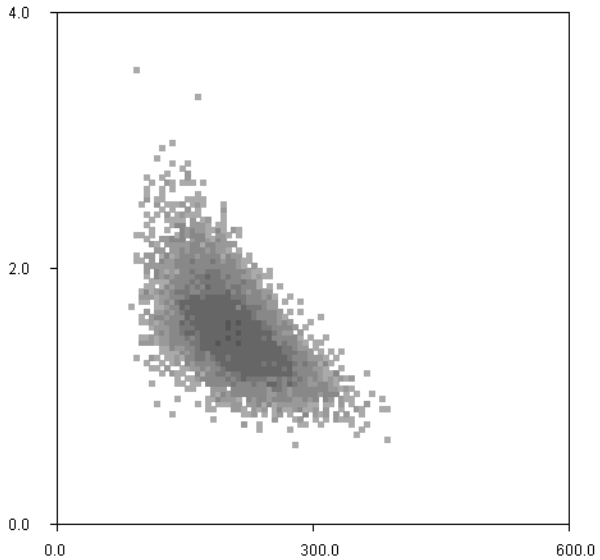

Figure 6: Scenario 2 10K Stochastic Trajectories

The reason for this measured poorer performance by the multitrajectory method "mt4" became apparent. The multitrajectory states have a very wide range of probabilities, and the less important ones count for very little in assessing distances to the reference. The more probable ones carry more weight and contribute to a pronounced quantization effect. In contrast, the stochastic run trajectories all have the same probability weight, and do not suffer this problem. In information theory terms, the wide range of probabilities means that the set of trajectories cannot convey as much information. This problem led to the development of a new multitrajectory management method "mt6", intended to "fix" the problem. In this method, there is a "soft" state limit, typically 1% to 10% of the ultimate state limit, beyond which only the high probability trajectories (those with a probability of 1/(m*c) or greater are treated in multitrajectory fashion, where m is the maximum state limit and c is a parameter chosen by the analyst. (Values of c much larger than 1 can limit the final state count to less than m.) Lower probability trajectories use a stochastic choice. It was thought that the random choices for the low probability trajectories and "splitting" of the high probability trajectories would result in an accurate distribution better than for purely random (stochastic) choices.

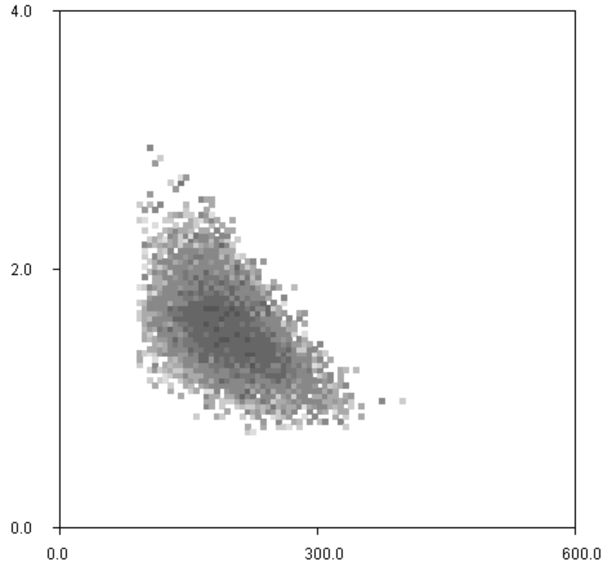

Figure 7: Scenario 2 10K Multitrajectory "mt4" Trajectories

Method "mt6" was shown to perform better than stochastic simulation in at least some cases, assuming that the criterion parameter c and soft state limits are well chosen. The MOE plots appear indistinguishable from the stochastic ones to the eye. Table 1 shows what can be achieved in this (two division) scenario with a well chosen "soft" state limit and criterion factor c. Two different smooths were applied to the plots prior to making the comparisons to minimize the effect of quantization grain.

The multitrajectory method performs better than stochastic methods for this scenario for some settings, although not by much of a margin. (The multitrajectory method did considerably better in the cases of larger scenarios of four and eight divisions.) Based on these data, it was thought that the problem encountered earlier was indeed fixed.

