



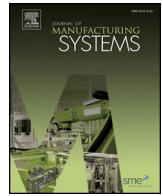
## **A prognostic algorithm to prescribe improvement measures on throughput bottlenecks**

Downloaded from: <https://research.chalmers.se>, 2026-04-04 12:03 UTC

Citation for the original published paper (version of record):

Subramaniyan, M., Skoogh, A., Sheikh, M. et al (2019). A prognostic algorithm to prescribe improvement measures on throughput bottlenecks. *Journal of Manufacturing Systems*, 53: 271-281. <http://dx.doi.org/10.1016/j.jmsy.2019.07.004>

N.B. When citing this work, cite the original published paper.



## A prognostic algorithm to prescribe improvement measures on throughput bottlenecks

Mukund Subramaniyan<sup>a,\*</sup>, Anders Skoogh<sup>a</sup>, Azam Sheikh Muhammad<sup>b</sup>, Jon Bokrantz<sup>a</sup>,  
Ebru Turanoğlu Bekar<sup>a</sup>

<sup>a</sup> Department of Industrial and Materials Science, Chalmers University of Technology, Gothenburg, 41296, Sweden

<sup>b</sup> Department of Computer Science and Engineering, Chalmers University of Technology, Gothenburg, 41296, Sweden

### ARTICLE INFO

#### Keywords:

Throughput bottlenecks  
Production system  
Manufacturing system  
Predictive analytics  
Maintenance  
Prescriptive approach

### ABSTRACT

Throughput bottleneck analysis is important in prioritising production and maintenance measures in a production system. Due to system dynamics, bottlenecks shift between different production resources and across production runs. Therefore, it is important to predict where the bottlenecks will shift to and understand the root causes of predicted bottlenecks. Previous research efforts on bottlenecks are limited to only predicting the shifting location of throughput bottlenecks; they do not give any insights into root causes. Therefore, the aim of this paper is to propose a data-driven prognostic algorithm (using the active-period bottleneck analysis theory) to forecast the durations of individual active states of bottleneck machines from machine event-log data from the manufacturing execution system (MES). Forecasting the duration of active states helps explain the root causes of bottlenecks and enables the prescription of specific measures for them. It thus forms a machine-states-based prescriptive approach to bottleneck management. Data from real-world production systems is used to demonstrate the effectiveness of the proposed algorithm. The practical implications of these results are that shop-floor production and maintenance teams can be forewarned, before a production run, about bottleneck locations, root causes (in terms of machine states) and any prescribed measures, thus forming a prescriptive approach. This approach will enhance the understanding of bottleneck behaviour in production systems and allow data-driven decision making to manage bottlenecks proactively.

### 1. Introduction

Manufacturing companies are constantly looking for ways to improve the productivity of production systems. Doing this requires an accurate performance estimate of the production system. Throughput is one of the main indicators for evaluating performance in a production system but is often constrained by one or more machines, referred to as “throughput bottlenecks” [1]. Previous research has shown that improvement measures on throughput bottlenecks, such as cycle time reduction [2,3] and prioritisation of maintenance measures [4,5] increase overall throughput in production systems.

With the development of digital technologies, many manufacturing companies have started collecting machine data in digital format [6]. This data enables the use of data-driven algorithms to detect throughput bottlenecks [7]. In general, such data-driven algorithms can be classified into two main categories of algorithm: 1) descriptive and 2) predictive. In descriptive algorithms, historical machine data is analysed to detect bottlenecks in a production system [8–11]. Due to

the dynamics of the production system, historical bottlenecks may not behave like bottlenecks in the future [12]. Therefore, it is important to use predictive algorithms to predict future bottlenecks. In the predictive algorithms category (also called “prognostic methods”, as prognostics is the discipline of forecasting future performance [13]), historical machine data is used to predict the location of bottlenecks in the production system. Such prediction can be made using statistically-based [12,14] or machine-learning-based methods [15]. It can also be made on a real-time basis, using buffer levels [16,17].

Prescriptive bottleneck management extends beyond predicting the location of throughput bottlenecks in a production system. It prescribes the necessary improvement measures in the predicted throughput bottlenecks. In the literature, considerable results have been achieved in developing data-driven predictive algorithms to predict future bottlenecks [12,15,17,10]. These are important contributions to advancing prescriptive bottleneck management. However, these research efforts focus only on predicting the location of future bottlenecks in a production system. Although predicting the bottlenecks leads to a generic

\* Corresponding author.

E-mail address: [mukunds@chalmers.se](mailto:mukunds@chalmers.se) (M. Subramaniyan).

<https://doi.org/10.1016/j.jmsy.2019.07.004>

Received 15 January 2019; Received in revised form 26 April 2019; Accepted 25 July 2019

Available online 18 November 2019

0278-6125/ © 2019 The Authors. Published by Elsevier Ltd on behalf of The Society of Manufacturing Engineers. This is an open access article under the CC BY license (<http://creativecommons.org/licenses/by/4.0/>).

prescription of measures (such as buffering the main bottlenecks, or scheduling around them [2,1]), implementation of these measures is more effective when there is knowledge of the bottlenecks' root causes [18]. Thus, further development of data-driven bottleneck prediction algorithms is required, to provide insights into those root causes. Moreover, understanding the root causes leads to the prescription of specific throughput improvement measures.

One way to understand the possible root causes of bottlenecks at production-system level is by analysing the different states of the machine during a production run. The state represents the activity carried out by, or on, the machines such as “producing”, “breakdown” and “changeover” [19]. A machine can be a bottleneck because it is “producing” for most the scheduled production time [3], or due to random breakdowns and stoppages which contribute to greater down-time [20], or a combination of these states. Forecasting the average duration of a machine state for future production run helps in prescribing specific improvement measures based on the states.

Therefore, the purpose of this study is to advance the field of data-driven throughput bottleneck analysis towards prescriptive bottleneck management. The aim of this paper is to propose a data-driven prognostic algorithm using the active-period bottleneck analysis theory as explained in [19]. This forecasts the individual active state durations of bottleneck machines using their event-log data. Forecasting the active states explains the root causes of bottleneck machines and enables development of a machine-state-based prescriptive approach to throughput bottleneck management. The proposed data-driven prognostic algorithm uses time-series modelling of machine states and supports the usage of any time-series forecasting methodology to be employed. To show an example, in this study, Auto regressive moving average (ARIMA) time-series forecasting methodology [21] is employed. This study advances the field of data-driven throughput bottleneck analysis. In contrast to previously published approaches to bottleneck location prediction, the bottlenecks' root causes in terms of machine active states are predicted and measures prescribed, thus forming a prescriptive approach to bottleneck management.

**2. Literature review**

Firstly, the need for a machine-states-based prescriptive approach is presented from an academic and industrial point of view. Secondly, as the proposed algorithm in this study is based on the active-period theory of bottleneck detection, this theory is briefly discussed. Thirdly, different predictive algorithms that use the active-period theory to predict bottlenecks are briefly discussed. Lastly, time-series modelling, forecasting and evaluation techniques are briefly presented.

*2.1. Need for prescriptive approach for throughput bottleneck management*

From a real-world, industrial practice point of view, when multiple machines are detected as probable bottlenecks, proactive planning of throughput improvement measures becomes complex and challenging. This raises the need to rank those machines for throughput improvement [11]. As a result, the production and maintenance teams often

need more information on the bottlenecks to identify the right improvement measures for them. Not having enough bottleneck information can lead to an ambiguous situation between the production and maintenance teams on the shop floor. This is because it is unclear whether the anticipated bottlenecks require maintenance or production improvement measures, or both, to improve throughput. This could be disadvantageous, because incorrect improvement measures in bottlenecks may reduce production system throughput [22]. The conventional approach to making the most of throughput improvement decisions in a production system is based on expert experience [23]. Instead, better results would be achieved if the subjective judgement were to be reduced and complemented with data-driven judgement from the algorithms. There is, therefore, a need for a data-driven support tool to help shop-floor teams make the right decisions on improving throughput. This was also reported as a result by [24] in the study of future shop-floor operators in Swedish manufacturing industries. One way to support production and maintenance teams in their decision-making processes in bottlenecks is to provide deeper insights into the different states of bottleneck machines and their anticipated behaviour in future production runs.

