

## CLINIC: AGGREGATING SUBSYSTEM MODELS INTO AN AUTOMOTIVE TOTAL PLANT THROUGHPUT MODEL

Jeffrey Scott Miller  
Randy Combs

Throughput Simulation, 30300 Mound Road  
Manufacturing B.  
Warren, MI 48090, U.S.A.

D.J. Medeiros  
(Expert Commentator)

Industrial and Manufacturing Engineering  
Penn State University  
University Park, PA 16802, U.S.A.

Earnest Foster  
Jeffrey Tew

Mfg Systems Research Lab, 30500 Mound Road  
General Motors R&D Center  
Warren, MI 48090, U.S.A.

Onur Ulgen  
(Expert Commentator)

University of Michigan-Dearborn and  
Production Modeling Corp., IMSE Department  
Dearborn, MI 48126, U.S.A.

### ABSTRACT

This paper presents problems encountered within the simulation modeling community of General Motors when they are faced with the requirement to verify all new plant designs for their entire global manufacturing enterprise. Given that the Body Shop, Paint Shop, and General Assembly areas of an automotive plant are modeled independently in two different simulation packages, we desire this clinic session to address issues encountered when attempting to model the entire plant with one simulation package. Obtaining accurate total plant throughput from this one model representation is of interest. The Paint Shop, in particular, is modeled in a 3D simulation package and its results need to be represented correctly in terms that are interoperable with a 2D simulation package.

### 1 INTRODUCTION

Various modeling difficulties are encountered within the simulation modeling community of General Motors when they are faced with the requirement to verify all new plant designs for their entire global manufacturing enterprise. Although plant simulation for the automobile manufacturing facilities of major automotive companies is a problem previously addressed in the literature (Shin et al. 2004; Williams and Celik 1998; Park, Matson, and Miller 1998) General Motors desires their simulation modeling to simultaneously comprehend the Body Shop, the Paint Shop, and the General Assembly areas of their vehicle manufacturing centers.

General Motors currently uses two different simulation packages for the modeling and analysis of total plant throughput. Package A is a 2D software package that is

particularly suited for the simulation of Body Shops (Figure 1) and General Assembly (Figure 2), where jobs flow through the areas in a fairly predictable pattern. The rectangles in Figure 2 represent stations with their jobs/hr, Stand Alone Throughput (SAT), and Stand Alone Availability (SAA); the circles represent buffers with their capacities inscribed inside. (For confidentiality, the data depicted is not actual data.) Simulation Package A has the advantage of being easy to use and has a fast run time when in non-graphical mode.

Package B is a 3D software package that is typically used for paint shop analysis. It is fully capable of modeling complex routings, controls logic, and other physical characteristics associated with a paint shop. Package B is needed for the paint shop to model certain conveyor constructs unique to paint shops and the unique multiple speed situations material throughput would experience. Rework, repair, and special material handling requirements in the paint shop necessitate having a more sophisticated simulation software package to emulate the controls logic.

### 2 SUBSYSTEMS FOR AGGREGATION

General Motors currently uses a multi-tiered approach towards simulation allowing for simulating a particular plant at various levels of aggregation. For example, the entire Plant can be represented by one station box, the Body Shop can also be simulated and represented as one station box (the first box in Figure 5), and its detailed sub-systems can be simulated (Figure 1). The results obtained when a simulation is run at a more detailed level can be used as input into a more aggregate model. Various throughput targets will exist at the sub-system level, the three “shop” levels (Figure 5), and at the “total plant” level.



GM currently utilizes a custom aggregation method to create a “black box” representation of more detailed simulation models for use in these total plant models. This aggregation method assumes an exponential distribution of down times, which may not be valid for paint shops due to repair rates and different break schedules within the shop. The fact that paint shops typically run slower at the front end (elpo/phosphate) and faster at the back end, where repair jobs are merged back into the flow, further complicates the question of how best to statistically represent aggregate shop level performance in terms of speed, mean cycles between failures (MCBF), mean time to recover (MTTR) and their respective distributions.