Table 1: Scenario 2 Distance Measurements for 10K Trajectories (mt6) compared to the Reference

10K trajectory runs (with varying soft limits. c=1.42):

	unsmoothed	3 x 3 sm	11 x 11 sm
stochastic:	2.69e-05	0.99e-05	2.59e-06
MT6 1:	2.75e-05	1.05e-05	2.91e-06 (c= .1)
MT6 32:	2.85e-05	1.05e-05	2.53e-06
MT6 128:	2.74e-05	1.10e-05	2.52e-06
MT6 512:	2.63e-05	0.98e-05	2.24e-06
MT6 1024:	2.73e-05	0.97e-05	2.41e-06
MT6 2048:	2.81e-05	1.05e-05	2.94e-06

When scenario 0 was first developed as a tutorial example, it was studied using the MOE convergence methods in order to gain an understanding of its behavior. To our surprise, Scenario 0 performed rather "badly" in a number of respects. Figure 8 shows an MOE plot for 50 Million stochastic replications. Several peculiarities are of note. Larger scenarios had MOE plots (for large numbers of replications) that were fairly smooth and regular, almost Gaussian. This was decidedly not the case for Scenario 0. The surface is uneven, and fairly high concentrations of outcomes appear at various points at the periphery of the pattern. Variations in probability include both fine and coarser components. In some areas there is a blurred, or smoothed, effect. In others, there are sharp boundaries.

In addition, the multitrajectory plots consistently compared badly with stochastic plots. Figure 9 shows an MOE plot for a 5M "mt6" multitrajectory run. This may look very similar to the 50M stochastic reference, but it differs significantly. Stochastic runs converge on the 50M stochastic reference as the number of replications increases. The "mt6" runs do not. Beyond about 10K replications the distance to the reference remains constant, and poorer than that for same sized stochastic runs. This is not obvious. By taking a difference between the two histograms, we can see large scale differences that conform to a peculiar pattern, as seen in Figures 10 and 11.

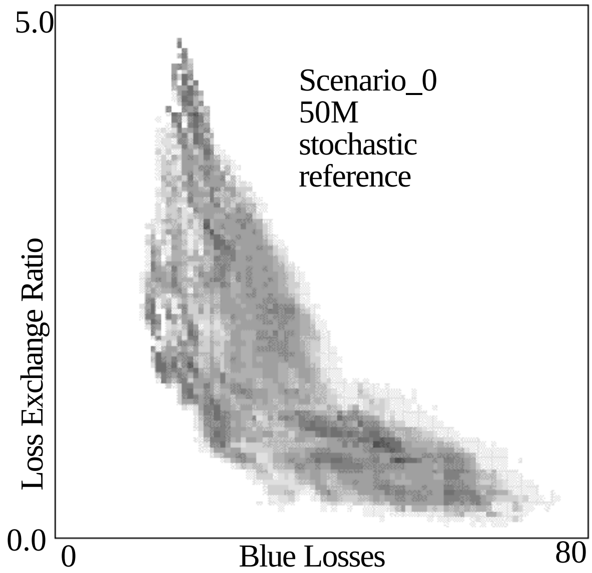

Figure 8: MOE Histogram for Scenario 0, 50M Stochastic Replications Used as a Reference

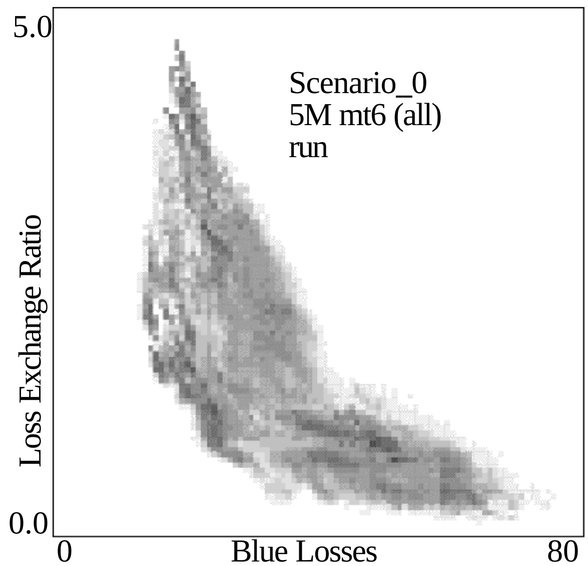

Figure 9: MOE Histogram for Scenario 0, 5M Trajectories, Method "mt6"

