Proceedings of the 2015 Winter Simulation Conference
L. Yilmaz, W. K. V. Chan, I. Moon, T. M. K. Roeder, C. Macal, and M. D. Rossetti, eds.

TOWARDS A VIRTUAL FACTORY PROTOTYPE

Sanjay Jain

The George Washington University
2201 G Street NW
Funger Hall, Suite 415
Washington, DC- 20052, USA

David Lechevalier
Jungyub Woo
Seung-Jun Shin

Systems Integration Division
Engineering Laboratory
National Institute of Standards and Technology
Gaithersburg, MD – 20899, USA

ABSTRACT

A virtual factory should represent most of the features and operations of the corresponding real factory. Some of the key features of the virtual factory include the ability to assess performance at multiple resolutions and generate analytics data similar to that possible in a real factory. One should be able to look at the overall factory performance and be able to drill down to a machine and analyze its performance. It will require a large amount of effort and expertise to build such a virtual factory. This paper describes an effort to build a multiple resolution model of a manufacturing cell. The model provides the ability to study the performance at the cell level or at the machine level. The benefits and limitations of the presented approach and future research directions are also described.

1 INTRODUCTION

Progress towards achieving the vision of smart manufacturing systems requires the abilities to conduct detailed analytics on current performance, evaluate potential future courses of actions, and set the course that best leads towards the goals. These abilities can be respectively termed as diagnostic, predictive and prescriptive analytics. Diagnostic analytics assesses past and current performance and cause and effect relationships among major control factors and performance metrics. Predictive analytics evaluates future performance of a system operating under selected policies and forecasted requirements such as demand scenarios. Prescriptive analytics helps develop future courses of actions using approaches such as optimization and combined simulation-optimization. The efforts to move towards smart manufacturing thus need to be supported by diagnostic, prescriptive and predictive analytics (Shao, Jain, and Shin 2014).

Jain and Shao (2014) proposed the virtual factory, a high-fidelity simulation of the manufacturing system, to support data analytics. The term virtual factory has been used with multiple meanings in the research and professional literature. We utilize the definition of the virtual factory as “an integrated simulation model of major subsystems in a factory that considers the factory as a whole and provides an advanced decision support capability.” Other terms used to describe the concept with minor variations include digital factory, virtual copy, and virtual plant model. The latter two terms have been used in the description of the recent Industrie 4.0 concept by Mario et al. (2015). They define cyber physical systems (CPS) as a key component of Industrie 4.0 because they utilize virtual copies of the physical world to support decentralized decision making.

The virtual factory concept encompasses the ability to analyze the manufacturing system at any desired level of detail just as one would have for a real factory. One should be able to focus on 1) a single process step and analyze the performance of the associated equipment, 2) a particular line or department in the system, or 3) the factory as a whole. The virtual factory concept is represented in Figure 1. The figure shows models of manufacturing system at multiple levels of resolution extending from factory level at the top to device level at the bottom. These models should be integrated vertically across the hierarchy and horizontally with input data sources and output data analytics systems. Please see Jain and Shao (2014) for more details of the concept and the figure.

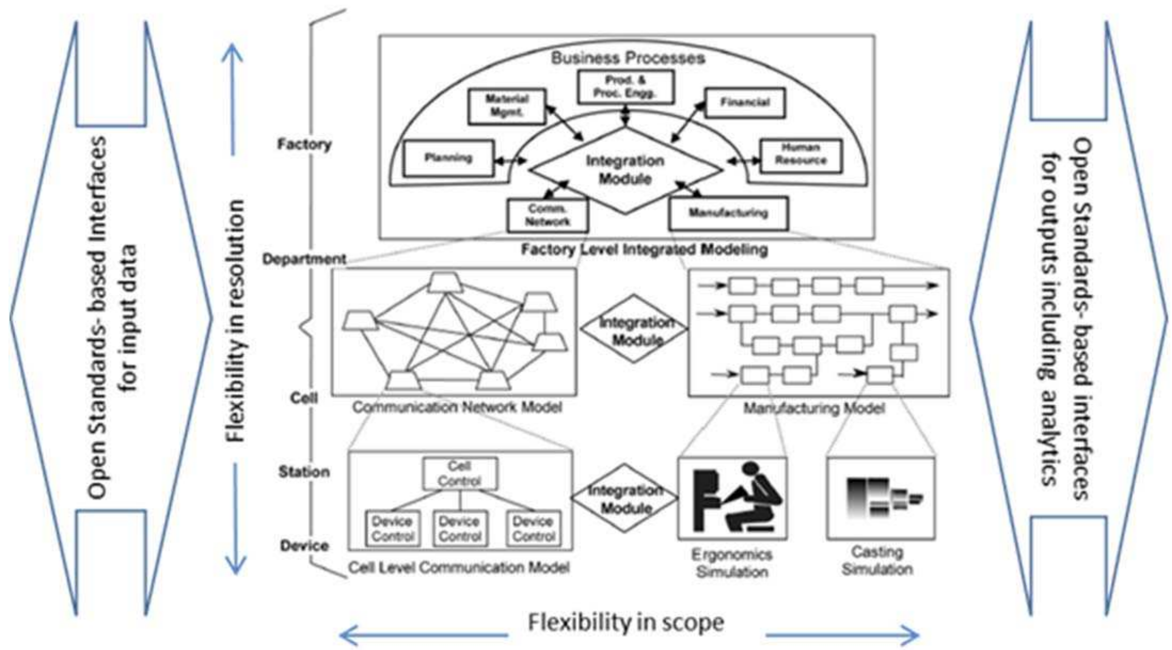


Figure 1: Virtual Factory concept (adapted from Jain and Shao 2014).

