Research Article

Internal Due Date Assignment in a Wafer Fabrication Factory by an Effective Fuzzy-Neural Approach

Toly Chen

Department of Industrial Engineering and Systems Management, Feng Chia University, 100 Wenhwa Road, Seatwen, Taichung City 407, Taiwan

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

Owing to the complexity of the wafer fabrication, the due date assignment of each job presents a challenging problem to the production planning and scheduling people. To tackle this problem, an effective fuzzy-neural approach is proposed in this study to improve the performance of internal due date assignment in a wafer fabrication factory. Some innovative treatments are taken in the proposed methodology. First, principal component analysis (PCA) is applied to construct a series of linear combinations of the original variables to form a new variable, so that these new variables are unrelated to each other as much as possible, and the relationship among them can be reflected in a better way. In addition, the simultaneous application of PCA, fuzzy c-means (FCM), and back propagation network (BPN) further improved the estimation accuracy. Subsequently, the iterative upper bound reduction (IUBR) approach is proposed to determine the allowance that will be added to the estimated job cycle time. An applied case that uses data collected from a wafer fabrication factory illustrates this effective fuzzy-neural approach.

1. Introduction

Internal due date assignment is to quote an attractive but attainable due date for an arriving customer order. However, the completion time of an order is highly uncertain. It is therefore difficult to accurately forecast the completion time. For this reason, an allowance has to be added to the estimated completion time to reduce the risk [1].

Wafer fabrication is the most technologically complex step in semiconductor manufacturing, which exacerbates the difficulties of internal due date assignment [2]. In theory, this problem is NP-hard. That is why wafer fabrication is investigated in this study. Internal due date assignment in a wafer fabrication factory is difficult because of the following reasons.

(1) Shop floor control in a wafer fabrication factory is a nontrivial task owing to the complexity of wafer fabrication. Some wafer fabrication processes are repeated processes. Thus, wafers need to visit a machine multiple times. An average job cycle time is several months with hundreds of hours of standard deviation. Many studies have shown that accurately predicting the cycle/completion times for such large systems is very difficult [1, 3, 4].

(2) In addition, the completion time predicted using existing approaches is generally unbiased. This means that if the internal due date is set to be equal to the mean of the estimated completion time, then the probability of on-time delivery is only about 50% on average. To reduce the risk, an allowance or fudge factor has to be added to the estimated completion time [5]. The due date allowance factor is determined on the basis of the feedback information about the factory status at the time a job arrives at the factory.

(3) Due date assignment, release control, and buffer control affect each other. Make-to-order wafer fabrication factories are confronted with both due date quotation and production scheduling problems at the same time [6]. If due date assignment and factory scheduling are processed separately by two systems, the overall performance is unlikely to be satisfactory because the two tasks are actually interrelated. Therefore, the interaction between due date assignment methods

and scheduling rules in a wafer fabrication factory needs to be investigated.

To tackle these problems, some treatments have been carried out in the literature. First, various research works have been dedicated to estimate the cycle time using hybrid approaches. For example, Gupta and Sivakumar [7] presented look-ahead batch scheduling for the real-time control of due date objectives. Chen [8] proposed the look-ahead self-organization map (SOM)-fuzzy back propagation network (FBPN) approach for this purpose. A set of fuzzy inference rules were also developed to evaluate the achievability of a cycle time forecast. Subsequently, Chen et al. [1] added a selective allowance to the cycle time estimated using the look-ahead SOM-FBPN approach to determine the internal due date. Further, Chen [9] showed that the combination of SOM and FBPN could be improved by a minor adjustment of the classification results with the estimation error. Chen et al. [10, 11] proposed a postclassification fuzzy-neural approach in which a job was not preclassified but rather postclassified after estimating the cycle time. Experimental results showed that the postclassification approach was better than the preclassification approaches in some cases. To balance the influence of the preclassification results with that of the postclassifying results, Chen [12] proposed a bidirectional classifying approach, in which jobs are not only preclassified but also postclassified. Ankenman et al. [13] proposed a meta-modeling approach, which integrates discrete-event simulation, adaptive statistical methods, and analytical queueing analysis to quantify the cycle time-throughput relationship. Chien et al. [4] used nonlinear regression equations and then related the forecasting error to some factory conditions and job attributes with a back propagation network (BPN) to improve the forecasting accuracy. The major disadvantage of statistical analysis is the lack of forecasting accuracy [8].

Second, in traditional due date setting rules, the fudge factor is usually equal to a multiple of the standard deviation of the predicted cycle time [14]. Recently, Chen et al. [1] proposed a selective allowance policy in which the allowance was only assigned to some preselected jobs. In this way, the sum of the allowances added to all jobs was controlled. However, even though the probability of on-time delivery in Chen et al.'s study was only 77% for the testing data, showing that improving the probability of on-time delivery while controlling the fudge factor is a real challenge. In addition, the allowances that were assigned to the chosen jobs in this study were equal, leaving room for improvement. Another way of taking this issue into account is to construct a confidence interval containing the actual completion time [3]. The upper confidence limit sets the internal due date. However, the probability of a job delivered on time is only 99.7% for the testing data, under the assumption that residuals follow a normal distribution. From another point of view, Chen and Wang [15] incorporated the fuzzy c-means (FCM)-BPN approach with a nonlinear programming (NLP) model to construct the inclusion interval of the predicted completion time. Similarly, the upper inclusion limit sets the internal due date. An inclusion interval is narrower than a confidence interval, and the probability of a job delivered on time is 100%,

at least for the training data. Chen and Lin [16] modified this approach by gathering a group of experts in related fields to set the due date in a collaborative way. Fuzzy intersection is applied to combine the due dates into a representative value.

The existing approaches have the following problems.

(1) Some factors used to forecast the job cycle time are dependent on each other, which may cause problems in classifying jobs and in fitting the relationship between the job cycle time and these factors.

(2) In Chen and Wang [15] and Chen and Lin [16], NLP models are solved to determine the upper bound of the job cycle time. However, the NLP models involve complicated constraints and therefore are difficult to solve. The NLP models will become too huge if many jobs are to be considered.

To tackle these problems, an effective fuzzy-neural approach is proposed in this study to improve the performance of internal due date assignment in a wafer fabrication factory. The literature provides probabilistic (stochastic) and fuzzy methods that can consider the uncertainty or randomness in the completion time. However, the longest average cycle time exceeds three months with a variation of more than 300 hours. Fitting the cycle time within a future month with a distribution function is not easy, implying that a stochastic approach might not be applicable. That is why a fuzzy approach is proposed in this study.

The effective fuzzy-neural approach has the following innovative characteristics.

(1) Variable replacement using principal component analysis (PCA): PCA uses orthogonal transformation to convert a set of observations of possibly correlated variables into a set of values of linearly uncorrelated variables to reflect information in a better way.

(2) Updating the upper bound of the job cycle time using the iterative upper bound reduction (IUBR) approach: the IUBR approach is proposed to determine the upper bound of the completion time forecast. A tight upper bound means that the allowance assigned to a job is minimized.