System Aggregates	
<b>Speed:</b>	72.0 jph
<b>SAA:</b>	93.2 %
<b>SAT:</b>	67.1 jph
<b>MTBF:</b>	25.2 min
<b>MCBF:</b>	30.3 cycles
<b>MTTR:</b>	1.8 min

Figure 3. System aggregates for the Body Shop

The question we would pose to a simulation expert would be how to statistically represent a detailed paint shop model as one “work center” in a three work center total plant model (body shop, paint shop and general assembly). Jobs per hour (JPH) is a metric used by many automotive companies who are required to use simulation to measure plant throughput. Given the mean and distribution of the JPH results from simulation Package B, GM desires statistically similar JPH results to come from Package A when the solution proposed by the simulation expert is used. That is, what 2D “substitution code” can effectively replace detailed 3D modeling so that the overall aggregate level throughput results and their distributions for the entire plant are valid.

System Aggregates	
<b>Speed:</b>	70.0 jph
<b>SAA:</b>	95.1 %
<b>SAT:</b>	66.5 jph
<b>MTBF:</b>	9.0 min
<b>MCBF:</b>	10.5 cycles
<b>MTTR:</b>	0.5 min

Figure 4: System aggregates for General Assembly

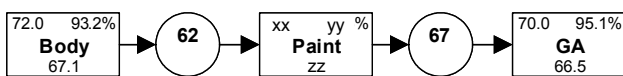


Figure 5: Total state model

### 3 SIMULATION MODELING APPROACHES

One may suggest that the authors 1) feed the exit times from the Body Shop into the Paint Shop model to drive it, and then 2) feed the exit times from the Paint Shop into General Assembly model to drive it. This approach is not desired for two reasons. It would likely prove to be to be a very time consuming exercise to create the IT mechanisms and interfaces required to efficiently link simulation Packages A and B so that they could effectively interoperate with such exit time data. Also, an integrated system that includes General Assembly means that a blocking effect may exist. That is, the exit times from the Paint Shops could be influenced or “blocked” when problems exist in the General Assembly area. Therefore, the strategy of separately feeding exit times would not capture all the complexities seen when viewing the system as an integrated whole.

A good question one may ask involves how we would go about verifying the correctness of our aggregated models? This represents a challenge to us because we would not want to validate each and every simulation when a new project needed completion. Ideally, we desire a process that allows us to obtain an accurate modeling representation for the whole plant without the need to revalidate results between projects. If a validation process were required, this would defeat the purpose of creating an aggregated model representing the entire plant because the time saved through aggregate modeling would be lost when the verification process was performed.

We desire a MCBF/MTTR combination to represent the whole shop (i.e. the body shop, the paint shop, and general assembly). In the case of the paint shop, we have provided the expert(s) with data representing exit times and inter-arrival times for vehicles. Perhaps this data can be used to determine an appropriate representative distribution for the paint shop. For example, a bi-model model may be the appropriate distribution represented by combining the more common distributions (e.g. the exponential) with their respective CBF/TTRs.

The simulation model for the entire plant should produce valid results when any smaller detail in any substation is changed. For example, when cycle time or downtime is changed for one robotic station in the paint shop, a new aggregate model representation can be obtained reflecting perhaps a different throughput result. Downtime can force the whole paint shop and ultimately the entire plant to stop.

### 4 QUESTIONS FOR THE SIMULATION COMMUNITY

General Motors desires to represent the entire Paint Shop with new input parameters which would form a station box for the Paint Shop similar to Figures 3 and 4. Figure 5

also depicts the desired station box input in the context of the Body Shop and the General Assembly station boxes.