Developing virtual factories that correspond to real factories can be a large undertaking particularly if each such virtual factory is custom developed. One way, and perhaps the only way, to realize the virtual factory concept is via "crowd sourcing." That is, a number of researchers need to contribute to build the models of sub-systems and atomic components in a way that can be integrated to realize the concept. The models should be generic with the capability to customize them based on data describing the sub-systems and atomic components. Interested researchers should come together to 1) define an overall open architecture supported by relevant standards and 2) develop open modules that can be integrated to realize a specific virtual factory. Such an effort can take several years to come to full fruition. The capability can be built in stages targeting more common manufacturing system configurations first. Even partial implementations of the virtual factory can be useful to industry. Indeed, industry has been using models of individual sub-systems to support manufacturing system design and operations as evidenced by papers presented in this conference over the years. Efforts such as Industrie 4.0 appear to be taking a similar approach for an even wider scope.

The groundwork to support any such scope includes defining, and exploring the feasibility of, the key aspects of the virtual factory concept. The brief description above suggests that any kind of feasibility checking requires multi-resolution modeling, the ability to model parts of a system at varying levels of detail. For example, one should be able to model a machine of interest in detail at the unit level or as part of a higher level system. Our research goal is to build a prototype of a virtual factory that can be used to assess the feasibility of any proposed key aspect. This will allow other researchers to assess the feasibility of their proposed key aspect of the concept.

This paper represents a small step towards building a complete virtual factory prototype by exploring what capabilities that prototype needs to estimate the feasibility of multi-resolution modeling. Our research used a limited scenario - a small job shop with a single manufacturing cell comprising four turning machines. Our virtual prototype captures this scenario at three levels of detail. The top layer, the cell, has a model can that tracks the processing of each part as a single block of time. Typically, a cell model is implemented in a discrete event simulation (DES). At the machine level, each machine can be modeled at a greater granularity level of detail to 1) track the granular movements needed to process the part and 2) predict characteristics such as temperature and energy use. Typically, a machine model is implemented using the agent-based simulation (ABS) paradigm.

The next section of the paper provides a brief literature review. Section 3 presents the proposed approach for developing the virtual factory and for the multi-resolution modeling for the small job shop model. The implementation of the small job shop model with the three levels of detail and identified issues are discussed in section 4. Section 5 concludes the paper with discussion of next steps.

2 LITERATURE REVIEW

This section briefly reviews the recent literature in relevant areas, which include virtual factory, multi-resolution modeling, and hybrid simulation.

2.1 Virtual Factory

Jain and Shao (2014) provided a brief overview of virtual-factory literature. A few additional efforts employing the virtual factory concept have been reported since then. Yang et al. (2015) emphasize the use of virtual reality for collaborative development of virtual factory. They present three application scenarios, one each at production-system, production-cell, and workstation levels. The granularity of detail varies from level to another. For, example, information about cutting tools and workpieces is taken account at the workstation level but not at the cell level. The three applications reported by Yang et al. (2015) do not appear to have the flexibility of combining different levels of details in the same model and thus the effort has not fully implemented multi-resolution modeling.

Mourtzis et al. (2015) report on the increasing use of simulation in conjunction with digital manufacturing. The combination of simulation and digital manufacturing will lead towards a capability that is close to virtual factory per the definition used in this paper. Terkaj and Urgo (2015) describe a Virtual Factory Data Model (VFDM) to support the development of the virtual factory model. They also describe a connector that automatically generates a simulation model based on the VFDM description. All these efforts appear to be aiming for the similar goal of realizing the vision of virtual factory while addressing different aspects. The aspect of multi-resolution modeling, the focus of this effort, does not appear to have been addressed in these efforts.

2.2 Multi-Resolution Modeling

Multi-resolution modeling appears to have received more attention in the context of combat simulation than in the context of manufacturing. Hong and Kim (2012) identify two major challenges in multi-resolution modeling: seamless data aggregation and disaggregation, and dynamic replacement of models in different resolutions. They develop a specification to address these challenges and show its application in an air combat scenario. Guan et al. (2012) propose a framework for digital factory technology that includes both multi-level modeling and multi-resolution simulation. They utilize a distributed simulation framework to integrate simulations of process, plant layout, and supply chain. They demonstrated the use of this framework in a case study that addresses the integration of a material handling simulation with a virtual reality model for static layout analysis.

