International Journal of Industrial Engineering, 16(1), 32-40, 2009.

A Comparative Study of Three Predictive Tools for Forecasting a Transfer Line's Throughput

Patrick A. Hillberg[1], Sankar Sengupta[2] and Robert P. Van Til[3]

[1]Senior Manufacturing Consultant
DELMIA Corp.
900 N. Squirrel Rd.
Auburn Hills, MI 48326

[2]Associate Professor, Industrial and Systems Engineering Dept.
Oakland University
Rochester, MI 48309

[3]Professor, Industrial and Systems Engineering Dept.
Oakland University
Rochester, MI 48309

Corresponding author's Email: {Sankar Sengupta, sengupta@oakland.edu}

A study comparing the performance of three predictors of a manufacturing system's future state based upon its current state is presented. The three predictors are constructed using statistical regression, neural networks and case-based reasoning techniques. An asynchronous transfer line with unreliable machines is considered for this study. The line's current Work-In-Process (WIP) is used to forecast future throughput. A simulation model of the system is used to generate training data for constructing the predictors as well as data used for validation. The impacts of the number of machines and the number of line partitions on forecast accuracy are investigated. Results indicate that the predictors based on statistical regression and neural network techniques offered comparable performance, and both performed better than the predictor based on case-based reasoning technique. Future work includes development of a prediction model to assist in day-to-day operational tasks such as scheduling of opportunistic maintenance and manpower planning.

Significance: The objective is to develop a decision support tool to help the managers in an automobile assembly plant to make efficient day-to-day operational decisions. These decisions have severe impacts on unit cost and resource utilization.

Keywords: Forecasting, statistical regression, neural networks, case-based reasoning, system simulation.

(Received 15 June 2007; Accepted in revised form 3 March 2009)

1. INTRODUCTION

Manufacturing environments are dynamic and stochastic in nature. In order to maximize productivity, people and processes must respond to changing conditions as, or even before, they occur. Real-time scheduling techniques are often used to identify and respond to the current state of a manufacturing system (Soroush et al., 2005) and (Lee, 2007). However, accurate forecasts of the system's future state also provide useful information for managers as they try to optimize its performance (Cox et al., 1997), (Moon et al., 2003), (Hsu et al., 2004) and (Kwak et al., 2005).

A key performance measure for most manufacturing systems is the throughput (# parts/unit time). This paper will consider the construction of throughput prediction tools. A prediction tool collects real-time performance data from the manufacturing system and uses it to make a forecast of future throughput. Three separate prediction tools are constructed using the following data-driven modeling techniques: neural networks (Jang et al., 1997); statistical regression (Neter et al., 1996); and case-based reasoning (Lenz et al., 1998).

Neural networks have been studied since the late 1950's when Rosenblatt (Rosenblatt, 1962) first applied single layer perceptrons to pattern classification. They are composed of simple elements operating in parallel. The elements are inspired by biological nervous systems. How the network functions is determined largely by the connections between its elements and the weighting values associated with each connection. A comprehensive discussion of neural networks is presented in (Hagan et al., 1996) and (Degbtose et al., 2003).

Regression analysis is a statistical technique for modeling and investigating the relationship between two or more variables (Neter et al., 1996). Regression analysis is used to build a model that describes some output value as a function of input values.

Case-based reasoning, also known as memory-based reasoning, is an approach to problem solving by means of reasoning through analogy (Han et al., 2000). It provides a means of comparing an event that has not been encountered before, by associating it with events that have occurred. Hence, it concerns the use of analogy and has been extended to the development of computer support of human problem solving. Note that analogical reasoning presumes that experiential case history may provide insight towards the solutions of new problems.

In this paper, throughput prediction tools are constructed using these three techniques and their forecasting performance is compared. Note that to successfully use these three data-driven modeling techniques, a large set of input-output data reflecting the system's behavior is required.

The manufacturing system used in this study is an asynchronous transfer line with branches and a merge, Figure 1. This transfer line model is developed from the framing line in an automobile assembly plant and contains 53 stations. Parts enter the transfer line from the right side of Figure 1 (load station) and exit from the left (unload station).

