Available online at www.sciencedirect.com

ScienceDirect

Procedia Computer Science 200 (2022) 1145–1154

Procedia Computer Science

www.elsevier.com/locate/procedia

3rd International Conference on Industry 4.0 and Smart Manufacturing

Bottleneck prediction and data-driven discrete-event simulation for a balanced manufacturing line

Eugénio M. Rocha[a,b], Maria J. Lopes[c,*]

[a] *Department of Mathematics, University of Aveiro, 3810, Portugal*
[b] *Center for Research and Development in Mathematics and Applications (CIDMA), Aveiro, 3810, Portugal*
[c] *Bosch Termotecnologia S.A., 3800, Portugal*

Abstract

Bottleneck identification is a relevant tool for continuous optimization of production lines. In this work, we implement a data-driven discrete-event simulator (DDS) based on experimental distributions, obtained from real historical data. The DDS allows to analyse the behavior of a balanced manufacturing line at Bosch Thermotechnology, under different hypotheses. It shows that some scenarios perceived as likely to increase output may actually decrease production metrics, reveals the importance of line injection rates, and leads to the need for adequate real time bottleneck forecasting tools, which allow shift managers intervention in a useful time frame. Eleven prediction models are tested, where a random forest and a multi-layer perceptron attain the best performances (above 95% in all metrics). This data flow is operationalized through a micro-services pipeline which is briefly discussed.

Keywords: data-driven discrete-event simulation; bottleneck prediction; machine learning; manufacturing lines

1. Introduction

This work demonstrates how to provide prediction capabilities to a bottleneck framework which uses very basic input data and as such is relevant in balanced manufacturing lines through the validation of different scenarios using data-driven discrete-event simulation. The methods described here will later be applied in a concrete pipeline at a Bosch Thermotechnology plant, which uses the same abstraction concept to compute higher level metrics based on different levels of information as explored in [1]. It constitutes a new layer on top of a recent work [2].

Bottleneck mitigation is a core problem in almost every industry [3]. Simulation is a powerful tool to be used together with optimization processes as illustrated in [4], specially when traditional discrete-event simulation based on distribution models is replaced by discrete-event simulation based on fitted distributions from real data. In this last

* Corresponding author. Tel.: +351(234)925-643.
E-mail address: mariajoao.lopes@pt.bosch.com

1877-0509
10.1016/j.procs.2022.01.314

application, the authors have resorted to expert knowledge for bottleneck identification while simulating line output which was then optimized through a genetic algorithm. Operators were reassigned to different workstations based on their skill set level which greatly improved overall line productivity. These were particularly manual stations with very little automation and where the operator's ability had a very large impact. This can also be used to test different load balancing algorithms [5]. Such approach is not so well suited when the line is already quite balanced, and bottleneck shifts are quite dynamic in nature.

Simulation can also stem from lean mapping tools such as value stream mapping (VSM), a visual modelation of value stream operations which acts as a screenshot. In [6], the authors propose making the VSM more useful by using it as an input for a system dynamics framework. In a similar manner, [7] also makes use of VSM coupled with *Siemens Tecnomatix Plant Simulation* software. The authors of [8] follow a similar approach. In these examples, an extensive effort is done in the VSM part and physical modelling alone in order to identify bottlenecks and eliminate them. In this article, the authors will present a more lightweight methodology which does not require any previous mapping nor a significant training effort for people to use it. Instead, the data gathered from the line will automatically reveal its typology making it much less labor intensive. The work [9] also makes use of *Siemens Tecnomatix Plant Simulation* software along with the use of a prediction algorithm based on Petri nets and a fuzzy algorithm. Most of these use cases are based on sets of constraints. We will see later on that the simulation proposed here has as basis the statistical distributions of historical data of processing times and injection rates.

Applications of data-driven discrete-event simulations to a balanced manufacturing line will show that bottleneck identification is not as useful as its application in an unbalanced manufacturing line. To actively act and improve performance of a balanced manufacturing line, bottleneck prediction is a requirement. The work [10] mentions a prediction algorithm based on actual buffer levels. Another approach is used in [11], where complex network theory is used. In [12], authors introduce a prediction method based on the autoregressive integrated moving average algorithm (ARIMA) using as data from a manufacturing execution system (MES) as an input. These authors encourage future research directions with other algorithms that may convey more accurate results, which will be presented in later sections.

Using this literature review as basis, it is clear that there is room for new research and that this is still an area where significant innovation can be pursued and achieved. Here the authors will be dwelling in the area of bottleneck prediction in balanced production lines in real time and assessing how simulation scenarios can be used to infer their usefulness. Real data from a manufacturing line of Bosch Thermotechnology will be used for this use case analysis.