Jain et al. (2013) utilized multi-resolution modeling of a supply chain. The high-level, supply-chain model is developed using a system dynamics simulation (SDS) paradigm with the ability to execute one of the manufacturing nodes at more detail using discrete event simulation (DES). The effort reported in this paper seeks to implement the idea within the virtual factory context with integrated modeling of cell and equipment levels.

2.3 Hybrid Simulation

Multi-resolution modeling often involves modeling different levels of abstraction using different simulation paradigms and thus can be viewed as hybrid simulations. For example, as mentioned above Jain et al. (2013) utilized SDS at the supply-chain level and DES at the factory level. Hermann et al. (2011) combined discrete event simulation to model manufacturing processes with a continuous simulation to model the energy flows for planning manufacturing systems with consideration of environmental impact. Fakhimi et al. (2014) utilized a hybrid of agent-based simulation (ABS) and DES for strategic planning and simulation analytics of health care services. In their work, the two simulations interact to improve the performance of the system. The effort reported in this paper also utilizes an interaction between ABS and DES to implement multi-resolution modeling.

3 APPROACH

The approach is discussed in two sub-sections. First, the overall proposed approach for creating virtual factories is discussed. This is followed by discussion of the approach used for implementing multi-resolution modeling in a small prototype.

3.1 Overall Approach for Virtual Factory

Developing a full scale virtual factory will be difficult for most organizations to take on by themselves. We propose an approach that allows multiple participants – individual, groups, and organizations to develop modules that can be integrated to create the virtual factory. This approach would first require development of an open architecture based on standards that allows integrating modules for modeling virtual factories. The Industrie 4.0 effort mentioned in Section 2 includes the goal of developing virtual versions of real factories through a large coordinated effort (Mario, Tobias, and Boris 2015). It appears to be targeting a standard architecture and thus may provide an opportunity to integrate other independently developed modules.

The capability to develop virtual factories will be realized primarily using software. This presents an opportunity to develop the capability iteratively starting from a prototype and successively adding capabilities. The needed concepts, standards, and interfaces can be tested as corresponding capabilities are developed. As suggested earlier, such iterative development can be done by multiple participants on various sub-systems and components of the virtual factory related to their interest and applications.

Development of software by multiple participants in an open community requires common understanding and agreement on several aspects including scoping of constituent modules, selection of standards, and selection of applicable ontologies. The alternatives for each aspect need to be carefully explored and considered. It will help significantly, and may indeed be required, to develop prototypes exploring the alternatives for at least the major aspects to capture the issues involved and associated advantages and disadvantages. Prototypes would also help communicate the long term vision and serve to capture feedback from the end users. An initial push towards development of the virtual factory can occur via developments of prototypes exploring different aspects by multiple interested researchers and associated discussions at forums such as simulation conferences.

The development reported in this paper is an initial prototype that explores the idea of multi-resolution modeling in the context of a virtual factory. It considers three levels of resolution, a process level, a machine level and a manufacturing cell level. The three levels are implemented in the same simulation software to keep the focus on the issues in integrating multiple levels of resolution. Implementing the three levels in different simulation software would have required a mechanism to synchronize executions such as distributed simulation and would have added another layer of complexity.

3.2 Approach for Multi-Resolution Modeling

Multi-resolution modeling (MRM) requires the capability to execute different parts of a model at different levels of resolution. It is noted that hierarchical levels in manufacturing context have been defined for decades (e.g., Jones and Mclean, 1986; Williams, 1994) and have been recently captured in standards such as IEC 62264-3 (ISO, 2013). Unfortunately it appears that there isn't one widely accepted standard definition of such levels. The hierarchical levels are generally defined with the idea of control and may not correspond with the software applications that implement that control. To gain acceptance from industry users, the levels in virtual factory will need to be set up to match the standards that have wider acceptance than others. The virtual factory will also need to have the flexibility to modify level definition to match hierarchies defined in other official and de-facto standards.

The lower levels of the manufacturing control hierarchy may be defined to include a manufacturing cell level, followed successively by machine/equipment and process levels. The prototype reported in this paper represents these three levels with modeling of 1) physics of the process with time modeled in milliseconds, 2) operations at machine level, with events occurring every few seconds, and 3) functions at the cell level, with events occurring in the range of every few minutes.

In addition to the time granularity, the three levels are different in other ways. The implementation of the three levels makes certain scoping decisions. The machine level operations treat a batch as a collection of individual parts and track batch loading, individual part set-up, execution of turning process on individual parts, followed by part unload and repeating of this cycle for all parts in the batch. A batch unload step is modeled after all parts have been processed. While most of the actions are modeled in discrete event paradigm, the actual turning process is represented in continuous time in the process level model. At the manufacturing cell level the batch is treated as a single item and processing times are modeled accordingly using discrete event paradigm.

