Proceedings of the 1993 Winter Simulation Conference
G.W. Evans, M. Mollaghasemi, E.C. Russell, W.E. Biles (eds.)

SIMULATION ANALYSIS USING SIMSTAT 2.0

Warren E. Blaisdell

MC[2] Analysis Systems
667 Pinewoods Ave.
Troy, NY 12180, U.S.A.

Jorge Haddock

Department of Decision Sciences & Engineering Systems
Rensselaer Polytechnic Institute
Troy, NY 12180, U.S.A.

ABSTRACT

SIMSTAT 2.0 is an interactive graphical software tool that performs statistical analysis on simulation input and output data. It is designed to work seamlessly with most simulation packages, such as GPSS/H, SIGMA, SIMAN, SLAMSYSTEM and ProModelPC. Simulation analysis is an essential part of every simulation study, but is often neglected or under-used because there has been no easy way to accomplish it. SIMSTAT 2.0 bridges the technology gap by providing essential analysis techniques in a user-friendly environment.

1 INTRODUCTION

Most discrete event simulations are based on modeling the random behavior of systems. Because of the random nature of the input data, the output data is also random. Thus, due to this stochastic nature of the input and output variables, the simulation can be considered a sampling experiment where the simulation output values are observed for a given set of random numbers. Therefore, it is essential to employ valid statistical methods in order to reach legitimate conclusions.

In general, the purpose of simulation input analysis is to determine appropriate sampling distributions for the random input data; the purpose of simulation output analysis is to determine point and interval estimates for one or more system parameters. The resulting information can then be used to compare several system configurations to determine the best.

2 OVERVIEW OF SIMSTAT

2.1 The Environment

SIMSTAT features easy-to-use pull down menus and is fully integrated into the Windows™ 3.x environment. Data is maintained in a spreadsheet format for editing, examination and analysis. SIMSTAT takes advantage of the Windows™ Clipboard to allow the transferring of data and graphics directly to other Windows™-based word processing, spreadsheet, and graphics packages. SIMSTAT takes full advantage of the extended memory and memory swapping features of Windows™. Printing can be accomplished with a large variety of dot matrix, laser, ink jet, and color printers, as well as plotters. Additionally, SIMSTAT has an extensive on-line help facility to provide quick answers to questions that may arise.

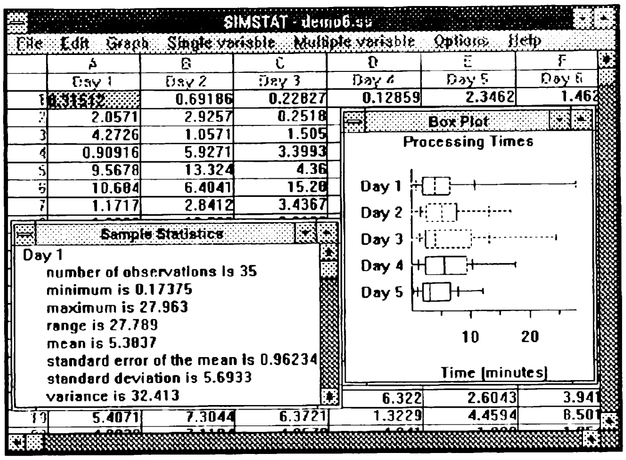

Figure 1: SIMSTAT 2.0

2.2 The Data

SIMSTAT has no fixed limit on the amount of data that can be held in a spreadsheet, rather, the limit is only determined by the memory configuration of your computer system.

Data can be "marked" - to allow a versatile analysis to be done. This is beneficial to iteratively examine and determine the initialization phase of a system. Data can also be cut and pasted to and from different columns in the spreadsheet. This functionality is useful when the simulation does not yield its output data in an organized fashion.

Data may be entered directly, or can be easily imported from many simulation packages.

2.2.1 Importing from GPSS/H

SIMSTAT can import files created by a GPSS/H model by using the BPUTPIC command to create a data file with each line having the format:

Variable Name: *data value*

The data value can be formatted by using asterisks ("*") to define editing fields. For example, to collect output from two variables, one BPUTPIC statement

```
BPUTPIC (X1)
Waiting Time:  ****.***
```

and the second BPUTPIC statement located elsewhere in the model

```
BPUTPIC (X2)
Processing Time:  **** ***
```

could generate the following output:

```
Waiting Time:  78.539
Waiting Time:  28.900
Processing Time:  1211.280
Waiting Time:  41.607
Processing Time:  953.203
Processing Time:  1086.877
Processing Time:  804.729
Waiting Time:  9.545
```

Please refer to Henriksen and Crain (1989) for more information on using the BPUTPIC command to create a data file.

2.2.2 Importing from SIGMA

SIMSTAT can import SIGMA output files. All numeric columns will be imported, and the variable names that appear in the output file will also be imported. For more information on generating SIGMA output files, see Schruben (1992).

