行政院國家科學委員會專題研究計畫年度報告

以分散式代理程式系統建置與模擬虛擬晶圓廠的完整訂單流程 (1/2)

(Implementation and Simulation of a Virtual Fab based on Distributed Multi-agent Architecture)

計畫編號：NSC90-2212-E-002-222

執行期限：90 年 8 月 1 日至 91 年 7 月 31 日

主持人：黃漢邦 台灣大學機械工程學系

Email: hanpang@ccms.ntu.edu.tw

研究人員：游志源 台灣大學機械工程學系

一、中文摘要

本計畫主要建置一個虛擬的晶圓廠，用以模擬自接訂單到完成訂單的企業流程。由於系統規模大，本計畫將以分散式環境作為模擬與測試的基礎架構。整個系統包括下列模組：企業對企業的資訊交換介面、訂單管理系統、生產規劃、機台產能規劃、晶圓優先權設定規劃、製造執行系統、事件通報系統與中控系統。每個子系統將建置成代理程式(agent)架構在分散式的環境之中, 各自執行任務與達成目標。

在第一年中，我們將使用 UML 的系統分析法，找出整個系統所需要的子程式有哪些。另外，我們引入 Tool model 來預測生產週期，所提出的類神經學習法有較佳的預測能力。

關鍵詞：虛擬晶圓廠、生產週期預測、製造執行系統、分散式系統

Abstract

The key success of the foundry fab is based on the order fulfillment and the customers' satisfactory. The main objective of order fulfillment is to deliver the products on time. However, many processes, including due date setting, planning and scheduling of the order and real-time shop floor operations, are involved in accomplishing this goal. In order to enhance the customers' satisfactory, the foundry company should not only promise on-time delivery but also provide the real-time information of orders. Virtual fab (VF) can realize these two important issues.

The core of virtual fab is the order fulfillment process. In the first year, the entire model of the order fulfillment process is constructed. Besides, the event monitoring process and the tool model are implemented. Each subsystem is constructed as an agent on the distributed framework and performs its task.

Keywords: Virtual Fab, Cycle Time Estimation, MES, Distributed Environment

二、計畫緣由及目的

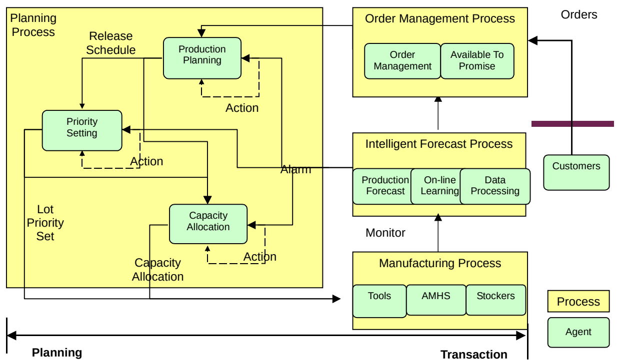


Fig. 1 The order fulfillment process

The core of virtual fab is the order fulfillment process. The process is triggered by the customers' orders in the foundry fab environment. This project will focus on the entire order fulfillment process. It includes order management process, planning processes, shop-floor operation, and event monitoring process, as shown in Fig. 1.

The major functions of the order management process are: 1) as customer portal,

2) due date quotation, 3) order status/tracking, and 4) order commitment.

The planning process includes production scheduling [1], capacity planning, lot priority setting and scheduling. The duty of production planning is to convert the orders into lot release schedule and to consider long-term capacity and WIP profile. The duty of priority management is to assign the priority value for all new released lot and re-assign the priority value for WIP. As to capacity planning, it should be responsible for tool allocation adjustment and re-allocation. The manufacturing process is modeled using Colored Timed Petri net to reflect the capacity constraints and dynamic behavior of lots. There are many researches [5-9] related to the modeling of the manufacturing execution system in the foundry fab.

The focuses of the first year are described below. The event monitor senses all real-time data to reflect alarm situation or trigger other processes to action. It can provide 1) lot priority decision support for priority management process, 2) tool configuration decision support for capacity planning process, 3) available-to-promise (ATP) decision support and order status decision support for order management process WIP level and capacity decision support for production planning.

