Proceedings of the 1998 Winter Simulation Conference
D.J. Medeiros, E.F. Watson, J.S. Carson and M.S. Manivannan, eds.

DISCRETE TIME SIMULATION OF AN EQUIPMENT RENTAL BUSINESS

R. Alan Bowman

Graduate Management Institute
Union College
Schenectady, NY 12308, U.S.A.

Ira J. Haimowitz

Pfizer, Inc.
235 East 42nd Street, 7th floor
New York, NY, 10017, U.S.A

Robert M. Mattheyses
A. Yonca Özge
Mary C. Phillips

GE Corporate R & D
One Research Circle
Niskayuna, NY, 12309, U.S.A.

ABSTRACT

In this work we report the results of a discrete time simulation model that we developed for an equipment rental business to study the impact of business decisions. The whole tool consisted of a user interface that enabled efficient viewing and modifying of the input data, executing the simulation program, and viewing the output reports. Utilizing this we applied cost/benefit analysis to the results of the simulation runs and identified profitable investment alternatives for the business. We also measured their asset population in terms of their profitability and quantified the relation between utilization, repair times, and responsiveness to the customers.

1 INTRODUCTION

In this paper we report a simulation study of the dependence of various performance measures in an equipment rental business on controllable factors such as inventory levels and repair times. The size of the problem (with thousands of asset types and ten thousands of individual assets) and the complexity of the relation between the performance measures and controllable factors (decision variables) led us to use simulation in our analysis. This is a widely used technique to analyze such complex systems; using a finite but sufficiently long simulation run one can closely estimate performance at any setting of the decision variables. This approach when combined with gradient estimation (Ho and Cao 1991, Rubinstein and Shapiro 1993) becomes more powerful and can be used in the solution of stochastic optimization (Robinson 1996, Rubinstein and Shapiro 1993) and equilibrium problems (Gürkan *et al.* 1998).

Due to time and budget limitations commonly present in many real-world projects and the additional effort that would have been necessary to carefully optimize the system under study, we could not apply a simulation optimization approach. Instead, we applied Cost/Benefit analysis (Grant *et al.* 1990) to the results of the simulation runs and determined alternative operating policies that substantially improved performance, in this case utilization of assets (fraction of assets being rented at any given time), responsiveness to customers (i.e., fill rate), and return on investment. From an aggregate point of view there is a trade-off between utilization and responsiveness to customers. The leaders of the business tended to focus on utilization perhaps at the expense of investing to meet customer demand. In this study we showed that a careful choice (by quantifying the contribution to lost revenue of various asset types and using this information in the financial analysis) of asset types to invest in improved both measures.

The remainder of the paper is organized as follows: in Section 2 we motivate the work by describing the business and the problems they were facing. In analyzing the business we used a special simulation technique: discrete time simulation. In Section 3 we discuss the reasons behind choosing discrete time simulation. Section 4 describes interesting features of the problem resulting from unavailability of perfect data, bill of material structure of the orders, and efficiency considerations. We also describe how we handled these difficulties. In Section

5, we describe implementation issues concerning the user interface. This was an important portion of the work because our goal was to provide a tool that would be used repetitively for planning. Section 6 contains numerical results and sensitivity analysis; there we also describe how we applied cost/benefit analysis to the numerical results to compare investment alternatives. Finally, Section 7 contains conclusions and directions for future research.

2 PROBLEM DESCRIPTION

Rental or lease of equipment is a common practice in modern businesses. Rental periods may range from a few days or weeks to an almost semi-permanent length of time, as the business needs warrant. At that time the company, one of General Electric's many businesses, was in the business of serving this market in the area of special purpose equipment. As a rental agent, it has to respond rapidly to the needs of their customers. It is important for the equipment availability to match customer demand patterns. The business had revenues in the $\$50 - \$75M$ range while performing with a utilization around $60 - 65\%$. Management was of the opinion that utilization was too low and that revenues could be increased by reducing equipment maintenance and repair times to increase availability and therefore utilization. We were asked to develop a simulation model to help evaluate the impact of the proposed efforts.

