A Fuzzy-neural Approach Incorporating Exponentially Discounted Future Workload for Predicting Wafer Lot Output Time

Horng-Ren Tsai[†] and Toly Chen[††],

[†]Lingtung University, Taichung City, Taiwan
[††]Feng Chia University, Taichung City, Taiwan

Summary
Lot output time prediction is a critical task to a wafer fabrication plant (wafer fab). To enhance the effectiveness, a look-ahead FBPN incorporating the future release plan is constructed in this study. Three nearest exponentially discounted future workloads, modified from Chen's nearest future discounted workloads, are proposed for the look-ahead function. According to experimental results, the prediction accuracy of the look-ahead FBPN was significantly better than those of some existing approaches. Besides, the proposed nearest exponentially discounted future workload functions are shown to be more effective than Chen's nearest future discounted workload functions in incorporating the fab's future release plan.

Key words:
Fuzzy back propagation network, Output time prediction, Wafer fabrication.

1. Introduction

Predicting the output time for every lot in a wafer fab is a critical task not only to the fab itself, but also to its customers. After the output time of each lot in a wafer fab is accurately predicted, several managerial goals can be simultaneously achieved [6]. Predicting the output time of a wafer lot is equivalent to estimating the cycle time of the lot, because the former can be easily derived by adding the release time (a constant) to the latter.

There are six major approaches commonly applied to predicting the output/cycle time of a wafer lot: multiple-factor linear combination (MFLC), production simulation (PS), back propagation networks (BPN), case based reasoning (CBR), fuzzy modeling methods, and hybrid approaches. Among the six approaches, MFLC is the easiest, quickest, and most prevalent in practical applications. The major disadvantage of MFLC is the lack of forecasting accuracy [6]. Conversely, huge amount of data and lengthy simulation time are two shortages of PS. Nevertheless, PS is the most accurate output time prediction approach if the related databases are continuingly updated to maintain enough validity, and often serves as a benchmark for evaluating the effectiveness of another method. PS also tends to be preferred because it allows for computational experiments and subsequent analyses without any actual execution [3]. Considering both effectiveness and efficiency, Chang et al. [4] and Chang and Hsieh [2] both forecasted the output/cycle time of a wafer lot with a BPN having a single hidden layer. Compared with MFLC approaches, the average prediction accuracy measured with root mean squared error (RMSE) was considerably improved with these BPNs. For example, an improvement of about 40% in RMSE was achieved in Chang et al. [4]. On the other hand, much less time and fewer data are required to generate an output time forecast with a BPN than with PS. More recently, Chen [7] incorporated the fab's future release plan into a BPN, and constructed a "look-ahead" BPN for the same purpose, which led to an average reduction of 12% in RMSE. Chang et al. [3] proposed a k-nearest-neighbors based case-based reasoning (CBR) approach which outperformed the BPN approach in forecasting accuracy. In one case, the advantage was up to 27%. Chang et al. [4] modified the first step (i.e. partitioning the range of each input variable into several fuzzy intervals) of the fuzzy modeling method proposed by Wang and Mendel [13], called the WM method, with a simple genetic algorithm (GA) and proposed the evolving fuzzy rule (EFR) approach to predict the cycle time of a wafer lot. Their EFR approach outperformed CBR and BPN in prediction accuracy. Chen [6] constructed a fuzzy BPN (FBPN) that incorporated expert opinions in forming inputs to the FBPN. Chen's FBPN was a hybrid approach (fuzzy modeling and BPN) and surpassed the crisp BPN especially in the efficiency respect. Another hybrid approach was proposed in Chang and Liao [5] by combining self-organization map (SOM) and WM, in which a wafer lot was classified with SOM before predicting the lot's output time with WM. Recently, Chen [8] constructed a kM-BPN for the same purpose.

To enhance the effectiveness, a look-ahead FBPN incorporating the future release plan is constructed in this study. PS is also applied in this study to generate test examples. According to experimental results, the

Manuscript received December 5, 2006.
Manuscript revised January 20, 2007.

prediction accuracy of the look-ahead FBPN was significantly better than those of some existing approaches.

