Redesigning Check-processing Operations Using Animated Computer Simulation

Rohit Verma

DePaul University, Chicago, Illinois, USA

Gerald D. Gibbs

First Chicago NBD Corporation, Chicago, Illinois, USA

Richard J. Gilgan

First Chicago NBD Corporation, Chicago, Illinois, USA

This paper describes the steps taken by a major commercial bank in the USA to redesign a critical function within its check-processing operation. Animated simulation models of the current and new process were developed to understand the relationship between process parameters waiting times and productivity measures. We describe the animated simulation modeling approach in detail present sample results and provide directions for further use of such an approach in banking.

Introduction

The financial services industry is currently going through a major paradigm shift because of deregulation, cheap and easy access to information technology (e.g. Internet, e-commerce), and because of changing customer needs and preferences. Therefore during the last few months an increasingly large number of commercial and investment banks, insurance companies and financial services brokers have either initiated or proposed a wide variety of new services to their customers. While successful marketing offers a financial services package that appeals to the needs and desires of a particular segment of customers, this effort is futile without the ability to efficiently manage, control and improve the procedures which deliver those services. Therefore, banks and other financial institutions are also spending considerable money, time and effort reevaluating, redesigning, automating and improving the efficiency of their current operations.

In this paper, we describe steps undertaken by one of the largest commercial banks in the USA to upgrade and redesign check-processing operations at their central facility in Chicago. This facility processes approximately three million checks during each 24-hour cycle. The monetary amounts on each check vary from a few dollars to several million and on average a total over $3.0 billion daily. Therefore, the efficiency of this central check-processing operation is of utmost importance to the corporation. In addition, it is essential that check-processing processing complete its daily work in a timely manner so that customer accounts can be posted and on-line balance information updated for branch operations.

We provide an overview of current check-processing operations and describe steps taken to improve the reject repair process (focus of this paper). The current reject repair process is very labor-intensive and has been replaced by an automated process based on high-speed image technology connected to a series of computer workstations. We developed animated simulation models to understand the old and new reject repair processes. We believe that the simulation models have provided very valuable insights about the check- processing operation. For example, we have been able to identify bottlenecks, waiting times and productivity data for various operating scenarios. In this paper, we provide a general description of the simulation modeling approach employed.

Why Use Animated Simulation Models?

Owing to recent advances in computer simulation programs, it is becoming relatively less difficult to graphically model and evaluate alternative process configurations (Bateman et al., 1997, Chase et al., 1998; Fitzsimmons, and Fitzsimmons, 1998). For example, a graphical simulation model of a bank branch can demonstrate the impact of adding one or more tellers, and/or an ATM machine to reduce customer-waiting times. Simulation allows managers to evaluate multiple service designs and perform what-if types of analyses (Law and Kelton, 1991). It can also be used to evaluate the impact of any changes in operating or marketing strategies of service firms (e.g. the impact on checkout queue sizes when marketing promotions are introduced) (MicroAnalysis and Design Software, 1993). Managers can use computer simulation to determine the best way of controlling the flow of customers and materials and to find the most effective way to schedule and deploy resources. Simulation replaces the wasteful and often-unreliable practice of setting management policies based on trial-and-error methods.

Simulation has been used for the past two decades in actual commercial applications and in classrooms. During the DOS and mainframe-computing era, programming languages (e.g. FORTRAN, C, C++) were used to develop simulation models. Generally, programming even a simple model required several hours (often months) of development time. Although very sophisticated and detailed, these programs had either limited or no ability to graphically display the models. During recent years a number of graphical simulation programs have been developed which attempt to reduce model development time. XCELL is an excellent example of one of the first widely-used graphical simulation-modeling programs (Conway et al., 1990).

More recent simulation programs are relatively easy to use, display information visually, and do not require advanced knowledge of computer programming languages. They do, however, require an understanding of simulation modeling concepts, logic and statistics. Fast desktop computers now allow users to run even a very complicated model within a few minutes. Simulation users can visually analyze time series data during the simulation run or view the summary statistics after the run (e.g. queue size in a bank, call center, or Internet service provider).

ServiceModel (PROMODEL Corporation, 1997), a leading simulation program, allows users to design virtually any service process and graphically evaluate performance over time. Users can decide the layout of the service process to be simulated, customer arrival rates (including market segments), the number and schedule of service providers, capacity, resources, and other service attributes. Based on requirements and model assumptions, users can redesign various alternatives and run the model for several hours/days/months of real time within minutes of simulation time. Because of relative ease of development, analysis and visualization, we used ServiceModel to simulate the check reject repair process.