*2.2. Previous work on prescriptive approaches to throughput bottleneck management*

To address the above need, [25] developed a discrete-event, simulation-based optimisation solution based on the established bottleneck detection technique known as the “active-period theory” of bottleneck analysis (as proposed by [26]). This solution automatically identifies bottlenecks and prescribes improvement measures, based on machine states. The method prescribes improvement measures from historical machine data. This data is then input into a discrete-event simulation model of the production system, with no predictions on future performance. Moreover, as the simulation is a constrained environment, measures triggered by it may not always conform to the real-world constraints under which they will be used. Altogether, these factors limit their practical application in throughput bottleneck analysis in general, and in the context of the purpose of this paper in particular. Data-driven algorithms can further improve the practical applications, as it can overcome some of the limitations of discrete-event simulations. Thus, to make industrial practice more effective, a practical data-driven prognostic algorithm is needed to predict the machine states' behaviour and prescribe any improvement measures. The need for such data-driven prescriptive solutions, aligned to real-world requirements, was also emphasised by [27,28].

*2.3. Active-period theory of bottleneck detection*

The active-period theory of bottleneck detection was proposed by [19]. The machine states shown in Fig. 1 are divided into “active” and “inactive”. The active state is that state of the machine when something happens on the machine – when it is “producing”, or when it is in “breakdown” or “changeover”. Once the active times for a production run are computed, the active-period percentage of the machines in the

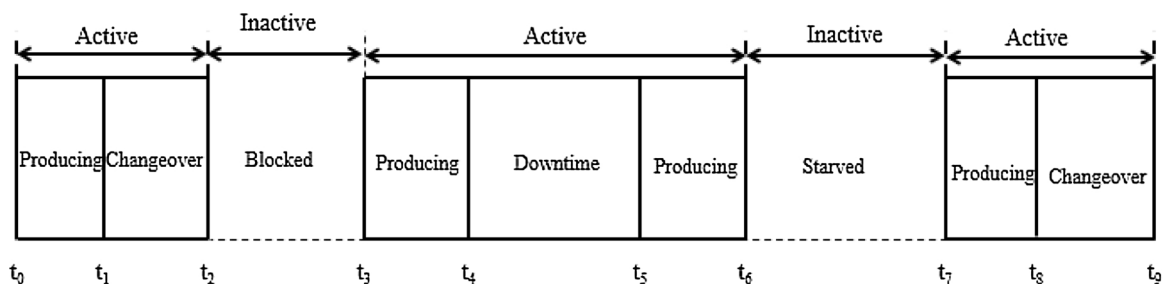


Fig. 1. Machine states during a production run (adapted from [19]).

production system may be determined. Once these percentages for machines are compared, the average bottleneck machines for the production run may be determined. And once the bottlenecks have been detected using active-period percentages, their root causes (in terms of duration of individual active states) may be determined. This is advantageous, as it provides insight into the type of improvement measures needed in the bottlenecks [10].

#### 2.4. Throughput bottleneck prediction using active-period theory

[17] proposes a bottleneck prediction algorithm based on the active-period theory of bottleneck detection [26]. This uses the active states of the machine and buffer information as inputs and predicts bottlenecks in real-time. Even though the approach detailed in [17] predicts bottlenecks in real-time, the reason that a bottleneck must be determined manually, is that an internal active state can be caused by a “producing” state, “breakdown” state, or “setup” state of a machine. When predicting bottlenecks in real-time, it can be overwhelming for production and maintenance teams to plan and carry out improvement measures in bottlenecks. Particularly so when there is a high degree of bottleneck shift between machines in a production run. It is therefore of more practical use to find the throughput bottlenecks over a set period such as a shift, or a day.

Recently, in [14], an event-log-based data-driven algorithm was developed to predict the average bottlenecks for the next production run by combining active-period bottleneck analysis theory (as proposed by [19]) with ARIMA methodology. Also, the advantages of the active-period bottleneck analysis theory compared to other bottleneck detection theories have been shown exclusively in [10]. Those are: (1) detecting potential throughput bottlenecks in a system and (2) indicating probable root causes of bottlenecks in terms of different machine states, which can be used to prescribe measures. However, the prediction algorithm developed in [14] only addressed the prediction of bottleneck locations in the production system.

Therefore, in this paper, we extend the work reported in [10] by proposing a prognostic algorithm to indicate the probable root causes of bottlenecks and develop a prescriptive approach to bottleneck management. The novelty of this study is its integration of the proposed prognostic algorithm with measures which form a prescriptive approach to throughput bottleneck management, using the active-period bottleneck analysis theory.

#### 2.5. Time-series-based ARIMA forecasting models

A time series is a set of observations on a variable collected at regular time intervals [29]. ARIMA models are the most commonly used statistical-based forecasting techniques for time series as they explicitly account for autocorrelation [29]. Moreover, ARIMA models are easy to interpret and can produce unbiased forecasts. ARIMA models can be expressed as ARIMA (p, d, q) where p, d and q are non-negative integers representing the order of autoregressive (AR), integrated (I), and moving average (MA) parts of the model. The AR part indicates that the variable is regressed on its own historical values while MA refers to the regression error terms that occurred in the past. In the real world, most of the data is non-stationary; not following AR or MA models, but a mixture of them [29]. Therefore, AR, MA or ARMA models cannot be used directly. The data can be made stationary by differencing d times before fitting the ARMA model. (I) denotes the number of non-seasonal differencing operations. Given a finite time-series sequence,  $X_1, X_2, X_3, \dots, X_t$ , one can find the continuation of  $X_{t+1}, X_{t+2}, \dots$  using an ARIMA model where  $\{X_t\}$  is the stochastic variable. An ARIMA model (p,d,q) can be expressed as:

$$X_t = \phi_1 X_{t-1} + \phi_2 X_{t-2} + \dots + \phi_p X_{t-p} + e_t - \theta_1 e_{t-1} - \theta_2 e_{t-2} - \dots - \theta_q e_{t-q} \quad (1)$$

Where  $\{\phi_t\}$  and  $\{\theta_t\}$  are autoregressive and moving average

parameters respectively.  $X_t$  is the modelled variable and represents proportions of active states such as “producing”, “breakdown” and so on and  $e_t$  is a disturbance random variable following normal independent distribution,  $e_t \in (0, \sigma^2)$ . The input to the ARIMA models is the time series observation, whilst the ARIMA output is the point forecasts for the future and various other parameters associated with point forecasts (including prediction interval, error associated with prediction and so on).

### 3. Methodology

In this section, the overall research approach is first presented. This is followed by the presentation of the proposed data-driven prognostic algorithm.

#### 3.1. Research approach

The research approach is divided into two broad stages. In the first stage, the active-period-theory-based prognostic algorithm is developed, to forecast the durations of individual active states of the predicted bottleneck machines. This step also includes a detailed analysis of different ways in which the algorithmic results can be used to identify the root causes, which then helps prescribe machine-state-based measures. The algorithm development is based on the detailed literature study of the data-driven throughput bottleneck prediction field and a study of the real-world MES-type data, taken from an automotive component machining line. The detailed procedure followed during this stage is explained in Section 3.2 of the paper.

In the second stage, the developed algorithm is tested using real-world industrial data to check its performance. The entire test was accomplished by uploading the real-world dataset into R software and using libraries (such as forecast and tseries) and ARIMA functions such as auto.arima() [30]. The detailed procedure followed during this stage is explained in Section 5 of the paper.

#### 3.2. Proposed data-driven prognostic algorithm to understand the root causes of bottlenecks

The proposed algorithm requires the event-log data containing machine states and time stamps from all machines in the production system, collected over a given period. The algorithm starts by predicting bottlenecks in the production system for a future production run using the algorithm presented in [14]. A “production run” is defined as one production cycle; a shift or day, for example. Thereafter, the algorithm identifies the active states of predicted bottleneck machines from their event-log data, computes the duration of active states and converts this into the proportion of active time across different production runs to form a time series. Any time series forecasting methodology can be applied but, in this study, we apply ARIMA forecasting methodology to forecast the future values of individual active states. The results obtained from the algorithm are analysed in different ways to understand the root causes. This, in turn, is used to prescribe state-based throughput improvement measures. The different steps of the proposed algorithm are shown in Fig. 2. The proposed algorithm is designed to study the bottlenecks and root causes on a production-run level, as this study is designed to recommend measures for the next production run.

The notation used to construct the algorithm is explained in Table 1.

The detailed computations under each step shown in Fig. 2 are explained below:

##### Step 1: Application of throughput bottleneck algorithm

The throughput bottleneck prediction algorithm as developed by [14] gives a set of predicted throughput bottlenecks, their corresponding mean active-period percentages and a window size. Window size defines the number of historical production runs needed to make a prediction for the next run.