2. Methodology

The parameters used in the following are defined:
(1) R_n: the release time of the n-th lot.
(2) U_n: the average fab utilization at R_n.
(3) Q_n: the total queue length on the lot's processing route at R_n.
(4) BQ_n: the total queue length before bottlenecks at R_n.
(5) FQ_n: the total queue length in the whole fab at R_n.
(6) WIP_n: the fab WIP at R_n.
(7) $D_n^{(i)}$: the delay of the i-th recently completed lots, i = 1~3.

2.1 Incorporating the Future Release Plan (Look-ahead)

Almost all existing methods are based on the historical data of the fab. However, a lot of studies have shown that the performance of sequencing and scheduling in a fab relies heavily on the future release plan, which has been neglected in this field. In addition, the characteristic re-entrant production flows of a fab lead to the phenomenon that a lot that will be released in the future might appear in front of another lot that currently exists in the fab. For these reasons, to further improve the accuracy of wafer lot output time prediction, the future release plan of the fab has to be considered (look-ahead). There are many possible ways to incorporate the future release plan in predicting the output time of a wafer lot currently existing in the fab. The first one is the three nearest future discounted workload functions proposed by Chen [7]:

(1) *The 1st nearest future discounted workload* ($FDW_n^{(1)}$): the sum of the (processing time/release time)'s of the operations of the lots that will be released within time $[R_n, R_n + T_1]$.
(2) *The 2nd nearest future discounted workload* ($FDW_n^{(2)}$): the sum of the (processing time/release time)'s of the operations of the lots that will be released within time $[R_n + T_1, R_n + T_1 + T_2]$.
(3) *The 3rd nearest future discounted workload* ($FDW_n^{(3)}$): the sum of the (processing time/release time)'s of the operations of the lots that will be released within time $[R_n + T_1 + T_2, R_n + T_1 + T_2 + T_3]$.

In this study, the three nearest exponentially discounted future workloads on the lot's processing route (according to the future release plan) are proposed for the same purpose:

(1) *The 1st nearest exponentially discounted future workload* ($EDFW_n^{(1)}$): the sum of the (processing time * exp(-release time))'s of the operations of the lots that will be released within time $[R_n, R_n + T_1]$.
(2) *The 2nd nearest exponentially discounted future workload* ($EDFW_n^{(2)}$): the sum of the (processing time * exp(-release time))'s of the operations of the lots that will be released within time $[R_n + T_1, R_n + T_1 + T_2]$.
(3) *The 3rd nearest exponentially discounted future workload* ($EDFW_n^{(3)}$): the sum of the (processing time * exp(-release time))'s of the operations of the lots that will be released within time $[R_n + T_1 + T_2, R_n + T_1 + T_2 + T_3]$.

Note that only the operations performed on the machines on the lot's processing route are considered in calculating these future workloads, which then become three additional inputs to the FBPN.

2.1 Output Time Prediction with FBPN

The configuration of the FBPN is established as follows:
(1) Inputs: eleven parameters associated with the n-th example/lot including U_n, Q_n, BQ_n, FQ_n, WIP_n, $D_n^{(i)}$ (i = 1~3), and $EDFW_n^{(r)}$ (r = 1~3). These parameters have to be normalized so that their values fall within [0, 1]. Then some production execution/control experts are requested to express their beliefs (in linguistic terms) about the importance of each input parameter in predicting the cycle (completion) time of a job. Linguistic assessments for an input parameter are converted into several pre-specified triangular fuzzy numbers (TFNs). The subjective importance of an input parameter is then obtained by averaging the corresponding fuzzy numbers of the linguistic replies for the input parameter by all experts. The subjective importance obtained for an input parameter is multiplied to the normalized value of the input parameter. After such a treatment, all inputs to the FBPN become TFNs, and the fuzzy arithmetic for TFNs is applied to deal with all calculations involved in training the FBPN.
(2) Single hidden layer: Generally one or two hidden layers are more beneficial for the convergence property of the network.
(3) Number of neurons in the hidden layer: the same as that in the input layer. Such a treatment has been adopted by many studies (e.g. [2, 5]).
(4) Output: the (normalized) cycle time forecast of the example.
(5) Network learning rule: Delta rule.
(6) Transformation function: Sigmoid function,

$$f(x) = 1/(1+e^{-x}). \quad (1)$$

(7) Learning rate ( ): 0.01~1.0.
(8) Batch learning.

