行政院國家科學委員會專題研究計畫 成果報告

RFID 系統發展及其產業應用--子計畫三：以 RFID 為基之診斷維修系統(3/3)

研究成果報告(完整版)

計 畫 類 別 ：整合型
計 畫 編 號 ：NSC 95-2218-E-002-015-
執 行 期 間 ：95 年 08 月 01 日至 96 年 07 月 31 日
執 行 單 位 ：國立臺灣大學機械工程學系暨研究所

計 畫 主 持 人：黃漢邦
共 同 主 持 人：陳正剛
計畫參與人員：碩士班研究生-兼任助理：陳恬穎、黃飛龍、張英婷
協同主持人：陳正剛

處 理 方 式 ：本計畫可公開查詢

中 華 民 國 96 年 09 月 04 日

行政院國家科學委員會專題研究計畫成果報告

RFID 系統發展及產業應用

子計畫三：以RFID為基之診斷維修系統(3/3)

RFID-BASED DIAGNOSIS AND MAINTENANCE SYSTEM (3/3)

計畫編號：NSC 94-2218-E-002-047
執行期限：95/08/01 ~ 96/07/31
主持人：黃漢邦 台灣大學機械工程學系
共同主持人：陳正剛 台灣大學工業工程研究所
研究人員:陳恬穎 台灣大學機械工程研究所
黃飛龍、張英婷 台灣大學工業工程研究所

一、 中英文摘要

本計畫主要發展一個以無線射頻識別(RFID)為基礎的智慧型遠端診斷及維修系統，包括：(1) 第一年-整合RFID技術與使用嵌入式WinCE .NET為作業系統之PDA元件，融入多感測器融合技術及設計資料發送/接收器之配置;(2) 第二年-智慧型遠端診斷維修系統之發展;(3) 第三年-知識管理系統的發展。

透過第二年所建置的智慧型遠端診斷維修系統，利用RFID技術為基的製造系統，整合多種資料探勘方法建構即時重排程系統及決策支援系統。

在即時重排程系統方面，提出了變動式間隔的排程機制。為了使全廠可以依照各個intrabay的屬性採用不同的派工法則，首先，先利用RFID技術萃取相關資訊，用k-means聚成幾群，再利用基因演算法搜尋出各群在不同狀態下哪些派工法則會使全廠效能較佳。排程器的訓練資料由蒐集各個狀態所對應的派工法則而得，以建立KNN排程器或SVM排程器，亦可先利用GDA演算法萃取出新的特徵值後再建排程器。另外，建立數個ANFIS預測模型以達成線上決定排程間隔。針對兩個規模不一樣的半導體廠，經模擬後，實驗結果驗證了所提出的知識管理方法是有效的。

另一方面，決策支援系統包含三個子系統：處理緊急插單子系統、診斷與預測性維修子系統及知識管理子系統，並可以和兩個半導體廠模型做連結以進行模擬、設定參數及觀察效能。至於知識管理子系統的部份，本計劃提出了知識萃取、學習和更新的方法，於每次更新後，能維持知識的品質。最後，提供四個支援決策的劇本，分別為決定產品混合比例、控制產品輸入、決定派工法則及掌握預防性維修時間之劇本，透過網頁可達到知識分享。

已完成之具體工作成果為 (1)透過RFID資訊平台得到即時資訊;(2)知識管理系統的建立，利用web服務技術，與RFID資訊平台達到資訊萃取、遠端診斷監控互動的功能;(3)各種知識管理與探勘方法的比較與實作。

關鍵詞：無線射頻識別、診斷、知識管理、派工法則、決策支援

Abstract:

This project aims to develop an intelligent remote diagnosis and maintenance system which is based on the brand-new technology RFID including: (1) the first year - combine RFID technology with PDA (embedded WinCE .NET operating system) using the multi-sensor fusion algorithm and design the allocation of the transmitter and receiver of RFID; (2) develop of the intelligent remote diagnosis and maintenance system in the second year; (3) the third year - construct the knowledge management system.

Through the intelligent remote system constructed in the second year, RFID technology and various data mining methods are integrated to construct the on-line rescheduling system and the decision support system for manufacturing environment. For on-line rescheduling system, an interval variant rescheduling mechanism is proposed. In order to deploy different dispatching rules to different intrabays, we use RFID techniques to extract related information, and k-means is used for clustering. Then genetic algorithm (GA) is employed for searching dispatching rule sets which promote better performance. In terms of the system conditions corresponding to dispatching rules, SVM (Support Vector Machine) classifier is constructed as scheduler. In addition, the ANFIS (Adaptive Neuro-Fuzzy Inference System) prediction model is built for the sake of on-line deciding the scheduling intervals. The experiment results indicate that applying the proposed Knowledge Management mechanism to obtaining dispatching strategies is an effective method

considering the complexity and variation of semiconductor wafer fabrication systems.

The decision support system communicates with the fab model and contains three subsystems. They are rush order handling subsystem, diagnosis and maintenance subsystem, and knowledge management subsystem. In particular, the methods for knowledge extraction, learning, and update are provided. Also, four scenarios are provided to support decision making. All these scenarios are through web pages to achieve knowledge sharing.

In the third year, the achievements that have already been completed are: (1) obtain real-time information via RFID information platform; (2) construction of Knowledge Management system; (3) comparison and implementation between all kinds of knowledge management methods.

Keywords: RFID, Diagnosis, Knowledge Management, Dispatching Rules, Decision Supporting

二、 緣由與目的

Since the RFID technology has boosted rapidly and been integrated with manufacturing system, it demands effective scheduling, especially in complex manufacturing systems such as semiconductor wafer manufacturing. Scheduling of semiconductor wafer fabrication systems is a complicated and difficult task because of their distinguishing characteristics, such as re-entrant product flow, high uncertainties in operations, and rapidly changing products and technologies. Among them, the re-entrant product flow makes production planning and scheduling of wafer fabrication difficult. Wafers at different stages of their life in a fab have to compete with each other for the same machines. Thus, wafers need to spend a larger amount of their time simply waiting for machines, rather than for processing. The system uncertainties include machine failure, uncertain process yield, and rework. The unreliable machines disrupt the flow of wafers in a fab and cause the cycle time to increase and fluctuate. It is thus a significant challenge to develop effective scheduling methods in a wafer fabrication system.