An alternate implementation may model times for processing of individual part features at the machine level and time for processing the entire part at manufacturing cell level. The prototype thus allows exploring and highlighting some of the scoping options. Alternate assumptions and/or selections can be made in other prototype efforts or even in future version of this prototype based on inputs from other researchers and practitioners.

The three levels have been implemented using a bottom-up approach. The process level model was developed first and calibrated against real machines that were instrumented to capture the measures of interest. The machine level models was developed next and validated against the real machine data. The validated virtual machine models were executed multiple times and the resulting batch processing times were captured. The batch processing times are computed using the start of batch set-up to end of batch unload. Therefore, it includes multiple cycles of individual part set-up, processing and unload times. These batch process times are used to model the machine operations at the manufacturing cell level. The user is provided an option to model selected machines at the machine level while the rest of the cell can be modeled at the manufacturing cell level. Of course, the user can run the entire cell with all machines modeled at machine level of detail and they can run the entire cell with all machines at manufacturing cell level of detail.

The current prototype represents batch processing times with the assumption of the times being normally distributed. The collected individual batch times are analyzed to determine the means and standard deviations and recorded for use in manufacturing cell level execution. In future, more advanced curve fitting analysis will be used to identify and select distributions that most closely represent the collected times.

4 IMPLEMENTATION

This section describes the implementation of the prototype using a simulation environment. The process level model is briefly discussed first. The development of the machine level model using agent-based simulation (ABS) is discussed next together with the capabilities to execute it in summary or detailed mode. Next the development of the manufacturing cell level using discrete event simulation is presented that integrates ABS model and the capability of execution in summary or detailed mode.

4.1 Process Level Model

The process level model is an implementation of the virtual turning machining model that was developed to simulate machining process based on process planning data (Shao, Jain, and Shin 2014). It utilizes discretized continuous equations that represent the physics of the process dynamics and kinematics of a machine tool. It models machine components such as the spindle motor and servo motors, parameters such as depth of cut and feed rate to determine cutting forces and the resulting energy and time consumption. The inputs to the simulator are machine parameters and process planning data in STEP-NC format (ISO 2007). The outputs are generated in format compliant with the MT-Connect standard (MTConnect 2014) and include parameters such as time and energy consumption. For the current MRM prototype, only the time values are passed to the machine level model. In near future, other parameters in particular the energy consumption will be passed and aggregated at higher levels.

The process level model was originally developed in C++ and transformed to Java for ease of integration with the machine level model developed in AnyLogic (Grigoryev 2015). The machine level model is described next.

4.2 Machine Level Model

The machine level model has been implemented as an agent utilizing the Agent-based Modeling constructs in AnyLogic. Specifically the model has been implemented using the Statechart construct of the Agent palette in AnyLogic to mimic the modeled states of the machine as shown in Figure 2. The default machine state is the *Idling* state. During the simulation, the machine stays in this state as long as it does not get any batch to process. As soon as a batch arrives (represented by transition 1 in Figure 2), the machine goes to the *batchSetup* state that models the machine set up for processing the batch. The following sequence of states depends on the level of detail being modeled.

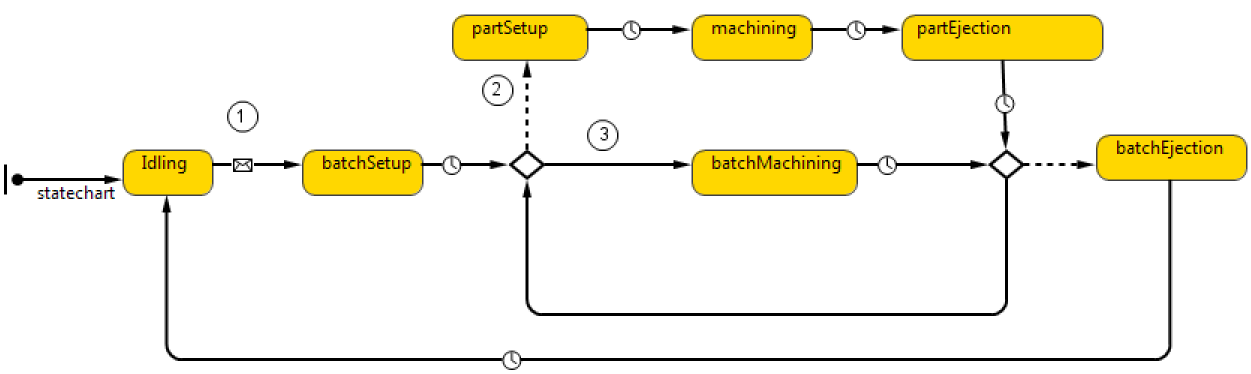


Figure 2: State chart for the machine level model.

If the machine is running in detailed mode (transition 2) representing the machine level of detail, the next state is the *partSetup* state where the machine sets up each part in order to execute needed operations. The corresponding functions configure machine parameters depending on the material that has been set up in the previous state. These parameters include feed rate and spindle speed. Following the completion of *partSetup,* the state transitions to *machining* state that represents the metal cutting process. After the machining state, the machine goes to the *partEjection* state that models unloading the part. The logic loops through the states as many times as there are parts in the batch. The times to process one part, that is the transition from beginning of the *partSetup* state to the end of the *partEjection* state are recorded and used to calculate the average and standard deviation of the part processing time.

