Proceedings of the 2010 Winter Simulation Conference
B. Johansson, S. Jain, J. Montoya-Torres, J. Hugan, and E. Yücesan, eds.

INVERSE DISCRETE EVENT MODELING FOR FACILITY PARAMETER ESTIMATION

Reid Kress

Y-12 National Security Complex
P.O. Box 2009, Bear Creek Road
Oak Ridge, TN 37831, USA

Alma Cemerlic
Jessica Kress
Jacob Varghese

SimCenter, The University of Tennessee
701 E. M. L. King Blvd
Chattanooga, TN 37403, USA

ABSTRACT

Particular applications require analysts to estimate plant throughput from external observables via inverse modeling techniques. For example, auditors, law enforcement personnel, and financial planners might need to perform these types of analyses. Researchers at the SimCenter at The University of Tennessee Chattanooga have elected to model several simple basic production models as well as a fictional bicycle factory to do a preliminary investigation into the viability of implementing an inverse model using a discrete-event simulation software package. The fictional bicycle model will eventually include several simulation features such as a discrete event component, a flow portion, an agent based part, equation based power portion, and optimization. The results indicate that the approach is viable and that inverse modeling can be used to estimate internal activities. Future work will involve more detailed models with larger parameter sets.

1 INTRODUCTION

Many applications rely on inverse modeling. Laboratory analysis, instrumentation applications, machine health monitoring, and vibration analysis (Gladwell 2004) are all examples of tradition inverse problems. These applications generally have a strong mathematical focus and rely heavily on computer solution. Most often, processes are modeled as continuous in time with data collected on evenly-spaced intervals with at most a finite number of variations in the time step. A new area of research is the application of inverse techniques to the modeling of systems containing processes that exhibit behaviors that have inherently uneven time intervals. This so called discrete event systems appear often in industrial arenas, manufacturing, call centers, server/client problems, health care, and other service-oriented businesses. They are often modeled directly using any of a number of discrete event simulation software packages (Zapata, Suresh and Reklaitis 2007).

2 APPROACH

2.1 Basic Inverse Models

A classic inverse problem in a manufacturing environment would be the estimation of internal plant parameters based on some measured output. For example, one might measure the flow of waste effluent from a factory, employ a physics-based model to describe the postulated internal processes, and then estimate values of specific model parameters that provide some agreement between the time profile of the modeled effluent stream and the time profile of the effluent stream measured from the plant. If the phys-

978-1-4244-9864-2/10/$26.00 ©2010 IEEE

ics-based model is a realistic representation of the internal processes and a valid solution to the parameter estimation problem is obtained, then one might have a reasonable representation of the internal processes of the factory. There are caveats, of course, for example solution uniqueness is always a concern when inverse solutions are sought as well as is the rate of convergence to a solution..

Consider a simple inverse problem in a discrete-event simulation model. One has a process consisting of two serial activities. Items enter to be processed at time intervals obtained from a random distribution (e.g. exponential), enter a queue, and then are processed first by activity one, exit to an intermediate queue, and are finally processed by activity two. This system is modeled in ExtendSim software <http://www.extendsim.com/index.html> in figure 1.