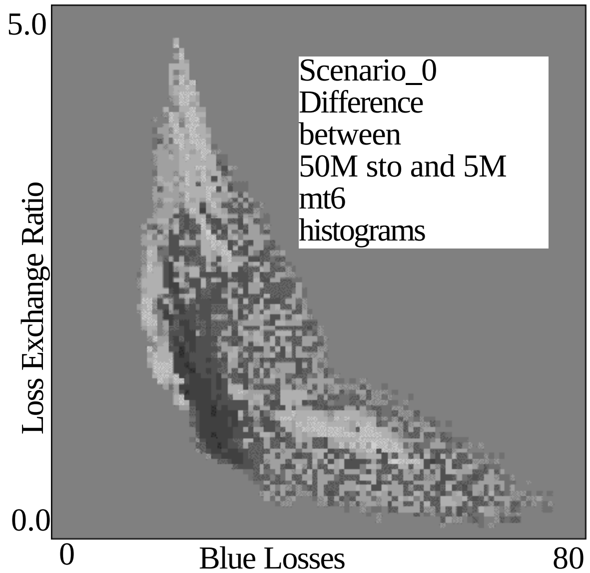

Figure 10: Unsmoothed Difference Plot Between the Stochastic 50M Reference and 5M mt6 Histograms

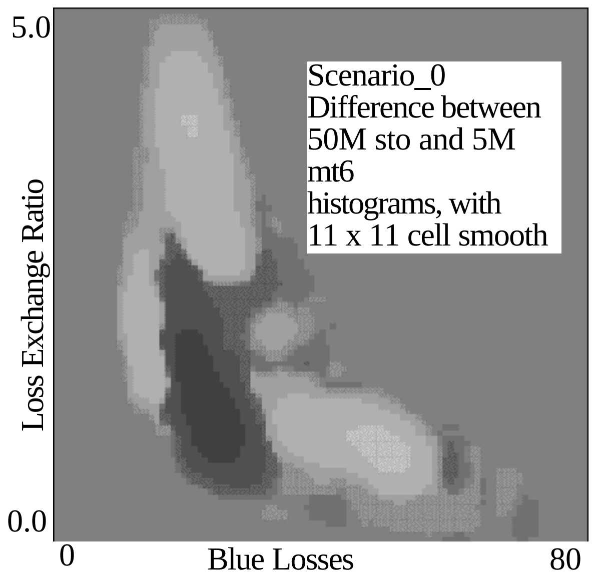

Figure 11: Smoothed Difference Plot Between the Stochastic 50M Reference and 5M mt6 Histograms

4 ANALYSIS

Ultimately, analysis showed that the fundamental problem was that trajectories bound for different parts of the MOE plot had significantly different average probabilities. Figure 12 shows a plot of average probability per trajectory for the 5M trajectory mt6 composite run.