2. Methodology

In this section, a production line of Bosch Thermotechnology (Bosch TT), the business unit facility of residential hot water located in Aveiro, Portugal, is analysed using the tools presented in previous sections. In this Bosch TT plant, three main groups of products are manufactured: water heaters (electric and gas), boilers and heat pumps. Each one possesses a large variety of references, depending on the type of material, technology and functionality. Such variety affects the entire production line flow and, consequently, its global productivity.

2.1. Mathematical Model

There are several ways to formally model workflows related with manufacturing production lines (MPL), as Petri nets variants or activity diagrams variants. However, fundamental issues turn their application on MPL too complex or lacking relevant features [13]. Several of these models assume that tokens are produced or consumed by actors and do not have explicitly the notion of layers of queues and their associated nodes [14]. Tokens are represented by the circles in grey, nodes are represented by the white circles in grey outline and the queues are represented by the rectangles split in three.

Therefore, an abstract model is introduced in order to: (a) precisely model a MPL under clear assumptions; (b) compute a state-space that completely represents the state of the process at each instant of time; (c) produce metrics for bottleneck identification under a minimal level of information; (d) normalize and extend the application of these methods to other use cases; (e) to be the theoretical support of the data-driven discrete-event simulator.

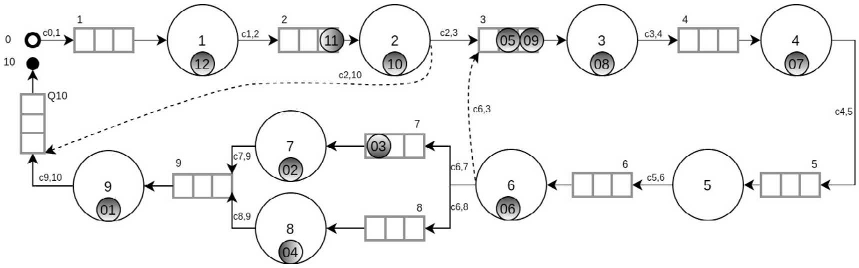

Fig. 1. Queue directed graph adaptation according to the formal assumptions.

Mathematically, any MPL outline will be represented by a generalization of a directed graph that is shown in Figure 1. The queue directed graph (QDG) is simple enough to model all the common situations in a manufacturing line and works with just minimal information being fed from a MES.

Running a scientific review shows that a myriad of methods for bottleneck identification have been introduced and used [15]. The vast majority of them ignores the data collection part altogether. Indeed, some of these methods happen to be derivations of others in order to compensate missing data or data not fitting to the required structure. With this being said, it is a must to define what one may establish as a minimum requirement for such calculations. In the event that other data is available, it may be possible to include it in order to further increase its impact in bottleneck detection.

It seems evident that a bare minimum approach should consist in a set of records with an identifier for the location (node), an identifier for the part/product (token), and a timestamp, corresponding to the departure time of a token from a node.

Using the state-space of a QDG it is possible to calculate the so-called *measuredTime*. This designation stems from the fact that such times are measured from the evolution of queues and departures times from nodes. In other words, for each token, it is the interval between consecutive departures of nodes minus the duration of time the token stayed in the corresponding queue.

The average active period method (AAPM) is a metric which uses the average of the *active* periods of a process and detects fluctuations to it. Formally, this may be defined as:

$$\Psi_{AAPM} = \underset{i\in\{1,\ldots,s_N-1\}}{argmax}\ \Phi_i \ \text{ with } \ \Phi_i = mean(\{\zeta^{node}_{w,i} \in \mathbb{R}^+_0\}), \tag{1}$$

where $\zeta^{node}_{w,i}$ is the active time of a node i at the instant of time w. Commonly, the notion of *active* periods is associated to the processing times obtained by hardware devices. In fact, AAPM can be seen as a family of methods, where variants are obtained by changing the underline notion of *active* period. We consider the following useful variants:

- Average Measured Period Method (AMPM): the average is made over the periods of *measuredTime*.
- Average Queue Period Method (AQPM): the average is made over the periods of time where the queue is occupied by at least one token (so-called *queueTime*).
- Average Total Period Method (ATPM): the average is made over periods measuring the elapsed time from consecutive departures of nodes.