The manufacturing process is triggered by the customers' orders in the foundry fab environment. The order is passed to the order management agent to decide the acceptance according to the capacity of the fab and Available To Promise (ATP) function [10,11]. If the order is received, it is passed to Production Scheduler (PS), which transfers the orders to the lot release schedule with the aid of the capacity planner. The fab model, including AMHS (automated material handling system), is constructed using Colored Timed Petri net (CTPN). Then the cycle time estimator and priority adjuster takes the real-time data to predict the cycle time of products for ATP function support and dynamically adjust lots' priorities for on-time delivery enhancement [12].

An interface will be defined to communicate with suppliers or customers to enable B2B integration and supply chain management, especially for information sharing and passing. The manager can simulate the virtual fab operation to check the bottleneck tools, capacity and on-time delivery performance. The customers can fetch the real-time information of their orders and monitor the status of lots to prevent the delay of orders through the web interface.

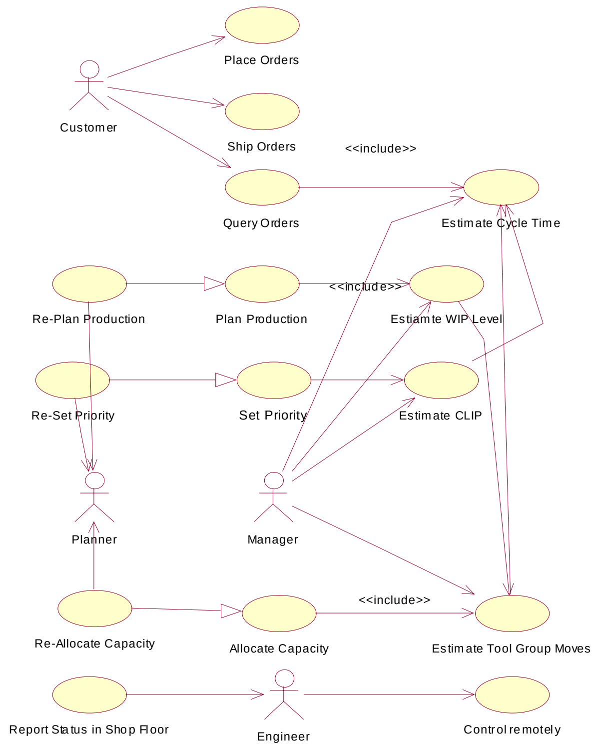


Fig. 2 Modeling of the order fulfillment process

三、研究方法

System Analysis

The modeling of the entire system is shown in Fig. 2. The system is decomposed into three processes, the order management process, intelligent forecast process and the planning process.

Order management process

The entire order management process involves order processing, order confirmation, order status, shipping, and billing. The general functions of each activity are stated below.

- Order processing

In this activity, the system acquires the orders from many ways including by telephone, by fax, EDI or through the Internet and then handles the orders.

- Order confirmation

After receiving the order, the system should check the credits of customers; manage the contract and confirm the ability to promise the on-time delivery of the order.

- Query status

After placing orders, customers always want to know the status of their orders. The statuses that customers concern are the estimated time to deliver and the condition of their wafers. Hence, the system should provide the real-time information about orders to customers.

- Shipping

After completing the orders, the system should transport the product to the indicated place (maybe the probe-and-test company) on time.

- Billing

The last activity in the transaction is the billing. The system should send the invoice and settle the bill.

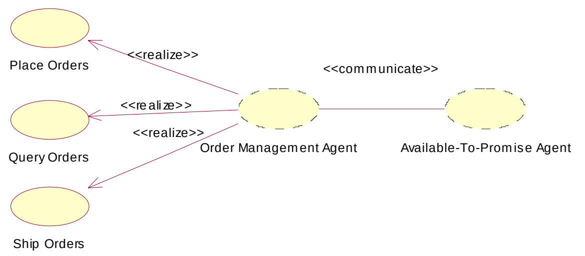


Fig. 3 The agent model in the order fulfillment system

In this project, the order management is modeled as the order management process. Since the accounting is not the focus of this project, billing is not the included in the system. The system will notify the customer when the order is completed however it does not manage the transportation of the product. Order processing, order query and order finished are included in the order management agent while the order confirmation is handled in the available-to-promise agent. The agent model in the order fulfillment system is shown in Fig. 3.