2.2.3 Importing from SIMAN

SIMSTAT can import binary data files created by a SIMAN model. There is no need to format the file using the EXPORT command in the Output Processor. Please refer to Pegden (1987) for further information on creating a data file.

2.2.4 Importing from SLAMSYSTEM

SLAMSYSTEM plot files are unformatted binary files generated by RECORD and VAR statements. For each RECORD statement there is a file whose first record has the plotting symbols and labels for up to 10 dependent variables (defined on VAR statements) plus the independent variable label. For example, to save the time in system data, use the control statements:

```
RECORD (1),TNOW,TIME,50;
VAR,XDAT,*,TSYS;
```

and the network statements:

```
ASSIGN,XDAT=TNOW-ATRIB(1);
GPLOT(1);
```

with the arrival time in ATRIB(1). SIMSTAT will automatically determine how many variables are in the plot file and import the data and the variable names. For more information on generating SLAMSYSTEM plot files, see Pritsker (1987).

2.2.5 Importing from ProModelPC

SIMSTAT can read in duration reports created by a ProModelPC simulation. Unique variable names are formed by combining the name of the Part with the name of the Resource and either 'start' (the entry time), 'end' (the departure time), or 'dur' (the elapsed time). For example, a Duration Report could generate three variables with the names: 'ALL LOG1 START', 'ALL LOG1 END', and 'ALL LOG1 DUR'. Please refer to Production Modeling Corporation of Utah's *ProModelPC User's Manual* for further information on creating a data file containing Duration Reports.

2.2.6 Importing from other Simulation Packages

SIMSTAT can import data from other simulation packages that can generate data files in ASCII text format. In text file format each variable of the data is in a vertical column. When this method is used, SIMSTAT automatically names the variables beginning with 'Variable 1'. Please refer to Blaisdell (1991) for more details on importing text files.

2.3 Graphical Analysis

SIMSTAT has many graphics capabilities. To view the data, simple scatter plots are available, as well as cumulative average plots and moving average plots. To assist in viewing the distribution of the data, SIMSTAT provides probability plots (for normal, exponential, and uniform distributions), frequency and cumulative frequency histograms, pie charts, box plots, as well as overlapping histograms of the data with theoretical distributions (for Beta, exponential Gamma, lognormal,

normal, triangular, uniform, and Weibull distributions), Also, SIMSTAT has bar charts for comparing multiple system designs, autocorrelation plots for checking independence, and Welch's method for graphically evaluating the warm-up phase.

2.4 Statistical Analysis

SIMSTAT calculates the sample statistics: mean, variance, minimum, maximum, range, standard error of the mean, standard deviation, coefficient of variation, skewness, and kurtosis. The hypothesis testing capabilities allow the testing of the mean and variance of one or two variables. Also included is the ability to perform a multiple means comparison using Fisher's Least Significant Difference method. SIMSTAT has six methods of calculating confidence intervals: replication, batch means, regenerative, spectral, autoregressive, and standardized time series. Finally, SIMSTAT includes two methods for performing variance reduction: antithetic variates and common random numbers. For further information on the technical details of these procedures, see Law and Kelton (1991) and Blaisdell (1991)

3. THE SIMULATION PROCESS

The best approach to understanding the power and flexibility of SIMSTAT is to see how it can be integrated into the simulation process. Although there is no fixed structure for the simulation process, the following will be used: 1. Define Problem, 2. Collect Data, 3. Create Model, 4. Verify/Debug Model, 5. Validate Model, 6. Analyze Model, 7. Present Results. For the sake of completeness, each step will include a brief description.

3.1 Define Problem

The first stage in every simulation study is to concisely define the problem, and can be accomplished in three parts. First, this involves a clear understanding of the system to be modeled as well as the bounds of the alternative systems to be evaluated. Second, there must be a determination of key system parameters, which are to be used as a basis for evaluation. Finally, it must be decided that the technique of simulation is indeed the best approach to the solution of this problem.

3.2 Collect Data

This step could easily be the most time consuming effort in the entire project. In order to accurately model any system, a thorough knowledge of all system parameters must be obtained. This may involve long periods of simply observing the system and recording what is seen. This will aid in recognizing the essential and non-essential portions of the system, and eventually in deciding which functions will need to be included in the model.

3.3 Create Model

The model creation step can be broken into two parts. First, the data that has been collected must be converted into probability distributions Then, the model must be written in some simulation language such as GPSS/H, or perhaps just in a programming language such as FORTRAN or C. Any alternative systems must also be codified.

