Available online at www.sciencedirect.com

ScienceDirect

Procedia CIRP 103 (2021) 55–60

www.elsevier.com/locate/procedia

9th CIRP Global Web Conference – Sustainable, resilient, and agile manufacturing and service operations : Lessons from COVID-19

Multi-variate time-series for time constraint adherence prediction in complex job shops

Marvin Carl May*[a], Lukas Behnen[a], Andrea Holzer[b], Andreas Kuhnle[a], Gisela Lanza[a]

[a] *wbk Institute of Production Science, Karlsruhe Institute of Technology (KIT), Kaiserstr. 12, 76131 Karlsruhe, Germany*
[b] *Infineon Technologies AG, Wernerwerkstraße 2, 93049 Regensburg, Germany*

* Corresponding author. Tel:+49-1523-950-2624; Fax:+49-721-60845005. *E-mail address:* marvin.may@kit.edu

Abstract

One of the most complex and agile production environments is semiconductor manufacturing, especially wafer fabrication, as products require more than several hundred operations and remain in Work-In-Progress for months leading to complex job shops. Additionally, an increasingly competitive market environment, i.e. owing to Moore's law, forces semiconductor companies to focus on operational excellence, resiliency and, hence, leads to product quality as a decisive factor. Product-specific time constraints comprising two or more, not necessarily consecutive, operations ensure product quality at an operational level and, thus, are an industry-specific challenge. Time constraint adherence is of utmost importance, since violations typically lead to scrapping entire lots and a deteriorating yield. Dispatching decisions that determine time constraint adherence are as a state of the art performed manually, which is stressful and error-prone. Therefore, this article presents a data-driven approach combining multi-variate time-series with centralized information to predict time constraint adherence probability in wafer fabrication to facilitate dispatching. Real-world data is analyzed and different statistical and machine learning models are evaluated.

Keywords: Production Planning and Control; Time Constraints; Data Analytics; Time Series Analysis; Machine Learning

1. Introduction

Wafer fabrication is among the most complex industrial environments as products require several hundred operations to be produced. It is associated with many process-related challenges leading to complex job shops [22]. An increasingly competitive market environment forces semiconductor companies to focus on operational improvements to realize necessary cost and cycle time reductions and makes quality a decisive competitive factor [21]. One industry-specific challenge emerges from product-related time constraints, which are imposed for quality and yield purposes [16]. A time constraint is defined by two operations O_{lr}, O_{ls} with $r < s$ that are linked by a lot-specific time limit t_{lrs}, which restricts the maximum time allowed between the completion of operation O_{lr} and start of operation O_{ls} for lot l [10]. Additionally, multiple time constraints might be nested or directly succeeding one another resulting in complex time constraints [28]. Violations of time constraints lead to scrapping or rework of an entire lot [2]. Thus, time constraint adherence is of utmost importance.

Dispatching decisions on an operational level heavily influence time constraint adherence and are often based on the operators' experience and therefore require additional manual effort and are error-prone [16]. Data availability increases manifold and, thus, this article presents a data-driven approach applying multivariate time series analysis to predict time constraint adherence probability in wafer fabrication dispatching. Real-world data is analyzed and different statistical and machine learning models are comparatively evaluated.

Therefore, modeling different types of time constraints is reviewed in Section 2. Section 3 introduces the modeling approach, which consists of a point estimator and a corresponding prediction interval. In Section 4, a case study is presented, whose results are subsequently discussed in Section 5. This paper concludes with an outlook and indications for further research in Section 6.

10.1016/j.procir.2021.10.008

2. Related Work

Dispatching, together with Scheduling, belongs to production control in the context of semiconductor manufacturing [22]. The former assigns lots waiting to be processed to resources dynamically in a minute-by-minute manner, while the latter prescribes a plan that assigns lots well into the future. Both can follow multiple objectives, typically aiming at decreasing cycle times, maximizing throughput or minimizing cost or time constraint violations. If retaining time constraints is targeted, dispatching is advantageous as sudden statistical realizations such as machine failures can be incorporated [20]. Hence, the following research focuses on dispatching in the context of time constraints. Thereby, a *gate-keeping decision* determining whether or not a waiting lot is processed is made.

