



Received: 09 June 2016
Accepted: 17 September 2016
First Published: 23 September 2016

*Corresponding author: Mukund Subramaniyan, Department of Product and Production Technology, Chalmers University of Technology, Gothenburg 41296, Sweden
E-mail: mukunds@chalmers.se

Reviewing editor:
Jun Guo, Wuhan University of Technology, China

Additional information is available at the end of the article

PRODUCTION & MANUFACTURING | RESEARCH ARTICLE

An algorithm for data-driven shifting bottleneck detection

Mukund Subramaniyan^{1*}, Anders Skoogh¹, Maheshwaran Gopalakrishnan¹, Hans Salomonsson², Atieh Hanna³ and Dan Lämkuhl⁴

Abstract: Manufacturing companies continuously capture shop floor information using sensors technologies, Manufacturing Execution Systems (MES), Enterprise Resource Planning systems. The volumes of data collected by these technologies are growing and the pace of that growth is accelerating. Manufacturing data is constantly changing but immediately relevant. Collecting and analysing them on a real-time basis can lead to increased productivity. Particularly, prioritising improvement activities such as cycle time improvement, setup time reduction and maintenance activities on bottleneck machines is an important part of the operations management process on the shop floor to improve productivity. The first step in that process is the identification of bottlenecks. This paper introduces a purely data-driven shifting bottleneck detection algorithm to identify the bottlenecks from the real-time data of the machines as captured by MES. The developed algorithm detects the current bottleneck at any given time, the average and the non-bottlenecks over a time interval. The algorithm has been tested over real-world MES data sets of two manufacturing companies, identifying the potentials and the prerequisites of the data-driven method. The main prerequisite of the proposed data-driven method is that all the states of the machine should be monitored by MES during the production run.

Subjects: Algorithms & Complexity; Operations Research; Production Systems

Keywords: data-driven; shifting bottleneck; active duration; bottlenecks; big data; decision support; production



Mukund Subramaniyan

ABOUT THE AUTHOR

Mukund Subramaniyan is currently working as a Project Assistant at the Department of Product and Production Development at Chalmers University of Technology. He holds a Msc degree in Production Engineering from Chalmers University of Technology. His research interests are data analytics in production, decision support systems, and machine learning applications for production system control. His current research focuses on developing data-driven algorithms for bottleneck detections in production systems within StreamMod Project (Streamlined Modelling for Decision Support for Fact-based Production Development) funded by Vinnova.

PUBLIC INTEREST STATEMENT

Manufacturing companies are collecting large amounts of machine data using IT systems. Decision-making based on the collected data, has gained substantial attention in the recent years in order to increase the productivity. One such useful decision support for the production and maintenance personnel is the real-time bottleneck detection as it helps to prioritise the resources on a real-time basis to the most critical machines. In this paper, the aim is to use the collected manufacturing data and develop an algorithm to detect the bottlenecks. Real-world production data is used to develop the algorithm and it is tested on real automotive production lines. The algorithm can detect the current and the average bottlenecks and also the non-bottlenecks in the production system. This type of decision support on the different bottleneck machines will help to frame effective strategies in order to increase the overall throughput of production systems.

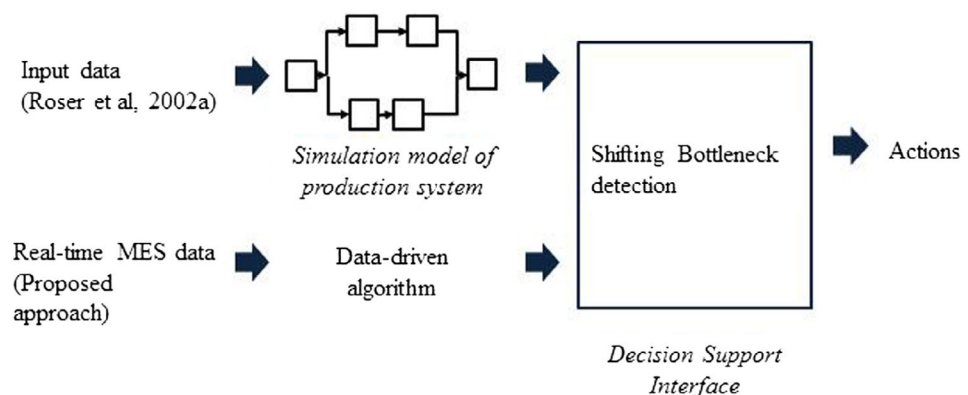
1. Introduction

Digital solutions are helping manufacturing companies to improve productivity and remain globally competitive. The manufacturing companies generate and store data from a multitude of sources, using sensor technologies, Manufacturing Execution Systems (MES), Enterprise Resource planning (ERP) and other production planning systems. Operations in many organisations are experiencing much more voluminous data environments because of real-time information, which could facilitate the improvement in manufacturing efficiency and effectiveness (Zelbst, Green, Sower, & Abshire, 2011). A preliminary study at an automotive manufacturing company in Sweden reveals that, on an average, 100 data rows are collected per hour per machine by the MES. This means that 500,000 data rows are collected per year per machine (Subramaniyan, 2015). The increasing complexity of production processes and layouts produced by big data has brought about a situation in which the human mind is not capable of analysing such large amounts of production data for decision-making. Many manufacturing companies have therefore started to explore better ways to utilise big data, using advanced analytics to make fact based decisions (O'Donovan, Leahy, Bruton, & O'Sullivan, 2015). Big data has the potential to enable data-driven decision-making, which helps to make better decisions (Bean & Kiron, 2013; Davenport, Barth, & Bean, 2012).

The production rate, often referred to as throughput, is a major indicator of the production system performance. The production system throughput is constrained by one or more machines in the system, called bottlenecks (Goldrat & Cox, 1992). The bottleneck is defined as the machine to which the overall system throughput has the largest sensitivity (Li, Chang, & Ni, 2009). The resources (e.g. machines, robots, operators etc.) available for the production are usually scarce, and hence, the decisions on improving the throughput requires the efficient utilisation of these resources (Li, Ambani, & Ni, 2009). Therefore, it is very important for the manufacturing companies to identify the machines constraining the system and then make efforts to increase the throughput of those machines. For example, efforts on cycle time improvement (Nishimura, 1997) and downtime variability reduction (Langer et al., 2010) can be focused on bottleneck machines to improve the throughput from the system. Furthermore, prioritizing various maintenance activities on bottleneck machines has proved a great potential to improve the throughput (Gopalakrishnan, Skoogh, & Christoph, 2014; Li & Ni, 2009). For example, the information on planning maintenance activities on bottlenecks is available in maintenance decision support systems such as Computerised Maintenance Management System (CMMS) or Plant Level Maintenance Decision Support tool (PMDSS) (Li, Ambani, et al., 2009; Li & Ni, 2009). To do all this, a system-level decision support tool is required to identify the bottlenecks. The tool should aid decision-making by individuals through easier access to problem recognition (Santana, 1995).