The active period method (APM) is another accurate way to detect bottlenecks in manufacturing systems. The method is highly accurate, not only giving the likelihoods of different processes being the (temporary) bottleneck, but also estimating the improvement of the entire system capacity if the bottleneck(s) are improved. It is also possible to observe the shifting of these bottlenecks over time. The APM may be determined as follows:

$$\Psi_{APM}(w) = \underset{i\in\{1,\ldots,s_N-1\}}{argmax}\ \Phi_{w,i} \ \text{ with } \ \Phi_{w,i} = -w + \underset{t\in\{w,\ldots,s_W+1\}}{argmin}\ \{v^{node}_{t,i} = 0\}, \tag{2}$$

$$\Psi_{APM}(w) = \Psi_{APM}(w) \cup \Psi_{APM}(w-1) \quad \text{if } w > 1 \text{ and } \Phi_{w-1,\Psi_{APM}(w-1)} > 0, \tag{3}$$

where $v^{node}_{w,i}$ gives the number of tokens that at time w are present in the node i and, by notation convenience, we assume that $v^{node}_{s_W+1,i} = 0$ for all $i \in \{1,\ldots,s_N-1\}$. Expression (2) accounts for sole bottlenecks and expression (3) for shifting bottlenecks. As with AAPM, APM can also be seen as family of methods with variants. The following are considered:

- Measured Period Method (MPM): the APM algorithm is applied over periods of *measuredTime*.
- Queue Period Method (QPM): the APM algorithm is applied over the periods of time where the queue is occupied by at least one token.

2.2. Pipeline

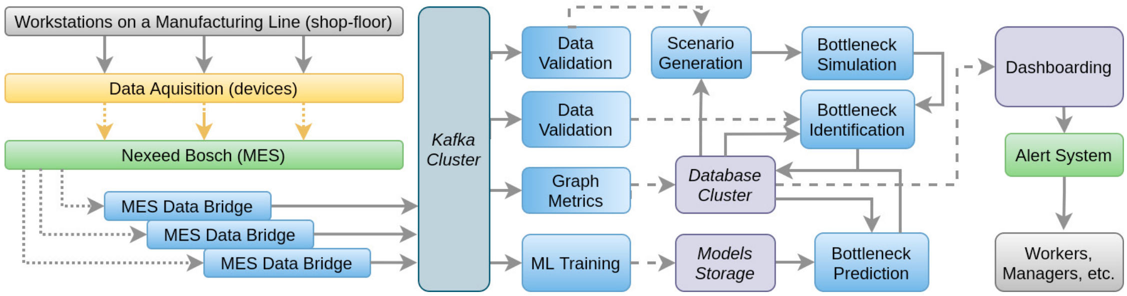

Fig. 2. Pipeline of the solution

Regarding the pipeline for data acquisition, processing and output generation, this is depicted in Figure 2. Workstations in the shopfloor communicate with the Bosch proprietary MES directly. This has been achieved through a long-term digitalization strategy. Older equipments can be replaced or connected through gateways. These equipments usually supply more limited data which can nevertheless be useful. There are some other devices (such as IoT ones) which can also be connected and that add some additional information. A MES service which handles the communications with the machines, receives these requests which are separated by MPLs (see the different MES Data Bridges in Figure 2). This decoupled approach makes it easier to perform maintenance on a MPL without disturbing other MPLs in the same production facility. The requests are then injected into a Kafka cluster. This broker has multiple topics, each one with a specific functionality. Data validation topics make sure that the events arrive in the expected sequence. Any event which violates the constraints is removed. These topics work in parallel in order to avoid disruption to the system's processing performance. Different scenarios can run on top of the data as it will be shown in Section 2.3. This allows us to pinpoint what would be the effect of certain actions. These scenarios are then used as input for bottleneck simulation and thus identification. This will allow establishing some guidelines for improvement. Besides these features, bottleneck prediction is also possible and is used to spot possible hindrances in real time. Managers and operators in the production line are notified via a dashboard which displays warnings and possible improvements. This allows for a more reactive approach which also contributes to an improvement of the line's performance.

2.3. Data Characterization

Production Line 7 was chosen as the use case application for this work. Every year, around 50.000 units of finished goods are produced there in 2 to 3 daily work shifts. The line is also characterized by 18 workstations that may be operated by 8 up to 14 operators with predefined balanced work. All processes should run synchronously, meaning that, in theory, no part queues should exist in between consecutive workstations. All queues are ruled according to first in first out (FIFO). Additionally, there are 2 workstations performing the exact same function in the line, in parallel (branches), that have product quality verification equipment. After some process improvements under the Lean philosophy, this line already possesses an overall equipment effectiveness (OEE) of 90%. Nevertheless, the company's objective is to eliminate performance outliers and to reach an efficiency level close to 100%, which translates into greater production capacity. This is the main reason behind the application of the content of this work.

