

HEALTHCARE SIMULATION MODELING AND OPTIMIZATION USING MEDMODEL

Rochelle N. Price

PROMODEL Corporation
1875 South State Street
Suite 3400
Orem, UT 84097, U.S.A.

Charles R. Harrell

Brigham Young University
Manufacturing Engineering
435 CTB
Provo, UT 84062, U.S.A.

ABSTRACT

In response to the expressed needs of MedModel users, PROMODEL Corporation has made substantial improvements in its MedModel healthcare simulation software. Beginning with its inception in 1993, MedModel was specifically designed to be simple to use and tailorable to the needs of healthcare managers, engineers and clinicians. As such, it provides a basis for the comprehensive evaluation of large, complex problems which are representative of healthcare systems in general. This paper serves as a preliminary examination of some of MedModel's comprehensive features and capabilities. Independent arrivals and scheduled appointments as well as new statements and functions have been carefully designed to solve unique hospital and healthcare specific simulation problems.

1 BACKGROUND

Hospital systems analysts are constantly faced with the difficult challenge of making sense of the various hospital systems they are chartered to study and with recommending courses of action to improve operations. This means they must recognize the types of systems found within a hospital, what each system is actually doing, what causes delays and bottlenecks, which actions are efficient and which are not, and how the adoption of new policies, technologies or changes to existing structures overall will affect that system. The difficulty in arriving at this understanding derives from the very complexity of the systems under study. First, they are human systems subject to the vagaries of human actions. Second, the healthcare process has too many interrelated and highly varied steps involved in even the simplest of healthcare processes to watch them all at once. Accordingly, the analyst has usually made decisions relying on massive databases that record and correlate all manner of clinical and administrative information. Although accurate in recording information on workload, staffing, patient mix, capacity and policy as they relate to cost and quality, the database is

often unable to provide the true correlation among these factors, and the dynamic interaction of each upon the other is usually lost. Moreover, the database approach relies heavily on past performance and is less helpful for predicting the result of anticipated changes to the system.

Finding an analytical tool that can handle the exhibited complexities of healthcare systems has proven to be a daunting task. What has been needed is a tool that allows the investigator to model the gamut of healthcare activities quickly and efficiently, from patient admission to disposition. It is one that would be as effective for modeling a single activity (clinic, emergency department, nursing unit, and radiology) as it would for a linked network of activities (ambulatory care, surgical services, food services, etc.).

Moreover, the problem of finding such a tool actually extends beyond simulation itself since simulation long ago proved its value to the manufacturing sector and has been used to evaluate process problems in healthcare as well. However, pure manufacturing is anything but an accurate reflection of what occurs in a healthcare setting. And, as such, simulators and simulation languages designed with the manufacturing environment in mind rarely contain the constructs and algorithms necessary to handle unique healthcare issues.

2 HEALTHCARE/HOSPITAL NEEDS

Analysis of the hospital environment leads to the recognition that there are five broad categories of systems operating within its walls. Each has its own characteristics, and each challenges the analyst attempting to develop a system model with its unique pattern of behaviors. These broad systems are emergency departments, operating suites, nursing units, ambulatory patient care, and ancillary services.

MedModel's genesis was prompted by the need for a simulation package that could adequately model each of these broad systems and yet be flexible enough to cross between them on models that link areas of more than one system. It needed to have constructs and algorithms that

would assist the analyst in modeling healthcare and hospital systems. These requirements are based upon knowledge of the hospital environment and requests from scores of healthcare professionals. Constructs and mechanisms were developed that allowed for complex pathing, transparent relationships, multiple and repetitive activities and great operational variety. These constructs were deemed crucial to a simulation software designed to solve even the most routine healthcare problems.

More importantly however, developmental concern went beyond simply capturing the essence of a healthcare system. Rather, it focused on ensuring that specific constructs represented precisely what was required to model the system of interest and did so with as much statistical validity as possible. For example, an initial concern for all five hospital systems was that patients and their records often took different operational paths, but they were, in fact, always linked to each other by the very fact that they were all a part of the patient's entire medical "persona". As such, one could not complete a process without the other being a part of the result. When the lab result is delayed, so is the patient's treatment. Accordingly, specifically matching records with patients at various times in a simulation was often critical to the performance of a model.