The business shipped equipment in response to customer demand. The duration of a rental depended on the equipment type. Upon returning from rental the equipment could be in need of repair. This generic cycle through the four major states, Available, Rental, Test, and Repair, is illustrated in Figure 1. The actual state structure was

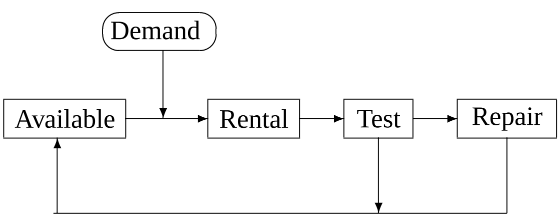

Figure 1: Material Flow Through States

more complicated with particular detail in the repair states incorporating in house and outsourced repairs, spare part deliveries, etc.. State residence times and transition probabilities were determined using historical data from the business. As a further complication, the product line had an extensive and complex interdependence due to the fact that some items were accessories that were required or optional add-ons for other items in the product line. Thus utilization of a piece of equipment could be influenced by its own population, its demand, and the population and demand for other items using the same set of accessories. To produce useful results we found it necessary to include this kitting structure in the demand model; see Section 4.3 for more about kitting.

Our objective in this effort was to deliver a model that embodied the relationship between equipment population levels, utilization, fill rate (percentage of orders that could be filled when received hence responsiveness to the customer), repair times, and revenues. Our model tracked the equipment populations and flows among the states of Figure 1. Based on the population distribution and stochastic models of demand, we estimated performance measures such as fill rates and utilization by equipment type. In addition, we measured complex phenomena such as contribution to lost revenue, the income that was lost due to the unavailability of a piece of equipment. Although the results of this study are specific to a single business, the resulting population model is generally applicable to rental businesses.

3 SIMULATION STRUCTURE

The events in the simulation were state changes for any particular asset. Since there were many thousands of assets, maintaining an ordered event list in a standard discrete event simulation would have been extremely time-consuming. We made two modeling assumptions that allowed us to pursue a discrete-time approach that was more efficient:

1. The routing of an asset from state to state depended only on its asset type (in this and the next section we refer to asset types as SKU, i.e., stock keeping unit) and not on its individual identity.

2. Changes of state within a day (although they did happen) could be ignored with no substantial effect on performance measures.

Under these assumptions, a discrete-time simulation model was built. Shedler (1993) contains a brief discussion of this particular simulation methodology and discusses translation of information from a discrete event model to a discrete time model.

In the model time unit was a day. Two arrays were maintained that tracked, for each SKU, the number of SKU's leaving each state each number of days into the future. One array tracked SKU's leaving the rental state and the other tracked the number of SKU's leaving all other states. Let N be the number of SKU's, T_1 be the maximum number of days an SKU could spend in rental, T_2 be the maximum number of days an SKU could spend in a state other than rental, and K be the number of states excluding rental. Then the size of the first array was $N \times T_1$ whereas the size of the second array was $N \times T_2 \times K$. SKU's leaving rental status were tracked separately because rental times for certain equipment were (possibly) much larger than times in other states. Since these arrays were not sorted, no insertion routines were required.

This modeling approach also was very consistent with the focus on utilization of assets that was present in the company. At the end of the simulated day, the arrays described above (projecting equipment movement among states) would be updated as would the number of assets of each type in each state. These latter numbers were at the heart of the performance measures used by the company and featured in the simulation model.

4 MODELING ISSUES

4.1 Rental Times

Our primary source of rental time data was a shipping file that recorded key information about each asset shipped for rental since a certain date. It was decided to aggregate the data by grouping the rental times of similar SKU's (using a group code maintained by the company) since individual SKU's often had very limited sample sizes. For each grouping, an empirical rental time distribution was built using the percentiles of the sample rental times for that group. These rental times were heavily censored, however, as many of the assets that had been shipped were still being rented as of the date the shipping file was created. The number of assets shipped, however, was not biased by this data censoring. In addition, the company's strong focus on utilization meant that accurate records were available for utilization (number of assets being rented). This allowed us to use Little's Law (Little 1961), $L = \lambda W$, where:

L = average number of assets being rented (rental utilization),

λ = average number of assets shipped for rental per day, and

W = average number of days an asset is rented,

to calculate the average rental time for each SKU grouping. The empirical rental time distributions were then factored up to yield the calculated average rental time for each grouping.

4.2 State Residence Times and Transition Probabilities

We had two primary sources of data to model asset progress through states other than rental. First, we had the utilization reports, which gave the average number of assets in each state at an aggregated level. Second, we had a daily asset tracking report, which showed the current state of each asset each day, along with the previous state for that asset. Due to the sparseness of data for individual SKU's we aggregated the information provided by this file. The equipment rental business labeled all SKU's by four life cycle codes, indicating the maturity of a particular product. The life cycle codes were: new product introduction, mature, recently obsolete, and obsolete. These life cycle codes proved valuable as a natural grouping for the SKU's to determine the transition probabilities and state residence times. For each life cycle code, the number of transitions from state to state was extracted from the asset tracking file and used to form a transition probability matrix.

