An Agent-based Approach for Managing Symbiotic Simulation of Semiconductor Assembly and Test Operation

Malcolm Yoke Hean Low,
Kong Wei Lye, Peter Lendermann
Singapore Institute of
Manufacturing Technology
71 Nanyang Drive, Singapore 638075
(65)6793 8309
yhlow@simtech.a-star.edu.sg

Stephen John Turner, Reman Tat Wee Chim,
Surya Hadisaputra Leo
School of Computer Engineering
Nanyang Technological University
Nanyang Avenue, Singapore 639798
(65)6790 4054
ASSJTurner@ntu.edu.sg

ABSTRACT

The rapid changing business environment of high-tech asset intensive enterprises such as semiconductor manufacturing constantly drives production managers to look for better solutions to improve the manufacturing process. Simulation, though identified to be the most appropriate technique to generate and test out possible execution plans, suffers from long cycle-time in the process of model update, analysis and verification. It is thus very difficult to carry out prompt "what-if" analysis to respond to abrupt changes in these systems. Symbiotic simulation systems have been proposed as a way of solving this problem by having the simulation and the physical system interact in a mutually beneficial manner. In this paper, we describe our work in developing a prototype proof-of-concept symbiotic simulation system that employs software agents in the monitoring, optimization and control of a semiconductor assembly and test operation.

Categories and Subject Descriptors

H.4.2 [**Information Systems Applications**]: Types of Systems - *Decision support*; I.2.11 [**Artificial Intelligence**]: Distributed Artificial Intelligence – *Multiagent systems*; I.6.8 [**Simulation and Modeling**]: Types of Simulation – *Discrete event*; J.7 [**Computers in Other Systems**]: Industrial Control.

General Terms

Management, Performance, Design, Experimentation.

Keywords

Symbiotic Simulation, Agent-supported Simulation, Semiconductor Manufacturing.

AAMAS'05, July 25-29, 2005, Utrecht, Netherlands.

1. INTRODUCTION

With increasing complexity and the rapid changing business environment in semiconductor manufacturing, where product life cycle is constantly being shortened, semiconductor manufacturers are constantly being driven to look for better solutions to improve the manufacturing process. At the same time, as yield and efficiencies from wafer fabs continue to increase, the challenge for semiconductor backend is to maximize equipment utilization, productivity and yield, and reduce lead-time [26]. While most research focuses on throughput, utilization and cycle-time as key performance parameters when optimizing existing operations and for the complex planning of new facilities [5], there is also an increasing focus in optimizing order tardiness using due-date based dispatching in the manufacturing shop floor [15, 16, 19].

A typical semiconductor backend assembly and test facility generally has a high variety of products each requiring different route specifications resulting in a substantially large number of process flows or routings. A high product mix also leads to variations in routing and process time. The cycle-time of the assembly operation in a typical backend usually falls in the range of 3 to 6 days. The typical test operation cycle-time is in the region of 1.5 to 15 days [25].

Simulation has proved to be an important decision making tool in carrying out “what-if” analysis to evaluate complex operating scenarios in both the semiconductor frontend [7, 15] and backend operations [5, 25]. However, these systems often cannot be modeled with sufficient fidelity due to the fact that the physical system is constantly changing and very often simulation models are only updated with real-time manufacturing system data on an ad-hoc basis. There is also a very tedious process involved in the manual validation of simulation model and the analysis results. Thus, it is very difficult to carry out prompt "what-if" analysis to respond to abrupt changes in the physical system.

Symbiotic simulation systems are an emergent technology proposed by the Parallel and Distributed Simulation working group at the 2002 Dagstuhl seminar on Grand Challenges for Modelling and Simulation [9]. Essentially, a symbiotic simulation system consists of a simulation model interacting with the physical system in a mutually beneficial way. The simulation system benefits from the continuous supply of the latest data and the automatic validation of its simulation outputs, whereas the physical system benefits from optimized performance obtained from the analysis of simulation experiments. A symbiotic integration of real-time physical system and the corresponding decision support module will enable prompt response to be carried out to handle abrupt changes in the physical system.

Agent technology has been hailed as a new paradigm for developing software applications [11]. An agent is a software program that has the characteristics of being autonomous, proactive, responsive and adaptive. A multi-agent system is a collection of agents where each agent carries out its own action and performs decision making based on local knowledge, cooperating with one another at the same time with the common desire to achieve global optimal performance of the whole system. Such a system has the characteristics of distribution of control, resources as well as expertise among the different agents.

In this paper, we investigate the use of agent-based technology to realize the symbiotic integration between a simulation-based decision support optimization module and the corresponding real-time model of a semiconductor backend system. The study will look at how symbiotic integration using software agents can generate prompt response to abrupt changes in the physical system through simulation-based optimization. This approach will allow the use of simulation to address not only strategic and tactical but also operational challenges in the semiconductor manufacturing shop floor.

