Prediction of Inventory Levels and Capacity Utilization with Artificial Neural Networks

BERND SCHOLZ-REITER, FLORIAN HARJES, AMIR KAVIANI MEHR
BIBA – Bremer Institut für Produktion und Logistik GmbH at the University of Bremen
University of Bremen
Hochschulring 20, 28359 Bremen
GERMANY
{bsr, haj}@biba.uni-bremen.de, amir.kavianimehr@uni-bremen.de http://www.biba.uni-bremen.de/

Abstract: - Coping with increasingly complex production processes requires a continuous advancement of production control techniques. In this context, artificial neural networks have proven their potential in optimization, prediction, classification, control and other production related areas. This paper presents an approach for the workstation-specific prediction of inventory levels and capacity utilization within a shop floor environment. This includes the selection of the appropriate network architecture, the determination of suitable input variables as well as the training and validation of the applied neural networks. Further, an evaluation of the proposed networks takes place by means of a generic shop floor model.

Key-Words: Artificial intelligence, artificial neural networks, Elman networks, prediction, shop floor production, predictive control, inventory, capacity utilization

1 Introduction

Multi variant and customized products with short lifecycles are typical for today`s market [*1*]. The corresponding production processes and material flows are often complex and dynamic. Consequently, established production planning and control approaches need a continuous advancement [*2*].

Particularly in the field of shop floor production, prototypes and small series as well as the special technical organization complicate the handling of control related tasks. At this point, artificial neural networks have proven their applicability as methods for classification, pattern recognition or production control [*3*], [*4*].

This paper introduces an approach of a neural network based prediction of inventory levels and capacity utilization for workstations within a shop floor environment. The approach can be seen as a contribution to the development and implementation of innovative decentralized and/or predictive control strategies [*5*].

The next section introduces neural networks in general, followed by a brief description of the newly developed neural predictors regarding their structure and training results in section 3. Section 4 presents the shop floor model for the evaluation of the new predictors and the obtained experimental results. Finally, the article closes with a summary and an outlook on future research in section 5.

2 Artificial Neural Networks

Artificial neural networks emulate the structure and functionality of neural systems in nature [*6*]. They typically consist of nodes, which are arranged in at least two or more layers and are interconnected via weighted links [*7*] (Fig. 1). At this point, the number of layers and the direction of the connections depend on the type of network [*8*].

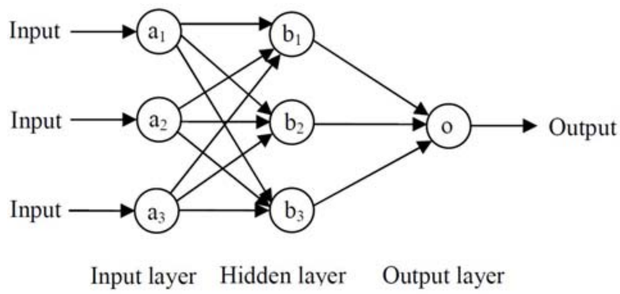

Fig. 1 Example of a neural network

Neural networks offer a fast data processing, a comparatively small modelling effort and the ability to learn from experience [*9*]. Further, they are able to approximate complex mathematical coherences that are either unknown or not completely describable. At this point, neural networks act in a black box manner [*10*].

Depending on the type of neural network, three general learning procedures can be distinguished. Supervised Learning denotes a procedure, where pairs of input and output data are presented to the neural network. During the learning process, the network adapts its connection weights, so that the input leads to the desired output [***8***]. Reinforcement Learning only comprises the presentation of input data. Instead of the corresponding output, the network receives a feedback, whether the output was correct [***6***]. Finally, Unsupervised or Self-Organized Learning takes place without any default values for the output or the corresponding feedback. At this point, the neural network tries to recognize patterns within the input data autonomously [***11***].

Common for all approaches is the validation of the learning results with a second dataset. This ensures the generalization of the learning process and avoids a mere memorization of the training data, the so called Overfitting [***6***].

3 The Neural Predictors

3.1 Elman Networks

As mentioned above, the structure of a neuronal network strongly depends on the application area. For prediction purposes, recurrent or partly recurrent architectures are common [***12***]. The approach presented in this paper focuses on Elman networks, a partially recurrent network architecture [***13***]. Elman networks are feedback networks, containing a special layer of so called context cells (see Fig. 2).

