Research Article

A Fuzzy Collaborative Sensor Network for Semiconductor Manufacturing Cycle Time Forecasting

Yu-Cheng Lin, Toly Chen, and Yu-Cheng Wang

1 Department of Industrial Engineering and Management, Overseas Chinese University, 100 Chiao Kwang Road, Taichung 408, Taiwan
2 Department of Industrial Engineering and Systems Management, Feng Chia University, 100 Wenhwa Road, Seatwen, Taichung City 408, Taiwan

Correspondence should be addressed to Toly Chen; tcchen@fcu.edu.tw

Received 26 September 2012; Revised 12 March 2013; Accepted 14 March 2013

Academic Editor: Tai-hoon Kim

Copyright © 2013 Yu-Cheng Lin et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Network based sensing has become an important field of research, and it is expected that new applications of remote sensing will be developed. A fuzzy collaborative sensor network is developed in this study to predict the cycle time of a job in a semiconductor manufacturing factory, which is an important task for the factory. In the fuzzy collaborative sensor network, each sensor detects the status of a particular job as well as various environmental conditions present in the factory and uses a fuzzy neural network to analyze the received information. Each sensor communicates its settings and forecasting results to other sensors with the aid of a central control unit. According to the experimental results, the aggregate forecasting performance was considerably improved through the sensors’ collaboration.

1. Introduction

A sensor network is a remarkable technological achievement because of its promising use of human-unattended information collection [1]. Wang et al. [1] argue that there are two performance measures in evaluating the optimal routing performance of a sensor network—the network lifetime finally acquired and the total information finally collected. However, in different circumstances, these measures may not be the same. In Yan et al. [2], it was found that opportunistic collaboration has better performance than direct transmission while Sang et al. [3] developed a sensor network “testbed.” A good literature review of sensor networks can be found in Yick et al. [4]. Network based sensing has become an important field of research and new developments in remote sensing are expected. For example, a synchronized sensor network system for vibration measurement was developed in Uchimura et al. [5]. It is now possible to obtain environment information from difficult to reach places [6]. In Morreale’s opinion, sensor networks have potential applications in urban "telehealth" [7].

A fuzzy collaborative sensor network is constructed in this study to predict the cycle time of a job in a semiconductor manufacturing factory, which is an important task in a modern factory environment [8–11]. The cycle time of a job is the time required to complete its operations in the factory, which can take up to several months. After the cycle time of each job in a factory is accurately predicted, several managerial goals (including internal due-date assignment, output projection, ordering decision support, enhancing customer relationships, and guiding subsequent operations) can be simultaneously achieved [12–16].

The existing approaches used to predict the cycle time of a job in a semiconductor manufacturing factory can be divided into the following categories: statistical analysis, simulation, artificial neural networks (ANNs), case-based reasoning (CBR), fuzzy theory, and hybrid approaches [15, 16]. A comprehensive comparison of these methods can be found in Chen [19]. In recent years, other scholars have engaged in similar research, which focuses on job cycle time forecasting using hybrid approaches. Some researchers classified jobs before forecasting the cycle times, for example,
the pre-classifying approaches (e.g., [18, 20–22]). With pre-classifying approaches, jobs with similar attributes are classified into the same category according to job attributes. However, there is no absolute measure of the similarity between two jobs. On the contrary, in postclassification approaches, jobs with the same cycle time forecasting accuracy are gathered in the same category, and the classification algorithm is tailored to the forecasting approach [10, 11, 23]. However, to classify a job by considering only the forecasting error, and not all attributes, is difficult.

In the previous research, there have been some collaborative networks for forecasting job cycle time. Most use experts and have not been automated. For example, Chen [17] utilized domain experts for predicting the cycle time of a job using fuzzy multiple linear regression (FMLR), and the forecasts were aggregated using fuzzy intersection (FI) and back propagation network (BPN). Similarly, in Chen and Lin [18], the domain experts established fuzzy back propagation networks (FBPNs) for the same purpose, and a partial consensus among the experts was sought instead. In order to improve the effectiveness and efficiency of forecasting the cycle time of a job in a semiconductor manufacturing factory, a fuzzy collaborative sensor network is developed in this study. Each sensor is a programmed device that can be used to detect the environment, or retrieve data and information from specific equipment and systems. Why is a system that is complex, time consuming, and requires the collaboration of a number of sensors used in this study?

| Method               | System type       | Collaboration method | Forecasting method | Aggregation method |
|----------------------|-------------------|----------------------|--------------------|--------------------|
| Chen [14]            | Standalone        | No                   | FBPNN              | No                 |
| Chen [17]            | Expert network    | FI                   | FMLR               | BPN                |
| Chen and Lin [18]    | Expert network    | PCFI                 | FBPNN              | BPN                |
| The proposed methodology | Distributed sensor network | P2P | FBPNN | RBF |

Table 1: The differences between the proposed methodology and existing methods.