2.1. Literature Review

Although priority-based dispatching rules (heuristics) lack a global perspective and are oversimplified, they are still widely used in practice due to the computational limitations of scheduling approaches. Regarding time constraints, there are several studies that evaluate the robustness of different dispatching heuristics and propose new procedures to reduce violations as summarized in Table 1.

The modeling approaches can be categorized in Mixed Integer Programming (MIP), dynamic decision models based on Markov Decision Problems (MDP), experiment analyses as well as graph or queuing theory based approaches. Heuristic approaches prevail, focusing on several stage policies, which integrate batching and dispatching [2, 4, 23]. Throughput oriented heuristics are based on (Work-In-Progress) WIP levels [12] or capacity thresholds [18] as well as queuing theory based approaches [31]. However, the heuristically regarded problems are limited, i.e. to wet etch - furnace operations or implantation [4, 31] and reducing cycle time violations is typically secondary to throughput maximization [2], cycle time minimization [4] or utilization maximization [31]. In a similar vein simulations are used to identify recommendations to reduce the time constraint impact [25] or to identify and verify heuristics [33]. Yet, their limited scope impede transferability to the real world.

Modeling	Solution	Objective	Ref.
MIP	heuristics & neural network	max. throughput, min. time constraint violations	[13]
	heuristic control	min. avg. cycle time	[4]
Queuing Theory	heuristic	reduce setup times	[31]
MDP	RL	increase utilization	[1]
	decomposition-based opt.	max. production rate & min. scrap rate	[27]
	value iteration algorithm	min. inventory holding & scrap costs	[30, 29]
Disjunct. graph	sampling-based heuristic	estimate adherence probability	[15, 14, 17, 24]
Experiments	plan-based heuristic	min. violation ratio & max. avg. cycle time	[33]
	heuristic	min. violations	[12]
		multi-objective	[2, 11, 23, 26]
	simulations	max. machine utilization	[25]
	data analysis	min. violations	[20]

Table 1. Classification of the relevant literature at a dispatching level based on objective, modeling approach and solution technique

Data-based dispatching based on a MDP regards two-stage production and batching [30], but only recently addressed a simplified production system [1] or multiple products [29]. Time constraints are implicitly regarded in the objective function as costs [1, 30]. Algorithms are found on value iteration [30, 29], decomposition [27] and deep Reinforcement Learning (RL) [1]. Depending on the problem size, traditional heuristic approaches are outperformed [1], but high training effort and low generalization [27] preclude real world application.

Few studies try to predict the probability of time constraint violations to support the gate keeping decision in dispatching. Probabilistic, disjunctive graph model approaches deal with complex, nested time constraints involving a randomized list scheduling algorithm [24], improved dispatching policies [14, 17] and decision support for nested time constraints [15]. The violation probability on order release can be based on conservative queue time predictions and time limit comparison [28]. In contrast to these studies, real world data is used to learn a dispatching rule in form of a neural network [13] or predict WIP levels and arrival rates at tool groups to reduce the WIP level based on tool allocation [6], which is beneficial for time constraint adherence, but on a planning level. Real world data from a large fab is analyzed with a single-variate time-series approach [20] outperforming traditional and manual approaches.

2.2. Research agenda

The literature review reveals a plethora of approaches to deal with time constraints in semiconductor manufacturing in dispatching. Multi-variate approaches and the study of entire productions systems with real world data are neglected. Due to the strong assumptions applied in optimization, simplifications in heuristic solutions and applied simulation as well as the limited scope of single-variate learning, practical application in real world production systems is hindered. Thus, this paper presents a multi-variate machine learning based approach to accurately predict time constraint violations in wafer fabs, validated with extensive real world data.