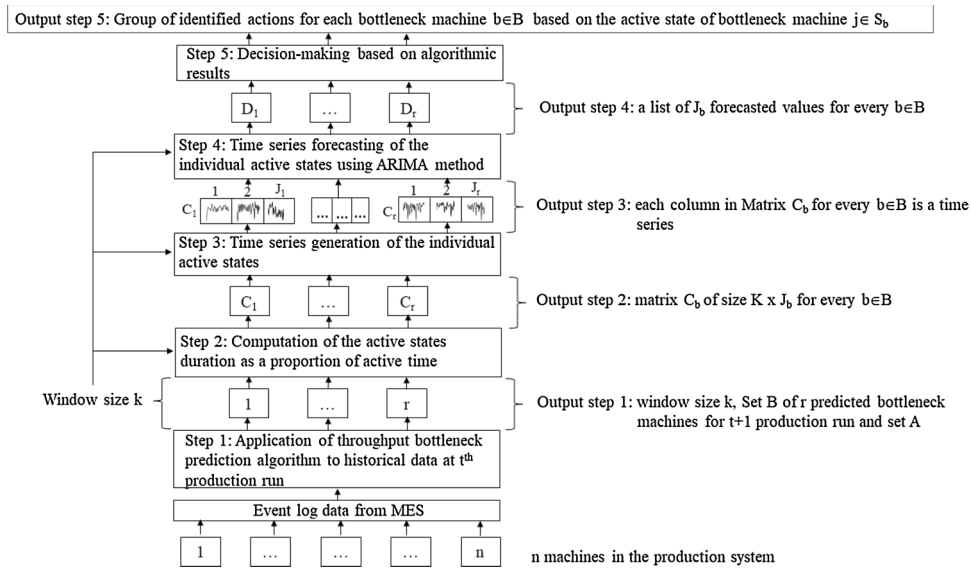


Fig. 2. Step-by-step procedure for the proposed prognostic algorithm.

Let us apply this algorithm at the  $t^{\text{th}}$  production run to get the predictions for  $(t + 1)^{\text{th}}$  production run. The three outputs from the algorithm are sets B and A and window size k. These three outputs form the input to our proposed algorithm in this study. We then move to Step 2.

**Step 2: Computation of each of the active states’ duration as a proportion of active time**

This step formally marks the beginning of the proposed algorithm in this study. The first thing we need to calculate is the total active duration for every production run of a bottleneck machine. In other

words, we need to calculate  $E_{bi}$  for all  $b \in B$  and  $i \in I$ . To do that, the cartesian product of the bottleneck machines with the set of production runs is calculated:  $\{1, 2, \dots, r\} \times \{(t-k+1), (t-k+2), \dots, t-1, t\}$ . The total active duration for a machine  $b \in B$  on a production run  $i \in I$  is calculated using the following formula:

$$E_{bi} = \sum_{j \in S_b} E_{jbi}; \quad b \in B, \quad i \in I \tag{2}$$

We then need to calculate the elapsed time of each active state of a bottleneck machine, as a proportion of the total active time for a

**Table 1**  
Notation used to construct the algorithm.

Notations	Explanations
<i>The following notation is used to develop Step 1 of the prognostic algorithm.</i>	
n	Number of machines in the production system.
M	Set of machines, i.e. $M = \{1, 2, 3, \dots, n\}$ in the production system. To consider any machine, we write $m \in M$ .
r	Number of predicted bottleneck machines for a production run.
B	Set of predicted bottleneck machines, i.e. $B = \{1, 2, 3, \dots, r\}$ for a production run in the production system. To consider any bottleneck machine, we write $b \in B$ .
A	Set of predicted mean active-period percentages of the machines in the set B for a production run, i.e. $A = \{a_1, a_2, a_3, \dots, a_r\}$ .
t	Last production run.
k	Window size (i.e. number of historical production runs used for bottleneck prediction in the production system).
I	An ordered set of k production runs, i.e. $I = \{(t-k+1), (t-k+2), \dots, t-1, t\}$ . To consider any production run, we write $i \in I$ .
<i>The following notation is used to develop Step 2 of the prognostic algorithm.</i>	
$S_b$	Set of active states of every $b \in B$ . (This is represented as a set, as every machine may have a distinct number of active states). E.g. $S_1$ will represent the active states of bottleneck machine 1.
j	An index used to iterate the set of active states for any $S_b$ , where $b \in B$ .
$J_b$	Cardinality of the set $S_b$ , i.e. $ S_b  = J_b$ and $1 \leq j \leq J_b$ (every machine in the production system has at least one active state). E.g. for the bottleneck machine 1, $S_1 = \{1, 2, \dots, J_1\}$ , where $J_1$ is the number of distinct active states on bottleneck machine 1.
$E_{jbi}$	Elapsed time of an active state $j \in S_b$ for a bottleneck machine $b \in B$ in a production run $i \in I$ . E.g. $E_{111}$ represents the elapsed time of active state 1 for bottleneck machine 1 in the first production run.
$E_{bi}$	Total active duration for a machine $b \in B$ in a production run $i \in I$ . E.g. $E_{11}$ represents the total active duration of bottleneck machine 1 in the first production run.
$f'_{jbi}$	Values of an active state $j \in S_b$ of a bottleneck machine $b \in B$ for every production run, with $i \in I$ as a proportion of the active time.
$C_b$	Matrix of size $k \times J_b$ that stores every active state's value $j \in S_b$ as a proportion of active time of the bottleneck machine $b \in B$ .
<i>The following notation is used to develop Step 4 of the prognostic algorithm.</i>	
$f'_{jbi}$	Forecast values of an active state $j \in S_b$ of a bottleneck machine $b \in B$ for every production run $i \in I$ as a proportion of the active time.
$p_b^{(j)}$	Number of autoregressive terms of the active state $j \in S_b$ of a bottleneck machine $b \in B$ .
$q_b^{(j)}$	Number of logged forecast error terms of the active state $j \in S_b$ of a bottleneck machine $b \in B$ .
$D_b$	A list of size $J_b$ that stores the forecast values of each active state $j \in S_b$ of a bottleneck machine $b \in B$ for $(t + 1)^{\text{th}}$ production run. E.g., $D_1 = [f'_{11(t+1)}, f'_{21(t+1)}]$
<i>The following notation is used to develop Step 5 of the prognostic algorithm</i>	
$(f_{jb})_c$	Pre-defined cut-off value for the active state $j \in S_b$ of a bottleneck machine $b \in B$ .
O	Represents a set of options to choose from in order to analyse the forecast. $O = \{O_1, O_2, O_3, \dots\}$ . E.g. $O_1$ represents the first possible option.
G	Against every option, there are 3 or more conditions, represented as $G \in \{G_1, G_2, G_3, \dots\}$ .
OG	For every possible option-condition pair, we define a group of pre-defined recommended actions. It is given by the cartesian product of the sets O and G, written as $OG = \{O_1G_1, O_1G_2\}$ .

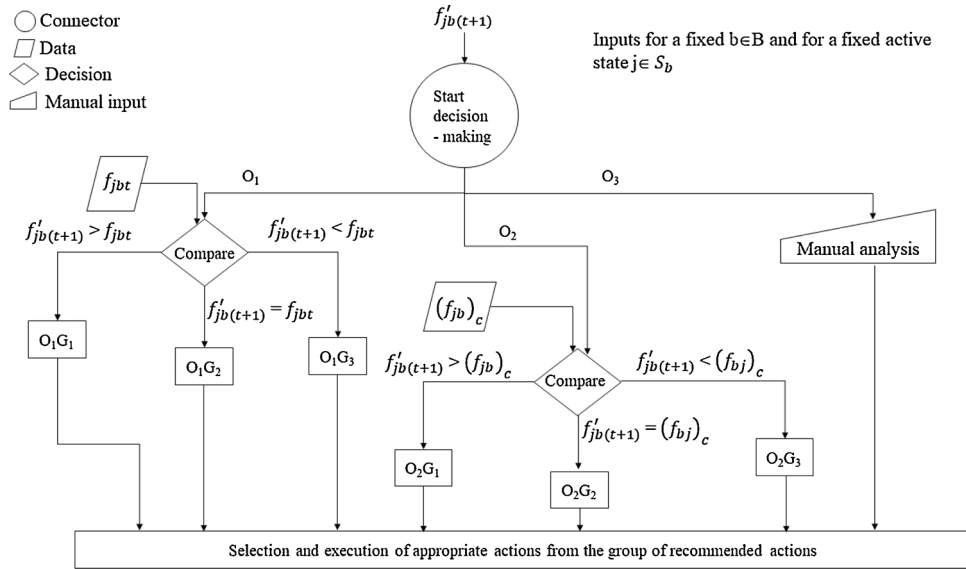


Fig. 3. A use case with three options for using the forecast results for decision-making.