2.1 Matching Entities

Since the treatment of a patient may well depend on the return of a lab or x-ray result, and since the movement of the test result may depend on the availability of the patient, the ability to match patients and their supporting medical documents is a necessary capability. In the Acute Minor Illness Clinic shown as Figure 1, each patient has had a lab test done and waits in the patient waiting room for the results to be made available. Although one could assume that the longest waiting patient should be given the next available lab result, this is not necessarily the case. It has been shown that assuming that the longest waiting patient must be the next in the line for the next available lab result will yield waiting times that are significantly different statistically from those that correctly match an individual patient with his or her exact lab result.

MedModel easily handles the requirement to keep identically "owned" entities together by using the built-in "matching" construct provided in the software. Shown in Figure 2, the lab result is directly matched with its patient based upon a unique ID established early in the model and automatically transferred as an attribute to any entity "cloned" from the patient.

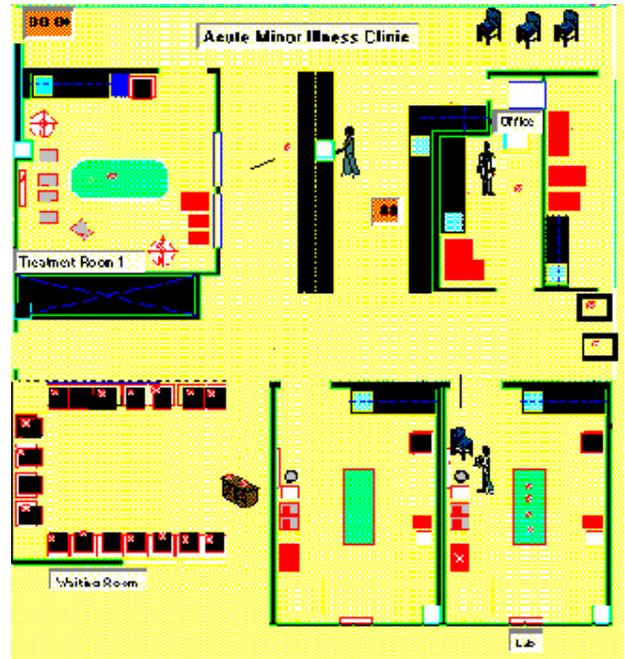


Figure 1: Acute Minor Illness Clinic with Lab

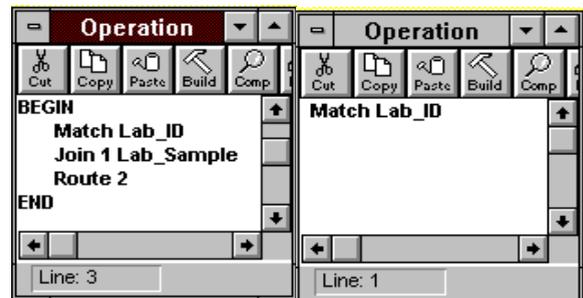


Figure 2: Syntax for Matching Lab Results

2.2 Preempting Locations and Resources

Emergency Department modeling requires, among other items, a sophisticated and versatile capability to preempt rooms that are currently occupied by less acute patients. It also requires a capability to take staff resources from the work they are currently doing and moving them to an acute patient requiring immediate treatment. Also essential is the ability of the staff member to "remember" where he or she was in the treatment of the first patient and to pick up from that point upon return.

In the case of the Emergency Department shown in Figure 3, patients are placed in one of many exam rooms, or they may be placed in a trauma or cardiac room if their condition or lack of other space requires. But if a more serious trauma/cardiac patient should arrive, the room must be made immediately available to the arriving patient.

To handle this situation, MedModel has an entire hierarchy of preemption constructs that have the ability to

preempt locations from the currently occupying patient and take resources from another patient when the new patient's condition requires immediate assistance. MedModel can even set up an entire system of preempting levels so that a patient with an acuity level of 1 will be preempted by a patient of acuity 2, an acuity 3 will preempt the acuity level 2 patient, and so on. By merely filling in the priority level for the capturing of a location or resource and applying the preempting hierarchy levels, MedModel will automatically preempt from one patient to another and remember where in the course of treatment the clinician left the patient. MedModel has preempting process logic to provide for complete control of what happens to the preempted patient rather than limiting the model to program defaults. A preempting process record can be defined to postpone the actual preempting of a location or resource until the current entity explicitly releases it or to route the preempted entity to an alternate location.

