IMPLEMENTING AUTONOMOUS PRODUCTION IN COMPLEX JOB-SHOPS WITH DEEP NEURAL NETWORKS

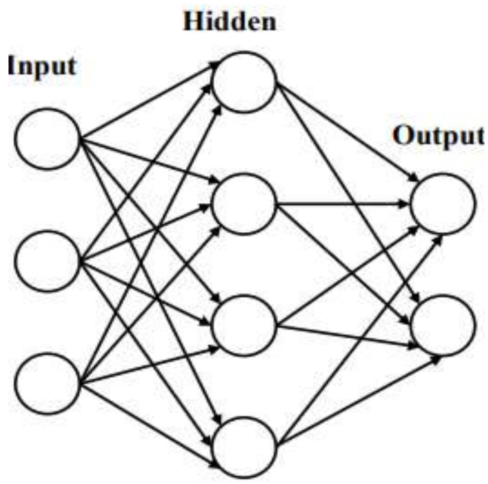
[1]Dr.Kalyanapu Srinivas,[2]Rakesh Datta Vadicherla,[3]Ranjith Kumar Kaata
[1]Professor,[2,3]Assistant Professor
Department of CSE
Vaagdevi Engineering College, Bollikunta, Khila Warangal, Warangal, Telangana, India

ABSTRACT
In the modern day, Deep Neural Networks (DNN) present fresh possibilities for handling more complicated production processes. In order to create an autonomous job shop, a process-based association and discrete event simulation are used to train DNN in this paper. This intelligent system makes the best choices possible while functioning effectively in a challenging setting with time constraints. From this point forward, the production control process uses the DNN in conjunction with time constraints. The system is commonly installed on intricate job shops of a wafer fab case in semiconductor production sectors. The DNN is trained to function in a complicated context with time limitations, ensuring that it completes tasks accurately and without errors. The DNN favours the pick that operates with the most critical batch list and has less time constraints. Accordingly, the study demonstrates that the DNN outperforms the conventional benchmark technique in handling the timing constraints.
Keywords: deep neural networks, job shops, scheduling control, and time limitations.

I. INTRODUCTION

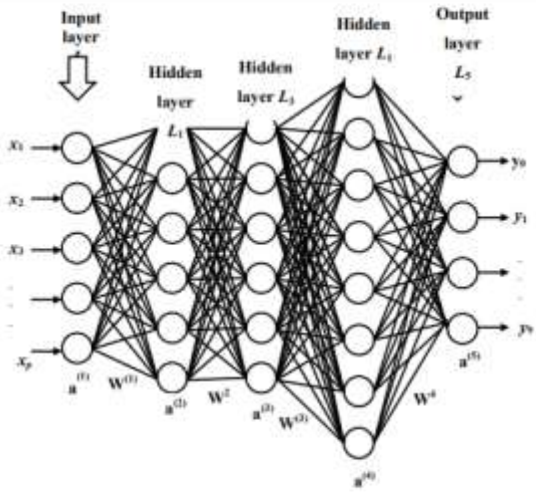
Production companies are constantly undergoing internal processes and environmental transformations [1]. Emerging industries are already well-known for globalization and competition in developing nations [2]. Specifically, accelerating industrial markets are continuing to becomeunpredictable with digitalization and hence the companies react decisively and quickly [3]. Under such circumstances, it is extremely vital to optimize the operational resources and capabilities.Complex workshops often face similar challenges associated with operational resources and capabilities in case of a semiconductor industry [4]. In various other industries, the job shops are widely assessed as it creates demands for flexible and hence there is a rapid increase in its variable production systems [5]. Therefore, opportunities, complexity and pressure continues makes use of the changeable production systems and considered as essential to secure a competitive position [4].Methods on artificial intelligence like deep learning, the development of deep neural networks and other machine learning approaches have the potential over new quantitative methods to meet these challenges. Such methods are supported by significantly decrease in its computational times, easy-to-use libraries and achievements such as the over-performing the decisions made by human experts in case of strategic games [6].Production planning and control (PPC) focus entirely on productionorganisations and it carries out the process of optimization of the internal processes [7]. PPC defines the productionprogrammesthat includes the process of production information. Production controls take into concern the predetermined inputs in all necessary processes that completes the production plan and, in view of changing circumstances such as machinery, material shortages or workforce. These changing circumstances make optimal utilization of available factors of production [8, 9]. The release of orders is a hence considered as a link between planning production and control of production.The dispatch of orders, which is at the center of this work, constitutes one PPC job and takes into consideration the orders placed on the next machine given a number of machines and orders available. Other tasks are, sequence ordering that defines the sequences of orders processed by individual machines [10, 11].Optimization and decision support methods are used for PPC that are often categorized into two classes [12]: 1) Mathematical optimization, which allows finding an optimal solution but it often requires a higher computational effort. 2)Heuristics are often deployed in a wide-spread manner to avoid the challenges associated with computations in real-world problems.In much less time, these tends to achieve good results, but often falls before attaining the optimal solution.The aim of this work is to develop an autonomous solution using the deep neural network algorithm (DNN). This tends to address the delivery of orders in complex job shops with strict time constraints.