production run and do this for all production runs. Let  $f_{jbi}$  represent the values of an active state  $j \in S_b$  of a bottleneck machine  $b \in B$  for every production run  $i \in I$ , as a proportion of the active time  $E_{bi}$  using the following formula:

$$f_{jbi} = \frac{E_{jbi}}{E_{bi}}; j \in S_b, b \in B, I \in I \quad (3)$$

For a fixed  $b \in B$  and for every active state  $j \in S_b$ , when the above formula is iterated for  $k$  production runs, we get a matrix  $C_b$  of size  $k \times |S_b|$ , where each column corresponds to one particular active state. A sample matrix for a bottleneck machine  $b$  is shown below:

$$C_b = \begin{matrix} & \text{Individual active states} & \longrightarrow \\ \text{Production runs} & \downarrow & \\ \begin{bmatrix} f_{1b(t-k+1)} & \dots & \dots & \dots & f_{j_b b(t-k+1)} \\ f_{1b(t-k+2)} & \dots & \dots & \dots & f_{j_b b(t-k+2)} \\ \dots & \dots & \dots & \dots & \dots \\ f_{1b(t-1)} & \dots & \dots & \dots & f_{j_b b(t-1)} \\ f_{1b(t)} & \dots & \dots & \dots & f_{j_b b(t)} \end{bmatrix} \end{matrix} \quad (4)$$

For every bottleneck machine  $b \in B$ , a matrix  $C_b$  is generated.

**Step 3: Time series generation of individual active states**

For a fixed  $b \in B$ , each column of the matrix  $C_b$  represents a univariate time series for the corresponding active state  $j \in S_b$ . This is under the assumption that each active state  $j \in S_b$  is dependent on its own historical values. Yet another assumption of the every state  $j \in S_b$  is independent of other states was made according to the theory of active-period proposed in [31]. These assumptions are in accordance with the assumptions defined in [14].

**Step 4: Time series forecasting of individual active states using the ARIMA method**

For a fixed  $b \in B$ , the time series forecasting is carried out for the  $(t + 1)^{th}$  production run of each active state proportion  $f_{jbi}$ ,  $j \in S_b$  in matrix  $C_b$  from  $(t-k + 1)^{th}$  to  $t$ , using an ARIMA model. In other words, given a state proportion for every production run up to the time  $t$ , a forecast can be made for  $(t + 1)^{th}$  production run. Separate ARIMA models are deployed for every distinct time series in  $C_b$ . This is shown in Eq. (5). The results of the forecasting for  $(t + 1)^{th}$  production run are stored in a matrix  $D_b$ , which is of size  $1 \times |S_b|$ .

$$D_b = \begin{bmatrix} \phi_{1b(t-1)}f_{11(t-1)} & \dots & \dots & \dots & \phi_{j_b b(t-1)}f_{j_b b(t-1)} \\ + & \dots & \dots & \dots & + \\ \phi_{1b(t-2)}f_{11(t-2)} & \dots & \dots & \dots & \phi_{j_b b(t-2)}f_{j_b b(t-2)} \\ + & \dots & \dots & \dots & + \\ \dots & \dots & \dots & \dots & \dots \\ + & \dots & \dots & \dots & + \\ \phi_{1b(t-p_b^{(1)})}f_{11(t-p_b^{(1)})} & \dots & \dots & \dots & \phi_{j_b b(t-p_b^{(1)})}f_{j_b b(t-p_b^{(1)})} \\ + & \dots & \dots & \dots & + \\ e_{1b(t)} & \dots & \dots & \dots & e_{j_b b(t)} \\ - & \dots & \dots & \dots & - \\ \theta_{1b(t-1)}e_{1b(t-1)} & \dots & \dots & \dots & \dots \\ - & \dots & \dots & \dots & - \\ \dots & \dots & \dots & \dots & \dots \\ \dots & \dots & \dots & \dots & \dots \\ - & \dots & \dots & \dots & - \\ \theta_{1b(t-q_b^{(1)})}e_{1b(t-q_b^{(1)})} & \dots & \dots & \dots & \theta_{j_b b(t-q_b^{(j_b)})}e_{j_b b(t-q_b^{(j_b)})} \end{bmatrix} \quad (5)$$

For a fixed  $b \in B$ , the forecast value of each active state for  $(t + 1)^{th}$  production run may be denoted by  $f'_{j_b b(t+1)}$ , where  $j \in S_b$ . Thus, the final  $D$  matrix containing the forecast values for  $(t + 1)^{th}$  production run for a bottleneck machine  $b$  is shown in Equation 6.

$$D_b = [f'_{1b(t+1)} \dots \dots \dots f'_{j_b b(t+1)}] \quad (6)$$

**Step 5: Decision-making based on algorithmic results**

The next step is to explore the different ways in which forecast values of individual active states of the bottleneck machines can be used to understand the possible root causes and base decisions. The analysis of the forecast values represented in this step is generic. For the demonstration purpose, we have shown a use case consisting of three options and three conditions for each of them. More options and conditions can be added, based on the nature of the production system.

The input for this step is the forecast value of each active state for  $(t + 1)^{th}$  production run, given by  $f'_{j_b b(t+1)}$ , where  $j \in S_b$  for a fixed  $b \in B$ . For a fixed  $b \in B$  and a fixed active state  $j \in S_b$ , there are three different options by which the forecast  $f'_{j_b b(t+1)}$  can be used to understand the root causes of the bottleneck behaviour of  $b$  in terms of active states. This allows prescription of state-based improvement measures. The three different options are represented in Fig. 3 and briefly discussed below. For a fixed  $b \in B$  and a fixed active state  $j \in S_b$ :

- i The forecast for  $(t + 1)^{th}$  production run can be compared with the actuals of  $t^{th}$  production to analyse the increasing or decreasing trends. In other words,  $f'_{j_b b(t+1)}$  can be compared with  $f_{j_b b(t)}$ . Based

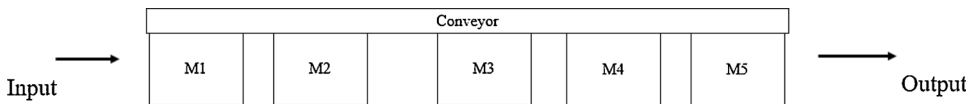


Fig. 4. Production line layout.

on these trends for an active state, the root causes of  $b$  can be understood and a group of measures  $G$  recommended for  $O_1$ . In other words, if  $f'_{jb(t+1)} > f_{jb(t)}$ , then the group of measures  $G$  is recommended.

- ii The forecast for  $(t + 1)^{th}$  production run can be compared with the cut-off value pre-defined for that active state. Let the pre-defined cut-off value defined for that active state be  $(f_{jb})_c$ , where  $j \in S_b$ . If  $f'_{jb(t+1)}$  exceeds the cut-off value  $(f_{jb})_c$ , then the corresponding active state is dominant in contributing to the root causes of  $b$  and a group of measures  $G$  can be recommended for  $O_2$ . In other words, if  $f'_{jb(t+1)} > (f_{jb})_c$ , then group of measures  $G$  is recommended.
- iii The forecast for  $(t + 1)^{th}$  production run can be manually assessed using the engineer’s tacit knowledge and experience to understand the root causes of  $b$ . They may then plan the required measures  $G$  under the option  $O_3$  for different machine states, based on their experience.

From the group of recommended measures, the production and maintenance engineers can choose the most appropriate measures and proactively plan resources for them. They may then execute them on the bottleneck machines during the production run, to reduce the bottleneck effects on system throughput.