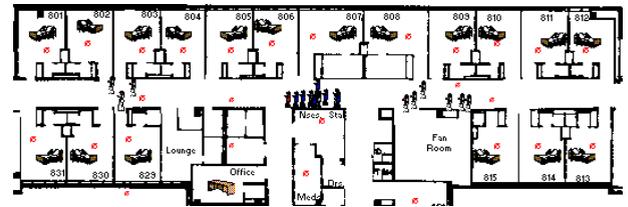


Figure 4: Nursing Unit

MedModel allows the modeler to copy one set of logic statements from one room to another, yet an even better feature is the ability to write the logic once as a subroutine or macro and then just use the name of the subroutine or macro whenever the modeler wishes its logic to be used.

For example, a patient who is feeble or comatose requires the nursing staff to roll the patient every four hours. This takes two staff members to accomplish the task but is only done for patients who can't turn themselves. Since it isn't known what type of patient may be in a particular room, MedModel simply allows the modeler to write the logic to a subroutine. And since many activities in a nursing unit are time-based rather than activity-based, the subroutine can also be run independently of other activities that are occurring in the model. The subroutine logic for the scenario described above could look like that shown in Figure 5. Notice that the subroutine can also call a macro from within the subroutine logic.

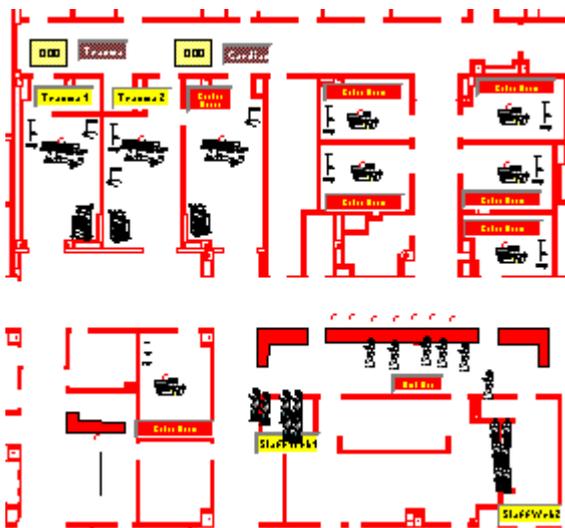


Figure 3: Emergency Department

2.3 Multi-Use Subroutines and Macros

Healthcare in general and hospital systems in particular are characterized by many locations capable of having the same activity performed in each area. Non-specialized operating room suites, clinic exam rooms and inpatient rooms are a few examples. A quick and effective way to model similar activities at different locations is essential to any simulation software professing to emulate healthcare activities accurately. Not only must the simulation software be quick to use, but it must be able to recall repetitive activities. In the nursing unit model depicted in Figure 4, all the patient rooms are identical and can house any of the patients assigned to the unit. Therefore, similar activities such as medication administration, meal serving, etc. can be handled as the same set of process logic.

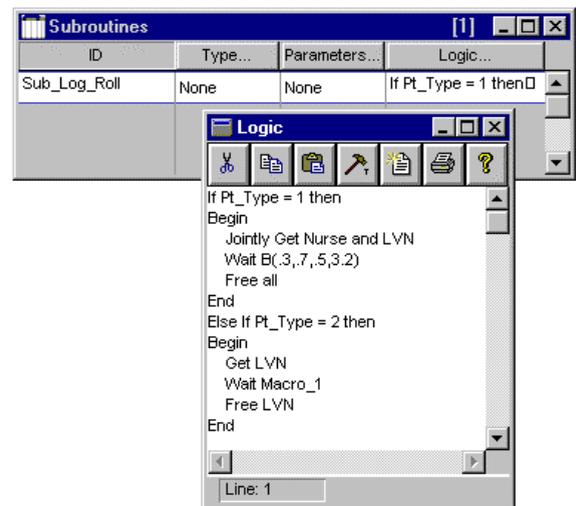


Figure 5: Subroutine and Macro Usage

2.4 Shifts

Certainly not the least of the problems handled deftly by MedModel concerns the management of resources and their accurate arrival and departure times. Nearly all healthcare workers are assigned to recurring shifts. Activity planning is often based on the knowledge of who will be at work during certain hours of the day. In fact, task responsibility changes from one hospital worker to another

when certain shifts do not have a particular type of resource assigned. MedModel allows the modeler to create any number of shifts of any length or days of the week, along with break periods to represent breaks or meals (Figure 6). Single or multiple resources may be assigned to any shift. Locations may also be assigned to shifts, which facilitates closing certain areas of the hospital during certain hours of the day. A resource may work the day shift one week and the night shift the following week.