In previous research, fuzzy sensor networks have rarely been applied. Odeberg [25] focused on measuring the optimal distance between fuzzy sensors. Goebel and Agogino [26] proposed a systematic procedure for fuzzy sensor fusion. The fusion algorithm calculated the weighted average of the confidence value of each sensor reading. Deng et al. [27] believed that fuzzy sensor networks can provide flexibility in the processing of symbolic information. Born and Wright [28] established an architecture for guiding robot sensors using a fuzzy sensor fusion network (FSFN). However, despite having so much research, determining how to integrate the observations of different fuzzy sensors is still a major problem.

The concept of the fuzzy collaborative sensor network is illustrated in Figure 1. There are six operational procedures for the fuzzy collaborative sensor network.

1. The fuzzy collaborative sensor network starts from the deployment of a group of sensors.

2. The administrators of these sensors determine the data that is to be collected and incorporated into the sensors’ FBPNN reasoning modules.

2. Fuzzy Collaborative Sensor Network

In previous research, fuzzy sensor networks have rarely been applied. Odeberg [25] focused on measuring the optimal distance between fuzzy sensors. Goebel and Agogino [26] proposed a systematic procedure for fuzzy sensor fusion. The fusion algorithm calculated the weighted average of the confidence value of each sensor reading. Deng et al. [27] believed that fuzzy sensor networks can provide flexibility in the processing of symbolic information. Born and Wright [28] established an architecture for guiding robot sensors using a fuzzy sensor fusion network (FSFN). However, despite having so much research, determining how to integrate the observations of different fuzzy sensors is still a major problem.

The concept of the fuzzy collaborative sensor network is illustrated in Figure 1. There are six operational procedures for the fuzzy collaborative sensor network.

1. The fuzzy collaborative sensor network starts from the deployment of a group of sensors.

2. The administrators of these sensors determine the data that is to be collected and incorporated into the sensors’ FBPNN reasoning modules.
(3) Each sensor detects and analyzes the local conditions based on its own observations.

(4) Each sensor communicates its settings and analysis results to other sensors with the aid of the central control unit. After receiving the settings and analysis results of other sensors, a sensor may be prompted to modify its settings.

(5) A RBF network is employed to aggregate the sensor’s forecasts.

(6) The collaboration process is terminated if the improvement in the aggregate performance becomes negligible. Otherwise, the system automatically returns to step (4).

Before introducing the details of the fuzzy collaborative sensor network, we first must define all required parameters:

(1) \( a_j \): the normalized value of the actual cycle time of job \( j \).

(2) \( \bar{a} \): the FBPN output, which is the normalized fuzzy cycle time forecast of job \( j \).

(3) \( \bar{h}_l \): the output from hidden-layer node \( l \).

(4) \( \bar{w}_{ij} \): the weight of the connection between hidden-layer node \( l \) and the output node.

(5) \( \bar{w}_{il} \): the weight of the connection between input node \( i \) and hidden-layer node \( l \).

(6) \( \bar{\theta}_l \): the threshold for screening out weak signals by hidden-layer node \( l \).

(7) \( \bar{\theta}^o \): the threshold for screening out weak signals by the output node.

In the fuzzy collaborative sensor network, each sensor detects the following job actions in the factory:

(1) the machine status, which is the equipment utilization information,

(2) the job status,

(3) the job queuing information on the processing route or before bottlenecks,

(4) the factory workload information,

(5) the waiting times of some recently completed jobs.

It uses a FBPN to analyze the information (see Figure 2). Although there have been some more advanced artificial neural networks, such as a compositional pattern-producing network, a cascading neural network, and a dynamic neural network, a well-trained FBPN is utilized in this study as it allows an optimized structure to fit very complex relationships.