Some recent works in this field are relevant. The differences between the proposed methodology and these methods are summarized in Table 1.

The remainder of this paper is organized as follows. Section 2 introduces the proposed methodology which is composed of four steps. A practical example is used to validate the effectiveness of the proposed methodology. The performance of the proposed methodology is evaluated and compared with those of some existing approaches. Finally, the concluding remarks and some directions for future research are given in Section 4.

2. Methodology

The operating procedure of the effective fuzzy-neural approach consists of several steps that will be described in the following sections.

Table 1: The differences between this study and some recent references.

Method	Data preprocessing method	Forecasting method	Upper bound	Optimization method	Computation complexity
Chen et al. (2008) [1]	SOM	FBPN	Yes	Simulation + fuzzy rules	High
Chien et al. (2011) [4]	No	Nonlinear regression	No	BPN	Low
Chen and Lin (2011) [16]	FCM	BPN	Yes	NLP	Very high
Chen and Wang (2013) [23]	PCA + FCM	BPN	No	An iterative process to reduce outliers	Low
The proposed methodology	PCA + FCM	BPN	Yes	IUBR	Low

NLP: nonlinear programming.
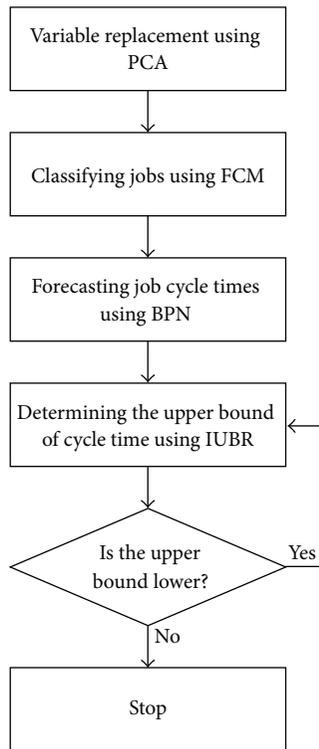
Figure 1: The flowchart of the proposed methodology.

Figure 2: The PCA process.

Step 1. Forming new variables by constructing linear combinations of the original variables using PCA.

Step 2. Classifying jobs using fuzzy c-means (FCM).

Step 3. Forecasting the cycle times of jobs in each category using a BPN.

Step 4. Determining the upper bound of the cycle time using the IUBR approach.

A flow chart of the proposed methodology is shown in Figure 1.

2.1. Step 1: Forming New Variables Using PCA. First, PCA is used to replace the inputs to the BPN. PCA was invented by Pearson [18] as an analogue of the principal axes theorem in mechanics; it was later independently developed by Hotelling [19]. In the literature, there are more advanced applications of PCA. For example, Jaiswal et al. [20] used a hybrid of PCA and partial least squares for face recognition. In Mohtasham et al. [21], linear and exponential weighted PCA techniques based on spectral similarity were employed to predict the dye concentration in coloured fabrics. The operating procedure of PCA consists of several steps that are illustrated in Figure 2.
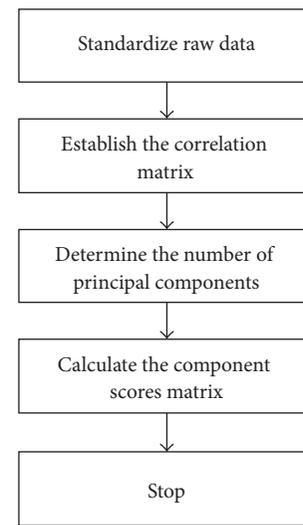
The references on the combination of PCA, FCM, and BPN are still very limited [17, 22, 23].

2.2. Step 2: Classifying Jobs Using FCM. After employing PCA, examples are then classified using FCM. FCM is one of the most popular fuzzy clustering techniques because it is efficient, straightforward, and easy to implement. However, FCM is sensitive to initialization and is easily trapped in local optima.

The objective function of FCM is to minimize the weighted sum of squared distances such that the jobs in a category will be similar (or related) to one another and
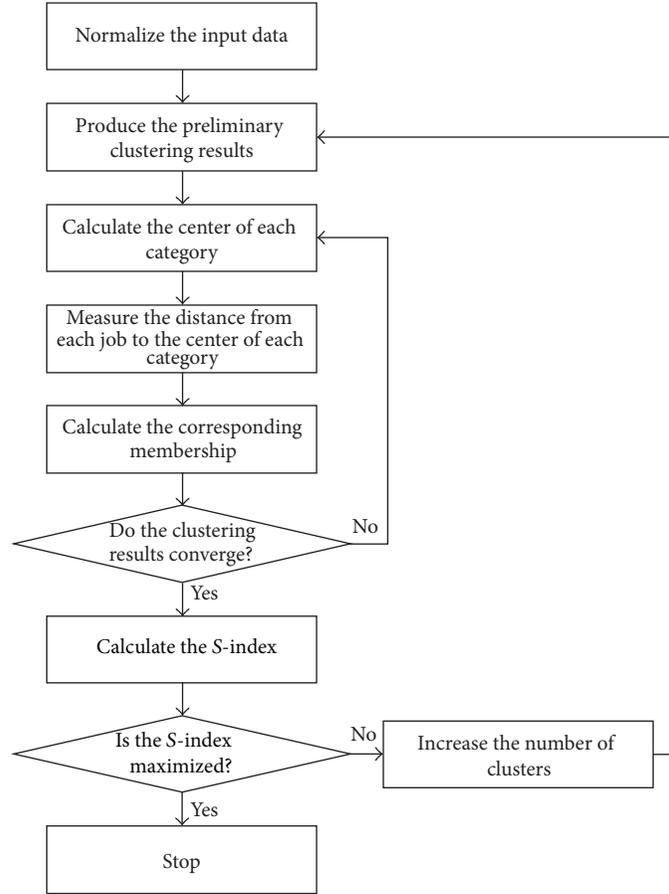
FIGURE 3: The FCM procedure.

different from (or unrelated to) the jobs in other categories. In FCM, the Euclidean distance between two jobs is measured:

$$e_{j(k)} = \sqrt{\sum_{q=1}^{p} \left(z_{jp} - \overline{z}_{(k)p}\right)^2}, \quad j = 1 \sim n, \ k = 1 \sim K, \quad (1)$$

where

$$\overline{z}_{(k)} = \left\{\overline{z}_{(k)q}\right\}, \quad k = 1 \sim K,$$

$$\overline{z}_{(k)q} = \frac{\sum_{j=1}^{n} \mu_{j(k)}^{m} z_{jq}}{\sum_{j=1}^{n} \mu_{j(k)}^{m}}, \quad k = 1 \sim K, \ q = 1 \sim p. \quad (2)$$

The weight of a job is a function of its membership:

$$\mu_{j(k)} = \frac{1}{\sum_{g=1}^{K} \left(e_{j(k)}/e_{j(g)}\right)^{2/(m-1)}}, \quad j = 1 \sim n, \ k = 1 \sim K. \quad (3)$$

However, FCM requires prior knowledge about the number of clusters in the data, which may not be known for new data. Then, fuzzy clustering is carried out through an iterative optimization of the objective function (see Figure 3). The clustering process stops when the maximum number of iterations is reached or the improvement in the objective function

becomes negligible with more iterations. In addition, the S-index proposed by Xie and Beni [24] is used to give the ideal number of categories automatically:

$$S = \frac{J_m}{n \times e_{\min}^2}, \quad (4)$$

where

$$J_m = \sum_{k=1}^{K} \sum_{j=1}^{n} \mu_{j(k)}^{m} e_{j(k)}^2,$$

$$e_{\min}^2 = \min_{k1 \neq k2} \left(\sum_{\text{all } p} \left(\overline{x}_{(k1)p} - \overline{x}_{(k2)p}\right)^2\right). \quad (5)$$

Chen and Wang [23] found the empirical relationship between the S-index and the estimation performance.