For another aspect, there is general agreement on the importance of knowledge management applying to the manufacturing systems. In the human resources field, knowledge management embraces the ability to define the skills required to perform a manufacturing operation, and to teach those skills for consistent, effective execution. To the information technologist, it may be a strategy for assuring that knowledge is systematically captured, updated, and available in a useful form to all who need it. To senior executives, it usually implies information and tools to assure effective decisions. To the manufacturing strategist, it may be the capture of best knowledge to create design advisors and automated information systems to assure that the best product and process designs are created and executed. It is important to note that, no matter what the perspective, knowledge management must always be applied to enable organizational success and never to store information for information's sake. All levels of the organization must be enabled to react with instinct and creativity towards an internal improvement or an external market opportunity.

三、 文獻探討

3.1 Radio Frequency Identification System

RFID is a wireless communication technology applicable to various applications such as distribution, storage management, healthy care, campus security, etc. RFID systems consist of Radio Frequency (RF) tags, networked RF tag readers and antennas. Through radio, a reader can read the data from remote tags whether there are some obstacles between them.

A reader can identify several different signals from different tags at the same time and collect data at a high rate.
Tags can update the stored data dynamically.

There are other features, such as durability, real-time transmission, etc.

A discussion is warranted to better understand RFID. The following sections briefly describe some topics of RFID.

Radio frequency identification (RFID) is expected to become pervasive and ubiquitous, as it can be embedded into everyday items as smart labels. This traditional and emerging technology has attracted considerable press attention in recent years. It has been used for automatic data collection since World War II, but its growth has been relatively slow through the past decades until the Wal-Mart publicly embraces this technology, the future of RFID looks increasingly brighter.

From a functional point of view, an RFID system consists of three components: tag (transponder), reader and software application.

Tags can either be passive or active. Passive tags are those which work without a power source. Active tag has an embedded power source. Tags which identification data can be embedded are devices that identify the item to which they are attached. RFID tags are also called transponders or contact-less data carriers.

Antenna is a device which is connected with reader and uses radio waves to read and write data to the tags. The antenna is mounted under deionized water to read/write data to the tags while submerged. Other examples include antennas that offer portals around conveyors or even dock doors. These portals (also called tunnels and gates) read or write to Tags/Labels/PCBs as they pass through.

Reader manages the communication interface between antennas and a PC server or network interface module. Some systems integrate the antenna and reader into a single reader.

Software application reads/writes data from/to tags through the reader. The software application in a workstation or pc initiates all communications between the readers and the tags. Both the readers and the tags are equipped with antennas that receive and emit electromagnetic waves.

3.2 Related Data Mining Methods

Theory of constraints (TOC) is used for dealing the product mix problem. Then other methods are used for on-line rescheduling. Among these methods, k-means is used for clustering the intrabays of the fab, and Genetic Algorithm (GA) [10] is for searching dispatching rule sets which promote better performance. The details of these two methods can be found in [6] and [10], respectively. Other three methods will be introduced below.

A. Theory of Constraints (TOC)

A resource is an internal constraint if and only if the output of the resource is less than the market demand. If a plant faces a multiple constraints situation, the capacity constraint resources (CCRs) may change as the product mix decision is changed. When solving the product mix problem, non-CCRs can be divided into three groups: first-level non-CCRs, second-level non-CCRs and the third-level non-CCRs [7].

Following are the steps of the TOC-based algorithm:

Preparation step: Delete the first-level and the second-level non-CCRs using Lemmas 1 and 2 in [7]. The current product mix solution is the market potential of each product type. Set n = 1, and no previous CCR_i and P_i exist (i < n).

Step 1: Identify the system's constraint. Calculate the load of each resource based on the current product mix solution. Then compare the load of each resource with its capacity limit. The resource with the highest overload amount is identified as the CCRn. If no CCR exists, then stop and the current solution is the final solution.

Step 2: Decide how to exploit the system's constraint

(a) Treat the CCR_n resource constraint as an equality equation. Delete previous P_i (i < n) terms from this equation by manipulating the row operations between CCR_n and CCR_i (i < n) equation.

(b) Make the objective equation be the new one without the previous P_{n-1} term. It also can be done by substituting the value of P_{n-1}, which is derived from the CCR_{n-1} equality equation, into the previous objective equation.

(c) Calculate the "\$/constraint-time" for all product types that have positive coefficients in CCR_n equality equation, according to the new objective equation (step 2-(b)) and the new CCR_n equality equation (step 2-(a)).

(d) Choose the P_n which has the smallest "\$/constraint-time" value for CCR_n. Reduce the quantity of P_n until the load of the CCR_n just equals its capacity limit.

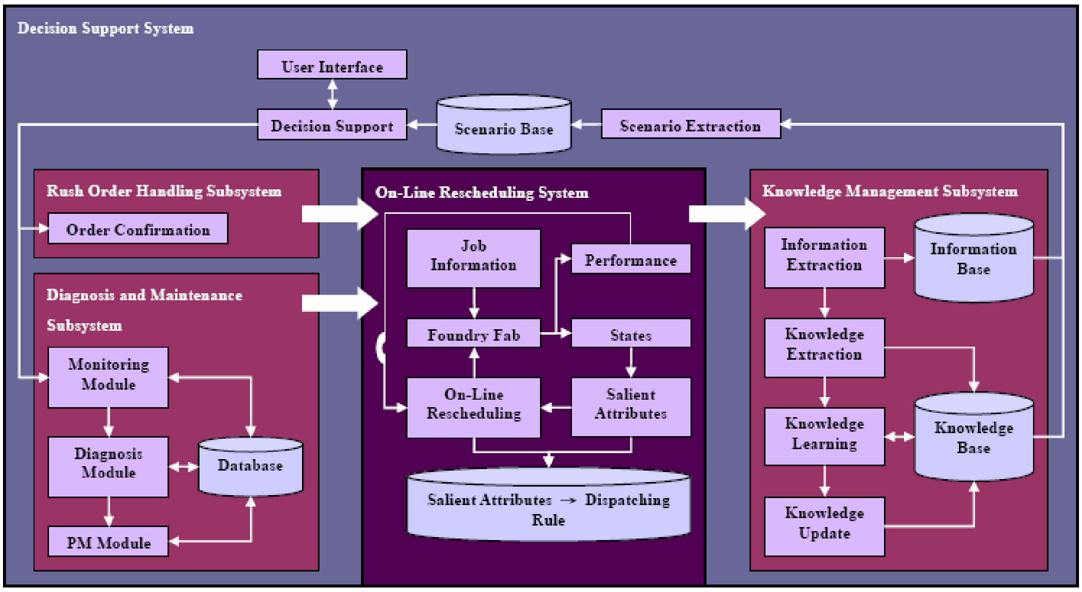


Fig. 1 System architecture.