The procedure for determining the parameter values is now described. A portion of the examples is fed as "training examples" into the FBPN to determine the parameter values. Two phases are involved at the training stage. At first, in the forward phase, inputs are multiplied with weights, summated, and transferred to the hidden layer. Then activated signals are outputted from the hidden layer as:

$$\tilde{h}_j = (h_{j1},\ h_{j2},\ h_{j3}) = \frac{1}{1+e^{-\tilde{n}_j^h}} = \left(\frac{1}{1+e^{-n_{j1}^h}},\ \frac{1}{1+e^{-n_{j2}^h}},\ \frac{1}{1+e^{-n_{j3}^h}}\right), \quad (2)$$

where

$$\tilde{n}_j^h = (n_{j1}^h,\ n_{j2}^h,\ n_{j3}^h) = \tilde{I}_j^h (-) \tilde{\theta}_j^h = (I_{j1}^h - \theta_{j3}^h,\ I_{j2}^h - \theta_{j2}^h,\ I_{j3}^h - \theta_{j1}^h), \quad (3)$$

$$\tilde{I}_j^h = (I_{j1}^h,\ I_{j2}^h,\ I_{j3}^h) = \sum_{all\ i} \tilde{w}_{ij}^h (\times) \tilde{x}_{(i)} \cong \left(\sum_{all\ i} \min(w_{ij1}^h x_{(i)1},\ w_{ij3}^h x_{(i)3}\right), \quad (4)$$

where $(-)$ and $(\times)$ denote fuzzy subtraction and multiplication, respectively; $\tilde{h}_j$'s are also transferred to the output layer with the same procedure. Finally, the output of the FBPN is generated as:

$$\tilde{o} = (o_1,\ o_2,\ o_3) = 1/1+e^{-\tilde{n}^o}, \quad (5)$$

where

$$\tilde{n}^o = (n_1^o,\ n_2^o,\ n_3^o) = \tilde{I}^o (-) \tilde{\theta}^o, \quad (6)$$

$$\tilde{I}^o = (I_1^o, I_2^o, I_3^o) = \sum_{all\ j} \tilde{w}_j^o (\times) \tilde{h}_j. \quad (7)$$

To compare with the normalized actual cycle time o, the fuzzy-valued output $\tilde{o}$ is defuzzified according to the centroid-of-area (COA) formula:

$$o = \mathrm{COA}(\tilde{o}) = (o_1 + 2o_2 + o_3)/4. \quad (8)$$

Then RMSE is calculated:

$$\mathrm{RMSE} = \sqrt{\sum (o-a)^2 / \mathrm{number\ of\ examples}}. \quad (9)$$

Subsequently in the backward phase, the deviation between o and a is propagated backward, and the error terms of neurons in the output and hidden layers can be calculated, respectively, as

$$\delta^o = o(1-o)(a-o), \quad (10)$$

$$\tilde{\delta}_j^h = (\delta_{j1}^h,\ \delta_{j2}^h,\ \delta_{j3}^h) = \tilde{h}_j (\times)(1-\tilde{h}_j)(\times)\tilde{w}_j^o \delta^o. \quad (11)$$