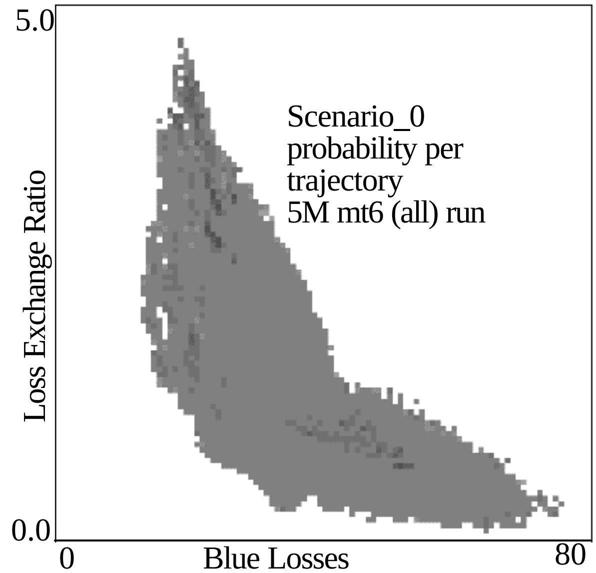

Figure 12: Probability per Trajectory for Scenario 0, mt6

The “mt6” method preferentially continues trajectories with higher probabilities in order to achieve a small spread in trajectory probabilities. The variations in probability per trajectory are small. The variations seen correlated with the discrepancies in the MOE plots. This same kind of discrepancy pattern was seen over a large number of different conditions, including some runs using techniques more complex than mt6, but was particularly characteristic of mt6. (The 5M histograms are shown because this was the largest mt6 data set available. It is a composite of ten 500K runs.) A plot of trajectory probability for stochastic runs would be uniform over the region having outcomes, because each stochastic trajectory is assumed to carry the same weight.

The mt6 method, by using a probability sensitive criterion for multitrajectory resolution, also subtly changed the distribution function itself. This hypothesis was tested by looking at results from a large “mt4” run, in which probability of a trajectory plays no role in the decision whether to continue a trajectory or not. Trajectories are spawned until the state limit is reached. This final limit is, of course, reached much earlier than for mt6 runs. Once the limit is reached, stochastic resolution is used. Figure 12 shows the MOE plot for an “mt4” 800K trajectory run. Neither mt4 nor stochastic choices would be biased by any consideration of trajectory probability. Thus, we can see how trajectory probability might correlate with where on the MOE plot trajectories are found. Figure 13 below shows the original mt4 plot. Keep in mind that this is a much smaller set than the 5M mt6 plot, although even at the same number of trajectories mt4 gives a much “grainier” (more noisy) plot that does not approach the stochastic reference as well as mt6.

