Proceedings of the 2000 Winter Simulation Conference
J. A. Joines, R. R. Barton, K. Kang, and P. A. Fishwick, eds.

IMPROVED DECISION PROCESSES THROUGH SIMULTANEOUS SIMULATION AND TIME DILATION

Paul Hyden

School of Operations Research and Industrial Engineering
Cornell University
206 Rhodes Hall
Ithaca, NY 14853, U.S.A

Lee Schruben

Department of Industrial Engineering
and Operations Research
University of California at Berkeley
4135 Etcheverry Hall
Berkeley, CA 94720-1777, U.S.A

ABSTRACT

Simulation models are often not used to their full potential in the decision-making process. The default simulation strategy of simple serial replication of fixed length runs means that we often waste time generating information about uninteresting models and we only provide a decision at the very end of our study. New simulation techniques such as simultaneous simulation and time dilation have been developed to produce improved decisions at any time with limited or even reduced demands on analysts. Furthermore, we have the tools to determine whether a study should be terminated early or extended based on the demands of the decision-responsible managers and the time-crunched analysts. By collecting information from multiple models at the same time and using this information to continuously update the allocation of finite computational resources, we are able to more effectively leverage every minute of calendar time toward making the best choice. Strategies and tactics are discussed and highlighted through the implementation and analysis of a job shop model. Target success probabilities are achieved faster while achieving goals in study length flexibility at low cost to analyst time.

1 INTRODUCTION AND BACKGROUND

Simulation study design often focuses on the serial allocation of a predetermined computer budget, such as the sequential efforts of Chick and Inoue (1998) and Chen, Yücesan, and Dai (1998). In addition, these approaches focus on output that is independent and identically distributed. As in Schruben (1997) and Hyden and Schruben (1999), we want to exploit some of the advantages of simultaneously replicating different models with continuously variable allocations of effort to each model. Note that this work appears as part of a broader context in Hyden (2000).

The motivations for such an approach are numerous. For one, simulation experiment budgets can include more than computing resources. For example, a study deadline or constraints on the analysts' time may be more critical than computer time. Furthermore, the exact amount of computing resources available until the study deadline is reached may not be known - particularly for important studies where extending deadlines may be preferred to making the wrong decision. Automation of experimental design decisions may be critical to efficiently using analysts' time or maximizing the use of available computing resources before a deadline is reached.

This is further complicated by the fact that when an experiment is halted may depend on the results obtained - budgets may not be set a priori but are negotiable as the study proceeds. A study may end sooner than anticipated when the early results suggest a clear answer, allowing attention to be directed toward other studies or other aspects of the same project. Conversely, more information may be desired when the performances of competing alternatives are close and the project will require further resources. An analyst may realize that a marginal increase in effort will result in a significantly improved answer. Not uncommonly, a reformulated study objective may be dictated to support a failing "pet" option or to invalidate the current best choice, requiring the study to continue. For these, and other, reasons it makes sense that simulation studies be designed to give the best possible answers at any time during the study.

To make this discussion explicit, we lay out several key assumptions about our decision process.

1. We have a limited time horizon for our decision and/or the value of our decision decreases with time.
2. We have a limit on computing power per unit of time.
3. Analyst time is a bottleneck

If 1 or 2 do not hold, we have virtually unlimited computing budget and hence do not need to be careful about where we spend our time. If 3 does not hold our analysis is still relevant, but it means that we do not really need to automate our decisions about resource allocations.

Schruben first introduced time dilation as a vehicle for model selection in 1997. The idea is to execute several simulation models simultaneously by varying their local clock speeds. Just as in physics where time slows down for objects moving at high rates of speed relative to objects which move more slowly, the local clock speeds for less interesting models will be slowed and faster clock speeds will be allocated to more interesting models. The net effect is that we spend more time simulating more interesting models.

To aid the reader, the term "model" will be consistently used to refer to one complete specification from the design region or search space. They may have identical, unrelated, or similar structure and parameter values (continuous or discrete). They may have independent or related initial conditions and random variates. However, they share a comparable metric on performance. Experiments will refer to a simultaneous simulation of a subset of the models from the search space, where the result is a decision on which of the models are preferred. Studies will refer to a collection of experiments, which taken together with a decision rule, will provide the decision prescribed by the total of the experiments.