II. DEEP NEURAL NETWORK Neural network is a computing model based on neural network structure and biological functions. Typically, it is best suited for non-linear data. ANN ability to learn by training data sets constitutes one of the best known advantages. They are connected to the first layer of neurons that send data to a second layer, which then sends data to the output layer neurons of the third layer, by three layers, as indicated in Figure 1.All these links are related to weight learned by a neural network during workouts and this weight controls the overall cost of the model. If there is a minimum error in the prediction, the cost is low. The test cases are independent and are widely applied in different domains in these traditional feed-forward neural networks, such as image processing, object recognition and data classification.Recurrent neural neurons (RNNs) are another type of ANN that are structurally identical to FFNNs, but allow for connections within the same hidden layer between the neurons. It can map all historical input data to the last output by allowing historical input data to be stored in the internal network state.



III. **Figure 1. Architecture of ANN**

Even if one or several units react to the network, ANN works. There must be extensive and efficient neural networks in many production and storage resources. Whereas the brain has hardware for processing signals via a neurons, an ANN developer can be compelled to fill millions of database lines with connections using Von Neumann Technologies, which can use a vast amount of computer memory and hard drive space, to simulate a simplified form.Analyzed data teaches the ANN and does not require any re-programming. It is also known as the model blackbox and offers little insight into the real impact of the models. The user only needs to feed and track the input and wait for the output. ANN is seen as simple mathematical models for the improvement of existing data analysis technology.Deep Learning with Long Short-Term Memory Deep learning that ends the ending process is a division of machine learning. Machine learning algorithms are structured and further results with more datasets are created by understanding these structured data, which requires human intervention if the desired result is not obtained. In contrast, a deep learning network works on ANN layers as shown in figure 2 and learns from own mistakes. Therefore, no human intervention is required. One of advantages is its ability to study high-level functionality in an incremental order from the data provided that eliminates the need for expertise on the field.



Support Vector Regression (SVR) was used for the time series data prediction. However, this method is not structured to establish the parameters that are essential to the model effectiveness. Due to its flexible structure, the deep learning technique gains popularity. RNNs are an amendment to conventional neural networks, which exponentially decreases the input impact on hidden layers and output while cycling around the recurrent networks. Again, the structure of the hidden neurons in RNN was changed in the Long Short Term Memory (LSTM)

III.LONG SHORT TERM MEMORY MODELS (LSTM)

LSTM is a repetitive neural network (RNN) evolution. Normal RNN modules take over a single Tanh function the output from the last layer whereas LSTMs rely to remember feedback loop and gates. In every module, LSTM have 4 interactive NN layers, consisting of a cell, a gate, an output gate and a forgotten gate where cells record values over time intervals that are selfasserting, while the three gates direct data transition to and from cells. LSTM can add or delete data from the module state through sigmoid gates. The architecture of the LSTM cell is represented in Figure 3.

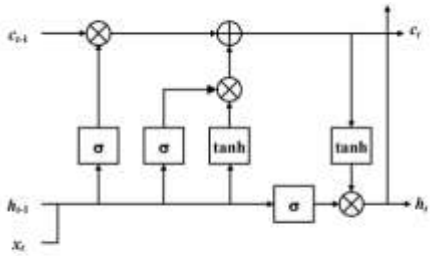


Figure 3. Long Short- Term Memory cell

LSTM is a type of supervised, highly effective, deep learning for prediction processing. Information is transmitted here through a mechanism to these cell states. This way LSTM chooses which data will be retained selectively.

IV.USE CASE DESCRIPTION AND MODELLING

This section introduces the case from a wafer factory in the real world. The modelling of DNN is described in relation to the optimization by DNN with the time constraints.

4. 4.1. Description of the wafer-fab use case The use case represents a production process in five machine groups with ten specialized machines sub-organized. Each machine group, also known as the job shop, has a buffer stock with 20 buffer slots. Many time-coupled process steps up across multiple labor centers often present challenges. The high proportion of time couplings limitations in about 30% of operations makes the order process and internal logistics more complex. In addition, the availability of the machines is also reduced and in previously visited machine groups, order flow is considered to be re-entrant.