The process level model in section 4.1 models the detailed steps for all the states that have corresponding STEP-NC instructions. The process level determines the times required for execution of the STEP-NC instructions and passes it back to the statechart to model the passage of time. The structure allows modification or even replacement of the process model without affecting the machine level or the higher level models. The STEP-NC source file is customized using the machine parameter values generated in the *partSetup* state. With this file as input, a specific function of the process level model models the cutting process and determines the machining time to process one part of the batch.

If the machine is running in the summary mode representing the manufacturing cell level, the state chart goes directly to the *batchMachining* state (transition 3). This path summarizes the other path by modeling the processing of the entire batch at one time using the average and standard deviation of the individual part-processing times. The concept of Central Limit theorem is used to aggregate the individual part process times into batch process times. Generally, minimum 30 data points are recommended for application of Central Limit theorem (Berenson, Levine, and Krehbiel 2002) and this criteria will be implemented in the model. This is admittedly a simple approach. Future versions of the prototype may allow more options such as empirical representation and fitted continuous distributions. Finally the last state is the *batchEjection* state, when the batch unload step is modeled. The time for processing successive batches are recorded and can be used for analysis such as aggregating them for representing the process at a further lower resolution such as line or plant level.

4.3 Manufacturing Cell Level Model

The manufacturing cell level model has been developed using discrete event simulation capability of AnyLogic as shown in Figure 3. The manufacturing cell is composed of four turning machines that are represented using the process modeling library provided in AnyLogic.. A S*ource* node generates part batch arrivals following a uniform distribution between 6 and 8 per hour. Each batch can contain ten to fifteen parts and the parts can be in aluminum, steel or titanium. The batch is sent to a *Queue* and then to an object called *SelectOutput* that chooses the machine to which the batch is routed. The S*electOutput* utilizes the shortest queue dispatching rule for this decision.

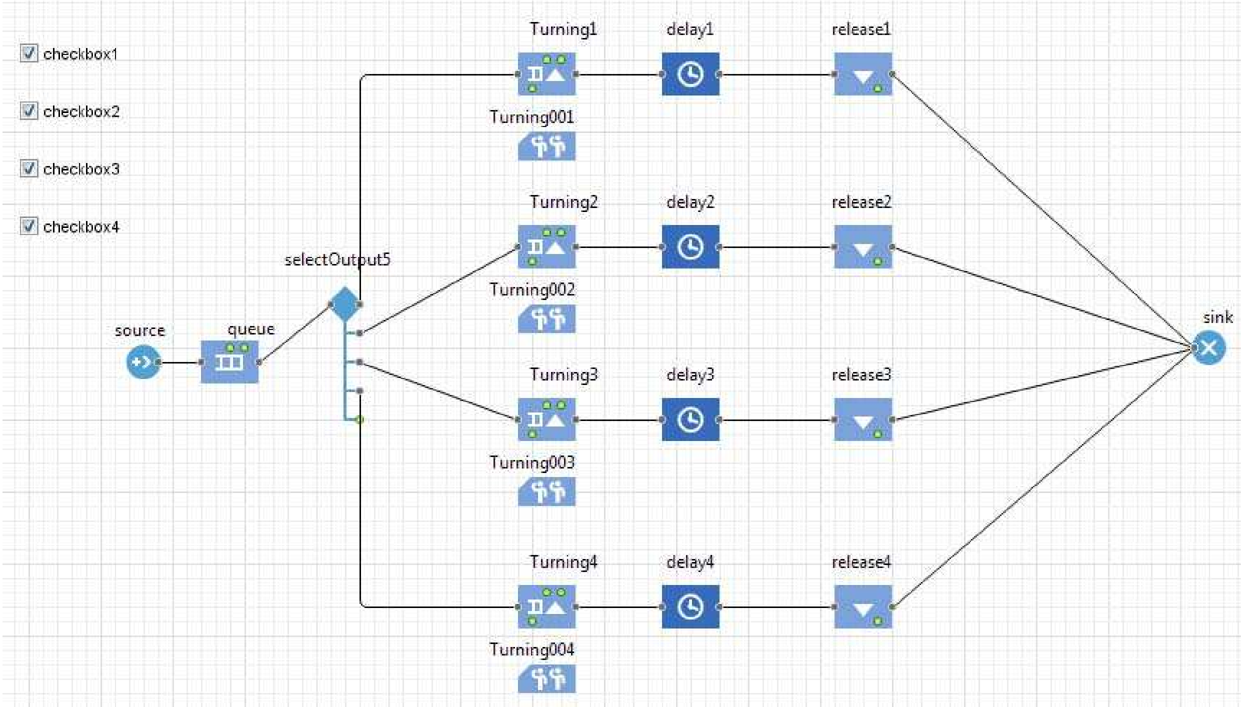


Figure 3: Manufacturing cell level model represented using discrete event simulation.