Most current bottleneck detection methods are simulation-based approaches using a simulation model of the production system. This approach is time consuming and requires huge efforts in terms of developing the simulation model of the production system, updating the simulation model with the changes made in the real production system (Skoogh, 2011). Moreover, there are a lot of assumptions and approximations made to input data in building the simulation model (Skoogh, 2011). Also, it takes

Figure 1. Simulation tool and data-driven method for decision support system.



a lot of time to run the analysis and therefore it is difficult to obtain results on a real-time basis (Li, Chang, et al., 2009). With the development of computer technology and the amounts of data collected by MES, data-driven algorithms using the online production data present a new way to detect the bottlenecks on a real-time basis. Data-driven algorithms can detect the bottleneck in shorter time and improve the performance through a fact based decision-making process. Fortunately, considerable research has been done on how to detect the bottlenecks using available online data without building a model of the production system. For instance, Li, Chang, et al. (2009) developed a novel method called turning point method, which uses the blockage and the starvation online data of machines to detect the bottlenecks. However, the online measurements of blockage and starvation times of the machines are greatly affected by the buffers in the production system and therefore fail to detect the true bottlenecks in a completely decoupled system with large buffers and frequent small stoppages in the machines. On the other hand, there is limited research or previous contributions explaining how the various existing simulation based bottleneck detection techniques can be used over real-world MES data to detect bottlenecks without building a model of the production system.

The primary purpose of this paper is to increase the productivity and the robustness in the production system by providing faster decision-making in identification of the bottlenecks from the emerging sets of data in MES systems using data-driven algorithms. The aim of this paper is to develop and test the data-driven algorithm for the active period bottleneck detection method called shifting bottlenecks.

The theory of the shifting bottleneck detection method was developed by Roser, Nakano and Tanaka (2002a). It was tested and validated in a simulation environment (Roser, Nakano, & Tanaka, 2002a, 2002b, 2003a, 2003b) using the discrete simulation model of the production system as shown in Figure 1. This method includes more information on the bottlenecks (e.g. information on current, average and non-bottleneck) when compared to the data-driven turning point method as proposed by Li, Chang, et al. (2009) which only gives information on the average bottlenecks. In this paper, a data-driven shifting bottleneck detection algorithm is developed which uses the MES information to identify the bottleneck, as shown in Figure 1. Also, examples are provided on how the decision support based on data-driven shifting bottleneck can be used. The practical uses of the data-driven shifting bottleneck detection in production management are explained in this paper.

2. Literature review

In the first part of this section, the different bottleneck detection methods and the tools used to develop and illustrate those methods are presented. Also, the motivation of choosing the simulation based shifting bottleneck detection for data-driven approach is explained. In the second part of this section, the theory of the shifting bottleneck detection method and the types of information the method gives on the bottlenecks are explained.

Table 1. Bottleneck detection methods with the support tools used for development

Support tool	Method	Reference
Analytical	Queue length	Lawrence and Buss (1994)
	Utilisation	Hopp and Spearman (2000)
Simulation	Active period percentage (also known as <i>machine workload, Utilisation</i>)	Roser, Nakano, and Tanaka (2001)
	Average active period	Roser et al. (2001)
	Shifting bottleneck detection	Roser et al. (2002a)
	Queue time	Faget et al. (2005)
	Inactive period	Sengupta, Das, and VanTil (2008)
	Inter-departure time variance	Betterton and Silver (2012)
Simulation model coupled with real-time input data	Sensitivity based bottleneck detection	Chang et al. (2007)
Data-driven	Turning point	Li, Chang, et al. (2009)

2.1. Bottleneck detection methods and development tools

Existing studies on bottleneck detection methods can be classified into three categories: Analytical, simulation and data-driven approach as shown in Table 1. The relevant explanations are available in Betterton and Silver (2012).

Most of the aforementioned works in bottleneck detection as shown in Table 1 are model-based approaches, heavily reliant on the explicit process of modelling or developing closed form solutions of mathematical models. However, as the system characteristics change along with the time and the ageing of the machine, modelling errors are introduced in the simulation and in the analytical models, leading to inaccurate bottleneck detection. Moreover, the input data modelling of the parameters of simulation model (e.g. processing times, setup times, Mean Time between Failures (MTBF) etc.) is done using statistical or empirical distributions and is a time consuming activity and includes approximations (Leemis, 2004; Skoogh, 2011). Due to these characteristics, the explicit analytical solution or simulation based solution is not easy to construct for complex production systems. Incorporating real-time data into the simulation model of the production system can greatly improve the accuracy of the bottleneck detection (Chang, Ni, Bandyopadhyay, Biller, & Xiao, 2007). But this approach still needs a simulation model of the production system, which consumes a lot of time and introduces error because of its computational abstraction (Fowler & Rose, 2004).

Motivated by the works on data-driven approaches as shown in Table 1, this paper introduces a data-driven algorithm for an existing simulation based shifting bottleneck detection method (Roser et al., 2002a). The shifting bottleneck detection method was chosen over other bottleneck detection techniques as it provides more information about the bottlenecks e.g. it detects the current bottlenecks at any given time, average bottlenecks and the non-bottlenecks in the production system over the time interval of interest. Also, the information on current bottlenecks and where and when the previous bottleneck was shifting to the current bottleneck helps to better understand the behaviour of the dynamic bottlenecks on the shop floor. This in turn will help to identify the machines for prioritization of improvement activities on a short-term basis e.g. machine working time improvement to improve the throughput, dynamic buffer allocation, prioritization of reactive and preventive maintenance on bottleneck machines. Moreover, it has been proved by Roser et al. (2002a) that this method detects the bottlenecks with more accuracy.

3. Theoretical framework on shifting bottleneck detection

Roser et al. (2002a) demonstrates the shifting bottleneck method using the simulation model of the production system. The term shifting bottleneck is based on the active periods of the machines. The term active period is defined as the state of the machine when the machine is producing a part or some actions are being performed on the machine, such as repair activities during breakdowns, setups, tool changes etc. (Roser et al., 2002a). An example timeline of the states of a machine during a production run is shown in Figure 2.