2 METHODOLOGY

First of all, we lay out the necessary steps for our study.

1. Design individual simulation models. The first thing we need is a model with which to experiment. This is not the best place to begin in the larger context of a decision process, but it is where we will start our story. We will presume that a space of potential models has been defined and suitably refined to the point that simulation is necessary.

2. Determine surrogate for measuring simulation effort. Since a central notion of time dilation is the dynamic adjustment of simulation resources, we must have a measure for the consumption of those resources. We chose events as a reasonable surrogate for measuring simulation effort. This is advantageous because state variables only change at event executions and events represent the real processing that the CPU must face. It also makes management of simulation effort transparent. Simulation time is less useful because effort can only be affected indirectly by adjusting the relative clock speed of each simulation. Although not implemented in this example, it is certainly possible to adjust the relative costs of event execution for each simulation during the run, allowing the surrogates to more closely match the true costs to the CPU.

3. Modify existing model for simultaneous simulation. Fortunately, this potentially daunting task is rather straightforward. The key step is to add an additional index to all state variables local to each model to indicate which model each state variable is associated. This will allow each model to exist in the same simulation environment simultaneously. As an alternative, the models can be launched as independent but manageable threads, as done by Biles and Kleijnen (1999) using Silk. Note that values that are shared across models do not need to be indexed by model and instead can stored in one memory location, leading to savings in memory and lower overhead in loading values. This can be a significant savings for large models. In addition, slower memory locations can be used for relatively inactive models.

The other key step is to modify the simulation structure to allow for different resource usage rates for each model. Since we have chosen events to be our surrogate for simulation effort, the "time unit" in the experiment space becomes number of event executions. After a_i event executions, our experiment will select which model to be simulated for the next a_{i+1} time units, when a new model will be selected for simulation, where a_i may be a fixed parameter or it may be a function of the output. In multiple processor environments, each processor can take on a portion of the experiment, or each processor can take on an experiment wholly on its own. Alternatively, we may take a decentralized view where all models are running at the same time, but a central authority determines their relative rate of event execution. Further analyst effort can be employed to incorporate independent efforts to speed up simulation times, such as distributed or parallel computing techniques. (In fact, our decisions about which models are more important to study could be used to make decisions on which models are worth spending the development time on speeding up.)

In our implementation, we employ a simple structure that restricts us to simulating one model at a time. This was done to make the procedure transparent and easy to analyze. However, all models are active and can be executed at any time. For this example, the number of event executions between model switches was fixed at 1, but this could easily be a function of the output. For example, highly serially dependent output may not change significantly for several hundred events, necessitating less frequent switches.

4. Remove the analyst as much as possible. This refers to our commitment to limiting the analyst's involvement in the details of simulation experiment design. Of course, the judgment of the analyst will be necessary to direct higher-level strategies. In addition, the complex behavior of simulated systems will require the analyst to monitor any automated decisions.

At a closer level, the shell algorithm for time dilation has several key parts:

A. Choose a subset of n models from the set of possible model configurations (search space)
B. Warm up the models

C. Warm up the experiment
D. Assign probabilities $\{p_j\}$ that the next $\{a_{i,t}\}$ units of effort will be allocated to model i, i=1,2,…,n for experiment step t.
E. Reassign $\{p_i\}$ and select $\{a_{i,t+1}\}$.

On top of our system, we will generally need to employ some sort of optimization methodology to select which models in our search space are suitable for simulation. For our example, the search space is small enough to be simulated in its entirety.

Warming up the models refers to the typical warm up process required for discrete event simulation. In fact, time dilation and simultaneous simulation can make this process more efficient as well, but this is beyond the scope of this paper. The warm up of the experiment refers to the fact that the time dilation process must warm up as well. In our example, the experiments executed 2000 events before models were scored and time dilation was applied. Further work is necessary to study the impact of experiment warm up.

