ChE classroom

DISCRETE-EVENT SIMULATION IN CHEMICAL ENGINEERING

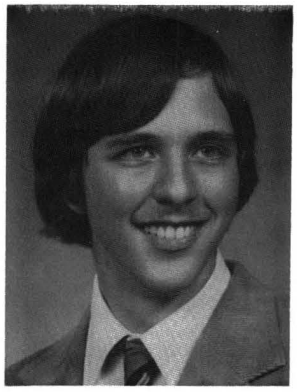
DANIEL J. SCHULTHEISZ and
JUDE T. SOMMERFELD
Georgia Institute of Technology
Atlanta, GA 30332-0100

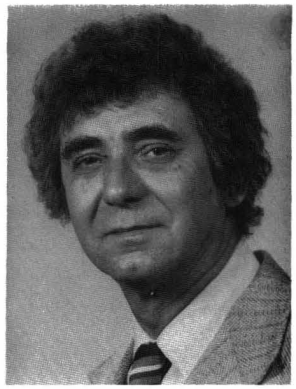
RECENT YEARS HAVE witnessed a rapid and dramatic change in the nature of the chemical process industries in the developed countries of the world. Specifically, there has been an intense revival of commercial interest in batch chemical processes, such as those employed in the manufacture of fine and specialty chemicals, at the expense of traditional continuous steady-state processes for the manufacture of commodity chemicals. One large British chemical company reports that specialty chemicals manufactured by batch processing contributed over 30% to their total profits in 1983 as opposed to 18% in 1977 [1]. Certainly, one of the primary driving forces for this change has been the recent commissioning of many world-scale commodity chemicals plants in various developing countries.

SIMULATION SYSTEMS

Concomitant with these industry changes, significant developments have occurred in the modeling and simulation of chemical processes. To be sure, usage (including academic) and development of conventional steady-state process simulators continue at an active level. Thus, the FLOWTRAN system [2] developed by the Monsanto Company was made available to chemical engineering schools in 1973 and has been extensively employed for educational purposes ever since [3]. Subsequently, newer steady-state process simulation systems such as PROCESS, ASPEN PLUS and DESIGN II became available to academic users.

Discrete-event simulators were originally developed as numerical aids to solve complex queuing theory problems which were not amenable to analytical solution. Such problems occur routinely in the field of industrial engineering . . .





Daniel J. Schultheisz received his bachelor's degree in chemical engineering from the University of Pennsylvania and is currently completing the requirements for his master's degree in chemical engineering at Georgia Tech. For his thesis, he is preparing an instructor's manual of exercises in chemical engineering using GPSS, to be distributed and marketed by the CACHE Corporation. (**L**)

Jude T. Sommerfeld is professor and associate director of the School of Chemical Engineering at Georgia Tech. He received his BChE degree from the University of Detroit and his MSE and PhD degrees, also in chemical engineering, from the University of Michigan. His 25 years of industrial and academic experience have been primarily in the area of computer-aided design, and he has published over seventy articles in this and other areas. (**R**)

We have also witnessed the development and application of various simulators for batch chemical processes in recent years. These developments have included both discrete-event and combined (discrete + continuous) systems, as employed in the industrial engineering field. There is an unfortunate confusion in terminology here: the industrial engineering interpretation of the term 'continuous' is not the same as that associated with chemical engineering usage, namely, steady-state operation. Rather, the industrial engineering meaning of continuous should be construed by chemical engineers as dynamic or unsteady-state.

The progenitor of discrete-event simulation systems is GPSS [4], which dates back to 1959 and is still used extensively in many manufacturing sectors. Because of its easy use, availability, reliability, and efficient operation (integer arithmetic only in many versions), GPSS is a very effective tool if only discrete simulation capability is required. Other popular dis-

TABLE 1
Recent Applications of GPSS to Chemical and Allied Processes

APPLICATION	[REF.]
Two batch reactors in parallel followed by a batch still	[7]
DDT manufacture	[8]
Chocolate manufacturing	[9]
Sequence of batch distillation columns	[10]
Large-scale poliomyelitis vaccine production	[11]
Choline chloride manufacture	[12]
Polyvinyl chloride (PVC) manufacture	[13]
Sequencing batch reactors (SBR) for wastewater treatment	[14]

crete-event simulation systems include SIMULA [5] (more prevalent in Europe) and SIMSCRIPT [6]. In general, however, there are not many published applications of discrete-event simulation systems to batch chemical processes. Morris [7], for example, has described a very simple application of GPSS to a batch process comprised of two reactors in parallel followed by a still. Other recent applications of GPSS to chemical and allied batch (or semi-continuous) processes are listed in Table 1.