The main idea of this method is that, at any given time instant t , the machine with the longest uninterrupted active period among the other machines in the production line is called the momentary bottleneck or current bottleneck at the time instant t . If no machines are active at time t , then there is no bottleneck. The overlap of the active period of the bottleneck machine with the previous

Figure 2. Active and inactive states of the machine.

Source: Adopted from Roser et al. (2002a).

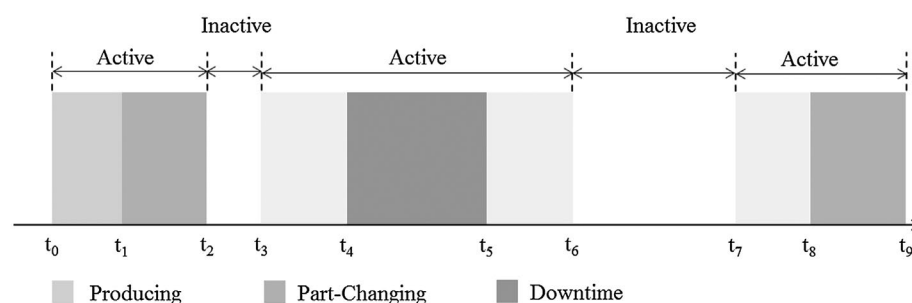


Figure 3. Shifting bottlenecks.

Source: Adopted from Roser et al. (2002a).

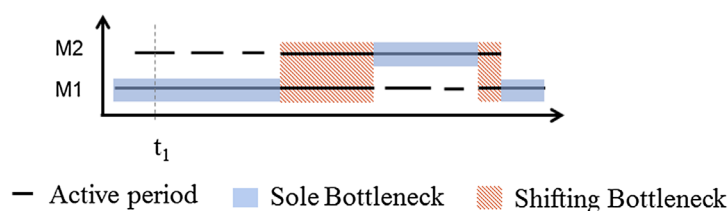
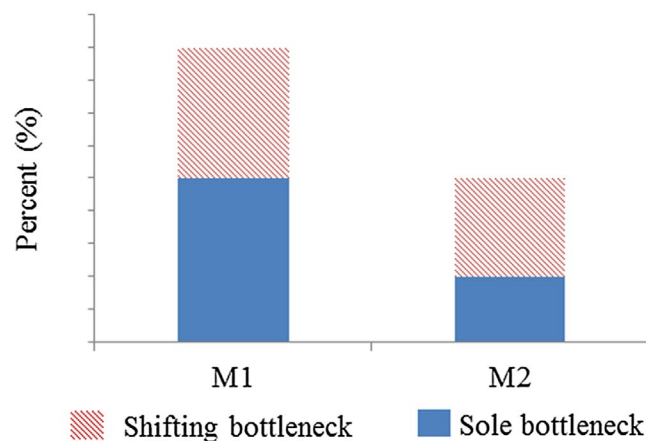


Figure 4. Average bottlenecks.

Source: Adopted from Roser et al. (2002a).



or the subsequent bottlenecks represents the shifting of the bottleneck from one machine to the other. If the bottleneck is not shifting, then the current bottleneck machine is the sole bottleneck. For example, Figure 3 shows the shifting bottlenecks and sole bottlenecks of two machines (M1 and M2) in a production system during a production run. The amount of time during the production run when machines M1 and M2 are sole bottlenecks and shifting bottlenecks is shown in Figure 4. From Figure 4, it can be seen that, on an average, machine M1 had the highest impact in limiting the throughput of the production line. Hence machine M1 is the main bottleneck.

Figure 5. Overview of the approach.

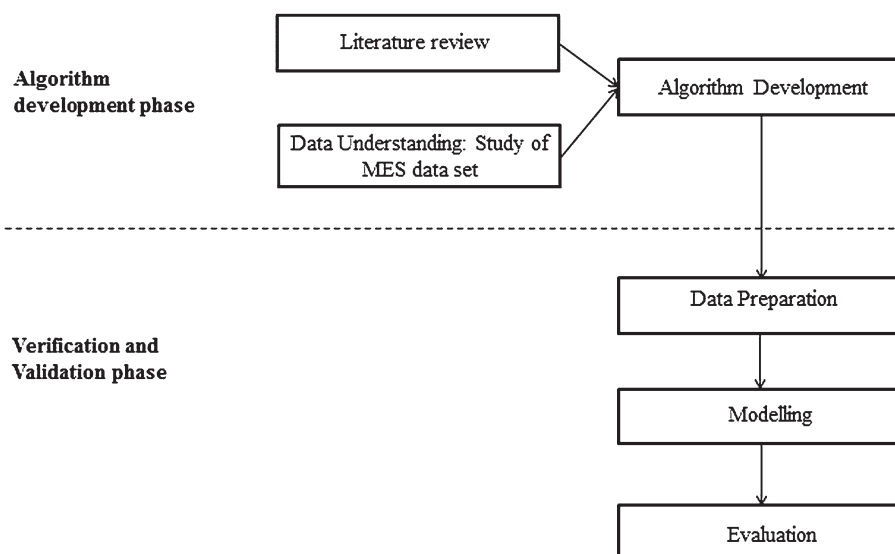


Table 2. Example MES record of one machine with the machine status and time stamps

Production area	Work area	Date with time	State of the machine
Line 1	M1	01-09-2014 06:28:02	Not active
Line 1	M1	01-09-2014 06:28:25	Comlink up
Line 1	M1	01-09-2014 06:29:20	Not active
Line 1	M1	01-09-2014 06:29:34	Waiting
Line 1	M1	01-09-2014 06:29:34	Waiting
Line 1	M1	01-09-2014 06:42:46	Producing

4. Methodology

The main aim of this paper is to develop data-driven algorithms for the simulation based shifting bottleneck detection method. The methodology adopted was based on the Cross-Industry Standard Process for Data Mining (Pete et al., 2000). Figure 5 shows the work flow of the methodology adopted.

