IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received April 9, 2021, accepted April 19, 2021, date of publication April 26, 2021, date of current version August 3, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3075620

Machine Learning-Based Prediction of Missing Components for Assembly – a Case Study at an Engineer-to-Order Manufacturer

PETER BURGGRÄF, JOHANNES WAGNER, BENJAMIN HEINBACH, AND FABIAN STEINBERG

Department of Mechanical Engineering, Chair of International Production Engineering and Management (IPEM), University of Siegen, 57076 Siegen, Germany

Corresponding author: Fabian Steinberg (fabian.steinberg@uni-siegen.de)

ABSTRACT For manufacturing companies, especially for machine and plant manufacturers, the assembly of products in time has an essential impact on meeting delivery dates. Often missing individual components lead to a delayed assembly start, hereinafter referred to as *assembly start delayers*. Identifying the assembly start delayers early in the production process can help to set countermeasures to meet the required delivery dates. In order to achieve this, we set up 24 prediction models on four different levels of detail utilizing different machine learning-algorithms – six prediction models on every level of detail – and applying a case-based research approach in order to identify the model with the highest model quality. The modeling approach on the four levels of detail is different. The models on the coarsest level of detail predict assembly start delayers utilizing a classification approach. The models on the three finer levels of detail predict assembly start delayers via a regression of different lead times and subsequent postprocessing operations to identify the assembly start delayers. After training of the 24 prediction models based on a real data set of a machine and plant manufacturer and evaluating their model quality, the classification model utilizing a Gradient Boosting classifier showed best results. Thus, performing a binary classification to identify assembly start delayers was the best modelling approach. With the achieved results, our study is a first approach to predict assembly start delayers and gives insights in the performance of different modeling approaches in the area of a production planning and control.

INDEX TERMS Production control, assembly, prediction methods, lead time reduction, machine learning, supervised learning, classification algorithms, regression analysis, data analysis.

I. INTRODUCTION

Production companies are facing an ongoing change. They are challenged to assert themselves in international markets and to differentiate their products from other products available on the market in in terms of functionality, quality and price. Furthermore, the logistics performance, such as high adherence to delivery dates or short delivery and lead times, is becoming a key competitive factor [1]–[3]. A typical example for this are machine and plant manufacturers, whose products often consist of a large number of customized components to enable a tailor-made solution for the respective customer [4], [5]. To ensure high adherence to delivery dates and short lead times, the punctual assembly of a product is a central factor, as the product can only be delivered to the customer on time if it has been manufactured and assembled on time. The task of the assembly is to assemble a product of higher complexity with predefined functions with a certain quantity of components in a partly multi-stage process in a given time [6]. The manufacturing processes upstream of the assembly therefore have a direct influence on the performance of the assembly process, since a large number of material flows from different supply chains converge in the assembly process [7]. Often it is not possible to provide the required components on time and simultaneously. Under the assumption that all components required for assembly must be available at the start of assembly, the assembly process is subsequently delayed, if only one component is provided too late [8].

In order to avoid delays of the assembly start and thus to meet delivery dates, it would be helpful to predict potential missing components, we define those components as

The associate editor coordinating the review of this manuscript and approving it for publication was Yanbo Chen.

'assembly start delayers', in early phases of the manufacturing process. By subsequently taking appropriate countermeasures, such as adding extra shifts in production or outsourcing of individual components, the assembly start delayers could be prevented. A central factor for the prediction of assembly start delayers is the lead time of the manufacturing processes upstream of the assembly. The aim of these manufacturing processes is the production of individual components. This is usually done in one or more sequentially executed orders, which in turn consist of one or more operations [9]. The lead time can therefore be considered at three different levels of detail: The component lead time, the order lead time and the operation lead time.

Due to the influence of the lead time on meeting the start of assembly, it seems obvious to predict assembly start delayers based on a lead time prediction. In addition, the lead time prediction can vary in the three levels of detail – component, order and operation lead time. It is also conceivable to predict the assembly start delayers directly via a classification, without a prior lead time prediction. This results in four different options with different level of detail to predict assembly start delayers. Thus, the aim of our paper is to set up a model for the prediction of assembly start delayers and to analyze and systematize the influence of the level of detail of the model on the model quality. As a research method we applied a case study at a machine and plant manufacturer. With the achieved results our paper provides two main contributions:

- We implemented machine learning models based on different algorithms to predict assembly start delayers.
- We identified the coarsest level of detail utilizing a binary classification as the best modeling approach.

Our paper is structured as follows. Section II first introduces the product structure and manufacturing processes in an engineer-to-order environment as well as available approaches for lead time prediction. Section III elaborates the prediction model to identify assembly start delayers utilizing different levels of detail. In section IV the results are presented and discussed. Section V critically reviews the limitations of the applied research method and the results obtained. Furthermore, the implications for further research are derived. Finally, a summary is given in the last section.

II. STATE OF THE ART

The products of machine and plant manufacturers usually consist of several components. These are procured from suppliers or manufactured in the company's own production facilities [7]. Purchased components can be procured on an order-anonymous basis, such as for standard components, and on an order-specific basis, such as for special and drawing components. The procurement of components from suppliers as well as the manufacturing of components in the in-house production belong to processes upstream of the assembly [10]. Since the assembly is a convergence point where several material flows converge, the risk of delays due to missing components is increased.

One established model to analyze converging material flows is the assembly flow element developed by Schmidt [11] with further developments and applications in the assembly flow diagram and supply diagram [10], [11]. In all models, the so-called completer is the last inflow to an assembly order and is therefore the component that was supplied last by the processes upstream of the assembly. A completer can be completed on time – before the planned start date of the assembly, or late – after the planned start of the assembly. A late finalization of a completer therefore leads to a delay in the start of assembly. In this article we define such components as ''assembly start delayer'' (see also chapter 1). Assuming that all components are necessary to start the assembly, the schedule variance of the assembly start delayer determines the earliest possible start date of the assembly. Accordingly, a temporal acceleration of the manufacturing and/or procurement process of an assembly start delayer has the biggest potential to push a delayed assembly start back to the target date. However, the supply diagram is primarily designed to analyze data relating to the past and to identify general issues such as an overall bad assembly supply situation in individual assembly areas. To derive case-specific countermeasures to accelerate individual production orders further analysis is needed.

The lead time of the processes upstream of the assembly has a central influence on meeting the target start date of the assembly and thus on meeting customer requirements. A single component is typically manufactured in one or more sequentially executed orders [9]. Consequently, we distinguish between a component lead time and an order lead time. Further, an order is typically subdivided into individual operations [12], [13]. Thus, we can differentiate between order lead times and operation lead times. The operation lead time is further subdivided into the operation time and interoperation time. As is well known, the interoperation time tends to have a higher share in the lead time than the operation time [12], [14].

In production, lead times are determined by setting up a production schedule taking into account the available production capacities, the technical requirements, the demand dates and the system status [8], [15], [16]. The order sequence is defined according to certain rules in order to calculate start and end dates of the orders at the workstations [17] and is one of the main applications for machine learning (ML) [18]. In addition to the calculation of the lead time based on scheduling, it is also possible to predict lead time directly. By predicting the lead times, completion dates can be determined early and deviations from the schedule can be detected [19]. In the past, many approaches for the prediction of lead times have been established. For example, Cheng and Gupta [20] investigated methods from the field of operations research (OR) such as Constant (CON), Random (RAN) or Total-Work (TWK). With the increasing development of ML, new methods for predicting lead times have emerged (see, for example, [21]–[23]).

A systematic literature review conducted by Burggräf *et al.* [24] has analyzed existing approaches focusing on the prediction of lead times in the research fields of ML and OR and classified them according to the three criteria *data class*, *data origin* and used *method/algorithm*. Looking at the *data class*, the authors identified that the majority of publications examined use order data and information about the system status of the production system (see, for example, [25], [26]). In contrast, material data is rarely used, and employee data is never used to predict lead times. Given the fact that the products of machine and plant manufacturers are typically designed tailor-made to meet the specific customer needs, and that the material data therefore characterize a product, this information should be considered when predicting lead times. The authors in [27], [28] have already used material data utilizing artificial neural networks (ANN) and random forest (RF) for the prediction of lead times, but without using the primarily used information about the system status and machine data and furthermore not in the case of machine and plant manufacturers. According to Burggräf *et al.* [24], there is a lack of prediction models for machine and plant manufacturers that use the primarily used data classes and material data for the prediction of lead times. ANN and RF have already proven successful in including material data in the prediction model. When looking at the *data origin*, the authors of [24] also identified that the use of real data strongly decreases with an increasing number of considered data classes. Thus, with increasing complexity of the prediction model they identified a lack of models using real data.