**Question:** Given the provided data (a 1000-hours simulation of exit times and inter-departure times is given in seconds using modified data for confidentiality reasons), what is a reasonable method or process to obtain distributions that generates system aggregate data (See Figure 3 which is an example of system aggregate data for the Body Shop. We desire similar data for the Paint Shop.) representing the Paint Shop? In particular, what downtime distributions (MCBF and MTTR) represent the paint shop accurately? Can the reviewers comment on issues such as:

1. GM’s preferred approach is to represent the Paint Shop output in system aggregate terms compatible with our 2D simulation software. Is this approach advisable or is there a better approach?
2. The Paint Shop will have material transfer speeds that are slower at the front end and faster at the back end. Using one station box to represent this creates a difficulty--one box represents one speed although many speeds are present in the Paint Shop. Can a “one box” representation exist that yields valid results?
3. Is there a solution that fairly represents the possibility of multiple downtimes occurring during the simulation? In particular we discussed the blocking effect caused by downtime in General Assembly, and the subsequent downtime effect in the Paint Shop. Keep in mind that prolonged downtime in the Body Shop will also effect the “one box” solution. Is there a “one box” solution for the Paint Shop that can fairly represent the downtime delay that needs to be added to the simulated exit times?

## 5 COMMENTS FROM D.J. MEDEIROS

### 5.1 Introduction

The GM authors present problems encountered when modeling different components of a production system using different simulation languages. Difficulties arise when attempting to link the models to obtain an estimate of overall plant throughput. The current robust research and development work in HLA lends hope that interoperability methods will be available in commercial simulation software at some point in the future. There is clearly a need for such capabilities, especially if implemented in a manner that is easily accessible to modelers.

The GM authors approach the problem through model aggregation: high-level approximations of the detailed simulation models are employed to estimate overall plant throughput. This approach also has the benefits of

fast model run time and simplified experimentation due to the small number of parameters used to represent the system. The difficulty arises in attempting to reduce a very detailed model containing complex routing, controls logic, and conveyor systems to a few distributions.

Sub-system models are provided for the body and assembly shops. The models consist of stations (represented by blocks) and buffers (represented by circles). Each station has a standalone throughput, which appears to be calculated by multiplying jobs per hour by availability. Each buffer has a capacity, and presumably blocks the upstream station when full.

Each of the subsystem models is aggregated into a single block. The aggregate speed depends on the bottleneck station in the submodel; presumably the aggregate availability is obtained from running the detailed model. The system aggregates also include failure data, but the source of this data is not explained.

A similar aggregate block is desired for the paint shop. Its performance should be described by distributions representing throughput in jobs per hour, time between failures and/or number of cycles between failures, and repair time. Sample output data from the detailed paint shop model for a simulation run of 1000 hours was provided.

### 5.2 Characterization of the Data

Summary statistics for the provided dataset are shown in Table 1. The data represent time (in seconds) between departures from the system of interest. Figures 6 and 7 contain a histogram of selected data and a dotplot for the full dataset, respectively.

Table 1: Descriptive statistics for sample dataset.

Number of observations	64928
Mean	53.88
Standard deviation	37.28
Minimum	30.00
Maximum	2880.00
First quartile	35.00
Median	35.00
Third quartile	65.00

Clearly for much of the time the simulation model is outputting items at a rate of approximately 1 every 35 seconds, or 102.9 jobs/hr. (Note the first quartile, and median of the data.) There are a large number of short interruptions (see the third quartile of the data and Figure 6) and some extremely long interruptions (illustrated in Figure 7), leading to an overall output rate for the system of 64928 items in 1000 hours, or 64.9 jobs/hr.

One issue raised in the problem description is the possibility of using the exponential distribution to represent inter-departure times. The distribution of data was shifted to the left by subtracting the minimum and a probability plot was created for comparison to an exponential. The probability plot in Figure 8 clearly shows that the exponential distribution is not an appropriate choice.

Follow-up discussion with the authors revealed that inter-departure times are constrained by the characteristics of the conveyor system to multiples of 5 seconds. Figure 9 illustrates the most common values of inter-departure time; together these represent 95% of the dataset. Further analysis requires understanding what system characteristics cause this behavior (for example, there are no 40 second inter-departure times), and establishing if the same behavior would be present at different throughput or failure rates.



Figure 8: Probability plot comparing the dataset to an exponential distribution.