- Order Management Agent
  The order management agent receives orders through the Internet on the B2B communication platform. The use cases, “Place Orders”, “Query Orders” and “Ship Orders” are realized in this agent. Besides, Microsoft BizTalk server is used as the communication platform on customers’ company and eFoundry fab. On the other hand, RosettaNet standard is used as the format of the business document when sending orders.

- Available-To-Promise Agent
  The responsibilities of the available-to-promise agent are to check the estimated due date of customer orders and the available capacities for new orders. It is called by the order management agent. Any function that is related to the information about receiving orders is implemented in this agent.

Planning process

The planning process takes care about the release schedule, priority setting and the capacity allocation activities. After an order is submitted to the foundry fab, the next manufacturing and management problems, such as scheduling or evaluating due date etc., have to be undertaken by the production staffs in order to officially confirm the feasible due date and production schedule. One of the main reasons is because the product quantity cannot be precisely translated into the production capacity in the timely demand. After these obstacles are resolved reasonably, the ordering system can provide the real-time due date commitment to customers. In this study, the planning system is proposed to solve these problems. This system can be not only utilized to construct the interactive due date commitment system for the foundry fab, but also applied to the B2B manufacturing service. The agent model of the planning process is shown in Fig. 4.

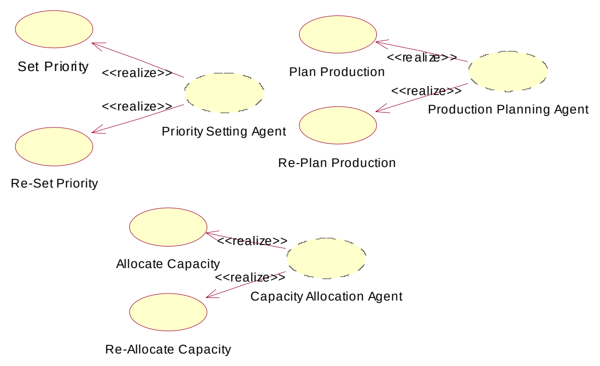


Fig. 4 The agent model in the planning process

- Production Planning Agent
  The production planning agent implements the use case, "Plan Production" and "Re-Plan Production". The function of this agent is to transfer the orders into the release schedule of lots into the manufacturing process. There are two important issues concerned in this agent. The release sequence and the time to release to manufacture of each lot.

- Priority Setting Agent
  The priority setting agent implements the use cases, "Set Priority" and "Re-Set Priority". When the new lot is released to manufacturing, it is assigned a priority class by this agent. Besides, it detects the delayed lots and re-assigns the priority class in order to meet the due dates of orders.

- Capacity Allocation Agent
  The capacity allocation agent implements the use case, "Allocate Capacity" and "Re-Allocate Capacity". The duty of this agent is to allocate the capacities of tools for lots and find out the bottleneck tools and tool groups. The issue, "Coupling Effect", is also considered. Besides, a new tool-dispatch rule is proposed to reduce the bottleneck and enhance the throughput.

Intelligent Forecast Process

The intelligent forecast process is the supporting process for other processes. It monitors the status of tools and lots. The information is redirected to the managers and the other agents. The agent model of the intelligent forecast process is shown in Fig. 5.

- Data Processing Agent
  The duties of the data processing agent are to collect the data generated from MES, filter the data, transform the data into physical meaning and process the data for supporting on-line learning agent. It is the interface between the shop floor and decision support systems.

- On-Line Learning Agent
  The duties of the on-line learning agent are to learn the behavior of the foundry fab. Hence, the information about tools and lots can be off-line retrieved. In this project, it applies neural networks on tool models to model the fab.