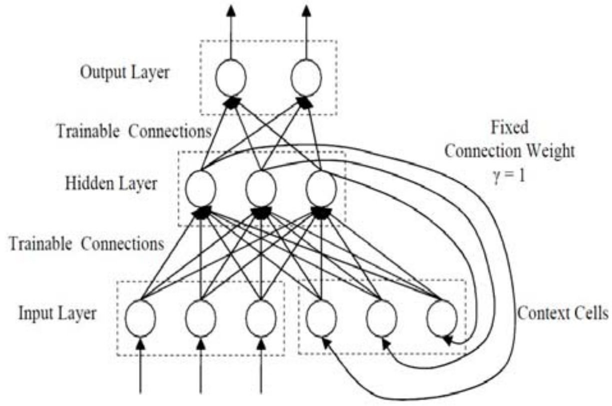

Fig. 2 Elman Network (following [***14***])

These context cells save the neural activation of previous states and therefore ensure that the prediction takes past events into account. Thus, the connection weight between the hidden layer and the context cells determines how much past states influence the prediction. A connection weight near or equal to 1 stands for a strong influence of past states, a smaller value mitigates this effect.

3.2 Structure of the Neural Predictors

The proposed concept comprises the workstation-specific prediction of inventory level and capacity utilization. For this purpose, the neural networks consider the actual state of the regarded workstation as well as the conditions of the predecessors. Correspondingly, the predictor networks` topology depends on the position, the considered workstation has within the material flow.