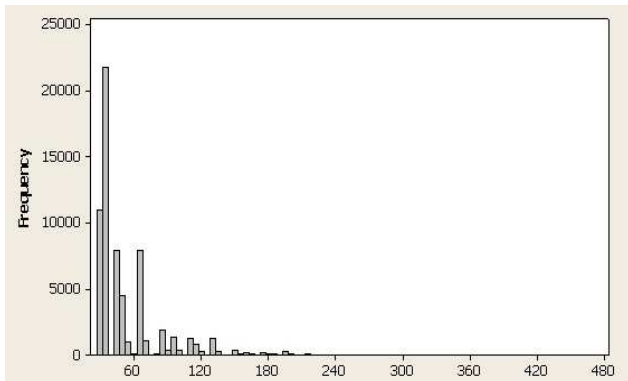


Figure 6: Histogram showing inter-departure times shorter than 500 seconds.

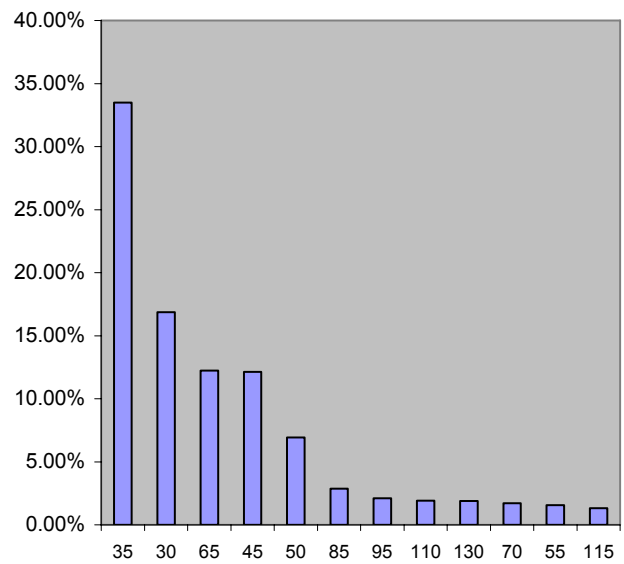


Figure 9: Largest values in the dataset.

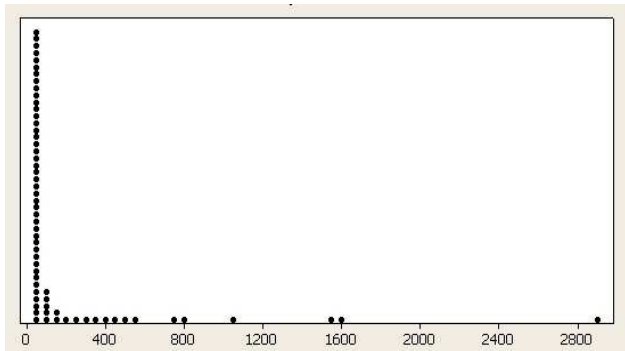


Figure 7: Dotplot for sample dataset. Each symbol represents up to 1315 observations.

Even if we are willing to ignore the fact that the data are integer valued, the exponential distribution is not a good choice to represent this data set. The very large spike at the left tail and the additional spikes shown in Figures 6 and 9 are not consistent with most commonly used distributions. If the highest probability values (30 and 35 seconds) are removed from the data set (under the assumption that they represent typical operation without downtime) the resulting data will still have the large spikes shown in Figure 9 and a left tail too heavy to be consistent with an exponential distribution.

### 5.3 Response to Questions Posed

The authors pose three specific questions at the conclusion of the paper; comments are provided below.

### 5.3.1 System Throughput

Question 2 concerns setting an appropriate speed for the aggregate block given that the front end of the paint shop operates at a slower speed than the back end. System throughput is constrained by the bottleneck operation in most manufacturing systems. There will be periods of time in which the system in question outputs at a higher rate, but the long-term throughput is limited by the speed and uptime of the bottleneck.

A question arises concerning the impact of these speed differentials on the buffers upstream and downstream of the paint shop, illustrated in Figure 5 of the paper. The input and output rates of the paint shop must be equal in the long run, but the variability in input times could be considerably different from the variability in output times, due to the differences in speeds and downtimes between the front and back end. If speeds and downtimes are significantly different between the front end and back end, incorrect estimates will be obtained for utilization of the upstream and downstream buffers and blocking due to buffer capacity.