Note that units of effort are assigned to models randomly to account for the following problem. Any scheme that allocates effort deterministically will necessarily remain fixed for the duration of the deterministic schedule on that particular allocation, overemphasizing the accuracy of the current allocation. In addition, small relative apportionments of effort will require longer schedules, making this problem worse. To maintain our focus on using all of our available information, we propose random allocations, which will necessarily allocate the appropriate amount of effort in the long run. This eliminates the need for bookkeeping schemes, simplifies implementations and also provides for short-term deviations from the strict allocations prescribed by the scoring system. However, the authors have developed deterministic allocation methods, which will not be covered here.

Probability assignment and their subsequent update throughout the experiment get to the heart of the procedure. Considerations that make a model more or less informative include the mean and variance of the output measure. For individual models, serial correlation as well as trends in mean and variance of the output will be important as well. Furthermore, inter-model relationships such as cross correlation and continuity in output measure will affect information value. Finally, the cost of execution across models will also affect the relative information value of a model

Finally, various parameters appear as we begin to implement our simultaneous environment. For example, the frequency of probability update as well as the sensitivity of this update to new data needs to be considered. These parameters will be highlighted as they appear in our example.

3 RESULTS WITH JOBSHOP MODEL

To provide a tangible example to experimentally test some of our intuition, we study a job shop model given by Law and Kelton (2000). A job shop consists of multiple product types, each having a different routing through a network of job stations. Each job station has multiple machines.

For our example, there are 5 machine groups numbered 1,2…5 consisting of 3,2,4,3 and 1 machine(s) respectively. There are three job types occurring with probability 0.3, 0.5, and 0.2. Their routings are (3,1,2,5), (4,1,3), and (2,5,1,4,3) and their mean service times are (along routing, in hours) (0.50,0.60,0.85,0.50), (1.10,0.80,0.75), and (1.20,0.25,0.70,0.90,1.00). All service times at stations are 2-Erlang random variables. Job interarrival times are exponential random variables with mean 0.25 hours. Our performance measure is mean time in system per job.

Our decision problem is the following. We can add 1 machine for the same price at any of the five machine groups. Where should be add this machine? This gives us five natural models, each only differing in the number of machines at each station. The ith model will correspond to adding a machine at station i. Note that all other model definitions can be shared across models. Values such as number of served available at each station will need to be indexed by model during execution.

We will be seeking to choose the model with the lowest mean time in system in a sustained operation, i.e. the model with the lowest asymptotic mean time in system. Note that, by this definition, the correct model selection for every experiment and every study is the same. The authors were able to determine that model 2 is the correct selection. Of course, since every experiment is necessarily finite, the actual model selected will vary. This methodology seeks to make the usage of resources more effectively so that the ultimate decision of the experiment and the overall study is correct more often.

Inspired by intuition and simplicity, we will use a very simple scoring mechanism for determining the probability that a model will be selected for execution. First, we compute the distance between each models mean performance and its best competitor and call this D. The best competitor is the best performing model that is not itself. If we are allowing ourselves to select a subset of the models at the termination of our experiment, we may instead define the best competitor as the best performing model that is not in its current indifference set. Indifference sets are formed by placing the best performing model in a set S and adding all other models that perform within α units of the best performing model, where α is the indifference zone. Further sets are formed similarly by adding the best performing model not already in a set and adding all models within α units of that model. Next we compute the sample variance of the mean performance measure for each model

and call this V. In our implementation, a naïve estimation procedure was used, but efforts to account for inherent serial correlation should only improve the results. Each model is scored with V/D^2. Hence, we will be interested in simulating models that are more variable or more preferred with a greater portion of our simulation effort. Note that this scoring mechanism requires little analyst effort to construct, but it is certainly open to enrichment. A similar asymptotic allocation by Chen et al. (1999) was implemented by the authors with less successful results. The results will show improved decisions at all times during the study.

Note that two systems could be eliminated from the study without simulation after a bottleneck analysis of the initial model. However, our analysis will study our ability to select from all five models, since analyst time may not be available in a real study. However, we should expect our method to quickly be able to pick out the correct model with at least a probability of 33%. For further comparison, we also solve the case where those two models have already been eliminated from our search space.