Simulation Illustration: Check-processing Operations

Efficient check processing is essential to the operations of all commercial banks in the USA. Typically, checks written/collected by customers (individuals, businesses, other banks) are deposited at bank branches, ATM machines and other authorized locations. These deposits are transported to a central check- processing throughout the day. An overview of the check-processing operation analyzed in this study is shown in Figure 1.

Deposits contain varying numbers of checks that are either pre-encoded or need manual encoding. Encoding magnetically stores information on the check (e.g. amount, account number, and bank). (This process will be referred to as encoding or Magnetic Ink Character Recognition (MICR) encoding throughout the rest of this article.) After encoding, deposit tickets and checks are placed in trays, separated into batches by header cards and sent to the high-speed sorting operation. High-speed sorters are million dollar machines, which separate

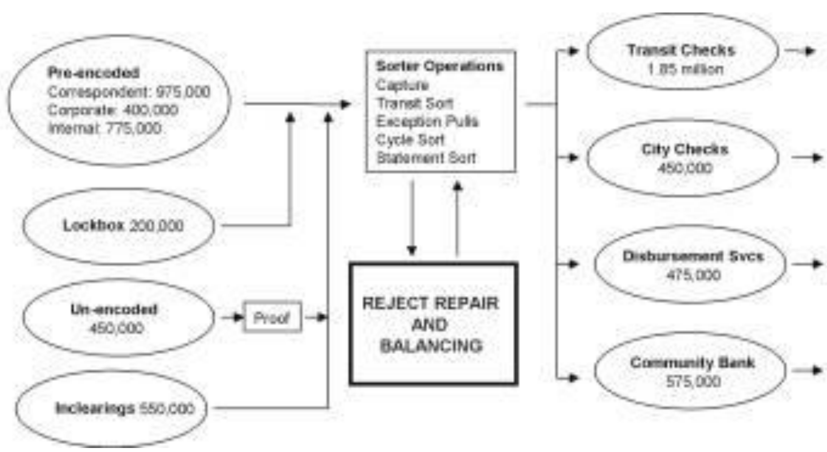

Figure 1. Check-processing workflow schematic

separate checks by different sort keys, e.g. account number, bank routing number, etc. Generally, each check passes through the high-speed sorters multiple times.

During high-speed sorting a number of checks are rejected because they are not readable by the machine. There can be several reasons why the sorter is unable to read a check. For example, the check may have been damaged, or improperly MICR encoded. Rejected checks are combined into batches of varying sizes and sent to the reject repair unit. As mentioned earlier, the reject repair process is the focus of this paper. Figure 2 shows a simplified diagram of the old and new reject repair processes.

The Current Check-reject Repair Process

The current reject repair process is a simple two-step operation. The first step (PREP) involves opening each batch of checks and manually affixing a strip to each check. Up to 20 employees perform this function throughout the day. The checks are then sent to an operator-assisted semi-automatic process (ENCODE - eight parallel stations), which print the MICR encoded information on the newly affixed strip.

Since most of the work in this department was performed manually, processing speed is relatively low which leads to delays and long queues. Additionally, processing speed varied a lot because of productivity differences among employees. The net result was huge backlogs of unrepaired checks would develop frequently in the reject-repair unit.

The Resigned Check-reject Repair Process

Although fast, the new reject check-repair process is more complicated. As displayed in Figure 2, the new process is comprised of four steps. The first step (PREP - three parallel stations) is similar to the old reject repair process but is considerably faster (approximately 1,500 checks per hour). After PREP, each batch of checks along with its control documents is

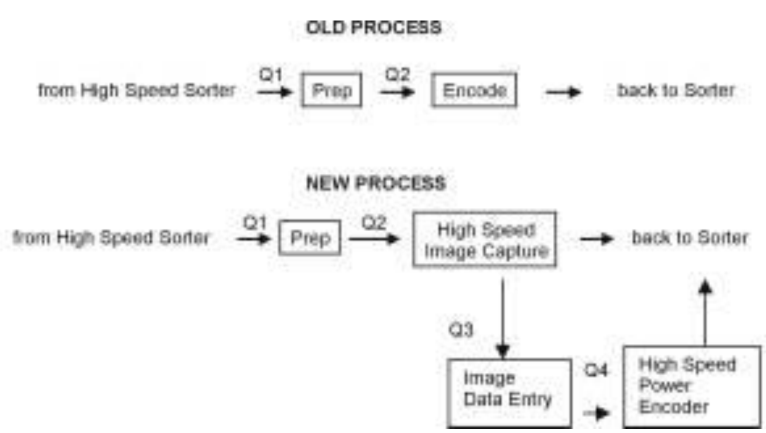

Figure 2. Reject check-repair process

run through one of two high-speed check transports, which creates an electronic image of each check and captures the MICR information on the checks.