2.3. Step 3: Forecasting the Cycle Times of Jobs in Each Category with a BPN. Subsequently, the jobs/examples of a category are learned with the same BPN. BPN is a popular tool with applications in a variety of fields. Nevertheless, different problems may require different parameter settings for a given network architecture. In the literature, researchers have used BPNs for estimating cycle times and assigning due dates. The configuration of the BPN is established as follows.

(1) Inputs: the new factors determined by PCA associated with the jth example/job. These factors have to be partially normalized so that their values fall within $[0.1, 0.9]$ [10, 11].

(2) Single hidden layer: generally one or two hidden layers are more beneficial for the convergence property of the BPN [25].

(3) The number of neurons in the hidden layer: 1 to $2K$. An increase in the number of hidden-layer nodes lessens the output errors for the training examples but increases the errors for novel examples. Such a phenomenon is often called "overfitting". There exist many different approaches such as the pruning algorithm, the polynomial time algorithm, the canonical decomposition technique, and the network information criterion for finding the optimal configuration of a BPN [26]. In addition, there has been some research considering the relation among the complexity of a BPN, the performance for the training data, and the number of examples, for example, using Akaike's information criterion (AIC) [27] or the minimum description length (MDL) [28].

(4) Activation/transformation function: there are a number of common activation/transformation functions, such as identity function, binary step function, bipolar step function, sigmoid functions (binary sigmoid function and bipolar sigmoid function), and ramp function. In the proposed methodology, the binary sigmoid function is used:

$$f(x) = \frac{1}{1 + e^{-x}}. \tag{6}$$

Therefore, the output ranges between 0 and 1.

(5) Output (o_j): the (normalized) cycle time forecast of the example. o_j is compared with the normalized cycle time $N(\mathrm{CT}_j)$, for which root mean squared error (RMSE) is calculated:

$$\mathrm{RMSE} = \sqrt{\frac{\sum_{j=1}^{n} \left(o_j - N\left(\mathrm{CT}_j\right)\right)^2}{n}}. \tag{7}$$

o_j is derived by transforming the signal transferred to the output layer:

$$o_j = \frac{1}{1 + e^{-n_j^o}}, \tag{8}$$

where

$$n_j^o = I_j^o - \theta^o,$$

$$I_j^o = \sum_{l=1}^{L} w_l^o h_{jl}. \tag{9}$$

Similarly, h_{jl} is derived by transforming the signal transferred to the hidden layer:

$$h_{jl} = \frac{1}{1 + e^{-n_{jl}^h}}, \tag{10}$$

where

$$n_{jl}^h = I_{jl}^h - \theta_l^h,$$

$$I_{jl}^h = \sum_{q=1}^{p} w_{ql}^h z_{jq}. \tag{11}$$

Some algorithms are applicable for training a BPN in the backward phase, such as the gradient descent algorithms, the conjugate gradient algorithms, and the Levenberg-Marquardt algorithm. In this study, the Levenberg-Marquardt algorithm is applied. The Levenberg-Marquardt is the most widely used optimization algorithm. It outperforms simple gradient descent and other conjugate gradient methods in a wide variety of problems. The Levenberg-Marquardt algorithm uses approximation and updates the network parameters in a Newton-like way, as described below.

The network parameters are placed in vector $\boldsymbol{\beta}$. The network output o_j can be represented with $f(\mathbf{x_j}, \boldsymbol{\beta})$. The objective function of the BPN is to minimize RMSE or equivalently the sum of squared error (SSE):

$$\mathrm{SSE}(\boldsymbol{\beta}) = \sum_{j=1}^{n} \left(N\left(\mathrm{CT}_j\right) - f\left(\mathbf{x_j}, \boldsymbol{\beta}\right)\right)^2. \tag{12}$$

The Levenberg-Marquardt algorithm is an iterative procedure. In the beginning, the user should specify the initial values of the network parameters. In each step, the parameter vector is replaced by a new estimate, and the network output by its linearization. When the network converges, the gradient of the objective function will be zero. It should be noted that while the Levenberg-Marquardt method is in no way optimal but is just a heuristic, it works extremely well in practice.

2.4. Step 4: Establishing the Upper Bound for the Job Cycle Time Using the IUBR Approach. In order to apply the BPN obtained at the previous step to determine the internal due date of a job, the parameter values in the BPN must be adjusted. To this end, in Chen and Wang [15] and Chen and Lin [16], the NLP model is constructed to adjust the connection weights and thresholds in the BPN, which is not easy to solve. In the IUBR approach, only the threshold of the output node will be adjusted in an iterative way. This way is much simpler and can also achieve satisfactory results.

Substituting (9) into (8),

$$o_j = \frac{1}{1 + e^{-n_j^o}} = \frac{1}{1 + e^{-(I_j^o - \theta^o)}} = \frac{1}{1 + e^{\theta^o - I_j^o}}. \tag{13}$$

Therefore,

$$\ln\left(\frac{1}{o_j} - 1\right) = \theta^o - I_j^o. \tag{14}$$

So

$$I_j^o = \theta^o - \ln\left(\frac{1}{o_j} - 1\right). \tag{15}$$

Assume that the adjustment made to the threshold of the output node is indicated as $\Delta\theta^o$. After adjustment, the output from the new BPN, o'_j, determines the upper bound of the cycle time:

$$o'_j = \frac{1}{1 + e^{-n_j^{o'}}}, \tag{16}$$

where

$$n_j^{o'} = I_j^o - \theta^{o'} = I_j^o - (\theta^o + \Delta\theta^o). \tag{17}$$

Substituting (17) into (16),

$$o'_j = \frac{1}{1 + e^{-(I_j^o - \theta^o - \Delta\theta^o)}}. \tag{18}$$

Substituting (15) into (18),

$$o'_j = \frac{1}{1 + e^{-(\theta^o - \ln((1/o_j) - 1) - \theta^o - \Delta\theta^o)}} = \frac{1}{1 + e^{\ln((1/o_j) - 1) + \Delta\theta^o}}$$
$$= \frac{1}{1 + e^{\Delta\theta^o}\left((1/o_j) - 1\right)}. \tag{19}$$

Obviously, the maximum of $\Delta\theta^o$ establishes the lowest upper bound.