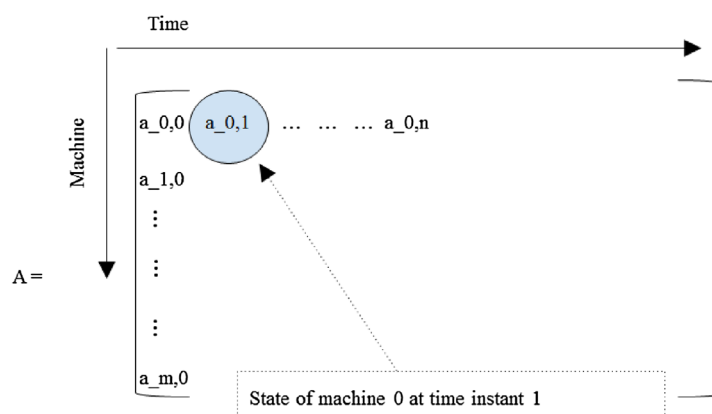
Before developing the algorithm, a detailed literature study was carried out, focussing on step by step logics and methodology used to develop the shifting bottleneck detection method in the simulation environment as proposed by Roser et al. (2002a). In addition to that, a detailed study on a MES data-set of an automotive production line as shown in Table 2 was carried out. This was to get a clear understanding of the structure of the MES data-set and the machine information as captured by the MES. The knowledge gained by studying the MES data-set of a production line in combination to the detailed literature studies, was used to develop the data-driven algorithm for a shifting bottleneck detection method. This constitutes the algorithm development phase.

The next phase is the verification and validation phase, in which the algorithm is tested in a real world scenario. Verification is defined as the process of evaluating the rule definitions of the algorithm satisfying the necessary conditions and validation is defined as process of evaluating the rules satisfying the end user requirements (Lengyel, 2015). The algorithm was tested on the MES data of two production lines to verify whether the algorithm is in accordance with the shifting bottleneck theory developed by Roser et al. (2002a) and validated this by evaluating whether it satisfies the types of decision support required by the production and maintenance personnel in charge of the production lines. The data preparation step in this phase includes the preparation and the cleaning of the MES data. The cleaning of the MES data includes removal of the duplicate data rows and removing the non-production days log information from MES. In modelling step, the algorithm is applied to the cleaned real-time MES data. The results obtained from applying the algorithms are interpreted in order to identify the bottleneck machines in the production line in the evaluation step.

5. Data-driven shifting bottleneck algorithm

A data-driven shifting bottleneck detection algorithm is proposed with the step-by-step construction of the algorithm in the first part of this section, followed by an example to illustrate the working methodology of the algorithm.

Figure 6. Input matrix A.



5.1. Construction of the algorithm

In order to find the current bottleneck, average bottleneck and the non-bottlenecks over the time interval, the first step is to identify the different potential bottlenecks and the second step is to identify the shifting and sole bottlenecks within the potential bottlenecks. The following are the notations used to develop the algorithm.

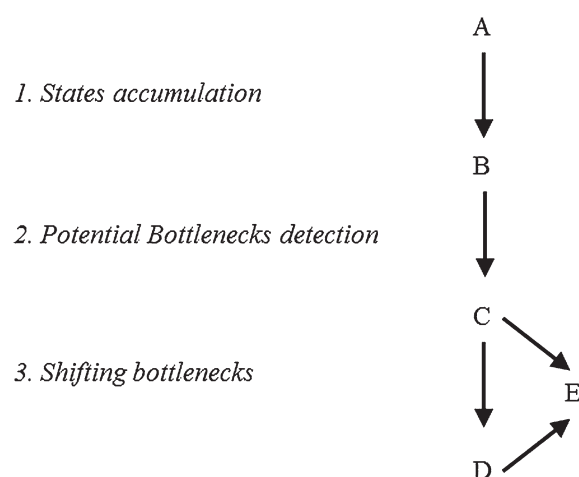
m = number of machines; n = number of time instants; A = states matrix of size $m \times n$; B = accumulated states matrix of size $m \times n$; C = bottlenecks matrix of size $m \times n$; D = sole bottlenecks matrix of size $m \times n$; E = shifting bottlenecks matrix of size $m \times n$; i = index of machine; i^* = index of current bottleneck machine; j = index of time instant; s_j = number of bottlenecks at time instant j ; a_{ij} , b_{ij} , c_{ij} , d_{ij} , e_{ij} = Elements in the matrix A , B , C , D and E .

The algorithm starts with a matrix A of size $m \times n$. Each entry in the matrix, a_{ij} , corresponds to the binary state of the machine i at time j and takes the value 1 if the machine is active and 0 if the machine is inactive as shown in Figure 6.

The matrix A is then transformed in three steps to reveal the sole and the shifting bottlenecks. The three step transformations are shown in Figure 7. The sole bottlenecks in the matrix E are calculated as the element-wise difference between matrices C and D .

The following is the general description of the step by step procedure in the sole and the shifting bottleneck detection algorithm.

Figure 7. Overview of the algorithm.



Algorithm: Sole and shifting bottleneck detection method

Input: Matrix A

Output: Current bottleneck, average bottlenecks and non-bottlenecks

States Accumulation transformation

- 1 Set $i = 0$ and $j = 0$
- 2 If $i < m$ go to Step 3. Otherwise go to Step 7
- 3 If $j < n$ go to Step 4. Otherwise set $j = 0$ and go to Step 6
- 4 If j is 0 set $b_{ij} = a_{ij}$. Otherwise set $b_{ij} = a_{ij} * (b_{ij-1} + 1)$
- 5 Set $j = j + 1$ and go to Step 3
- 6 Set $i = i + 1$ and go to Step 4

Potential Bottlenecks detection transformation

- 7 Set $j = n-1$ (last time instant)
- 8 Find machine i^* with the highest accumulated state, b_{ij^*} at time instant j and set $c_{i^*j} = 1$. If all elements are zero set $j = j-1$ and go to Step 11. Otherwise go to Step 9
- 9 Set every element $c_{i^*j^*} = 1$ for $k < j^* < l$, where k is the last time instant machine i^* was inactive before j and l the first inactive state after j
- 10 Set $j = k$
- 11 If $j < 0$ go to Step 13. Otherwise go to Step 8
- 12 Machine i^* is the current bottleneck

Shifting bottlenecks

- 13 Set $j = 0$
- 14 Calculate $s_j = \sum_{i=0}^{m-1} c_{ij}$. If $s_j > 1$ set $d_{ij} = c_{ij}$ for all i
- 15 Set $j = j + 1$
- 16 If $j < n$ go to Step 14. Otherwise go to Step 18
- 17 Calculate the shifting bottleneck percentage of machine i according to equation below,

$$r_i^{(shifting)} = \frac{1}{n} \sum_{j=0}^{n-1} d_{ij} \quad (1)$$

Sole Bottlenecks

- 18 Set $e_{ij} = c_{ij} - d_{ij}$
- 19 Calculate the sole bottleneck percentage of machine i according to equation below,

$$r_i^{(sole)} = \frac{1}{n} \sum_{j=0}^{n-1} e_{ij} \quad (2)$$

The output of the algorithm is the current bottleneck and the average bottlenecks. The non-bottlenecks are those machines which do not have any sole and the shifting bottlenecks empirics. The algorithm detects the bottlenecks at any instant of time as the newly streamed data will consist of a new column that is appended to matrix A and matrix B is trivially updated with a new column filled with values according to Step 4. C is updated by finding i^* , as in Step 8, and updated as in Step 9. D is updated by Step 14 for the new column. Finally, E is updated by Step 18.