During the first pass, checks are read both optically and magnetically. If the system gets an acceptable, it will encode the item and pocket it for reentry into the check-processing system. Because of optical resolution and MICR quality problems, 50 per cent of the checks must be repaired by a data entry operator at an image workstation (12 parallel stations). After being repaired logically, the items, are put back through the high-speed check transports, which encode and pocket them for reentry into the check processing system. In summary, the new reject repair process is relatively more complex and requires several additional steps. At the same time, because of automation and optical character recognition, the new process should be much faster than the old process.

Based on process data collected from the current process (in operation when this study was conducted) and manufacturer's specifications for the new process, we developed two simulation models. The rest of the paper describes the models and results.

Simulation Model Development

It is generally not possible to include all features of a real system in any simulation model. It is also important to realize that a model is only an abstraction of reality. Therefore, models should include all essential and relevant elements of the real system and leave the non-essential elements out. Using these guidelines, and based on real process data, we modeled the current and new check reject-repair processes.

Model Parameters

In order to develop realistic models, we first collected actual process data for several days to estimate various model parameters and input distributions. The mainframe computer tracks of batch sizes and arrival and departure times at various check-processing stations. Therefore, the task of identifying input parameters was reduced to going through several pages of mainframe computer output and recording relevant information for each job (e.g. location, size, arrival times, departure times).

Checks rejected from the high-speed sorter operation are combined into jobs and delivered to the reject repair unit in trays. The number of checks in each job (INBAG) is not fixed. Additionally, check volume at the high-speed sorters (which operates 24 hours a day) also changes throughout the day. However, these distributions were found to be statistically similar for each data collection day. Figures 3 and 4 show typical job sizes and arrival time distributions.

On average, approximately 225 jobs arrive in the reject repair unit each day. The processing time per job at the prep station for the old process was approximately normally distributed with a mean of 129 minutes and a standard deviation of 30 minutes. After prep is complete, checks are recombined into another job with varying numbers of checks per tray. Figure 5 shows the typical job size distribution after the prep operation. The encoding process can process approximately 6,000 checks per hour.

The PREP station for the new process is expected to take only 9.6 minutes per batch with a standard deviation of three minutes whereas the high-speed image capture machine can scan approximately 8000 documents per hour. Each image data entry workstation can process 1,000 checks each hour.