DISCRETE-EVENT SIMULATION

Discrete-event simulators were originally developed as numerical aids to solve complex queuing theory problems which were not amenable to analytical solution. Such problems occur routinely in the field of industrial engineering and typical example applications include machine shops, customer service stations, and transportation networks.

TABLE 2
Terminology in Usage of the FLOWTRAN and GPSS Simulators

ITEMS	FLOWTRAN (Steady-State Process Simulator)	GPSS (Discrete-Event Simulator)
Precoded functional subroutines	Blocks	Blocks
Arguments of functional subroutines	Parameters	Operands
Items moving through the model	Streams	Transactions
Numerical characteristics of moving items	Properties (*e.g.*, temperature, composition)	Parameters
Output quantities from subroutines (other than moving items)	Retention vector contents	Standard numerical attributes

Despite their considerable differences in origin and application, there are noteworthy similarities among the various types of simulators . . . For example, FLOWTRAN and GPSS have a number of precoded functional subroutines.

Most discrete simulation systems have stochastic capabilities for the scheduling of time events. To support this function, most such systems also have one or more built-in random-number generators. Output from the latter is used to sample event times (or durations between time events) from various probability distributions. In GPSS, the only easy-to-use, built-in distribution is the uniform or rectangular distribution. Thus, for example, a service time can take the form, $A \pm B$, where A represents the mean value and B is the half-width, in appropriate time units, of the distribution.

SIMILARITIES IN SIMULATORS

Despite their considerable differences in origin and application, there are noteworthy similarities among the various types of simulators described above. For example, FLOWTRAN and GPSS have a number of precoded functional subroutines (generally written in FORTRAN). In both FLOWTRAN and GPSS, these functional subroutines are known as blocks.

There is a number of other similarities between these two systems, obscured only by the technical jargon employed. In conventional steady-state chemical process simulators such as FLOWTRAN and PROCESS, the items which move from one block to another in the model are known as streams. Each individual stream has a set of properties (composition, temperature, pressure) associated with it, which are typically modified as the stream passes through a block. In analogous fashion, the items which proceed from block to block in a GPSS model are known as transactions. Also in analogy with stream properties in a steady-state process simulator, GPSS transactions have associated with them various parameters (such as priority level or lifetime in the model) which can be modified by the passage of the transaction through certain blocks. In a GPSS model of a batch chemical plant, for example, transactions could represent batches of material proceeding through the process. Table 2 summarizes these similarities and terminology for the FLOWTRAN and GPSS simulators.

GPSS PROCESSOR

There are about thirty-five different blocks in GPSS, roughly the same number as in the FLOWTRAN system. A listing of these GPSS blocks is given in Table 3. It is common to construct block diagrams in the development of GPSS models. In contrast with FLOWTRAN where each block in such diagrams is represented more or less by a rectangle, each different functional block in a GPSS block diagram has its own distinctive shape (see Schriber [4]).

Some of the GPSS blocks listed in Table 3 are quite complicated and would typically be used only by more sophisticated analysts. There are others, however, which would be common to any GPSS model. Thus, GENERATE blocks are used to provide transactions to a model, much like a chemical engineer inputs feed streams to a FLOWTRAN model. Conversely, a transaction is removed from a GPSS model by a TERMINATE block.

There is a block named SPLIT in both FLOWTRAN and GPSS, but there is one fundamental difference between the two. In the FLOWTRAN system, the sum of each extensive property over all of the output streams from the block equals that property for the incoming stream. In discrete-event simulation with GPSS, however, the SPLIT block really performs a cloning operation. That is, one or more identical offspring transactions are created from the single parent transaction (which retains its existence) entering the block.

GPSS OUTPUT

As with the FLOWTRAN system which provides a summary table of the streams passing through the model and output results from each of the blocks in the model, GPSS automatically prints out a variety of output statistics at the conclusion of a simulation. These statistics pertain primarily to the various facilities, queues, and storages in the model.