Since such systems in modern automobile assembly plants are highly automated, cycle time variation in the line's stations are usually small. Hence, the stations in this line are modeled with constant cycle times that vary from station to station. Several small buffers, or accumulators, of varying sizes are located throughout the line.

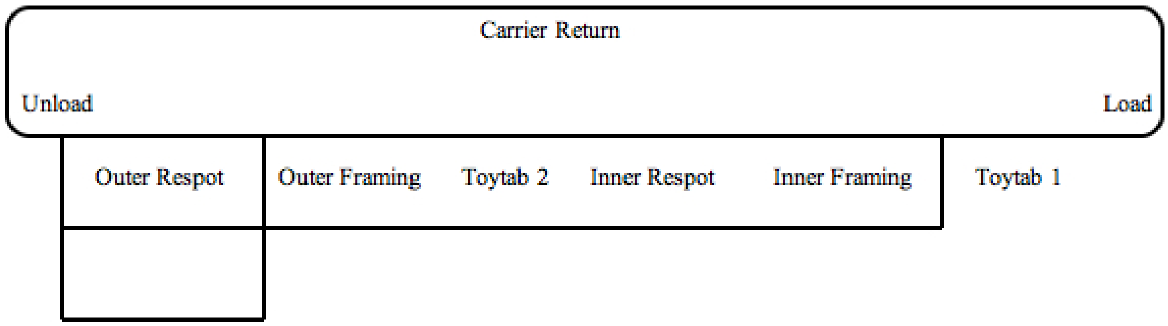

Figure 1. Transfer line

The line's stations are subject to failure. Each station is modeled with random failure and repair rates that are statistically independent from those of the other stations. The time-to-failure of each station is assumed to be exponential distributed while its time-to-repair is modeled with a gamma-2 distribution. The mean and variance of these distributions are obtained from actual production data collected from the stations in the framing line of an automobile assembly plant. Note that the mean and variance of these distributions vary from station to station. Details concerning the data used to construct these distributions are presented in (Hillberg, 2004).

The three prediction tools compared in this study use the line's Work-In-Progress (WIP) to forecast its throughput at the pay-point (located at the end of the line). The line is partitioned in several "areas" with the accumulation of WIP in each area being an input to the throughput prediction tool. The impact on the forecast accuracy of the number of stations in the line and the number of areas partitioning the line is also considered.

2. THROUGHPUT PREDICTION TOOLS

An overview of the throughput prediction tools is provided in this section. More details concerning these tools are presented in (Hillberg, 2004). Note for each prediction tool, the inputs are the WIP in the line's areas while the output is the line's throughput.

2.1 Neural Network Prediction Tool

In this study, several different neural network layouts were analyzed. The final network used for the throughput prediction tool has one input layer, one hidden layer and a single output node representing the throughput. The hyperbolic tangent sigmoidal transfer function is used for the nodes in the input and hidden layers while the output node's transfer function is linear. MATLAB's Neural Network Toolbox is used to implement the networks.

The number of input nodes is set equal to the number of areas which the line is partitioned, while the inputs for these nodes consist of each area's WIP. Following standard practice, the hidden layer contains 50% more neurons than the input layer, (Haykin, 1999).

Back propagation training using a Bayesian Regularization algorithm is used since it is the least sensitive to memorization. Details concerning the various neural network layouts and training methods studied, as well as an analysis of their forecasting performance, is presented in (Hillberg, 2004).

2.2 Statistical Regression Prediction Tool

The statistical regression prediction tool is designed to include both first order interactions and second order effects. No clear evidence was found over the course of this study to investigate higher order regressions (Hillberg, 2004). The method of least squares is used to calculate the regression coefficients.