(1) Inputs: the five types of information detected by the sensor. To facilitate the search for solutions, it is strongly recommended to normalize the inputs to a range narrower than [0 1] [10, 11]:

\[
N(x) = N_L + \frac{x - x_{\min}}{x_{\max} - x_{\min}} \cdot (N_U - N_L),
\]

where \( N(x) \) is the normalized value of the original value \( x \); \( N_L \) and \( N_U \) indicate the lower and upper bounds of the range of the normalized value, respectively. \( x_{\min} \) and \( x_{\max} \) are the minimum and maximum of \( x \), respectively. The formula can be written as

\[
x = N(x) - N_L \cdot \frac{x_{\max} - x_{\min}}{N_U - N_L} + x_{\min}
\]

if the unnormalized value is to be obtained instead. In addition, some inputs may be dependent on each other, which makes the unnormalized value more difficult to discover the relationship between \( x_{\min} \) and \( x_{\max} \). To solve this problem, principal component analysis (PCA) is considered to be helpful [29].

(2) The FBPN has only one hidden layer. Two or more hidden layers slow down the convergence speed and may not lead to any better solution. The number of nodes in the hidden layer is chosen (from 1 to 10) after trying each of them.
In order to determine the parameter values, after pre-classification a portion of the adopted examples is fed into the FBPN as a "training example." Two phases are involved at the training stage. First, in the forward phase, inputs are multiplied with weights, summed, and transferred to the hidden layer. Second, activated signals are outputted from the hidden layer as

\[
\bar{h}_j = (h_{i1}, h_{i2}, h_{i3}) = \frac{1}{1 + e^{-\eta_j}} = \left( \frac{1}{1 + e^{-\eta_{i1}}}, \frac{1}{1 + e^{-\eta_{i2}}}, \frac{1}{1 + e^{-\eta_{i3}}} \right),
\]

where

\[
n_i^h = (n_i^{h1}, n_i^{h2}, n_i^{h3}) = \bar{h}^o (-) \bar{h}^o = \left( I_{i1}^h - \theta_{i1}^h, I_{i2}^h - \theta_{i2}^h, I_{i3}^h - \theta_{i3}^h \right),
\]

\[
\bar{I}_i^h = \left( I_{i1}^h, I_{i2}^h, I_{i3}^h \right) = \sum_{all k} \bar{w}_k^h \times x_k
\]

\[
= \left( \sum_{all k} \min \left( w_{k1}^h, x_k, w_{k2}^h, x_k \right), \sum_{all k} \max \left( w_{k1}^h, x_k, w_{k3}^h, x_k \right) \right)
\]

and (−) and (×) denote fuzzy subtraction and multiplication, respectively; \( \bar{h}_j \)'s are also transferred to the output layer with the same procedure. Finally, the output of the FBPN is generated as follows:

\[
\bar{o} = (o_1, o_2, o_3) = \frac{1}{1 + e^{-\bar{o}}} = \left( \frac{1}{1 + e^{-\bar{o}_{i1}}}, \frac{1}{1 + e^{-\bar{o}_{i2}}}, \frac{1}{1 + e^{-\bar{o}_{i3}}} \right),
\]

where

\[
\bar{a}^o = (a_i^o, a_{i2}^o, a_{i3}^o) = \bar{I}^o (-) \bar{I}^o = \left( I_{i1}^o - \theta_{i1}^o, I_{i2}^o - \theta_{i2}^o, I_{i3}^o - \theta_{i3}^o \right),
\]

\[
\bar{I}^o = \left( I_{i1}^o, I_{i2}^o, I_{i3}^o \right) = \sum_{all l} \bar{w}_l^o \times (\bar{h}_l)
\]

\[
= \left( \sum_{all l} \min \left( w_{l1}^o h_{i1}, w_{l2}^o h_{i2}, w_{l3}^o h_{i3} \right), \sum_{all l} \max \left( w_{l1}^o h_{i1}, w_{l3}^o h_{i3} \right) \right).
\]

Subsequently in the backward phase, the training of the FBPN is decomposed into three subtasks: determining the center value, upper, and lower bounds of the parameters [30]. First, to determine the center value of each fuzzy parameter (e.g., \( \theta_{i1}^h, \theta_{i2}^h, \theta_{i2}^o \)), the FBPN is treated as a crisp one. Some algorithms are applicable for training a crisp feed-forward neural network, such as the gradient descent (GD) algorithms, the Levenberg-Marquardt algorithm, the conjugate gradient algorithms, the resilient back-propagation algorithm, and the BFGS quasi-Newton back-propagation algorithm.