In addition to the selection of suitable data and a suitable approach, the level of detail of the model is crucial for a successful model application. According to the authors in [29], the level of detail refers to the system that the model represents (e.g. in the case of a model of a production line, the number of machines, components, etc. contained in the model), and not to the exact way in which the model is implemented (e.g. number of data fields used). Consequently, with respect to a model focusing on assembly start delayers, considering the lead time at the level of components, orders or operations would be possible levels of detail. An increase in the level of detail usually leads to a higher model accuracy, but with a degressive characteristic [30]. A 100% accurate model is only possible if the real system is fully mapped, which is typically not achieved [31]. Furthermore, an increase in the level of detail beyond a certain point can also lead to a less accurate model [30]. Therefore, in this work the impact of the degree of detail on the model quality will be investigated.

In summary, currently there is no model available for the prediction of assembly start delayers in the field of machine and plant manufacturers that considers the necessary complexity of the respective industry sector. There are models available for the prediction of lead time, but they are not explicitly used for the prediction of assembly start delayers. In addition, decisive data classes for machine and plant manufacturers such as material data are not used and there are also deficits in considering real data as the base for training the models. Furthermore, the level of detail of the model is not considered in any of the existing approaches.

In this work we will focus on investigating the influence of the level of detail of the modelling on the model quality. For this purpose, the following research question is posed, considering the previous explanations: ''How does the level of detail of the modelling affect the model quality to predict assembly start delayers?'' Considering the argumentation of the authors in [30], we formulate the following working hypothesis: ''The model quality for the prediction of assembly start delayers increases with a finer level of detail.''

III. MODELLING APPROACH

A case-based research approach is used to answer the research question and to investigate the working hypothesis. A case-based research approach is an objective, detailed investigation of a current phenomenon where the researcher has little control over real events [32]. One motivation for a case-based research approach is to gain insights for real needs, for example the needs of manufacturing companies, rather than to develop theories without practical relevance [33]. Furthermore, a case-based research approach has already been successfully applied in the area of lead time prediction (see, for example, [28], [34]–[36]. Although the research question focuses on the prediction of assembly start delayers and not on a lead time prediction, the lead time is one of the central factors for an assembly in time and thus a related research area. Accordingly, a case-based research approach is an appropriate method to answer the research question and to investigate the working hypothesis.

As representative case for the case-based research approach, a machine and plant manufacturer was chosen. A product of this company usually consists of several hundred components and is then used in steel production. An analysis carried out in the company beforehand showed that approx. 95% of the assembly start delayers are components produced in the company's own production. Accordingly, components procured from suppliers were not considered in the developed prediction model. Thus, the scope of this article is limited to the production of components in in-house manufacturing.

A. THE PREDICTION MODEL

To answer the research question, 24 ML-models were created in total, which differ in their level of detail (see Fig. 1) and the utilized ML-algorithm. The models at the different levels of detail are independent of each other, but all pursue the same goal: the prediction of assembly start delays. To achieve this goal, each model comprises various operations. Further, we compared the performance of the different ML-algorithms on each level of detail to identify the best performing ML-algorithm by evaluating the achieved model qualities.

The first and coarsest level of detail (1) is the prediction of assembly start delayers using a binary classification. On this level of detail, components are classified directly as

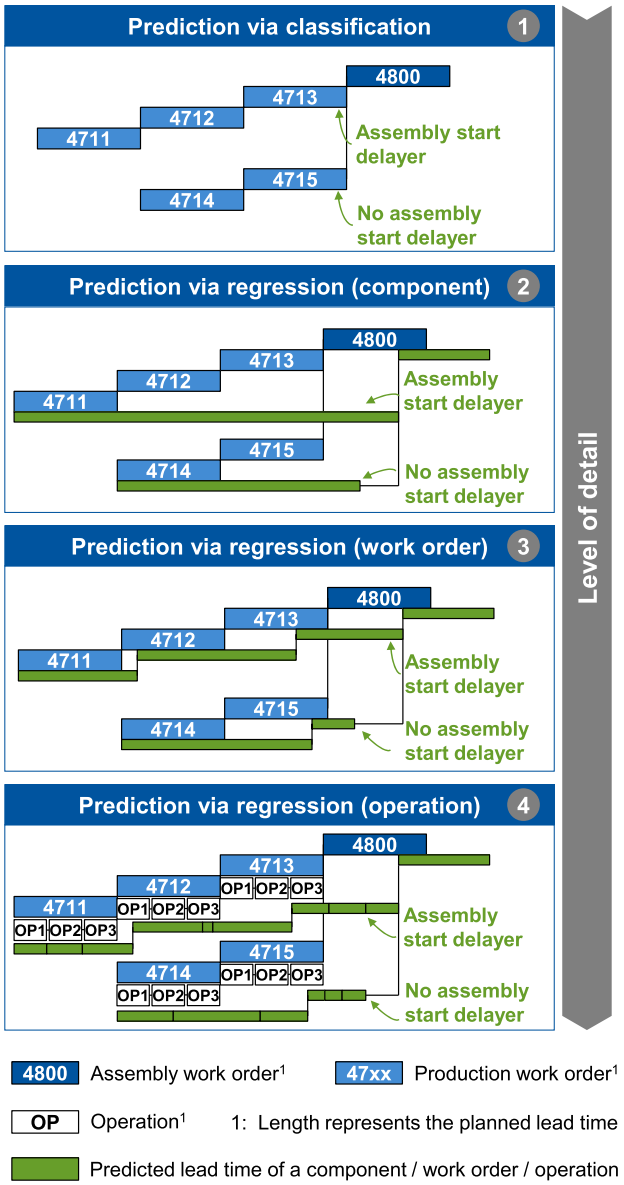

FIGURE 1. Concept of the prediction models on four levels of details.

"assembly start delayer" or as "no assembly start delayer". On the levels of detail (2)-(4) the assembly start delayers are indirectly predicted based on a lead time prediction. With increasing level of detail, a finer granular consideration of the lead time, according to the definition of lead times by the authors in [12] (see chapter 2), is used for the prediction. Consequently, the component lead time is used on the second level of detail (2), the order lead time at the third level of detail (3) and the operation lead time at the fourth and thus finest level of detail (4).

The detailed explanation of the operation principals including the ML-algorithms used on the four levels of detail (see Table 1) is first given for the coarsest level of detail (1). Afterwards the operation principal of the levels of detail (2)-(4) is explained. In the explanation the levels

TABLE 1. ML-algorithms utilized in the considered four levels of detail.

ML-algorithm	Level of detail 1 Classification	Levels of detail (2)-(4) Regression
Support Vector Machine (SVM)	X	
Linear Regression (LR)		X
Decision Tree (DT)	X	X
Random Forest (RF)	X	X
Adaptive Boosting (AdaBoost)	X	X
Gradient Boosting (GB)	X	X
Artificial Neural Network (ANN)	X	X

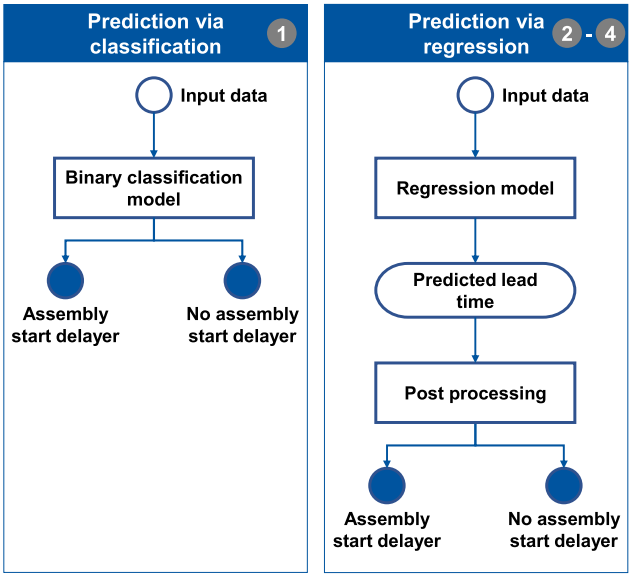

FIGURE 2. Modelling architecture for the classification and the regression approaches.

of detail (2)-(4) are considered together since their operation principal and the ML-algorithms used are analogous and differs only in the considered lead time. The prediction models on all levels of detail were implemented in Python 3.7 utilizing the scikit-learn library.