2.3 Case-Based Reasoning Tool

While case-based reasoning is applied to many types of problems, its application to this study fits most closely with the classification methodology (Jain et al., 1999). When a new WIP data set, consisting of the WIP in each area, is presented to the prediction tool, WIP data sets in an experiential data-base are sorted based on their distance from this new data set. A finite number k of these data sets then "vote" for an appropriate classification value for the new data set. This method is known as "k-nearest neighbor" classification (Dasarathy, 1990) and (Jain et al., 1999).

One of the appeals of the case-based reasoning technique is its ability to use data "as-is". It is concerned only with the following two functions.

- A distance function which assigns a distance between the new data set and the experiential data sets.
- A combination function which adapts a finite subset of the "nearest" experiential data to generate a classification.

In this study, the distance function used is the Euclidean distance between the new data set and an experiential data set. The combination function uses a fuzzy set scoring method with a weighted mean, where the weight is derived using a sigmoidal membership function (Cox et al., 1997).

The case-based reasoning tool used over 600 sets of WIP data in its experiential data-base. In the course of this study, differing values for k (the number of nearest neighbors) were used. Little difference was seen in the efficacy of the prediction tool when between 20 and 50 neighbors were used. Hence, 30 neighbors were selected when performing the comparison tests presented in this paper.

Note an additional advantage of the case-based reasoning technique is its ability to adapt. New data sets are added to the experiential data-base as they occur without the need for re-training the prediction tool, as is required by the neural network and the statistical regression prediction tools. A disadvantage of the case-based reasoning technique is that it tends to consume a large amount of both processing time and computer memory (Hand et al., 2001). A large amount of experiential data must be available in order to find a reliable set of neighbors.

3. COLLECTION OF TRAINING DATA

Numerous experiments were performed using the transfer line described in Section 1, Figure 1. A simulation model of this line was implemented in Automod (Automod Users Manual, 2001). Approximately 2,000 records of data were collected as training sets for the predictive tools. Simulations were run for 8 replications of 40 hours past the warm-up period. The simulation model was reset to a common initial condition at the beginning of each replication. Two-thirds of the collected data was designated for training the predictive tools while the remaining third was used to test the training.

4. EVALUATION OF THE PREDICTIVE TOOLS

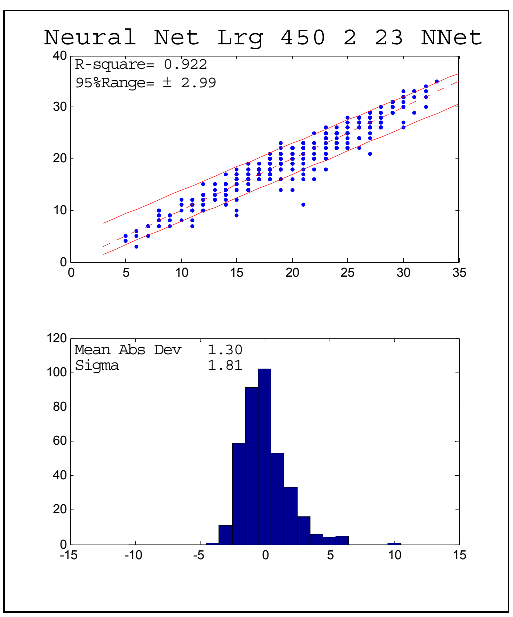

Figure 2. Evaluation of prediction tool performance

A procedure was developed for comparing the efficacy of each predictive tool, see Figure 2. The upper chart in the figure is a scatter plot of data collected from the simulation model. It is compared to the forecast from the prediction tool using the same set of inputs. A linear regression is calculated between the actual result from the simulation model and the forecast from the prediction tool. An R-squared value and a 95% range are also determined. The scatter plot shows the 95% confidence interval as two parallel lines. The lower chart in Figure 2 provides a histogram showing the distribution of the differences between the actual and forecast values. Also shown in the histogram plot are the mean absolute deviation and the standard deviation.