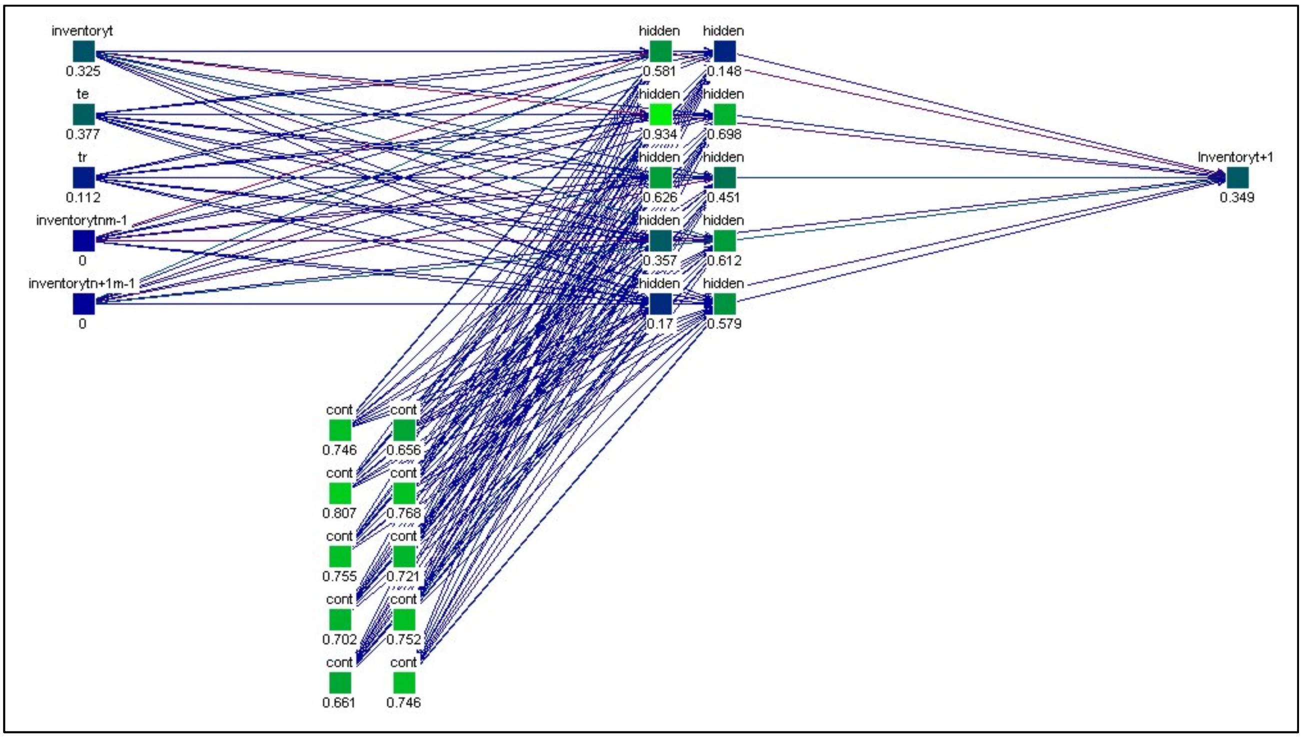

Fig. 3 Topology of the inventory predictor (screenshot)

ISBN: 978-1-61804-031-2

In the following, a workstation with two predecessors serves as an example. The neural predictor for the inventory level is a 5:10:10:1 Elman Network (Fig.3). It processes 5 input values, these are:

1. The actual inventory level of workstation *n*, manufacturing stage *m* at time *t* (Inventory $(t)_{n,m}$),
2. the machining time ($te_{n,m}$) and
3. the setup time ($tr_{n,m}$)of all orders waiting in front of the workstation,
4. the actual inventory level of predecessor *n*, production stage *m-1* at time *t* (Inventory$(t)_{n,m-1}$),
5. the actual inventory level of predecessor *n+1*, production stage *m-1* at time *t* (Inventory $(t)_{n.m-1}$.

The output value of the network represents the predicted inventory level at time *t+1*. At this point, the prediction horizon amounts four hours, depending on the shift plan of the underlying shop floor model.

The capacity predictor has a quite similar 4:10:10:1 topology. While the number of hidden neurons and context cells is identical, the network needs only four input neurons. These neurons process the following values:

1. The capacity of workstation *n*, production stage *m* at time *t* (Capacity $(t)_{n,m}$),
2. the occupancy of workstation *n*, production stage *m* at time *t* (Occupancy $(t)_{n,m}$),
3. the current inventory level of workstation *n*, production stage *m* at time *t* (Inventory $(t)_{n,m}$) and
4. the waiting time of workstation *n*, production stage *m* at time *t* (Waiting $(t)_{n,m}$).

At this point, capacity defines the maximum number of workpieces that can be produced within the prediction horizon of 4 hours (half a work shift). The determination of the corresponding period length is described in section 4. Finally, the waiting time denotes the amount of time, the workstation pauses due to disturbances, breaks, etc.

3.3 Training and Validation

The initial training and validation process of both network types bases on the supervised learning method using the Resilient Propagation algorithm. Experiments with Quick Propagation and Backpropagation with Momentum term show inadequate results.

The necessary datasets result from test runs of the shop floor model that is also used for evaluation purposes in the next section. The test runs take approximately 30 days with an average of 1770 orders. At this point, the recording of input/output pairs takes place every four hours. Fig. 4 depicts the learning curve of the network for capacity prediction. The lower line represents the results (summed square error) of the training dataset, while the upper line denotes the same for the validation data. The training process converges after approximately 700 cycles, when both curves reach their minimum. A further training would lead to an increasing error for the validation data and a slight improvement for the initial training set. This is a typical indication for an overfitting of the neural network [***15***].

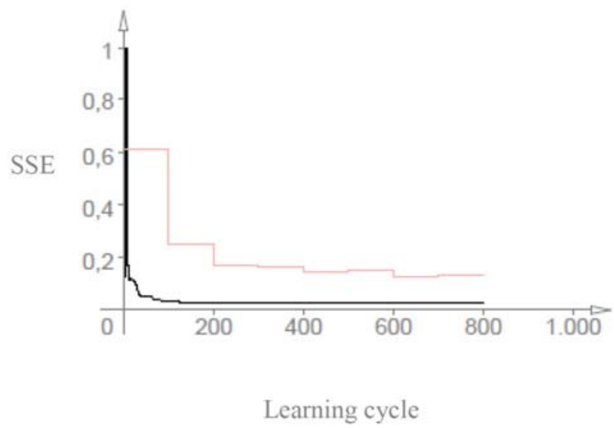

Fig.4 Learning process of the capacity predictor

The minimal error during the training process is less than 0,1 (1≈100%). Transferred to the original prediction task, this implies an average prediction error of approximately 5%.