(e) Starting from CCR_{n-1} through CCR_1, adjust the quantity of previous P_i (i < n) to make the load of

CCR$_i$ (which is changed after performing step 2-(d)) equal to its capacity limit again. Then, the new product mix is derived. Let n = n + 1 and go to step 1.

B. Support Vector Machine (SVM)

The original idea of support vector machine (SVM) classification is to use a linear separating hyperplane to create a classifier. The support vector technique tries to find the separating hyperplane with the largest margin between two classes, measured along a line perpendicular to the hyperplane. This means it would find a line with parameters **w** and b such that the distance between $\mathbf{w}^{\mathbf{T}}\mathbf{x} + b = \pm 1$ is maximized [16].

However, practical problems may not be linearly separable. For this reason, SVM introduces slack variables ξ to original objective function and constraints [5].

If data are distributed in a highly nonlinear way, employing only a linear function causes many training instances to lie on the wrong side of the hyperplane. So underfitting occurs and the decision function does not perform well. To fit the training data better, a nonlinear curve should be used. Instead of adopting sophisticated curves, SVM maps data into a higher dimensional space. In higher dimensional space, it is more possible that data can be linearly separated. SVM then tries to find a linear separating plane in a higher dimensional space. To solve the problem effectively, the dual problem is dealt.

The remaining issue of using the dual problem is about the inner product $\phi(x_i)^T\phi(x_j)$. If $\phi(x)$ is an infinite-long vector, there is no way to fully write it down and then calculate the inner product. This is resolved by using special kernel function $K(x_i, x_j) = \phi(x_i)^T\phi(x_j)$ so that $\phi(x_i)^T\phi(x_j)$ is efficiently calculated. Some popular kernels are polynomial kernel and radial basis function (RBF) kernel. For example, the RBF kernel is $e^{-\gamma\|x_i - x_j\|^2}$.

The discussion so far assumes that data are in only two classes. Many practical applications involve with more classes. There are many ways to extend SVM for such cases. Two simple methods are one-against-all and one-against-one. The detail of these methods for multi-class support vector machines can be found in [8].

C. Adaptive Neuro-Fuzzy Inference System (ANFIS)

In this section, a classic adaptive network which is functionally equivalent to a fuzzy inference system will be introduced [9]. For simplicity, we assume that the fuzzy inference system under consideration has two inputs x and y and one output f. For a first-order Sugeno fuzzy inference system, a common rule set with two fuzzy if-then rules is the following:

Rule1 - If x is A1 and y is B1, then f1 = p1x + q1x + r1,

Rule2 - If x is A2 and y is B2, then f2 = p2x + q2x + r2.

As Fig. 2 shown, there are five layers of the network. Thus an adaptive network which is functionally equivalent to a Sugeno fuzzy inference system has been constructed. Next it shall demonstrate how to apply the hybrid learning algorithm to identify ANFIS parameters. For the hybrid learning, each epoch is composed of a forward pass and a backward pass. TABLE I below summarizes the activities in each pass.

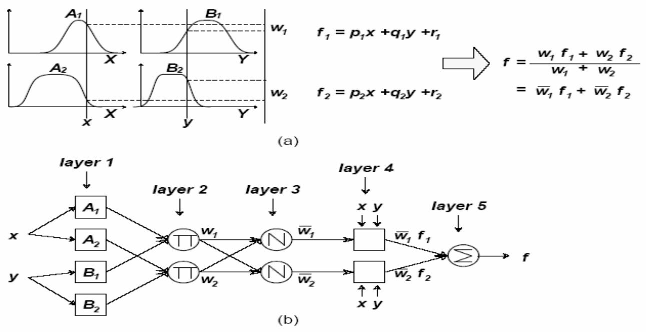


Fig. 2 (a) A two-input first-order Sugeno fuzzy inference system with two rules (b) equivalent ANFIS architecture.

TABLE I. Two passes in the hybrid learning procedure for ANFIS.

	Forward Pass	Backward Pass
Antecedent Parameters	Fixed	Gradient Descent Method
Consequent Parameters	Least Squares Method	Fixed
Signals	Node Outputs	Error Rates

D. Decision Tree

Among several different decision trees, the CART (Classification and Regression Tree) methodology [18] is technically known as binary recursive partitioning. The process is binary because parent nodes are always split into exactly two child nodes and recursive because the process can be repeated by treating each child node as a parent.

CART is to look at all possible splits for all variables included in the analysis. Any problem will have a finite number of candidate splits and CART will conduct a brute force search through them all. Then next activity is to rank order each splitting rule on the basis of a quality-of-split criterion. The default criterion used in CART is the gini index, essentially a measure of how well the splitting rule separates the classes contained in the parent node. When a best split is found, CART repeats the search process for each child node, continuing recursively until further splitting is impossible. Once a terminal node is found, the group with the greatest representation determines the class assignment.

At the time the maximal tree is grown and a set of sub-trees are derived from it, CART determines the best tree by testing for error rates or costs. With sufficient data, the simplest method is to divide the samples into learning and test sub-samples. When studies have insufficient data to allow a good-sized separate test samples, CART employs the technique of cross validation. The upshot of this complex process is a fairly reliable estimate of the tree. This means that how well any tree will perform on completely fresh data can be known.

四、 研究成果

I. ON-LINE RESCHEDULING

The illustration for the interval variant rescheduling is shown in Fig. 3. A long run of the fab consists of several decision points. Between two decision points, it contains some scheduling intervals (SI). The length of the scheduling intervals may differ in different two decision points. At each decision point, the interval variant rescheduling mechanism decides the new scheduling intervals for this decision period with the constructed ANFIS models. And at the beginning of a scheduling interval, the interval variant rescheduling mechanism determines a dispatching rule set for all machines with the constructed SVM models.