- Production Forecast Agent
  The production forecast agent implements the use cases, "Estimate Cycle Time", "Estimate WIP Level", "Estimate CLIP" and "Estimate Tool Group Moves". It generates the managerial information for others agents.

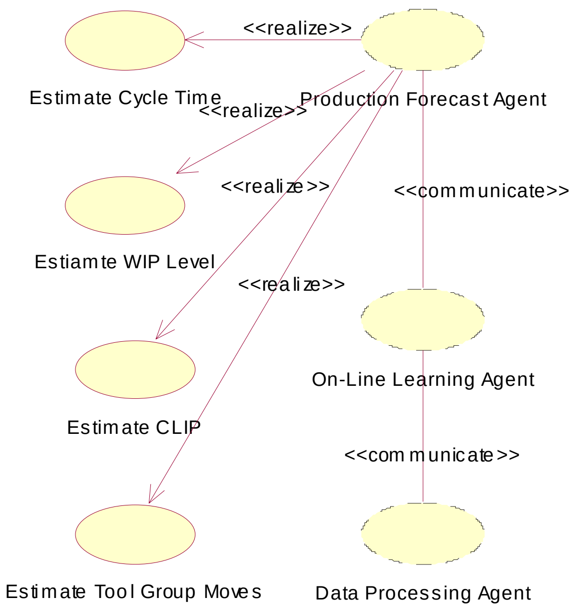


Fig. 5 The agent model in the intelligent forecast process

Intelligent Forecast Process

Tool Model

However, this feature cannot be used to characterize the attributes of each operation and waiting effects in detail. Thus, a tool model [10], which characterizes the dynamics of a single tool and considers the operation time as the prime unit, is proposed. Every operation time consists of waiting time and processing time. Setup time and transportation time are categorized as waiting time in this paper. The tool model processing time denotes the time that a lot requires for processing in a tool. Similarly, waiting time is the queue time for a lot waiting for service at a tool. All relevant attributes affecting the waiting time and processing time will be discussed in this paper. According to the lot flow, the time spent in the fab is modeled at each step in detail, while waiting time and processing time are considered separately. A waiting model is responsible for a tool group and a processing model is responsible for a tool. There are about 200 waiting models and 500 processing models in this implementation. Since each model is considered and calculated separately, the model is a more flexible and configurable for the entire fab.

Tools are the real working machines, while tool groups are man-defined or virtual units used to define the flow of a product. Usually, one or more tools that do the same operation belong to a tool group. The flow of a lot is the sequence of tool groups that only define the operations (or tool groups) it takes but not the specified tools. The flow of a lot is related to its product type or typically its route type. Although a lot is processed inside a tool, the route of a lot is the flow of the tool groups. Hence, a lot after processing in the previous step only knows the next tool group that it should go. Any available tool belonging to the tool group can serve this lot. If no tool is available, this lot waits in the queue or we say that it waits in a tool group. Lots waiting in the same tool group will compete for the same resources, i.e. the tools.

From the viewpoint of the lot, it takes about 200 to 300 steps to complete the process. The model proposed in this chapter attempts to divide the flow into the steps, rather than stages. A stage is a larger unit that combines some sequential steps to form a meaningful action. The tool model concept involves building a model to determine the time required for a step of a lot.

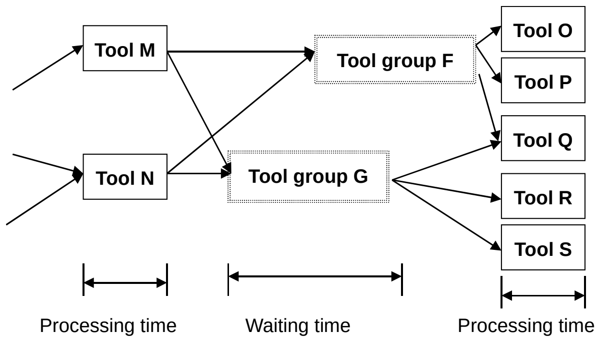


Fig. 6 The relationship of tool, tool group and lot flow

In detail, the tool model can be divided into two parts, the waiting part (wait in the tool group) and the processing part (processed in the tool). The waiting time and processing time are determined by different attributes. Since the tool model is divided into two parts, the lot flow in the tool model is a stream of waiting, processing, waiting, processing and so on. The waiting model covers the waiting time that a lot waits in one tool group, while the processing model deals with the processing time that a lot is processed by a tool.