In the models on the coarsest level of detail (1) (see Fig. 2), we compared the performance of a Support Vector classifier (SVC), a Decision Tree (DT) classifier, a Random Forest (RF) classifier, an Adaptive Boosting (AdaBoost) classifier utilizing a DT-classifier as base estimator, a Gradient Boosting (GB) classifier and an ANN, since they are established approaches for binary classifications [37]–[39]. For the ANN, specifically, a single hidden layer feedforward net with a sigmoid function as activation function and a stochastic gradient descent (SGD) optimizer was applied. The sigmoid function as activation function is particularly suitable for binary classifications [40]. The number of nodes was 46 nodes on the input layer to cover all input features after performing One-Hot-Encoding, 16 nodes on the hidden layer and two

nodes on the output layer to ensure the binary classification 'assembly start delayer' and 'no assembly start delayer'. The number of hidden layers, the number of nodes on the hidden layers and the activation function on the hidden layers were defined by a continuous optimization of the model quality. In detail, we compared different network architectures ranging from one to ten hidden layers with 1 to 100 nodes per hidden layer and different activation functions on the hidden layers such as ReLu function, sigmoid function, tanh function and He function. The best network structure was the above mentioned single hidden layer net. An overview of the optimized hyperparameters used in each of the classification models is given in the appendix in Table 5.

For the classification a slightly modified version of the definition of the assembly start delayers given in chapter 2 is applied: Instead of considering only one single assembly start delayer as a date determining factor for the assembly start according to the definition of Beck [10] and Schmidt [11] and thus assigning the highest potential for improvement to this component, several assembly start delayers were considered for each assembly order. We recommend this extension, since considering only one assembly start delayer is not revealing whether this single one is an outlier or whether a large portion of the components are completed at a similar time. The modified assembly start delayer classification was defined as follows: If the schedule variance of a component is larger or equal to 80% of the maximum schedule variance of all components of an assembly order, which is the schedule variance of the actual assembly start delayer, then this component is considered as an assembly start delayer.

The models on the levels of detail (2)-(4) (see. Fig. 2) are based on a lead time prediction using a regression approach. Here we compared the performance of a linear regression (LR), a DT-regressor, a RF-regressor, an AdaBoost-regressor utilizing an DT-regressor as base estimator, a GB-regressor and an ANN, since they are established approaches for regression which have already been successfully applied in lead times prediction [24]. For the ANN, specifically, a single-hidden-layer feedforward net with a rectified linear unit as activation function utilizing a Keras regressor was applied. This activation function is particularly suitable for the prediction of lead times, since its output is limited to positive values only (negative lead times are not plausible), and it is an established activation function for regression models in ML [41]. The number of nodes was 46 nodes on the input layer on the level of detail (2) and 45 nodes on the levels of detail (3) and (4) to cover all input features after performing One-Hot-Encoding, 12 nodes on the hidden layer and one node on the output layer to enable the lead time as output of the regression model. The network architecture was also continuously optimized by comparing different numbers of hidden layers, number of hidden nodes on the hidden layers and activation functions on the hidden layers. In detail, we followed the same procedure as for the classification models and compared network architectures ranging from one to ten hidden layers with 1 to 100 nodes per hidden layer and different activation functions on the hidden layers such as ReLu function, sigmoid function, tanh function and He function. The best network structure was the above mentioned single hidden layer net. An overview of the optimized hyperparameters used in each of the regression models is also given in the appendix in Tables 6 - 8.

However, based on the predicted lead time only, it is not yet possible to make a statement about a potential assembly start delayer. In order to be able to identify the assembly start delayers at the levels of detail (2)-(4), additional subsequent operations were implemented (cf. "postprocessing" in Fig. 2): A completion date was calculated individually for each component, starting from a fictitious start date and using their respective predicted lead times. The fictitious start date was assumed to be either the target start date of the component or, if the target start date was already in the past at the time of creation of the corresponding production order and thus could not be realized, the date of the order creation and thus the completion of order planning. Typical examples of components for which the target start date at the time of order creation is in the past are supplement orders. At the levels of detail (3) and (4), an intermediate step was performed before calculating the completion date based on a fictitious start date: All predicted lead times (order lead times or operation lead times) of the respective component were summed up to a component lead time. Subsequently, at all three finer levels of detail (2)-(4), the assembly start delayers were determined according to the modified assembly start delayer logic based on the prior calculated completion dates of all components of an assembly order. In detail we utilized the formula

$$Class_i = \begin{cases} ASD;\, SV_{i,j} \geq 0, 8 * SV_{j,max} \wedge SV_{i,j} > 0 \\ NASD;\, SV_{i,j} < 0, 8 * SV_{j,max} \vee SV_{i,j} < 0 \end{cases} \quad (1)$$

to assign one of the two classes "assembly start delayer" (ASD) and "no assembly start delayer" (NASD) to every component i, where $SV_{i,j}$ is the schedule variance of component i of assembly order j, calculated by

$$SV_{i,j} = CD_{i,j} - TSD_j \quad (2)$$

where $CD_{i,j}$ is the calculated completion date of component i of assembly order j based on the predicted lead time of the prediction model and TSD_j the target start date of assembly order j, and $SV_{j,max}$ the maximum schedule variance of all components of assembly order j, calculated by

$$SV_{j,max} = CD_{j,max} - TSD_j \quad (3)$$

where $CD_{j,max}$ is the latest completion date of all components of assembly order j.

After performing the subsequent operations, the output of the models on the three finer levels of detail (2)-(4) is also "assembly start delayer" or "no assembly start delayer".

The applied procedure in the regression models, first to predict a lead time and, based on this, to calculate the completion dates of the components based on a fictitious start date, seems to be a cumbersome process. One could also

think of directly predicting the completion dates without the workaround of predicting lead times. However, a direct prediction of the completion dates of the components is not possible with a supervised learning approach: Supervised learning is based on historical training data. If completion dates were directly predicted based on this historical training data, all completion dates would be in the past and not in the future. Therefore, predicting lead time is used as a workaround, as lead times depend on technical and organizational factors such as the available capacity or the required processing order, whereas they are usually independent of the considered date.

B. THE DATA MODEL

The data model on all four levels of detail consists of the four data classes *order data*, *machine data*, *material data* and *system status* and thus follows the recommendation of the authors in [24]. For the data acquisition, we followed the procedure of Fayyad *et al.* [42] and Han *et al.* [43] consisting of the four steps *selection*, *pre-processing*, *reduction* and *transformation*. In the *selection*, the data for predicting the assembly start delayers was selected from the Enterprise Resource Planning System (ERP) and Advanced Planning and Scheduling System (APS) of the machine and plant manufacturer under consideration. According to the recommendation of the authors in [44] we included experts from the machine and plant manufacturer in this process. We considered all data fields that, based on the experience of the experts, have an impact on production orders meeting the target start date of the assembly and thus should be included in the prediction model. In addition, further data fields were selected which the experts classified as only potentially relevant. The data export included assembly orders, the corresponding production orders and operation as well as information on the material and the systems status. The period under review was set to one year. In the *preprocessing* which followed the selection, the data set was corrected by formatting individual data fields and cleaning up data gaps. Here, we also included the company's experts to avoid deleting data e.g. with data gaps. In the preprocessing we also analyzed the data structure and combined the different raw data tables, which were basically separate csv-files, to one data model. For this, we set up an entity-relationship diagram (see Fig. 3) enabling us to identify the primary keys, which are the prerequisite for the connection.