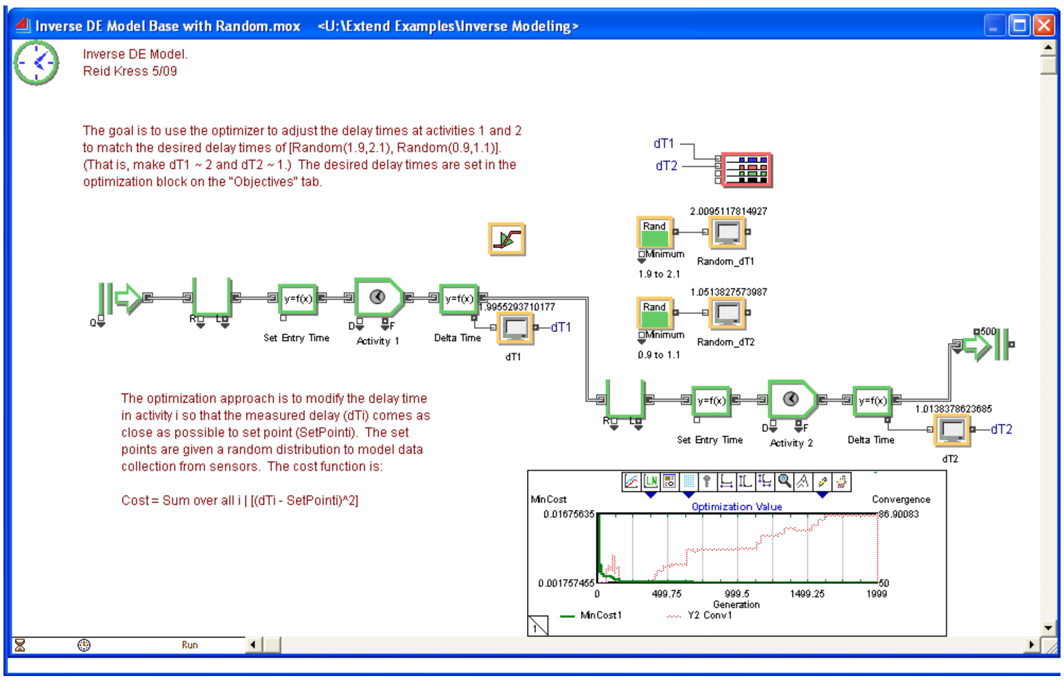

Figure 1: Simple two-activity system used in basic inverse example.

A simple test of inverse analysis in a discrete-event simulation paradigm consists of assuming that a delay time has been specified for items in each of the two activities (e.g. via direct observation or inference), measuring the actual delay time of items in the activities, and optimizing the values of the actual delays in the modeled activity until a specific cost (profit) is minimized (maximized). Since ExtendSim has a built-in optimizer this is easy to implement within the simulation environment. Other variables are important in this basic analysis: 1) the number of constraints in the cost (profit) function versus the number of parameters in the inverse solution and 2) the affect of random variation in parameters. More parameters than constraints is under constrained, more constraints than parameters is over constrained. It is anticipated that over/under constraining and random variation will impact both solution uniqueness and rate of convergence. Table 1 summarizes the models, cost functions, and parameters tested in the basic inverse models.

Table 1: Set of basic inverse models.

Model Identifier	Number of Activities	Parameters	Cost Function	Comment
Base	2	Delay1 & Delay2	$(dT1\text{-}SetPt1)^2+(dT2\text{-}SetPt2)^2$	Direct inversion, deterministic model only
Base with Random	2	Delay1 & Delay2	$(dT1\text{-}SetPt1)^2+(dT2\text{-}SetPt2)^2$	Direct inversion, stochastic model
Under constrained with Random	4	Delay1, Delay2, Delay3, Delay4	$(dT1\text{-}SetPt1)^2+(dT2\text{-}SetPt2)^2$	More parameters than constraints, stochastic model
Over constrained with Random	4	Delay1 & Delay2	$(dT1\text{-}SetPt1)^2+(dT2\text{-}SetPt2)^2 + 0.01(dT3\text{-}SetPt3)^2+ 0.01(dT4\text{-}SetPt4)^2$	More constraints than parameters, stochastic model, note weights on set points 3 & 4

Note: Delayi = model delay in activity i; dTi = measured model delay at activity i; SetPti = the desired delay (set point) at activity i.

These inverse models were used in preliminary analyses to determine if an inverse approach using ExtendSim's built-in optimizer was feasible, to establish whether random variables would make the analysis fail (e.g. no convergence), and to determine if under or over constrained conditions created problems.

2.2 Fictional Bicycle Plant

As a more rigorous evaluation of the inverse discrete event modeling approach using ExtendSim software, a fictional bicycle factory was simulated. (See for example Kress 2007.) The fictional bicycle shop is a facility that manufactures new bicycles from raw materials. The prototypical bicycle needs to have a frame and two wheels (among other less important items such as seat, brakes, and handlebars!). The frame consists of two parts, and a wheel consists of a rim, tire, and spokes.