2. RELATED WORK

Simulation has been used extensively to carry out static analysis and optimization of semiconductor backend operation. For example, simulation is used in [5] to analyze the Assembly, Burn-in, and Test operations for the Infineon Backend Assembly and Test facility at Dresden, Germany. The authors showed that many of the recommendations identified in the study can be implemented at no additional cost to the factory.

In [25], Sivakumar carried out a simulation-based cause and effect analysis of cycle-time distribution in semiconductor backend using a validated model of an actual backend facility. The study concluded that lot release scheduling to the first operation has the greatest impact on cycle-time distribution and throughput in semiconductor backend manufacturing. The author also noted that theoretical cycle-time distribution (which is based on a fixed lot-size and excludes transport time, material handling time, and queuing time) and lot-size distribution have complex effects on cycle-time distribution.

In [22], Potoradi et. al. examined the issue of optimal lot-size for the Infineon Malacca backend factory. Analysis of two different models, one for the combined assembly/end-of-line areas and one for the burn-in/test areas, were carried out. The authors showed that the operating curves produced for different lot-sizes for the two models can be used to determine the optimum lot-size that maximizes throughput while maintaining the established cycle-time target for both the assembly and burn-in/test areas.

In a conventional simulation project, decision support business applications are often translated and the business logics are implemented directly into the simulation model. Any changes or enhancement to the business processes in the simulation model must then be translated back to the business application. A new approach of integrating standalone business process application with an existing simulation model has been proposed and studied [13], where the analysis and enhancement of business applications for scheduling and tracking are carried out in a simulation test-bed.

In [4, 10], distributed simulation techniques are used to evaluate business strategies for a decision support system for a semiconductor supply-chain. In the former, the decision support module is implemented as a synchronized federate (a term for a simulator in distributed simulation) and executed together with the other parts of the simulation system. In the latter, the decision support module is a decoupled system that interfaces with the simulation system through a gate-way federate.

Researchers have studied the integration of an expert system with a simulation model. In [23], the authors demonstrated how a simulation model developed using a commercial-off-the-shelf (COTS) simulation package can be linked to an external expert system and how decisions generated by the expert system can be fed back into the simulation system subsequently.

However, rapid changes in operating conditions in a backend factory often means that any results obtained from static analysis of the factory will not be valid when the corresponding action is applied back into the factory. The status of the factory needs to be constantly updated into the simulation model and simulation results must be generated in a timely fashion so that such results can be of any use to the factory manager.

Simulation-based real-time dispatching is one such technique that constantly monitors shop floor conditions and carries out simulation-based evaluation to generate schedules for lots processing. In [17], the authors examined different simplification methods that can be used in simulation-based optimization in order to carry out efficient and prompt real-time dispatching for semiconductor frontend manufacturing.

Symbiotic simulation is another new development area that was proposed at the Dagstuhl seminar by the Parallel and Distributed Simulation working group [9] as a grand challenge with relevance to various application domains. In symbiotic simulation, real-time information from the physical system is constantly fed and updated into the simulation model; and results obtained from simulation-based optimization are also used at the same time to fine-tune the physical system.

Paul Reynolds and his team at the University of Virginia have developed the technique COERCE for simulation reuse as part of the Dynamic Data-Driven Application Simulation (DDDAS) project funded by the National Science Foundation. In [3, 18], the authors described the input and runtime parameters of an existing simulation model that can be coerced to facilitate experimentation and composibility of simulation components comprising DDDAS.

Agent-based technology has been used in the area of manufacturing enterprise integration, supply-chain management, manufacturing planning, scheduling and control, materials handling, and holonic manufacturing systems [24]. An agent-based approach for supporting logistics and production planning is demonstrated in [14] using a case study that optimized the production planning of a virtual manufacturing enterprise in

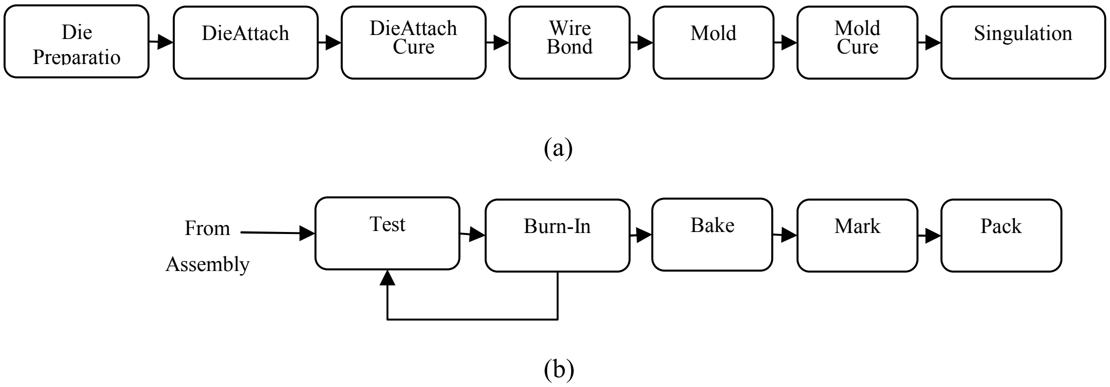


Figure 1. A Representative Model of Semiconductor Backend (a) Assembly and (b) Test Operation.