Next, performances of the three predictive tools are compared. Tables 1 and 2 show the performance of three predictive tools on data collected from the transfer line, Figure 1. The Kolmogorov-Smirnov variance test (Walpole et al., 2002) show a strong similarity between the residual distributions found in the regression and the neural network prediction tools. However, the performance of the case-based reasoning tool differed from that of the other two tools. The performance metrics such as R-squared, mean absolute deviation, etc. are worse for the case-based reasoning tool, so this tool is rejected as an adequate predictor. Anova and Kruskal-Wallis mean tests (Walpole et al., 2002), were then performed to recognize a difference in the residual averages of the regression and the neural network tools with no statistically significant difference between them being found, Table 2.

Table 1. Performance of prediction tools for small transfer line

Regression		**Neural Network**		**Case-Based Reasoning**	
R-Squared	0.871	R-Squared	0.883	R-Squared	0.819
95% Range	3.666	95% Range	3.488	95% Range	4.342
Mean Abs. Dev.	1.644	Mean Abs. Dev.	1.587	Mean Abs. Dev.	1.952
Mean Residuals	-0.074	Mean Residuals	-0.063	Mean Residuals	0.294
Conf. Interval	[-4.43, 4.29]	Conf. Interval	[-4.21, 4.08]	Conf. Interval	[-4.88, 5.46]

Table 2. Variance and means tests for small transfer line (P-values)

	Regression vs. NN	**Regression vs. CBR**	**NN vs. CBR**
Kolmogorov-Smirnov II	0.994	0.028	0.038
Anova	0.953		
Kruskal-Wallis	0.873		

4.1 Impact of Transfer Line Size on Forecast Accuracy

The number of stations in the transfer line has an impact on the ability of the prediction tools to accurately forecast throughput. It was found that as the line increases in size, the accuracy of the forecast improves for each of the prediction tools. This result was obtained by analyzing the performance of the prediction tools while varying the size of the transfer line.

Table 3 Performance of neural network prediction tool for different transfer line size

Small Line		**Medium Line**		**Large Line**	
R-Squared	0.758	R-Squared	0.872	R-Squared	0.879
95% Range	4.680	95% Range	3.648	95% Range	3.585
Mean Abs. Dev.	2.073	Mean Abs. Dev.	1.652	Mean Abs. Dev.	1.618
Mean Residuals	0.020	Mean Residuals	0.053	Mean Residuals	0.098
Conf. Interval	[-5.55, 5.59]	Conf. Interval	[-4.39, 4.28]	Conf. Interval	[-4.36, 4.16]

Table 4 Variance and means tests for different transfer line sizes (P-values)

	Small vs. Medium	**Small vs. Large**	**Medium vs. Large**
Kolmogorov-Smimoff II	0.067	0.101	0.940
Anova			0.403
Kruskal-Wallis			0.358

The smallest line contained 53 stations (the original line, Figure 1) while the medium and largest lines contained 75 and 100 stations, respectively. Note that in each case, the transfer line was divided into 10 areas. The general configuration of the transfer line (a single line feeding two parallel lines which feed three parallel lines finally merging back into a single line) was unchanged.

Since the neural network prediction tool produces an adequate forecast, it is used to determine the impact of transfer line size on forecast accuracy. The results are shown in Tables 3 and 4. The distribution of the residuals in Table 3 indicates that there is a difference in the forecast accuracy due to the size of the transfer line. It shows the performance metrics improve as the line gets larger.

The Kolmogorov-Smirnov variance test indicates a statistically significant difference in the small versus the medium and the small versus the large lines at 10% level of significance. The mean residual values are all near zero, indicating that the predictive errors do not include a bias. The Anova and the Kruskal-Wallis means tests indicate no substantial difference in the mean residual of the medium versus large lines. Since the residual distribution of the small line could not be considered the same as the other two, it was not included in the means tests.

4.2. Impact of the Number of Areas on Forecast Accuracy

In the preceding analysis, the transfer line was partitioned into ten areas. In this section, the effect of increasing the number of areas on forecast accuracy is considered. This is easiest to accomplish on the large transfer line which contains 100 stations. The performances of the three prediction tools created using the large line divided into 10 areas are compared with three corresponding prediction tools created with the line divided into 23 areas.