Table 1. Characterization of the data sets of this use case, where the theoretical cycle time is 200 seconds.

ID	*Shift Type*	*Date*	*Duration*	*Workers*	*Production Quantities*	*Quality %*	*Reworks %*
D01	Random	23/Nov/2020	08:08:21.848	14 (T1)	112	94.07%	0.726%
D02	Good	08/Feb/2021	08:08:33.519	14 (T1)	132	97.98%	1.260%
D03	Good	09/Feb/2021	08:05:13.552	14 (T1)	132	97.78%	0.782%
D04	Bad	17/Feb/2021	07:53:47.996	14 (T1)	98	72.59%	0.813%
D05	Bad	18/Feb/2021	07:59:28.464	14 (T1)	110	81.48%	0.416%

Table 2. Bottleneck results according to multiple bottleneck identification algorithms.

			AAPM Variants			APM Variants	
ID	*Shift Type*	*Production/Target*	*AMPM*	*AQPM*	*ATPM*	*MPM*	*QPM*
D01	Random	112/135	11,12	5	5	11	12
D02	Good	132/135	11,12	5	11,12	12,11	1,11
D03	Good	132/135	11,12	5	11	11,12	12
D04	Bad	98/135	11,12	5	11,5,9,12	11	12
D05	Bad	110/135	11,12	2	11,2,12	11	6

For algorithm validation, the company provided 3 types of data sets pertaining to the production line with the time duration of a complete shift: one perceived as good, another chosen randomly and another perceived as bad. The classification of these data sets was based on the following parameters presented in Table 1.

The notion of quality is defined by a Bosch key performance indicator (KPI) that uses information from the number of all types of reworks, cycle times and overall working times. This parameter is important to calculate the well-known KPI, OEE. Furthermore, these parameters characterise the problem data sets, and later help to explore the results obtained by the different approaches. Table 2 shows the different bottleneck metrics that were used and their results for the data set at hand.

3. Main Results

3.1. Data-Driven Discrete-Event Simulation

To better understand how the system behaves under different conditions, a discrete-event simulator for a general manufacturing line was created based on the Python Package Salabim 21.0.4 (`https://www.salabim.org/manual/`). All distributions are experimental distributions modeled from historical data. The discrete-event actions are ruled by the formal model of a QDG (see Figure 1). The distributions belong to two main classes: distributions of the rates of injection of new parts into the system, i.e., the number of parts being introduced at the start of the manufacturing line, and distributions of the pair operator-machine modeling the processing times of a part. This will act as the basis to run test scenarios as those featured in Table 3.

In order to be able to run the test scenarios mentioned in Table 3, some fitting was done to the real distributions using the Python Package Scikit Learn 0.24.0 (`https://scikit-learn.org/`) and the kernel density estimate technique. The inferred probability density functions (PDFs) have been approximated by histograms with 50 bins as depicted in Figure 3. The histograms for the real distributions have been obtained using the total number of rows for data set D01 while the simulated histograms are obtained through the collection of 1000 random samples from the fitted distributions. The kernel distribution used for the rate of injection of new parts into the system was a Gaussian with a bandwidth of 0.12. The pair operator-machine has also been fitted using a Gaussian kernel with a bandwidth of 0.10.

Simulation results show that, although workstation 11 is the main bottleneck identified by AMPM and MPM, a hypothetical reduction up to 20% of the processing times of workstation 11 would not improve line performance. The main reason is that work is distributed between workstation 11 and workstation 12. On the other hand, these workstations are quality test machines for which processing times are quite difficult to improve since 100% testing is mandatory. Such means that R03 will be excluded from the results analysis, since it is not realistic to be implemented at the MPL at Bosch Thermotechnology.