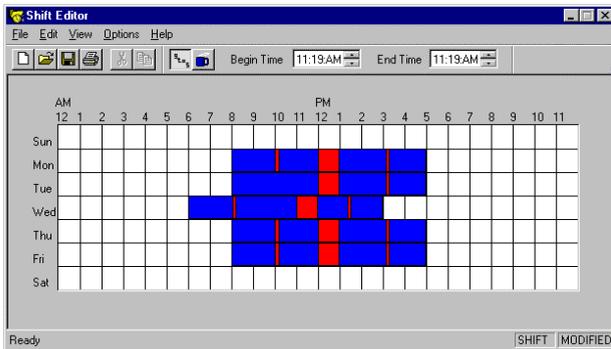


Figure 6: Shift Editor

2.5 "Point and Click" Approach

Because MedModel was written under Windows, model definition depends, to a large extent, on nothing more than using a mouse to identify, select or place appropriate components of the model. The Logic builder is comprehensive and powerful tool which helps the modeler create valid logic and expressions. Since it is fully menu-driven, the modeler need only "click" on an appropriate field to select an element for entry of information. In essence, a model can be built in its entirety simply by placing representative icons on the screen and then selecting different descriptive fields to define movement, relationships and activities between and among model entities and locations.

2.6 Custom Icons

MedModel also comes with an impressive library of colorful, pre-designed healthcare icons representing hospital-specific patients, staff members, materials, treatment fixtures, instruments, used surgical trays, baby bassinets and the like. MedModel is also accompanied by an icon editor that enables the design of any manner of icon desired using an almost limitless array of colors and shapes (Figure 7). The modeler may use the icons provided in the Graphics Library, or draw new icons using the Graphic Editor. MedModel allows for the import of Windows metafiles or .PCX clipart to be used as graphics too. The value of this capability transcends simple esthetics and allows the modeler to represent the events of the hospital environment with accuracy.

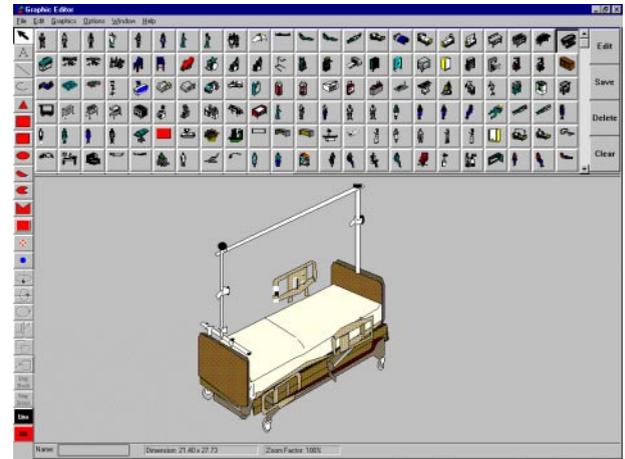


Figure 7: Graphic Editor

2.7 Automatic Processing and Path Entries

Unlike the majority of simulation software packages, MedModel presents the modeler with the capability to design and construct a model using little more than the mouse. This is especially useful when identifying movement and processing steps. In this case, the modeler only needs to click on the succession of locations to which a patient may move, and the required movement entries are made automatically. This holds true for both resource (i.e. doctor, nurse, technician, etc.) movement as well as processing. To create the required path for a clinical patient for example, the modeler needs only to draw a line from the clinic entry to the different locations the patient might go. In this manner, people can be made to walk down hospital corridors, through doors, up stairs and the like. Paths may be bi-directional or uni-directional. MedModel automatically enters all distances required to complete the path and calculates correct movement times based on the distance and speed of the moving entity or resource.

2.8 In General

What has been addressed above is only a small part of what MedModel is capable of doing. Because the modeling of healthcare processes is almost always characterized by the need for complex process logic due to the seemingly endless variety of patients and activities, MedModel is equipped with an impressive collection of pre-programmed constructs that handle the vast majority of related problems. These constructs handle issues like the simultaneous but conditional use of different members of a healthcare team or the requirement to preempt certain medical activities when higher priorities come along. This makes it easier for the modeler to concentrate on modeling rather than on developing complex expressions to represent common activities.