3. Modeling Approach

In order to support the operators in their gate-keeping decision for *simple time constraints* at a dispatching level, a data-driven model is developed to predict time constraint adherence probability. A simple time constraint as depicted in Figure 1 involves two consecutive process steps that are linked by a time limit d^u such that the operation at the downstream equipment m_2 has to start within the prescribed time limit after the operation at the upstream equipment m_1 is completed.

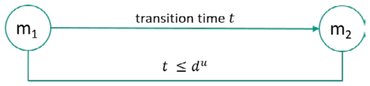

Fig. 1. Simple time-constrained transitions

The modeling approach is based on predictions of the transition time for each job from one equipment to another, which is supplemented by a prediction interval in order to derive the adherence probability, in similar vein to May et al. [20]. A prediction interval is a statistical interval, which specifies the range a future value is expected to lie in with a prescribed probability, called confidence level [5]. In order to evaluate the probability of not exceeding the imposed time limit, a one-sided prediction interval is constructed as follows:

$$(-\infty, \hat{y} + t_{n-1,1-\alpha} \cdot \sqrt{Var(e)}] \tag{1}$$

with $\hat{y}$ being the point estimator, $t_{n-1,1-\alpha}$ the $1-\alpha$-quantile of the student's t-distribution and $Var(e)$ the prediction error variance.

3.1. Point Estimators

Different point estimators for the prediction of transition time are implemented and comparatively evaluated. These models are centered around a transition time autocorrelation, first exploited by a previous study May et al. [20].

Autoregressive integrated moving average (ARIMA) models are linear, univariate statistical models that are defined by the three components Autoregression (AR), Moving Average (MA) and an integration operator I, which transforms the time series so that it is stationary by differencing. Once a stationary time series is obtained, an ARMA model combining a p-th order Autoregressive process with a q-th order Moving Average process is fitted and derives a prediction of the current value X_t, according to Equation 2, based on the weighted sum of the past p observations and q error terms plus a constant c and an error term ϵ_t obtained from a white noise process.

$$x_t = c + \epsilon_t + \sum_{i=1}^{p} \phi_i x_{t-i} + \sum_{j=1}^{q} \theta_j \epsilon_{t-j} \tag{2}$$

Appropriate selection of the two hyperparameters p, q representing the order of the Autoregressive or respectively, the Moving Average process is crucial for the model's performance. The weight parameters $\phi_{t-p}, ..., \phi_{t-1}$ and $\theta_{t-q}, ..., \theta_{t-1}$ are estimated through least square minimization.

Neural networks are increasingly gaining attention in time series analysis as they are able to incorporate exogenous variables by design and allow for end-to-end modeling and automatic feature extraction. In contrast to ARIMA models, neural networks are data-driven and non-parametric, which makes them less dependent on assumptions about the underlying data generating process [32]. Artificial neural networks are able to solve complex tasks by non-linear combinations of inputs, which are passed through the network from an input layer through one or multiple hidden layers to an output layer. The involved weights and bias parameters are learned from backpropagation minimizing a loss function. However, common feedforward neural networks are limited in terms of processing sequential time series data as they require a fixed input size, impeding modeling variable sequence lengths or long-term dependencies. Furthermore, parameter sharing is infeasible.

Recurrent neural networks (RNNs) are particularly suitable to sequential time series analysis due to their recurrent architecture. An internal, hidden state h_t is maintained and updated at each time step t and passed on to the next time step $t+1$, which allows information to persist over time. Thereby, modeling of long-term dependencies becomes feasible and parameter sharing is enabled by reusing the same weight matrices. However, RNNs suffer from the vanishing-gradient problem, since backpropagation over time involves repeated gradient multiplications. **Long Short-Term Memory (LSTM)** networks, as an extension of standard RNNs, use more complex, gated cells and maintain two separate hidden states to deal with the vanishing gradient problem, which enables them to model long term dependencies [8].