Based on them, adjustments that should be made to the connection weights and thresholds can be obtained as

$$\Delta\tilde{w}_j^o = (\Delta w_{j1}^o,\ \Delta w_{j2}^o,\ \Delta w_{j3}^o) = \eta\delta^o \tilde{h}_j, \quad (12)$$

$$\Delta\tilde{w}_{ij}^h = (\Delta w_{ij1}^h,\ \Delta w_{ij2}^h,\ \Delta w_{ij3}^h) = \eta\tilde{\delta}_j^h (\times)\tilde{x}_i, \quad (13)$$

$$\Delta\theta^o = -\eta\delta^o, \quad (14)$$

$$\Delta\tilde{\theta}_j^h = (\Delta\theta_{j1}^h, \Delta\theta_{j2}^h, \Delta\theta_{j3}^h) = -\eta\tilde{\delta}_j^h. \quad (15)$$

Theoretically, network-learning stops when RMSE falls below a pre-specified level, or the improvement in RMSE becomes negligible with more epochs, or a large number of epochs have already been run. In addition, to avoid the accumulation of fuzziness during the training process, the lower and upper bounds of all fuzzy numbers in the FBPN will no longer be modified if Chen's index [6] converges to a minimal value. Then test examples are fed into the FBPN to evaluate the accuracy of the network that is also measured with the RMSE. Finally, the FBPN can be applied to predicting the cycle time of a new lot. When a new lot is released into the fab, the eleven parameters associated with the new lot are recorded and fed as inputs to the FBPN. After propagation, the network output determines the output time forecast of the new lot.

3. Simulation

In practical situations, the history data of each lot is only partially available in the factory. Further, some information of the previous lots such as Q_n, BQ_n, and FQ_n is not easy to collect on the shop floor. Therefore, a simulation model is often built to simulate the manufacturing process of a real wafer fabrication factory [1-6, 10, 12]. Then, such information can be derived from the shop floor status collected from the simulation model [3]. To generate a demonstrative example, the simulation model constructed in Chen [8] is adopted in this study.

3.1 Results and Discussions

In the demonstrative example, the following five approaches were all applied for comparison to five test cases containing the data of full-size (24 wafers per lot) lots with different product types and priorities:

(1) BPN.
(2) FBPN.
(3) CBR.
(4) Look-ahead FBPN with Chen's nearest future discounted workload functions, indicated with L/a FBPN-1.
(5) Look-ahead FBPN with the proposed nearest exponentially discounted future workload functions, indicated with L/a FBPN-2.

The minimal RMSEs achieved by applying these approaches to different cases were recorded and compared in Table 1. The convergence condition was established as either the improvement in RMSE becomes less than 0.001 with one more epoch, or 1000 epochs have already been run. According to experimental results, the following discussions are made:

(1) From the effectiveness viewpoint, the prediction accuracy (measured with RMSE) of L/a FBPN-2 was significantly better than those of the other approaches in all cases by achieving a 11%~35% (and an average of 23%) reduction in RMSE over the comparison basis – the BPN. The average advantage over CBR is 23%.

(2) The effect of "look ahead" by incorporating the three nearest exponentially discounted workloads is revealed with the fact that L/a FBPN-2 surpassed the FBPN (without look-ahead) in all cases. The advantage ranges from 8% to 29%.

(3) As the lot priority increases, the superiority of L/a FBPN-2 over FBPN becomes more evident.

(4) The proposed nearest exponentially discounted future workload functions are more effective than Chen's nearest future discounted workload functions in incorporating the fab's future release plan, which leads to an average reduction of 8% in RMSE.

Table 1: Comparisons of the RMSEs of various approaches

RMSE	A (normal)	A (hot)	A (super hot)	B (normal)	B (hot)
(1)	177.1	102.27	12.23	286.93	75.98
(2)	171.82 (-3%)	89.5 (-12%)	11.34 (-7%)	286.14 (-0%)	76.14 (+0%)
(3)	172.44 (-3%)	86.66 (-15%)	11.59 (-5%)	295.51 (+3%)	78.85 (+5%)
(4)	163.45 (-8%)	85.6 (-16%)	8.09 (-34%)	264.8 (-8%)	69.65 (-8%)
(5)	158.24 (-11%)	79.46 (-22%)	7.95 -35%)	231.51 (-19%)	54.23 (-29%)

3. Conclusions and Directions for Future Research

A look-ahead FBPN incorporating the future release plan is constructed in this study to enhance the effectiveness of wafer lot output time prediction. Three nearest exponentially discounted future workloads, modified from Chen's nearest future discounted workloads, are proposed for the look-ahead function. According to experimental results, the prediction accuracy of the look-ahead FBPN was significantly better than those of some existing approaches. Besides, the proposed nearest exponentially discounted future workload functions are shown to be more effective than Chen's nearest future discounted workload functions in incorporating the fab's future release plan.