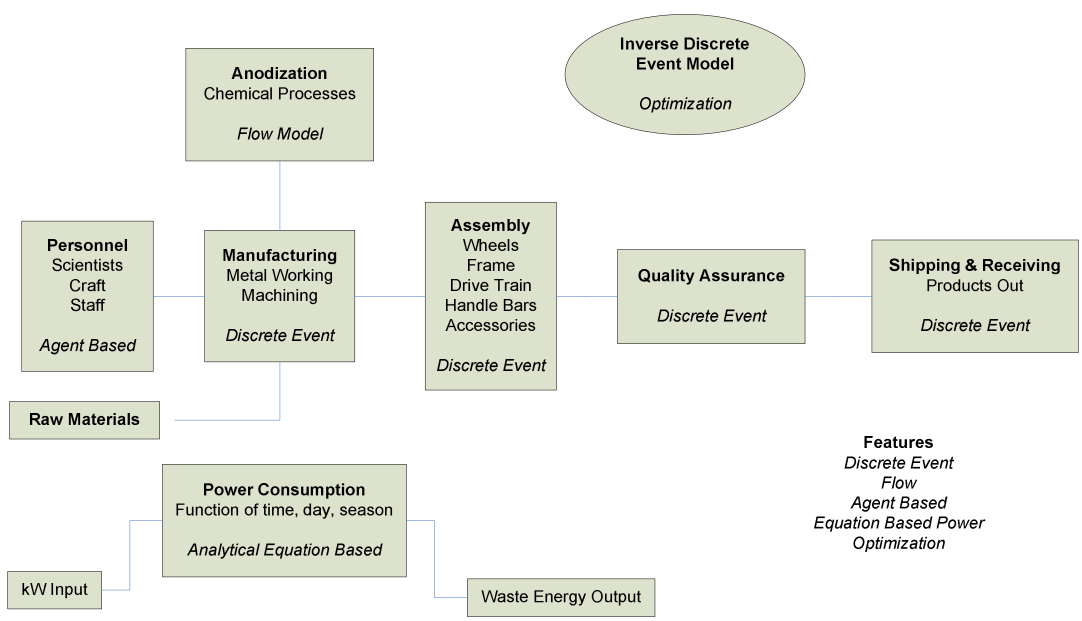

Figure 2: Fictional bicycle factory block diagram. Italicized words indicate particular model features associated with the block.

This bicycle is assembled using different labor personnel in a series of steps that require time, materials, power, and labor. In addition, the handlebars of the bicycle are to be anodized for appearance and durability. The manufacturing process is illustrated in figure 2 on the following page. This simulation has several unique discrete event modeling features including: basic discrete event portions, continuous flow portions, agent based section, an analytical equation-based part, and an optimizer. Combining these features exercises a broad spectrum of ExtendSim's capabilities and creates a more difficult case for the inverse solver.

3 RESULTS

3.1 Basic Model Verification and Validation

A basic model with only one variable parameter was used for verification and validation of the model and approach. Figure 3 on the following page shows a flow diagram illustrating how the simple model was used for verification and validation. The system was established with a randomly varying parameter. The magnitude of the random variations was one of the independent variables. The optimizer was then used to match an estimator's parameter to the system's parameter. The randomly varying parameter changed from run to run and the estimator attempted to establish a fixed value. If the optimizer was estimating the values of a variable parameter in a real system, one would hope that the estimated value would be representative of the system parameter's statistical distribution; for example, a mean. Thus, the error between the system parameter's mean value and the estimated parameter was calculated for various levels of parameter variation. This is shown in figure 4 on the following page.