The learning process of the inventory predictor converges after approximately 400 cycles (Fig. 5).

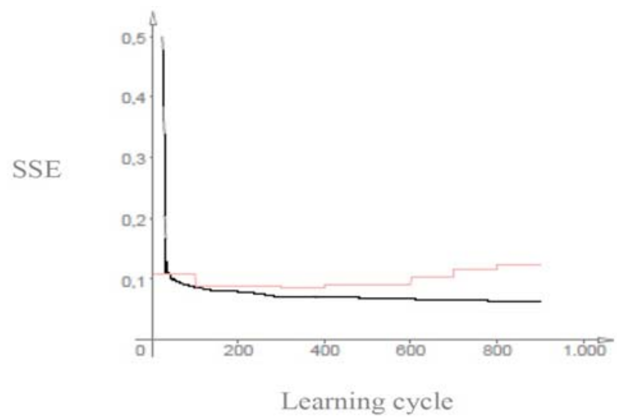

Fig. 5 Learning process of the inventory predictor

ISBN: 978-1-61804-031-2

At this point, the minimal error is again less than 0,1, but slightly higher than the capacity predictor`s result.

4 Experiments

4.1 Settings

The evaluation of the neural predictors takes place by means of a generic shop floor model. The model comprises eight workstations on four production stages (Fig 6). Every workstation has an input buffer in front of it. At this point, the workpieces pass the buffer following the FIFO principle (First-In-First-Out). The shop floor operates in three shifts of eight hours each. To enable a quick reaction to changing production situations, the prediction horizon is set to the half of a shift (four hours).

During the simulated period of 30 days, six different workpiece types run through the shop floor. The order release takes place piecewise the setup and processing times differ for every type of workpiece, depending on the technical properties of the workstations. Hence the processing and setup times are in the range of one up to 40 minutes.

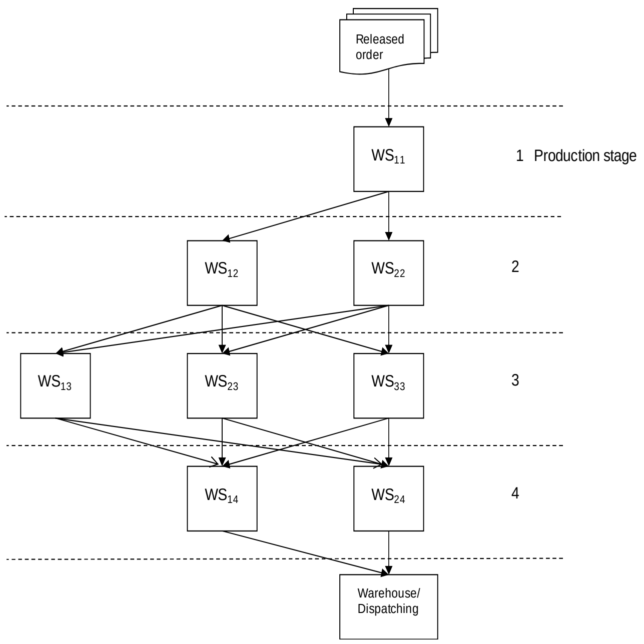

Fig. 6 Layout of the shop floor model

The processing order is sequential, so that every workpiece passes all four production stages. The distribution of workpieces between the production stages follows an inventory based control approach. A finished workpiece is always transferred to the successor at the following production stage with the comparatively lowest inventory level.

4.2 Results

In the following, the prediction results of workstation$_{13}$ serve as an example for the whole shop floor. Fig. 7 depicts the comparison between the actual and the predicted capacity utilization for this workstation over a period of 20 hours. This timeframe contains five predictions with a horizon of four hours each. At this point, the curve for the actual values represents continuously recorded data. The prediction curve depicts an approximation between the performed five predictions. This results in a relatively uneven curve shape.