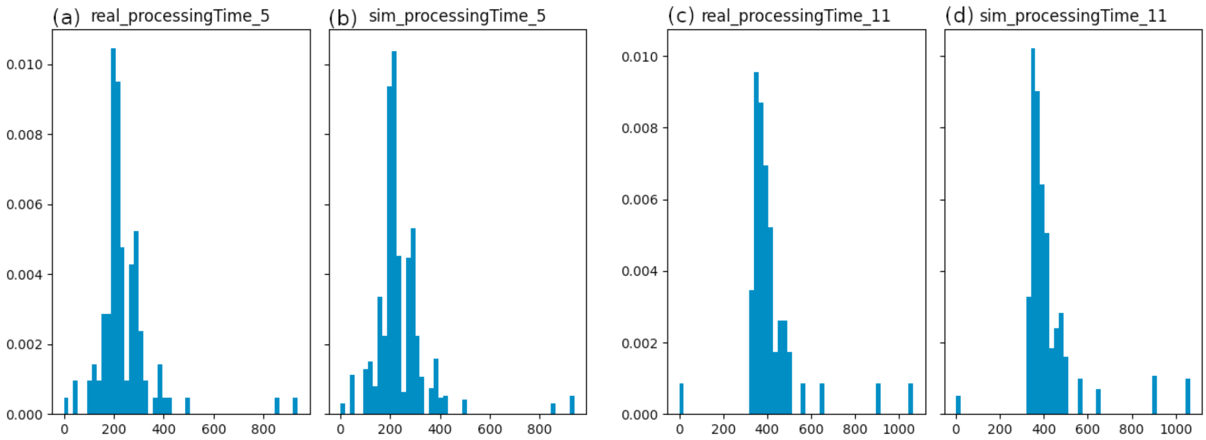

Fig. 3. (a) Real distribution of the processing time in Station 5; (b) Simulated distribution of the processing time in Station 5; (c) Real distribution of the processing time in Station 11; (d) Simulated distribution of the processing time in Station 11. The simulated graphs are made by obtaining 1000 samples from the respective kernel distribution.

Table 3. Relevant experiments acting on the distributions modeling the processing times of workstations.

Scenarios	*Node ID*	*Percentage*	*Description*
R01	all	–	Remove upper outliers of all nodes ($\alpha = 0.5$)
R02	6	10%	Reduce processing times of node 6 in 10%
R03	11	20%	Reduce processing times of node 11 in 33%
R04	5	10%	Reduce processing times of node 5 in 10%

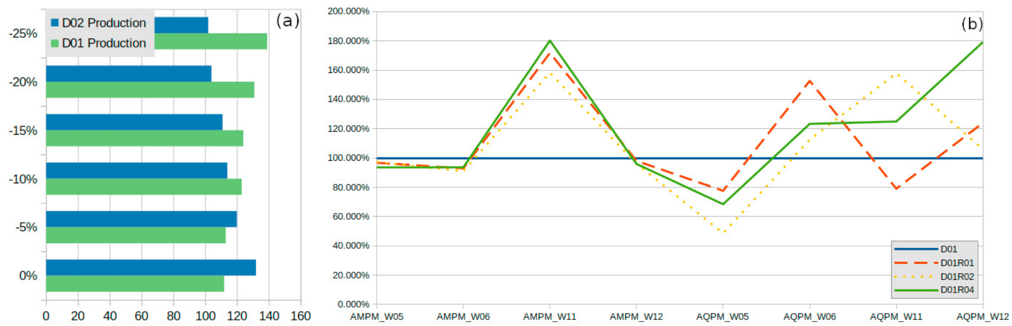

Fig. 4. (a) Effects of reducing the time between injections on D01 and D02. (b) Bottleneck index variation for the original data and scenarios.

Figure 4(a) shows how reducing the time between injections (thus increasing the number of parts entering the line) will affect the production. As expected, the response is not linear. In the case of D01 (random), increasing the number of parts entering the MPL will also imply an increase in the production. However, in the case of D02 (good) which is already near their saturation level, increasing the number of parts entering the line will not improve production. Instead, it increases the bottleneck indexes of other workstations (by increasing the number of parts in queues) which will not translate in better performance. These scenarios show that in balanced lines the rate of injection plays a fundamental role.

Figure 4(b) exhibits the variation of the bottleneck indexes rate for the main bottleneck workstations with respect to the scenarios R01, R02, R04. All scenarios impose more stress in the processing times of workstation 11 and generally reduce the queue of workstation 5. As expected, because workstations 11 and 12 are in a bifurcation, the queue of workstation 12 is generally increased.