What makes MedModel unique then are the features that it calls on to make healthcare modeling far easier.

Coupled with responsive and versatile animation, MedModel provides a basis not only for rapid model building but ease of validation as well. This use of animation enhances the modeler's ability to validate the modeled system and helps communicate the system behavior to others.

3 NEW FEATURES

MedModel 4.2's Dynamic Plot feature (Figure 8) enables graphical observation of statistical information during runtime, as well as the ability to export this information to an Excel spreadsheet. This new version also now features extensive import and export capabilities for the handling of model data. Excel spreadsheet data can be read directly into arrays, and array values can be directly saved as spreadsheet files. Additionally, statistical data can be saved directly as an Excel file.

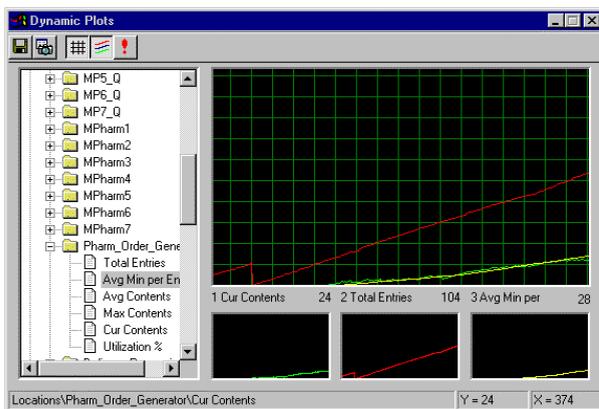


Figure 8: Dynamic Plots

4 OPTIMIZATION

SimRunner Optimization is an add-on capability for MedModel simulation software. SimRunner turns the simulation model into a solution machine by giving the best possible answer in the least amount of time. The optimization process takes an existing MedModel simulation model and performs sophisticated "what-if" analysis and optimization automatically.

For each optimization project, the modeler tells SimRunner which model to analyze/optimize, which input factors to change, and how to measure system simulation performance. Three types of projects may be run with SimRunner: Statistical Advantage, Design of Experiments, and Optimization.

Statistical Advantage helps select a warm-up length, run length, and the number of replications to make the model statistically more valid. Design of Experiments is a tool to help determine whether or not changing a given input factor affects the objective function, and the significance of the effects. Optimization is a multi-variable evaluation that tries various combinations of input factors

to arrive at the combination that provides the best objective function value.

The SimRunner Output report generates three types of data and reporting:

1. Data reports for import into spreadsheets
2. Analysis reports for text and word processing reports
3. Charts for graphic display of results. Each of these can be printed or exported to other applications using common Windows clipboard functions.

5 CONCLUSIONS

MedModel is one of the most powerful tools available for analysis of the healthcare environment. It represents an opportunity for all elements of the healthcare sector to get involved with analytical tools that are far simpler, easier to use and more powerful than those previously available. Of greater significance however, is that due to the comprehensive nature of the tools, assumptions and short cuts that have routinely characterized healthcare and hospital simulations are no longer necessary. Rather, it is now possible to model complex healthcare systems accurately and with confidence in the results.

AUTHOR BIOGRAPHIES

ROCHELLE N. PRICE is an Application Engineer with PROMODEL Corporation. Her responsibilities include consulting services, technical support, technique support, and product training. Ms. Price has a B.S. degree in Design Engineering Technology and an M.S. degree in Manufacturing Engineering from Brigham Young University. Prior to joining PROMODEL, she worked as a Tool Engineer and as a Manufacturing Engineer for the Boeing Company, using simulation for capital equipment justification and process evaluations. Her professional affiliations include the Society of Women Engineers, the Institute of Industrial Engineers and the Society for the Advancement of Material and Process Engineering.

CHARLES R. HARRELL is an Associate Professor of Manufacturing Engineering at Brigham Young University, and founder and chairman of PROMODEL Corporation in Orem, Utah. Dr. Harrell received his B.S. in Manufacturing Engineering Technology from Brigham Young University; M.S. in Industrial Engineering from the University of Utah; and Ph.D. in Manufacturing Engineering from the Technical University of Denmark. Prior to forming PROMODEL, he worked in simulation and systems design for Ford Motor Company and Eaton Kenway Corporation. Dr. Harrell is a senior member of IIE and SME.