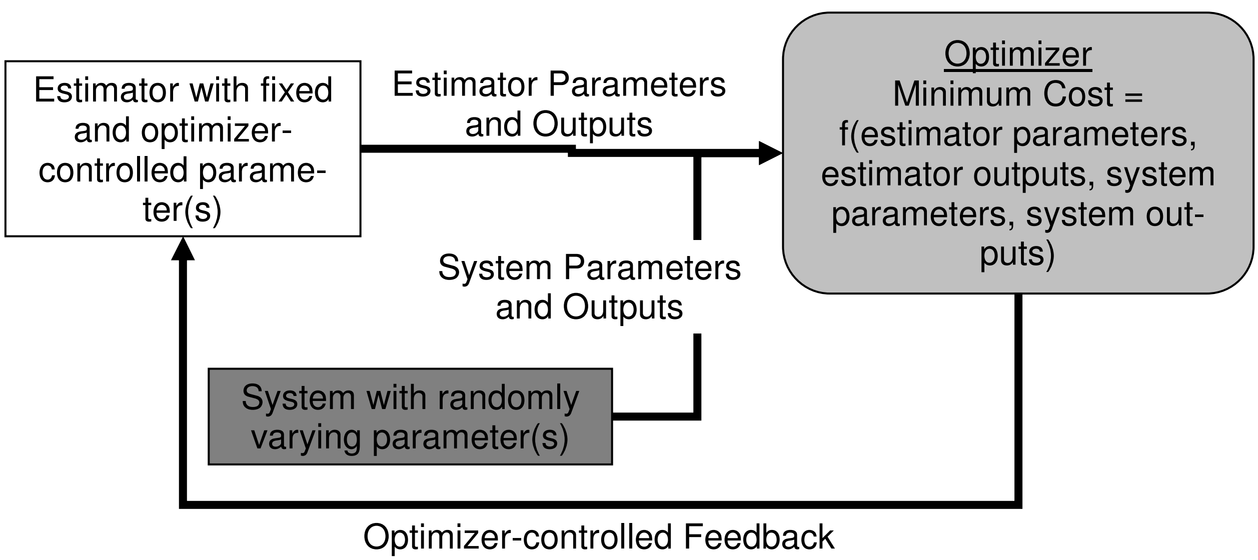

Figure 3: Schematic of the verification and validation basic model with randomly varying system and optimizer/estimator.

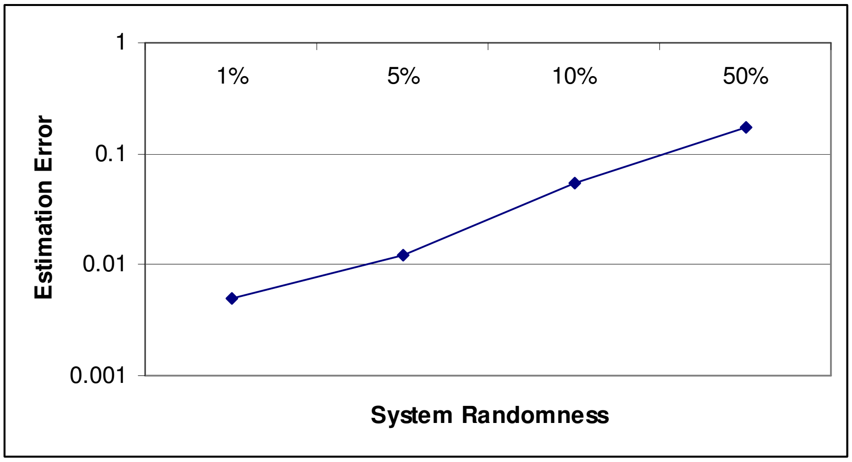

Figure 4: Error between system parameter mean as a function of system randomness level for basic model verification and validation.

Note that with no system randomness, error was essentially zero, being primarily a function of the optimization stopping criteria.