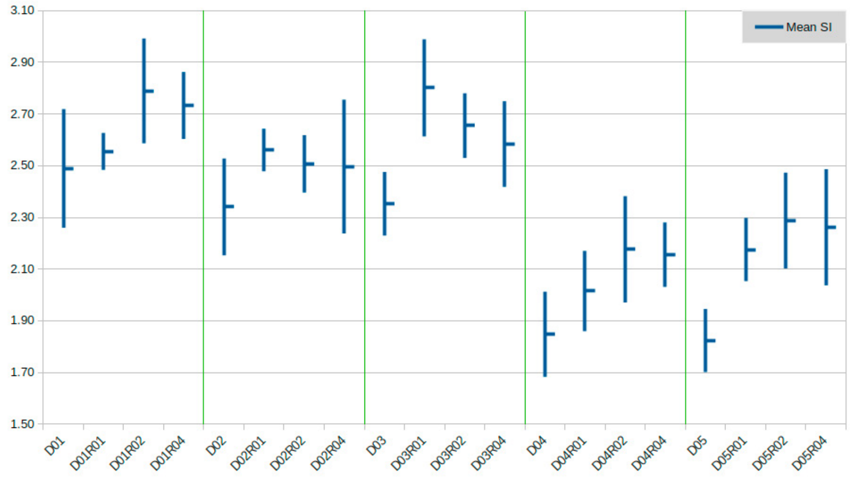

Fig. 5. Smoothness index variation for the original data and scenarios.

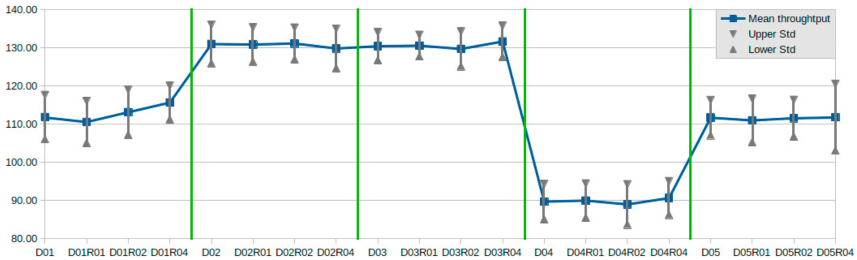

Fig. 6. Productivity results of the original data and scenarios.

Figure 5 shows the smoothness index mean and standard deviation of 12 simulations under the scenarios in Table 3. The smoothness index is widely used as a metric for balancing MPL. Smoothness index is an index that becomes the relative refining index of an assembly line balance. The smaller the smoothness index, the closer the model is to the perfect balance [16, 17]. Since all scenarios increase the smoothness index of the original data set, we may conclude that these scenarios make the line more unbalanced. On the other hand, Figure 6 shows the effect of the scenarios on the production, by averaging the results of 12 simulations. Combining Figures 5 and 6, one can conclude that the line is already quite balanced and variation in production is not derived from unbalanced task distribution but rather by indirect factors such as the human factor. We point out that the data sets are for the same line, the same shift, the same workers but for different production days. This suggests that an active action in the machine-operator pair that is currently the bottleneck may be a valuable approach to improve the performance of a balanced manufacturing line. For such approach by a production manager, bottleneck prediction information is of major importance.

3.2. Bottleneck Prediction

In order to understand the probability of bottlenecks occurring at some point in the future, labels must be defined to which supervised learning techniques can be applied to solve this classification problem. Data set D01 will be used and the setting of the reasonable time period for a bottleneck to occur will be considered. Each row of the data set is a snapshot of the line at a certain point in time, where the two most significant line bottlenecks at the time are identified. To create meaningful labels one can use for bottleneck prediction, it is required to define the minimum prevalence of the bottleneck. If it is shifting all the time, then it makes no sense to consider it for classification purposes or expect that a shift manager would have time to act on it. For this reason, three distinct cut-off durations are considered: 3 minutes, 5 minutes and 7 minutes. If the bottleneck stays active for longer than the cut-off duration, then it is considered as it is and not as a shifting bottleneck. Otherwise, it is a shifting bottleneck or a negligible one, and they are encoded by the label value '-2'. As an example, label '11_12' means that workstations 11 and 12 were the two most significant bottlenecks at the time.

Table 4. Influence of cut-off duration in label distribution.

	Most Representative Labels			Relative Frequency			Other Labels	
Cut-Off Duration	**L1**	**L2**	**L3**	**P1**	**P2**	**P3**	**Number of Labels**	**Remaining Frequency**
3 minutes	11_12	-2	11	23.97%	21.07%	13.63%	51	41.33%
5 minutes	-2	11_12	11	66.86%	12.23%	8.02%	29	12.89%
7 minutes	-2	11_12	11	81.08%	7.22%	5.98%	17	5.72%

The effect of the cut-off duration is exhibited in Table 4. One can see that the cut-off durations significantly affects the class balance of the data sets. When we consider the more extreme case, more than 80% of the bottlenecks do not hold for more than 7 consecutive minutes. This is an expected result since we know that this is a highly balanced MPL. In order to achieve a realistic labelling, we will use as baseline the data set created by the 5 minutes cut-off duration. Shifting bottlenecks represent more than half of the data set and it is also not expected that the shift manager is able to deploy any significant action in a timespan of less than 5 minutes. If this would be the case, by the time the action was defined, the bottleneck would have shifted once again. This is one of the main concerns of the authors: not only to predict bottlenecks, but to do it in a timely manner for people to be able to respond to them. It is of no use to predict these effects, if no appropriate measures can be deployed in time.