Since o'_j is the upper bound of the cycle time, $o'_j \geq N(\mathrm{CT}_j)$,

$$\frac{1}{1 + e^{\ln((1/o_j) - 1) + \Delta\theta^o}} \geq N\left(\mathrm{CT}_j\right), \tag{20}$$

$$\Delta\theta^o \leq \ln\left(\frac{1}{N\left(\mathrm{CT}_j\right)} - 1\right) - \ln\left(\frac{1}{o_j} - 1\right). \tag{21}$$

Equation (21) holds for all jobs, so

$$\Delta\theta^o \leq \min_j\left(\ln\left(\frac{1}{N\left(\mathrm{CT}_j\right)} - 1\right) - \ln\left(\frac{1}{o_j} - 1\right)\right). \tag{22}$$

According to (19), the optimal value of $\Delta\theta^o$ should be set to the maximum possible value:

$$\Delta\theta^{o*} = \min_j\left(\ln\left(\frac{1}{N\left(\mathrm{CT}_j\right)} - 1\right) - \ln\left(\frac{1}{o_j} - 1\right)\right). \tag{23}$$

Then the optimization results of the BPN are sensitive to the initial conditions and may be different for each iteration. Assume that the optimal value of o'_j in the tth iteration is indicated with $o'_j(t)$. After some iterations,

$$o'_j \text{ (all iterations)} = \min_t o'_j(t). \tag{24}$$

In this way, the upper bound of the cycle time is decreased gradually (see Figure 4). Another merit of the IUBR approach is that it does not rely on the parameters of the BPN.
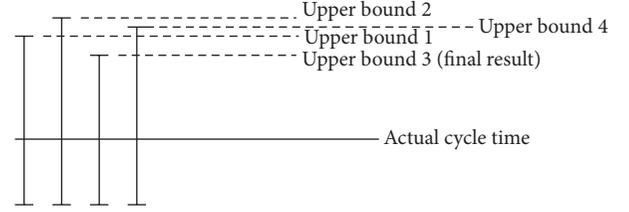
Figure 4: The upper bound is reduced in an iterative manner.

2.5. *Ensemble Learning.* Ensemble learning is based on the notion of perturbing and combining. An ensemble consists of a collection of ANNs and combines their predictions to obtain a final prediction. In FCM, a job can be classified into several categories to different degrees. In theory, the BPNs of all categories can be applied to predict the cycle time of a job. The forecasts obtained by using the BPNs may not be the same and need to be aggregated. To this end, some treatments have been carried out in the literature.

(1) Linear aggregation [29]:

$$\mathrm{CTE}_j = \frac{\sum_{k=1}^{K}\left(\mu_{j(k)} \cdot \mathrm{CTE}_j(k)\right)}{\sum_{k=1}^{K} \mu_{j(k)}} = \sum_{k=1}^{K}\left(\mu_{j(k)} \cdot \mathrm{CTE}_j(k)\right), \tag{25}$$

where $\sum_{k=1}^{K} \mu_{j(k)} = 1$. $\mathrm{CTE}_j(k)$ is the cycle time of job j estimated by the BPN of category k.

(2) BPN aggregation [29]: the membership and cycle time forecast of a job are fed into another BPN to be aggregated. Consider

$$\mathrm{CTE}_j = \mathrm{BPN}\left(\mu_{j(1)}, \mathrm{CTE}_j(1), \ldots, \mu_{j(K)}, \mathrm{CTE}_j(K)\right). \tag{26}$$

(3) Generalized average method [30]: in FCM, the error is proportional to the distance to the center. For this reason, a natural way to aggregate the forecasts is

$$\mathrm{CTE}_j = \frac{\sum_{k=1}^{K} \sqrt[2/(m-1)]{1/\mu_{j(k)}} \cdot \mathrm{CTE}_j(k)}{\sum_{k=1}^{K} \sqrt[2/(m-1)]{1/\mu_{j(k)}}}. \tag{27}$$

3. Application and Analyses

To demonstrate the application of the proposed methodology, a real case with the data of 40 jobs from a wafer fabrication factory located in Taichung City Scientific Park, Taiwan (see Table 2), was used, where $x_{j1} \sim x_{j6}$ stand for the job size, factory utilization, the queue length on the route, the queue length before the bottleneck, the work in progress (WIP), and the average waiting time. The wafer fabrication factory produces more than ten products and has a monthly capacity of 20,000 wafers. The wafer fabrication processes include photolithography, thermal processes, implantation, chemical vapor deposition, etching, physical vapor deposition, chemical mechanical polishing, process diagnostics and control, and cleaning. The production characteristic of

TABLE 2: An example.

j	x_{j1}	x_{j2}	x_{j3}	x_{j4}	x_{j5}	x_{j6}	CT_j
1	24	1261	181	781	112	0.92	935
2	24	1263	181	762	127	0.90	958
3	24	1220	176	761	127	0.89	1047
4	23	1282	178	802	127	0.94	1011
5	23	1303	180	780	175	0.93	1068
6	23	1281	183	782	175	0.93	1143
7	23	1242	184	741	163	0.89	1103
8	24	1262	182	681	139	0.86	1250
9	22	1260	182	701	98	0.86	1181
10	22	1260	179	700	257	0.87	1194
11	24	1301	163	722	99	0.84	1260
12	22	1221	184	641	131	0.82	1240
13	23	1323	159	740	247	0.87	1180
14	24	1362	181	782	191	0.95	1227
15	24	1261	181	762	219	0.91	1236
16	23	1321	177	801	219	0.96	1215
17	22	1343	180	822	219	0.97	1228
18	24	1321	177	762	54	0.93	1266
19	25	1343	179	781	54	0.96	1285
20	25	1300	180	740	54	0.92	1272
21	22	1320	181	721	54	0.91	1310
22	24	1321	182	742	49	0.92	1265
23	23	1262	165	680	201	0.80	1308
24	22	1240	161	722	103	0.82	1331
25	23	1183	183	661	53	0.82	1294
26	23	1282	184	701	53	0.88	1314
27	22	1202	177	680	248	0.84	1321
28	23	1202	178	681	248	0.85	1353
29	24	1202	185	701	82	0.86	1226
30	23	1202	158	721	98	0.81	1301
31	24	1343	181	760	67	0.94	1280
32	24	1381	185	801	67	0.97	1286
33	22	1362	156	780	67	0.91	1252
34	23	1282	179	782	223	0.92	1214
35	23	1320	180	782	176	0.93	1251
36	25	1340	176	801	462	0.97	1222
37	23	1320	182	781	168	0.95	1187
38	22	1361	181	781	141	0.94	1205
39	22	1381	179	781	95	0.97	1120
40	23	1363	178	802	179	0.97	1133

FIGURE 5: The comparison of the standard deviations of the inputs.