5. Finally, machines need to be set up specifically for the product. The use case therefore includes the drivers of complexity that includes: re-entry flows, nonlinear process flows, sequence-related setup times and time constraints. It offers a suitable scenario for the use of DNN to analyses time limit influences in a complicated environment.

6. In this case, two product variants are modelled with the mix ratio of a same product but with different process times and time restriction intervals lengths (Table 1). The first variant of the product is considered faster, but

with stricter time constraints. The second product variants can be processed on any machine, and therefore have identical processing chains and the same basic variants. On average, orders in both product variants must therefore basically wait for the same time, depending on the order in front of the machine. The decision of the agent could, however, promote one product variety over the other.

Table 1 simulation parameters

Parameters	Use case requirement
Machine Groups (MG)	5
Total machines/MG	5/4/2/1/2 (five groups)
Product Types (PT)	2
Generated PT Probability	50% for both PT
Buffer size prior MG	20
Flow restriction	80
Time interval on order release	70s
Failures of machines	Varies based on the machine (probability of order 0.1)
Setup time	0-5s
Target flow factor	12 and 8
Coupling time constraint	225 – 600s

The state space shows the information that the agent receives as the basis for the decisionmaking process. The state space defined in this work is quite rich and consists of 210 entries, which are described in the following elements:Current action of the machine • Machine loading status • Setup of product per machine • Variant of a product in the buffer slots • Status of order per buffer slot • Status per buffer slot (Full or empty)The state values range from 0 to 1 and are binary. A single-hot encoding is applied to normalize the categorical data. The state values are based on the environmental status at the decisionmaking point. In this case, the status of the ordersare observed by the diagnosis using DNN, which is determined according to an urgency ratio rating of an order. The range values: 3, 2, and 1 are given for the three most urgently required orders. All other commands have a mark of 0 and a value of -5 is indicated for blank slots in the buffer of the machine group. The jobs in various product variants are therefore comparable to each other with different time constraints and processing times. Apart from the urgency ratio of an order, a due date deviation is used as the second priority rule based on the planned cycle time of a job, determined by the raw process time multiplied with the flow factor.

The process parameters and the product variant specifies the raw process times. A default input value that takes strategic objectives such as customer lead times into account is target flow factor.Before the order status is communicated to the agent by a buffer slot, it is important to be noted that order observation in the workplace can be shifted.

This process causes random slots in all commands and empty slots to be allocated. This is a key factor to prevent an agent from learning a slot-oriented, not order-specific policy. In addition, every tool configuration, the status type is included to allow the agent to optimize machine configuration sequences. A better installation and load management is required by the combination of the machines with respect to the variants of the product.

Action space modelling After the request from an agent, one of the 21 options is selected: • Order selection of the buffer slot • Chooses nil order upon idle state

Each of the 20 buffer slots is either busy or empty. It is only a valid measure to select an occupied slot. Invalid actions are not performed and are used as feedback to learn. The agency has a negative reward and is asked that another action be selected.The agent is given the option of raising or decreasing the rate of order new start to control and optimise the level when experimenting with DNN.

V.RESULTS AND DISCUSSIONS

Depending on a review of other DNN applications [13] – [17], of some of the most important settings and hyper-parameters (see Table 2).

Table 2 DNN settings for the use case scenario

Parameters	Use case requirement
Total decisions by DNN	10 billion steps
Learning rate	Upto 5 billion steps
Target model update	Every 0.01 billion steps
Warm up phase	Every 0.01 billion steps
Sequential Memory Size	1 billion
Discount factor	0.85
Learning rate	24×10[illegible]

DNN performance is compared with the state-of-the-art time limit heuristics. The idea behind the DNN is to model a realistically timely approach only, if an order is nearly in violation of its time limit.

In addition, a FIFO heuristic is included as a benchmark, which is well established in the application for production control but does not take account of time constraints and due dates directly. The results are based on simulation runs for the scenario above. Simulation steps (1.2 million) are performed in each experiment, which defends a simulation step as the selection and implementation of action by the agent. Initially, the exploration value, i.e. the part of exploratory activities, is set to 1 and declines linearly to 0.01 over the initial5 billion steps. Thus, the agent is only using the best learned action so far for the last 5 billion steps.However, it should be noted that the learning phase fails to stop since random machine failures and sequence of order lead to the randomness of the learning process.