After completing the data preprocessing, in the *reduction*, from the large number of data fields selected by expert knowledge, those that have an influence on the start of assembly were selected. For this purpose, a correlation analysis was performed according to the recommendation of the authors in [45]. In the final step, the *transformation*, the data fields were modified in order to define suitable features for the four prediction models. Here we applied typical methods such as discretization, decomposition, normalization, and aggregation (see, for details, [43], [46]). In the following, the transformation of the data fields 'workstation type' from the data class *machine data* and the data field 'order creation-delay' from the data class *order data* are explained as examples. Initially, the data field 'workstation type' was a free text field with many different characteristics. For the definition of the feature, the workstations were grouped according to their processing type. For example, all machines that perform a turning operation were grouped into 'turning machine'. The data field 'order creation-delay' has been calculated based on the deviation between the target start date of an order and the actual date of the order creation and thus indicates a delay in the order creation. In total, the application of the methodology of Fayyad *et al.* [42] and Han *et al.* [43] results in 17 features, although not all features are applied at all levels of detail (see Table 2).

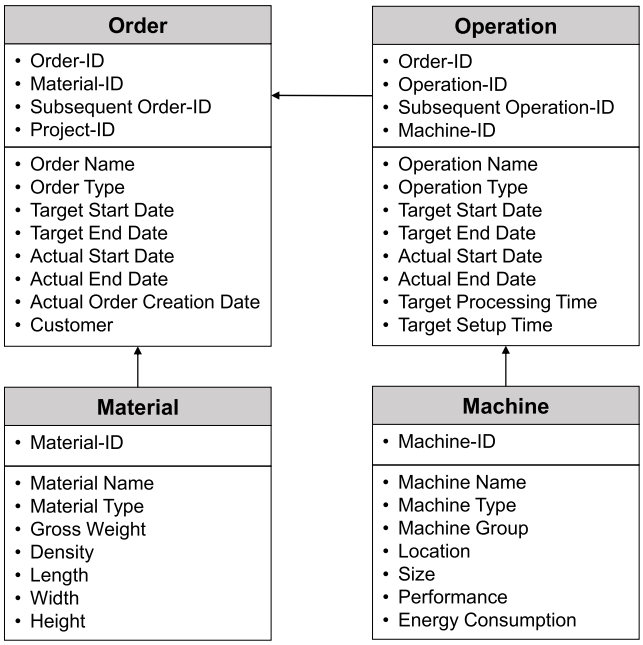

FIGURE 3. Entity-relationship diagram with an excerpt of available features per data table.

After performing One-Hot-Encoding for each level of detail we increased the number of features to a total of 375 features on the levels of detail (1) and (2) and 374 features on the levels of detail (3) and (4) due to many values in the categorical features. We further evaluated the dependence between the features by creating a 375×375 correlation matrix in form of a lower triangular matrix for the coarsest level of detail leading to 71.631 individual correlation coefficients. To get an overview of the overall correlation in our dataset we assigned all correlation coefficients to bins of different correlation strengths following the established rules for interpreting correlation coefficients [47], [48] leading to a total of five bins. Finally, we calculated the share of the individual bins in the number of all correlation coefficients (cf. Table 3). Based on the overview, we identified that 1.4 % of all correlation coefficient show at least a moderate correlation. This indicates an existing dependency between our features. Thus, a Principal Component Analysis (PCA)

TABLE 2. Features used in the prediction model.

Data class	Feature	Level of detail			
		1	2	3	4
Order Data	Target lead time	X	X	X	X
	Total number of orders	X	X		
	Total number of operations	X	X	X	
	Target processing time				X
	Target setup time				X
	Order creation-delay	X	X	X	X
	Priority	X	X	X	X
	Operation type	X	X	X	X
	Dispatcher	X	X	X	X
	Number of production areas a component/order passes through	X	X	X	
System status	Number of orders in system	X	X	X	X
Material data	Gross weight	X	X	X	X
	Component name	X	X	X	X
Machine data	Production area	X	X	X	X
	Workstation type	X	X	X	X
	Workstation number	X	X	X	X
	Workstation capacity	X	X	X	X

TABLE 3. Correlation between features before and after PCA following the bin sizes of [47], [48].

Bin	Correlation Coefficient	Before PCA	After PCA
Very high correlation	1.0 to 0.9 (-1.0 to -0.9)	0.3 %	0.0 %
High correlation	0.9 to 0.7 (-0.9 to -0.7)	0.6 %	0.0 %
Moderate correlation	0.7 to 0.5 (-0.7 to -0.5)	1.5%	0.0 %
Low correlation	0.5 to 0.3 (-0.5 to -0.3)	6.5%	0.0 %
Negligible correlation	0.3 to 0.0 (-0.3 to -0.0)	91.1%	100.0%

was performed to reduce the dependencies between the features and to ensure a good model quality. The improvement of the model quality by using a PCA has already been demonstrated in other studies (see, for example, [49]). By applying a PCA, we identified 46 principal components (PC) on the levels of detail (1) and (2) and 47 PC on the levels of detail (3) and (4) as an appropriate number of PC. After performing the PCA, we again performed a correlation analysis and assigned all correlation coefficients to the equal five bins (cf. Table 3) showing that the PCA eliminates the dependency between the features.

Our final dataset consisted of 356 assembly orders comprising 1,506 components supplied by the in-house production. Of course, the in-house-components were only a subset of all components needed for assembly. Components purchased from suppliers were excluded based on an analysis previously performed by the machine and plant manufacturer showing that the in-house-components are predominantly responsible for a delayed start of the assembly. These 1,506 in-house-components are manufactured by a total of 3,187 production orders comprising 15,772 operations. With our modified definition of an assembly start delayer we had a total of 24 % ''assembly start delayers'' and 76 % ''non-assembly start delayers'' of all in-house-components.

C. TRAINING/TEST SPLIT AND PREDICTION TIME

After defining the data model and before training the prediction models, the data set was divided into training and test data with a ratio of 80% training to 20% test data. In selecting the ratio, we followed established ratios. These are approx. 75%-80% training data to 25%-20% test data [50]. When splitting the data, we ensured that the components of one assembly order are not separated. Thus, the data-subsets (training and testing) always contain the complete bill of materials of an assembly order produced in in-house production including all corresponding production orders and operations. By this, we ensured that the prediction model is subsequently able to predict the actual assembly start delayers.

The time of application of the prediction models (prediction time) and thus the time of gaining knowledge about potential assembly start delayers should be as early as possible within the production process, so that companies have as much time as possible to implement acceleration measures. For the four models within this study, we set the date of order creation and thus the completion of order planning as prediction time. At this point, all necessary information, such as bill of materials, operations and machine assignments are available.

D. EVALUATION OF THE MODEL QUALITY

To evaluate the model quality of all models we applied a confusion matrix, since the output on all four levels of detail is the binary output ''assembly start delayer'' or ''no assembly start delayer''. The evaluation of the model quality with a confusion matrix is an established method and has already been demonstrated in other studies (see, for example, [51], [52]). Following the authors in [53] we used the Matthew's correlation coefficient (MCC) [54] as an evaluation metric, since it considers the balance ratios of all four confusion matrix categories and thus is the most informative metric to evaluate a confusion matrix. Considering the MCC also ensured that our model was not just predicting the majority class in our data set, which is ''no assembly start delayer''. Furthermore, we considered the F-score, precision and recall [52] as evaluation metrics, since they focus on the prediction of positives (assembly start delayer) only, which is the most important category in our case of interest. In the F-score we weighted the recall twice as high than the precision, deviating from a regular harmonic mean. This weighting is based on the assumption that it seems more important to identify as many of the actual assembly start delayers as possible, in case of doubt even more than exist, and to define acceleration

measures for them, than not to identify individual assembly start delayers at all. By evaluating each prediction of the four different levels of detail using these metrics, the dependence of the model quality on the level of detail of the modeling can be determined.