A sample of state residence times was also compiled from the same data. The overall sample size for each state was quite small, leading us to model the state residence times (other than rental) as exponentially distributed random variables. Once again, we applied Little's Law to estimate the average time spent in each state. Before doing this, however, for each life cycle code we converted the average number of assets shipped per day, for which there were accurate data, into an average number of visits per day to each state. This would provide the λ for each life cycle code for each state to be used in Little's Law. We computed the average number of visits to each state using standard absorbing Markov Chain theory (see, for example Winston 1987). Upon being returned from rental, there were any number of states that an asset might enter. Eventually, all assets would progress to the Available state, the unique absorbing state, which indicated readiness to be rented again. The average number of visits to state j per asset shipped is calculated by:

$$\sum_i p_i[(I-Q)^{-1}]_{ij},$$

where

p_i = the probability that an asset returning from rental is initially placed in state i,

I = the identity matrix, and

Q = the portion of the transition probability matrix involving transient states.

The average number of visits per asset to the absorbing state is, of course, 1. These calculations were done separately for each life cycle code. For a given life cycle code, for each asset type the resulting average number of visits to a state is then multiplied by the average number of assets shipped per day to yield the average number of visits per day to that state (λ in Little's Law). The average number of assets in each state (L in Little's Law) is then divided by the corresponding λ to give the estimate of the average time spent per visit to the state (W in Little's Law).

4.3 Kitting

Customer demands were often for "kits" of assets. A kit consisted of a mainframe (the main piece of equipment desired) and accessories (pieces of equipment used along with the mainframe). The shipping file listed all the assets that were shipped in a particular customer order but did not indicate which assets were mainframes, which were accessories, or which accessories went with which mainframes (many customer orders were for multiple mainframes). To supplement our analysis we combined the shipping information with a file describing which assets were mainframes, which were accessories, and which accessories went with which mainframes. This file contained some simplifications since there were often times some substitutability among the accessories. For each accessory/mainframe combination listed in this file, the shipping file is used to estimate the probability that a customer would request that accessory along with that mainframe along with the number of units needed (if any at all). The customer demand process is then simulated as a three-stage process:

1. The customer arrivals were created according to a Poisson process with rate estimated from the shipping file.
2. The number of mainframes demanded for each order is generated from an empirical distribution created according to the proportions of orders in the shipping file.
3. The accessories for each mainframe in the order were generated according to the probability the accessory would be included and the number needed.

4.4 Deriving Demand from Shipments

We encountered an interesting and difficult problem in trying to model demands when all our data were for actual shipped units. The company had some information on lost rentals but it was relatively sparse, anecdotal, and not consistent across items. Simulations of the system with simulated demand rate equal to actual shipping rate indicated that there would be significantly more lost rentals (due to equipment unavailability) than were being recorded.

The company wanted to assess the business impact of changes (for example, the overall effect on utilization if all time spent in certain repair states could be reduced), as well as to examine issues on an asset basis. For example, what assets should have their population increased or decreased? Therefore, we initially tried to iteratively factor demand on an SKU basis until the simulated ship rate would equal the actual known ship rate. Let t be the current iteration count, $d_t(i)$ be the demand rate for the ith asset type in iteration t, $fdm_t(i)$ be the fraction of demand for asset type i met during iteration t, $ssr_t(i)$ be the shipping rate of the current iteration, and $asr(i)$ be the actual ship rate compiled from their shipping files. The iterative procedure with which we factored demand is done as follows:

1: Let $t = 1$ and simulate the system with $dr_1(i) = asr_(i)$

2: For each iteration t, check whether the vectors asr and ssr_t are sufficiently close (closeness is measured by the Euclidean distance). If close stop, else go to Step 3.

3: For each asset type i, set $dr_{t+1}(i) = dr_t(i) + [asr(i) - ssr_t(i)]/fdm_t(i)$

4: Simulate the system with dr_{t+1}, set $t = t + 1$, and go to Step 2.