3.2. Prediction Interval construction

According to Equation 1, a point estimator $\hat{y}$, the corresponding quantile of the student's t-distribution as well as the variance of the prediction error $Var(e)$ are required to construct a prediction interval. A prediction $\hat{y}$ for the next value is, for example, obtained by applying the models presented in Section 3.1. Computation of the variance of the prediction error $Var(e)$ is less trivial and derived in the following.

Since an observed target t_i is composed of a signal y_i and inherent noise ϵ_i, the prediction error e_i, given a point forecast $\hat{y}_i$ is computed as follows [9]:

$$e_i = (y_i - \hat{y}_i) + \epsilon_i. \tag{3}$$

Assuming independence of both terms in Equation 3, the variance of the prediction error can be decomposed into the two components epistemic or model uncertainty $\sigma^2_{\hat{y}_i}$ and aleatoric uncertainty or inherent noise $\sigma^2_{\hat{\epsilon}_i}$ as shown in Equation 4 [9].

$$Var(e) = \sigma^2_{\hat{y}_i} + \sigma^2_{\hat{\epsilon}_i} \tag{4}$$

For ARIMA models, an analytically derived formula is used for the computation of the variance of the prediction error and is, for one-step ahead predictions, given by:

$$Var(e) = \sigma_e^2(1 + k\frac{1}{n}), \quad (5)$$

where σ_e^2 quantifies the inherent noise, which can be estimated on an independent hold-out dataset [5]. The second term accounts for the model uncertainty from weight parameter estimation through least square minimization, which decreases with increasing data size n.

For neural networks however, a general formula cannot be derived. While the inherent noise can also be estimated on an independent hold-out dataset, estimation of the model uncertainty component in Equation 4 requires empirical techniques. Bayessian neural networks, where a probability distribution is placed over the network's parameters have a strong mathematical foundation for uncertainty quantification, but they require immense computational effort [9]. However, an approximate Bayesian approach for a Gaussian process, called Monte-Carlo dropout, exists [7]. By applying dropout to each hidden layer at inference and performing B stochastic forward passes through the network for a specific input i, multiple predictions are sampled. Model uncertainty can be approximated by the sample variance according to Equation 6.

$$\hat{\sigma}_{\hat{y}_i}^2 = \frac{1}{B-1}\sum_{b=1}^{B}(y_i - \hat{y}_i^b)^2 \quad (6)$$

Monte-Carlo dropout quickly provides model uncertainty estimates and is easy to implement, since it is directly applicable to different existing neural network architectures [34].

3.3. Derivation of the time constraint adherence probability

The goal is to estimate the time constraint adherence probability for each transition individually based on a one-sided prediction interval given in Equation 1. At confidence level $1 - \alpha$, the time constraint is not violated if the upper bound of the prediction interval is less than or equal to the lot-specific time limit d^u illustrated in Figure 2.

$$\hat{y} + t_{n-1,1-\alpha}Var(e) \leq d^u. \quad (7)$$

By transforming Equation 7, the time constraint adherence probability can be derived from the cumulative density function of the student's t-distribution for the result of Equation 8.

$$t_{n-1,1-\alpha} = \frac{d^u - \hat{y}}{\sqrt{Var(e)}} \quad (8)$$

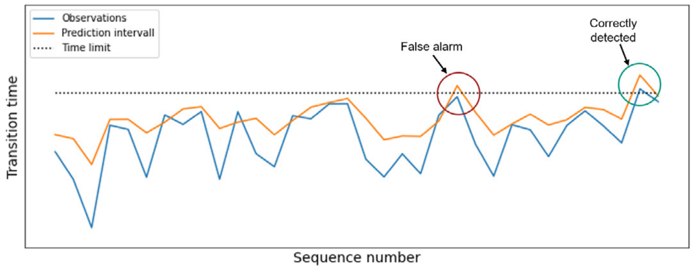

Fig. 2. Classification of transitions based on a one-sided prediction interval

4. Case Study

The proposed modeling approach combining a point estimator for time-constrained transition times with a prediction interval in order to derive the time constraint adherence probability is applied to real historical wafer fabrication data. The analyzed transactional log data queried from the Manufacturing Execution System, which documents the lot progress and provides further relevant information, is used to reconstruct the highly right-skewed transition times [20]. The minority of transitions takes extremely long and, thus, the data is logartihmically transformed to reduce the extreme range [20]. Hence, the dataset is also extremely imbalanced in terms of the ratio of time constraint violations to non-violated ones, motivating the regression modeling approach with uncertainty quantification.