Subsequently, we standardize the data (see Table 3) and obtain the correlation matrix as

$$R = \begin{bmatrix} 0.97 & 0.10 & 0.16 & 0.21 & -0.03 & 0.25 \\ 0.10 & 0.98 & 0.01 & 0.70 & -0.01 & 0.78 \\ 0.16 & 0.01 & 0.98 & 0.05 & -0.07 & 0.37 \\ 0.21 & 0.70 & 0.05 & 0.98 & 0.15 & 0.86 \\ -0.03 & -0.01 & -0.07 & 0.15 & 0.98 & 0.10 \\ 0.25 & 0.78 & 0.37 & 0.86 & 0.10 & 0.98 \end{bmatrix}. \tag{28}$$

The eigenvalues and eigenvectors of R are then calculated. Based on them, the variance contribution rates can be derived as

$$\eta_1 = 46\%, \qquad \eta_2 = 20\%, \qquad \eta_3 = 16\%,$$
$$\eta_4 = 14\%, \qquad \eta_5 = 4\%, \qquad \eta_6 = 0\%. \tag{29}$$

Summing up η_q's, we obtain

$$\eta_\Sigma(1) = 46\%, \qquad \eta_\Sigma(2) = 65\%, \qquad \eta_\Sigma(3) = 81\%,$$
$$\eta_\Sigma(4) = 95\%, \qquad \eta_\Sigma(5) = 100\%, \qquad \eta_\Sigma(6) = 100\%. \tag{30}$$

After conducting a Pareto analysis, p is chosen as 3 to meet the requirement $\eta_\Sigma(p) \geq 85\% \sim 90\%$. The first three principal components explain roughly 80% of the total variability in the standardized data, so that it might be a reasonable way to reduce the dimensions in order to visualize the data.

Subsequently, the component scores are calculated (see Table 4), which contain the coordinates of the original data in the new coordinate system defined by the principal components, and will be used as the new inputs to the FCM-BPN.

Subsequently, jobs are classified using FCM based on the new variables. The results of the S-test are summarized in Table 5. In this case, the optimal number of job categories was 5. However, there will be some categories with very few jobs. For this reason, the second best solution is used, that is, 4 categories, by setting the threshold of membership to 0.3. The classification results are shown in Table 6.

After preclassification, the three-layer BPN of each category was applied to predict the cycle times of jobs belonging

"reentry," which is highly relevant to the semiconductor industry, is clearly reflected in this problem. It also shows the difficulties facing production planners and schedulers who attempt to provide an accurate due date for a product with a very complicated routing.

The standard deviations of the six inputs are compared in Figure 5. Note that the variability in x_{j2}, x_{j4}, and x_{j5} is substantially higher than that in the remaining variables.

Table 3: The standardized data.

j	x_{j1}	x_{j2}	x_{j3}	x_{j4}	x_{j5}	x_{j6}
1	0.88	−0.53	0.40	0.74	−0.40	0.37
2	0.88	−0.49	0.48	0.30	−0.22	−0.05
3	0.88	−1.27	−0.17	0.29	−0.22	−0.31
4	−0.22	−0.15	0.07	1.18	−0.22	0.65
5	−0.22	0.24	0.37	0.71	0.35	0.58
6	−0.22	−0.17	0.78	0.74	0.35	0.45
7	−0.22	−0.87	0.90	−0.14	0.21	−0.19
8	0.88	−0.51	0.53	−1.45	−0.08	−0.87
9	−1.32	−0.55	0.60	−1.01	−0.56	−0.81
10	−1.32	−0.54	0.23	−1.03	1.34	−0.55
11	0.88	0.20	−1.87	−0.57	−0.56	−1.19
12	−1.32	−1.25	0.80	−2.33	−0.18	−1.64
13	−0.22	0.60	−2.40	−0.16	1.22	−0.60
14	0.88	1.31	0.47	0.75	0.55	0.94
15	0.88	−0.53	0.40	0.31	0.88	0.19
16	−0.22	0.57	−0.05	1.17	0.88	1.13
17	−1.32	0.97	0.33	1.62	0.88	1.38
18	0.88	0.56	−0.01	0.31	−1.09	0.62
19	1.97	0.96	0.15	0.74	−1.09	1.06
20	1.97	0.19	0.38	−0.16	−1.09	0.36
21	−1.32	0.55	0.51	−0.57	−1.09	0.10
22	0.88	0.55	0.54	−0.13	−1.16	0.32
23	−0.22	−0.52	−1.59	−1.47	0.67	−2.00
24	−1.32	−0.91	−2.11	−0.57	−0.51	−1.54
25	−0.22	−1.95	0.72	−1.89	−1.11	−1.64
26	−0.22	−0.15	0.89	−1.01	−1.11	−0.36
27	−1.32	−1.59	−0.07	−1.47	1.23	−1.26
28	−0.22	−1.60	0.07	−1.47	1.23	−1.11
29	0.88	−1.60	0.93	−1.03	−0.76	−0.87
30	−0.22	−1.59	−2.57	−0.58	−0.56	−1.86
31	0.88	0.97	0.41	0.27	−0.94	0.70
32	0.88	1.65	1.00	1.16	−0.94	1.27
33	−1.32	1.30	−2.73	0.71	−0.94	0.10
34	−0.22	−0.14	0.16	0.74	0.92	0.33
35	−0.22	0.55	0.34	0.75	0.36	0.54
36	1.97	0.91	−0.23	1.17	3.79	1.28
37	−0.22	0.55	0.61	0.73	0.27	0.91
38	−1.32	1.28	0.42	0.72	−0.05	0.81
39	−1.32	1.66	0.14	0.72	−0.60	1.36
40	−0.22	1.33	0.13	1.18	0.40	1.42

Table 4: New inputs to the FCM-BPN.

z_{j1}	z_{j2}	z_{j3}
−0.56	0.91	−0.19
−0.13	0.87	−0.34
0.51	0.57	−0.37
−0.97	−0.10	0.20
−0.87	−0.20	−0.26
−0.75	0.14	−0.51
0.57	0.56	−0.66
1.30	1.18	−0.55
1.55	0.31	0.47
1.37	−0.87	−1.04
1.11	−0.59	0.91
3.04	0.63	−0.20
0.51	−2.44	−0.02
−1.94	0.12	−0.43
−0.30	0.35	−1.29
−1.62	−0.84	−0.48
−2.04	−1.24	−0.17
−0.87	0.77	0.89
−1.92	1.34	0.64
−0.58	1.70	0.34
0.22	0.23	1.29
−0.62	1.31	0.73
2.54	−1.26	−0.16
2.39	−1.64	1.20
3.02	1.57	0.14
0.89	1.21	0.66
2.56	−0.74	−1.19
2.19	−0.13	−1.54
1.61	1.90	−0.42
2.72	−1.23	0.87
−1.27	0.99	0.71
−2.56	1.07	0.78
−0.37	−2.44	2.47
−0.60	−0.51	−0.82
−1.06	−0.27	−0.17
−2.54	−1.36	−3.41
−1.31	−0.02	−0.18
−1.32	−0.63	0.67
−1.77	−0.58	1.32
−2.13	−0.66	0.12

to the category according to the new variables. Different network architectures were evaluated to compare the forecasting performance. The best-fitted network which was selected, and, therefore, the architecture which presented the best forecasting accuracy, is composed of three inputs, six hidden and one output neurons.