We envisioned that the iterations would cause the simulated ship rate to converge (from below) to the actual ship rate. In practice, however, this convergence is not realized. This lack of convergence is due to the sparseness of the demand (on an SKU basis), the lumpiness of the demand (caused in large part by the kitted demand), and the extreme variability in both the rental times and the repair times. It became obvious that we would have to aggregate the SKU's in some manner to be successful in converting the ship rates to demand rates. We did not have time to explore fully the best approach for aggregating the SKU's but we would identify this as an area for further research. We proceeded to aggregate the SKU's by a code kept by the company based on the industry segment served. The iterative factoring on this aggregated basis worked well and the simulated ship rate quickly converged to the actual ship rate. We were satisfied that this approach led to good validity for examining business impact of decisions but we would have felt more comfortable with measures involving individual SKU's if either there were a less aggregate method of converting ship rate to demand rate or else comprehensive demand data for SKU's could be gathered.

4.5 Warmup Period

The final simulation model was quite large and complex. The simulation tool was to be used in an iterative fashion for extensive sensitivity analysis and, to some extent, for optimization (of SKU levels, for example). It was also to be used on an ongoing basis as parameter estimates within the model changed. For these reasons, it was imperative that the simulation generate output measures efficiently. As one effort to ensure that this happened, the simulation tool was carefully examined for efficiencies; several were successfully implemented. Another effort involved the determination of the shortest warmup period that would be sufficient to ensure the system was approaching steady-state. The simulation generated many performance measures at many levels of aggregation. Clearly, however, the key measure was utilization. We divided the states into four categories: Available, Rental, Test, and Repair. We then tracked the fraction of SKU's in each of these states as simulation time advanced. The initial state for each run of the simulation was that all SKU's were in the Available state. Figure 2 shows the average fraction of

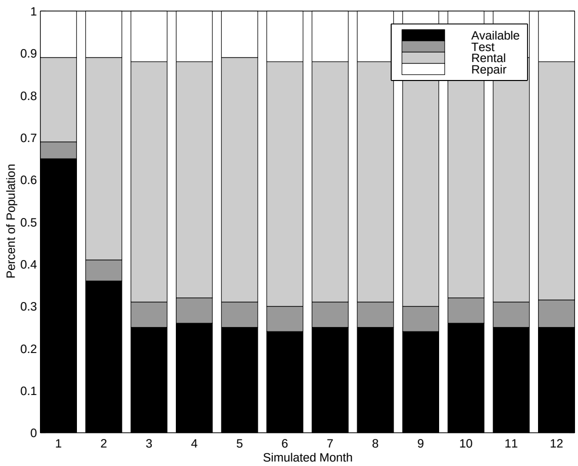

Figure 2: Warmup Behavior

the total SKU's that fell into each of the four categories for each month of simulated time. A visual inspection of the graph revealed that a 6-month warmup period was necessary and also sufficient. We discussed some ways to shorten the warmup period needed without biasing the performance measures (this would have been especially useful considering the potential use of the tool on a frequent basis as a planning device) but left this for further research and a later phase of the project.

5 IMPLEMENTATION ISSUES

Since we built our own simulation engine to model the population flows among the states in the business we needed a framework within which to develop a user interface. The interface needed to provide a flexible interface for decision makers in the business to use when defining and evaluating alternative business policies and equipment investment decisions. We chose Microsoft Excel (versions 5 and 7), using Visual Basic for Applications, as the framework for reasons of availability in the customer business and ease of programming. The latter reason applies to most other commercial spreadsheet tools. Due to reasons of complexity the interface was partitioned into input and output tools.

5.1 The Input Tool

The input tool provides an interface for displaying and editing demand/inventory, repair, and rental information. A main selection screen furnishes the basis of navigation through these subsets of information. Buttons are provided for editing the data, running the simulation, and viewing the output. This main selection screen also provides an overview of the program - alerting the user to those items which should be considered before running the simulation. Baseline values for demand/inventory, repair, and rental information are displayed on spreadsheets which could be edited by the user. Buttons are provided to sort the data in a variety of ways (by industry segment, life cycle stages, etc) allowing easier group selection and editing of data. When the data is ready to run a button push would save the final information to a file, to be read by the simulation program which is implemented in Fortran 90. Again, by button push, the simulation program is started (shelled from Excel); it reads the newly generated input file and, upon completion, generates an output report file. A button on the navigation screen starts up the output tool which displays the results to be viewed immediately or saves for later viewing. The input tool also allows the user to save information for a complete simulation session, so that runs could be repeated at a later time.