From the viewpoint of the tool, the processing time in this tool is related to the attributes of the lot, such as recipe, stage, and number of wafers, regardless of the product type or route type. On the other hand, the waiting time for a lot is related to its priority class and the circumstances it suffers. When there are many lots waiting in the tool group, the lot takes a longer time to be processed because it has to compete with other lots to gain the processing resource.

The relationships among the tool, the tool group and the lot flow can be further elaborated. Fig. 6 is a part of the topology of the tools and virtual tool groups in the fab. The tool group frame is dotted to represent the virtual condition. For example, tools O, P and Q are the members of tool group F while tools Q, R and S belong to tool group G. A lot K just finishing processing in tool M is preceding to the next step, tool group G. If tools Q, R and S are busy, then the lot K waits in tool group G, where there may be some lots already waiting inside. The higher priority lot will be chosen when a tool is free.

Hence, the priority is a critical attribute for deciding the waiting time in a tool group. The cycle time of a lot is the summation of the time at each step as

$$CT(i) = \sum_{s=1}^{N} (W_s + P_s) \qquad (1)$$

where N is the total numbers of steps for the lot. W_s and P_s are the waiting time and processing time in step s respectively.

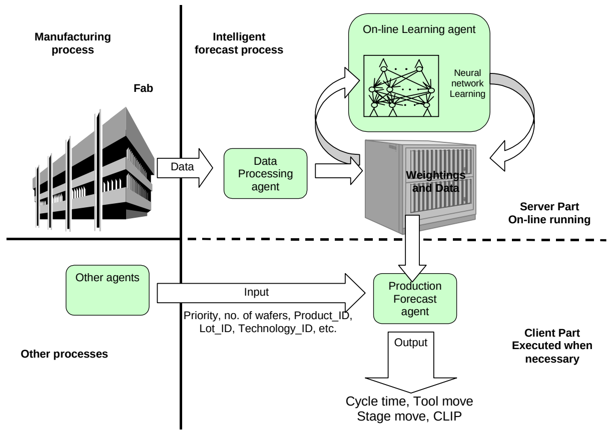


Fig. 7 The client/server architecture of the intelligent forecast process

Based on the decomposed tool model architecture, the scale and complexity of modeling are reduced enormously. Neural networks are used to forecast the waiting time and processing time of both models. A waiting model is used to forecast the waiting time of this tool group. The number of waiting models is equal to the number of tool groups. Similarly, a processing time model is used to forecast the processing time of the tool, and the number of processing models is equal to the number of tools.

Table 1 The important attributes related to time estimation

Attributes	Description
Recipe	Different recipe represents different operation and processing time.
Technology	Different technology means that a lot will be fabricated into different types.
Technology group	A higher-level classification of technology.
Product type	Different products will have different processing properties including technology and route, etc.
Priority class	Each lot is assigned an index based on a least slack policy in order to commit to a due date. The priority represents the urgency of the lot.
Number of wafers in a lot	Usually, the processing time is proportional to the number of wafers. It also plays an important role in the waiting time model because a pod bustled with wafers will be selected first in order to increase production.
Number of pods in five priorities	This means the number of waiting lots in a tool group for each priority. This attribute is used to represent the competence of the waiting lots competing for resources (tools). The larger the number of the high priority pods is, the larger the waiting time for lower priority pods will be.
Push index	If there are too many waiting wafers in a tool group, the tool group should push the wafers to the next tool. The case may be due to a breakdown or lack of the tool capability so that the processing and waiting times abnormally increases.
Pull index	The pull index ensures that a feasible amount of waiting wafers is pulled from the previous tools to avoid a tool being idle. If the pull index from the downstream step is higher, a lot will be chosen with a higher probability in order to alleviate the starvation of a tool group in the next step.
Month index	This index represents the current day in a month.
Day index	This index represents the current time in a day.