The convergence condition in training networks was established as either the improvement in MSE becomes less than 10^{-6} with one more epoch or 1000 epochs have already been run. 3/4 of the adopted examples in each category are fed as "training examples" into the BPN. The remaining 1/4 is left for testing. For example, category 3 has 8 jobs; 6 of them are randomly chosen for training the BPN while the remaining 2 jobs are left for testing. The forecasting accuracy can be evaluated with mean absolute error (MAE), mean absolute percentage error (MAPE), and RMSE. The forecasting performances are summarized in Table 7. The forecasting results are shown in Figure 6. The performance of the proposed methodology is compared with those of statistical analysis (i.e., multiple linear regression), BPN,

TABLE 5: The results of the S-test.

Number of categories (K)	J_m	e_{min}^2	S
2	1.96	0.14	0.34
3	1.21	0.09	0.34
4	0.86	0.07	0.30
5	0.67	0.06	0.26
6	0.53	0.03	0.43

TABLE 6: The classifying results ($\mu_L = 0.3$).

Category	Jobs
1	4–6, 14–17
2	10-11, 13, 23-24, 27, 30, 33
3	1, 18–22, 31-32
4	2-3, 7–9, 12, 25-26, 28-29, 34–40

TABLE 7: The forecasting performances.

Category	MAE (hrs)	MAPE	RMSE (hrs)
1	20	1.7%	44
2	6	0.5%	14
3	5	0.4%	12
4	8	0.6%	18
Total	11	0.9%	29

TABLE 8: Comparison of the forecasting performances.

Category	MAE (hrs)	MAPE	RMSE (hrs)
Statistical analysis	73	6.1%	99
BPN	30	2.4%	69
FCM-BPN	15	1.2%	38
PCA-BPN	27	2.3%	67
PCA-FCM-BPN	11	0.9%	29

FCM-BPN, and PCA-BPN in Table 8. The nonlinear nature of this problem is obvious since the performance of statistical analysis (a linear approach) is poor. In addition, the simple combination of PCA and BPN does not have much effect. The main effect of PCA is to improve the correctness of job classification, as mentioned in Chen and Wang [23].

Subsequently, the IUBR approach is applied to determine the upper bound of the cycle time. In the first iteration, $\Delta\theta^{o*}(t)$ is -0.808, and the upper bounds of the cycle times are shown in Figure 7.

The process stops after five iterations because the upper bounds remain unchanged after the fifth iteration. The results of the five iterations are summarized in Table 9, from which the allowances which are 25, 33, 54, 48, 56, 57, 58, 44, 54, 53, 42, 46, 55, 48, 47, 50, 48, 41, 36, 39, 30, 41, 31, 24, 34, 29, 27, 13, 37, 24, 47, 15, 34, 50, 44, 49, 53, 50, 53, and 53 added to the cycle times are derived with an average of 42 (hours). The due date of a job is then set to the release time plus the upper bound of the cycle time.

To make a comparison, six other allowance determination policies are also applied to the collected data.
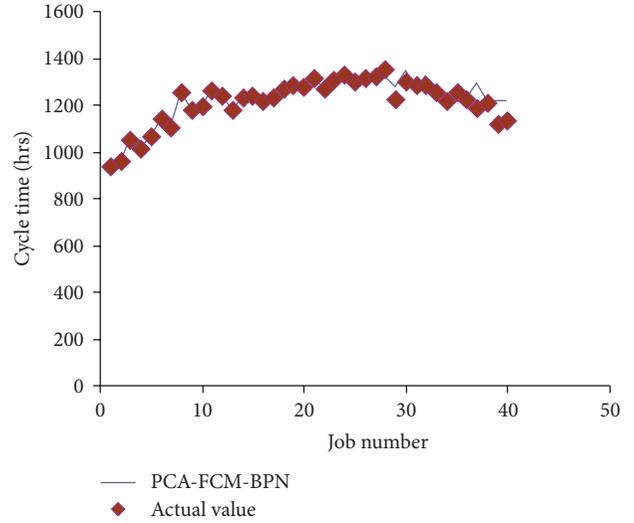
FIGURE 6: The forecasting results using PCA-FCM-BPN.
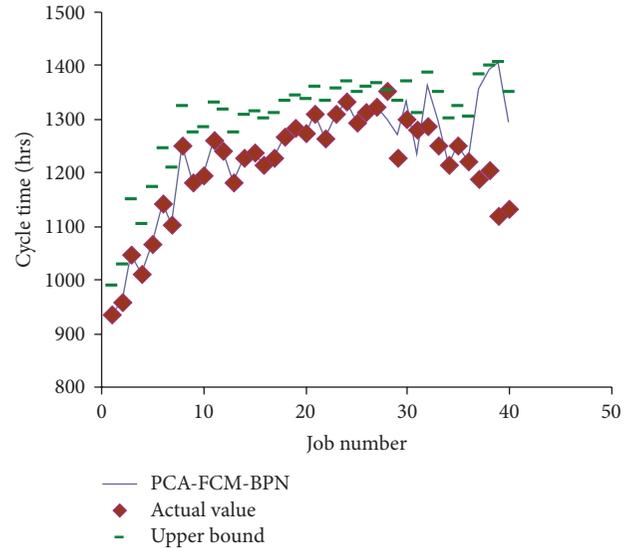
FIGURE 7: The upper bounds of the cycle times.

(1) Total work content policy (TWK): in TWK, the due date allowance factor is estimated based on historical data by a regression model. There is another product in the wafer fabrication factory with an average cycle time of 1278 hours. The total processing time and cycle time standard deviation of the product are 317 and 87 hours, respectively. The product was adopted as the comparison basis, and in this case the cycle time forecast and allowance are determined as follows:

$$\text{Cycle time forecast} = \frac{1278}{317} * \text{the total processing time,}$$

$$\text{Allowance} = 3 * 87 * \left(\frac{\text{the total processing time}}{317}\right).$$

(31)

TABLE 9: The results of the five iterations.