### 5.3.2 Multiple Downtimes

Question 3 concerns the presence of multiple downtimes. A “single block” aggregate model implies that the downtimes specified reflect unavailability of the entire aggregated system. Thus, a downtime would imply that the entire paint shop was inoperable. If there are significant differences in downtime between the front end and the back end, a “single block” model will not be appropriate because it wouldn’t allow the front end to buffer work for the back end by continuing to operate when the back end is down. It may also cause overflows in the buffer upstream of the Paint station, thus blocking the Body station.

If the downtimes occur in different blocks of Figure 5 in the paper, the buffers between the operations will fill and cause blocking. This would automatically be reflected in the exit times from the blocked operations, assuming that the 2D simulation modeling tool used does indeed block when buffers are full.

### 5.3.3 Aggregate Model Approach

Question 1 is addressed to the advisability of using an aggregate representation for the paint shop. It seems unlikely that a “single block” aggregate model could adequately capture the behavior of the paint shop as it interacts with the upstream and downstream systems. However, it might be possible to create a satisfactory aggregate representation with a small number of operation blocks and buffers, for example Front End Paint, Buffer, Back End Paint.

If the 2D simulation software used for aggregate modeling requires that downtimes be exponentially distributed, further concern is warranted. This distribution might be a satisfactory approximation, depending on the needed accuracy of the results, but it cannot be justified based on the data provided, and thus model validity is of great concern.

## 6 COMMENTS FROM ONUR ULGEN

In the paper titled “Aggregating Subsystem Models into an Automotive Total Plant Throughput Model” Miller et al. discuss issues and raise several questions in aggregation of three detailed sub-models into one simple aggregate model with three processes connected serially. In what follows, we will first identify different levels of aggregation one can use in reducing the complexity of detailed models (Ulgen and Gunal 1998). Then, we will identify different approaches that one can undertake in capturing characteristics of detailed models so that these characteristics can be transferred to the aggregate models. This information will then we used in addressing the specific questions that the authors raise in their paper.

### 6.1 Aggregation Levels Of Models

The aggregation level of models are generally identified by the requirements of each company for simulation at the different phases of the manufacturing system development. In other cases, these levels are identified by the specific objectives of a one-time study required by management such as plant or enterprise level performance measures and in many cases for getting answers quickly to management. We identify five levels of aggregation below (one can have more or fewer based on the simplification levels required and if the company requires simulation models at the early stages of the manufacturing system design):

1) *Level Zero Model* is the highly detailed model which includes in a typical automotive assembly plant information such as individual work stations (with typical attributes MTBF, MTTR, setup time, preventive maintenance schedule, labor requirements), operators (skill level, multi-tasking capabilities, workstations assigned with priority, break schedule), material handling equipment used for the main product such as robots, turn tables, and conveyors (type, speed, MTBF, MTTR, end of shift and end of day policies), buffers (capacity, accumulating or non-accumulating, banking logic), shifts (policies, starting and ending logic). These models when used for each subsystem generally assume that the product is arriving to the subsystem at a fixed rate (e.g., 72 jph with no variability) and never starving the system while the jobs that exit the subsystem can always leave it because it is

commonly assumed that there is unlimited buffer after the last station.

2) *Level One Aggregate Models* typically assume that labor and material handling equipment details be left out from the model completely. Conveyors at this level can be assumed to be operating like multi-part processing workstations with a fixed capacity. Each workstation is still kept but with fewer attributes (only with cycle time, MTBF, MTTR) and the same applies for buffers and shifts details.

3) *Level Two Aggregate Models* generally combine several serial workstations that generally perform similar function into one workstation with a composite capacity, processing time and MTBF and MTTR. These workstations generally have one part for storage or none between each other. Larger buffers between the composite stations are generally modeled with their capacity attribute and the shifts with their shift length and the number of shifts per day and week that the line operates.