Subsequently, the following goal programming (GP) problem is solved to determine the upper bound of each fuzzy parameter (e.g., \( u_{k1}^h, \theta_{i1}^h, u_{i2}^o, \) and \( \theta_{i3}^o \)), so that the actual value will be less than the upper bound of the network output:

\[
\text{Min} \sum_{all j} \pi_j (g),
\]

subject to \( \ln \left( \frac{1}{o_j} - 1 \right) = \theta_j^o - \sum_{all l} w_{lj}^o h_{lj}, \)

\[
\sum_{all l} w_{lj}^o h_{lj} - \theta_j^o \leq - \ln \left( \frac{1}{\pi_j} (g) - 1 \right),
\]

\[
\sum_{all l} w_{lj}^o h_{lj} - \theta_j^o \leq \ln \left( \frac{1}{\pi_j} (g) - 1 \right),
\]

\[
\sum_{all k} w_{kl}^h x_k - \theta_{i}^h \leq - \ln \left( \frac{1}{h_{kj}} - 1 \right),
\]

\[
\sum_{all k} w_{kl}^h x_k - \theta_{i}^h \leq \ln \left( \frac{1}{h_{kj}} - 1 \right),
\]

\[
k = 1 \sim K; \ l = 1 \sim L.
\]

In a similar way, another GP problem is solved to determine the lower bound of each fuzzy parameter (e.g., \( u_{k1}^h, \theta_{i1}^h, u_{i2}^o, \) and \( \theta_{i3}^o \)), so that the actual value will be greater than the lower bound of the network output:

\[
\text{Min} \sum_{all j} \pi_j (g),
\]

subject to \( \ln \left( \frac{1}{o_j} - 1 \right) = \theta_j^o - \sum_{all l} w_{lj}^o h_{lj}, \)

\[
\sum_{all l} w_{lj}^o h_{lj} - \theta_j^o \leq - \ln \left( \frac{1}{\pi_j} (g) - 1 \right),
\]

\[
\sum_{all l} w_{lj}^o h_{lj} - \theta_j^o \geq \ln \left( \frac{1}{\pi_j} (g) - 1 \right),
\]

\[
k = 1 \sim K; \ l = 1 \sim L.
\]
The forecasts by all sensors can be communicated to each other, so that they can modify their settings and generate more accurate forecasts if more viewpoints are taken into account. A collaboration mechanism is established to this end. The view by a sensor is indicated with VS for the gravity method: 

\[
\text{Sensor } S_g, 1 \leq g \leq G, \text{ provides input data } \overline{d}_j \text{ for } N \text{ jobs, where } 1 \leq j \leq N. \text{ In case of computing the network output, the view vector } \text{VS}_g \text{ is public. }
\]

\[
\text{Output. } S_g, 1 \leq g \leq G, \text{ learns } (D(\overline{d}_j) - a_j)/a_j \text{ without anything else, where } D(\overline{d}_j) \text{ is computed using the center-of-gravity method: }
\]

\[
d(\overline{d}_j) = \frac{a_{j1} + a_{j2} + a_{j3}}{3}.
\]

After receiving this information, if it reveals that the forecasting performance of a sensor is very prominent, the others may change their settings, so that the settings will be more consistent. In addition, if a sensor obtains unsatisfactory forecasting results, then it will automatically cross-reference the results with other sensors.

Subsequently, a RBF is used to aggregate the forecasts. In past studies, Aliustaoglu et al. [31] fused the monitoring results of different sensors with a fuzzy inference system. The RBF network has three layers: the input, hidden (middle), and output layers. Inputs to the RBF are the three corners of the fuzzy forecasts by all sensors. Each input is assigned to a node in the input layer and passed directly to the hidden layer without being weighted. The transfer function used for the hidden layer is a Gaussian transfer function, while the output layer uses the linear transfer function. For determining the parameter values, k-means (KM) is first used to discover the centers of the RBF units. Subsequently, the nearest-neighbor method is used to determine their widths. The weights of the connections can be obtained by linear regression. Compared with BPN, RBF has a better chance of escaping the local optimum.

3. Experiment

To illustrate the application of the fuzzy collaborative sensor network, an example containing the data of 500 jobs was used. There are more than ten products manufactured in the semiconductor factory. The semiconductor factory has a monthly capacity of 30,000 wafers and is expected to be fully utilized. Jobs are released into the semiconductor factory at a rate of one per 0.85 hours, namely, the mean release rate \( \lambda = 1/0.85 = 1.18 \) jobs per hour. Three types of priorities (normal, hot, and super hot) are assigned to jobs, depending on the customer requests. The job percentages with these priorities released into the factory are restricted to approximately 60%, 30%, and 10%, respectively. The manufacture of each product has hundreds of steps, as well as “re-entry” to certain machines that can cause significant bottlenecks. Reentry is the singular production characteristic of the semiconductor industry. Providing an accurate due date for the product with such a complicated routing also shows the difficulty inherent in production planning and scheduling staff. A total of more than 400 machines (including alternative machines) are provided to process wafers, in either single or multiple batch operation. Sensors were deployed particularly at bottleneck locations.