The detailed procedures for building the SVM model and ANFIS model are coming next.

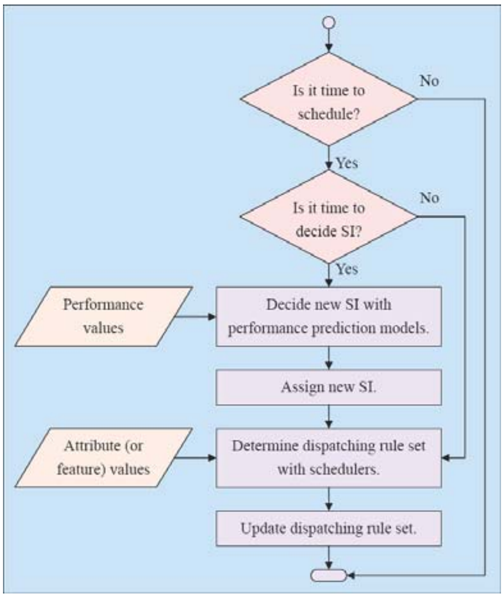


Fig. 3 On-line rescheduling mechanism.

A. Procedure of Training SVM Classification Model

The overall procedure of training SVM classification model is shown in Fig. 4. After a long run of simulation, each attribute value of every intrabay is collected. Traditional scheduling methods make decisions based on the explicit system attributes directly, such as machine utilizations, job tardiness, etc. In summary, six local attributes for the intrabay and one global attribute for the fab are chosen as follows: the mean utilization of the machines in the intrabay, the number of the jobs in the intrabay, the mean sojourn time of the jobs in the intrabay, the mean remaining processing time of the jobs in the intrabay, the mean slack time of the jobs in the intrabay, the mean tardiness time of the jobs in the intrabay, and the level of machine breakdown in the fab.

In order to deploy different dispatching rules to different intrabays, k-Means is used for clustering the intrabays of the fab. In particular, the reason for clustering is to reduce the search space and simulation time. That is, if there are 23 intrabays in the fab and these intrabays are clustered to 5 clusters, and then the search space is reduced from 7^{23} ($2.737\text{x}10^{19}$) to 7^5 ($1.6807\text{x}10^4$).

Then GA is employed for searching dispatching rule sets which promote better performance. With attributes as the system conditions corresponding to dispatching rules, the SVM classifier is constructed as the scheduler. The dispatching rules used in this paper are selected from previous researches [14] [1] and described in TABLE II.

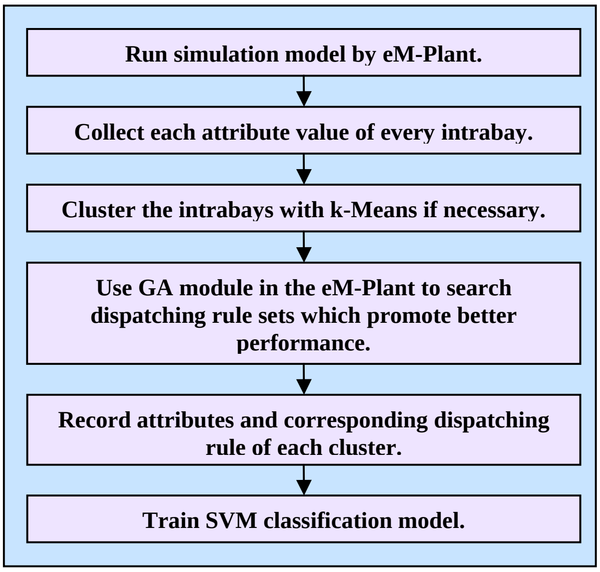


Fig. 4 Procedure of training SVM classification model.

TABLE II. Dispatching rules.

Dispatching Rule	Description	Criterion
FIFO	First In First Out	First in first out

SPT	Shortest Processing Time	Shortest processing time first
TIS	Time In System	Longest one first
SRPT	Shortest Remaining Processing Time	Shortest RPT first
CR	Critical Ratio	Smallest one first
LS	Least Slack	Smallest slack time first
EDD	Earliest Due Date	Earliest due date first

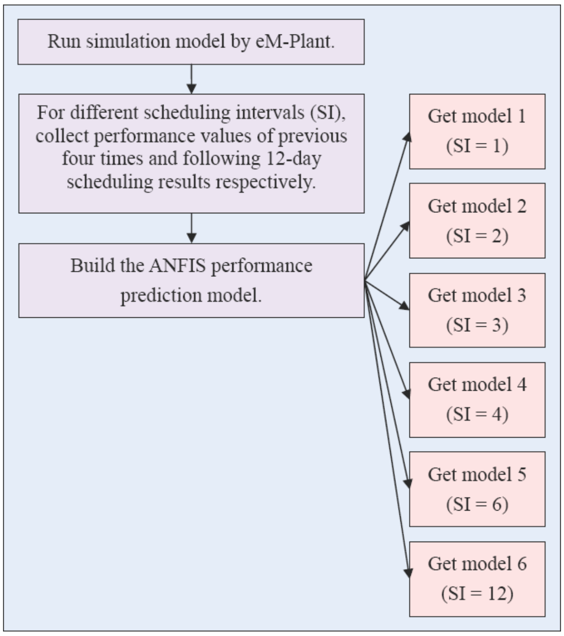


Fig. 5 Procedure of building ANFIS prediction model.

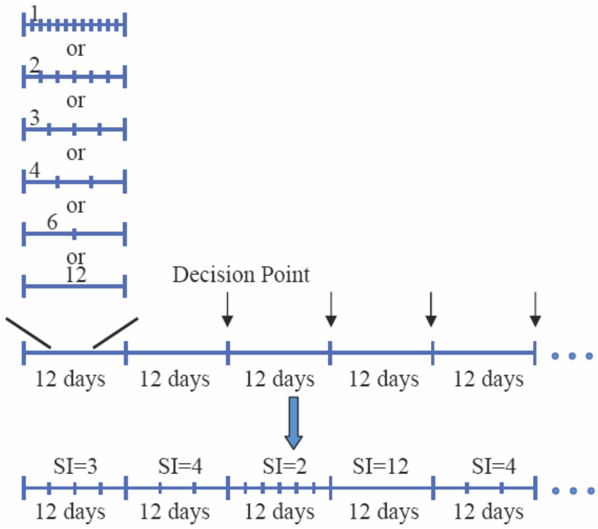


Fig. 6 Sketch map of interval variant rescheduling.