Besides considering the metrics MCC and F-score only, one could think to consider the model accuracy, which is the portion of correctly predicted assembly start delayers and non-assembly start delayers to all predictions, as well. Nevertheless, the model accuracy is not a suitable metric for our study, as there is an imbalance between assembly start delayers and non-assembly start delayers (in our dataset 24 % to 76 %). This is due to the definition of assembly start delayers, according to which the assembly start delayers are only a small portion of all components of an assembly order. A typical example would be an assembly order consisting of 100 components, 5 of which are assembly start delayers. If the model would predict ''non-assembly start delayers'' for all components, the accuracy would be 95 %. Nevertheless, none of the assembly start delayers, thus none of the critical components, would have been identified and consequently the goal of the prediction model would not have been reached. With the original definition of the completer given by the authors in [10], [11], according to which there is only one assembly start delay per assembly order, this imbalance would have been even stronger. Therefore, we only considered the MCC and F-score as suitable metrics to evaluate the model quality for the prediction of assembly start delayers.

In summary, we implemented and compared 24 prediction models on four different levels of detail (six models per level). The target was first, to identify the ML-algorithm reaching the highest model quality per level of detail and based on that, to identify the dependence of the model quality on the level of detail of the modeling. The models on the coarsest level of detail (1) utilizing a classification to directly predict assembly start delayers differ strongly from the models on the finer levels of detail (2)-(4) utilizing a lead time prediction based on a regression to predict assembly start delayers. The models on the three finer levels of detail (2)-(4) only differ in the utilized ML-algorithms and the considered lead time, which becomes increasingly finer with the level of detail: from the component lead time to the order lead time to the operation lead time. In all 24 models the output was the binary classification ''assembly start delayer'' or ''no assembly start delayer''. To enable the binary classification for the regression models on the levels of detail (2)-(4) the model output was postprocessed. As metrics to evaluate the model quality we used the MCC, F-Score, precision and recall based on a confusion matrix.

IV. RESULTS

After the definition of the concept, the data model and the model evaluation, we trained the prediction models on our data set. Hyperparameter tuning was performed to optimize the model quality in the best possible way (cf. Tables 5 - 7).

TABLE 4. Reached model qualities of all prediction models.

Level of detail (1)				
Model	MCC	F-Score	Precision	Recall
SVM	0.55	72 %	56 %	78 %
DT	0.56	68 %	60 %	71 %
RF	0.52	60 %	64 %	59 %
GB	**0.65**	**75 %**	**67 %**	**75 %**
AdaBoost	0.58	71 %	61 %	74 %
ANN	0.45	65 %	43 %	76 %
Level of detail (2)				
Model	MCC	F-Score	Precision	Recall
LR	0.22	38 %	33 %	40 %
DT	0.30	43 %	40 %	44 %
RF	0.10	26 %	24 %	27 %
GB	0.22	35 %	35 %	35 %
AdaBoost	0.13	35 %	24 %	40 %
ANN	**0.29**	**48 %**	**36 %**	**53 %**
Level of detail (3)				
Model	MCC	F-Score	Precision	Recall
LR	0.27	41 %	38 %	42 %
DT	0.32	42 %	43 %	42 %
RF	0.22	35 %	34 %	35 %
GB	0.21	33 %	34 %	33 %
AdaBoost	0.13	44 %	26 %	44 %
ANN	**0.32**	**51 %**	**40 %**	**51 %**
Level of detail (4)				
Model	MCC	F-Score	Precision	Recall
LR	0.35	53 %	39 %	58 %
DT	**0.41**	**58 %**	**42 %**	**65 %**
RF	0.38	54 %	42 %	58 %
GB	0.34	49 %	40 %	52 %
AdaBoost	0.36	57 %	38 %	65 %
ANN	0.34	50 %	41 %	53 %

Subsequently, the confusion matrices were created for each model on the different levels of detail to determine the model quality. Based on the respective confusion matrix, the metrics MCC, F-score, precision and recall were calculated for each model (cf. Table 4). These metrics enabled us to determine the best performing ML-algorithm on each level of detail and the dependence of the model quality on the different level of detail.

Evaluating the metrics on the various levels of detail, it is particularly noticeable that the best result was achieved at the coarsest level of detail (1): The direct prediction of assembly start delayers utilizing a GB-classifier achieves the highest

model quality with an MCC of 0.65 and an F-score of 75 %. With MCCs of approx. 0.3 to 0.4 and F-scores of approx. 50 % to 60 %, the best models on the finest three levels of detail (2)-(4) do not reach the result of the best model on the coarsest level of detail (1). Considering the levels of detail (2)-(4) the MCC, F-Score, precision and recall of the best performing model on each level increases with a finer level of detail. Thus, the model quality of the best regression models increases with a finer level of detail but still lower than the model quality of the best classification model, which was on the coarsest level of detail (1).

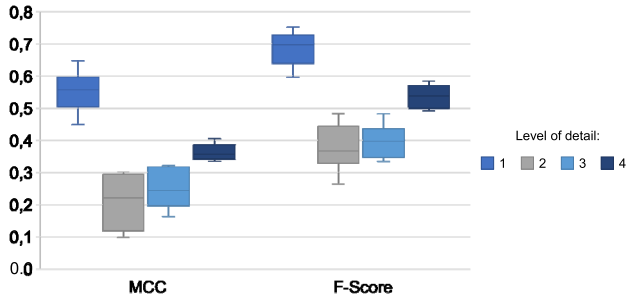

FIGURE 4. Boxplot of MCC and F-Score for all prediction models on each of the four levels of details.

Furthermore, we created boxplots for the four levels of detail to visualize the spread of all models in the MCC and F-Score within the respective levels of details and the dependence of the model quality on the level of detail (cf. Fig. 4). It is particularly noticeable, that the models on the level of detail (1) strongly differ from the models on the levels of detail (2) - (4) emphasizing that the classification approach outperforms the regression approaches. In addition, the models on the level of detail (2) differ only slightly from the models on the level on detail (3), whereas the models on the level of detail (4) differ more strongly from the levels of detail (2) and (3). That emphasizes again an increasing model quality for the regression model with a finer level of detail. Furthermore, there are no outliers in any of the boxplots. Consequently, none of the prediction models within the four levels of detail differs significantly from the other models on the respective level of detail. Nevertheless, the decreasing spread from the level of details (2) to level of detail (4) indicates that with a finer level of detail the model quality of the regression models converges. One possible explanation for the decreasing spread in the regression models is the increasing amount of training data with a finer level of detail – from components to orders to operations – leading to a more solid data base for training the models.

Considering all achieved model qualities, the working hypothesis cannot be confirmed, since – contrary to the working hypothesis – the best result is achieved at the coarsest level of detail (1). Considering the three finer levels of detail only, the model quality is increasing with a finer level of detail, but still, the model quality is below the results on the coarsest level of detail (1).

Consequently, we analyzed possible explanations for the different behavior of the models in our approach regarding their achieved model quality. One possible explanation for the lower model quality on the finer levels of detail (2)-(4) could be, that errors occurring during the prediction of the lead times are cumulated in the postprocessing operations: For each assembly order, n completion dates are predicted according to the number of components. Subsequently, the final output "assembly start delayer" or "no assembly start delayer" is calculated for each component. This calculation of the assembly start delayers is based on the calculated completion dates of all n components and thus includes the errors of all individual calculations of the completion dates. Furthermore, each calculated completion date is composed of a fictitious start date and a predicted lead time. Both the fictitious start date and the predicted lead time can be subject to errors. In reality, an order can also start on a start date other than the fictitious start date, which can result in a deviation between the predicted completion date and the actual completion date. In summary, the cumulation of errors is one potential explanation for the lower model quality at the finer levels of detail (2)-(4). This explanation is supported, for example, by the authors of [55], who compare simpler with more complex prediction models in their study. Contrary to their initial assumption that under certain conditions more complex prediction models are more accurate, they conclude that simpler models achieve better results. Thus, we recommend for future models for the prediction of assembly start delayers to follow the structure at the coarsest level of detail.