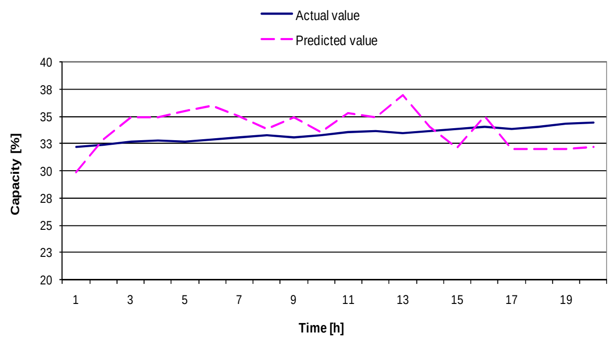

Fig. 7 Actual and predicted capacity utilization for WS_{13}

The evaluation shows an average workload scarcely above 34%. The time of inactivity is attributable to disturbances, breaks, setup times and maintenance. The predicted capacity utilization is close to the actual data, with a deviation of 3,2% maximum(Fig.8).

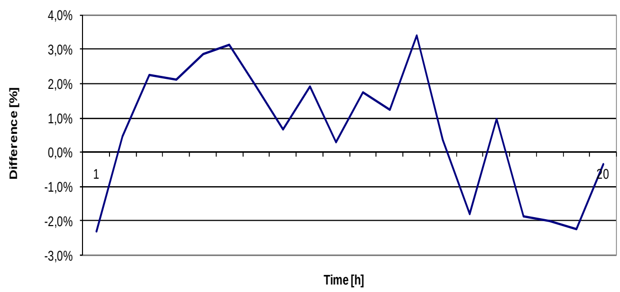

Fig. 8 Deviation of the prediction error for the inventory levels

The course of the inventory prediction is quite similar, with an error between nearly zero and a maximum of approximately 6% (Fig. 9). As it is for

ISBN: 978-1-61804-031-2

the capacity prediction, the actual values represent continuous and event-oriented data. In contrast, the predicted values depict an approximation of the inventory development.

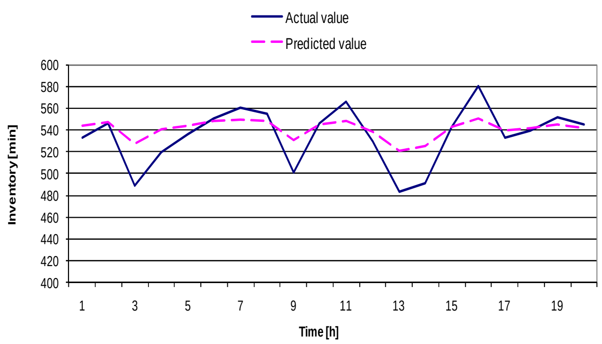

Fig. 9 Actual and predicted inventory level for WS_{13}

The predicted values differ from the real inventories averagely 2.5% (Fig. 10). Nevertheless, the prediction deviates up to 40 minutes from the recorded inventory level. Due to the setup and processing times, deviation can correspond to 1-4 workpieces.

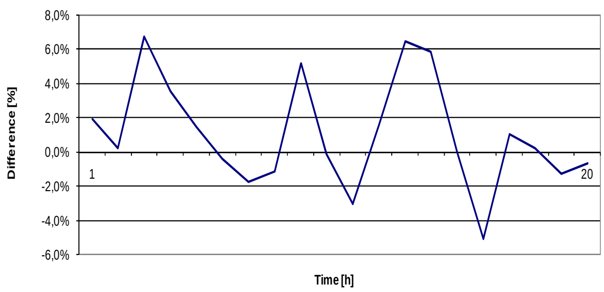

Fig. 10 Deviation of the prediction error for the capacity utilization

5 Summary and Outlook

This paper introduces an approach for the workstation-specific prediction of capacity utilization and inventory levels using Elman networks. The experimental results render a low monadic prediction error with a maximum of 6% for a prediction horizon of four hours. This is sufficient in the case of capacity utilization. For the inventory levels, an even more precise prediction is desirable. At this point, the deviation between the real and predicted values can correspond to multiple workpieces.

Therefore, future research should focus on the reduction of prediction errors in coordination with an increase of the prediction horizon. Another point of interest should be the integration of the introduced prediction approach into modern production control strategies, e.g. Model Predictive Control (MPC).