SIMSTAT can be used to fit the data to theoretical probability distributions (Beta, exponential, Gamma, lognormal, normal, triangular, uniform and Weibull). Figure 2 shows overlapping histograms of data that were fit to a normal distribution using maximum likelihood estimation. Once the distribution has been fitted, the goodness of fit can be tested using the chi-squared or the Kolmogorov-Smirnov test feature. SIMSTAT can also be used to generate empirical distributions when theoretical distributions are not appropriate. For the case when data collection is not possible, SIMSTAT also provides functionality to graph up to ten overlapping probability distributions to aid in visual distribution selection.

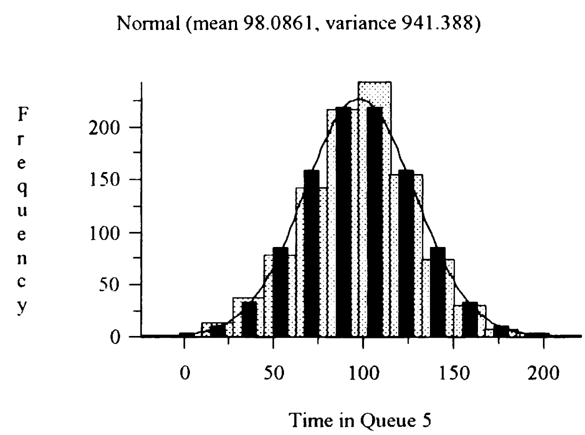

Figure 2: Fitting Distributions to Data in SIMSTAT

3.4 Verify/Debug Model

This is the tedious process of making sure that no typographic, syntax, and logic errors have been made in the model.

3.5 Validate Model

This step may be closely tied to the Model Verification step. Additionally, there should be an evaluation of some trial runs of the model to make sure that model is an adequate representation of the actual system. SIMSTAT can be used to test how appropriately the random samples used in the simulation fit the data that was collected. SIMSTAT can also be used to perform some hypothesis testing on the mean and variance of the key output variables. The data that was collected should be closely reflected in the behavior of the model. For a more in-depth study of model validation, see Sargent (1991).

3.6 Analyze Model

This step is essential. If the model is not analyzed properly, invalid conclusions may result. The simulationist must decide on the number of runs of the model and on the alternative systems as well as the length of each run. For a more thorough examination of the analysis process see Seila (1991) and Law (1983).

To illustrate the use of SIMSTAT for this step, we will limit our analysis to two examples. Figure 3 shows the processing time data from a simulation of 1000 parts and five runs displayed using Welch's method with a moving average window of 15. We can visually eliminate an initialization period of the first 200 parts. However, if we additionally wish to test this using the Schruben, Singh & Tierney (1983) test, we find that at the 90% confidence level there is still initialization bias present after the deletion of the first 200 and 300 parts. Not until 400 parts are deleted does the bias disappear (at the 90% level). Therefore, it may be necessary to run the simulation for more than 1000 parts to be sure that steady state has indeed been reached.

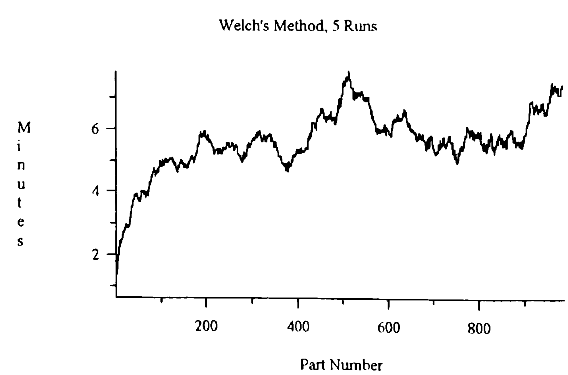

Figure 3: Welch's Method for Determining Initialization Bias in SIMSTAT

Figure 4 shows the strong autocorrelation structure that may exist in simulation output data, especially in queueing models. The data was batched into batches of size 15. The resulting autocorrelation structure of the batched data is shown in Figure 5. It is obvious that batching the data has had a significant effect on the autocorrelation. SIMSTAT also has the capability to use the Fishman procedure for determining the optimal batch size.