relation to sub-contracted logistic services used to transport materials between the enterprise units. In [2], the authors described a simulation test-bed for the evaluation of a distributed multi-agent control architecture for holonic manufacturing systems that integrates discrete-event simulation software into its design to allow the control architecture to be evaluated with a variety of emulated manufacturing systems.

3. ASSEMBLY AND TEST CASE STUDY

The case study used in this paper is a representative model for the semiconductor backend assembly and test operation. Figure 1 shows different process steps in the semiconductor backend model. The details of this model have been previously described in [19], where the effectiveness of different dispatching rules to minimize order tardiness in a semiconductor backend operation are evaluated.

For the purpose of this study, a simulation model of the semiconductor backend factory implemented in the commercial simulation software package Witness [20] is used to emulate the real factory, where the symbiotic integration for monitoring and control using software agents is carried out. We will refer to this model as the “emulated system” for the rest of this paper.

The emulated system is modified so that out-sourcing to an external vendor can take place at the Wire-Bond and Test machine groups whenever the in-house capacity cannot handle the weekly lot arrival demand. As out-sourcing is a pre-planned activity, if out-sourcing at a particular machine is decided, lots are marked for out-sourcing only at the beginning of the assembly line and are routed to the subcontractor when they reach the out-sourcing machine.

In this case study, the factory incurs a penalty cost of $\$l$ for each day an order is late. Each production order in the system has an associated due-date for shipping to the customers. The emulated system keeps track of the tardiness of each order with respect to its due-date. For each 1K pieces of integrated circuit (IC) out-sourced to external vendors, the factory also incurs a penalty cost of $\$k_1$ and $\$k_2$ for the Wire-Bond and Test machine group respectively.

The emulated system is run for 1 year of simulated time. All machines in the system use the first-in-first-out (FIFO) dispatching rule for lot processing. The execution speed of the simulation can be controlled using the walk speed control available on the Witness simulation package. A walk speed of 1 gives the slowest execution speed and for our emulated simulation model of the semiconductor backend, the simulation-to-real-time ratio is about 4. A walk speed of 7 gives a simulation-to-real-time ratio of about 1000. As the Witness model is used to emulate the physical system, varying the walk speed can be considered as changing the relative execution speed of the external decision support module with respect to the real system.

4. SYMBIOTIC SIMULATION

In this paper, we describe our work in developing a symbiotic simulation system for semiconductor backend assembly and test operation. In our implementation, software agents are employed to serve the roles of monitoring, control, verification and also carrying out simulation-based optimization for the symbiotic integration with the semiconductor backend operation. Software agents are used to realize the symbiotic simulation system due to their ability to carry out autonomous monitoring and control of the physical system, as well as their mobility that allows optimization to be carried out on distributed resources.

Figure 2 shows the architecture of the agent-based symbiotic simulation system for the semiconductor backend operation. The semiconductor backend physical system is emulated using a simulation model developed in Witness. The different agents in this symbiotic simulation system are implemented using the JADE agent toolkit [6].

4.1 Monitoring

In this system, the system management agent (SMA) and the monitoring agents (MA) monitor key performance indicators of the physical system. The monitoring agent causes the status of the manufacturing system to be updated into a database periodically or on demand. The Witness Command Language (WCL) is a scripting language that allows modification and control of a Witness simulation model during runtime.