The form of our study will, at least initially, consist of one experiment. This experiment will consist of all five models being run concurrently, with the probabilities of model selection corresponding to the score of the model divided by the sum over all scores. Model reselection will occur after every event execution. Probability reassignment will occur every 100 events. Note that all of these choices correspond to parameters that are open to alternative choices. Clearly, many alternative scoring mechanisms and translations of those scores to probabilities exist. A score based on weighing new and old data differently is just one example. Furthermore, a model could be executed for any number of events before a new model is selected, perhaps even as a function of the score. For example, we may wish to implement scores of $(a_1,\ldots,a_k)$ by executing a_i events of model i with probability 1/k rather than 1 event of model i with probability a_i / $(a_1+\ldots+a_k)$. Finally, probability reassignment could occur more or less frequently, or update frequency could be a function of the experiment itself.

Our decision rule for each experiment will be the natural one: the model with the lowest mean service time at the conclusion of our experiment will be selected as preferred. Our study will select the model that is preferred by our single experiment. One way we will be judging the effectiveness of our studies by looking at the probability that we select model 2. Note that this judgment becomes a multi-objective optimization problem very quickly if we consider that we might only slightly prefer the optimal model choice to the others, so that the probabilities of choosing sub optimal systems become important as well. Here we are thinking of the case where we only value the optimal choice. For example, suppose if we choose incorrectly we will be out of business because some competitor can be assured to make the winning choice. However, this method can be applied to other ways of weighing outcomes.

First, we considered studies with a single experiment where the number of events is capped at 100,000. Four cases are considered. Two cases will employ common random numbers (CRN) and two will not. CRN is implemented to insure that each of the five models will always see the same stream of arrivals and jobs types, and service times at each station for each of the models within any experiment. Two of the cases will allocate exactly 20,000 events to each model in the experiment. While this experiment is still more robust that a serially conducted experiment since it can provide an answer at any time, the resource allocation is typical of a serial study. The other two cases will employ time dilation, utilizing the scoring method described earlier.

The study was repeated with independent seeds 1,000 times for the two cases without CRN, and with 10,000 seeds for the two cases with CRN. The probability of selecting model 2 for each study is given below.

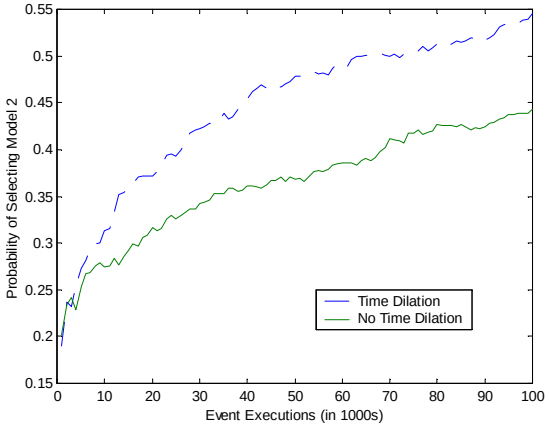

Figure 1: Probability of Correctly Selecting Model 2 without CRN.

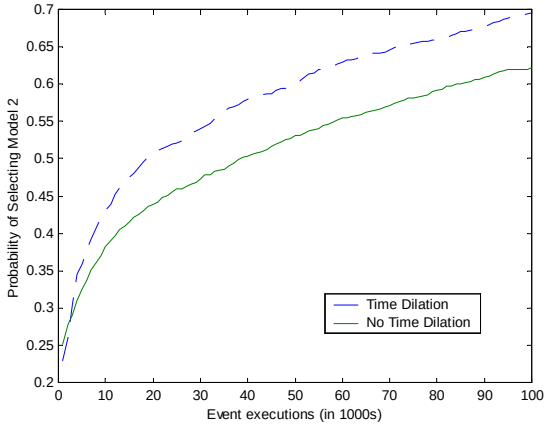

Figure 2: Probability of Correctly Selecting Model 2 with CRN.