**4. Evaluation and benchmarking metrics to assess algorithm performance**

The forecasting performance of the proposed algorithm is evaluated using mean squared error (MSE) and mean absolute error (MAE) metrics. These are the standard metrics for evaluating time-series-forecasting algorithms [32]. The purpose of MAE is to capture the accuracy, whereas MSE captures bias and variance. The mathematical equations for MSE and MAE are shown in (2) and (3), where  $F_t$  denotes the forecast based on the previous values at time  $t$  and  $Y_t$  denotes the true value at  $t$  and  $t = 1, \dots, T$ :

$$MAE = T^{-1} \sum_{t=1}^T |F_t - y_t| \tag{7}$$

$$MSE = T^{-1} \sum_{t=1}^T (F_t - y_t)^2 \tag{8}$$

We benchmark the proposed algorithm using the naïve method, as these are the most commonly used benchmarking models in univariate time series forecasting [33,34]. This method uses the most recent observation to forecast the future, without adjusting for causal factors. Though the naïve method is simple, it is still very helpful in assessing whether the different sophisticated forecasting models add any value for the decision-maker. In real-world production systems, it is common for production and maintenance teams to examine the most recent historical data and base their decisions and planned improvement measures on that. Therefore, it is logical that any forecasting model exceeding the naïve performance should be deemed to add value to the forecasts. The naïve method assumes that all the effects of historical data values on the future are contained in the present value. For example, in a discrete time stochastic process,  $X_0, X_1, X_2, \dots, X_t$ , the forecast at  $t + 1$  is given by:

$$F_{t+1} = X_t \tag{9}$$

Any forecasting model should be benchmarked with that of the naïve (in terms of performance evaluation metrics such as MAE and

MSE), to assess the value of the forecasting models. The performance stability of absolute error and squared error of the proposed algorithm and naïve should be checked and the statistical significance of the mean difference assessed. A common test of statistical significance of the difference in mean is  $t$ -tests [35]. It is reasonable to assume that the absolute error values are independent of each other and follow normal distribution. Likewise, the squared error values are independent of each other and follow normal distribution. The mathematical expression of the  $t$ -test is shown in Eq. 6, where  $X$  is the mean of the proposed algorithm’s performance metric and  $Y$  is the mean of the naïve algorithm.  $SE_x$  is the standard error of the proposed algorithm and  $SE_y$  the standard error of the naïve algorithm.

$$t_{stat} = \frac{X - Y}{\sqrt{SE_x^2 + SE_y^2}} \tag{10}$$

**5. Real-world industrial test study**

An industrial test study of an automotive component machining line is used to illustrate implementation of the proposed algorithm. Fig. 4 shows an automated machining production system in an automotive manufacturing company in Sweden, with machines M1 to M5. All machines are computer numerical control (CNC) machines. The production and maintenance teams wanted to know the future bottlenecks in the production system, plus the durations of different active states which explain the root causes of bottlenecks. Fig. 5 shows the different steps followed to test the proposed algorithm on the production system.

**5.1. Data collection and data cleaning**

Table 2 shows a sample MES record of a production run. This contains event log information from the machines, in terms of their states, classification of active/inactive state, duration of each machine state and corresponding timestamps of machine M2. The active states include “producing” and “down”, whereas the inactive states constitute “blocked”/”starved”/”idle”. “Producing” refers to the state when the machine is engaged in producing a product, whereas “down” represents the stoppages in the machine (including random small and long stops). The MES stores production run data from the machines; no more than two years at any point in time. One production run constitutes 17 scheduled production hours. The MES data is cleaned by removing the information outside the scheduled production hours, plus all other

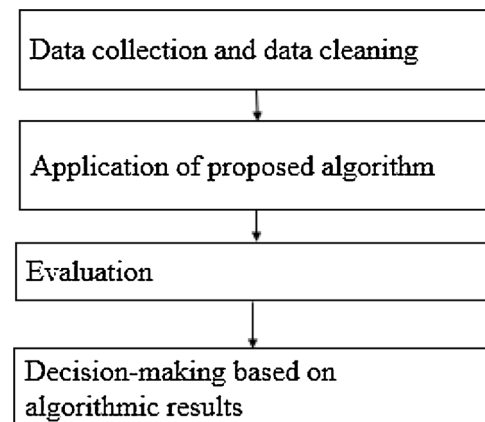


Fig. 5. Steps for testing the proposed algorithm.

**Table 2**  
Sample MES record of machine M2.

Machine	Machine state	Active state	Duration(s)	Date and time
M2	Producing	Active	80	01-07-2016 06:02:18
M2	Down	Active	997	01-07-2016 06:03:38
M2	Blocked/starved/idle	Inactive	39	01-07-2016 06:20:15
M2	Producing	Active	997	01-07-2016 06:20:54
M2	Blocked/starved/idle	Inactive	39	01-07-2016 06:37:31

obvious outliers (such as weekends, or long stops caused by any kind of failure). The number of useful production runs after data cleaning is 315.

5.2. Application of proposed algorithm

The step-by-step procedure of the proposed algorithm as explained in Section 3.2 is applied over the test dataset.

**Step 1: Application of throughput bottleneck prediction algorithm on historical data**

From MES data, the active-period percentages of each machine are calculated for each of the 315 production runs. Thereafter, the bottleneck prediction algorithm (as developed in [14]) is applied to predict the bottlenecks for 171<sup>st</sup> and 172<sup>nd</sup> production runs. The following are the outputs from the algorithm:

- For the production line, the machines’ active-period percentages for the past 50 production runs is a better predictor of the future.

The following are the results of the 171<sup>st</sup> production run:

- Set of predicted bottleneck machines: M5.
- Set of active-period percentages of the predicted bottleneck machines: 85.38%.

The following are the results of the 172<sup>nd</sup> production run:

- Set of predicted bottleneck machines: M4, M5.
- Set of active-period percentages of the predicted bottleneck machines: 87.12%, 87.94%.

Detailed results of the 171<sup>st</sup> and 172<sup>nd</sup> production runs are shown in Table 3.

From Table 3, we observe that the actual bottleneck for a 170<sup>th</sup> production run is M5. Furthermore, it is predicted that for a 171<sup>st</sup> production run, the bottleneck machine will be the same machine, as it has the highest predicted active-period percentage. This indicates that the dynamics of M5’s active period are expected to affect system performance more significantly. Thus, the various individual active states of M5 need to be predicted, so as to better understand the bottleneck’s root causes in terms of the machine active states. Similarly, it can be

**Table 3**  
Actuals for bottleneck prediction results, production runs 171 and 172 (Table format adapted from [14]).

Production run	Machine	M1	M2	M3	M4	M5
170	Actual active period (%)	67.47	65.47	89.37	84.26	94.36
171	ARIMA model order(p,d,q)	1, 0, 1	1, 0, 0	3, 0, 0	1, 1, 1	2, 1, 1
	Forecast active period %	70.41	68.60	78.81	79.59	85.38
	Standard error	1.30	1.29	1.36	1.27	1.45
	t-value	7.69	8.67	3.31	2.99	–
	Actual active period %	68.99	58.45	88.44	90.63	96.09
172	ARIMA model order(p,d,q)	0, 1, 1	1, 0, 0	1, 0, 1	4, 1, 1	3, 1, 1
	Forecast active period %	70.22	68.54	78.94	87.12	87.94
	Standard error	1.29	1.28	1.37	1.30	1.47
	t-value	9.06	9.95	4.49	0.41	–
	Actual active period %	71.01	67.29	72.97	75.03	69.89

noted that for a 172<sup>nd</sup> production run, M4 and M5 are predicted as a group of bottlenecks. So, there is a possibility that M4 or M5’s active-period dynamics will affect system performance more significantly. The dynamics of the individual active states of these machines need predicting, so as to better understand the root causes of bottlenecks.

**Step 2: Computation of active states’ durations as a proportion of active time**

From the bottleneck prediction algorithm, the bottleneck for the next production run is predicted using data from the previous 50 runs. The same number of historical production runs is also used to predict different active states for each of the bottlenecks, so as to explain the nature of the predicted bottleneck. Thus, the proportion of each active state for each bottleneck machine is computed for each of the 50 historical production runs’ data using Eq. (2). To predict the 171<sup>st</sup> production run, examples of historical proportion data from two active states of machine M5 (“producing” and “down”) are shown in Table 4 (Table 4 represents the matrix, as shown in Eq. (4)).

**Step 3: Time series generation of individual active states**

The historical proportion data of each active state for every production run (shown in Table 4) is considered a time series.

**Step 4: Time-series forecasting of individual active states using the ARIMA method**

In this step, forecasting of the “producing” and “down” states of the machine is conducted separately using the ARIMA technique, as explained in Eq. (5). Using one-step-ahead prediction, we forecast the “producing” and “breakdown” components of M5 for 171<sup>st</sup> production run. Because “producing” and “down” are the only two active states contributing to the machine’s active period, forecasting the “producing” state also leads to “down” state forecasts, as they are mirrored values. These forecasting results are recorded in Table 5. This table shows that the machine is “producing” for 86.73% of its active time and “down” for 13.27%. For 170<sup>th</sup> production run, the actuals of the “down” state (as a proportion of active time) are 0.07%. Therefore, it can be inferred that the “down” state of the machine is expected to increase; this requires attention.