Figure 8. State accumulation transformation.

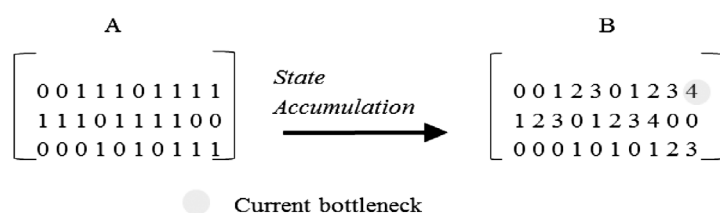
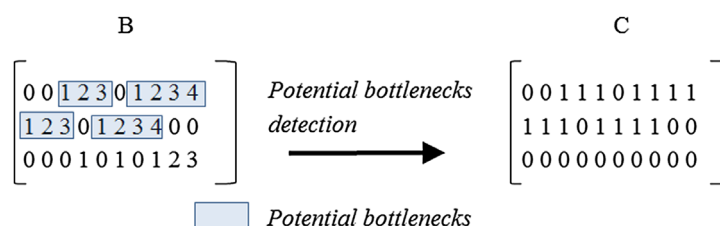


Figure 9. Potential bottlenecks detection transformation.



5.2. Illustration of the algorithm

Consider three machines in a serial production line with two buffers (3M2B). The current bottleneck and the average bottlenecks are calculated at the tenth second after the production run starts. Matrix A represents the states of the machines at each time instant as shown in Figure 8. “1” represents the active of the machine and “0” represents inactive state of the machine. In the states’ accumulation transformation, the active states are accumulated through time for each machine and are reset when there is an inactive state as shown in Figure 8.

The potential bottlenecks are found from highest accumulated active state in the matrix B for each time instant. Since a machine could be active before it becomes a bottleneck, it is necessary to go back in time to find the new potential bottleneck until the machine is inactive as shown in Figure 9.

The shifting bottlenecks are then revealed by the potential bottlenecks matrix C, where there is an overlap between two bottlenecks as shown in Figure 10.

The sole bottlenecks are then revealed by subtracting the matrix D from matrix C as shown in Figure 11.

Figure 10. Shifting bottlenecks.

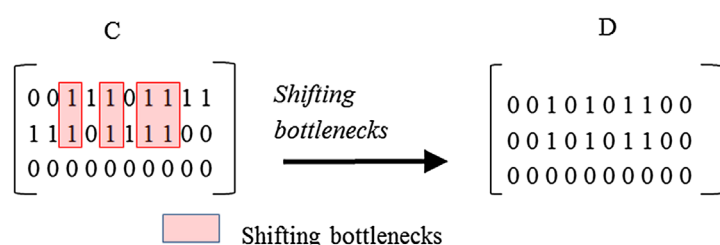


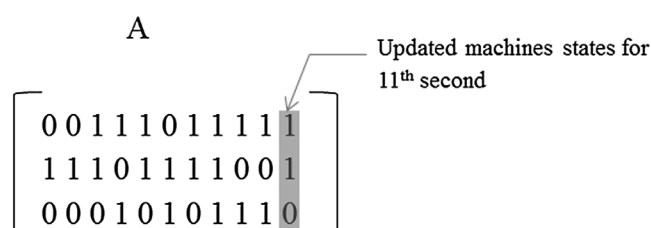
Figure 11. Sole bottlenecks.

$$\begin{bmatrix} 0 & 0 & 1 & 1 & 1 & 0 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 0 & 1 & 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} - \begin{bmatrix} 0 & 0 & 1 & 0 & 1 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 1 \\ 1 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

Table 3. Sole and shifting empirics

Machine	Sole (%)	Shifting (%)	Total (%)
M1	30	40	70
M2	30	40	70
M3	–	–	–

Figure 12. Updated Matrix A for the new data.



The sole and the shifting percentages of each machine are then calculated using Equations (1) and (2) as shown in Table 3.

The result from the algorithm shows that machine *M1* is the current bottleneck at the tenth second after production started, the average bottlenecks are machines *M1* and *M2* and the non-bottleneck is machine *M3*.

When the new data of the machine states is available, e.g. when the data is available for the eleventh second for all the three machines, then matrix *A* gets updated as shown in Figure 12 and the new set of sole and shifting bottlenecks calculations can be carried out as described in Section 5.2.

6. Industrial test results

To test and verify the implementation and efficiency of the developed data-driven shifting bottleneck detection algorithm, this was implemented in two production lines taken from two different automotive manufacturing companies. The two production lines, referred to as Test Study 1 and 2, are described here along with the results. Production and maintenance engineers working at these two production lines face challenges on where to allocate the resources and prioritize the improvement and maintenance activities in order to increase the throughput. Therefore, it is important to be aware of the bottleneck machines at any given point of time from a production system perspective in order to improve the throughput of the system.

Table 4. Definition of the states of the machine

Machine states	Explanation
Producing	Machine is engaged in producing a product
Part changing	Machine in undergoing a setup
Error	Machine is taken down due to tool or machine failure
Comlink down	Connection between MES system and the main server is interrupted
Comlink up	Connection between MES system and the main server is resumed
Waiting	When the machine is blocked or starved
Not active	When the machine stops apart from above mentioned reasons
Empty run	When the machine produces new products or trial products

Figure 13. Layout of the production line.