Iteration 1 $(\Delta\theta^{o*}(t) = -0.808)$	Iteration 2 $(\Delta\theta^{o*}(t) = -0.669)$	Iteration 3 $(\Delta\theta^{o*}(t) = -0.446)$	Iteration 4 $(\Delta\theta^{o*}(t) = -0.446)$	Iteration 5 $(\Delta\theta^{o*}(t) = -0.446)$
987	976	960	960	960
1025	1011	991	991	991
1147	1129	1100	1100	1100
1103	1086	1059	1058	1058
1171	1153	1124	1123	1123
1243	1227	1200	1199	1199
1207	1190	1161	1161	1161
1322	1312	1294	1294	1294
1274	1259	1235	1235	1235
1283	1270	1246	1246	1246
1329	1319	1302	1302	1302
1316	1305	1286	1286	1286
1273	1259	1234	1234	1234
1307	1296	1276	1275	1275
1313	1302	1282	1282	1282
1299	1286	1265	1265	1265
1308	1296	1276	1276	1276
1332	1323	1307	1307	1307
1344	1336	1321	1321	1321
1336	1327	1311	1311	1311
1358	1352	1340	1340	1340
1331	1322	1305	1305	1305
1357	1350	1338	1338	1338
1369	1364	1355	1355	1355
1349	1342	1328	1328	1328
1360	1354	1343	1343	1343
1364	1358	1348	1348	1348
1353	1353	1353	1353	1353
1334	1334	1320	1320	1320
1369	1364	1355	1355	1355
1311	1300	1280	1280	1280
1384	1381	1376	1376	1376
1350	1343	1329	1329	1329
1298	1286	1264	1263	1263
1323	1313	1295	1295	1295
1304	1292	1271	1271	1271
1381	1378	1371	1250	1250
1399	1399	1397	1265	1265
1405	1405	1405	1247	1247
1348	1340	1326	1244	1244

(2) Gamma distribution fitting method (Gamma): the waiting time of a job is fitted with a Gamma distribution. For example, the waiting time of a job with 24 pieces of wafers is fitted with a Gamma distribution in Figure 8. The 50% and 95% percentiles are 929 and 1160, respectively, and the total processing time is 251 hours. So the cycle time forecast is 1160 + 251 = 1411 hours, and allowance is 1160 − 929 = 231 hours.

(3) Constant allowance policy (CON, PCA-FCM-BPN + CON): add three times the RMSE of the prediction approach to the completion time forecasts to determine the due date.
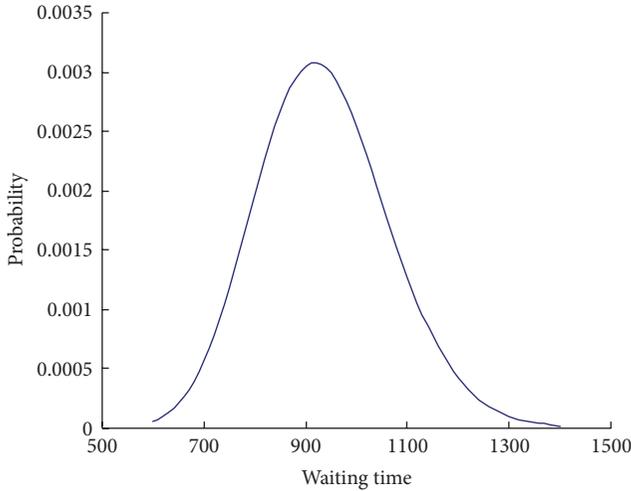
FIGURE 8: Fitting the waiting time with a Gamma distribution (jobs with 24 wafers).

(4) Selective allowance policy (SAP, PCA-FCM-BPN + SAP): add three times the RMSE of the prediction approach to the completion time forecasts of a small quantity of jobs that might encounter difficulties in keeping the internal due date. Such jobs are chosen in the following way:

$$\sum_{i=1}^{6} N\left(x_{ji}\right) \geq 0.5 \cdot \max_{j} \sum_{i=1}^{6} N\left(x_{ji}\right) + 0.5 \cdot \min_{j} \sum_{i=1}^{6} N\left(x_{ji}\right).$$
(32)

In other words, these jobs are among the 50% percentiles.

(5) Random assignment policy (RAP, PCA-FCM-BPN + RAP): add the extra allowance to the completion time forecasts of the same quantity of jobs that are randomly chosen.

(6) No allowance policy (NAP, PCA-FCM-BPN + NAP): no allowance will be assigned to any job.

Due date related performances are impacted by the quality of the due date assignment methods. After applying the seven allowance determination policies, the following performance measures are compared:

(1) number of tardy jobs (N_T);

(2) mean tardiness ($\overline{T}$);

(3) sum of allowances.

The comparison results are summarized in Table 10. The proposed IUBR approach outperforms the other allowance determination policies.

(1) It guarantees the on-time delivery of the jobs. Both N_T and $\overline{T}$ are zeros. Among the other allowance determination policies, only Gamma and CON can achieve that at the expense of adding some extra allowance.

(2) The percentage of reduction in the sum of allowances over CON is 52%. The advantages over TWK, Gamma, SAP, and RAP are 79%, 74%, 12%, and 12%, respectively. The percentage of on-time delivery is not derived from a greater buffer on the completion time prediction.

(3) The performance of SAP is not better than that of RAP, which shows it is not easy to anticipate jobs that may delay.

(4) Compared with TWK and Gamma, the other policies effectively reduce the allowances added to the job cycle times, which is due to the forecasting accuracy of the PCA-FCM-BPN approach.

4. Conclusions and Directions for Future Research

Owing to the complexity of the wafer fabrication, the due date assignment of each job presents a challenging problem to the production planning and scheduling people. The firm has to offer a price reduction if the due date is far away from the expected one. Conversely, the looser the due date is set, the higher the probability that the job will be completed or delivered on time is. That is very important to maintain a good reputation with the customers. This study explores a new application of fuzzy-neural approaches in the due date assignment problem of the wafer fabrication factory. The proposed methodology decomposes internal due date assignment in a wafer fabrication factory into two subproblems: completion time prediction and allowance determination. To overcome the problems with the existing approaches, two innovative treatments are taken in the proposed methodology. First, PCA is applied to construct a series of linear combinations of the original variables to form a new variable, so that these new variables are unrelated to each other as much as possible, and the relationship among them can be reflected in a better way. The combination of PCA and BPN also reduces the space for storing the input variables in the modeling of the wafer fabrication system. In addition, the simultaneous application of PCA, FCM, and BPN further improved the estimation accuracy. Subsequently, the IUBR approach is proposed to determine the allowance that will be added to the estimated job cycle time. Our result is existentially tight.

The validity that the effective fuzzy-neural approach for internal due date assignment is able to improve on-time delivery has been proved by the case study. Based on the above analysis,

(1) the forecasting accuracy (measured with MAE, MAPE, and RMSE) of the PCA-FCM-BPN was significantly better than those of many existing approaches;

(2) it is easier to determine the allowance in the IUBR method than the method based on NLP;

(3) the bound on the job cycle time is tighter than the bounds by TWK, Gamma, and CON and simpler than the bound by Chen and Wang [15], which requires NLP optimization.

TABLE 10: The performances of various allowance determination policies.

	TWK	Gamma	CON	SAP	RAP	NAP (basis)	IUBR
Sum of allowances	7982	6479	3480	1914	1914	0	1680
N_T	31	0	0	9	7	14	0
$\overline{T}$ (hours)	82	0	0	3.9	2.1	3.5	0

However, there are two limitations that need to be acknowledged and addressed regarding the present study.