Five existing approaches, statistical analysis (i.e., MLR), CBR, BPN, FCM-BPN [20], and collaborative FMLR [17] were also applied to the case. Performance measures including the mean absolute error (MAE), the mean absolute percentage error (MAPE), and the minimal root mean squared error (RMSE) were evaluated. The six approaches recorded and compared the manufacturing performance and are summarized in Table 2. The BPN approach, with one hidden layer with 1–10 nodes, depended on the results of a preliminary analysis for establishing the best configuration. The optimal value of parameter \( k \) in the CBR approach was equal to the value that minimized the RMSE. In FCM-BPN, jobs were classified into some categories using FCM before forecasting their cycle times. The number of categories was determined by the S test. In collaborative FMLR, some domain experts gathered and predicted the cycle time of a job using FMLR. Then, the forecasts by these experts were aggregated using FI-BPN to derive the crisp, representative value.

According to the experiment’s results, we have the following:

(1) From the tabulated results it can be seen that the difference between the forecasting results and the collected data is very small. The magnitude of the errors ranged from 0.00% to 2.65% with an average of only 1%. In fitting the collected data, the fuzzy collaborative sensor network achieved a very good performance.
Table 2: Comparisons of the performances of various approaches.

| Performance measure | Statistical analysis | CBR | BPN | FCM-BPN | Collaborative FMLR | The proposed methodology |
|---------------------|----------------------|-----|-----|---------|-------------------|--------------------------|
| RMSE                | 77                   | 74  | 53  | 36      | 31                | 19                       |
| MAE                 | 63                   | 62  | 43  | 21      | 23                | 11                       |
| MAPE                | 5%                   | 5%  | 4%  | 2%      | 2%                | 1%                       |

(2) The estimation accuracy in terms of three measures, RMSE, MAE, and MAPE, has been assessed for each method. The estimation accuracy of MFLR was clearly the most inaccurate, which revealed the non-linear nature of the problem. Nonlinear approaches such as BPN, FCM-BPN, and this study’s proposed methodology all achieved satisfactory performances.

(3) In particular, this study’s proposed methodology was superior to the five existing methods, by improving MAPE up to 33%.

(4) Compared with the collaborative forecasting method that uses real experts, collaborative FMLR and the proposed fuzzy collaborative sensor network automate the necessary actions and are therefore more efficient. Within the same time, it is more likely to find a better cycle time forecast using the proposed fuzzy collaborative sensor network.

(5) With the increase in the number of sensors, the collaboration time significantly increased (see Figure 3). As a result, if many sensors are involved, the computation becomes very complicated. For this reason, the size of the sensor coalition needs to be restricted. If a distributed P2P communication architecture is used, then the collaboration time will be much longer.

4. Conclusions and Directions for Future Research

The cycle time of a specific job has the greatest potential impact for a factory. Data analysis and forecasting in this area are extremely important. There is increasing evidence that demonstrates the widespread and long-term trend toward leaner production. Job cycle time forecasting is considered to be one of the most important tasks in determining how to make a manufacturing facility more efficient.

Empirical evidence reveals that collaborative intelligence has tremendous potential in the job cycle time forecasting application. On the other hand, network based sensing has become an important field of research, and new applications of remote sensing are expected to appear. Both points motivated us to develop a fuzzy collaborative sensor network to enhance the performance of the job cycle time forecasting. In the fuzzy collaborative sensor network, a group of sensors detect the local conditions and use FBPNs to analyze the received information. Each sensor communicates its setting and analysis results to other sensors with the aid of the central control unit under a centralized P2P architecture. The forecasts by all sensors are then aggregated by a RBF network.

After applying the fuzzy collaborative sensor network to an illustrative case, the following experimental results were obtained.

(1) Via the sensors’ automatic collaboration, the aggregate forecasting performance was seen to improve considerably. The accuracy of forecasting the job cycle time, measured in terms of MAPE, improved by up to 33%.