5.2 The Output Tool

The output tool is a separate Excel application which can also be run stand-alone using the output file from a previously generated simulation run as well as being initiated from the Input tool. The output tool gives a quick presentation of the statistics derived from the simulation program. In addition, the tool provides an easy way to display to the user various views (tables, charts, and trade-off plots) of the simulation output at different levels of detail. The formatting and charting functions in Excel also supported rapid response to customer requests for interface changes. An example of such an augmentation of the interface is the rapid implementation of cost-benefit analysis in Section 6 for evaluating results.

5.3 Efficiency Considerations

File I/O was a major time consideration as the interface was developed. Due to the large quantity of data which had to be read and written, it was decided to read all baseline information into the Input tool once at the start of the project and work from there as opposed to rereading each time the tool was used. Thus, baseline values were saved in a worksheet; as changes were made, the worksheet was updated. The worksheets could be reset to the original values, but, since this was quite time consuming, it was done on request, not as a default activity.

To keep the system as flexible as possible, an indexing system was used between the working data files and the worksheets seen and edited by the user. This allowed the user to sort the worksheets and not have to wait when the data was updated for the simulation.

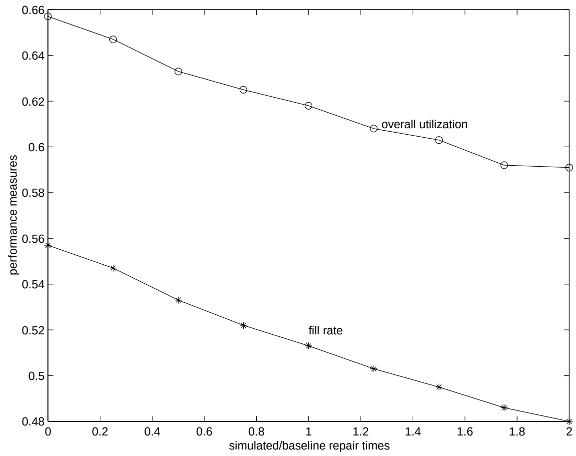

Figure 3: Sensitivity to Repair Times

6 SENSITIVITY AND COST/BENEFIT ANALYSIS

We studied the sensitivity of performance measures to both controllable factors such as repair times and inventory levels as well as uncontrollable factors such as demand rates. In all the experiments reported, a warmup period of 400 days and simulation length of 510 days was used. In Figures 3-5 a point in the x-axis means that the simulation is run when the corresponding parameter is set to that point times its baseline value, for example 0.9 in Figure 4 means that the results are obtained from a simulation run when the demand rates are 0.9 times the baseline rates.

Interestingly, the original perception concerning the importance of the repair process turned out to be incorrect. As can be seen from Figure 3, even in the case when repair is instantaneous, utilization and fill rates did not improve significantly over the baseline. On the average and across all asset types 1% decrease in repair times caused 0.033% increase in utilization and 0.038% increase in the fill rate. This "counter intuitive" phenomenon could be explained by the interplay of equipment age (hence need for repair) and demand. The newer the equipment the more demand it experiences and the less maintenance it needs, the required maintenance after coming off rental is driven by demand and is done quickly. The opposite is true for old equipment; the older an asset, the less demand it experiences and the more maintenance it requires. Since repair is demand driven, equipment that is obsolete and hence not demanded frequently spends a long time in the repair process. Thus, since there was low demand for older equipment, shortening long repair times wouldn't affect utilization significantly.

It is no surprise that among all the factors considered, the performance measures turned out to be most sensitive to changes in demand: 1% increase in demand caused 0.17% increase in utilization and 0.36% decrease in fill rate. Figure 4 displays very clearly the trade-off between utilization and fill rate at a high level of aggregation.