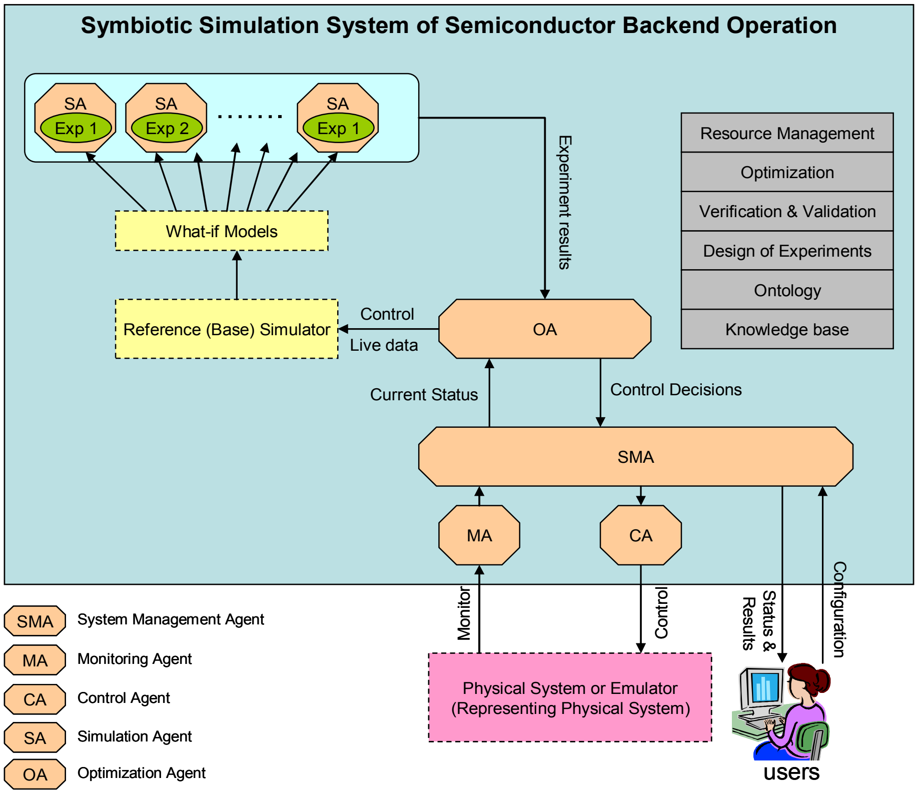


Figure 2: Symbiotic Simulation of Semiconductor Backend Operation.

As Witness only provides access to WCL through Visual Basic interfaces, the Java-COM Bridge (Jacob) [1] is used to implement the interface between the JADE agents and the Witness system.

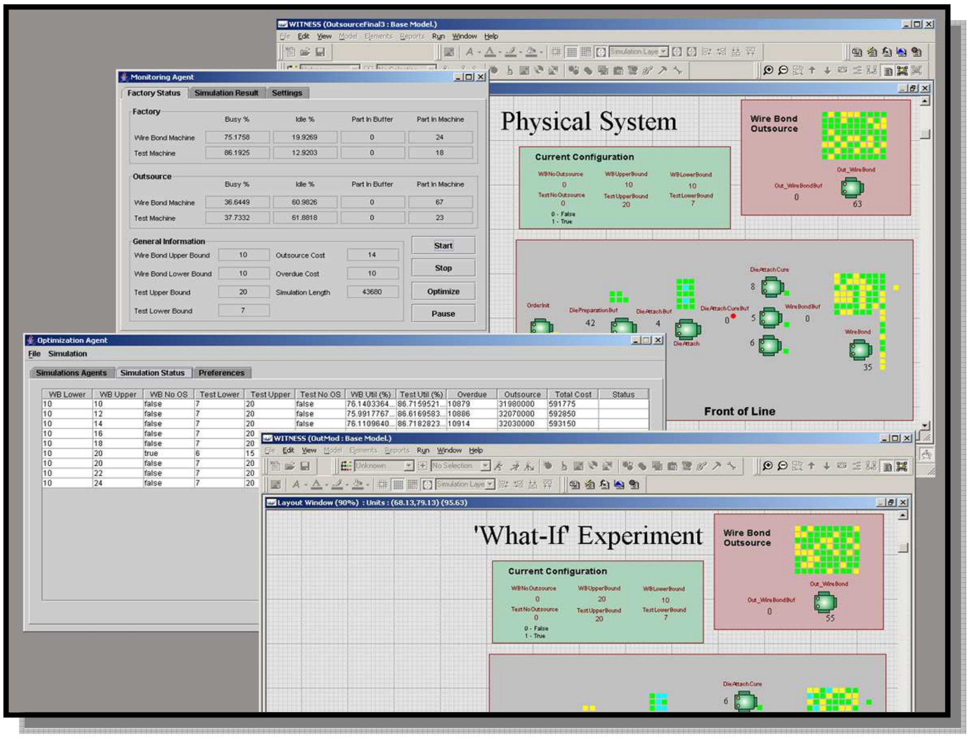
The SMA will report any anomalies in the emulated system to the Optimization Agent (OA). This triggers a "what-if" analysis to be carried out by the OA. Both the SMA and the OA, as well as the other agents in the system communicate with one another using ACL messages that are encoded using a customized semiconductor manufacturing ontology developed using the Protégé ontology authoring tool [21]. A Protégé plug-in further converts the concepts and predicates in the manufacturing ontology to Java classes that can be used directly by the JADE agents. The use of ontology will allow future adaptation of this system to other domains such as the military and logistics.