V. CRITICAL REFLECTION OF RESULTS, LIMITATIONS, AND IMPLICATIONS FOR FURTHER RESEARCH

Missing individual components leading to a delayed assembly start is often an issue for engineer-to-order manufacturers. Thus, the object of consideration in our study tends to be of general nature. Consequently, the case-based research approach as applied research methodology induces legitimate questionability of its comprehensiveness and representativeness for engineer-to-order manufacturers in general. Our results obtained rely on one single exemplary case which might not be representative for all engineer-to-order manufactures. Thus, it might not be generalizable to all cases. Furthermore, due to the defined scope of this study, the considered input features of the prediction models rely on the interviewed experts of the observed company. When transferring the approach to other cases, a new identification of the considered input features might be necessary. Although we tried to overcome these limitations by considering established input features for prediction of lead times in previous studies, further relevant input features might have remained undetected or the considered input features might not be relevant in other cases. Consequently, in future works, the results should be verified with other cases or from a generic point of view. Nevertheless, in the research areas of machine learning and lead time prediction case-based-research is an established research method as it provides necessary training data (see, for example, [28], [34]–[36]).

In our case study, the F-scores of all models with a maximum of 75 % and MCC with a maximum of 0.65 were still low and not fully reliable for a practical application. Reasons for not reaching fully reliable ML-models with F-scores close to 100 % and MCCs close to 1.00 are typically a combination of the considered modeling approach, the ML-algorithm, and the data base [30], [56]–[58]. To reach the current values of our best F-score and MCC we applied four different modeling approaches and several different ML-algorithms including different structures of the ANN and tuning of the hyperparameters. Thus, we infer, that neither a further optimization of the modeling approach nor the ML-algorithm used leads to a significant improvement of the model quality. One possibility to further improve the model quality could be to enhance the data basis used for training the models, as the data base also has an essential influence on the model quality [58], [59]. In manufacturing processes, especially at machine and plant manufacturers, there are typically many reasons for a delay such as missing raw material, problems when setting up the machine, machine downtimes, issues during the execution of an operation, rework, quality problems with a certain material, or also nonproduction related reasons such as issues in a global supply chain or even the weather (see, for example, [1], [60], [61]). Thus, to ensure a fully reliable model, all the potential disruptions would need to be considered in the machine learning model, and consequently, the data base needs to cover all that information about the respective disruptions as well. In our case study, with a selection of all available order data, machine data, material data and system information, we cover a subset of all information about potential disruptions only. Accordingly, we assume that considering further information about typical disruptions occurring at the exemplary chosen machine and plant manufacturer such as detailed information about the production process at the raw material supplier or maintenance data, could significantly improve the model quality. Consequently, we encourage further studies to consider additional data fields about potential disruptions when setting up a model predicting assembly start delayers to further optimize the model. Without an improvement of further approx. 15-20 % in F-score, the model will not be suitably usable for manufacturing companies. Nevertheless, our study is a good starting point in the research area of predicting assembly start delayers analyzing essential basics regarding the modeling approach for future studies.

A further area for future work could be the provision of background information on the identified assembly start delayers. The current models are only able to identify the assembly start delayers. However, there is no information on the reasons for the occurrence of an identified assembly start delayer given that would explain why the component was supplied late. In order to be able to prevent a potential assembly start delayer by defining suitable counter measures, information about the causes of the delay is of immense importance. Thus, the investigation of how methods from the area of explainable AI can support the provision of background information in the prediction of assembly start delayers could be a potential further research field.

Besides the considered limitations and implications for further research, we could satisfactorily answer our initially stated research question ''how does the level of detail of the modelling affect the model quality to predict assembly start delayers''. In our case study, we could show, that the level of detail of the modelling significantly affects the model quality. The best modelling approach in our case study was to apply a classification model to predict assembly start delayers. Thus, the target of our study was achieved.

VI. CONCLUSION

Adherence to delivery dates is a decisive factor for manufacturing companies to assert themselves in globalized markets. A central aspect to meet delivery dates is an assembly of a product on time. Delays in the processes upstream of the assembly such as the in-house production of individual components can have a negative effect on the adherence to delivery dates. In order to prevent delays in the processes upstream of the assembly, in this work a supervised learning model to predict missing components for the assembly start, so-called assembly start delayers, in early phases of the production process was developed. Here we analyzed the level of detail of the prediction model since it can have a significant impact on the model quality. An increase in the level of detail usually leads to a higher model accuracy, but with a degressive characteristic [30]. Thus, we formulated the following working hypothesis: ''The model quality for the prediction of assembly start delayers increases with a finer level of detail.'' In order to verify the working hypothesis, in total 24 ML-models were created, which differ in their level of detail and the utilized ML-algorithm, but with the prediction of assembly start delayers as their common target. Here a case-based research approach was applied. As an exemplary case for this research approach, a machine and plant manufacturer was chosen and real-world data was applied.

The model architectures of the models on the four levels of detail are different. The models on the coarsest level of detail predict assembly start delayers utilizing a classification approach. The models on the three finer levels of detail predict assembly start delayers based on a prior lead time prediction via a regression and subsequent postprocessing operations. The regression models differ in the lead times considered. A finer level of detail corresponds to a finer consideration of the lead time. Specifically, the component, order and operation lead times were considered. In the subsequent postprocessing operations, the assembly start delayers were identified based on the predicted lead times. Finally, the output of all 24 prediction models on the four levels of detail was the binary classification ''assembly start delayer'' or ''no assembly start delayer'' for every component. To evaluate the model quality of all 24 models a confusion matrix was created and the metrics MCC, F-score, precision and recall were calculated.

The comparison of the model qualities at the four levels of detail showed that, contrary to the working hypothesis, the model on coarsest level of detail – the classification approach – had the best model quality. In contrast, an increase in model quality with a finer level of detail was evident within the regression models. In our study, in total, a finer level of detail did not lead to the best result obtained. Consequently, the working hypothesis could not be confirmed. As a possible explanation for the lower model quality on the three finer levels of detail we identified a cumulation of errors occurring during the prediction of the lead times in the postprocessing operations.

In total, we successfully implemented 24 ML-models to predict assembly start delayers and gave insights in the performance of different modeling approaches. Such prediction models can be useful to identify assembly start delayers in early phases of the manufacturing process and to enhance the delivery performance machine and plant manufactures if a sufficiently high model quality is achieved.

TABLE 5. Hyperparameters of the prediction models on level of detail (1).

Model	Hyperparameters					
	c	penalty	loss	max iter	dual	
SVM	1	l2	hinge	1000	True	
	min samples split	min samples leaf	max features	max depth	learning rate	n estimators
DT	2	1	None	None	-	-
RF	2	2	70	None	-	500
GB	15	6	80	7	0.2	2000
Ada-Boost	30	15	50	3	1	800
	momentum	weight decay	dampening	epochs	learning rate	
ANN	0.9	0	0	450	0.01	

TABLE 6. Hyperparameters of the prediction models on level of detail (2).

Model	Hyperparameters					
	normalize	fit intercept	copy x	n_jobs		
LR	False	True	True	None		
	min samples split	min samples leaf	max features	max depth	learning rate	n estimators
DT	12	9	10	10	-	-
RF	12	8	auto	10	-	60
GB	2	3	10	7	0.8	10
Ada-Boost	25	12	22	5	0.7	30
	verbose	batch size	epochs			
ANN	1	64	450			

APPENDIX

The hyperparameter used in the prediction models were optimized utilizing a grid search and cross validation algorithms (GridSearchCV) from scikit learn. Tables 5 to 8 summarize the utilized hyperparameters in the different models on the four levels of detail.

TABLE 7. Hyperparameters of the prediction models on level of detail (3).

Model	Hyperparameters					
	normalize	fit intercept	copy x	n_jobs		
LR	False	True	True	None		
	min samples split	min samples leaf	max features	max depth	learning rate	n estimators
DT	5	1	40	5	-	-
RF	10	9	auto	None	-	140
GB	2	3	None	2	0.5	10
Ada-Boost	5	2	30	7	10	90
	verbose	batch size	epochs			
ANN	1	64	500			

TABLE 8. Hyperparameters of the prediction models on level of detail (4).

Model	Hyperparameters					
	normalize	fit intercept	copy x	n_jobs		
LR	False	True	True	None		
	min samples split	min samples leaf	max features	max depth	learning rate	n estimators
DT	2	7	30	None	-	-
RF	10	3	auto	None	-	100
GB	8	7	None	7	2	10
Ada-Boost	2	8	33	None	2	40
	verbose	batch size	epochs			
ANN	0	32	600			

REFERENCES

[1] P. Burggräf, J. Wagner, K. Lück, and T. Adlon, "Cost-benefit analysis for disruption prevention in low-volume assembly," *Prod. Eng. Res. Develop.*, vol. 11, no. 3, pp. 331–342, Jun. 2017, doi: 10.1007/s11740-017-0735-6.