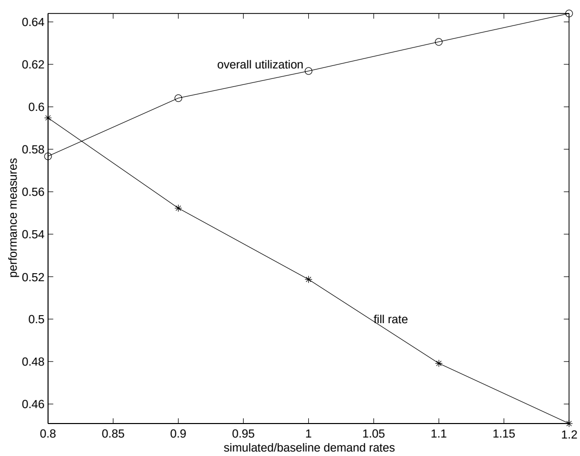

Figure 4: Sensitivity to Demand Rates

Figure 5 displays an aggregate point of view: inventory levels affect the performance measures in a similar but opposite sense as demand rates. This is best demonstrated by quantifying the trade-off between utilization and fill rate: on average 1% increase in the inventory levels caused 0.4% increase in the fill rate and 0.086% decrease in utilization. However, if one chooses the set of asset types to invest in carefully one can increase both of these performance measures. To better understand this phenomenon, during the simulation, we tracked lost revenue due to unavailability of equipment (by equipment type). Then we sorted the asset types by the ratio of their contribution to lost revenue divided by their first costs (original purchasing prices); the larger that ratio, the cheaper the asset is as

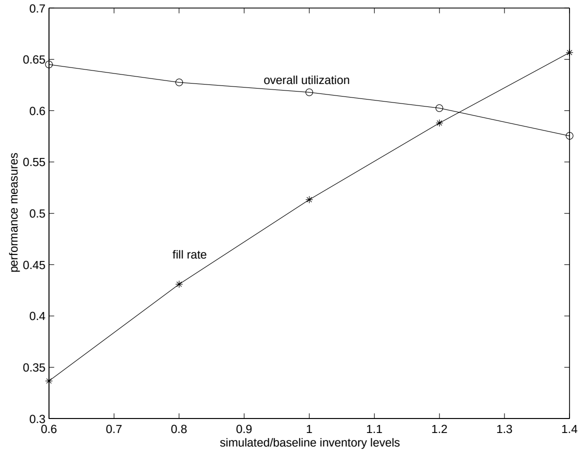

Figure 5: Sensitivity to Inventory Levels

compared to the loss of revenue caused by its lack of availability. After running the simulation with the baseline parameters we increased the inventory levels of the top 1% contributors to lost revenue proportionally to their contributions. In this way we obtained the investment alternative A_1. Then we ran the simulation with the new parameters and obtained a new set of contributors to lost revenue. We iterated in this fashion until we got A_2 through A_4. Investment alternatives B_1 and B_2 are obtained in a similar fashion but this case looking iteratively at the top 0.5% contributors to lost revenue. Figure 6 compares these different alternatives with respect to utilization and fill rates. Then we investigated costs and benefits of these alternatives, the results are reported in Table 1.

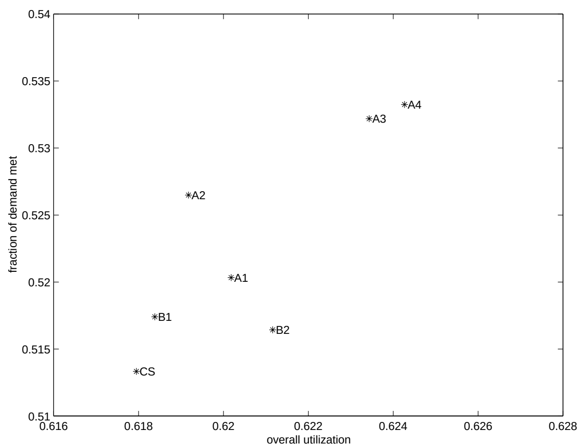

Figure 6: Comparison of Investment Alternatives

In Table 1, CS refers to the alternative called Current Situation representing the baseline, ROI refers to return on investment, and ΔNI refers to change in net income.

Table 1: Comparison of Investment Alternatives

Alternatives	Monthly ROI	Monthly Δ NI
$A_1 - CS$	2.57%	$\$0.40M$
$A_2 - CS$	2.66%	$\$0.68M$
$A_3 - CS$	2.19%	$\$0.95M$
$A_4 - CS$	0.86%	$\$0.44M$
$B_1 - CS$	5.38%	$\$0.40M$
$B_2 - CS$	-1.06%	$\$-0.10M$
$A_1 - B_1$	0.09%	$\$0.01M$
$A_2 - B_1$	1.76%	$\$0.38M$

Both of these measures, ROI and ΔNI, are based on the cash flows produced and additional investment required by those alternatives in the next 17 months. When measuring ROI we assumed that the investment took place in the beginning in one big chunk whereas for ΔNI we assumed that the investment is evenly distributed through the next 17 months. For example, ROI and ΔNI for $A_1 - CS$ are computed by finding the difference between monthly revenues produced under A_1 and CS and determining the additional investment requirements for A_1. Figure 6 and Table 1 together provide a means of comparing these alternatives: for example A_4 seems to be a strong alternative in terms of utilization and fill rates whereas A_3 seems stronger in terms of ΔNI.