The machines are represented using a *ResourcePool* object (called Turning001, Turning 002, etc…) in the figure. A *ResourcePool* object can comprise of a number of resources and allows the facility of the resources being agents. In this model, each *ResourcePool* includes a turningMachine agent as the resource. This structure allows linking the manufacturing cell level model represented using the process modeling library to the machine level model represented using the agent library. Again, the machine level model can be replaced easily without impacting the manufacturing cell level model.

The processing of the batches by machines is modeled using the sequence of *Seize*, *Delay*, and *Release* objects. The *Seize* objects have been named as machines Turning1, Turning2, etc. in Figure 3. The arrival of the batch at the machine, i.e., on the *Seize* step, triggers the arrival of the batch on the *batchSetup* state of the corresponding machine agents state chart discussed in the preceding sub-section. The processing of the batch is modeled in the machine using the agent-based model. During this time, the batch is held in the *Delay* object at the cell level. Once the unit has processed the full batch, the agent sends the signal to release the batch from the *Delay* object. The batch is released and its exit from the cell is modeled via the *sink* object. The corresponding machine agent goes to the idling state at the agent-based model level. The user can specify the choice of resolution level for each machine at the manufacturing cell level using the checkbox on the left. Each checkbox is associated with one machine. Depending on the choice made by the users, the batch will choose either the machine level of detail (detailed) path or the manufacturing cell level of detail (summary) path in the state chart described in the previous sub-section.

4.4 Execution at Multiple Resolution Levels

The implementation of the prototype model allows executing the simulation at multiple resolution levels as listed below.

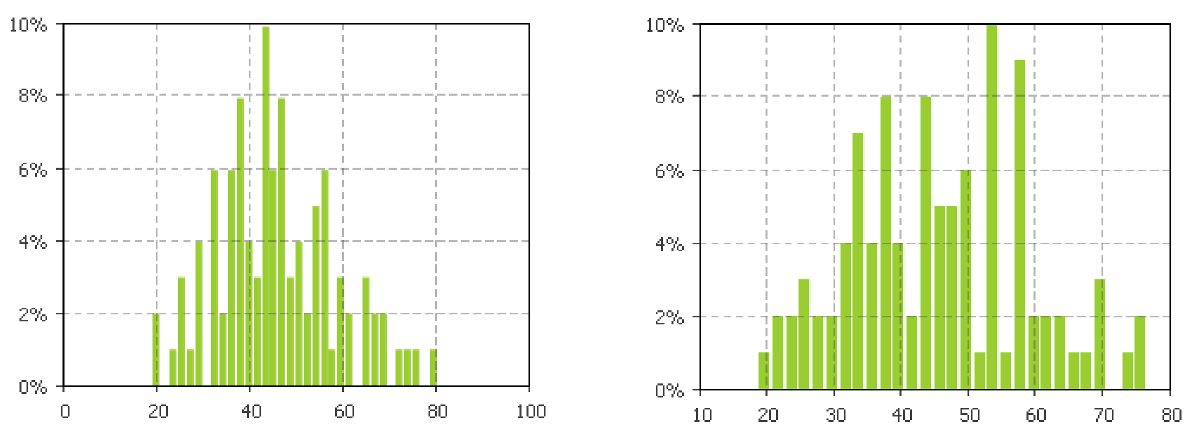
- The manufacturing cell can be modeled with all machine models executing at manufacturing cell level, that is, with processing modeled for entire batch at a time.

- The manufacturing cell can be modeled with all machine models executing at the machine level, that is, with processing modeled at individual part level complete with determination of time and energy consumption based on the physics of the process.
- The manufacturing cell can be modeled with user-selected machines executing at the manufacturing cell level and other machines executing at the machine level of detail.

The capability of executing the model at multiple resolution levels is available via the checkboxes provided at the manufacturing cell level. The checkboxes allow the user to allow execution in the default machine level of detail (detailed) or select manufacturing cell level of detail (summary).

The current implementation of the model is set up to allow the selection of the resolution level or level of detail only after at least one batch using a given material has been processed. The first batch is always executed at the machine level of detail. The execution of the first batch of parts is used to capture the data for individual part processing and generate the parameters for use in the distribution of the batch processing times. For instance, if the first batch contains aluminum parts, the checkbox would be unavailable. As soon as a second batch of aluminum parts arrives to the machine, the check box would be available to be selected. Again, this is a simple approach used for this prototype. In future versions, the model may be executed with machine level of detail for longer runs and data collected for aggregation and use in execution with multiple resolution level. Capabilities can be developed to set up the length of the run based on the desired width of the confidence interval for the individual part processing times.