[2] J. L. Wahlers and J. F. Cox, "Competitive factors and performance measurement: Applying the theory of constraints to meet customer needs," *Int. J. Prod. Econ.*, vol. 37, nos. 2–3, pp. 229–240, 1994, doi: 10.1016/0925-5273(94)90173-2.

[3] D. X. Peng and G. Lu, "Exploring the impact of delivery performance on customer transaction volume and unit price: Evidence from an assembly manufacturing supply chain," *Prod. Oper. Manage.*, vol. 26, no. 5, pp. 880–902, May 2017, doi: 10.1111/poms.12682.

[4] A. E. Coronado, A. C. Lyons, D. F. Kehoe, and J. Coleman, "Enabling mass customization: Extending build-to-order concepts to supply chains," *Prod. Planning Control*, vol. 15, no. 4, pp. 398–411, Jun. 2004, doi: 10.1080/0953728042000238809.

[5] P. Fredriksson and L.-E. Gadde, "Flexibility and rigidity in customization and build-to-order production," *Ind. Marketing Manage.*, vol. 34, no. 7, pp. 695–705, Oct. 2005, doi: 10.1016/j.indmarman.2005.05.010.
[6] S. Y. Nof, W. E. Wilhelm, and H.-J. Warnecke, *Industrial Assembly*. Boston, MA, USA: Springer, 1997.
[7] D. E. Whitney, *Mechanical Assemblies: Their Design, Manufacture, and Role in Product Development*. New York, NY, USA: Oxford Univ. Press, 2004.
[8] H. Lödding, *Handbook of Manufacturing Control: Fundamentals, Description, Configuration*. Heidelberg, Germany: Springer, 2013.
[9] W. J. Hopp and M. L. Spearman, *Factory Physics*, 3rd ed. Long Grove, IL, USA: Waveland Press, 2011.
[10] S. Beck, *Modellgestütztes Logistikcontrolling Konvergierender Materialflüsse*. Hanover, Germany: Hannover, Univ., 2013.
[11] M. Schmidt, *Modellierung Logistischer Prozesse der Montage*. Hanover, Germany: Hannover, Univ., 2010.
[12] H.-P. Wiendahl, *Load-Oriented Manufacturing Control*. Berlin, Germany: Springer, 1995.
[13] P. Nyhuis and H.-P. Wiendahl, *Logistische Kennlinien: Grundlagen, Werkzeuge und Anwendungen*, 3rd ed. Berlin, Germany: Springer, 2012.
[14] G. Schuh, J.-P. Prote, F. Sauermann, and B. Franzkoch, "Databased prediction of order-specific transition times," *CIRP Ann.*, vol. 68, no. 1, pp. 467–470, 2019, doi: 10.1016/j.cirp.2019.03.008.
[15] K. R. Baker and D. Trietsch, *Principles of Sequencing and Scheduling*. Hoboken, NJ, USA: Wiley, 2009.
[16] G. E. Vieira, J. W. Herrmann, and E. Lin, "Rescheduling manufacturing systems: A framework of strategies, policies, and methods," *J. Scheduling*, vol. 6, no. 1, pp. 39–62, 2003, doi: 10.1023/A:1022235519958.
[17] O. Kocatepe, "An application framework for scheduling optimization problems," in *Proc. IEEE 8th Int. Conf. Appl. Inf. Commun. Technol. (AICT)*, Astana, Kazakhstan, Oct. 2014, pp. 1–4.
[18] P. Burggraf, J. Wagner, and B. Koke, "Artificial intelligence in production management: A review of the current state of affairs and research trends in academia," in *Proc. Int. Conf. Inf. Manage. Process. (ICIMP)*, London, U.K., Jan. 2018, pp. 82–88.
[19] W. H. M. Raaymakers and A. J. M. M. Weijters, "Makespan estimation in batch process industries: A comparison between regression analysis and neural networks," *Eur. J. Oper. Res.*, vol. 145, no. 1, pp. 14–30, 2003, doi: 10.1016/S0377-2217(02)00173-X.
[20] T. C. E. Cheng and M. C. Gupta, "Survey of scheduling research involving due date determination decisions," *Eur. J. Oper. Res.*, vol. 38, no. 2, pp. 156–166, 1989, doi: 10.1016/0377-2217(89)90100-8.
[21] A. Öztürk, S. Kayalıgil, and N. E. Özdemirel, "Manufacturing lead time estimation using data mining," *Eur. J. Oper. Res.*, vol. 173, no. 2, pp. 683–700, Sep. 2006, doi: 10.1016/j.ejor.2005.03.015.
[22] A. Alenezi, S. A. Moses, and T. B. Trafalis, "Real-time prediction of order flowtimes using support vector regression," *Comput. Oper. Res.*, vol. 35, no. 11, pp. 3489–3503, Nov. 2008, doi: 10.1016/j.cor.2007.01.026.
[23] C. Wang and P. Jiang, "Deep neural networks based order completion time prediction by using real-time job shop RFID data," *J. Intell. Manuf.*, vol. 30, no. 3, pp. 1303–1318, Mar. 2019, doi: 10.1007/s10845-017-1325-3.
[24] P. Burggräf, J. Wagner, B. Koke, and F. Steinberg, "Approaches for the prediction of lead times in an engineer to order environment—A systematic review," *IEEE Access*, vol. 8, pp. 142434–142445, 2020, doi: 10.1109/ACCESS.2020.3010050.
[25] T. Berlec, E. Govekar, J. Grum, P. Potocnik, and M. Starbek, "Predicting order lead times," *J. Mech. Eng.*, vol. 54, no. 5, pp. 308–321, 2008.
[26] J. Gramdi, "Elaborating actual lead time with both management and execution data," in *Proc. Int. Conf. Comput. Ind. Eng.*, Troyes, France, Jul. 2009, pp. 674–677.
[27] D. Gyulai, A. Pfeiffer, G. Nick, V. Gallina, W. Sihn, and L. Monostori, "Lead time prediction in a flow-shop environment with analytical and machine learning approaches," *IFAC-PapersOnLine*, vol. 51, no. 11, pp. 1029–1034, 2018, doi: 10.1016/j.ifacol.2018.08.472.
[28] A. D. Karaoglan and O. Karademir, "Flow time and product cost estimation by using an artificial neural network (ANN): A case study for transformer orders," *Eng. Economist*, vol. 62, no. 3, pp. 272–292, Jul. 2017, doi: 10.1080/0013791X.2016.1185808.
[29] A. M. Law, "Simulation-models level of detail determines effectiveness," *Ind. Eng.*, vol. 23, no. 10, pp. 16–18, 1991.
[30] S. Robinson, "Conceptual modelling for simulation Part I: Definition and requirements," *J. Oper. Res. Soc.*, vol. 59, no. 3, pp. 278–290, Mar. 2008, doi: 10.1057/palgrave.jors.2602368.
[31] S. Robinson, "Sumulation projects: Building the right conceptual model project's objectives: The key to simulation modeling is to use the minimum amount of detail required to achieve the," *Ind. Eng.-Norcross*, vol. 26, no. 9, pp. 34–36, 1994.
[32] R. K. Yin, *Case Study Research and Applications: Design and Methods*. Newbury Park, CA, USA: SAGE, 2017.
[33] D. M. McCutcheon and J. R. Meredith, "Conducting case study research in operations management," *J. Oper. Manage.*, vol. 11, no. 3, pp. 239–256, 1993, doi: 10.1016/0272-6963(93)90002-7.
[34] I. Tirkel, "Forecasting flow time in semiconductor manufacturing using knowledge discovery in databases," *Int. J. Prod. Res.*, vol. 51, no. 18, pp. 5536–5548, Sep. 2013, doi: 10.1080/00207543.2013.787168.
[35] L. Lingitz, V. Gallina, F. Ansari, D. Gyulai, A. Pfeiffer, W. Sihn, and L. Monostori, "Lead time prediction using machine learning algorithms: A case study by a semiconductor manufacturer," *Procedia CIRP*, vol. 72, pp. 1051–1056, Jan. 2018, doi: 10.1016/j.procir.2018.03.148.
[36] S. Singh and U. Soni, "Predicting order lead time for just in time production system using various machine learning algorithms: A case study," in *Proc. 9th Int. Conf. Cloud Comput., Data Sci. Eng. (Confluence)*, Noida, India, Jan. 2019, pp. 422–425.
[37] P. Jeatrakul and K. W. Wong, "Comparing the performance of different neural networks for binary classification problems," in *Proc. 8th Int. Symp. Natural Lang. Process.*, Bangkok, Thailand, Oct. 2009, pp. 111–115.
[38] S. Dreiseitl and L. Ohno-Machado, "Logistic regression and artificial neural network classification models: A methodology review," *J. Biomed. Inform.*, vol. 35, nos. 5–6, pp. 352–359, 2002, doi: 10.1016/s1532-0464(03)00034-0.
[39] R. Kumari and S. Kr., "Machine learning: A review on binary classification," *Int. J. Comput. Appl.*, vol. 160, no. 7, pp. 11–15, Feb. 2017.
[40] A. Ghatak, *Deep Learning With R*. Singapore: Springer, 2019.
[41] A. F. Agarap, "Deep learning using rectified linear units (ReLU)," Mar. 2018, *arXiv:1803.08375*. [Online]. Available: https://arxiv.org/abs/1803.08375
[42] U. Fayyad, G. Piatetsky-Shapiro, and P. Smyth, "From data mining to knowledge discovery in databases," *AI Mag.*, vol. 17, no. 3, p. 37, 1999, doi: 10.1609/aimag.v17i3.1230.
[43] J. Han, M. Kamber, and J. Pei, *Data Mining: Concepts and Techniques*, 3rd ed. Amsterdam, The Netherlands: Elsevier, 2012.
[44] R. Roscher, B. Bohn, M. F. Duarte, and J. Garcke, "Explainable machine learning for scientific insights and discoveries," *IEEE Access*, vol. 8, pp. 42200–42216, 2020, doi: 10.1109/ACCESS.2020.2976199.
[45] M. Kuhn and K. Johnson, *Applied Predictive Modeling*, 5th ed. New York, NY, USA: Springer, 2016.
[46] G. Dong and H. Liu, *Feature Engineering for Machine Learning and Data Analytics*. London, U.K.: Chapman & Hall, 2018.
[47] D. E. Hinkle, S. G. Jurs, and W. Wiersma, *Applied Statistics for Behavioural Sciences*, 5th ed. Boston, MA, USA: Houghton Mifflin Company, 2003.
[48] M. M. Mukaka, "A guide to appropriate use of correlation coefficient in medical research," *Malawi Med. J., J. Med. Assoc. Malawi*, vol. 24, no. 3, pp. 69–71, 2012.
[49] J. Mohamad-Saleh and B. S. Hoyle, "Improved neural network performance using principal component analysis on MATLAB," *Int. J. Comput., Internet Manage.*, vol. 16, no. 2, pp. 1–8, 2008.
[50] I. A. Basheer and M. Hajmeer, "Artificial neural networks: Fundamentals, computing, design, and application," *J. Microbiol. Methods*, vol. 43, no. 1, pp. 3–31, 2000, doi: 10.1016/S0167-7012(00)00201-3.
[51] J. Davis and M. Goadrich, "The relationship between precision-recall and ROC curves," in *Proc. 23rd Int. Conf. Mach. Learn. (ICML)*, Pittsburgh, PA, USA, 2006, pp. 233–240.
[52] D. M. W. Powers, "Evaluation: From precision, recall and F-measure to ROC, informedness, markedness and correlation," *Int. J. Mach. Learn. Technol.*, vol. 2, no. 1, pp. 37–63, 2020. [Online]. Available: https://arxiv.org/abs/2010.16061
[53] D. Chicco and G. Jurman, "The advantages of the matthews correlation coefficient (MCC) over F1 score and accuracy in binary classification evaluation," *BMC Genomics*, vol. 21, no. 1, p. 6, Dec. 2020, doi: 10.1186/s12864-019-6413-7.
[54] B. W. Matthews, "Comparison of the predicted and observed secondary structure of T4 phage lysozyme," *Biochimica et Biophysica Acta (BBA)-Protein Struct.*, vol. 405, no. 2, pp. 442–451, 1975, doi: 10.1016/0005-2795(75)90109-9.
[55] K. C. Green and J. S. Armstrong, "Simple versus complex forecasting: The evidence," *J. Bus. Res.*, vol. 68, no. 8, pp. 1678–1685, Aug. 2015, doi: 10.1016/j.jbusres.2015.03.026.