Table 5 Performance of regression prediction tool for different areas

10 Areas		23 Areas		Notes	
R-Squared	0.842	R-Squared	0.85151		
95% Range	4.244	95% Range	4.13131	Likelihood of same underlying	
Mean Abs. Dev.	1.844	Mean Abs. Dev.	1.82777	distribution is 72.6% from	
Mean Residuals	0.061	Mean Residuals	0.06767	Kolmogorov-Smirnov II test.	
Conf. Interval	[-4.98, 5.10]	Conf. Interval	[-4.85, 4.988]		
				Likelihood of same mean.	
				Anova:	96.7%
				Kruskal-Wallis:	61.9%

The results are shown in the Tables 5, 6 and 7. In each case, the forecast time is 30 minutes. The notes adjoining each table indicate the likelihood that the residual distributions and means are the same, indicating whether there is (or is not) an advantage in dividing the line into smaller sized areas. In the case of the regression and the neural network prediction tools, no statistically significant difference in forecast accuracy was found.

Increasing the number of areas decreases the forecast accuracy for the case-based reasoning prediction tool. This is most likely due to the distance heuristic used within the case-based reasoning algorithm. In this algorithm, the Euclidean distances from the current system state to all previously recorded states are calculated to define the group of nearest neighbors. This distance function makes no allowance for the physical proximity of one area of the line to another area. Hence, if a large number of small areas are used, it is more likely to result in a larger distance between two system states than when a small number of large areas are used.

For example, consider a transfer line containing 100 stations partitioned into five areas with 20 stations per area. A part moving from station 5 to station 6 will not change the WIP count for area 1. Next, consider the same 100 stations partitioned into twenty areas with 5 stations per area. A part which moves from station 5 to station 6 will change areas, and there will be an impact on the WIP count in the distance function. The difficulty lies in that this change generates exactly the same distance value as a change between station 5 and station 100, which is much further away. So while a change between adjoining areas may not truly imply a significant change in the system state, the distance function considers it to be identical to a much more significant state change.

Table 6 Performance of neural network prediction tool for different areas

10 Areas		23 Areas		Notes	
R-Squared	0.8511	R-Squared	0.8522		
95% Range	4.1200	95% Range	4.1099	Likelihood of same underlying	
Mean Abs. Dev.	1.7855	Mean Abs. Dev.	1.7666	distribution is ~100% from	
Mean Residuals	0.0533	Mean Residuals	0.0200	Kolmogorov-Smirnov II test.	
Conf. Interval	[-4.84, 4.95]	Conf. Interval	[-4.86, 4.90]		
				Likelihood of same mean.	
				Anova:	79.9%
				Kruskal-Wallis:	76.3%

Table 7 Performance of case-based reasoning prediction tool for different areas

10 Areas		23 Areas		Notes
R-Squared	0.8244	R-Squared	0.7933	
95% Range	4.4811	95% Range	4.8600	Likelihood of same underlying distribution is 5.1% from Kolmogorov-Smirnov II test.
Mean Abs. Dev.	2.0255	Mean Abs. Dev.	2.2966	
Mean Residuals	0.0066	Mean Residuals	0.3466	
Conf. Interval	[-5.34, 5.33]	Conf. Interval	[-6.17, 5.47]	Means tests were not performed since underlying distributions are not similar.

4.3 Effects of Additional Buffer Capacity on Forecast Accuracy

In the systems described, buffering is implemented simply by designating that some stations require no processing time. The large system, which has 100 stations, was used to determine if the relative amount of buffering in the system has an impact on predictive ability. The model was executed in three experiments, where the difference between the models is in the number of buffer stations in the system. Zero, seven and twenty buffer stations were used in the three experiments; as in earlier examples, these values were chosen arbitrarily. The criteria being that they fit easily into the structure of the system, and that there is a proportionally large difference from one experiment to the next. The number of stations in the system remains at 100, but in the cases of seven and twenty buffers, evenly dispersed stations were given zero processing times.