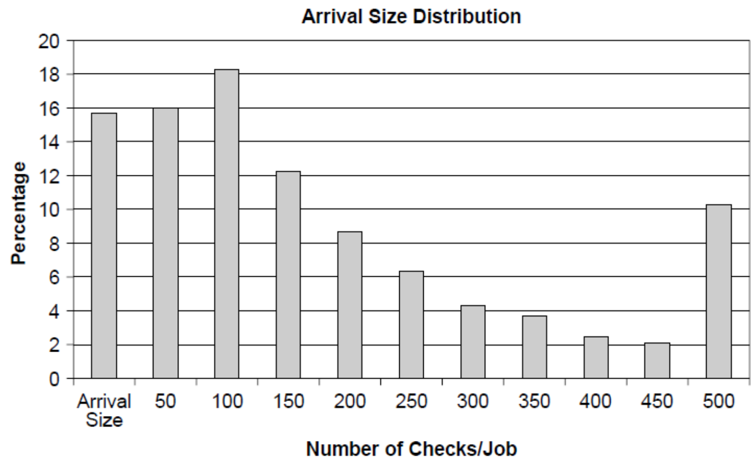

Figure 3. Number of checks per arrival batch

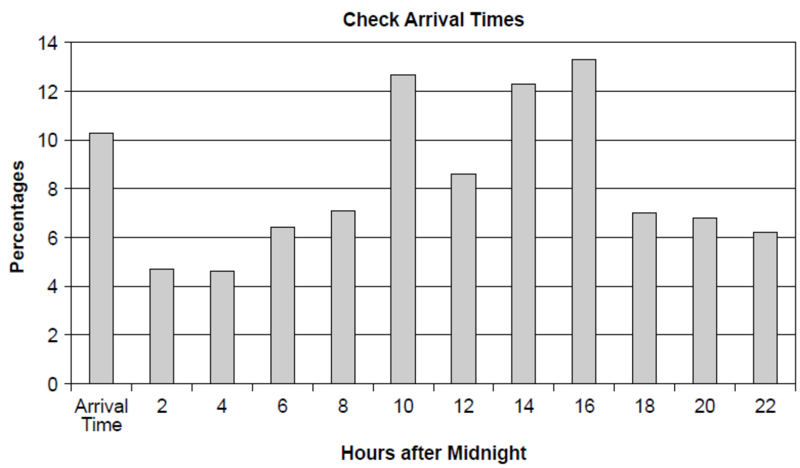

Figure 4. Arrival time distribution

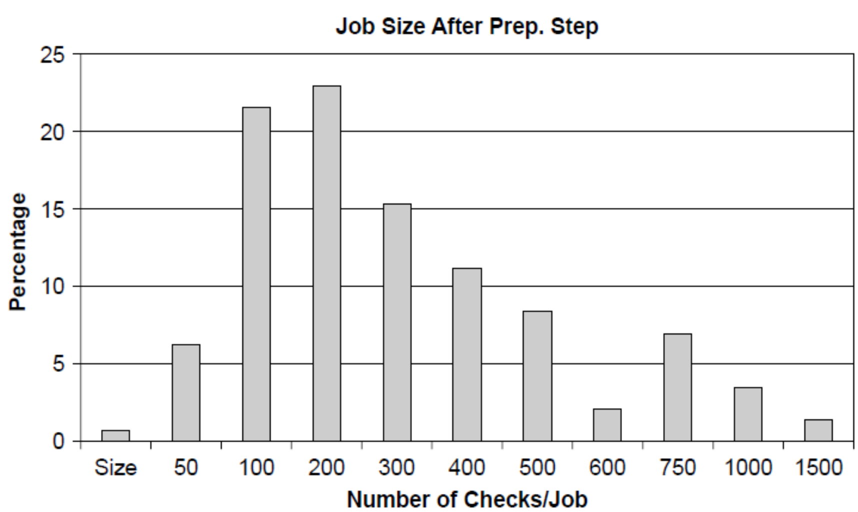

Figure 5. Number of checks per job prior to encoding

Simulation Models

Based on the identified input parameters and process logic (briefly explained) two models were developed using ServiceModel. Figure 6 shows screen-prints of the models and the lists below describe the simulation logic.

Old Reject Check Repair Process: Simplified Simulation Steps

(1) INBAG arrive according to INTIME (Figure 4) distribution. Approximately 225 jobs per 24 hours.