3.2 Basic Model Results

The optimizer was set up to perform the inverse analysis for the four simulations described in table 1. Run parameters were set to have the optimizer make five runs per case and to examine fifty cases before checking convergence. Termination was set at either a maximum number of runs or when the difference between the best and worse of the top ten cases is within ninety five percent. Table 2 on the following page illustrates a typical result from the inverse analysis showing the top ten delays for an inverse solution with the over constrained and random simulation. The set points were set to 2 for delay number 1 and for 1 on delay number 2. The cost function was that established in the fourth row of table 1 with set points uniformly randomly selected between 1.9 and 2.1 and 0.9 and 1.1 for activity 3 and activity 4 respectively. Note that the costs associated with activities 3 and 4 are weighted two orders of magnitude lower than those costs for activities 1 and 2. Figure 5 shows the minimum cost for the over constrained random simulation as a function of time. Notice how the cost function generally decreases as time increases becoming monotonic as the number of samples (i.e. time) increases. Finally, figure 6 on the following page compares the time to execute for the four different inverse model scenarios of table 1.

Table 2: The ten minimum cost results from the optimizer for the over constrained and random inverse model.

Delay 1	Delay 2	Minimum Cost
2.057445955	0.97078475	0.001792878
2.054829589	0.971002603	0.001882526
2.056285571	0.97100526	0.002452868
2.055117767	0.971057697	0.002529458
2.056285571	0.97100526	0.002601504
2.074828576	0.970558299	0.002642186
2.057236062	0.970935182	0.002743123
2.054548752	0.971064065	0.002763853
2.054548752	0.971064065	0.002923926
2.065574867	1.040252161	0.007365218

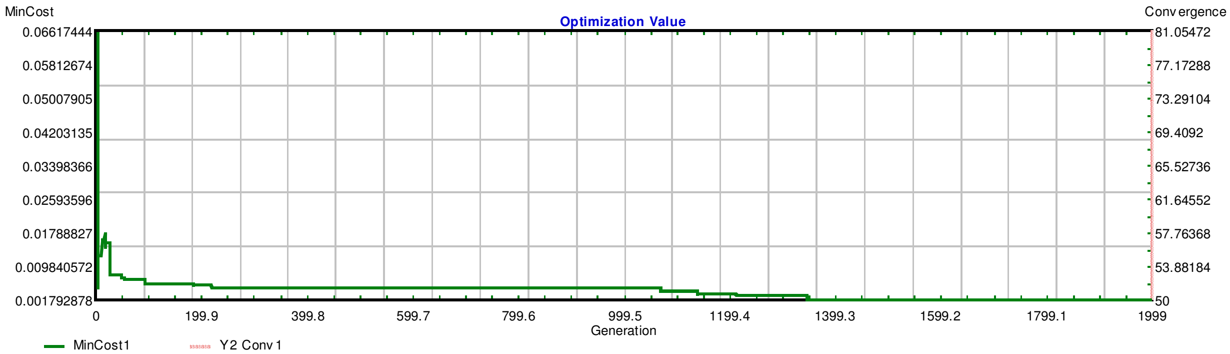

Figure 5: Minimum cost as a function of time for the over constrained random case (line 4 in table 1).

Figure 6: Execution times for the basic inverse models described in table 1.

3.3 Bicycle Model

Typical results for number of operations calculated from the bicycle development model are shown in Figure 7.

This simulation used a customer with an initial inventory of 100 bicycles and an estimated sales rate (consumption) of one bicycle per day. Sales that reduced the inventory below 100 bicycles were immediately followed by orders for new bicycles in batches of five. The simulation was run for 70 days and it was assumed that each operation in the assembly required one of the same pieces of equipment (e.g. an assembly fixture or stand). Figure 7 shows the maximum number of concurrent operations required to fulfill the customer's demands. Note the set of operations includes both assembly and disassembly as well as "decision" operations where the disassembled parts are assessed relative to their reusability (e.g. Frame is Scrap operation).