[56] V. Sessions and M. Valtorta, "The effects of data quality on machine learning algorithms," in *Proc. ICIQ*, vol. 6, 2006, pp. 485–498.
[57] M. I. Jordan and T. M. Mitchell, "Machine learning: Trends, perspectives, and prospects," *Science*, vol. 349, no. 6245, pp. 255–260, Jul. 2015, doi: 10.1126/science.aaa8415.
[58] A. Zheng and A. Casari, *Feature Engineering for Machine Learning: Principles and Techniques for Data Scientists*. Boston, MA, USA: O'Reilly, 2018.
[59] M. F. Kilkenny and K. M. Robinson, "Data quality: 'Garbage in—Garbage out,'" *Health Inf. Manage. J.*, vol. 47, no. 3, pp. 103–105, Sep. 2018, doi: 10.1177/1833358318774357.
[60] J. Wagner, P. Burggräf, M. Dannapfel, and C. Fölling, "Assembly disruptions. Empirical evidence in the manufacturing industry of Germany, Austria and Switzerland," *Int. Refereed J. Eng. Sci.*, vol. 6, no. 3, pp. 15–25, 2017.
[61] S. Chopra and M. S. Sodhi, "Supply-chain breakdown," *MIT Sloan Manage. Rev.*, vol. 46, no. 1, pp. 53–61, 2004.

PETER BURGGRÄF was born in 1980. He received the M.B.A., Dr.-Ing., and professor degrees. He studied mechanical engineering in Aachen and London. He wrote his Ph.D. Thesis at the Laboratory for Machine Tools and Production Engineering (WZL), RWTH Aachen University, in the field of factory planning. From 2011 to 2017, he was the Chief Engineer with the Chair of Production Engineering, WZL. Since 2013, he has been the Managing Director of the StreetScooter Research GmbH. Since 2017, he has been with the Chair of International Production Engineering and Management (IPEM), University of Siegen, Germany.

JOHANNES WAGNER was born in 1986. He received the M.B.A. and Dr.-Ing. degrees. He studied industrial engineering at RWTH Aachen University and Tsinghua University, Beijing. He wrote his Ph.D. Thesis at the Chair of Production Engineering, Laboratory for Machine Tools and Production Engineering (WZL), RWTH Aachen University, in the field of disruption management in low-volume assembly. Since 2017, he has been the Chief Engineer with the Chair of International Production Engineering and Management (IPEM), University of Siegen. He is also the Founder and the CEO of the SDFS Smart Demonstration Factory Siegen.

BENJAMIN HEINBACH was born in 1988. He received the dual M.Sc. degree. He is currently pursuing the Ph.D. degree with the Chair of International Production Engineering and Management (IPEM), University of Siegen. He studied international project management at the Chalmers University of Technology and Northumbria University. He is also writing his Ph.D. thesis in the field of automated factory planning based on artificial intelligence. Since 2017, he has been working with the Chair of International Production Engineering and Management (IPEM), University of Siegen, where he has been managing the group factory planning for medium-sized businesses, since 2019.

FABIAN STEINBERG was born in 1990. He received the M.Sc. degree. He is currently pursuing the Ph.D. degree with the Chair of International Production Engineering and Management (IPEM), University of Siegen. He studied production engineering at the RWTH Aachen University. He is also writing his Ph.D. thesis in the field of predicting missing parts for the assembly start in engineer-to-order environment based on artificial intelligence. Since 2018, he has been working with the Chair of International Production Engineering and Management (IPEM), University of Siegen.