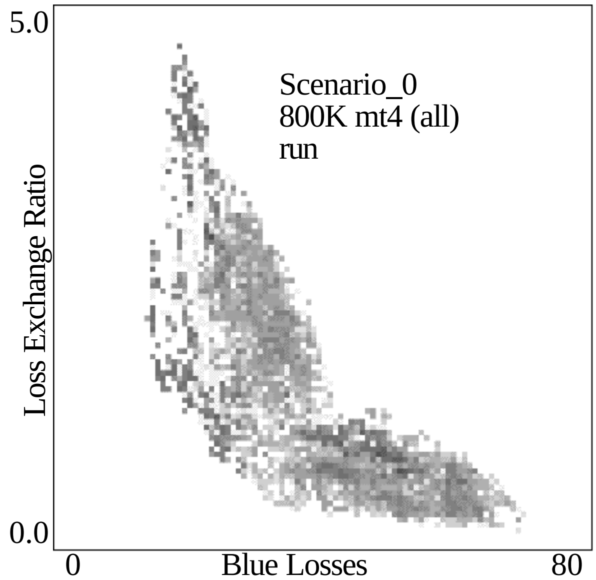

Figure 13: Scenario 0 MOE Histogram for 800K Trajectories, Method mt4

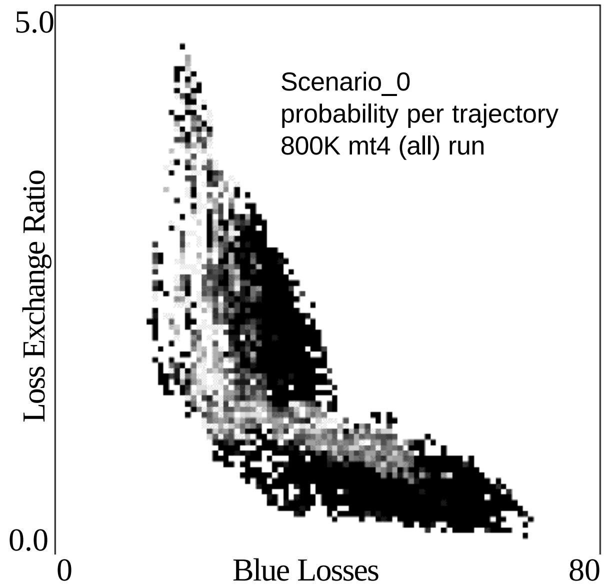

Figure 14: Scenario 0 Histogram Probability per Trajectory for mt4

The remarkable insight from the mt4 run is seen in the plot of average probability per trajectory, seen in Figure 14. Here we see dramatic differences in the probabilities of trajectories bound for different parts of the MOE plot. The shadings show a range from 1/20 the average of 1/N to 20 times the 1/N average trajectory probability, with many cells of the histogram exceeding the factor of 20. The regions of low probability (lighter shading) per trajectory correlate with regions in which the mt6 runs came out short of the stochastic reference: those areas that are lightly shaded in Figure 11 earlier. What has happened is that most of these trajectories have been lost under method mt6, with a resulting higher concentration of trajectories in the areas of the MOE plot having high probability per trajectory.

A test of this hypothesis is a difference plot between the mt4 histogram and the stochastic reference. It should not show the distinctive difference pattern that the mt6 method did, and should not show much of a pattern at all. Figures 15 and 16 show the unsmoothed and smoothed difference plots.

The unsmoothed difference plot seems to show no discernible pattern, in contrast to the unsmoothed mt6 pattern. The mt4 run is much smaller than the mt6 run seen earlier, and approximates the stochastic reference less well due to the wide probability range. With that considered, the smoothed difference plot shows differences, but of a seemingly random nature as might arise from noise, and certainly not conforming to the mt6 characteristic discrepancy pattern.

What we have is, essentially, an ergodic assumption gone awry. The mt4 method, which was not sensitive to probability, did not suffer from this problem. But the mt4 method was even less satisfactory than mt6 with respect to distance from the stochastic reference due to the granularity problem described earlier.

The nature of this problem was not apparent until we looked at Scenario 0 runs. This line of inquiry revealed an important potential pitfall in multitrajectory methods. How pervasive this problem might be is not known, since time and resources to pursue it further were not available. The very small size and wide range of possible outcomes for this scenario are unusual. There was no indication that the other scenarios were affected by this phenomenon nearly as much, if at all. However, it may be that this problem is most likely to occur for precisely those scenarios where analysis is most important: those in which the issue is in doubt and may swing on relatively few critical events which are, a-priori, not well understood.

5 CONCLUSIONS

Currently, we must conclude that the stochastic behavior of military simulations is not yet sufficiently well understood to recommend the implementation of multitrajectory methods in operational analytic simulation tools. This effort proceeded with the assumption that this might be the outcome, and used an experimental method to assess the potential benefits. This experimental method has so far not been undergirded with a firm theoretic understanding of the simulations of interest. Consequently, it is not possible at this time to suggest exactly what might be done to make trajectory management more effective. Rather, a deeper investigation into the stochastic nature of combat simulations seems to be needed, together with a more precise definition of those attributes of a simulation outcome set that most benefit analysis.

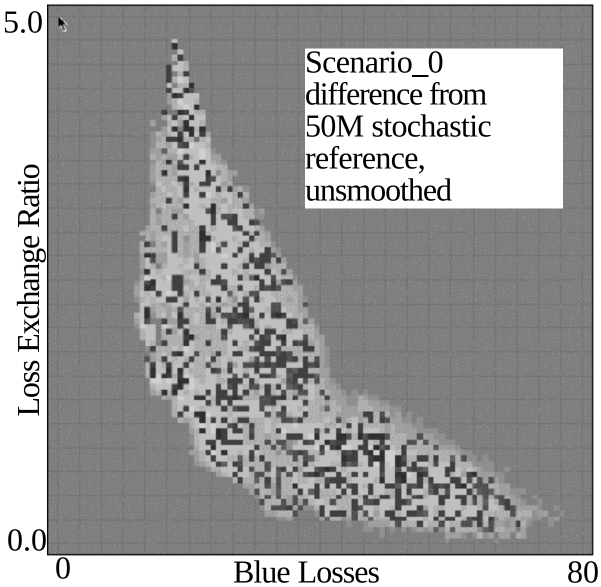

Figure 15: Unsmoothed Histogram Differences for 800K mt4 Run Compared to 50M Stochastic Reference