4) *Level Three Aggregate Models* generally combine several composite workstations into one sub-area model with a composite sub-area capacity, processing time and MTBF and MTTR. For example, in the body shop, all the underbody operations can be identified as a sub-area while in the general assembly operations, trim can be a separate sub-area. Buffers and shifts are modeled very similar to the Level Two Aggregate Models.

5) *Level Four Aggregate Models* combine all the operations in a subsystem into one sub-model with one composite workstation which has the composite capacity, processing time, MTBF and MTTR of the whole subsystem. In some cases, the modeling approach may require up to two additional dummy stations to be added to each subsystem to represent starving and blocking conditions caused by the predecessor and successor subsystems.

Note that level zero and level four models are the detailed subsystem and aggregate models discussed in the paper by Miller et. al.

## 6.2 Approaches For Incorporating Characteristics Of Detailed Models Into Aggregate Models

Miller et. al. raise several issues in incorporating the characteristics of detailed models into aggregate models, namely; (a) starving from the predecessor subsystem and blocking from the successor system, (b) interfacing issues in directly using the exit times from the predecessor subsystem as arrival times to the successor subsystem (actually to the buffer between the two subsystems), (c) representing different speeds that exist in the segments of the line in a detailed model appropriately in the aggregate model, (d) representing multiple downtimes that occur in the detailed model as one or multiple downtimes in the aggregate sub-model, and (e) representing (a) through (d) in combinations. Before discussing these issues in detail,

I would like to discuss several general approaches that one can use in incorporating the characteristics of detailed models into aggregate models.

a) Approaches for incorporating starving from the predecessor subsystem and blocking from the successor system:

a.1) In the first approach, actual exit and blocking times are recorded from the detailed subsystem models and re-used in running the other subsystem models. Here are the detailed steps of this approach:

a.1.1) Run the first subsystem detailed model and record all the exit times of jobs from it.

a.1.2) The second subsystem detailed model reads these directly to attempt to bring the jobs to the subsystem using a dummy workstation. If the subsystem is blocked, the jobs will wait FCFS until they can enter the subsystem. The second subsystem detailed model exit times are also stored in a separate file.

a.1.3) The third subsystem detailed model similarly reads the entry times of the jobs using a dummy workstation and whenever a job cannot enter the third subsystem due to internal blocking, blocking intervals are recorded in a separate file.

a.1.4) The second subsystem detailed model is now run with both the exit times from the first subsystem and blocking times from the third subsystem. During the blocking times of the third subsystem, the jobs cannot exit the second subsystem. Blocking times of the second subsystem are also recorded in a separate file.

a.1.5) The first subsystem detailed model is run for the final time with the blocking times of the second subsystem. The new exit times are recorded in a separate file. This is the final output characteristics of the first subsystem.

a.1.6) The second subsystem detailed model is run with the new exit times from subsystem one and old blocking times from subsystem three. The exit times are recorded in a separate file. This is the final output of the second subsystem.

a.1.7) The third subsystem detailed model similarly is run with the latest exit times of the second subsystem. Blocking intervals are recorded in a separate file. This is the final output of the third subsystem.

Note that one can run few more iterations of the above steps until the results are within a certain confidence interval but it is assumed here that after blocking is incorporated to the results of the first subsystem, the exit times will be stabilized relatively quickly.

a.2) In the second approach, rather than reading the exit times and blocking times from files, one can analyze the distributions of interarrival times and inter-blocking times. If these times are independent, one can develop distributions for them and use the distributions to generate arrivals and blockings. On the other hand, if they are de-

pendent, one can develop ARIMA (Box-Jenkins) type models to generate the arrivals and blockings.