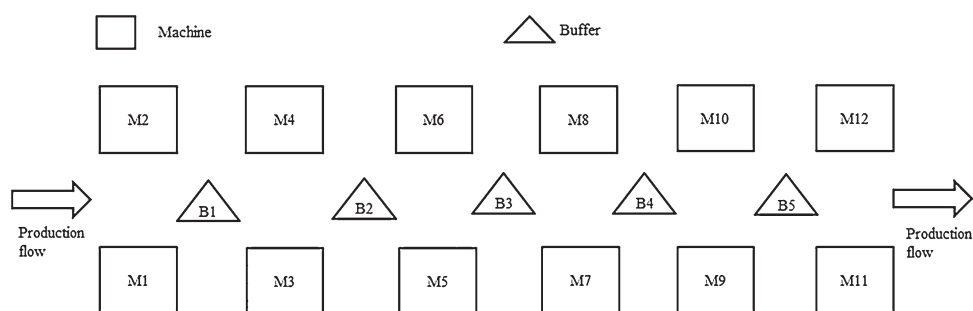
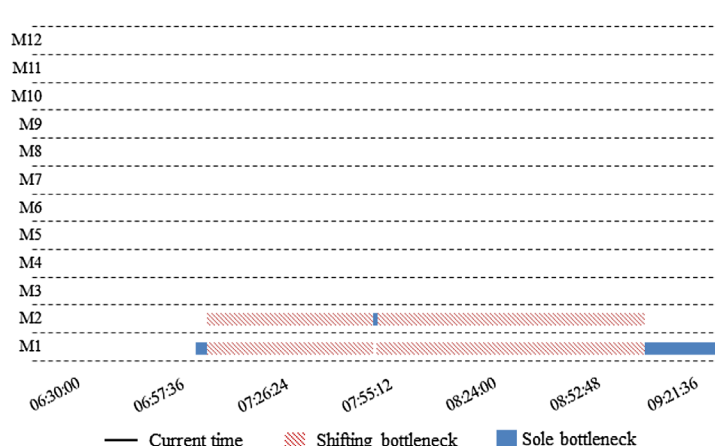


Figure 14. Sole and shifting patterns of the machines at 09:30:00.



6.1. Test study 1

The first test study was made on the production system from an automotive manufacturing company consisting of parallel machines. The production and the maintenance team of this line are looking for methods that can use the MES real-time data to detect the bottlenecks more quickly and accurately. The production system consists of 12 machines as shown in Figure 13. The machines M1 and M2, M3 and M4, M5 and M6, M7 and M8, M9 and M10 and M11 and M12 are parallel machines. There are buffers between the parallel set of machines. Each of the machines is connected to the MES. The machine activities at different time instants are recorded by the MES.

There are eight states of the machine as recorded by the MES. These are *Producing*, *Partchanging*, *Error*, *Comlink down*, *Comlink up*, *Waiting*, *Not active* and *Empty run*. The definitions of each state are explained in Table 4. All the machines are at one of the eight given states at a given time. An example of the MES of machine M1 is shown in Table 4.

From Table 5, it can be seen that the MES monitors all the different states of the machine along with the time stamps in a continuous manner. This satisfies the primary requirement for the data-driven algorithm to be implemented.

Table 5. MES data structure of the M2 machine

Production area	Work area	Date with time	State of the machine
Line 1	M2	01-09-2014 06:28:02	Not active
Line 1	M2	01-09-2014 06:29:30	Producing
Line 1	M2	01-09-2014 06:33:59	Part changing
Line 1	M2	01-09-2014 06:45:02	Waiting
Line 1	M2	01-09-2014 07:00:05	Producing
Line 1	M2	01-09-2014 07:30:12	Error

Figure 15. Average sole and shifting bottlenecks until 09:30:00.

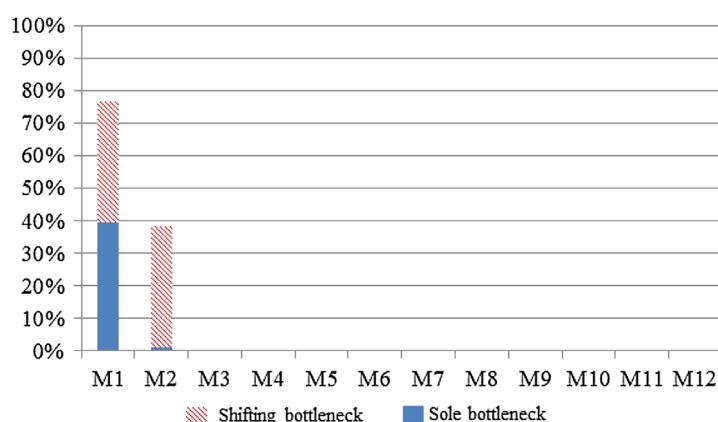


Figure 16. Sole and shifting patterns of the machines at 12:30:00.

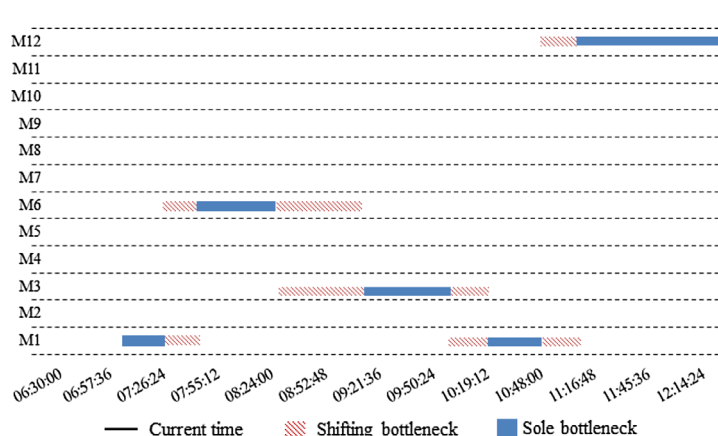
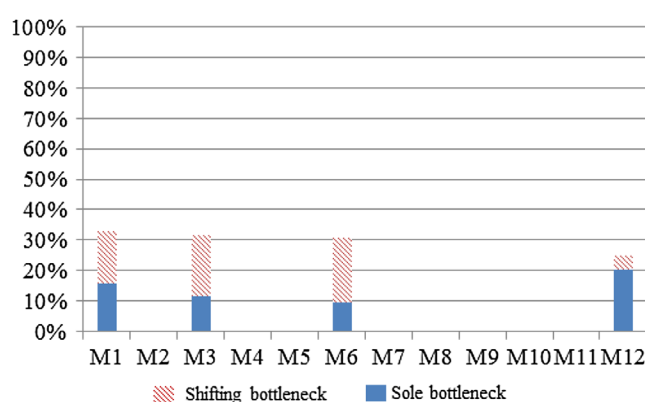


Figure 17. Average sole and shifting bottlenecks till 12:30:00.