Since ARIMA models require little computational effort and solely rely on previous observations, a separate ARIMA model is fitted to the sequence of transition times for each type of time-constrained transition defined by two consecutive equipments.

Regarding learning approaches, models are trained for all types of transitions using the following additional features, which are selected based on domain knowledge and data exploration:

- Current queue at time constraint's ending equipment [19]
- Time since the last downstream equipment breakdown
- Indicator features: e.g. work centers, operations, equipment, downstream equipment type & weekday.

A feed-forward neural network, an LSTM network and a model combining LSTM and fully-connected layers are trained. Dropout is applied to each hidden layer to estimate the model uncertainty using Monte-Carlo dropout. The most important hyperparameters such as the number of neurons and the dropout rates are optimized using Bayesian hyperparameter tuning [3].

4.1. Performance evaluation

The implemented models are evaluated in terms of the quality of the point forecasts and the resulting prediction intervals. While the transition time predictions are assessed on a hold-out dataset using common regression performance metrics such as the mean squared error (MSE) and visual examination, the prediction interval evaluation involves a trade-off between coverage and interval size.

	ARIMA	feed-forw. NN.	LSTM	LSTM & Dense
MSE	1.9640	1.0578	1.2254	1.2009
PICP	0.8953	0.9325	0.9328	0.9472
MPIW	9.6331	9.6485	9.6783	9.7900

Table 2. Summary of performance metrics of the different models

Coverage, measured by the prediction interval coverage probability (PICP) in Equation 9, refers to the number of target values t_i of a test dataset of size n that are comprised by the corresponding upper bound U_i of the one-sided prediction interval and should be equal to or higher than the prescribed nominal confidence level $1 - \alpha$ [9].

$$PICP = \frac{1}{n}\sum_{i=1}^{n} c_i, \text{ where } c_i = \begin{cases} 1, & t_i \leq U_i \\ 0, & \text{else} \end{cases} \tag{9}$$

The PICP is directly related to the size of the upper bounds, as arbitrary large, but not meaningful, prediction intervals can achieve a coverage of 1 [9]. Thus, the (mean) prediction interval width (MPIW) has to be taken into account, which can be measured according to Equation 10 by the mean of the upper bounds on a test dataset.

$$MPIW = \frac{1}{n}\sum_{i=1}^{n} U_i \tag{10}$$

A final evaluation is carried out in terms of the resulting classification of time-constrained transitions into adhered and violated ones based on Equation 7 at different confidence levels $1 - \alpha$.

4.2. Results

Performance results are summarized in Table 2 and indicate that the learning-based approaches provide better predictions of the transition times than the simple ARIMA model. The neural networks' prediction intervals, however, tend to be wider, which is shown by the corresponding performance metrics computed for a confidence level of 90%.

In terms of the resulting classification, however, ARIMA models are already able to correctly detect all time constraint violations using a threshold of 70% and achieve an overall accuracy of 96.31%. While accuracy and recall are high, precision of 11.73% is low due to the highly imbalanced data, which corresponds to a high false alarm rate. Increasing the threshold reduces precision and overall accuracy even further as prediction intervals become wider leading to a higher false alarm rate. Classifications based on the prediction intervals of neural networks are worse, especially at low thresholds due to wide prediction intervals leading to more falsely detected violations.