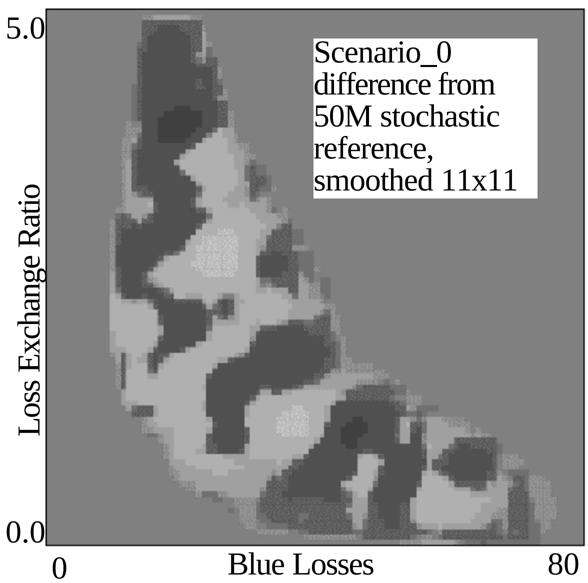

Figure 16: Smoothed Histogram Differences for 800K mt4 Run Compared to 50M Stochastic Reference

In particular, there are significant ergodicity issues in combat simulations, if it can be accepted that eaglet is at all representative. Some parts of the response surface may be populated by many very small probability outcomes, other parts having about the same probability density are populated by fewer higher probability trajectories. Techniques (such as our mt6 method) which try to maximize coverage by focusing on higher probability outcomes will distort the understanding of the outcome set, at least when the number of trajectories gets large enough that quantum effects are small. This issue could be very important in the context of planning systems, where there is a temptation to prune unlikely possibilities to conserve planning resources.

We consider this a "non-ergodic" effect, in which assumptions that some outside variable (time, or in our case, probability), is unimportant, turns out to be wrong. The discovery that this effect was significant was an unpleasant surprise in the course of a project focused on other objectives. It did prove interesting that the particular effect could measured in terms of a number of replications which beyond which no improvement could be achieved. This relationship between a shortcoming of the model and the number of worthwhile replications may prove quite useful. For example, a modeling decision to use deterministic rather than stochastic representation of some event is also an assumption that the random effects will balance out, that is, it is an ergodicity assumption similar in character to what we have investigated here. It may prove possible to characterize each such assumption in terms of its effect on the fidelity of the model as an impact on the maximum number of useful replications. Further development of this concept would seem to be worthwhile.

ACKNOWLEDGMENTS

This project and those immediately preceding it were funded by the US Army Research Office under Grants DAAH04-95-1-0350, DAAG55-97-1-0360, and DAAG55-98-1-0451 with the sponsorship of the US Army Center for Army Analysis, particularly Mr. Gerry Cooper, Col. Andrew Loerch, and Dr. Robert Alexander. We also appreciate the help given by Dr. Kosmo Tatalias. The authors are indebted to these persons for their encouragement and support.

REFERENCE

Gilmer, John B. Jr. and F. J. Sullivan. 1998.. Alternative implementations of multitrajectory simulation. *Proceedings of the 1998 Winter Simulation Conference*, ed. D.J. Medeiros, F. J. Watson, J. S. Carson, and M. S. Manivannan, 865-872. San Diego, California: Society for Computer Simulation.

AUTHOR BIOGRAPHIES

JOHN B. GILMER JR. worked in the development of combat simulations, with a focus on C2 representation and parallelism, at BDM, Inc. He was the chief designer of the CORBAN combat simulation. He has a Ph.D. in EE from VPI and currently teaches Electrical and Computer Engineering at Wilkes University. His e-mail address is `<jgilmer@wilkes.edu>`.

FREDERICK J. SULLIVAN is Dean of Technology and teaches Computer Science at Wilkes University, and earlier taught at Rose-Hulman and SUNY Binghamton. His expertise is in operating systems and object oriented software. He received his Ph.D. in Mathematics, from LSU. His e-mail address is `<sullivan@wilkes. edu>`.