As the way to decide what constitutes an anomaly in the system can be based on a combination of different performance indicators, the Java Expert System Shell (JESS) [8] is used in the symbiotic simulation system by the SMA to decide when simulation-based optimization is necessary. In this case study, the JESS system is initialized with the rule that detects an anomaly when the utilization of the Wire-bond or Test machines exceeds 80%. The use of the JESS system also allows decision logics for other domains be added and used by the symbiotic simulation system for application on other domains.

4.2 Optimization

When an anomaly in the backend system is reported to the OA by the SMA, a copy of the emulated system (in the form of a Witness simulation model with its current execution state) is also sent to the OA. This simulation model with the saved state of the current physical system will be referred to as the "base model" for the rest of this paper. In a full-fledged implementation of the symbiotic simulation system, a reference simulation model should be maintained and will be constantly running alongside the physical system. The base model and the physical system will be constantly monitored and compared for any deviation so that model maintenance and optimization can be carried out promptly.

The OA has to carry out simulation-based optimization to decide the best approach for handling a given situation. In this case study, the OA tries to decide the upper and lower queue size, S_u and S_l, for the queue before machine *M*, which can be a group of Wire-bond or Test machines. If the queue size for machine *M* exceeds S_u, all subsequent lots released into the system will be marked for out-sourcing at machine *M*. Similarly, if the queue size for machine *M* falls below S_l, all subsequent lots released into the system will not be marked for out-sourcing at machine *M*.



Factory

	Busy %	Idle %	Part In Buffer	Part In Machine
Wire Bond Machine	75.1758	19.9269	0	24
Test Machine	86.1925	12.9203	0	18

Outsource

	Busy %	Idle %	Part In Buffer	Part In Machine
Wire Bond Machine	36.6449	60.9826	0	67
Test Machine	37.7332	61.8818	0	23

General Information

Wire Bond Upper Bound	10	Outsource Cost	14
Wire Bond Lower Bound	10	Overdue Cost	10
Test Upper Bound	20	Simulation Length	43680
Test Lower Bound	7		

WB Lower	WB Upper	WB No OS	Test Lower	Test Upper	Test No OS	WB Util (%)	Test Util (%)	Overdue	Outsource	Total Cost	Status
10	10	false	7	20	false	76.1403364...	86.7159521...	10879	31980000	591775	
10	12	false	7	20	false	75.9917767...	86.6169583...	10886	32070000	592850	
10	14	false	7	20	false	76.1109640...	86.7182823...	10914	32030000	593150	
10	16	false	7	20							
10	18	false	7	20							
10	20	true	6	15							
10	20	false	7	20							
10	22	false	7	20							
10	24	false	7	20							

Current Configuration (Physical System)

WBNoOutsource	WBUpperBound	WBLowerBound
0	10	10

TestNoOutsource	TestUpperBound	TestLowerBound
0	20	7

0 - False
1 - True

Current Configuration ('What-If' Experiment)

WBNoOutsource	WBUpperBound	WBLowerBound
0	20	10

TestNoOutsource	TestUpperBound	TestLowerBound
0	20	7

0 - False
1 - True

Figure 3: Screen Capture of Symbiotic Simulation System for Semiconductor Backend.

Using the base model, the OA creates a number of models with different S_u and S_l combinations and distributes these models to the simulation agents (SA) to carry out "what-if" analysis. As the study of the optimization algorithm to determine the optimal S_u and S_l is not the main focus of this phase of the project, an iterative search method is used instead. The OA uses a fixed value of 10 for S_l and incrementally searches for an optimal value of S_u between 10 and 200 that will yield the minimum operating cost for the simulation model.

From the results collected from the SAs, the OA decides if more experiments have to be conducted or if the result can be sent back to the SMA. As the JADE toolkit has built-in mobility for agents, these SAs can also be created locally and migrated to other dedicated machines to execute the simulation concurrently. The use of Grid computation resources to execute simulation-based optimization has been studied in another related project [12].

4.3 Control

On receiving the evaluation result from the OA, the SMA instructs the control agent (CA) to make the necessary modifications to the emulated system. In this case, the modifications involved setting the upper and lower threshold for the queue size at the Wire-bond and Test machines.

Note that while the OA is evaluating the different scenarios, the emulated system continues running (mimicking what happens in the real physical system). Thus, the results returned by the OA will actually be applied in the physical system with a different shop floor condition compared to when the evaluation first took place. There is therefore a need to validate and verify if the evaluation result is still applicable on the physical system. This process often requires domain expert knowledge and will be studied in the next stage of this project.