Table 8 shows the results of the experiments. The performance metrics (R-square, 95% Range, etc.) indicate a loss of predictive ability as the number of buffer stations increase. While the variance tests cannot conclude that there is a significant difference between none and 7 buffer stations, or between 7 and 20 buffer stations, it can conclude (with 97.6% certainty) that there a statistically significant difference between none, and 20 buffer stations. Presumably, if enough data is collected, a significant difference between none and seven buffer stations may be found as well.

Adding small capacity buffer stations to the system increases its cycle time variability. A buffer station can more quickly change from full to empty than can a processing station, and this increase in variability decreases predictive ability. For example, consider a data collection area of five stations. At a particular point in time it is observed that both the fourth and the fifth stations in the area are holding parts. The other stations are empty, and the WIP count for this area is two (a part in each of stations 4 and 5).

Suppose there is no buffer in this area and the parts are blocked while waiting on a downstream station. Once the downstream blockage clears, the part in station 5 exits while the part in station 4 moves into station 5. Hence, the WIP becomes one (a 20% drop in the maximum WIP count for this area). Barring any new part entries into the area, WIP will remain at one at least until station 5 has completed processing the part.

Table 8 Performance of neural network prediction tool for different buffer capacity

No Buffers		Buffer Capacity 7		Buffer Capacity 20	
R-Squared	0.889	R-Squared	0.865	R-Squared	0.823
95% Range	3.436	95% Range	3.893	95% Range	4.416
Mean Abs. Dev.	1.508	Mean Abs. Dev.	1.720	Mean Abs. Dev.	1.943
Mean Residuals	-0.032	Mean Residuals	-0.004	Mean Residuals	0.029
Conf. Interval	[-4.11, 4.05]	Conf. Interval	[-4.63, 4.62]	Conf. Interval	[-5.27, 5.21]

Table 9 Variance and means tests for different buffer capacity (P-values)

Kolmogorov-Smirnov II Variance Tests:	P-values	Means Tests using:	P-values
None: 7 Buffers	0.682	Anova	0.802
None: 20 Buffers	0.024		
7: 20 Buffers	0.506	Kruskal-Wallis	0.618

Next, consider the same scenario, but with station 5 a buffer station. When the downstream blockage clears, the part in station 4 may be able to move immediately past station 5 (a buffer) and into the next area. The WIP count has now changed nearly instantaneously from two parts to none (a 40% drop in the maximum WIP count), twice what was seen in the previous scenario. This larger change increases the variability of the input data, which is reflected in the increased variability of the output data, and results in the negative impact on predicting throughput.

It appears that this may be a fruitful area for further optimization. In collecting the WIP count, no special allowances are made to identify how the WIP is distributed between operational and buffer stations. If the WIP count was enhanced to incorporate this information, a better prediction may be observed.

5. CONCLUSIONS AND PRACTICAL ASPECTS

Three data-driven modeling techniques were studied for constructing a throughput prediction tool: neural networks, statistical regression and case-based reasoning. Results of this comparison indicate that the prediction tools designed using statistical regression and neural network techniques offered comparable performance. It was also observed that the prediction tools constructed using these two techniques performed better than the one designed using case-based reasoning techniques.

Note that to successfully use these techniques to construct a prediction tool, a large set of input-output data reflecting the manufacturing system's performance is required. This set of input-output performance data can be referred to as historical performance data. Unfortunately for many modern manufacturing systems, a sufficiently large set of such historical performance data required to use these data-driven modeling techniques may not be available.

There are two main reasons for this shortage of data. First, the manufacturing system may be too new to have accumulated a large enough set of historical performance data.

A more important reason for an insufficient amount of historical performance data is due to the continuous improvement process that is applied to most manufacturing systems (Askin et al., 2002). Most manufacturing systems are modified on a weekly, if not daily, basis. Hence, the current behavior of a manufacturing system is usually not reflected in the input-output performance data collected during the previous month due to system modifications.