5. Discussion

Although ARIMA models are simplistic and their point estimates of transition times are worse than all learning-based approaches, they ultimately yield the best results in terms of classifying time-constrained transitions into violated and adhered ones, which might seem contradictory, but can be explained by the much higher number of parameters in neural networks leading to a higher model uncertainty. As a result, prediction intervals become wide and many transitions are falsely classified as violations. The performance metrics in Table 2 confirm this finding as the models containing more parameters have a higher average upper bound and an actual coverage that is higher than the nominal level of 90%.

6. Outlook

The proposed model addressing simple time constraints combines a point estimator with a prediction interval to derive the adherence probability of simple time constraints aiming at detecting potential violations in advance and is applied to real-world manufacturing data of an entire wafer fab. The results are promising as simple ARIMA models are able to detect all time constraint violations. However, precision is low, since many transitions are falsely classified as a violation. Furthermore, the prediction intervals of learning-based approaches, which provide better transition time forecasts, become too wide as a result of the model uncertainty quantification using Monte-Carlo dropout, resulting in worse classifications.

One major problem for modeling is the heterogeneity of observed transition time sequences depending on the involved equipment due to the extremely complex manufacturing environment and diversity of process steps in wafer fabrication. Therefore, future research can focus on decomposing the problem, for example by work areas and constructing area-specific features to improve the prediction of transition times. Furthermore, a more detailed equipment type consideration is necessary. While the constructed features significantly influence transitions ending at cluster or single tools, other factors seem to be decisive for batch equipment. Instead of decoupling prediction interval construction from the network training, prediction intervals can be modeled end-to-end instead. Future research shall also extend this modeling approach to more complex time constraints comprising multiple process steps.

Acknowledgements

This research work was undertaken in the context of DIGIMAN4.0 project ("DIGItal MANufacturing Technologies for Zero-defect Industry 4.0 Production", http://www.digiman4-0.mek.dtu.dk/). DIGIMAN4.0 is a European Training Network supported by Horizon 2020, the EU Framework Programme for Research and Innovation (Project ID: 814225).

References

[1] Altenmüller, T., Stüker, T., Waschneck, B., Kuhnle, A., Lanza, G., 2020. Reinforcement learning for an intelligent and autonomous production control of complex job-shops under time constraints. Production Engineering 14, 319–328. doi:10.1007/s11740-020-00967-8.

[2] Arima, S., Kobayashi, A., Wang, Y.F., Sakurai, K., Monma, Y., 2015. Optimization of re-entrant hybrid flows with multiple queue time constraints in batch processes of semiconductor manufacturing. IEEE Transactions on Semiconductor Manufacturing 28, 528–544. doi:10.1109/TSM.2015.2478281.

[3] Bergstra, J., Bardenet, R., Bengio, Y., Kégl, B., 2011. Algorithms for hyper-parameter optimization, in: Proceedings of the 24th International Conference on Neural Information Processing Systems, Curran Associates Inc., Red Hook, NY, USA. p. 2546–2554.

[4] Chang, C.Y., Chang, K.H., 2012. An integrated and improved dispatching approach to reduce cycle time of wet etch and furnace operations in semiconductor fabrication, in: Gao, L. (Ed.), IEEE 16th International Conference on Computer Supported Cooperative Work in Design (CSCWD), 2012, IEEE, Piscataway, NJ. pp. 734–741. doi:10.1109/CSCWD.2012.6221901.

[5] Chatfield, C., 1993. Calculating interval forecasts. Journal of Business & Economic Statistics 11, 121. doi:10.2307/1391361.

[6] Chien, C.F., Kuo, C.J., Yu, C.M., 2020. Tool allocation to smooth work-in-process for cycle time reduction and an empirical study. Annals of Operations Research 290, 1009–1033. doi:10.1007/s10479-018-3034-5.

[7] Gal, Y., Ghahramani, Z., 2016. Dropout as a bayesian approximation: Representing model uncertainty in deep learning, in: Proceedings of the 33rd International Conference on Machine Learning - Vol. 48, pp. 1050–1059.