(1) The first limitation concerns the experimental nature of this research. The proposed methodology was studied within a short period of time. There is an apparent danger involved whenever conclusions are drawn from such a limited sample and then applied in the highly dynamic semiconductor manufacturing environment.

(2) The BPN part in the methodology is usually regarded as a black box. To exploit the knowledge embedded in the back box, and to facilitate the practical application of the proposed methodology, some association rules have to be extracted from the estimation results.

The IUBR approach only modifies the threshold of the output node. In future studies, other parameters in the BPN can be modified in similar ways. However, it is a challenge to make the modification results independent of the original parameter values. In addition, the concept of customer satisfaction can be incorporated into the proposed methodology; thereby, the due date can achieve a higher level of customer satisfaction. In contrast, the proposed methodology only guarantees a positive level of customer satisfaction.

Acknowledgment

References

[1] T. Chen, A. Jeang, and Y. Wang, "A hybrid neural network and selective allowance approach for internal due date assignment in a wafer fabrication plant," *International Journal of Advanced Manufacturing Technology*, vol. 36, no. 5-6, pp. 570–581, 2008.

[2] T. Chen, "A flexible way of modelling the long-term cost competitiveness of a semiconductor product," *Robotics & Computer Integrated Manufacturing*, vol. 29, no. 3, pp. 31–40, 2013.

[3] W. L. Pearn, S. H. Chung, and C. M. Lai, "Due-date assignment for wafer fabrication under demand variate environment," *IEEE Transactions on Semiconductor Manufacturing*, vol. 20, no. 2, pp. 165–175, 2007.

[4] C. F. Chien, C. Y. Hsu, and C. W. Hsiao, "Manufacturing intelligence to forecast and reduce semiconductor cycle time," *Journal of Intelligent Manufacturing*, vol. 23, no. 6, pp. 2281–2294, 2011.

[5] Y. Wilamowsky, S. Epstein, and B. Dickman, "Optimal common due-date with completion time tolerance," *Computers and Operations Research*, vol. 23, no. 12, pp. 1203–1210, 1996.

[6] Y. C. Wang, T. Chen, and C. W. Lin, "A slack-diversifying nonlinear fluctuation smoothing rule for job dispatching in a wafer fabrication factory," *Robotics & Computer Integrated Manufacturing*, vol. 29, no. 3, pp. 41–47, 2013.

[7] A. K. Gupta and A. I. Sivakumar, "Optimization of due-date objectives in scheduling semiconductor batch manufacturing," *International Journal of Machine Tools and Manufacture*, vol. 46, no. 12-13, pp. 1671–1679, 2006.

[8] 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.

[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, Y. Wang, and H. 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.

[11] T. Chen, H. Wu, and Y. 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. 1225–1231, 2009.

[12] T. Chen, "Job cycle time estimation in a wafer fabrication factory with a bi-directional classifying fuzzy-neural approach," *International Journal of Advanced Manufacturing Technology*, vol. 56, no. 9–12, pp. 1007–1018, 2011.

[13] B. E. Ankenman, J. M. Bekki, J. Fowler, G. T. Mackulak, B. L. Nelson, and F. Yang, "Simulation in production planning: an overview with emphasis on recent developments in cycle time estimation," *International Series in Operations Research & Management Science*, vol. 151, pp. 565–591, 2011.

[14] V. Gordon, J.-M. Proth, and C. Chu, "A survey of the state-of-the-art of common due date assignment and scheduling research," *European Journal of Operational Research*, vol. 139, no. 1, pp. 1–25, 2002.

[15] T. Chen and Y. Wang, "Incorporating the FCM-BPN approach with nonlinear programming for internal due date assignment in a wafer fabrication plant," *Robotics and Computer-Integrated Manufacturing*, vol. 26, no. 1, pp. 83–91, 2010.

[16] T. Chen and Y. Lin, "A collaborative fuzzy-neural approach for internal due date assignment in a wafer fabrication plant," *International Journal of Innovative Computing, Information and Control*, vol. 7, no. 9, pp. 5193–5210, 2011.

[17] T. Chen, "A collaborative and artificial intelligence approach for semiconductor cost forecasting," *Computers & Industrial Engineering*, vol. 66, no. 2, pp. 476–484, 2013.

[18] K. Pearson, "On lines and planes of closest fit to systems of points in space," *Philosophical Magazine*, vol. 2, no. 11, pp. 559–572, 1901.

[19] H. Hotelling, "Analysis of a complex of statistical variables into principal components," *Journal of Educational Psychology*, vol. 24, no. 6, pp. 417–441, 1933.

[20] A. Jaiswal, N. Kumar, and R. K. Agrawal, "A hybrid of principal component analysis and partial least squares for face recognition across pose," in *Progress in Pattern Recognition, Image Analysis, Computer Vision, and Applications*, vol. 7441 of *Lecture Notes in Computer Science*, pp. 67–73, Springer, Berlin, Germany, 2012.

[21] J. Mohtasham, A. S. Nateri, and H. Khalili, "Textile colour matching using linear and exponential weighted principal component analysis," *Coloration Technology*, vol. 128, no. 3, pp. 199–203, 2012.

[22] T. Chen and Y. C. Wang, "Long-term load forecasting by the collaborative fuzzy-neural approach," *International Journal of Electrical Power and Energy Systems*, vol. 43, pp. 454–464, 2012.

[23] T. Chen and Y. C. Wang, "An iterative procedure for optimizing the performance of the fuzzy-neural job cycle time estimation approach in a wafer fabrication factory," *Mathematical Problems in Engineering*, vol. 2013, Article ID 740478, 15 pages, 2013.

[24] X. L. Xie and G. Beni, "A validity measure for fuzzy clustering," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 13, no. 8, pp. 841–847, 1991.

[25] J. Heaton, *Introduction to Neural Networks with Java*, Heaton Research, St. Louis, Mo, USA, 2013.

[26] M. Khashei, *Forecasting the Esfahan steel company production price in Tehran metals exchange using artificial neural networks (ANNs) [M.S. thesis]*, Isfahan University of Technology, Isfahan, Iran, 2005.

[27] H. Akaike, "A new look at the statistical model identification," *IEEE Transactions on Automatic Control*, vol. 19, no. 6, pp. 716–723, 1974.

[28] J. Rissanen, "Stochastic complexity and modeling," *The Annals of Statistics*, vol. 14, no. 3, pp. 1080–1100, 1986.

[29] T. Chen, "Incorporating fuzzy c-means and a back-propagation network ensemble to job completion time prediction in a semiconductor fabrication factory," *Fuzzy Sets and Systems*, vol. 158, no. 19, pp. 2153–2168, 2007.

[30] H. C. Wu and T. Chen, "A fuzzy-neural ensemble and geometric rule fusion approach for scheduling a wafer fabrication factory," *Mathematical Problems in Engineering*, vol. 2013, Article ID 956978, 14 pages, 2013.