The models at multiple resolution levels should be valid representations of the underlying real world phenomena based on the purpose of the model. Such validation will require comparison with the real world data. The performance of the models with selected measures will need to be compared and their accuracy for the desired purpose evaluated following procedures such as those described by Sargent (2014). The underlying model of the turning process has been previously validated for its prediction of energy consumption (Shao, Jain, and Shin 2014). While a detailed validation is out of scope for this prototype building exercise, a quick comparison of results from execution at the two levels of details was done using a two-tailed z-test. Both runs used a 2-day (16-hours) simulation time with 100 batches completed when executed at the machine level of detail and 101 batches completed at the manufacturing cell level of detail. The z-statistic of -0.145 compared to a two tailed p-value of 0.8847 at the 95% confidence level led to the conclusion that the two samples were not statistically different. Figure 4 shows the distribution of batch cycle times generated for the two runs. The batch cycle time is calculated as the time between the arrival at the source and the exit at the sink. The x-axis represents the time in minutes while the y-axis represents the percentage of batches that are in corresponding range of time.



(a) Manufacturing cell level of detail (b) Machine level of detail

Figure 4: Batch cycle times in minutes with all machines at (a) the manufacturing cell level of detail and (b) the machine level of detail.

5 CONCLUSION

This paper reports on an initial prototype to explore the feasibility of multi-resolution modeling in the context of virtual factory. By design, this first step took an approach that avoided other complicating factors. For example, the models at different resolution were within the same simulation environment and thus the complexity of implementing distributed simulation was avoided. Similarly, simple approaches were used for aggregation of data from machine level to manufacturing cell level and for setting up the multiple resolution execution. The exercise indicated that multiple resolution modeling is feasible at least in this simplified environment.

Future work will focus on iteratively adding capabilities and complexities. The initial step reported in this paper utilized machine level models for turning machines. Additional machining processes will be added in the near term and a range of process models may be considered in future. A process model for milling machines is nearly complete and will be the next one to be integrated in the prototype. The initial step reported here focused on use of a simulation environment that allows modeling at multiple resolutions. An alternate approach of representing the detailed level using tools specifically developed for process simulations is being explored. Integration with separate tools will require the use of a distributed simulation set up with its associated complexities. The current prototype used standards for the input and output for the machine level model. For future versions, additional interfaces based on standards will be developed. The factory data may be imported using the Core Manufacturing Simulation Data (SISO 2012) standard and the outputs may be generated using Business To Manufacturing Markup Language (B2MML; MESA 2013) standards. Also, the current implementation used ad-hoc terminology for the three levels of details. Standard terminology and scope of levels of resolution in manufacturing modeling will be explored for future iterations. The preceding are some of the ideas for enhancements under considerations. The actual iterative enhancements will be driven by the overall Smart Manufacturing System program that this initiative is a part of at the National Institute of Standards and Technology (NIST).

DISCLAIMER

No approval or endorsement of any commercial product by NIST is intended or implied. Certain commercial software systems are identified in this paper to facilitate understanding. Such identification does not imply that these software systems are necessarily the best available for the purpose.

ACKNOWLEDGMENTS

Sanjay Jain’s work on this effort has been sponsored under the cooperative agreement no. 70NANB13H158 between the NIST and George Washington University. The work described was funded by the United States Government and is not subject to copyright.

REFERENCES

Berenson, M.L., D.M. Levine, and T.C. Krehbiel. 2002. *Basic Business Statistics: Concepts and Applications*, 8th ed., p. 249-251. New Jersey: Prentice Hall.

Fakhimi, M., A. Anagnostou, L. Stergioulas, and S. J. E. Taylor. 2014. “A Hybrid Agent-Based and Discrete Event Simulation Approach for Sustainable Strategic Planning and Simulation Analytics.” In *Proceedings of the 2014 Winter Simulation Conference*, edited by A. Tolk, S. D. Diallo, I. O. Ryzhov, L. Yilmaz, S. Buckley, and J. A. Miller, 1573-1584. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers, Inc.

Grigoryev, I. 2015. *AnyLogic 7 in Three Days: A Quick Course in Simulation Modeling.* 2nd edition.

Guan, Z. L., C. J. Wang, Y. F. Wu, and X. Y. Shao. 2012. "A Framework of Digital Factory System Using Multi-Resolution Simulation." *Applied Mechanics and Materials* 159:12-17.

Herrmann, C., S. Thiede, S. Kara, and J. Hesselbach. 2011. "Energy Oriented Simulation of Manufacturing Systems – Concept and Application." *CIRP Annals - Manufacturing Technology* 60(1): 45-48.

Hong, S.-Y., and T.G. Kim. 2012. "Specification of Multi-Resolution Modeling Space for Multi-Resolution System Simulation." *Simulation: Transactions of the Society for Modeling and Simulation International* 89(1):28–40.

ISO. 2007. *ISO 10303-238:2007. Industrial automation systems and integration.* International Standards Organization (ISO).

ISO. 2013. *IEC 62264-1:2013. Enterprise-control system integration -- Part 1: Models and terminology*. International Standards Organization (ISO).