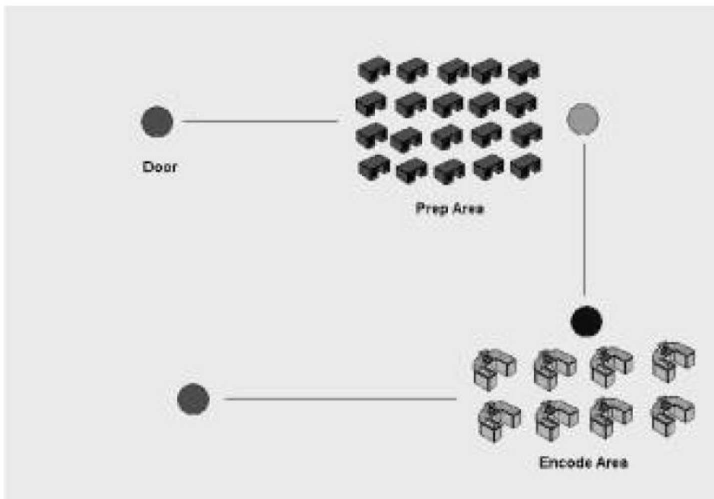

Figure 6a. Old reject check-repair process simulation model

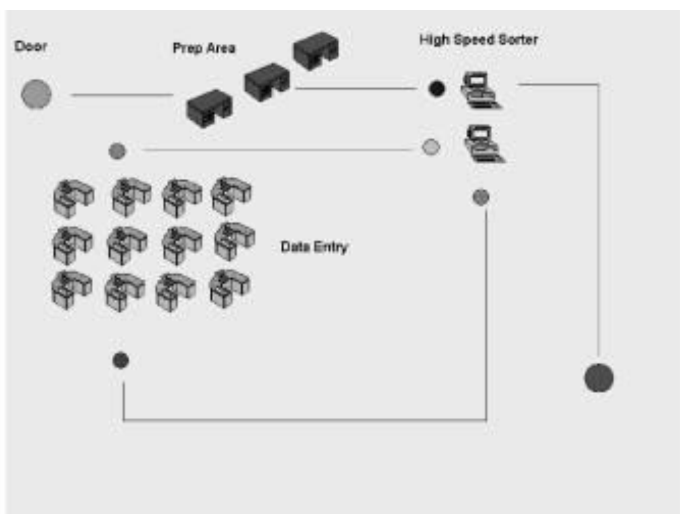

Figure 6b. New reject check-repair process simulation model

(2) INBAG transferred to the queue Ql.
(3) INBAG transferred to available PREP station. INBAG opened as CHECK according to ARRSIZE distribution. Processing time at PREP: Normal distribution with average 129 minutes standard deviation 30 minutes.

(4) Individual CHECK transferred to the queue Q2. CHECK grouped as BOX according to BOXSIZE distribution. BOX transferred to the available ENCODE station.
(5) BOX opened as CHECK. Processing time average 6,000 checks per hour.
(6) Processed CHECK transferred to queue Q3 and then EXIT (back to sorter).

New Reject Check-reject Process: Simplified Simulation Steps

(1) INBAG arrive at DOOR according to INTIME distribution. Approximately 225 jobs per 24 hours
(2) INBAG transferred to the queue Ql.
(3) INBAG transferred to available PREP station. INBAG opened as CHECK according to ARRSIZE distribution. Processing time at PREP: Normal distribution with mean 9.6 minutes. And standard deviation three minutes. One set of 70 documents (PlDOC) added to CHECKS from INBAG.
(4) INBAG + PlDOC transferred to queue Q2 and then to HIGH SPEED IMAGE CAPTURE. Processing speed 8,000 checks or documents per hour.
(5) 50 per cent of CHECKS go to queue Q3 and then EXIT. PlDOC documents EXIT.
(6) 50 per cent of CHECKS are combined into REJECTBAG and transferred to queue Q3, then to the available IMAGE DATA ENTRY station.
(7) REJECTBAG opened as CHECK. A set of 70 documents (P2DOC) created. Processing time for each CHECK: 0.06 minutes.
(8) CHECK combined into PASS2BAG and transferred to HIGH SPEED IMAGE CAPTURE along with P2DOC. Processing speed 8,000 checks or documents per hour.
(9) CHECK and P2DOC transferred to EXIT.

Results

The ServiceModel data analysis module can track time-series data for all entities,

Location	Average Processing Time	Utilization
PREP	122.95 mins	99.58%
ENCODE	0.01 mins.	4.14%

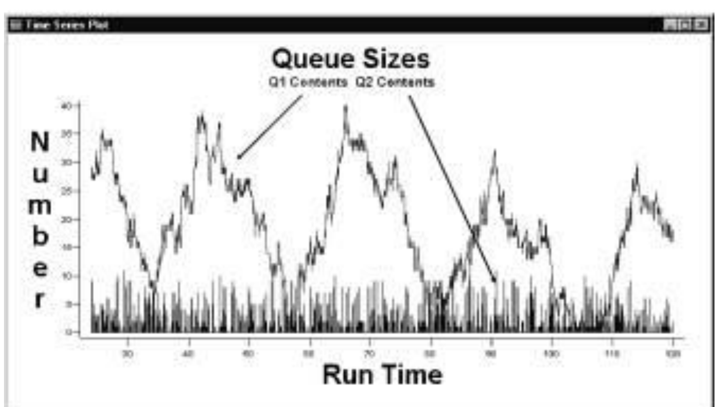

Figure 7. Summarized simulation results for the current process

locations and model parameters. Figures 7 and 8 show summarized results for both the current and new processes. As expected, PREP stations (manual operation) in the current process are being utilized at close to 100 per cent capacity. This leads to increased waiting time at queues Ql and Q2. Figures 7 also shows the waiting time patterns for these two locations. Observations of the actual process produced results very similar to output from the simulation model. There was always a big pile of input jobs before PREP stations, whereas ENCODEing stations were underutilized.