Thus, from an inspection of the facility output statistics from a GPSS simulation, an analyst might find that the average holding time per transaction for a given facility is considerably greater than the user-supplied average service time for that facility. In a chemical engineering application, for example, this could indicate that a reactor, after finishing processing of a batch (transaction), often cannot discharge the batch because of an unavailable downstream facility. The latter might correspond to a storage tank which is full or another processing unit (*e.g.*, still, centrifuge, dryer) which is engaged. The regular occurrence of such a situation would normally be accompanied by an average utilization (fraction of total time busy) approaching unity for the original upstream facility and would suggest the existence of some downstream bottleneck. The existence of similar bottleneck situations can also be deduced from the output statistics for GPSS storages. The productivity (number of batches produced) of the modelled process is, of course, related to the number of transactions passing through the GPSS model.

TABLE 3
Listing of GPSS Blocks

ADVANCE	LEAVE	RELEASE
ASSEMBLE	LINK	RETURN
ASSIGN	LOGIC	SAVEVALUE
BUFFER	LOOP	SEIZE
DEPART	MARK	SELECT
ENTER	MATCH	SPLIT
FAVAIL	MSAVEVALUE	TABULATE
FUNAVAIL	PREEMPT	TERMINATE
GATE	PRINT	TEST
GATHER	PRIORITY	TRANSFER
GENERATE	QUEUE	UNLINK

EXAMPLE APPLICATION

Let us consider a very simple application of GPSS to the modeling of a batch chemical process. This example is an adaptation of a problem (number 2.41.14) presented by Schriber [4]; the process flow diagram for this example is presented in Figure 1.

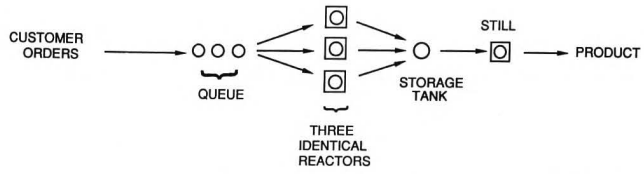


FIGURE 1. Sketch of batch process for example problem.

Thus, a small, single-product batch chemical plant has three identical reactors in parallel, followed by a single storage tank and a batch still. Customer orders (batches) to be filled (which begin with processing in the reactor) occur every 115 ± 30 minutes, uniformly distributed. The reaction time in a given reactor is 335 ± 60 minutes, and the distillation time in the still is 110 ± 25 minutes, both times uniformly distributed. The holding capacity of the storage tank is exactly one batch. Hence, the storage tank must be empty for a given reactor to discharge its batch; if not, the reactor cannot begin processing a new batch until the storage

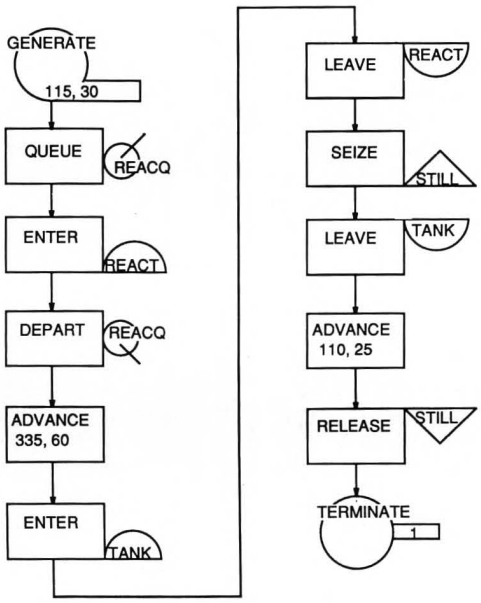


FIGURE 2. GPSS block diagram for example problem.

tank becomes empty. The simulation is to be run for 100 batches. The model should have the capability to collect waiting line statistics for the queue immediately upstream of the reactor.

The GPSS block diagram for this example model is shown in Figure 2. Note the distinctive shapes for each of the blocks employed. The first executable block is the GENERATE block, which creates transactions representing customer orders (batches). These transactions immediately queue up and attempt to capture an available reactor via the ENTER block. After capturing a reactor, a batch leaves the reactor queue through the DEPART REACQ block, and is processed in the ADVANCE 335,60 block. The batch must first be able to enter the storage tank (ENTER TANK block) before it releases its reactor in the LEAVE REACT block. The batch then attempts to capture the single still facility in the SEIZE STILL block. Having accomplished such, the batch leaves the storage tank, is processed in the still, releases the latter, and finally leaves the model through the TERMINATE block. Selected output statistics from this simulation are summarized in Table 4.