[8] Hochreiter, S., Schmidhuber, J., 1997. Long short-term memory. Neural Comput. 9, 1735–1780. URL: https://doi.org/10.1162/neco.1997.9.8.1735.

[9] Khosravi, A., Nahavandi, S., Creighton, D., Atiya, A.F., 2011. Comprehensive review of neural network-based prediction intervals and new advances. IEEE transactions on neural networks 22, 1341–1356. doi:10.1109/TNN.2011.2162110.

[10] Klemmt, A., Monch, L., 2012. Scheduling jobs with time constraints between consecutive process steps in semiconductor manufacturing, in: Staff, I. (Ed.), 2012 Winter Simulation Conference, IEEE. pp. 1–10. doi:10.1109/WSC.2012.6465235.

[11] Kobayashi, A., Kuno, T., Arima, S., 2013. Re-entrant flow control in q-time constraints processes for actual applications, in: E-Manufacturing & Design Collaboration Symposium (eMDC), 2013, IEEE. pp. 1–4. doi:10.1109/eMDC.2013.6756052.

[12] Lee, Y.Y., Chen, C.T., Wu, C., 2005. Reaction chain of process queue time quality control, in: ISSM 2005, IEEE, Piscataway, N.J. pp. 47–50. doi:10.1109/ISSM.2005.1513293.

[13] Li, L., Li, Y.F., Sun, Z.J., 2012. Dispatching rule considering time-constraints on processes for semiconductor wafer fabrication facility, in: IEEE International Conference on Automation Science and Engineering (CASE), 2012, IEEE, Piscataway, NJ. pp. 407–412. doi:10.1109/CoASE.2012.6386370.

[14] Lima, A., 2017. Analyzing different dispatching policies for probability estimation in time constraint tunnels in semiconductor manufacturing, in: Chan, W.K., D'Ambrogio, A., Zacharewicz, G., Mustafee, N., Wainer, G., Page, E.H. (Eds.), WSC'17, IEEE, Piscataway, NJ. pp. 4606–4607. doi:10.1109/WSC.2017.8248227.

[15] Lima, A., Borodin, V., Dauzere-Peres, S., Vialletelle, P., 2017. A decision support system for managing line stops of time constraint tunnels: Fa, ie, in: 2017 28th Annual SEMI Advanced Semiconductor Manufacturing Conference (ASMC), IEEE, Piscataway, NJ. pp. 309–314. doi:10.1109/ASMC.2017.7969250.

[16] Lima, A., Borodin, V., Dauzère-Pérès, S., Vialletelle, P., 2019. Sampling-based release control of multiple lots in time constraint tunnels. Computers in Industry 110, 3–11. doi:10.1016/j.compind.2019.04.014.

[17] Lima, A., Borodin, V., Dauzère-Pérès, S., Vialletelle, P., 2021. A sampling-based approach for managing lot release in time constraint tunnels in semiconductor manufacturing. International Journal of Production Research 59, 860–884. doi:10.1080/00207543.2020.1711984.

[18] Maleck, C., Eckert, T., 2017. A comparison of control methods for production areas with time constraints and tool interruptions in semiconductor manufacturing, in: 2017 40th International Spring Seminar on Electronics Technology (ISSE), IEEE, Piscataway, NJ. pp. 1–6. doi:10.1109/ISSE.2017.8000944.

[19] May, M.C., Albers, A., Fischer, M.D., Mayerhofer, F., Schäfer, L., Lanza, G., 2021a. Queue length forecasting in complex manufacturing job shops. Forecasting 3, 322–338. doi:10.3390/forecast3020021.

[20] May, M.C., Maucher, S., Holzer, A., Kuhnle, A., Lanza, G., 2021b. Data analytics for time constraint adherence prediction in a semiconductor manufacturing use-case. Procedia CIRP 100, 49–54. doi:10.1016/j.procir.2021.05.008.

[21] Mönch, L., Fowler, J.W., Dauzère-Pérès, S., Mason, S.J., Rose, O., 2011. A survey of problems, solution techniques, and future challenges in scheduling semiconductor manufacturing operations. Journal of Scheduling 14, 583–599. doi:10.1007/s10951-010-0222-9.