A potential solution to this problem is to use a verified simulation model of the manufacturing system to produce a set of simulated historical performance data. This set of simulated historical performance data could be used to train the throughput prediction tool via one of the data-driven modeling techniques. Once the throughput prediction tool is trained and verified, it will use real-time performance data collected from the manufacturing system in order to make its prediction. This is the subject of current research efforts.

A prevalent theory at DaimlerChrysler during the course of this research project was "the largest impediment to throughput is machine reliability". It was possible to identify the least reliable machines as having the lowest throughput, and place buffers upstream and downstream of that machine. However, this usually has little effect in a large system (dozens to hundreds of machines). The machine with the lowest reliability is generally only slightly worse than a number of other machines in the system. It is impractical, and usually impossible, to place buffers around each of these unreliable machines. What's more, this would violate one of the primary tenants of lean manufacturing, as it would increase WIP in order to minimize the effects of machine unreliability.

One approach to maximize reliability is the construction of intelligent machines that can identify their own pending failure (Higgs et al., 2004) and (Bengtsson, 2008). Once a possible failure condition is recognized, the machine would signal the appropriate personnel that preventive maintenance should be performed.

But an intelligent machine cannot determine an opportune time for conducting preventive maintenance unless it understands the state of the system in which it resides. An optimal time to conduct preventive maintenance is when the machine is not expected to be busy, that is, when it's expected throughput is low. By predicting periods of low throughput, the machine can be maintained at a time in which it would otherwise be starved for parts. This effectively synchronizes preventive maintenance to the throughput gaps which 'naturally' occur in the production system.

6. ACKNOWLEDGEMENTS

Funding was provided by the Advanced Manufacturing Engineering group at DaimlerChrysler Corp. and the National Science Foundation's GOALI Program under grant DMI-0228828.

7. REFERENCES

1. Askin, R.G. and Goldberg, J.B. (2002). Design and Analysis of Production Systems. John Wiley and Sons, New York, New York.
2. Automod User's Manual (2001). Version 10.0, Vol. 1. Brooks Automation Inc., Bountiful, Utah.

