UltraSAN VERSION 3 OVERVIEW*

D. D. Deavours, W. D. Obal II†, M. A. Qureshi, W. H. Sanders, and A. P. A. van Moorsel

Center for Reliable and High-Performance Computing
Coordinated Science Laboratory
University of Illinois at Urbana-Champaign
Urbana, IL 61801
{deavours, qureshi, whs, moorsel}@crhc.uiuc.edu
world wide web: http://www.crhc.uiuc.edu/UltraSAN

† Department of Electrical and Computer Engineering
University of Arizona
Tucson, AZ 85721
obal@ece.arizona.edu

Introduction *UltraSAN* [1, 2, 3] is a software package for model-based evaluation of systems represented as stochastic activity networks (SANs), e.g., [4]. The software has been implemented in a modular fashion, with clearly delineated interfaces between model specification, construction, and solution. *UltraSAN* offers an X Windows-based user interface and both analytical and simulation solution modules for transient and steady-state performance, dependability, and performability measures. Furthermore, the tool facilitates graphical representation of the obtained results by its report generator. This paper gives a very brief overview and pointers to more detailed descriptions of the software.

User interface and model specification The X Windows-based user interface of *UltraSAN* consists of several modules. First, the control panel serves to unify the rest of the modules under a single interface, expediting the process of running model specification, construction, and solution modules. In addition, the control panel provides access to *UltraSAN*'s automatic model and result documentation and several other useful utilities.

The SAN editor (Figure 1) is a graphical user interface used to draw and define SANs, that can then be replicated and joined together using the composed model editor (also a graphical editor). The performability variable editor is used to specify reward variables representing the performance measures of interest (a discussion of measures that can be expressed as reward variables can be found in [5]).

During the specification phase, global variables may be incorporated throughout the model and reward definition. A so-called study editor then is used to assign values to the global variables used to parameterize the model specification. The resulting experiments can be organized into logical groups called studies. For example, a study may consist of a set of experiments in which only one of several global variables changes.

Model construction When one or more studies have been defined, an executable model is constructed, by compiling and linking solution library code with model description code generated by the user interface modules. If analytic solution is intended, *UltraSAN* constructs the reduced base model generator, an executable program that, when run, creates a machine readable description of the Markov process underlying the model. The algorithms for construction of the reduced base model are detailed in [6]. Reduced base model construction takes advantage of symmetries in the composed model and properties of the specified reward variables to directly generate a smaller lumped state space sufficient for obtaining the desired results.

If simulation is chosen, *UltraSAN* creates an executable simulation by linking the model description modules with a SAN simulation library. The event scheduling in the simulation algorithms also take advantage of composed model symmetry [7].

*This work was supported in part by IBM, Motorola Satellite Communications, and US West Advanced Technologies.

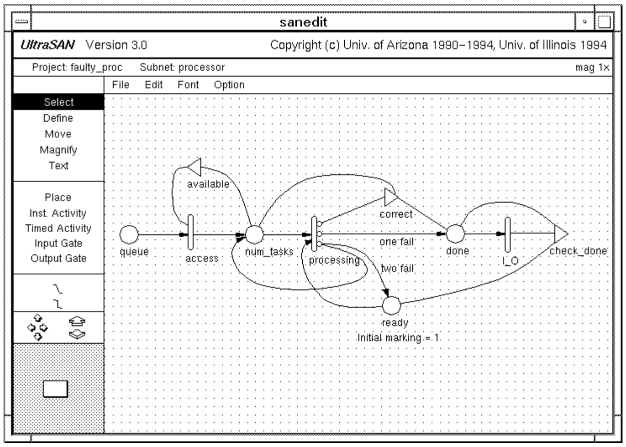

Figure 1: SAN editor.

Analytical solution After the construction of the stochastic process, a solver is chosen according to the measure of interest. For steady-state results of the Markov process, two solvers are available, a direct solver for relatively small models, and an iterative solver for large models up to several hundred thousand states. An analytic solver is also available for the steady-state solution of certain models with deterministic delays. All steady-state solvers report the mean, variance, probability density function, and probability distribution function of each variable, based on the steady-state occupancy probabilities.

For transient results the user has three more solvers at his disposal. The first solver reports the mean, variance, probability density function and probability distribution function for instantaneous reward at one or more time points. A second solver gives the expected accumulated reward over an interval of time. A third solver gives the distribution of reward accumulated over an interval of time. All these three solvers are based on uniformization.

Simulation If simulation is chosen as the solution method, one can either choose to carry out a direct simulation, or use importance sampling. For direct simulation, two simulation modules are available, one for steady-state measures and one for transient measures. The steady-state simulator uses an iterative batching technique to estimate the mean and variance of reward variables and time-between-completions of activities. For transient measures the simulator uses the method of independent replications. The simulation automatically terminates when the required accuracy in the result is obtained.

To speed simulation of systems with rare events, *importance sampling* is available for transient measures. In order to apply importance sampling one specifies with a graphical editor a so-called governor that controls which biasing scheme is used. The computation of likelihood ratios to correct for the introduced bias is automatically carried out, resulting in an unbiased estimate with confidence interval as output. The importance sampling tool makes it possible to conveniently experiment with importance sampling biasing schemes for the specified model and measure [8].

For more information For more information about *UltraSAN*, contact us by e-mail at `usan@crhc.uiuc.edu`, or at the address on the first page of the report. More information can also be found on the world wide web at `http://www.crhc.uiuc.edu/UltraSAN`, and research papers can be obtained by anonymous ftp at `ftp.crhc.uiuc.edu`, in the directory `pub/UltraSAN/USAN_papers`. *UltraSAN* is provided free to academics if used for educational or research purposes.

References

[1] J. Couvillion, R. Freire, R. Johnson, W. D. Obal II, M. A. Qureshi, M. Rai, W. H. Sanders, and J. Tvedt, "Performability modeling with *UltraSAN*," *IEEE Software*, vol. 8, pp. 69–80, September 1991.

[2] W. H. Sanders, W. D. Obal, M. A. Qureshi, and F. K. Widjanarko, "The *UltraSAN* modeling environment," Accepted for publication in *Performance Evaluation*, 1995.

[3] Center for Reliable and High Performance Computing, Coordinated Science Laboratory, University of Illinois, *UltraSAN Reference Manual*, 1995.

[4] A. Movaghar and J. F. Meyer, "Performability modeling with stochastic activity networks," in *Proceedings of 1984 Real-Time Systems Symposium*, (Austin, Texas), pp. 215–224, December 1984.

[5] W. H. Sanders and J. F. Meyer, "A unified approach for specifying measures of performance, dependability, and performability," in *Dependable Computing for Critical Applications* (A. Avizienis and J. C. Laprie, eds.), vol. 4 of *Dependable Computing and Fault-Tolerant Systems*, pp. 215–238, Vienna: Springer Verlag, 1991.

[6] W. H. Sanders and J. F. Meyer, "Reduced base model construction methods for stochastic activity networks," *IEEE Journal on Selected Areas in Communications*, vol. 9, pp. 25–36, January 1991.

[7] W. H. Sanders and R. S. Freire, "Efficient simulation of hierarchical stochastic activity network models," *Discrete Event Dynamic Systems: Theory and Applications*, vol. 3, pp. 271–300, July 1993.

[8] W. D. Obal II and W. H. Sanders, "An environment for importance sampling based on stochastic activity networks," in *Proceedings of the 13th Symposium on Reliable Distributed Systems*, (Dana Point, California), pp. 64–73, October 1994.