Time index	This describes the urgency of a lot relative to its due date.

Cycle time estimation

The cycle time estimator applies tool model to calculate the following information.

(1) The cycle time of the lot i (CT(i)) is the summation of the time of each step. Namely,

$$CT(i) = \sum_{i=1}^{N} (W_i + P_i)$$

(2) The number of steps that the lot will go through in a day can be obtained by

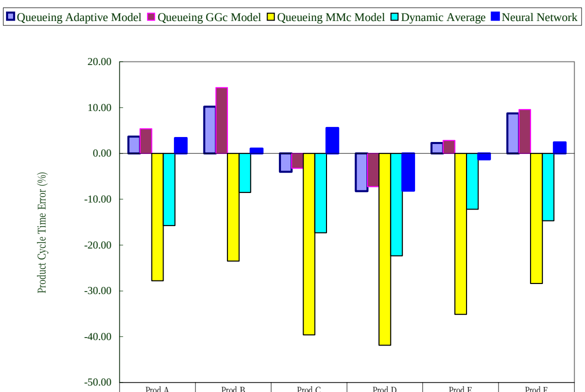
$$S(i) = \left\{ \arg\min_{s} \left\{ \sum_{i=k+1}^{s} (W_i + P_i) > 1\,day \right\} \right\} - k$$

where k is the current step, and s is the step that the lot i will arrive one day later.

(3) The ERCT of lot i which supports least slack policy can be obtained by

$$ERCT(i) = \sum_{i=m+1}^{N} (W_i + P_i)$$

where m is the current step and N is the total steps of the lot i.



	Prod A	Prod B	Prod C	Prod D	Prod E	Prod F
Queueing Adaptive Model	3.69	10.19	-4.04	-8.26	2.27	8.72
Queueing GGc Model	5.36	14.35	-3.27	-7.26	2.83	9.57
Queueing MMc Model	-27.81	-23.49	-39.62	-41.88	-35.13	-28.38
Dynamic Average	-15.75	-8.52	-17.31	-22.36	-12.18	-14.70
Neural Network	3.350	1.040	5.558	-8.134	-1.289	2.393

Product ID

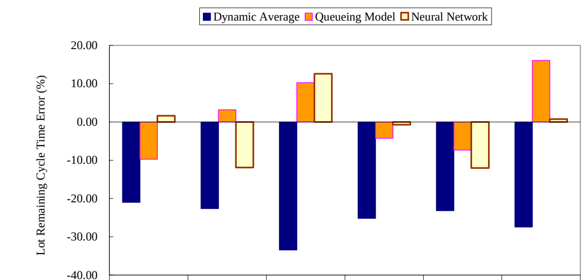
Fig. 8 The comparison of product cycle time

四、結論與成果

Three approaches are compared to forecast the time of each step. The followings are the comparison results of the product cycle time and lot remaining cycle time forecast generated by each approach. The data are retrieved from the real fab. The products and lots for comparison are randomly selected from the above data sets. The time horizon of the data set is 1998/11/1-1998/12/31.

Fig. 8 shows the product cycle time forecasting error by each approach. In the queueing model, three different models are used, MMc, GGc and queueing adaptive model. From the above result, we can find that MMc model has a larger error of all. The reason is that the patterns of all tool groups within the complex circumstance in the fab are not exponential distribution. On the other hand, GGc has much more accuracy than MMc. However, it is not satisfied. Institutionally, we can adopt MMc model in the tool groups whose pattern is close to exponential distribution and GGc model in the other tool groups. This is so called queueing adaptive model and the result also shows that adaptive model has more accuracy than MMc and GGc model.