For further comparison, we consider the case where we allow ourselves to eliminate two models without simulation effort. Models 3 and 5 can be removed from consideration due to a simple bottleneck analysis of the current configuration. A study consisting of a single experiment was conducted 1,000 times with and without time dilation and using CRN in both cases.

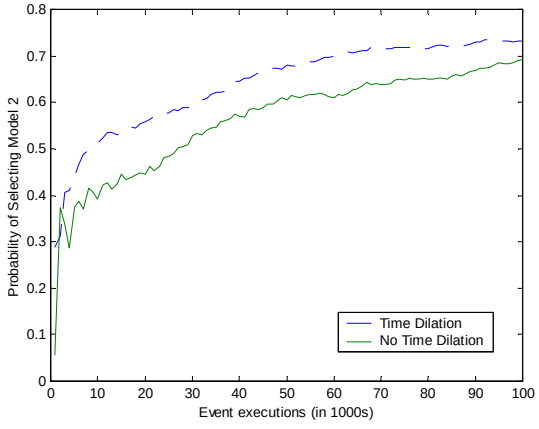

Figure 3: Probability of Selecting Model 2 from 3 Choices using CRN.

In all cases, we see that dynamic allocation of our simulation resources through time dilation significantly improved our capability of choosing the optimal model for all experiment lengths.

Next, 1,000 studies with a single experiment capped at 2,000,000 events was executed for both CRN cases. The following figure shows the probability of selecting model 2 as a function of the number of events executed.

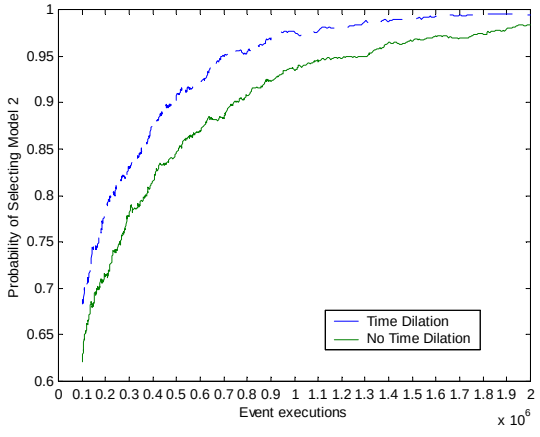

Figure 4: Probability of Selecting Model 2 with Longer Experiments using CRN.

Time dilation had superior performance throughout the range of potential stopping times. This is somewhat surprising, since the use of CRN may limit our ability to successfully vary the amount of effort on any model. This is because good or bad performance on additional observations of that model may unduly bias our results, since another model may have performed similarly good or bad if we had only observed it. Nonetheless, these results our encouraging. If we were aiming for a study design guaranteeing 95% probability of correct selection, our time dilation method requires a minimum of 600,000 events. Without time dilation, we would require 1,200,000 events.

One source of this improvement can be seen from two effects. First, time dilation helps us arrive at correct decisions earlier and hold on to those decisions upon further simulation. Second, time dilation keeps from holding onto as many incorrect decisions, and delays our choice on those decision which turn out to be incorrect longer. To see this effect, we consider the following graph. For each of the 10,000 replicated studies which utilized CRN in choosing between all 5 models, we compared the decision made after 100,000 events with the decision made after less than 100,000 events. Cases where these choices were the same were counted and summed across correct and incorrect decisions. Fractions do not sum to 1 since some decisions change during the course of the experiment.

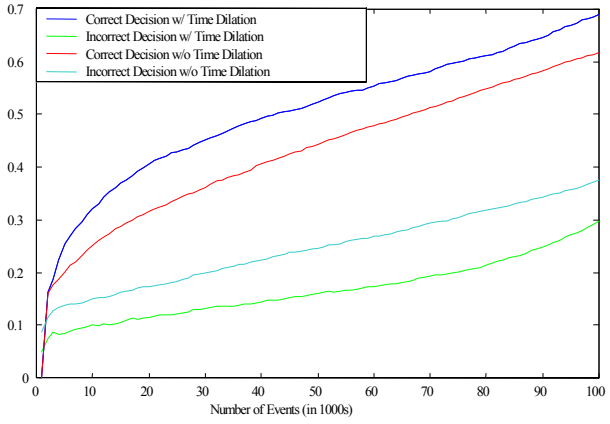

Figure 5: Fraction of Decisions That Match the Ultimate Decision of the Experiment for All Study Lengths.