B. Procedure of Building ANFIS Prediction Model

The entire procedure for building ANFIS prediction model is shown in Fig. 5. After observation of a long run simulation, the period between two decision points is set as 12 days. Between two decision points, the length of the scheduling intervals (SI) may be 1 day, 2 days, 3 days, 4 days, 6 days or 12 days. For example, if SI is 2 days, scheduling will be taken six times between these two decision points. A sketch map is shown in Fig. 6.

For the sake of on-line deciding the scheduling intervals between two decision points, six ANFIS prediction models are built, as shown in Fig. 5. Performance values of previous four-time scheduling are fed as the prediction model's input, and then the performance of following 12-day scheduling result is predicted. Predicted performance values of six ANFIS prediction models are compared to decide the actual scheduling intervals between these two decision points.

II. EXPERIMENT RESULTS

The model used in this paper is SEMATECH (SEmiconductor MAnufacturing TECHnology) model following "300 mm Factory Layout and Material Handling Modeling: Phase II Report" by Campbell and Ammenheuser [3]. This model represents a 300 mm wafer fab with 277 tools belonging to 43 different tool groups distributed in 23 intrabays. The wafer fab processes each lot with 316 process steps and the total processing time of each lot is about 6.756 days. For lot releasing, nine lots of 25 wafers are started every 8 hours. In addition, two single-wafer NPW (Non-Product Wafer) lots are released every 24 hours, and one 25-wafer hot lot is released every 72 hours.

Cluster	1	2	3	4	5
Gene	4	1	6	2	7

Fig. 7 Chromosome encoding.

TABLE III. The result of clustering the intrabays.

	Cluster		Cluster
A_Imp	**2**	**K_Etch**	**3**
B_CMP	**3**	**M_Etch**	**3**
D_CMP	**3**	**O_Etch**	**3**
C_Photo	**3**	**Q_Etch**	**3**
E_Photo	**3**	**S_Etch**	**3**
G_Photo	**3**	**L_Furn**	**2**
I_Photo	**3**	**N_Furn**	**1**
F_Wet_BEOL	**4**	**P_Metal**	**5**
H_Wet_BEOL	**4**	**R_Metal**	**5**
J_Wet_FEOL	**2**	**T_Metal**	**4**
U_Test	**3**	**V_CVD**	**3**
		X_CVD	**3**

The result of clustering the intrabays of the fab is exhibited in TABLE III. Intrabays with the same function are almost clustered in the identical cluster.

For running the genetic algorithm, every individual is evaluated for 15 days. The chromosome including 5 genes is encoded, as illustrated in Fig. 7. Each gene represents the corresponding dispatching rule of the cluster. Then, the fitness function needs to be maximized is defined according to the epsilon-constraint method [11]. That is, maximize throughput subject to tardy number less and equal to an epsilon value. As the sufficient training data is collected, SVM models for each cluster can be built.

The following experiment results give the mean outcomes over 30 runs to provide a comprehensive comparison. In these experiments, the simulation for each run is 60 days. TABLE V exhibits the experiment results of static scheduling using different dispatching rules. TABLE VI demonstrates the experiment results of interval constant rescheduling. Then the experiment results of interval variant rescheduling is revealed in TABLE VII. Also, dispatching rules for each cluster during on-line rescheduling is displayed in TABLE IV.

TABLE IV. Dispatching rules for each cluster.

	Dispatching Rules
Cluster 1	FIFO, SPT, CR, EDD
Cluster 2	SRPT, LS, EDD
Cluster 3	TIS, LS, EDD
Cluster 4	SPT, EDD
Cluster 5	SPT, TIS, SRPT, LS, EDD

TABLE V. Experiment results of static scheduling using different dispatching rules.

	Throughput (Wafer)		**Tardy Number (Wafer)**		**(Throughput - Tardy Number)**	
	Avg.	**Std.**	**Avg.**	**Std.**	**Avg.**	**Std.**
FIFO	36986.67	1135.98	1433.33	438.49	**35553.33**	**1528.63**
SPT	39057.50	91.62	1815.00	67.69	**37242.50**	**111.46**
TIS	39439.17	88.72	455.00	3.26	**38984.17**	**87.60**
SRPT	39170.00	84.42	703.33	24.47	**38466.67**	**97.32**
CR	38833.33	92.94	185.00	30.31	**38648.33**	**87.07**
LS	39354.17	85.11	72.50	5.27	**39281.67**	**85.05**
EDD	39294.17	83.29	78.33	5.72	**39215.83**	**80.93**

TABLE VI. Experiment results of interval constant rescheduling.

	Throughput (Wafer)		**Tardy Number (Wafer)**		**(Throughput - Tardy Number)**	
	Avg.	**Std.**	**Avg.**	**Std.**	**Avg.**	**Std.**
SI = 1	39565.00	64.18	78.33	5.72	**39486.67**	**63.42**
SI = 2	39597.50	75.17	79.17	5.88	**39518.33**	**74.97**
SI = 3	39579.17	70.71	80.83	6.19	**39498.33**	**70.72**
SI = 4	39620.00	72.60	80.00	5.67	**39540.00**	**72.86**
SI = 6	39580.83	72.33	82.50	5.89	**39498.33**	**72.35**
SI = 12	39615.83	72.22	85.83	5.96	**39530.00**	**71.84**

TABLE VII. Experiment result of interval variant rescheduling.

	Throughput (Wafer)		**Tardy Number (Wafer)**		**(Throughput - Tardy Number)**	
	Avg.	**Std.**	**Avg.**	**Std.**	**Avg.**	**Std.**
Variant SI	40495.00	62.20	22.50	4.54	**40472.50**	**62.53**

III. DECISION SUPPORT SYSTEM

This section is devoted to the proposed method for knowledge extraction, learning, and update. Right after that is the implementation of the overall decision support system.

A. Knowledge Extraction, Learning, and Update

Whereas data are a collection of facts, measurements, and statistics, information is organized or processed data that are timely and accurate. The information extracted here contains throughput, WIP, each product's mean flow time, tardy number, intrabay attributes, tool parameters, job conditions, etc. Then some information can be extracted to knowledge.