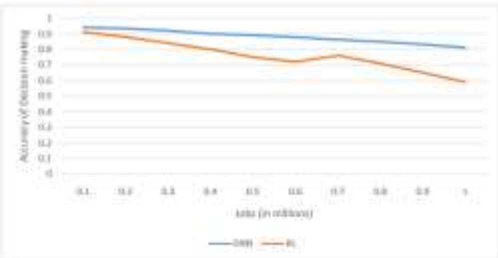


Figure 4. Accuracy of decisions made by DNN

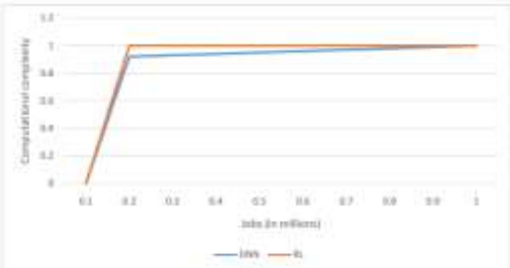


Figure 5. Computational Complexity of decisions making

Figure 4 shows the accuracy of decisions made by DNN, where the accuracy of the decision made by DNN is higher than the reinforcement algorithm. However with increasing job load, the accuracy tends to degrade. The Figure 5 shows the computational complexity of decisions making, where the present system using DNN has reduced computational complexity than RL but the complexity tends to increase with increased number of jobs.This research is supported by a study of order delivery in complex workshops within time constraints. The results of calculation are based on a semi-conductive simulation of a wafer fab and reveal the high potential of a DNN algorithm that is model-free. The results show that DNN agents can successfully be used in a real-world application case to control order dispatch. In addition to established industry-wide benchmarks, time limits are managed, i.e. a heuristic that optimizes time constraints or a FIFO-oriented heuristic.In addition, experiments demonstrate that reward modelling for successful DNN applications is a key and crucial element. This is especially important for problems of constraint such as time limits based on orders. This research contributes to the investigation into how best to implement DNN to optimize the validity of actions. It is also shown how a small extension, in conjunction with a complementary award, allows the scope of action and the policy area to be easily extended.

VI.CONCLUSION

Enhancing and refining state space and reward functions seem to provide more opportunities for even better outcomes. For example, by incorporating buffer usage or the anticipated residual machine life in state space, it would enable the agent to optimise flow and even preventative maintenance plans. In terms of deep learning capabilities, adding more layers and sophisticated layer architectures should be considered. Finally, while other DNN methods demonstrate robustness, especially in the face of changing issue features, they may also yield favourable outcomes.

Undoubtedly, real-world self-learning systems are still a long way from reaching their long-term goals of development. To achieve this, advancements are needed in a number of areas: data availability and quality remain a major issue and a determining factor for many businesses. Moreover, anomaly management and cyber security are involved in the deployment of autonomous systems.

Individual decisions must be understood and respected, and people need to be persuaded of this. To be employed as a single production decision-maker, DNN needs to be comprehended and developed further, as it is the most promising approach to do this.

REFERENCES

[1] Wiendahl, H. P., ElMaraghy, H. A., Nyhuis, P., Zäh, M. F., Wiendahl, H. H., Duffie, N., &Brieke, M. (2007). Changeable production-classification, design and operation. CIRP annals, 56(2), 783-809.

[2] Lanza, G., Ferdows, K., Kara, S., Mourtzis, D., Schuh, G., Váncza, J., ... &Wiendahl, H. P. (2019). Global production networks: Design and operation. CIRP annals, 68(2), 823- 841.

[3] Bauernhansl, T., Ten Hompel, M., & Vogel-Heuser, B. (Eds.). (2014). Industry 4.0 in production, automation and logistics: Application · Technologies · Migration (pp. 1-648). Springer Vieweg, Wiesbaden.

[4] Waschneck, B., Altenmüller, T., Bauernhansl, T., &Kyek, A. (2016, October). Production Scheduling in Complex Job Shops from an Industry 4.0 Perspective: A Review and Challenges in the Semiconductor Industry. In SAMI@ iKNOW (pp. 1-12).

[5] Greschke, P., Schönemann, M., Thiede, S., & Herrmann, C. (2014). Matrix structures for high volumes and flexibility in production systems. Procedia CIRP, 17(1), 160-165.

[6] Silver, D., Schrittwieser, J., Simonyan, K., Antonoglou, I., Huang, A., Guez, A., ... & Chen, Y. (2017). Mastering the game of go without human knowledge. nature, 550(7676), 354-359.

[7] Eversheim, W., &Wiendahl, H. P. (2000). Wörterbuch der PPS Dictionary of PPC. [8] Lödding, H. (2005). Process of production control Basics, description, configuration (Vol. 2). Berlin: Springer.