3. Bengtsson, M. (2008). Supporting Implementation of Condition Based Maintenance: Highlighting the Interplay Between Technical Constituents and Human and Organizational Factors. International Journal of Technology and Human Interaction, 4: 48-74.
4. Cox, J.F. and Blackstone, J.H. (1997). Managing by the Theory of Constraints. In Handbook of Production and Inventory Control (Ed.: J.H. Green). APICS, Falls Church, VA, 3rd ed., chapter 9.
5. Dasarathy, B. (1990). Nearest Neighbor Norms: NN Pattern Classification Techniques. IEEE Computer Society Press, Los Alamitos, CA.
6. Degbotse, A.T. and Nachlas, J.A. (2003). Use of Nested Renewals to Model Availability under Opportunistic Maintenance Policies. Proceedings of the Annual Reliability and Maintainability Symposium, pp. 344-350.
7. Hagan, M. T., Demuth, H. B. and Beale, M. (1996). Neural Network Design. PWS Publishing Co., Boston, Massachuetts.
8. Han, J. and Kamber, M. (2000). Data Mining: Concepts and Techniques. Morgan Kaufmann, San Francisco, California.
9. Hand, D., Mannila, H. and Smith, P. (2001). Principles of Data Mining. MIT Press, Cambridge, Massachuetts.
10. Haykin, S. (1999). Neural Networks: A Comprehensive Foundation. Prentice Hall, Upper Saddle River, New Jersey.
11. Higgs, P.A., Parkin, P., Jackson, M., Al-Habaibeh, A., Zorriassatine, F. and Coy, J. (2004). A Survey on Condition Monitoring Systems in Industry. Proceedings of the ASME Conference on Engineering Systems Design and Analysis, 3: 163-178.
12. Hillberg, P.A. (2004). Improving Manufacturing Productivity through Opportunistic Preventive Maintenance via Simulation and Analytical Modeling. Ph.D. Dissertation, Oakland University, Rochester, Michigan.
13. Hsu, S., Yang, C.-L. and Ching, C.Y. (2004). Applying Genetic Algorithms to a Large-Scale Flow Shop Scheduling Problem. International Journal of Industrial Engineering, 11: 288-296.
14. Jain, A.K., Murty, M.N. and Flynn, P.J. (1999). Data Clustering: A Review. ACM Computing Surveys, 31: 264-323.
15. Jang, R., Sun, T. and Mizutani, E. (1997). Neuro-Fuzzy and Soft Computing. Prentice Hall, Upper Saddle River, New Jersey.
16. Kwak, C. and Chang, O.K. (2005). Dispatching Decisions within Preemption Procedure. International Journal of Industrial Engineering, 12: 23-27.
17. Lee, Z.-H. (2007). A Practical Scheduling Rule for Due Date Quotation. International Journal of Industrial Engineering, 14: 90-99.
18. Lenz, M., Bartsch-Sporl, B., Burkhard, H. and Wess, S. (Editors) (1998). Diagnosis and Decision Support. Springer-Verlag, New York, New York.
19. Moon, C. and Lee, Y.H. (2003). Advanced Process Planning and Scheduling with Precedence Constraints and Machine Selection Using a Genetic Algorithm. International Journal of Industrial Engineering, 10: 26-34.
20. Neter, J., Kutner, M., Wasserman, W. and Nachtsleim, C.J. (1996). Applied Linear Statistical Models. Irwin, Chicago, Illinois.
21. Rosenblatt, F. (1962). Principles of Neurodynamics: Perceptrons and the Theory of Brain Mechanisms. Spartan, Washington.
22. Soroush, H.M. and Allahverdi (2005). Stochastic Two-Machine Flowshop Scheduling Problem with Total Completion Time Criterion. International Journal of Industrial Engineering, 12: 159-171.
23. Walpole, R.E., Myers, R.H. and Myers, S.L. (2002). Probability and Statistics for Engineers and Scientists. Prentice Hall, Upper Saddle River, NJ.

BIOGRAPHICAL SKETCH

Pat Hillberg has close to 20 years of industry experience in the field of robotics, machine vision, and manufacturing productivity. He has designed and implemented numerous machine vision and robotic factory-floor applications for a variety of industries. He was most recently a Senior Engineer in Product Development for Vision-Guidance for FANUC Robotics, North America. He was instrumental in the development of FANUC's visLOC product.

Pat received his Bachelor's degree in Computer Science from Michigan Technological University, and his Master's degree in Industrial & Systems Engineering from the University of Michigan. He is working on fellowship towards a Ph.D. in Systems Engineering at Oakland University, and expects to present his defense in the winter of 2004. His research topic includes finding productivity improvements through Automatic Data Collection, Discrete Event Simulation, and Data-Mining techniques within a manufacturing environment.

Sankar Sengupta currently is a faculty member in the Electrical and Systems Engineering Department at Oakland University. He earned a B.S degree in mechanical engineering from I.I.T. Kharagpur in India, an M.S degree in industrial engineering from the University of Oklahoma, and a Ph.D. in industrial engineering from Clemson University. He has more than 12 years of work experience in the process industry. His research interest includes modeling and analysis of manufacturing/service systems, quality control, application of lean principles in a manufacturing/service system, production planning and control, and modeling and analysis of a supply chain. He has published technical papers in refereed journals and presented technical papers in international conferences. He is actively involved with industry to establish a strong tie between OU and local partners. He is a member of IIE, ASQ and SME.

Robert Van Til is currently a professor in the Industrial and Systems Engineering Department at Oakland University. He earned a B.S degree in mechanical engineering from Michigan State University, an M.S degree and a PhD in mechanical engineering from the Northwestern University. His research interest includes modeling and analysis of manufacturing/service systems, quality control, application of lean principles in a manufacturing/service system, production planning and control, and modeling and analysis of a supply chain. He has published technical papers in refereed journals and presented technical papers in international conferences