5. EXPERIMENTAL RESULTS

In this section, we describe two sets of experiments that are conducted using the symbiotic simulation system on the semiconductor backend case study. For each set of experiments, the emulated system will have insufficient capacity at the Wire-bond and Test machines to handle the weekly input demand over time. Thus, the utilization of Wire-bond machines will reach 80% at some point in the execution of the emulated system. When this occurs, the SMA agent will send a trigger to the OA to start the simulation-based optimization. All experiments are conducted on a group of dedicated 2.4GHz PCs with 512MB memory connected on a 100Mbits Ethernet local area network. Version 2003 of the Witness simulation package is used for all simulation models.

5.1 Fixed Number of Scenarios

For the first set of experiments, the number of scenarios the OA can evaluate is limited to 3. The experiment is repeated using different walk speed for the emulated system after the triggering to the OA has been carried out. The experiments are also repeated using 1 and 3 PCs for executing the simulation agents. While each SA created by the OA evaluates its respective simulation scenario for a one-year operation using the Witness batch execution mode (which executes the simulation as fast as possible), the emulated system is run at different walk speeds for each experiment. The optimization result obtained by the OA is the same for all the experimental runs but the simulation time of the emulator when the result is returned to the SMA will be different.

Figure 3 shows the corresponding simulation time that elapsed between the time the OA is triggered and the time the optimization result is returned to the SMA. The graph shows that with a slower walk speed, the optimization result can be returned to the SMA within a shorter period (relative to the elapsed time on the emulated system). Moreover, migrating the SAs to 3 PCs to evaluate the scenarios further shortens the evaluation duration compared to the case with only 1 PC.

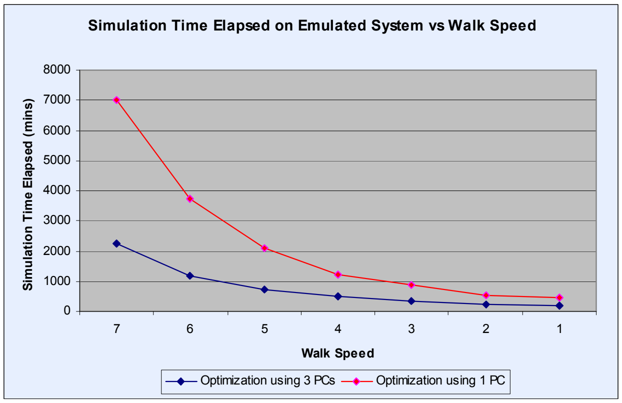


Figure 3: Simulation Time Elapsed on Emulated System vs Walk Speed for Experiment 1.

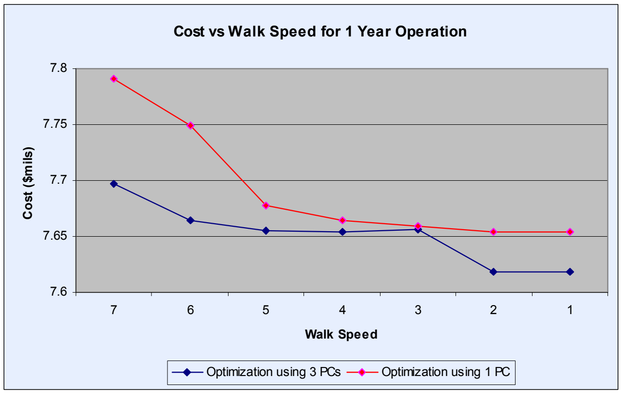


Figure 4: Operating Cost vs Walk Speed for 1 Year Operation for Experiment 1.

5.2 Fixed Duration for Optimization

For the second set of experiments, the duration of simulation optimization allowed for the OA is set to 1000 simulation time units. The OA must return the optimization result to the SMA when the emulated system has reached simulation time *t*+1000, where *t* is the simulation time when the triggering occurred. There is no restriction on the number of scenarios the OA can evaluate, but the OA has to terminate all SAs and return the current best result to the SMA when the time limit of *t*+1000 has been exceeded. Unlike in the first set of experiments where all SAs evaluate each scenario for one year of operation, in this set of experiments we vary the length of the scenario evaluation to between 1, 3 and 6 months. In this set of experiments, only one PC is used to evaluate the different scenarios.

Table 1 shows the number of scenarios evaluated by the OA during the 1000 simulation time units on the emulated system for different walk speeds. For walk speeds 6 and 7, the OA does not have time to finish evaluating any scenario by *t*+1000 for the experiments with 3 and 6-month evaluation. For walk speeds 4 and 5, the experiment with 6-month evaluation also did not finish evaluating any scenario. Since no result is returned by the OA for these six cases, the SMA did not modify the emulated system and allowed it to continue with the existing out-sourcing configuration.