6.1.1. Steps involved in the application of shifting bottleneck detection algorithm

Data Preparation: The data-driven shifting bottleneck algorithm detects the bottleneck at any time of interest during the production run. So the MES data of all the individual machines are collected from the time interval of interest. Thereafter the time stamps of each individual machine are checked for overlaps. Also, the data rows outside the shift time interval of interest are then removed. The cleaned MES data records of all the machines are then integrated into one single data file.

Figure 18. Sole and shifting patterns of the machines at 14:30:00.

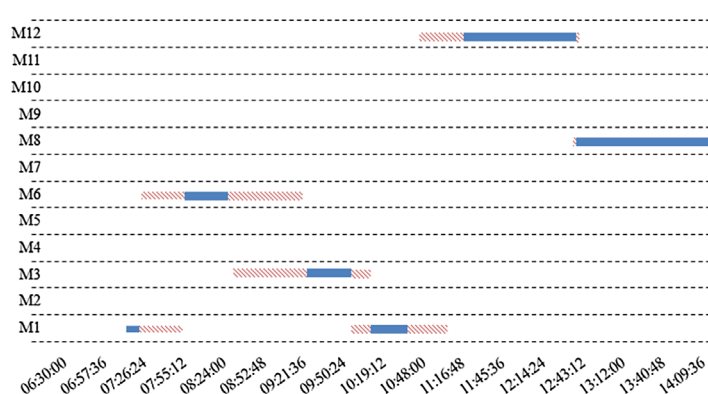
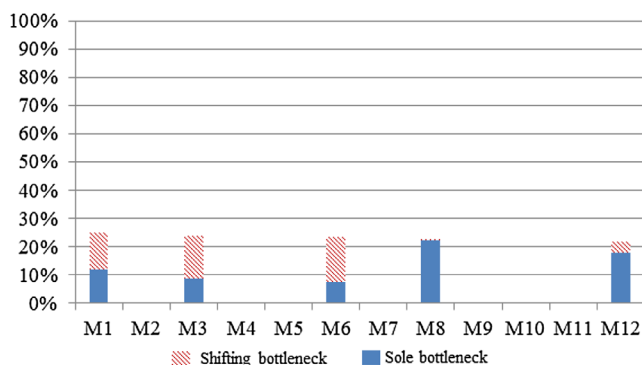


Figure 19. Average sole and shifting bottlenecks till 14:30:00.



Modelling: In this step the machine states are classified into active and not-active. The active states of the machine includes *Producing, Part changing, Error, Comlink down, Comlink up and Empty run*. The inactive states of the machine include *Not active and Waiting*. The time interval is distributed for every second and the machines active and inactive states are then mapped with respect to the time instants by 1 and 0 respectively for all machines. This forms the matrix A. The developed data driven algorithm is then applied at the time instant of interest over the MES data sets of different machines. And the sole and the shifting bottleneck periods are then calculated for the time interval.

Evaluation: The potential bottlenecks and the current bottleneck are found by interpreting the results. The information on the various types of bottleneck is then used as a decision support system to frame strategies to mitigate the bottleneck.

6.1.2. Application examples

The algorithm is tested at three different time instants during the production run on one particular day. The three examples referred to as Example 1, 2 and 3 and are described here along with the results. In Example 1, the bottleneck is found at 09:30:00 i.e. 180 min after the start of production. In Example 2, the bottleneck is found at 12:30:00, 360 min from the start time instant of production. In Example 3, the bottleneck is found at 14:30:00, 480 min from the start time instant of production. The product starts at 06:30:00 and the ends at 14:30:00. The three examples are taken to verify the scalability of the algorithm. The data preparation (which includes collection of MES data from all machines and data cleaning) and data modelling step (classification of active and not active states) remains the same for all the three examples. However, the evaluation step is different for each example as it involves the interpretation of the results obtained from the application of the algorithm.

Table 6. Average sole and shifting bottleneck empirics

Machine	Example 1			Example 2			Example 3		
	Sole (%)	Shifting (%)	Sum (%)	Sole (%)	Shifting (%)	Sum (%)	Sole (%)	Shifting (%)	Sum (%)
M1	39.40	37.31	77.71	15.62	17.35	33.15	11.72	13.15	24.86
M2	1.09	37.31	38.40	–	–	–	–	–	–
M3	–	–	–	11.68	20.02	31.70	8.76	15.02	23.78
M4	–	–	–	–	–	–	–	–	–
M5	–	–	–	–	–	–	–	–	–
M6	–	–	–	9.62	21.41	31.03	7.22	16.06	23.28
M7	–	–	–	–	–	–	–	–	–
M8	–	–	–	–	–	–	22.25	0.18	22.43
M9	–	–	–	–	–	–	–	–	–
M10	–	–	–	–	–	–	–	–	–
M11	–	–	–	–	–	–	–	–	–
M12	–	–	–	20.28	4.85	25.13	17.78	3.82	21.60

Example 1 At time instant 09:30:00

The shifting bottlenecks using the data-driven algorithm are determined during the production run at time instant 09:30:00. The shifting and the sole patterns of the machines across the entire duration are shown in Figure 14 and the empirics are shown in Table 6. The results are also shown in the graphical form in Figure 15.

Evaluation: From Figure 14, it can be seen that the current bottleneck at 09:30:00 is machine M1. On the other hand, it can be seen from Figure 15 that the machines M1 and M2 are the only machines that were limiting the system throughput during this time interval and hence these are average bottlenecks, with M1 having the largest effect. The remaining machines are non-bottlenecks.

Example 2 At time instant 12:30:00

The algorithm is applied at time 12:30:00 and the shifting and the sole patterns across the machines are observed from the start time of production and 12:30:00 as shown in Figure 16. The average sole and shifting bottlenecks are shown in graphical form in Figure 17 and the empirics are presented in Table 6.

Evaluation: From Figure 16, it can be seen that the current bottleneck at 12:30:00 is machine M12. Also, it can be seen from Figure 17 that the machines M1, M3, M6 and M12 were responsible for limiting the throughput of the production system and hence those are the average bottlenecks. However, the main average bottleneck out of these four machines is M1. The non-bottlenecks are M2, M4, M5, M7, M8 M9, M10, M11.

Example 3 At time instant 14:30:00

The algorithm is applied at the time instant 14:30:00, which is the end time instant of the production run during that day, and the shifting and the sole patterns across the machines are observed from the start time instant of production and 14:30:00. These are represented in Figure 18. The average sole and shifting bottlenecks on that day from the start time instant of the production run to the end time instant of production run are presented in Table 6 and are shown in graphical form in Figure 19.