From Table 4, one sees that the batch still was in use 91.1% of the time, and the average holding (processing) time per batch was 108 minutes. The average contents in the queue upstream of the reactors was 0.44 batch, and the average waiting time for all batches, including ones which experienced no waiting, in this queue was 50.5 minutes. The three reactors were in use 95.2% of the time, and the average holding time for a batch in a reactor was 329 minutes. Similarly the storage tank (with a capacity of one batch) was full 41.4% of the time, and the average holding time therein was 48.6 minutes. Although not presented in Table 4, the total simulation time, to completely process 100 batches, was 11,967 minutes.

One can easily explore proposed modifications to this process. Thus, one more reactor could be added in an effort to increase productivity. One might find as a result, however, a significant increase in the average reactor holding time beyond the nominal average reaction time of 335 minutes. In this case, one could explore increasing the intermediate storage capacity (TANK) and/or improving the downstream distillation operation.

SUMMARY

This article has attempted to serve as a brief introductory tutorial on discrete-event simulation, with emphasis on chemical engineering applications. For some simple batch process applications, only discrete simulation capability is required. More complex applications would require usage of a combined (discrete plus dynamic) simulation system, but knowledge of the essential features of discrete-event simulation remains useful background in such cases.

REFERENCES

1. Preston, M. L., and G. W. H. Frank, "A New Tool for Batch Process Engineers—ICI's BatchMASTER," *Plant/Operations Progr.*, 4, 217 (1985).
2. Seader, J. D., W. D. Seider, and A. C. Pauls, *FLOWTRAN*

TABLE 4
Selected GPSS Output from Example Simulation of a Batch Chemical Process

Facility Statistics (batch still, STILL):	
Average utilization	**0.911**
Average holding time per batch, min	**108**
Queue Statistics (reactor queue, REACQ):	
Maximum contents	**3**
Average contents	**0.44**
Average waiting time (all batches), min	**50.5**
Storage Statistics:	
Reactors (REACT):	
Average utilization	**0.952**
Average holding time per batch, min	**329**
Storage Tank (TANK):	
Average utilization	**0.414**
Average holding time per batch, min	**48.6**

Simulation—An Introduction, 2nd Ed., Ulrich's Bookstore, Ann Arbor, MI, 1977.
3. Clark, J. P., and J. T. Sommerfeld, "Use of FLOWTRAN Simulation in Education," *Chem. Eng. Ed.*, **10**, No. 2, 90 (1976).
4. Schriber, T. J., *Simulation Using GPSS*, Wiley, New York, 1974.
5. Franta, W. R., *The Process View of Simulation*, Elsevier North-Holland, New York, 1977.
6. Wyman, F. P., *Simulation Modeling: A Guide to Using SIMSCRIPT*, Wiley, New York, 1970.
7. Morris, R. C., "Simulating Batch Processes," *Chem. Eng.*, **90**, No. 10, 77 (1983).
8. Blaylock, C. R., C. O. Morgan, and J. T. Sommerfeld, "GPSS Simulation of DDT Manufacture," *Proc. 14th IASTED Intl. Conf. on Appl. Sim. and Mod.*, Vancouver, June, 1986, p. 314.
9. Passariello, I., and J. T. Sommerfeld, "GPSS Simulation of Chocolate Manufacturing" in *Tools for the Simulation Profession*, Soc. for Computer Simulation, San Diego, 1987, p. 1.
10. Barnette, D. T., and J. T. Sommerfeld, "Discrete-Event Simulation of a Sequence of Multi-Component Batch Distillation Columns," *Comp. & Chem. Eng.*, **11**, 395 (1987).
11. Kenvin, J. C., and J. T. Sommerfeld, "Discrete-Event Simulation of Large-Scale Poliomyelitis Vaccine Production," *Process Biochem.*, **22**, 74 (1987).
12. Bales, W. J., J. R. Johnson, and J. T. Sommerfeld, "Use of a Queuing Simulator in Design of a Batch Chemical Production System," *Prodn. and Inventory Mgmt.*, accepted for publication (1987).
13. Adebekun, A. K., Z. Q. Song, and J. T. Sommerfeld, "GPSS Simulation of PVC Manufacture," *Polymer Process Eng.*, **5**, 145 (1987).
14. Glenn, S. L., R. T. Norris, Jr., and J. T. Sommerfeld, "Discrete-Event Simulation of Sequencing Batch Reactors (SBR) for Wastewater Treatment," *J. Water Poll. Cont. Fedn.*, submitted for publication (1987). □

ENTROPY

Continued from page 97.