Table 1: Number of Scenarios Evaluated vs Walk Speed

Walk Speed	1-month Evaluation	3-month Evaluation	6-month Evaluation
1	58	21	12
2	10	3	2
3	5	3	1
4	2	1	0
5	2	1	0
6	1	0	0
7	1	0	0

Figure 5 shows the corresponding operating cost for the emulated system after one year of operation. As no out-sourcing is carried out for these 6 cases, the operating cost stays at a high level. The graph also shows that a short evaluation of one month can produce fast results that yield a comparable operating cost as a 6-month evaluation. This simplification method of carrying out simulation for a shorter simulated time has also been shown to be effective in improving the response time for simulation-based real-time scheduling in [17].

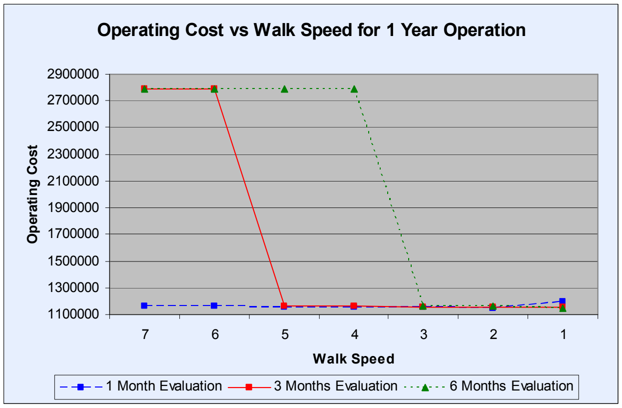


Figure 5: Operating Cost vs Walk Speed for 1 Year Operation for Experiment 2.

6. CONCLUSION

In this paper, we have described an agent-based symbiotic simulation system for semiconductor backend operation. Experiments conducted using our prototype show that agent technology can be used effectively to carry out the monitoring, optimization and control functions of the symbiotic simulation system. We have shown that mobile simulation agents can migrate the task of simulation scenario evaluation to multiple computers so that they can be carried out concurrently. Our experimental results also showed that simplification methods, such as simulating for a shorter simulated time, will allow the symbiotic simulation system to respond more promptly to abrupt changes in the physical system.

We are currently developing a roadmap towards a symbiotic simulation system with a semiconductor foundry based in Singapore. Further work will be carried out to study the following issues: 1) What are the appropriate levels of detail for the model and how to filter and analyze only those data that are required for decision support purposes? 2) What are the mechanisms or key performance indicators used to detect divergence between the simulation model and the physical system? Is it based on the throughput of the system, utilization of equipment, or a combination of different factors? How closely should the physical system be monitored to detect these divergences? 3) How can the automatic validation of the output from the simulation system against the physical system be carried out? Further work will also be carried out to integrate the system with the other subsystems (Grid resource management and simulation-based optimization) developed in other related projects.

7. ACKNOWLEDGMENTS

This project is part of a pilot project funded by the Agency for Science, Technology and Research (A-STAR) Thematic Strategic Research Program under the theme “Industrial IT - Integrated Manufacturing and Service System”. The authors would also like to thank Dr. Bud Fox and Dr. Xiang Wei from the Institute of High Performance Computing for their valuable input to the project.

8. REFERENCES

[1] Adler, D. The JACOB Project: A JAva-COM Bridge, Available at: http://danadler.com/jacob/, last accessed February 4th, 2005

[2] Brennan, R.W. and O, W. A Simulation Test-bed to Evaluate Multi-agent Control of Manufacturing Systems. In *Proceedings of the 2000 Winter Simulation Conference*, pp. 1747-1756, 2000.

[3] Carnahan, J.C., Reynolds, P.F., and Brogan, D.C. Visualizing Coercible Simulations. In *Proceedings of the 2004 Winter Simulation Conference*, pp. 411-419, 2004.

[4] Chong, C.S., Lendermann, P., Gan, B.P., Duarte, B.M., Fowler, J.W., and Callarman, T.E. Analysis Of A Customer Demand Driven Semiconductor Supply Chain In A Distributed Simulation Test Bed. In *Proceedings of the 2004 Winter Simulation Conference*, pp. 1902-1909, 2004.

[5] Domaschke, J., Brown, S., Robinson, J., and Leibl, F. Effective Implementation of Cycle Time Reduction Strategies for Semiconductor Backend Manufacturing. In *Proceedings of the 1998 Winter Simulation Conference*, pp. 985-992, 1998.

[6] Fabio Bellifemine, A.P., Giovanni Rimassa, Giovanni Rimassa. JADE - A FIPA-Compliant Agent Framework. In *Proceedings of Practical Apllications of Intelligent Agents and Multi-agent Systems (PAAM'99)*, pp. 97-108, 1999.

[7] Fowler, J., Knutson, K., and Carlyle, M. Comparison and Evaluation of Lot-to-Order Matching Policies for a Semiconductor Assembly and Test Facility. *International Journal of Production Reserch*, 38(8), pp. 1841-1853, 2000.