Note that the final files of arrival (exit) and blocking times or distributions or ARIMA models representing those times can be used in either detailed or aggregate models of the subsystems.

b) Interfacing issues in directly using the exit times from the predecessor subsystem as arrival times to the successor subsystem:

This was never an issue in the models that we used as storage space is cheap and any simulation software can read from files very easily. It somewhat slows the simulation execution but it was insignificant in the runs we have made.

c) Representing different speeds that exist in the segments of the line in a detailed model appropriately in the aggregate model:

The simplest way to solve this problem will be to use another level of aggregation than Level Four Aggregate Model for the Paint Shop such as Level Three or Level Two. We generally use Level Two Aggregate Models as the aggregate models for all the subsystems as the run time of such models are very reasonable with any simulation software and we can also provide detailed results to the management with a good understanding of the causalities in the system (e.g., bottlenecks that have moved).

d) Representing multiple downtimes that occur in the detailed model as one or multiple downtimes in the aggregate sub-model:

Of course if Level One through Level Three Aggregate Models are used as the aggregate model, this may not be an issue. On the other hand, if there are major events that take place in the subsystem that are easy to identify such as preventive maintenance or lunch break that are predictable and shut the whole subsystem down for a while, they can be scheduled at different frequencies for the composite workstation that represents that subsystem.

e) Representing (a) through (d) in combinations (e.g., multiple downtime effects and blocking effects at the same time:

Please see solution described in (a.1) above as all the interdependence is considered fully when the detailed models are used iteratively to capture the final characteristics of the aggregate models.

### 6.3 Conclusion

Aggregate models of manufacturing systems are very useful in the development stages of new programs in Automotive Industry. Highly simplified aggregate models require complex techniques to incorporate dependencies within and among the subsystems and it is therefore recommended that moderately detailed aggregate models should be used in such cases. In this paper, different levels of moderately complex aggregate models are sug-

gested to provide efficient execution time for the models while incorporating the essential dependencies within and among the subsystems.

### ACKNOWLEDGEMENTS

The GM authors would like to thank John Carson for his efforts in improving the clarity of this paper. We thank Robert Dych and Don Lyons for their contributions towards helping us understand the problem. Onur Ulgen would like to thank Ed Williams and Steve Beeler for their efforts in improving the clarity of his commentary and sharing their experience in aggregate modeling approaches.

### REFERENCES

- Park, Young H., Jack E. Matson, and David M. Miller. 1998. Simulation and Analysis of the Mercedes-Benz all Activity Vehicle (AAV) Production Facility. In *Proceedings of the 1998 Winter Simulation Conference*, eds. D.J. Medeiros, E.F. Watson, J.S. Carson and M.S. Manivannan, 921-926.
- Shin, Frank, Bala Ram, Aman Gupta, Xuefeng Yu, and Roland Menassa, A Decision Tool for Assembly Line Breakdown Action, In *Proceedings of the 2004 Winter Simulation Conference*, eds. R. G. Ingalls, M. D. Rossetti, J. S. Smith, and B. A. Peters, 1122-1127.
- Ulgen, Onur M., and Ali Gunal. 1998. Simulation in the Automobile Industry. In *Handbook of Simulation: Principles, Methodology, Advances, Applications, and Practice*, ed. Jerry Banks, pp. 547-570. New York, New York: John Wiley & Sons, Incorporated.
- Williams, Edward J., and Haldun Çelik, Analysis of Conveyor Systems within Automotive Final Assembly. In *Proceedings of the 1998 Winter Simulation Conference*, eds. D.J. Medeiros, E.F. Watson, J.S. Carson and M.S. Manivannan, 915-920.

### AUTHOR BIOGRAPHIES

**JEFFREY SCOTT MILLER**, is a Senior Throughput Simulation Engineer at General Motors Technical Center in Warren, MI. He is responsible for facilitating the development of global common simulation processes for GM regions world wide. Jeff was previously Vice President of the Detroit Chapter of IIE, and is currently a member of the steering committee for the Michigan Simulation Users Group. He received his B.S. and M.S. in Industrial Engineering from the University of Michigan. Jeff can be reached via e-mail at [<Jeffrey.1.miller@gm.com>](mailto:Jeffrey.1.miller@gm.com).