Similarly, forecasting of the “producing” and “down” states is carried out on a 172<sup>nd</sup> production run of M4 and M5 to predict their “producing” and “down” states. The results are recorded in Table 5. It can be seen that machine M5 is predicted to be in a “producing” state for 76.56% of its active time and “down” for 23.44%. Likewise, machine M4 is predicted to be in “producing” state for 78.45% of its active

**Table 4**  
Proportion of individual active states of machine M5.

Production run	“Producing” as % of total active period	“Down” as % of total active period
121	92.57	7.43
122	75.15	24.85
...	...	...
...	...	...
...	...	...
170	98.22	1.78

time and “down” for 21.55%. Moreover, machine M5’s “down” state proportion is predicted to increase from production runs 171 to 172, which could entail an alert to the maintenance teams. Similarly, the actual value for M4’s “down” proportion for a 171<sup>st</sup> production run is 14.14%, while the forecast value for a 172<sup>nd</sup> production run is predicted as 21.55%; this also shows an increasing trend.

**Step 5: Decision-making based on algorithmic results**

Using the proposed prognostic algorithm, the locations of future bottlenecks are predicted and the active state durations forecast for the next production run (as a proportion of active time) are made. A diagram of the bottleneck prediction results and forecast active states’ time proportion results appears in Fig. 6.

From the test study, it can be noted that there is no one root cause of bottlenecks, in terms of machine active states. Rather, there are several states. Therefore, understanding the behaviour of each active state is critical when it comes to planning specific improvement measures. The forecast individual active states’ durations can be interpreted in three different options, as a basis for improvement decisions. These are presented in Fig. 3.

Option 1 in Fig. 3 is to compare the duration of active states in the previous production run with the forecasts for future on bottleneck machines. This helps in identify increasing or decreasing trends in the individual active states of a bottleneck machine. The trends are more important than the absolute numbers. The increasing trends of different active states are used to understand why machines are likely to behave like bottlenecks. If a bottleneck machine is predicted to have greater downtime compared to the previous production run, maintenance teams can be alerted and any maintenance measures can be planned. An example from Table 3: in the 172<sup>nd</sup> production run, the forecast downtime proportion of machine M5 is expected to increase compared to the 171<sup>st</sup> production run actuals. Using these data-driven trends, the teams can make a Pareto chart of improvement measures, similar to the results in [25].

Option 2 in Fig. 3 is using historical data to establish a standard cut-off ratio of different active-state durations (such as “producing” and “down” states), exploring whether the forecast value is higher or lower than the cut-off value and making improvement decisions accordingly. For example, Table 3 shows that for the 171<sup>st</sup> production run, machine M5’s forecast “down” state duration is 13.27% of its total active time. If we assume that the cut-off time for the “down” state is 10% of the total active time, then the forecast value is higher than the cut-off value. This requires attention, with measures focussed on reducing the “down”

**Table 5**  
Individual active state forecasts for predicted bottleneck machines.

Production run	Predicted bottleneck machine	Predicted active period %	Individual active states forecast (% of total active period)					
			Producing			Down		
			ARIMA model order	Forecast	Actual	ARIMA model order	Forecast	Actual
170	–	–	–	–	99.93	–	–	0.07
171	M5	85.38	0, 1, 1	86.73	83.69	0, 1, 1	13.27	16.31
172	M4	87.12	0, 1, 1	78.45	74.64	0, 1, 1	21.55	25.36
	M5	87.94	1, 0, 1	76.56	73.06	1, 0, 1	23.44	26.94

state’s duration. If these options are not feasible, then the focus might shift to addressing the “producing” state. This might entail looking at various measures to further improve the “producing” state, such as analysing variations in cycle time, maximising utilisation of M5, exploring opportunities for reducing cycle time and improving the quality of products before M5.

Option 3 in Fig. 3 is manually assessing the forecast values of individual active states, using the production and maintenance teams’ experience as a decision basis.

Though there are different options, their aim is to guide production and maintenance teams in taking the right improvement measures. If multiple machine active states are contributing to a machine’s bottleneck behaviour, trade-offs can be made. Following careful consideration of the upsides and downsides, an evaluation can be made of which bottleneck machine states need more attention.

After assessing the specific active state that needs the most attention, appropriate throughput improvement measures can be prescribed. Ideally, the prescription of the improvement measures can be based on previous historical measures in bottlenecks. As there was no data available on historical improvement measures in MES by the manufacturing company, Table 6 presents a generic list of shop-floor throughput improvement measures based on the machines’ most common active states. These should be seen as generic guidelines to assist production and maintenance teams in proactive planning, thereby allowing them to manage the bottleneck effectively. However, these improvement measures need adaptation, based on the nature of the machines and production system.

**5.3. Evaluation**

To test the proposed algorithm’s forecasting accuracy for each active state, the individual active states’ proportion as a percentage of active time (“producing”, “down” states) for all machines are predicted for every production run in the test dataset. As the prediction window (meaning the historical data) is only the past 50 production runs, the algorithm is tested for 265 out of 315 production runs. As “producing” and “down” states are the only active-period states, predicting one state leads to predicting the other. Thus, the evaluation metric values of the “producing” state will be equal to those of the “down” state.

Table 7 has a summary of the MSE and MAE values in the proposed algorithm (including their standard errors) for the forecast proportion of “producing” state in all machines. Moreover, these MSE and MAE values are then compared with the naïve algorithm to assess the value added by the algorithm, in terms of predicting machine states (also summarised in Table 7). The overall performances of the two algorithms are then tested using a t-test to assess the statistical significance of their performance.

The t-values in Table 7 show that, statistically, the proposed algorithm significantly outperforms the naïve method in predicting future values of individual active machine states. This indicates that the algorithm may potentially explain variations in the “producing” state much better (in time order) than the naïve method, thus adding value towards prediction.

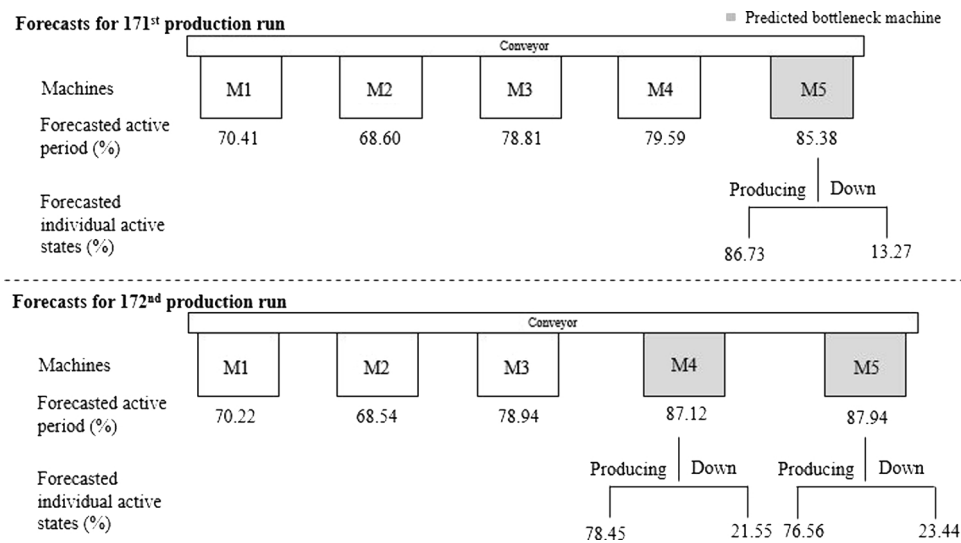


Fig. 6. Diagram of prediction results for production runs 171 and 172.

### 6. A data-driven prescriptive approach to bottleneck management

A diagram of the overall approach to realising prescriptive bottleneck management in the 171<sup>st</sup> production run of the test study appears in Fig. 7. Probable bottlenecks in the system are predicted from MES data, followed by forecasting of the individual active state as a proportion of active time. Finally, the forecast proportions (for example) are compared with the earlier production run to identify trends and suggest suitable prescriptive bottleneck management measures. This forms a prescriptive approach to throughput bottleneck management. The prognostic algorithmic insights are useful in assisting the production and maintenance teams in choosing the right measures, based on forecast machine active-state durations. The production and maintenance teams are experts when it comes to implementing specific measures. The suggested measures shown in Table 5 should be evaluated by the teams and relevant options explored. This integrates the experience-based knowledge of the production and maintenance teams within the prescriptive approach. Thus, this prescriptive approach aims to provide necessary assistance to the teams in making their final decisions. Moreover, the improvement measures taken by different teams can be updated continuously, to improve the prescription of different measures in different machines. Overall, this prescriptive approach reduces the ambiguity between production and maintenance teams by prioritising specific throughput improvement measures in bottlenecks. This approach is a step towards making the right throughput improvement decisions. It adapts the prescriptive information content (based on measures taken by different teams) and forecast future behaviour of the machine [24]. This type of systematic prescriptive

approach enables joint production and maintenance planning.