[8] Friedman-Hill, E., "Jess in Action": Manning Publications Co. 480. 2003

[9] Fujimoto, R., Lunceford, D., Page, E., and Uhrmacher, A. Technical Report of the Dagstuhl-Seminar Grand Challenges for Modelling and Simulation, available at: http://www.dagstuhl.de/02351/Report/. 2002.

[10] Gan, B.P., Lendermann, P., Chong, C.S., Duarte, B.M., Fowler, J.W., and Callarman, T.E. Architecture and Performance of an HLA-Based Distributed Decision Support System for a Semiconductor Supply Chain. In *Proceedings of the 2004 Fall Simulation Interoperability Workshop*, pp. No. 014F-SIW-109, 2004.

[11] Jennings, N.R. and Wooldridge, M.J. Applications of Intelligent Agents. In *Agent Technology: Foundations, Applications, and Markets*, pp. 3-28, 1998.

[12] Julka, N., Lendermann, P., Cai, W., Hung, T., and Khanh, N.Q. Grid Computing for Virtual Experimentation and Analysis. In *2nd Workshop on Grid Computing and Applications at Grid Asia 2005 (submitted)*, pp. 2005.

[13] Julka, N., Lendermann, P., Chong, C.S., and Liow, L.F. Analysis And Enhancement Of Planning And Scheduling Applications In A Distributed Simulation Testbed. In *Proceedings of the 2004 Winter Simulation Conference,*, pp. 1799-1806, 2004.

[14] Karageorgosa, A., Mehandjieva, N., Weichhartb, G., and H.ammerleb, A. Agent-based Optimisation of Logistics and Production Planning. *Engineering Applications of Artificial Intelligence*, 16, pp. 335-348, 2003.

[15] Kim, Y.-D., Kim, J.-G., Choi, B., and Kim, H.-U. Production Scheduling in a Semiconductor Wafer Fabrication Facility Producing Multiple Product Types With Distinct Due Dates. *IEEE Transactions on Robotics and Automation*, 17(5), pp. 589-598, 2001.

[16] Kim, Y.-D., Kim, J.-U., Lim, S.-K., and Jun, H.-B. Due-Date Based Scheduling and Control Policies in a Multiproduct Semiconductor Wafer Fabrication Facility. *IEEE Transactions on Semiconductor Manufacturing*, 11(1), pp. 155-164, 1998.

[17] Kim, Y.-D., Shim, S.-O., Choi, B., and Hwang, H. Simplification Methods for Accelerating Simulation-based Real-time Scheduling in a Semiconductor Wafer Fabrication Facility. In *IEEE Transactions on Semiconductor Manufacturing*, pp. 290-298, 2003.

[18] Loitiere, Y., Brogan, D.C., and Reynolds, P.F. Simulation Coercion Applied to Multiagent DDDAS. In *2004*

International Conference on Computational Science (ICCS), pp. 789-796, 2004.

[19] Low, M.Y.H., Kim, Y.-D., and Lee, T.-E. Due-date Oriented Dispatching Rules for Semiconductor Backend Operation. In *Proceedings of the 10th Asia Pacific Management Conference*, pp. 1043-1055, 2004.

[20] Markt, P.L. and Mayer, M.H. Witness Simulation Software A Flexible Suite of Simulation Tools. In *Proceedings of the 1997 Winter Simulation Conference*, pp. 711-717, 1997.

[21] Noy, N.F., Sintek, M., Decker, S., Crubezy, M., Fergerson, R.W., and Musen, M.A. Creating Semantic Web Contents with Protege-2000. *IEEE Intelligent Systems*, 16(2), pp. 60-71, 2001.

[22] Potoradi, J. and Lee, W.K. Determining Optimal Lot-size for a Semiconductor Back-end Factory. In *Proceedings of the 1999 Winter Simulation Conference*, pp. 720-726, 1999.

[23] Robinson, S., Edwards, J.S., and Yongfa, W. Linking the Witness Simulation Software to an Expert System to Represent a Decision–Making Process. *Journal of Computing and Information Technology - CIT*, 11(2), pp. 123-133, 2003.

[24] Shen, W. and Norrie, D.H. Agent-Based Systems for Intelligent Manufacturing: A State-of-the-Art Survey. *Knowledge and Information Systems*, 1(2), pp. 129-156, 1999.

[25] Sivakumar, A.I. Simulation Based Cause and Effect Analysis of Cycle Time Distribution in Semiconductor Backend. In *Proceedings of the 2000 Winter Simulation Conference*, pp. 1464-1471, 2000.

[26] Slocum, A.H. and Awtar, S. Backend Automation in the Semiconductor Industry. *Future Fab International*, 12, pp. 301-309, 2002.