To come up with a high performance classifier, several algorithms were tested. The summary of these methods is presented in Table 5. The methods were applied on 41 different features, where 38 of them are part of a state machine indicating whether a node (19 features) or a queue (19 features) is being populated at that particular moment. This particular data set has a total of 1856 rows. Stratified K-fold cross validation on 70% on the data was used to produce results with different validation subsets and create classifier metrics which are obtained over multiple runs. The classifiers were validated in 30% of the data, to minimize overfitting. Tree based methods showed a significantly better performance. The method with higher performance metrics was the Random Forest, but was closely matched by a feedforward artificial neural network (ANN) i.e., the multi-layer perceptron.

Based on these results, we can conclude that bottleneck prediction is possible with very accurate results. Using this information, it is possible to devise an Early Warning System (EWS) which can trigger line managers to act accordingly in a reasonable timeframe.

Table 5. Prediction metrics for the data set D01, with 70% of stratified K-fold cross-validation and 30% of test data (where "m-X" mean "macro X", "w-X" means "weighted X", and the values are the average of 4 runs).

Method	*w-F1-score (strat. K-fold)*	*m-Precision*	*w-Precision*	*m-Recall*	*w-Precision*	*m-F1-score*	*w-F1-score*
Decision Trees	98.3%	88%	97%	92%	95%	89%	96%
Gaussian Naive Bayes	76.5%	41%	91%	62%	59%	37%	67%
Gradient Boosting	99.4%	90%	98%	93%	98%	91%	98%
K-Neighbors	100.0%	73%	98%	81%	97%	75%	98%
Logistic Regression	88.6%	72%	98%	76%	97%	74%	98%
Multi-layer Perceptron / LBFGS	97.8%	98%	99%	94%	99%	95%	99%
Naive Bayes / Multivariate Bernoulli	95.9%	99%	99%	94%	99%	96%	99%
Random Forest[1]	**98.5%**	**98%**	**99%**	**94%**	**99%**	**95%**	**99%**
Regularized Linear models / SGD	82.9%	63%	97%	66%	98%	64%	98%
Support Vector Machine	90.7%	66%	98%	65%	98%	66%	98%
XGBoost	98.3%	98%	99%	94%	99%	95%	99%

[1] Bootstrap=False, criterion=entropy, max_features=0.45, min_samples_leaf=4, min_samples_split=8, n_estimators=100.

4. Conclusions

This work had two main goals: to evaluate in which ways bottleneck simulation allows us to understand if bottleneck identification and direct action on the bottleneck in a highly balanced line has some real improvements, and to validate if bottleneck prediction by classification methods could be used in a line which was already very balanced, with an OEE upwards of 90%. It is common to see in literature performance improvements upwards of 10%, which indicate that the MPL in those studies have quite some room for improvement (i.e. are unbalanced by nature). Devising measures that can be effectively used by shopfloor personnel needs to be the end goal since optimization is only meaningful if it can be effectively applied to the context it relates to. It may not be physically possible to reduce significantly cycle times in lines with these characteristics but there are other meaningful measures that can be adopted. In the simulation scenarios, it became clear that the rate of injection in the first node of the system produces a relevant strain, leading to a significant increase in the throughput of the line as a whole. Based on this, the authors recommend that more experienced workers should be placed both in the first station of the line and also in those where bottlenecks are most likely to occur such as test stations which generally require a higher know-how and experience to operate due to their more complex inner workings. Then, bottleneck prediction frameworks can support shift managers in real time to allow precise curative measures. As a future research direction, the authors would like to recommend assessing the impact that these frameworks may have on a plant's overall performance through the involvement of higher management taking into account that the throughput of a MPL is not the only force at play.

Acknowledgements. The present study was developed in the scope of the Project Augmented Humanity (PAH) [POCI-01-0247-FEDER-046103], financed by Portugal 2020, under the Competitiveness and Internationalization Operational Program, the Lisbon Regional Operational Program, and by the European Regional Development Fund. The first author was partially supported by the Center for Research and Development in Mathematics and Applications (CIDMA), through the Portuguese Foundation for Science and Technology, reference UIDB/04106/2020. The second author was supported by Bosch Termotecnologia.

Data availability statement. The data sets used to obtain the results are confidential information of Bosch's manufacturing system, so they are not publicly available.