Jain, S., S. Sigurðardóttir, E. Lindskog, J. Andersson, and B. Johansson. 2013. "Multi-Resolution Modeling for Supply Chain Sustainability Analysis." *Proceedings of the 2013 Winter Simulation Conference*, edited by R. Pasupathy, S.-H. Kim, A. Tolk, R. Hill, and M. E. Kuhl, 1996-2007. Piscataway, NJ: Institute of Electrical and Electronics Engineers, Inc.

Jain, S., and G. Shao. 2014. "Virtual Factory Revisited for Manufacturing Data Analytics." In *Proceedings of the 2014 Winter Simulation Conference*, edited by A. Tolk, S. D. Diallo, I. O. Ryzhov, L. Yilmaz, S. Buckley, and J. A. Miller, 887-898. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers, Inc.

Jones, A.T., and C.R. McLean. 1986. "A Proposed Hierarchical Control Model for Automated Manufacturing Systems." *Journal of Manufacturing Systems* 5(1):15-25.

Mario, H., P. Tobias, and O. Boris. 2015. "Design Principles for Industrie 4.0 Scenarios: A Literature Review." Working Paper No. 01 / 2015, Technische Universität Dortmund, Dortmund, Germany.

MESA 2013. *Business To Manufacturing Markup Language Release Notes Version 6.0 – March 2013.* MESA International, Chandler AZ.

Mourtzis , D., N. Papakostas, D. Mavrikios, S. Makris, K. Alexopoulos. 2015. "The Role of Simulation in Digital Manufacturing: Applications and Outlook." *International Journal of Computer Integrated Manufacturing* 28(1): 3-24.

MTConnect. 2012. "Part 1-Overview and protocol, Version 1.2.0" MTConnect Institute (2012).

Sargent, R.G. 2014. "Verifying and Validating Simulation Models." In *Proceedings of the 2014 Winter Simulation Conference*, edited by A. Tolk, S. D. Diallo, I. O. Ryzhov, L. Yilmaz, S. Buckley, and J. A. Miller, 118-131. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers, Inc.

Shao, G., S. Jain, and S.-J. Shin. 2014. "Data Analytics using Simulation for Smart Manufacturing." In *Proceedings of the 2014 Winter Simulation Conference*, edited by A. Tolk, S. D. Diallo, I. O. Ryzhov, L. Yilmaz, S. Buckley, and J. A. Miller, 2192-2203. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers, Inc.

SISO 2012. *SISO-STD-008-01-2012: Standard for Core Manufacturing Simulation Data – XML Representation*. Simulation Interoperability Standards Organization, Orlando, F L.

Terkaj, W., and M. Urgo. 2015. "A Virtual Factory Data Model as a Support Tool for the Simulation of Manufacturing Systems." *Procedia CIRP* 28:137–142.

Williams, T. J. 1994. "The Purdue Enterprise Reference Architecture." *Computers in Industry* 24(2):141-158.

Yang, X., R.C. Malak, C. Lauer, C. Weidig, H. Hagen, B. Hamann, J.C. Aurich, and O. Kreylos. 2015. "Manufacturing System Design with Virtual Factory Tools." *International Journal of Computer Integrated Manufacturing* 28(1): 25-40.

AUTHOR BIOGRAPHIES

SANJAY JAIN is an Associate Professor in the Department of Decision Sciences, School of Business at the George Washington University. Before moving to academia, he accumulated over a dozen years of industrial R&D and consulting experience working at Accenture in Reston, VA, USA, Singapore Institute of Manufacturing Technology, Singapore and General Motors North American Operations Technical Center in Warren, MI, USA. He served as an associate editor of the *International Journal of Simulation and Process Modelling* for 2004-2014 and continues to serve as a member of the editorial board of the *International Journal of Industrial Engineering*. His research interests are in application of modeling and simulation of complex scenarios including smart manufacturing systems and project management. His email address is jain@email.gwu.edu.

DAVID LECHEVALIER is a guest associate in NIST's Systems Integration Division of the Engineering Laboratory. His current research topics include advanced analytics and modeling for Smart Manufacturing. He received a computer science Master's degree in Database and Artificial Intelligence from University of Burgundy in 2012. His email address is david.lechevalier@nist.gov.

JUNGYUB WOO is a research associate in the Systems Interoperability Group in NIST's Software and Systems Division of the Information Technology Laboratory. His current research topics include data standard, data assurance, big data infrastructure and data interoperability for Health Care and Smart Manufacturing. He worked at Samsung SDS before working at NIST. He received a PhD degree in industrial and management engineering from POSTECH in 2008. His email address is jungyub.woo@nist.gov.

SEUNG-JUN SHIN is a guest associate in the Life Cycle Engineering Group in NIST's Systems Integration Division of the Engineering Laboratory. His current research topics include advanced analytics, big data infrastructure and STEP-based data interoperability for Smart Manufacturing. He worked at Samsung Electronics and Samsung SDS before working at NIST. He received a PhD degree in industrial and management engineering from POSTECH in 2010. His email address is seungjun.shin@nist.gov.