[22] Mönch, L., Fowler, J.W., Mason, S.J., 2013. Production planning and control for semiconductor wafer fabrication facilities: Modeling, analysis, and systems. volume 52 of *Operations Research / Computer Science Interfaces Series*. Springer, New York, NY. doi:10.1007/978-1-4614-4472-5.

[23] Pirovano, G., Ciccullo, F., Pero, M., Rossi, T., 2020. Scheduling batches with time constraints in wafer fabrication. International Journal of Operational Research 37, 1. doi:10.1504/IJOR.2020.104222.

[24] Sadeghi, R., Dauzere-Peres, S., Yugma, C., Lepelletier, G., 2015. Production control in semiconductor manufacturing with time constraints, in: 2015 26th Annual SEMI Advanced Semiconductor Manufacturing Conference (ASMC), IEEE, Piscataway, NJ. pp. 29–33.

[25] Scholl, W., Domaschke, J., 2000. Implementation of modeling and simulation in semiconductor wafer fabrication with time constraints between wet etch and furnace operations. IEEE Transactions on Semiconductor Manufacturing 13, 273–277. doi:10.1109/66.857935.

[26] Toyoshima, N., Hasegawa, T., Wu, K., Arima, S., 2013. Proactive control of engineering operations and lot loadings of product-mix and re-entrant in q-time constraints processes, in: E-Manufacturing & Design Collaboration Symposium (eMDC), 2013, IEEE. pp. 1–4. doi:10.1109/eMDC.2013.6756046.

[27] Wang, J., Hu, H., Pan, C., Zhou, Y., Li, L., 2020. Scheduling dual-arm cluster tools with multiple wafer types and residency time constraints. IEEE/CAA Journal of Automatica Sinica 7, 776–789. doi:10.1109/JAS.2020.1003150.

[28] Wang, M., Srivathsan, S., Huang, E., Wu, K., 2018. Job dispatch control for production lines with overlapped time window constraints. IEEE Transactions on Semiconductor Manufacturing 31, 206–214. doi:10.1109/TSM.2018.2826530.

[29] Wu, C.H., Chien, W.C., Chuang, Y.T., Cheng, Y.C., 2016. Multiple product admission control in semiconductor manufacturing systems with process queue time (pqt) constraints. Computers & Industrial Engineering 99, 347–363. doi:10.1016/j.cie.2016.04.003.

[30] Wu, C.H., Lin, J.T., Chien, W.C., 2010. Dynamic production control in a serial line with process queue time constraint. International Journal of Production Research 48, 3823–3843. doi:10.1080/00207540902922836.

[31] Yang, K.T., Ke, L., Shen, T., 2015. Modeling and dispatching refinement for implantation to reduce the probability of tuning beam, in: 2015 26th Annual SEMI Advanced Semiconductor Manufacturing Conference (ASMC), IEEE, Piscataway, NJ. pp. 190–194. doi:10.1109/ASMC.2015.7164467.

[32] Zhang, G.P., 2012. Neural networks for time-series forecasting, in: Rozenberg, G., Bäck, T., Kok, J.N. (Eds.), Handbook of Natural Computing. Springer Berlin Heidelberg, Berlin, Heidelberg, pp. 461–477.

[33] Zhang, T., Pappert, F.S., Rose, O., 2016. Time bound control in a stochastic dynamic wafer fab, in: Roeder, T.M., Frazier, P.I., Szechtman, R., Zhou, E. (Eds.), Simulating complex service systems, IEEE, Piscataway, NJ. pp. 2903–2911. doi:10.1109/WSC.2016.7822325.

[34] Zhu, L., Laptev, N., 2017. Deep and confident prediction for time series at uber, in: 2017 IEEE International Conference on Data Mining Workshops (ICDMW), IEEE. pp. 103–110. doi:10.1109/ICDMW.2017.19.