One of the important lessons we demonstrated is that simple business strategies such as limiting investment drastically until utilization increases, are likely to fail. There exist profitable investment alternatives that improved utilization. Simulation was instrumental in identifying such investment alternatives and evaluating the current position of the business as well as improving fill rates. In the next section we summarize the work and point out directions for future research.

7 CONCLUSIONS

We have presented our experiences and observations derived from applying simulation modeling to evaluate financial and business decisions for an equipment rental business. Certain aspects of the problem such as size of the asset population and the time grain of the underlying processes led us to construct a discrete time model. To support the model we had to extract model parameters from existing business databases. This was a source of challenges, not the least of which was the classic lack of true demand data. The resulting model provides a useful tool to help management develop improved investment strategies and evaluate impact of decisions on performance measures. The results reported made clear that simple one dimensional business strategies are not likely to achieve their goals without understanding complex interactions within the business. The simulation framework developed also provides a vehicle for better understanding such interactions. This work is being extended to include an improved engine for simulating large populations. Furthermore, the simulation tool is being combined with an optimization engine to address parametric optimization of business policies.

REFERENCES

Grant, E. L., Ireson, W. G., and Leavenworth, R. S. 1990. *Principles of Engineering Economy* Chichester and New York: John Wiley & Sons.

Gürkan, G., Özge, A. Yonca, and Robinson, S. M. 1998. Sample-path solution of stochastic variational inequalities. Accepted by *Mathematical Programming*.

Ho, Y.-C. and Cao, X.-R. 1991. *Perturbation Analysis of Discrete Event Dynamical Systems* Norwell, Massachusetts: Kluwer.

Little, J. D. C., 1961. A Proof for the Queueing Formula: $L = \lambda W$. *Operations Research* 9(3):383–387.

Robinson, S. M. 1996. Analysis of sample-path optimization. *Mathematics of Operations Research* 21: 513-528.

Rubinstein, R. Y., and Shapiro, A. 1993. *Discrete Event Systems: Sensitivity Analysis and Stochastic Optimization by the Score Function Method.* Chichester and New York: John Wiley & Sons.

Shedler, G. S. 1993. *Regenerative Stochastic Simulation* San Diego, CA: Academic Press.

Suri, R. 1989. Perturbation analysis: the state of the art and research issues explained via the GI/G/1 queue. *Proceedings of the IEEE* 77(1): 114-137.

Winston, W. L., 1987. *Operations Research: Applications and Algorithms.* Boston: Duxbury Press.

AUTHOR BIOGRAPHIES

R. ALAN BOWMAN is Associate Professor of Management at the Union College, Schenectady, NY. His research interests are in production and inventory management, simulation, and project management. He is a member of INFORMS. Dr. Bowman has published papers in Operations Research, Management Science and several others.

IRA J. HAIMOWITZ received his Ph.D. in Computer Sciences from M.I.T. in 1994. He is currently employed at Pfizer U.S. Pharmaceuticals, and previously worked for GE Corporate Research and Development. His research interests include data mining, database marketing, and artificial intelligence. He is a member of ACM and AAAI.

ROBERT M. MATTHEYSES received his Ph.D. from Clarkson University in Engineering Science. He has been working at GE Corporate Research & Development Center since 1978 as computer scientist. Over his tenure he has focused on various aspects of optimization with emphasis on problems of network design and analysis, parallel computing architecture, and algorithms for combinatorial problems. He is a member of INFORMS and SIAM.

A. YONCA ÖZGE received her Ph.D. in Industrial Engineering from the University of Wisconsin-Madison in 1997. Since then she has been working at GE Corporate Research & Development Center as operations researcher. Her research interests include analysis of complex stochastic systems by combining simulation and optimization with applications in production systems, option pricing, network design, and economic equilibrium problems. She is a member of INFORMS.

MARY C. PHILLIPS received her MS in Computer Science from Rensselaer Polytechnic Institute in 1986. She has been working as a computer scientist at General Electric Company since 1972. Her work includes a wide range of areas from software environments and compiler design to risk analysis.