The overall procedure for knowledge extraction, learning, and update is shown in Fig. 8. The first step is to reduce the amount of information and to get the representative information so that the size of built decision tree will be kept and small. Then, utilize the representative information to build decision tree or GDA-based [2] decision tree. The reason for building decision tree is that it is easy to transform it to rules as knowledge. And GDA-based decision tree is used to enhance the accuracy rate. When accumulating a certain amount of information, use k-means to update the representative information. The updated representative information is taken to build a new decision tree, so the accuracy rate will not be down.

The information here is extracted from the SEMATECH Model's system attributes. Fig. 9 shows the position of representative information with update, and it can be seen that the representative information is only be fine tuned. Fig. 10 demonstrates the appearance of decision tree with update. It can be observed that the circled nodes' attributes are almost the same with update and theirs cut values are only be fine tuned. For the un-circled nodes, they may be new nodes or will be automatically pruned with update. Fig. 11 shows the accuracy rate of knowledge with update and Fig. 12 reveals the structural complexity of knowledge with update.

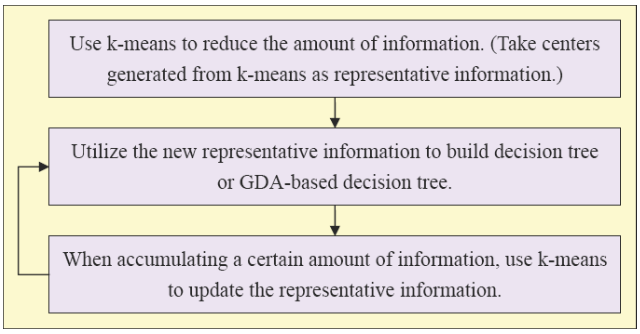
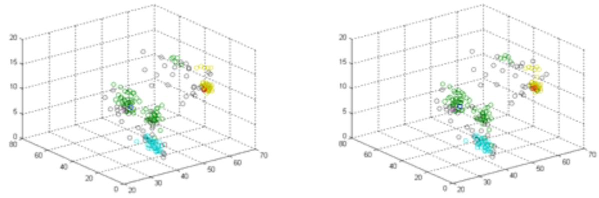


Fig. 8 Procedure for knowledge extraction, learning, and update.



(a) (b)

Fig. 9 Position of representative information with update.

(a) (b)

Fig. 10 Appearance of decision tree with update.

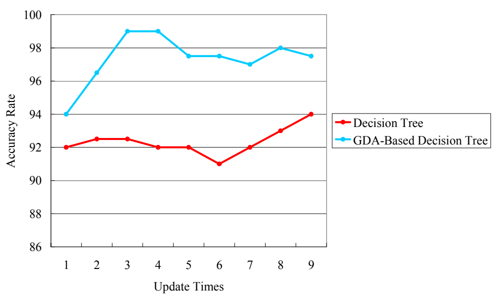


Fig. 11 Accuracy rate of knowledge with update.

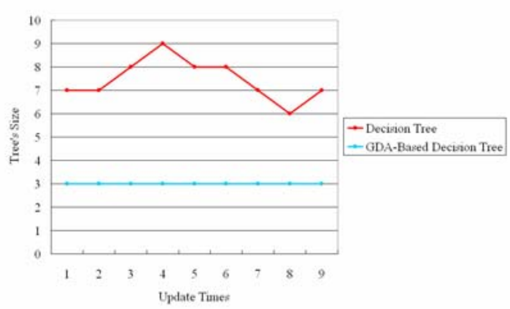


Fig. 12 Structural complexity of knowledge with update.

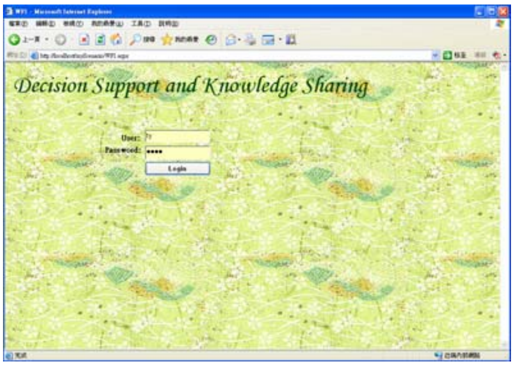
B. Decision Scenarios

Four scenarios are provided to support decision making. The first offers decision makers to decide product mix ratio with the concept of TOC. One key idea of TOC is that the system's constraints (also called bottleneck resources or capacity constraint resources, CCRs) determine the system's performance and should be the focus of management attention. In order to acquire the highest profit attainable, the CCR must be fully utilized [4]. TOC approach decides the product type and the corresponding quantity to produce a given market potential.

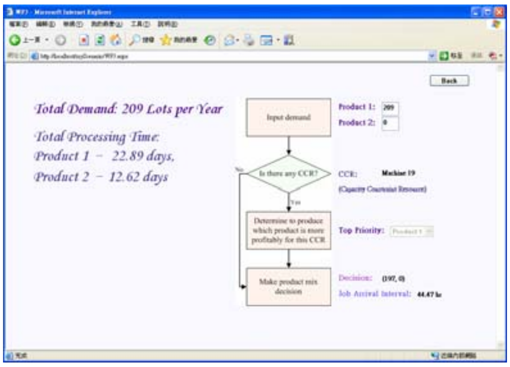
The second scenario can control job arrival rate through the monitoring of WIP. And then the third one supplies to decide dispatching rules in terms of the knowledge extracted from the method proposed in section A. The last scenario aids preventive maintenance with the information of each machine's PM schedule. All these scenarios are through web pages to achieve knowledge sharing (see Fig. 13(a) to Fig. 13(d)).