(2) The collaboration time associated with the number of sensors was also determined, which can be referenced to determining the suitable number of sensors. For example, with tens of sensors, the collaboration time was about five minutes, which was still acceptable.

(3) It is therefore possible to accurately forecast the job cycle time using a collection of sensors governed by a centralized collaboration mechanism.

This study’s research contribution includes the following.

(1) In the literature, fuzzy sensor networks have not been applied to cycle time forecasting. The treatment taken in this study is a viable strategy for similar purposes under a distributed decision-making environment.

(2) The centralized client-server architecture is an efficient framework for facilitating collaboration among fuzzy sensors.

If necessary, future studies may develop more sophisticated collaboration mechanisms.

Acknowledgment

This study is partially supported by the National Science Council of Taiwan.
References

[1] Q. Wang, T. Zhang, and S. Pettersson, "An effort to understand the optimal routing performance in wireless sensor network," in Proceedings of the 22nd International Conference on Advanced Information Networking and Applications (AINA '08), pp. 279–286, March 2008.

[2] Z. Y. Yan, B. Y. Zheng, and Z. W. Lin, "Research on opportunistic cooperation transmission and its performance in wireless sensor network," Journal of Electronics and Information Technology, vol. 31, no. 1, pp. 215–218, 2009.

[3] G. H. Sang, B. M. Young, J. P. Sang, and W. K. Whan, "Wireless sensor network testbed for real-time sensor monitoring," in Proceedings of the 3rd International Conference on Sensor Technologies and Applications (SENSORCOMM '09), pp. 486–489, June 2009.

[4] J. Yick, B. Mukherjee, and D. Ghosal, "Wireless sensor network survey," Computer Networks, vol. 52, no. 12, pp. 2292–2330, 2008.

[5] Y. Uchimura, T. Nasu, and M. Takahashi, "Time synchronized wireless sensor network for vibration measurement," in Proceedings of the Society of Instrument and Control Engineers Annual Conference (SICE '07), pp. 2940–2945, September 2007.

[6] T. Endo, A. Banno, and Y. Tamura, "Research into sensor networks and Web APIs—urban navigation systems utilizing sensor network data," in Proceedings of the 5th International Conference on Networked Sensing Systems (INSS '08), pp. 166–169, June 2008.

[7] P. A. Morreale, "Wireless sensor network applications in urban telehealth," in Proceedings of the 21st International Conference on Advanced Information Networking and Applications Workshops/Symposia (AINAW '07), pp. 810–814, May 2007.

[8] T. Chen, "A fuzzy-neural and multiple-bucket approach for estimating lot cycle time in a wafer fab with dynamic product mix," Computers & Industrial Engineering, vol. 55, pp. 423–438, 2008.

[9] T. Chen, "A SOM-FBPN-ensemble approach with error feedback to adjust classification for wafer-lot completion time prediction," International Journal of Advanced Manufacturing Technology, vol. 37, no. 7–8, pp. 782–792, 2008.

[10] T. Chen, H. C. Wu, and Y. C. Wang, "Fuzzy-neural approaches with example post-classification for estimating job cycle time in a wafer fab," Applied Soft Computing Journal, vol. 9, no. 4, pp. 1223–1231, 2009.

[11] T. Chen, Y. C. Wang, and H. R. Tsai, "Lot cycle time prediction in a ramping-up semiconductor manufacturing factory with a SOM-FBPN-ensemble approach with multiple buckets and partial normalization," International Journal of Advanced Manufacturing Technology, vol. 42, no. 11-12, pp. 1206–1216, 2009.

[12] T. Chen, "A fuzzy back propagation network for output time prediction in a wafer fab," Applied Soft Computing Journal, vol. 2, no. 3, pp. 211–222, 2003.

[13] T. Chen, "A hybrid SOM-BPN approach to lot output time prediction in a wafer fab," Neural Processing Letters, vol. 24, no. 3, pp. 271–288, 2006.

[14] T. Chen, "A look-ahead fuzzy back propagation network for lot output time series prediction in a wafer fab," in Neural Information Processing, vol. 4234 of Lecture Notes in Computer Science, pp. 974–982, Springer, New York, NY, USA, 2006.

[15] T. Chen, "A hybrid look-ahead SOM-FBPN and FIR system for wafer-lot-output time prediction and achievability evaluation," International Journal of Advanced Manufacturing Technology, vol. 35, no. 5-6, pp. 575–586, 2007.
Submit your manuscripts at
http://www.hindawi.com