We can find that the error percentage of the dynamic average method is larger than the queueing adaptive model and the neural network because the operation is modeled more roughly than the others. Because of real time updating the information from the fab, on-line learning by neural networks is better than the other two approaches in modeling of the operation. In average, on-line learning approach is better than queueing adaptive model from the comparison in Fig. 8. In the second part, we randomly select five lots to forecast the remaining cycle time of the lots. The results are shown in Fig. 9



	Lot A	Lot B	Lot C	Lot D	Lot E	Lot F
Dynamic Average	-21.00	-22.62	-33.41	-25.19	-23.19	-27.44
Queueing Model	-9.72	3.18	10.26	-4.22	-7.33	16.09
Neural Network	1.62	-11.88	12.62	-0.73	-12.03	0.76

Lot ID

Fig. 9 The comparison of lot remaining cycle time

Table 2 Comparison of the three algorithms.

	Dynamic average	Queueing model	On-line learning
Accuracy	**	****	*****
Modeling detail	**	****	*****
Calculation complexity	*****	***	**
Calculation time	*****	****	****

Only queueing adaptive model is used to be the representative of the queueing model. We can find that that the more detailed model, queueing adaptive and neural network model have much better forecasting ability than dynamic average method. In some cases, on-line learning approach has less forecasting error even down to below 2%. The result is the same as product cycle time estimation and our institution. Table 2 is the comparison of the three algorithms.

五、參考文獻

[1] S. H. Chung, H. W. Huang, “The Design of Constraint-Oriented Target Planning System for Wafer Fabrication Factories,” Joint Conference of the 5th International Conference on Automation Technology, Taiwan, 1998.

[2] C.-H. Kuo, H.-P. Huang, “Failure Modeling and Process Monitoring for Flexible Manufacturing Systems Using Colored Timed Petri Nets,” IEEE Transactions on Robotics and Automation, vol.16, no.3, pp.301-312, June, 2000.

[3] C.H. Lu, D. Ramaswamy, P.R. Kumar, “Efficient Scheduling Policies to Reduce Mean and Variance of Cycle-Time in Semiconductor Manufacturing Plants,“ *IEEE Transaction on Semiconductor Manufacturing*, vol.7, no.3, pp.374-388, 1994.

[4] M.A. Quaddus, “A Generalized Model of Optimal Due-Date Assignment by Linear Programming,” *Journal of Operation Research Society*, vol.38, no.4, pp.353-359, 1987.

[5] A. A. Rodriguez and M. Kawski, “Modeling and Robust Control of Re-entrant Semiconductor fabrication Facilities: Design of Low-Level Decision policies,” *A Proposal to Intel Research Council*, 1994.

[6] J. L. Snowdon and J. C. Ammons, “A Survey of Queueing Network Packages for the Analysis of Manufacturing Systems,” *Manufacturing Review*, vol.1, no.1, pp.14-25, 1988.

[7] L. Sattler, “Using Queueing Curve Approximations in a Fab to Determine Productivity Improvements,” *Proceedings of 1996 IEEE/SEMI Advanced Semiconductor Manufacturing Conference*, pp.140-145, 1996.

[8] K. S. Tsakalis, “Hierarchical Modeling and Control for Re-entrant Semiconductor Fabrication Lines: A Mini-Fab Benchmark,” *IEEE/SEMI Advanced Semiconductor Manufacturing Conference*, pp.508-513, 1997.

[9] C.Y. Yu and H. P. Huang, “Fab Model Based on Distributed Neural Network,” *National Conference on Automation Technology*, ChiaYi, Taiwan, pp.271-277, 1999.

[10] C.Y. Yu, H.P. Huang, J.Y. Juang, C.F Yeh, ”Cycle Time Estimation for Highly Product-Mixed Semiconductor Foundry,” *The First Cross Strait Workshop on Manufacturing*, Taipei, 2000.

[11] C.Y. Yu, S.W. Chen, H.P. Huang, “Estimation of Waiting Time for Products in an IC Fab, ” *International. Conference on Automation Technology,* Taipei, Taiwan, 1998.

[12] C.Y. Yu, H.P. Huang, "On-Line Learning Delivery Decision Support System for Highly Product Mixed Semiconductor Foundry," *IEEE Transactions on Semiconductor Manufacturing*, vol. 15, no. 2, pp. 1-5, May 2002.