C. Implementation

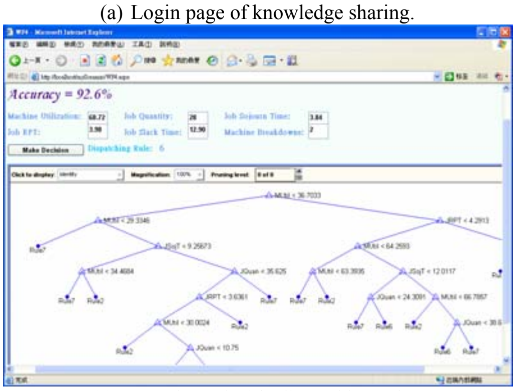
Fig. 13(e) shows the interface of the decision support system. It communicates with the fab model (Fig. 13(e) and Fig. 13(f)) and contains three subsystems. They are rush order handling subsystem, diagnosis and maintenance subsystem (Fig. 13(g) and Fig. 13(h)), and knowledge management subsystem (Fig. 13(i) and Fig. 13(j)). So here are two MPAS (Multi-Pass Adaptive Scheduling) strategies to determine the best dispatching rule: the simulation-based approach and the knowledge approach. Additionally for rush order handling, verify due date to assure delivery on time and cooperate with an event-driven rescheduling mechanism. That is, if any job's slack time is less than 12 hours, dispatching rules of the whole fab will be changed to LS (Least Slack) rule. Fig. 13(k) and Fig. 13(l) show the outputs of hybrid rescheduling.



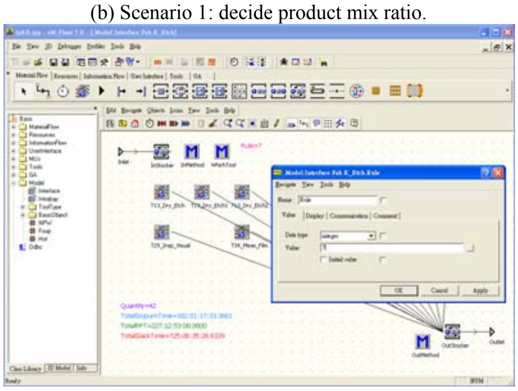
(a) Login page of knowledge sharing.



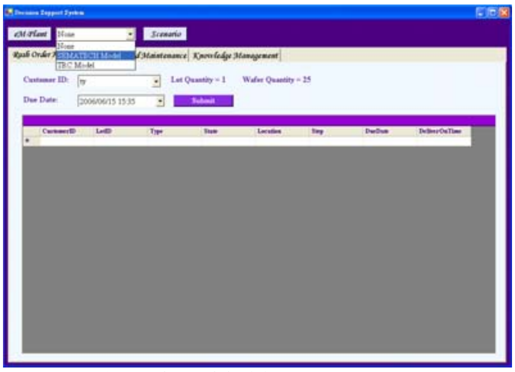
(b) Scenario 1: decide product mix ratio.



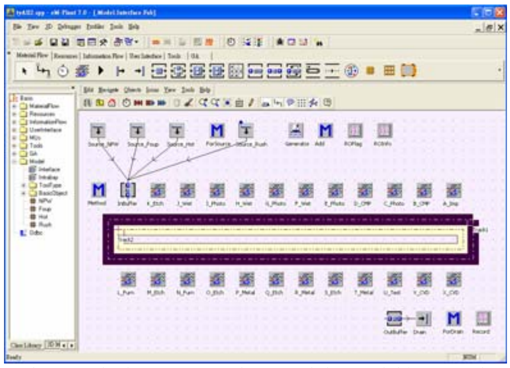
(c) Scenario 3: decide dispatching rules.



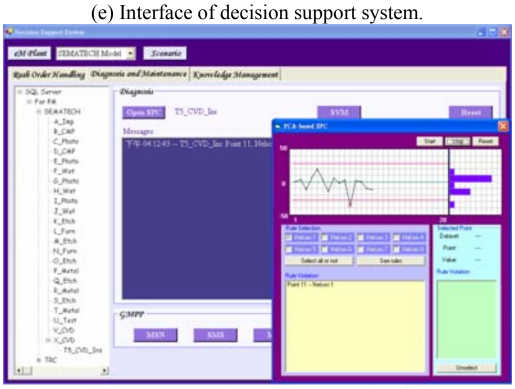
(d) Scenario 3: assign dispatching rule to the intrabay.



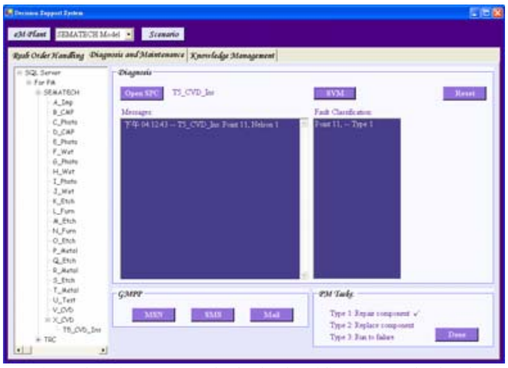
(e) Interface of decision support system.



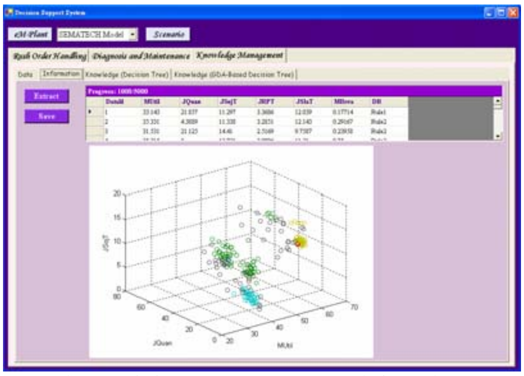
(f) Press the button “eM-Plant” and the model is opened.



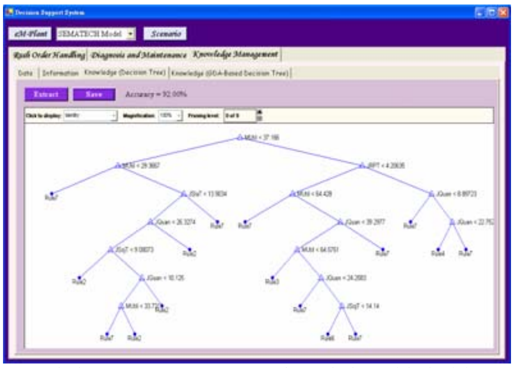
(g) Diagnosis and maintenance: select the concerned tool to monitor and start to monitor the tool.