Zenzen, *The Nature of Irreversibility*, D. Reidel Pub. Co., Dordrecht, Holland, 1985.
9. Teilhard de Chardin, P., *The Phenomenon of Man*, Harper and Row, New York, 1975. A good summary of Teilhard's arguments is presented in Barrow, J. D. and F. J. Tipler, *The Anthropic Cosmological Principle*, Ch. 3, Clarendon Press, Oxford, 1986.
10. Adams, Henry, *The Degradation of Democratic Dogma*, Reprint of the 1919 ed., Harper Torchbooks, New York, 1969.
11. Levi-Strauss, C., *Tristes Tropiques*, translated by J. and D. Weightman, Jonathan Cape, London, 1973.
12. Georgescu-Roegen, N., *The Entropy Law and the Economic Process*, Harvard University Press, Cambridge, 1971.
13. Daly, Herman E., *Steady-State Economics*, W. H. Freeman & Co., San Francisco, 1977.
14. Rifkin, J., *Entropy: A New World View*, Bantam Books, New York, 1980.
15. Tanner, Tony, *City of Words*, Chap. 6, Harper and Row, New York, 1971, and Lewicki, Zbigniew, *The Bang and the Whimper*, Greenwood Press, Westport, CT., 1984.
16. Greenland, Colin, *The Entropy Exhibition*, Routledge and Kegan Paul, London, 1983.
17. Wiener, Norbert, *Human Use of Human Beings*, Houghton-Mifflin, Boston, 1950.
18. Whether entropy can be linked to the information in a message is still an undecided question. The views of Brillouin and the Denbighs typify the pro and con, respectively. Brillouin, L., *Scientific Uncertainty and Information*, Academic Press, New York, 1964. Denbigh, K. G. and J. S. Denbigh, *Entropy in Relation to Incomplete Knowledge*, Cambridge University Press, Cambridge, 1985.
19. Arnheim, Rudolf, *Entropy and Art*, University of California Press, Berkeley, 1971.
20. de Santillana, G., and H. von Dechend, *Hamlet's Mill*, Gambit Inc., Boston, 1969.
21. Romans 8:21.
22. For lucid and concise discussion of the philosophical problem of time, see Denbigh, K. G., *Three Concepts of Time*, Springer-Verlag, Berlin, 1981.
23. Whorf, B. L., *Language, Thought and Reality*, M.I.T. Press, Cambridge, 1956.
24. See, for example, Capra, F., *The Tao of Physics*, Bantam Books, New York, 1975.
25. Both the external and internal views are discussed by Morris Kline in *Mathematics and the Search for Knowledge*, Oxford University Press, Oxford, 1985. The internal view is examined by Roger S. Jones in *Physics as Metaphor*, New American Library, New York, 1982. □

EDUCATOR: Bailey

Continued from page 61.

pitchers helps my students know each other better," Jay notes. "It also helps me maintain a friendly and open relationship with my group that's important in our work together."

Research by Bailey and his students was recognized by the Curtis W. McGraw Research Award of the American Society of Engineering Education in 1983, by Jay's election to the National Academy of Engineering in 1986, and by the AIChE Professional Progress Award in 1987.

Bailey does have interests outside of the lab. Everyone who knows him remarks on his devotion to Sean, his 18-year-old son, who's now a freshman at the University of Colorado, Boulder. Jay's an avid amateur musician—the guitar is his instrument—and he loves active sports such as tennis, racquetball, and bicycling. He and Arnold also love to travel. Says Bailey, "We went to Malaysia and Indonesia last summer and just wandered around for four weeks for absolutely no professional reason whatsoever. It was wonderful."

Frances Arnold sums up Jay Bailey's influence on his profession in the following way, "Jay stands out in the field as a pioneer in new techniques in the 8,000-year-old discipline of biochemical engineering. You won't find many new products coming out of his lab, but you will find many new ideas." □