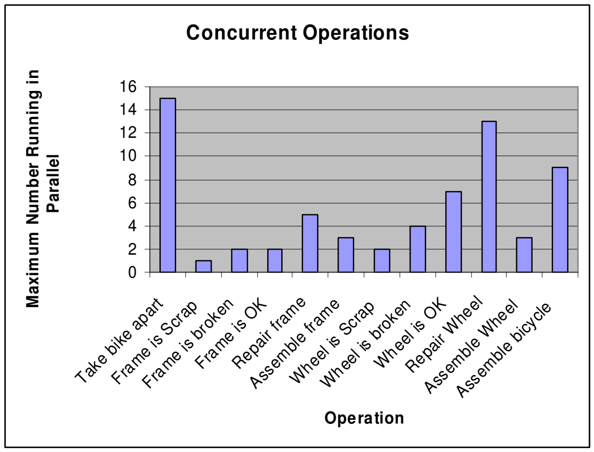

Figure 7: Maximum Number of Concurrent Operations in the Bicycle Shop Model

Initial results with the optimization show that the technique can estimate system internal parameters depending upon which outputs are measured and which parameters are being estimated. For example, table 3 illustrates estimation of one of the variable activity delays by measuring power consumed. (Note the "Power Consumption" block in figure 2.) The power consumption model had three levels of coupling between the power consumed and the product throughput: low, moderate, and high. For low coupling, power consumed was a weak function of throughput representing a factory with a base load largely determined by factors other than the manufacturing process. High coupling represented a factory where consumed power was highly dependent upon throughput. Moderate coupling was in between high and low representing a system with balanced base load and throughput-proportional load.

Table 3: Estimation of an internal system process activity delay having randomly varying delay time by measuring system power consumption that is dependent upon the product throughput.

System Parameter Variation (%)	Power Consumption Coupling	Estimated parameter Error (%)
0	Low	0.1
0	Moderate	0.07
0	High	0.09
10	Low	0.7
10	Moderate	0.83
10	High	0.40

Estimated parameter error does not seem to be affected by coupling at this level. As long as there is some relationship between power consumed and throughput, then the estimator was able to determine the parameter even with 10% variability.

4 CONCLUSIONS

The discrete event inverse analysis approach is certainly capable of evaluating the internal delays for the basic models. In these cases, small order model mismatches (under constrained or over constrained) and random numbers do not interfere with the solution progress. The solution may take longer; however, it still provides a viable answer. In the case of the more complex models such as the bicycle factory, much depends upon the data collected and coupling between model segments. In any simulation, it is possible to estimate internal parameters to some extent; however, convergence rate, accuracy, and uniqueness all depend upon the system and data. Details of this interdependence are a topic of further work.

REFERENCES

Gladwell, G.M.L. 2004. *Inverse Problems in Vibration*. Kluwer Academic Press, Boston.

Kress, R, J. Dixon, T. Insalaco, and R. Rinehart. 2007. A Supply Chain Paradigm to Model Business Processes at the Y-12 National Security Complex. In *Proceedings of the 2007 Winter Simulation Conference*, ed. B. Biller, M.-H. Hsieh, J. Shortle, J. D. Tew, and R. R. Barton, 1977–1985. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers, Inc.

Zapata, J. C., P. Suresh, and G.V. Reklaitis. 2007. Assessment of Discrete Event Simulation Software for Enterprise-wide Stochastic Decision Problems. In *17th European Symposium on Computer Aided Process Engineering – ESCAPE17*, ed. V. Plesu and P.S. Agachi.

AUTHOR BIOGRAPHIES

REID KRESS is a Senior Technical Advisor in the National Security Technology Center at the National Nuclear Security Administration's Y-12 Plant. His research interests include software development for specialized applications in national security, robotics, and information technology, simulation and systems modeling, dynamic system analysis (controls and vibrations), and applied mathematics. He is a visiting professor at The University of Tennessee Chattanooga in the Computational Engineering Department.

ALMA CEMERLIC is a graduate student in the Computational Engineering Department at The University of Tennessee Chattanooga. Her undergraduate major was computer science. Her research interests include computational fluid mechanics and discrete event simulation.

JESSICA KRESS is a graduate student in the Computational Engineering Department at The University of Tennessee Chattanooga. Her undergraduate major was mathematics. Her research interests include mathematics of computational engineering and discrete event simulation.

JACOB VARGHESE is a graduate student in the Computational Engineering Department at The University of Tennessee Chattanooga. His undergraduate major was engineering. His research interests include computational engineering and discrete event simulation.