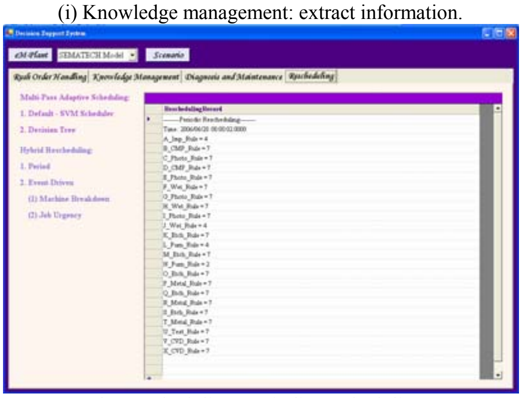
(h) Diagnosis and maintenance: do fault classification and take the PM task.



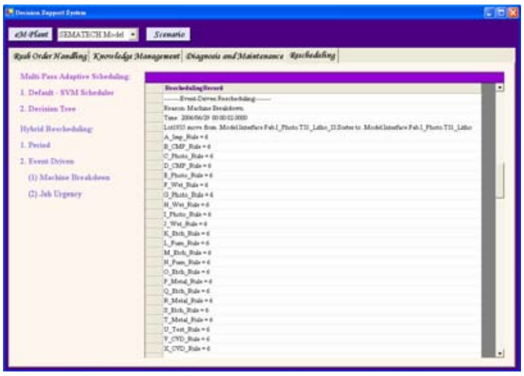
(i) Knowledge management: extract information.



(j) Knowledge management: extract knowledge with decision tree.



(k) Rescheduling: periodic rescheduling.



(l) Rescheduling: event-driven rescheduling.

Fig. 13 Some interfaces of decision support system.

IV. Conclusions

In this paper, for the sake of deploying different dispatching rules to different intrabays, several SVM schedulers are constructed. That is, local rescheduling can be performed. The proposed method can also consider multi-objectives at the same time, i.e., maximize throughput subject to tardy number less and equal to an acceptable value. Finally, on-line deciding the scheduling intervals is achieved. Reflection on the experiment results will make clear that the interval variant rescheduling mechanism presented in this paper is an effective method considering the complexity and variation of semiconductor wafer fabrication systems. Then the decision support system is built containing three subsystems and communicating with the fab model so that decision makers can initiate a simulation experiment, adjust the system parameters, and conduct real-time performance analysis. In knowledge management subsystem, the methods for knowledge extraction, learning, and update are offered. At every moment updating, the quality of the knowledge is assured and the structural complexity of the knowledge is kept and small. Four scenarios for decision making are also provided. These scenarios are through web pages to achieve knowledge sharing.

References

[1] N. Bahaji and M. E. Kuhl, "A Simulation Study of Composite Dispatching Rules, CONWIP and Push Lot Release in Semiconductor Fabrication," *International Journal of Production Research,* submitted for publication.

[2] G. Baudat and F. Anouar, "Generalized Discriminant Analysis using a Kernel Approach," *Neural Computation*, Vol. 12, No. 10, pp. 2385-2404, Jan. 2000.

[3] E. Campbell and J. Ammenheuser, "300 mm Factory Layout and Material Handling Modeling: Phase II Report," *International SEMATECH Technology Transfer Document,* No. 99113848B-ENG, 2000.

[4] S. H. Chung, H. I. Lee, and W. L. Pearn, "Product Mix Optimization for Semiconductor Manufacturing Based on AHP and ANP Analysis," *International Journal of Advanced Manufacturing Technology,* Vol. 25, No. 11, pp. 1144-1156, Jun. 2005.

[5] C. Cortes and V. Vapnik, "Support-Vector Networks," *Machine Learning,* Vol. 20, No. 3, pp. 273-297, Sep. 1995.

[6] R. O. Duda, P. E. Hart, and D. G. Stork, *Pattern Classification,* 2nd Edition, New York: Wiley, pp. 526-528, 2000.

[7] T. C. Hsu and S. H. Chung, "The TOC-Based Algorithm for Solving Product Mix Problems," *Production Planning and Control,* Vol. 9, No. 1, pp. 36-46, Jan. 1998.

[8] C. W. Hsu and C. J. Lin, "A Comparison of Methods for Multi-Class Support Vector Machines," *IEEE Transactions on Neural Networks,* Vol. 13, No. 2, pp. 415-425, Mar. 2002.

[9] S. R. Jang, "ANFIS: Adaptive-Network-Based Fuzzy Inference System," *IEEE Transactions on Systems, Man and Cybernetics,* Vol. 23, No. 3, pp. 665-685, Jun. 1993.

[10] S. R. Jang, C. T. Sun, and E. Mizutani, *Neuro-Fuzzy and Soft Computing,* 1st Edition, New Jersey: Prentice-Hall, pp. 175-180, 1997.

[11] R. T. Marler and J. S. Arora, "Survey of Multi-Objective Optimization Methods for Engineering," *Structural and Multidisciplinary Optimization,* Vol. 26, No. 6, pp. 369-395, Apr. 2004.

[12] H. S. Min and Y. Yih, "Selection of Dispatching Rules on Multiple Dispatching Decision Points in Real Time Schedule of Semiconductor Wafer Fabrication Systems," *International Journal of Production Research,* Vol. 41, No. 16, pp. 3921-3941, Nov. 2003.

[13] R. Rangsaritratsamee, W. G. Ferrell, and M. B. Kurz, "Dynamic Rescheduling that Simultaneously Considers Efficiency and Stability," *Computers and Industrial Engineering,* Vol. 46, No. 1, pp. 1-15, Mar. 2004.

[14] C. R. Tsai, "Development of a Real-Time Scheduling and Rescheduling System Based on RFID for Semiconductor Foundry Fabs," *Master Thesis,* Graduate Institute of Industrial Engineering, National Taiwan University, 2005.

[15] R. Uzsoy, C. Y. Lee, and L. A. Matrin-Vega, "A Review of Production Planning and Scheduling Models in Semiconductor Industry, Part 1: System Characteristics, Performance Evaluation and Production Planning," *IIE Transactions on Scheduling and Logistics,* Vol. 24, No. 4, pp. 47-59, 1992.

[16] V. Vapnik, *Statistical Learning Theory,* 1st Edition, New York: Wiley, 1998.

[17] G. E. Vieira, J. W. Herrmann, and E. Lin, "Analytical Models to Predict the Performance of a Single-Machine System Under Periodic and Event-Driven Rescheduling Strategies," *International Journal of Production Research,* Vol. 38, No. 8, pp. 1899-1915, May 2000.

[18] CART: http://www.salford-systems.com/421.php