**RANDY COMBS**, is currently the Engineering Group Manager of the Throughput Simulation Studies group at



General Motors. Randy has 32 years of work experience with GM and has held various positions in Assembly Plant Industrial Engineering and Manufacturing Engineering at the GM Technical Center in Warren, Michigan. For the last 15 years Randy has been the Manager of Throughput Simulation activities and has, along with other Throughput Simulation Managers, developed common business processes to meet GM global requirements. He is presently the manager responsible for all Throughput Simulation Studies for North America while supporting other GM regions globally. Randy has been certified to teach the Theory of Constraints within GM by the Goldratt Institute, and he is the Co-Chair of the General Motors Global Throughput Simulation Experts Team. He has been an active board member in the Michigan Simulation User Group for several years and holds a B.S. in Business Administration and an M.A. in Industrial Management. He can be reached via e-mail at [<Randy.Combs@GM.com>](mailto:Randy.Combs@GM.com).

**EARNEST FOSTER**, is a Senior Researcher with General Motors Research and Development Center in Warren, Michigan. His interests include multivariate statistical process control, statistical process control for time variables, and process-oriented approaches to variation reduction. He holds a Ph.D. in Industrial Engineering from the Pennsylvania State University. Earnest can be contacted at [<Earnest.Foster@gm.com>](mailto:Earnest.Foster@gm.com).

**JEFFREY TEW**, is a GM Technical Fellow and Group Manager of the Manufacturing Enterprise Modeling Group in the Manufacturing Systems Research Lab at General Motors' R&D Center in Warren, MI. Currently, Dr. Tew is an Adjunct Professor of Supply Chain Management at the Georgia Institute of Technology and a Visiting Professor of Industrial Engineering at Tsinghua University in Beijing. He was Coeditor of the Proceedings of the 1994 Winter Simulation Conference and is a past President of the INFORMS College on Simulation. He is the General Chair of the 2007 Winter Simulation Conference. He received a B.S. in mathematics from Purdue University in 1979, an M.S. in statistics from Purdue University in 1981, and a Ph.D. in industrial engineering from Purdue University in 1986. His current interests include the application of operations research and information technology tools to large-scale logistics (supply chain) systems and e-commerce. He is a member of Alpha Pi Mu, ACM, ASA, IIE, The Institute for Mathematical Statistics, INFORMS, SCS, and Sigma Xi. He can be reached via e-mail at [<jefftew2002@yahoo.com>](mailto:jefftew2002@yahoo.com).

**D.J. MEDEIROS** is an Associate Professor in the Industrial and Manufacturing Engineering Department at Penn State. Her research interests include simulation of manufacturing and health care systems, manufacturing system

control, and CAD/CAM. She holds a BSIE from the University of Massachusetts at Amherst and an MSIE and Ph.D. from Purdue. She has served as track coordinator, proceedings editor, and program chair for WSC.

**ONUR ULGEN**, PhD, is the President of PMC as well as a Professor of Industrial and Manufacturing Systems Engineering at the University of Michigan-Dearborn. Onur has published more than fifty papers, book chapters, and manuals on simulation, scheduling, operations management, and lean systems. He has more than twenty-five years of experience in applying simulation in Automotive and other industries. Currently he is involved in writing a series of books in applying different simulation tools to solve manufacturing and process problems. In 1980's Onur was involved in developing one of the first object-oriented simulation tools based on Smalltalk OOP software. Onur holds a BSME from Bogazici University, and MSc and PhD degrees in Industrial Engineering from Texas Tech University. Onur founded PMC in 1985 which grew into the largest simulation services company in the world with more than 100 engineers on its staff and offices in Dearborn (MI), Warren (MI), Austin (TX), Los Angeles (CA), Windsor (CAN), Sweden, India, and Turkey. Onur had been teaching at the University of Michigan-Dearborn since 1979 at the Industrial and Manufacturing Systems Engineering Department. He is a member of the Institute of Industrial Engineering (IIE), American Production and Inventory Control Society (APICS) and a founding member of the Michigan Simulation Users Group (MSUG). He can be reached via e-mail at [<ulgen@pmcorp.com>](mailto:ulgen@pmcorp.com) or [<ulgen@umich.edu>](mailto:ulgen@umich.edu).