Location	Average Processing Time	Utilization
PREP	9.68 mins	50.67%
IMAGE CAPTURE	0.022 mins.	21.51%
DATA ENTRY	0.03 mins	0.19%

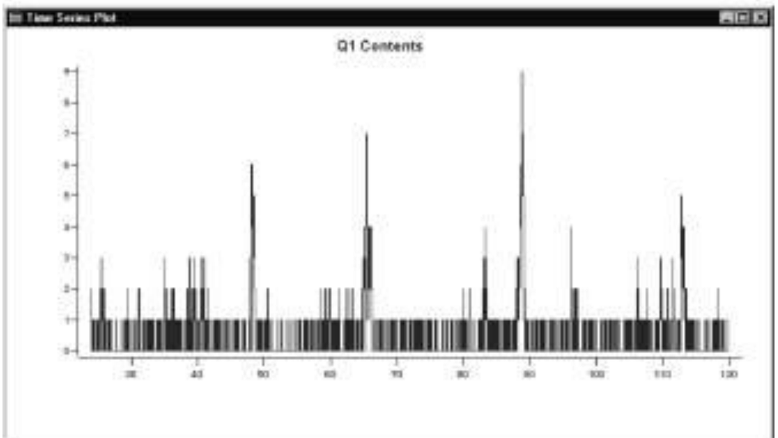

Figure 8. Summarized simulation results for the current process

Summary results for the new reject repair process are presented in Figure 8. Since automated PREP stations are now only utilized approximately half the time, there is almost no queue buildup. Even though 50 per cent of the checks require two passes through the high-speed check transports, these machines are utilized only about one-fifth of the time. The image data entry workstations are utilized at a very minimal level.

Lessons Learned and Future Work

This study demonstrated the value of animated simulation in the design and improvement of complex service processes. Data collection, model development, and analysis for the simulation exercise described in this paper were completed within one month. We successfully modeled two versions of an important banking process. During this project we developed valuable insights about the check reject repair and simulation modeling processes.

When the project started, the simulation exercise was perceived to be easy by one of the authors of this paper. However, this view was found to be completely wrong as soon as the process flows (Figure 1 and 2) and simulation logic were thoroughly analyzed. One especially challenging feature of this simulation exercise was to develop a model where the moving entities (jobs and individual checks) are combined, separated and recombined during the process. Fortunately, ServiceModel contained modules for keeping track of such changes in simulation entities. Otherwise modeling this process would have been extremely difficult and time-consuming.

We believe that developing these two models is the first step in a scientific approach to process improvement. The models have generated several questions, which should be looked at and analyzed for further improvement. For example, fixed and variable costs are not part of the modeling process. Similarly, it is assumed that the numbers of employees at different processing locations stays constant throughout the day/week. Similarly, we have not considered the impact of different labor schedules, or workload distribution changes on various operating parameters.

ServiceModel comes with an optimization and experimental design program SimRunner that can be used to test if changes in certain input parameters impact selected output measures. SimRunner can conduct full or fractional factorial experiments with multiple attributes with user-defined objective functions. The optimized regression models can estimate both the main effects and interaction affects among input variables on the objectives. The estimated equations can be easily incorporated into spreadsheet-based decision support systems for use in day-to-day management and decision making.

In summary, we believe that animated simulation modeling is a very valuable tool for banking and other service industries. Animated models visually display the characteristics of service processes and provide managerially-useful information. In addition, development time using current graphical simulation modeling programs (such as ServiceModel) is also very reasonable (i.e. few days). This makes it possible for business organizations to use such models as day-to-day management tools.

References

Bateman, R.E., Bowden, R.G., Gogg, T.J., Harrell, C.R. and Mott, J.R.A. (1997), *System Improvement Using Simulation*, PROMODEL Corporation, Orem, UT.

Chase, R.B., Aquilano, N.J. and Jacobs, F.R. (1998), *Production and Operations Management: Manufacturing and Services*, 8th ed., Irwin/McGraw-Hill, Boston, MA.

Conway, R., Maxwell, W.L., McClain, J.D., and Worona, S.L. (1990), *XCELL Factory Modeling System*, Version 4.0, 3rd ed., Scientific Press, San Francisco, CA.

Fitzsimmons, J.A. and Fitzsimmons, J.A. (1998), *Service Management: Operations, Strategy and Information Technology*, 2nd ed., Irwin/McGraw-Hill.

Law, A.M. and Kelton, W.D. (1991), *Simulation Modeling and Analysis*, McGraw-Hill, New York, NY.

MicroAnalysis and Design Software, Inc. (1993), Hospital overcrowding solutions are found with simulation, *Industrial Engineering*, December.

Promodel Coroporation (1997), *ServiceModel: The Standard in Service Modeling*, Version 3.5, Software and Users Model, Orem, UT.