In the field of neural network research there is a fundamental interest in making continuous adaptations to changing shop floor situations, such as shifting setup- and processing times and the varying number of workpiece types.

Acknowledgement

This research is funded by the German Research Foundation (DFG) as part of the project “Automation of continuous learning and examination of the long-run behaviour of artificial neural networks for production control”, index SCHO 540/16-1.

References:

[1] J. Barata and L. Camarinha-Matos, "Methodology for Shop Floor Reengineering Based on Multiagents," in *IFIP International Federation for Information Processing - Emerging Solutions for Future Manufacturing Systems*, L. Camarinha-Matos, Ed. Boston: Springer, 2005, vol. 159, pp. 117-128.

[2] W. Schäfer, R. Wagner, J. Gausemeier, and R. Eckes, "An Engineer’s Workstation to Support Integrated Development of Flexible Production Control Systems," in *Integration of Software Specification Techniques for Applications in Engineering*, vol. 3147/2004, Berlin Heidelberg, 2004, pp. 48-68.

[3] B. Scholz-Reiter, F. Harjes, J. Mansfeld, T. Kieselhorst, and J. Becker, "Towards a Situation Adaptive Shoop Floor Production," in *Proceedings of the Second International Conference on Business Sustainability 2011*, Guimarães, Porto,

2011, pp. 1-8.

[4] B Scholz-Reiter, T Hamann, H Höhns, and G. Middelberg, "Decentral Closed Loop Control of Production Systems by Means of Artificial Neural Networks," in *Proceedings of the 37th CIRP - International Seminar on Manufacturing Systems*, Budapest, Hungary, 2004, pp. 199-203.

[5] B. Scholz-Reiter and T. Hamann, "The behaviour of learning production control," *CIRP Annals - Manufacturing Technology*, vol. 7, no. 1, pp. 459-462, 2008.

[6] S. Haykin, *Neural Networks and Learning Machines (3rd Edition)*. New Jersey, USA: Prentice Hall, 2008.

[7] W-H. Steeb, *The Nonlinear Workbook: Chaos, Fractals, Neural Networks, Genetic Algorithms, Gene Expression Programming, Support Vector Machine, Wavelets, Hidden Markov Models, Fuzzy Logic with C++, Java and SymbolicC++ Programs*, 4th ed. Singapore: World Scientific Publishing Co. Pte. Ltd, 2008.

[8] D.K. Chaturvedi, "Artificial neural networks and supervised learning," in *Soft Computing: Techniques and its Applications in Electrical Engineering*. Berlin Heidelberg: Springer, 2008, pp. 23-50.

[9] G. Dreyfus, *Neural Networks Methodology and Application*. Berlin Heidelberg: Springer Verlag , 2005.

[10] D. Rippel, F. Harjes, and B. Scholz-Reiter, "Modeling a Neural Network Based Control for Autonomous Production Systems," in *Artificial Intelligence and Logistics (AILog) Workshop at the 19th European Conference on Artificial Intelligence 2010*, Amsterdam, 2010, pp. 49-54.

[11] T. Kohonen, *Self-Organizing Maps*, 3rd ed. New York: Springer, 2001.

[12] D. Mandic and J. Chambers, *Recurrent Neural Networks for Prediction: Learning Algorithms, Architectures and Stability (Adaptive and Learning Systems for Signal Processing, Communications and Control Series)*. Hoboken, USA: Wiley-Blackwell, 2001.

[13] J.L. Elman, "Finding structure in time," *Cognitive Science*, vol. 14, no. 2, pp. 179-211, 1990.

[14] W-M. Lippe, *Soft-Computing mit Neuronalen Netzen, Fuzzy-Logic und Evolutionären Algorithmen*. Berlin: Springer, 2006.

[15] S. Lawrence and C.L. Giles, "Overfitting and neural networks: conjugate gradient and backpropagation," in *Proceedings of the IEEE-INNS-ENNS International Joint Conference on Neural Networks. IJCNN 2000. Neural Computing: New Challenges and Perspectives for the New Millennium*, vol. 1, Como, Italy, 2000, pp. 114-119.

ISBN: 978-1-61804-031-2