### 7. Discussion

This study contributes to the development of the prognostic algorithm in predicted bottleneck machines, by which the root causes of bottlenecks (in terms of machine active states) can be understood. Furthermore, it enables prescription of improvement measures. The developed algorithm has been tested on a real-world production line. From this real-world test study, it is understood that there are multiple root causes of bottlenecks and decisions on improvement measures must be made after analysing the trends in bottlenecks. A practical explanation of the usage of the algorithm insights is also made through the test study. Compared to previous studies on data-driven bottleneck prediction (as proposed by [12,15,17] which provide no guidelines as to what measures can be taken based on active states), the algorithm proposed in this study explains the root causes of bottlenecks. It does so in terms of machines' active-state durations and provide guidelines on specific measures. This advances the understanding of data-driven bottleneck analysis, specifically by using predictive algorithms to prescribe measures before a production run, based on historical digital machine-state data. Moreover, the solution presented in this paper aligns with industry's need to develop data-driven solutions and prescribe potential throughput improvement measures. This is an alternative to using discrete event simulation models of bottleneck analysis, as presented by [25].

An additional novelty of the proposed algorithm in this paper is its structured conversion of event-log-type machine data into a set of

Table 6  
Generic, practical shop-floor improvement measures based on active states of predicted bottleneck machines.

Production resource	Active state	Specific measures
Machine	Producing	<ul style="list-style-type: none"> <li>● Maximising utilisation by running machines in scheduled and unscheduled breaks [2] or even over-time.</li> <li>● Checking cycle time variations of the machine [3].</li> <li>● Exploring opportunities to reduce cycle time [3].</li> <li>● Buffering before bottlenecks to ensure a continuous supply of materials [2].</li> <li>● Improving quality before the bottleneck.</li> <li>● Prioritising reactive maintenance [4] to improve response time.</li> </ul>
	Down	<ul style="list-style-type: none"> <li>● If an operator is involved, training operators, having an extra operator to share the workload or having relief staff.</li> <li>● Prioritising preventative (daily preventative measures) and reactive maintenance [4] to improve response time.</li> <li>● Checking the condition and monitoring component data (e.g. sensor data, logistics).</li> </ul>
	Changing tools	<ul style="list-style-type: none"> <li>● Estimating windows of opportunity during the production run to carry out maintenance measures and reduce overall downtime [5].</li> <li>● Reducing tool-changing time by implementing lean practices.</li> <li>● Predicting tool changeover time and exploring whether it can be done when machine is idle.</li> </ul>

**Table 7**  
Performance comparison of proposed diagnostic predictive algorithm with the naïve method.

	Proposed algorithm					Naïve algorithm				
	Machines					Machines				
	M1	M2	M3	M4	M5	M1	M2	M3	M4	M5
MAE	9.81	9.60	8.17	6.97	8.21	12.96	12.39	11.82	9.65	11.07
Standard error	0.58	0.75	0.77	0.40	0.76	0.77	0.83	0.80	0.62	0.84
t-state at 95% confidence level	–	–	–	–	–	3.27	2.50	3.29	3.65	2.52
MSE	186.45	157.96	158.20	90.56	119.02	322.48	333.42	309.77	194.12	307.31
Standard error	18.10	54.63	49.75	15.42	63.58	38.24	52.45	46.34	26.10	62.37
t-state at 95% confidence level	–	–	–	–	–	3.22	2.32	2.23	3.42	2.11

matrices. This allows the forecasting algorithms to predict the future and recommend machine active-state-based measures in bottlenecks. In the proposed prognostic algorithm, only step 4 shown in Fig. 2 is specific to the forecasting methodology and in this study, we have used ARIMA. Ideally, any other univariate time-series forecasting methodologies such as recurrent neural networks or even any ensemble forecasting methodology can be substituted in this step. In other words, matrix C (as shown in Eq. (4)) serves as the base model to which other forecasting methodologies can be applied. Moreover, the evaluation and benchmarking framework proposed in this study can be used to evaluate other forecasting algorithms when durations of active states are to be predicted. Thus, the evaluation framework can be considered as a tool to benchmark the performance of different algorithms. Overall, it should be noted that, to add value to the forecast, the different forecasting algorithms should be benchmarked with that of the naïve method [34]. In practice, this will help companies evaluate whether to have a predictive algorithm for bottleneck analysis.

There are some working limitations that to be factored in when considering the institutionalisation of this algorithm. MES data must record the individual active states and their timestamps, if they are to predict the future values. As the algorithm is based only on machine states, it can explain predicted bottlenecks only in relation respect to those states; there is no other information. There may be many factors that affecting each machine state, but no further insights into them. However, the algorithm can indicate trends in machine states. This can lead to further exploration of other factors, using other data sources to

investigate the changing trends. Finally, the proposed algorithm assumes that the existing historical patterns will continue into the future and that accounting for other causal factors of external events will be factored into future work.

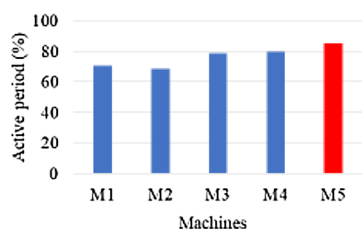
**8. Conclusion**

The production and the maintenance teams have many difficult decisions to make regarding bottleneck management. Having advance notice of a bottleneck’s location and possible root causes in terms of machine active states can help them make better-informed decisions. Previous data-driven research efforts were focused on predicting the location of throughput bottlenecks in the production system. They don’t give much information about the root causes of the bottlenecks to production and maintenance teams. Therefore, in this study, a data-driven prognostic algorithm was developed to forecast the individual active-state duration of a predicted bottleneck machine and explain the multiple root causes of bottlenecks. This was tested in a real-world production system. Explaining possible root causes allows prescription of specific potential machine-state-based throughput improvement measures.

Several areas of further research are needed to effectively prescribe improvement measures.

Firstly, in addition to the connection between system-level decisions on bottleneck machine detection and machine-level decisions on different bottleneck machine states from MES data (as presented in this

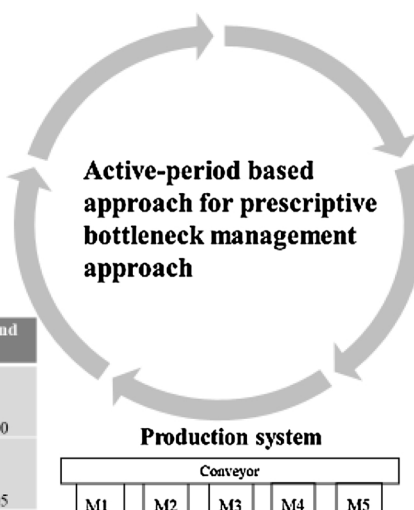
**2. Bottleneck location prediction**



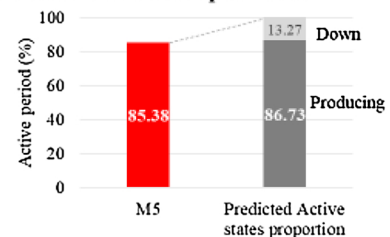
■ Predicted bottleneck machine

**1. MES data collection**

Machine	State of the machine	Duration (s)	Date and time
M2	Producing	65	05-09-2017 16:03:00
M2	Down	120	05-09-2017 06:04:05



**3. Bottleneck machine states prediction**



**4. Decision-making based on algorithmic insights (Prescribing actions)**

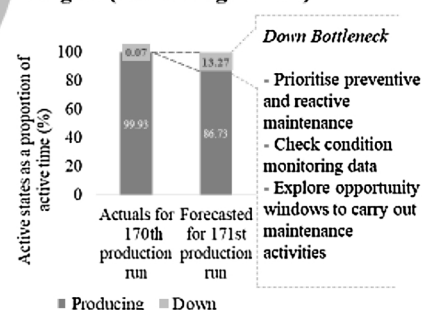


Fig. 7. A Data-driven prescriptive approach for throughput bottleneck management using active-period theory for a 171<sup>st</sup> production run.

paper), the research should focus on combining sensor information from components in the bottleneck machine with machine-state information. This will allow root causes of bottlenecks to be better described and measures to be prescribed. Doing this allows prescriptive algorithms to be used to recommend even more detailed specific, proactive measures for the shop-floor production and maintenance teams.

Secondly, the research presented in this paper forms the basis to develop a recommender system, based on active-states of bottleneck machines. This can recommend specific measures, based on historical measures conducted in bottleneck machines. This can be achieved using collaborative filtering or content-based, filtering-type recommender systems. However, to achieve this, manufacturing companies are encouraged to store detailed digital work records on bottlenecks.

### Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

### Acknowledgements

The authors would like to thank the FFI programme (funded by VINNOVA, the Swedish Energy Agency and the Swedish Transport Administration) for their funding of the Data Analytics in Maintenance Planning research project (DAiMP) [Grant number: 2015-06887], under which this research was conducted. The authors would also like to thank Anders Ramström, who furnished the real-time data from a real-world production system. The authors would also like to thank other industrial partners of the DAiMP research project, for sharing their views on the importance of bottleneck prediction in manufacturing. The authors are also grateful to Associate Professor Andreas Eriksson for sharing his insights on academic writing, which helped the authors improve the manuscript. This work has been conducted within the Sustainable Production Initiative and Production Area of Advance at Chalmers.

### References

- [1] Goldrat E, Cox J. *The goal: a process of ongoing improvement. Third Revi.* Great Barrington, MA: North River Press; 1990.
- [2] Roser C, Nakano M, Tanaka M. Constraint management in manufacturing systems. *JSME Int J Ser C Mech Syst Mach Elem Manuf* 2003;46:73–80.
- [3] Wu K. An examination of variability and its basic properties for a factory. *IEEE Trans Semicond Manuf* 2005;18:214–21.
- [4] Li L, Chang Q, Ni J, Biller S. Real time production improvement through bottleneck control. *Int J Prod Res* 2009;47:6145–58. <https://doi.org/10.1080/00207540802244240>.
- [5] Gu X, Jin X, Ni J. Prediction of passive maintenance opportunity windows on bottleneck machines in complex manufacturing systems. *J Manuf Sci Eng* 2015;137:1–9. <https://doi.org/10.1115/1.4029906>.
- [6] Tao F, Qi Q, Liu A, Kusiak A. Data-driven smart manufacturing. *J Manuf Syst* 2018;48:157–69. <https://doi.org/10.1016/j.jmsy.2018.01.006>.
- [7] Li J, Blumenfeld DE, Huang N, Alden JM. Throughput analysis of production systems: recent advances and future topics. *Int J Prod Res* 2009;7543. <https://doi.org/10.1080/00207540701829752>.
- [8] Li L, Chang Q, Ni J. Data driven bottleneck detection of manufacturing systems. *Int J Prod Res* 2009;47:5019–36. <https://doi.org/10.1080/00207540701881860>.
- [9] Yu C, Matta A. Data-driven bottleneck detection in manufacturing systems: a statistical approach. *Int J Prod Res* 2016. <https://doi.org/10.1080/00207543.2015.1126681>.
- [10] Subramaniyan M, Skoogh A, Salomonsson H, Bangalore P, Gopalakrishnan M, Muhammad AS. Data-driven algorithm for throughput bottleneck analysis of production systems. *Prod Manuf Res* 2018;6:225–46. <https://doi.org/10.1080/21693277.2018.1496491>.
- [11] Li L. A systematic-theoretic analysis of data-driven throughput bottleneck detection of production systems. *J Manuf Syst* 2018;47:43–52. <https://doi.org/10.1016/j.jmsy.2018.03.001>.
- [12] Li L, Qing C, Xiao G, Ambani S. Throughput bottleneck prediction of manufacturing systems using time series analysis. *J Manuf Sci Eng* 2011;133:1–8. <https://doi.org/10.1115/1.4003786>.
- [13] Lee J, Lapira E, Bagheri B, Han Kao. Recent advances and trends in predictive manufacturing systems in big data environment. *Manuf Lett* 2013;1:38–41. <https://doi.org/10.1016/j.mfglet.2013.09.005>.
- [14] Subramaniyan M, Skoogh A, Salomonsson H, Bangalore P, Bokrantz J. A data-driven algorithm to predict throughput bottlenecks in a production system based on active periods of the machines. *Comput Ind Eng* 2018. <https://doi.org/10.1016/j.cie.2018.04.024>.
- [15] Cao Z, Deng J, Liu M, Wang Y. Bottleneck prediction method based on improved adaptive network-based fuzzy inference system (ANFIS) in semiconductor manufacturing system. *Chin J Chem Eng* 2012;20:1081–8. [https://doi.org/10.1016/S1004-9541\(12\)60590-4](https://doi.org/10.1016/S1004-9541(12)60590-4).
- [16] Wedel M, Noessler P, Metternich J. Development of bottleneck detection methods allowing for an effective fault repair prioritization in machining lines of the automobile industry. *Prod Eng* 2016;10:329–36. <https://doi.org/10.1007/s11740-016-0672-9>.
- [17] Roser C, Lorentzen K, Lenze D, Deuse J, Klenner F, Richter R, et al. Bottleneck prediction using the active period method in combination with buffer inventories. Lödning H, editor. *IFIP Int. Fed. Inf. Process.* 2017, vol. 1. 2017. p. 374–81. <https://doi.org/10.1007/978-3-319-66926-7>.
- [18] Razzak Ma, Shi J, Robinson DE, Huang SH, Dismukes JP, Wang G, et al. Manufacturing system modeling for productivity improvement. *J Manuf Syst* 2002;21:249–59.
- [19] Roser C, Nakano M, Tanaka M. A practical bottleneck detection method. In: Peters B, Smith J, Medeiros D, Rohrer M, editors. *Proc. 2001 Winter Simul. Conf.* 2001.
- [20] Chiang S-Y, Kuo C-T, Meerkov SM. DT-bottlenecks in serial production lines: theory and application. *IEEE Trans Robot Autom* 2000;16:567–80. <https://doi.org/10.1109/70.880806>.
- [21] Brockwell P, Richard D. *Introduction to time series and forecasting.* Second. Springer; 2016.
- [22] Ignizio J. *Optimizing factory performance: cost-effective ways to achieve significant and sustainable improvement.* McGraw-Hill; 2009.
- [23] Guo W, Jin J, Hu SJ. Allocation of maintenance resources in mixed model assembly systems. *J Manuf Syst* 2013;32:473–9. <https://doi.org/10.1016/j.jmsy.2012.12.006>.
- [24] Holm M. The future shop-floor operators, demands, requirements and interpretations. *J Manuf Syst* 2018;47:35–42. <https://doi.org/10.1016/j.jmsy.2018.03.004>.
- [25] Pehrsson L, Ng AHC, Bernedixen J. Automatic identification of constraints and improvement actions in production systems using multi-objective optimization and post-optimality analysis. *J Manuf Syst* 2016;39:24–37. <https://doi.org/10.1016/j.jmsy.2016.02.001>.
- [26] Roser C, Nakano M, Tanaka M. Shifting bottleneck detection. In: Yucesan E, Chen C-H, Snowdon J, Charnes J, editors. *Proc. 2002 Winter Simul. Conf* 2002;2. <https://doi.org/10.1109/WSC.2002.1166360>.
- [27] Bokrantz J, Skoogh A, Berlin C, Stahre J. Maintenance in digitalised manufacturing : Delphi-based scenarios for 2030. *Int J Prod Econ* 2017.
- [28] Diez-olivan A, Del J, Galar D, Sierra B. Data fusion and machine learning for industrial prognosis : trends and perspectives towards industry 4.0. *Inf Fusion* 2019;50:92–111. <https://doi.org/10.1016/j.inffus.2018.10.005>.
- [29] Montgomery DC, Jennings CL, Kulachi M. *Introduction to time series analysis and forecasting.* Second Edi John Wiley & Sons; 2016.
- [30] Hyndman RJ, Khandakar Y. Automatic time series forecasting: the forecast package for R automatic time series forecasting : the forecast package for R. *J Stat Softw* 2008;27:1–22. <https://doi.org/10.18637/jss.v027.i03>.
- [31] Roser C, Nakano M. Confidence interval from a single simulation using Delta method. *JSME Int J Ser C Mech Syst Mach Elem Manuf* 2003;46:67–72.
- [32] Kolassa S. Evaluating predictive count data distributions in retail sales forecasting. *Int J Forecast* 2016;32:788–803. <https://doi.org/10.1016/j.ijforecast.2015.12.004>.
- [33] Hyndman RJ. *Forecasting: principles & practice.* University of Western Australia; 2014. doi:25- September-2014.
- [34] Makridakis S, Spiliotis E, Assimakopoulos V. Statistical and machine learning forecasting methods : concerns and ways forward. *PLoS One* 2018;13:1–26.
- [35] Montgomery DC, Runger GC. *Applied statistics and probability for engineers.* John Wiley & Sons; 2010.