However, to further evaluate the advantages and disadvantages of the proposed methodology, it has to be applied to a full-scale actual wafer fab in future research.

Acknowledgments

This work was support by National Science Council, R.O.C.

References

[1] Barman, S.: The impact of priority rule combinations on lateness and tardiness. IIE Transactions 30 (1998) 495-504.

[2] Chang, P.-C., Hsieh, J.-C.: A Neural Networks Approach for Due-date Assignment in a Wafer Fabrication Factory. International Journal of Industrial Engineering 10(1) (2003) 55-61.

[3] Chang, P.-C., Hsieh, J.-C., Liao, T. W.: A Case-based Reasoning Approach for Due Date Assignment in a Wafer Fabrication Factory. In: Proceedings of the International Conference on Case-Based Reasoning (ICCBR 2001), Vancouver, British Columbia, Canada (2001).

[4] Chang, P.-C., Hsieh, J.-C., Liao, T. W.: Evolving Fuzzy Rules for Due-date Assignment Problem in Semiconductor Manufacturing Factory. Journal of Intelligent Manufacturing 16 (2005) 549-557.

[5] Chang, P.-C., Liao, T. W.: Combining SOM and Fuzzy Rule Base for Flow Time Prediction in Semiconductor Manufacturing Factory. Applied Soft Computing 6 (2006) 198-206.

[6] Chen, T.: A Fuzzy Back Propagation Network for Output Time Prediction in a Wafer Fab. Applied Soft Computing 2/3F (2003) 211-222.

[7] Chen, T.: A Look-ahead Fuzzy Back Propagation Network for Lot Output Time Series Prediction in a Wafer Fab. Lecture Notes in Computer Science 4234 (2006) 974-982.

[8] Chen, T., Lin, Y. C.: A Hybrid and Intelligent System for Predicting Lot Output Time in a Semiconductor Fabrication Factory. Lecture Notes in Artificial Intelligence 4259 (2006) 757-766.

[9] Chung, S.-H., Yang, M.-H., Cheng, C.-M.: The Design of Due Date Assignment Model and the Determination of Flow Time Control Parameters for the Wafer Fabrication Factories. IEEE Transactions on Components, Packaging, and Manufacturing Technology – Part C 20(4) (1997) 278-287.

[10] Hung, Y.-F., Chang, C.-B.: Dispatching Rules Using Flow Time Predictions for Semiconductor Wafer Fabrications. In: Proceedings of the 5th Annual International Conference on Industrial Engineering Theory, Applications and Practice, Taiwan (2001).

[11] Ishibuchi, H., Nozaki, K., Tanaka, H.: Distributed Representation of Fuzzy Rules and Its Application to Pattern Classification. Fuzzy Sets and Systems 52(1) (1992) 21-32.

[12] Lin, C.-Y.: Shop Floor Scheduling of Semiconductor Wafer Fabrication Using Real-time Feedback Control and Prediction. Ph.D. Dissertation, Engineering-Industrial

Engineering and Operations Research, University of California at Berkeley (1996).
[13] Wang, L.-X., Mendel, J. M.: Generating Fuzzy Rules by Learning from Examples. IEEE Transactions on Systems, Man, and Cybernetics 22(6) (1992) 1414-1427.

Horng-Ren Tsai received the Ph.D. degree in Electrical Engineering from National Taiwan University of Science and Technology. He is currently an associate professor of the Department of Information Technology at Lingtung University.

Toly Chen received the Ph.D. degree in Industrial Engineering from National Tsin Hua University. He is currently an assistant professor of the Department of Industrial Engineering and Systems Management at Feng Chia University.