References

[1] Lopes, M.J., Rocha, E.M., Georgieva, P., Ferreira, N. (2022) "General Model for Metrics Calculation and Behavior Prediction in Manufacturing Industry – An Automatic Machine Learning Approach". In Chkoniya, V. (Ed.) *Handbook of Research on Applied Data Science and Artificial Intelligence in Business and Industry*. IGI Global.

[2] Brochado, A.F., Rocha, E.M., Almeida, D., de Sousa, A., Moura, A. (2021) "Data-Driven Bottleneck Identification with Minimal Information." *Manuscript submitted for publication.*

[3] Chhalani. C.P., Bhutoria. P., Agarwal. Y., Aravind, R.S. (2021) "Analyzing the Manufacturing Operations and Identifying the Bottlenecks in Food Processing Industry." In: Arockiarajan A., Duraiselvam M., Raju R. (Eds.) *Advances in Industrial Automation and Smart Manufacturing.* Lecture Notes in Mechanical Engineering, 87–97.

[4] Jung, W.K., Park, Y.C., Lee, J.W., and Suh, E.S. (2021) "Simulation-Based Hybrid Optimization Method for the Digital Twin of Garment Production Lines." ASME. *J. Comput. Inf. Sci. Eng.* **21** (3):031007.

[5] Cui, M., and L. Li, L., (2018) "A closed loop dynamic scheduling method based on load balancing for semiconductor wafer fabrication facility." *2018 IEEE International Conference on Smart Manufacturing, Industrial Logistics Engineering (SMILE)*, 1–6.

[6] de Assis, R., de Santa-Eulalia, L,, Ferreira, W,, Armellini, F,, Anholon, R,, Rampasso, I,, and Santos, J. (2021) "Translating value stream maps into system dynamics models: A practical framework." *International Journal of Advanced Manufacturing Technology*, **114** (11-12), 3537-3-550.

[7] Latha, B. M., Raghavendra, N. V., and Ramesh, J. (2021) "Application of value stream mapping using simulation tool in manufacturing assemble line: A case study." Paper presented at the IOP Conference Series: Materials Science and Engineering, 1065(1).

[8] Huynh, B., Akhtar, H. and Li, W. (2020) "Discrete Event Simulation for Manufacturing Performance Management and Optimization: A Case Study for Model Factory." *9th International Conference on Industrial Technology and Management (ICITM)*, 16–20.

[9] Zhang, Y., Zeng, L., Peng, E., Luo, Z., and Zhou, D. (2021) "An intelligent prediction model for bottleneck in production system based on cloud manufacturing." *Proceedings of IncoME-V & CEPE Net-2020*, 237–245.

[10] Wedel, M., Von Hacht, M., Hieber, R., Metternich, J., and Abele, E. (2015). "Real-time bottleneck detection and prediction to prioritize fault repair in interlinked production lines." *Procedia CIRP* **37**, 140–145.

[11] Zhu, F., Wang, R., and Wang, C. (2019) "Intelligent Workshop Bottleneck Prediction Based on Complex Network." *IEEE International Conference on Mechatronics and Automation (ICMA)*, 1682–1686.
[12] Subramaniyan, M., Skoogh, A., Salomonsson, H., Bangalore, P., and Bokrantz, J. (2018). "A data-driven algorithm to predict throughput bottlenecks in a production system based on active periods of the machines." *Computers and Industrial Engineering*, **125**, 533–544.
[13] Eshuis, R., and Wieringa, R. (2003) "Comparing petri net and activity diagram variants for workflow modelling a quest for reactive petri nets". In H. Ehrig, W. Reisig, G. Rozenberg, and H. Weber, editors, Petri Net Technology for Communication-Based Systems, volume 2472. Springer, Berlin, Heidelberg.
[14] Buck, J., and Lee, E. (1993) "Scheduling dynamic dataflow graphs with bounded memory using the token flow model." In *IEEE International Conference on Acoustics, Speech, and Signal Processing* **1**, 429–432.
[15] Roser, C., Lorentzen, K., and Deuse, J. (2015) "Reliable shop floor bottleneck detection for flow lines through process and inventory observations: the bottleneck walk." *Logistics Research*, **8**(1): 1–9.
[16] Baybars, I. (1986) "A survey of exact algorithms for simple assembly line balancing problem."" *Management Science* **32**, 11–17.
[17] Scholl, A., Becker, C. (2006). "State-of-the-art exact and heuristic solution procedures for simple assembly line balancing." *European Journal of Operation Research* **158**, 666–693.