The use of time dilation clearly shows improvement at keeping correct decisions and finding them earlier. At the same time, fewer incorrect decisions are kept and those that are made are not committed to until later in the experiment. For example, at the halfway point of the experiment, the case using time dilation has fixed on the wrong choice 50% fewer times. At the same time, time dilation has fixed on the correct decision 20% more often.

ACKNOWLEDGMENTS

The research reported here was partially supported by a joint SRC (FJ-490) and NSF (DMI-9713549) research project in semiconductor operations modeling, and a NSF Graduate Research Fellowship.

REFERENCES

Biles, W.E. and Kleijnen. 1999. A Java-based simulation manager for optimization and response surface methodology in multiple response parallel simulation. In *Proceedings of the 1999 Winter Simulation Conference*, ed., P. Farrington, H. B. Nembhard, D. T. Sturrock and G. W. Evans, 513-517. Institute of Electrical and Electronics Engineers, Piscataway, New Jersey.

Chen, C.H., E. Yücesan, and L. Dai. 1998. Computing budget allocation for simulation experiments with different system structures. In *Proceedings of the 1998 Winter Simulation Conference,* ed., D.J. Medeiros, E.F. Watson, J.S. Carson, and M.S. Manivannan, 735-741. Institute of Electrical and Electronics Engineers, Piscataway, New Jersey.

Chick, S. and K. Inoue. 1998. Sequential allocations that reduce risk for multiple comparisons. In *Proceeding. of the 1998 Winter Simulation Conference,* ed., D.J. Medeiros, E.F. Watson, J.S. Carson, and M.S. Manivannan, 669-676. Institute of Electrical and Electronics Engineers, Piscataway, New Jersey.

Law, A. amd W.D. Kelton. 2000. *Simulation Modeling and Analysis, Third Edition*. McGraw Hill.

Hyden, P. 2000. Time dilation and simultaneous simulation. Ph. D. Dissertation, School of Operations Research and Industrial Engineering, Cornell University. (In process).

Hyden, P. and L.W. Schruben. 1999. Designing simultaneous simulation experiments. In *Proceedings of the 1999 Winter Simulation Conference*, ed., P. Farrington, H. B. Nembhard, D. T. Sturrock and G. W. Evans, 389-394. Institute of Electrical and Electronics Engineers, Piscataway, New Jersey.

Schruben, L.W. 1997. Simulation optimization using simultaneous replication and event time dilation. In *Proceedings of the 1997 Winter Simulation Conference*, ed., S. Andradóttir, K.J. Healy, D.H. Withers, and B.L. Nelson, 177-180. Institute of Electrical and Electronics Engineers, Piscataway, New Jersey.

AUTHOR BIOGRAPHIES

PAUL HYDEN is a Ph.D. candidate at Cornell University's School of Operations Research and Industrial Engineering. His dissertation work focuses on designing efficient and practical simultaneous simulation experiments. He is also interested in semiconductor manufacturing, service operations, and finance. His email and web addresses are <hyden@orie.cornell.edu> and <www.orie.cornell.edu/~hyden>.

LEE W. SCHRUBEN is currently a Professor in Industrial Engineering and Operations Research at the University of California at Berkeley (IEOR@UC) and The Andrew S. Schultz Jr. Professor in Operations Research and Industrial Engineering at Cornell University (ORIE@CU) and is glad he is not dyslexic. His research interests are in statistical design and analysis of simulation experiments and in graphical simulation modeling methods. His simulation application experiences and interests include semiconductor production, dairy and food science, health care, banking operations, and the hospitality industry. Email addresses that might work are <lee@orie.cornell.edu> and <schruben@ieor.berkeley.edu>.