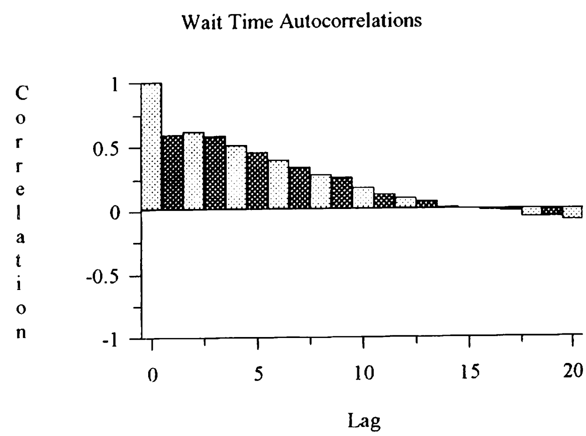

Figure 4: SIMSTAT Autocorrelation Plot for Wait Time Data

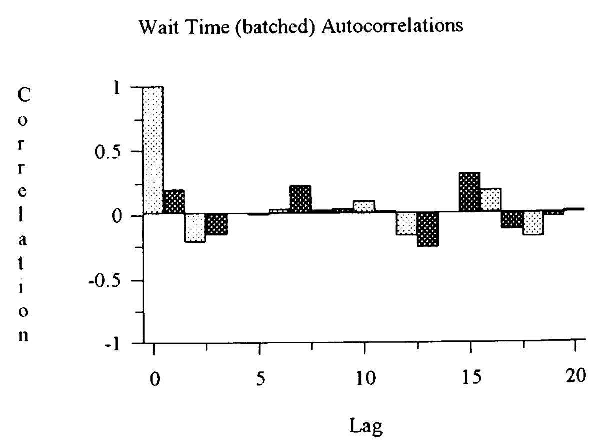

Figure 5: SIMSTAT Autocorrelation Plot for Batched Wait Time Data

3.7 Present Results

Once the analysis is completed, a report must be written to present the conclusions of the study. If a Windows™-based word processing program is used (such as Word for Windows™) to write the report, each graphical and statistical output window in SIMSTAT can be copied to the Windows™ Clipboard and from there be pasted directly into the report. This makes the report writing quite efficient.

4 SUMMARY

Proper analysis is an essential facet of every valid simulation study. SIMSTAT 2.0 is an easy to use graphical and statistical tool designed to work together with most simulation languages to provide the necessary analysis. SIMSTAT 2.0 is also designed to work together with Windows™-based word processing programs to generate effective simulation summaries.

5 THE FUTURE

SIMSTAT is constantly being updated to include the recent trends in simulation analysis. Some of the features that are currently being planned for future versions of SIMSTAT include: experimental design, response surface methods, nonparametric hypothesis testing and confidence intervals for mean, variance and independence, and simulation optimization.

REFERENCES

Banks, J. and J. S. Carson, II (1984). *Discrete-Event System Simulation*, Prentice-Hall, Inc., Englewood Cliffs, NJ.

Blaisdell, W. E. (1991). *SIMSTAT for Windows 3.0 User's Manual*. MC^2 Analysis Systems, Troy, NY

Bratley, P., B. L. Fox and L. E. Schrage (1983). *A Guide to Simulation*, Springer-Verlag, NY.

Henriksen, J. O. and R. C. Crain (1989). *GPSS/H Reference Manual*, 3rd ed., Wolverine Software Corporation, Annandale, VA.

Law, A. M. (1983). Statistical Analysis of Simulation Output Data. *Operations Research*, 31, pp. 983-1029.

Law, A. M. and W. D. Kelton (1991). *Simulation Modeling and Analysis*, 2nd ed., McGraw-Hill, NY.

Pegden, C. D. (1987). *Introduction to SIMAN*, Systems Modeling Corporation, State College, PA.

Pritsker, A. A. B. (1986). *Introduction to Simulation and SLAM II*, 3rd ed., Halsted Press, NY.

Sargent, R. G. (1991). Simulation Model Verification and Validation. *Proceedings of the 1991 Winter Simulation Conference*, pp. 37-47.

Schruben, L. W. (1987). Using Simulation to Solve Problems: A Tutorial on the Analysis of Simulation Output. *Proceedings of the 1987 Winter Simulation Conference*

Schruben, L. W. (1992). SIGMA: *Graphical Simulation Modeling*, The Scientific Press, San Francisco

Schruben, L. W., H. Singh, and L. Tierney (1983). Optimal Tests for Initialization Bias in Simulation Output. *Operations Research*, 31, pp. 1167-1178.

Seila, A. F. (1991). Output Analysis for Simulation. *Proceedings of the 1991 Winter Simulation Conference*, pp. 28-36.

AUTHOR BIOGRAPHIES

WARREN E. BLAISDELL is a partner at MC^2 Analysis Systems and is in charge of the development of SIMSTAT. He received a B.S. in mathematics from Worcester Polytechnic Institute, and he received a M.S. in operations research and statistics from Rensselaer Polytechnic Institute. His areas of research include statistical computing, simulation analysis, and regenerative sampling.

JORGE HADDOCK is an associate professor of Industrial Engineering and Operations Research in the Department of Decision Sciences and Engineering Systems at Rensselaer Polytechnic Institute. He holds a BSCE from the University of Puerto Rico, a MSMgtE from Rensselaer, and a PhD in Industrial Engineering from Purdue University. His primary research interests involve modeling of manufacturing/production and inventory control systems, as well as the design and implementation of simulation modeling and analysis tools. He received the Outstanding Young Industrial Engineer Award from IIE in 1990 and the Martin Luther King, Jr. Faculty/Staff Award at Rensselaer in 1992.