Figure 20. Layout of the production line.

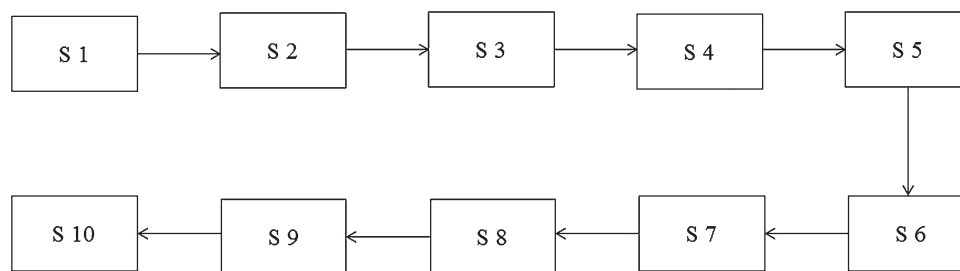


Table 7. Example MES record of one station including the station status and time stamps

Location	Start time	End time	State of the machine
1	01-09-2015 16:41:07	01-09-2014 16:43:36	Error
1	01-09-2015 16:58:20	01-09-2014 17:01:43	Error
1	01-09-2015 18:40:45	01-09-2015 18:41:20	Error
1	01-09-2015 21:17:07	01-09-2014 21:19:00	Error
1	01-09-2015 21:28:46	01-09-2014 21:30:43	Error
1	01-09-2015 21:54:05	01-09-2015 21:55:25	Error

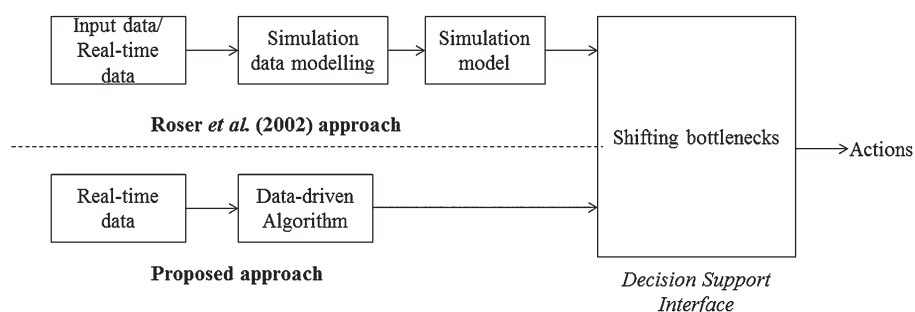
Evaluation: From Figure 18 it can be seen that the current bottleneck at 14:30:00 is machine M8. On the other hand, it can be seen from Figure 19 that machines M1, M3, M6, M8 and M12 were averagely responsible in limiting the output during the time interval, and hence those are average bottlenecks. However, the main average bottleneck machine out of these four machines is M1. The non-bottleneck machines are M4, M5, M9, M10, and M11.

6.2. Test study 2

The second study was of a serial production line from automotive industry. The production line development team of this production line is looking for quicker methods to identify the bottlenecks on a real-time basis using MES data. The main purpose of identifying the bottlenecks is to evaluate the possible buffer locations and prioritise the production improvement projects. Figure 20 shows the serial production line layout. The production line starts with station S1 and ends at Station S10. Each station is connected to the MES. The MES collects the data on the downtimes of the different stations along with the time stamps. An example MES data structure of station 1 is shown in Table 7. The production shift starts 16:30:00 and ends at 22:30:00.

The *Error* state of the machine is that state when the machine is undergoing maintenance. As can be seen from Table 7, the MES does not capture all of the machine state information. The MES captures only the Error information of the machine, leaving out the other active states of the machine such as producing, part changing etc. Hence the machine states are not captured on a continuous time scale and as a result the active duration of the machine cannot be computed from the given

Figure 21. Differences between the data-driven algorithm and the simulation based approach for shifting bottleneck detection.



MES. Therefore, the shifting bottleneck detection cannot be performed over this MES data structure. But, if the other states of the machines are also captured by the MES in the same format as shown in Table 7, then the developed data-driven algorithm for the shifting bottleneck detection approach could be directly be applied over the MES data sets without any modifications to the algorithms.

7. Discussion

This paper is aimed at developing a data-driven algorithm for an existing simulation based shifting bottleneck detection technique as proposed by Roser et al. (2002a). In this paper, the current bottleneck, average bottleneck and non-bottlenecks as proposed by Roser et al. (2002a) were achieved directly from MES data and without using a simulation model of the production system. The developed algorithm was tested and verified over the MES data sets of two different real-world production lines. The algorithm requires only the different states of the machine and the time stamps of those states. The algorithm was validated by evaluating the results with the requirements of production and maintenance engineers in identifying the bottleneck machines from the production system perspective at any given time instant during the production run.

The developed data-driven shifting bottleneck algorithm has many advantages over simulation based shifting bottleneck detection. Firstly, it uses easily obtained real-time production line data as captured by the MES. Secondly, it processes the latest data feed available and indicates the real-time bottleneck at any time instant during the production run. Thirdly, it does not use a model of the production system and therefore no approximations of the inputs are made as in simulation (Fowler & Rose, 2004) and no assumptions are made as in simulation when building the model (Skoogh, 2011). Also, the algorithm eliminates the steps of condensing the raw input data into statistical distributions for the input parameters for the simulation model, as is required in a simulation environment (Leemis, 2004; Skoogh, 2011) as shown in Figure 21. Moreover, the use of the data-driven algorithm eliminates the frequent calculations required in simulation to obtain such parameters as processing times, setup-times etc. which makes the algorithm more suitable for use on a real-time basis to obtain accurate insights on bottlenecks at any given time. Additionally, it takes a lot of effort and time to update the simulation model of the production system with improvements and developments made in the real production (Fowler & Rose, 2004). For example, change in layout, in which case the old simulation model cannot be used and the model needs to be re-built again, taking a lot of time and effort, whereas the algorithm can still be applied without any modifications made to the algorithm over the recent MES data collected from the machines to detect the bottlenecks. Fourthly, it can be easily integrated with the MES for decision support interface as shown in Figure 21. Fifthly, the algorithm can be used to reveal the bottlenecks in production systems with parallel machines, with buffers in-between the machines as proved in the test studies. It is also proved that this algorithm not only gives information on current and average bottlenecks, but also provides information on non-bottlenecks, whereas the data-driven turning point method, which is also a data-driven method proposed by Li, Chang, et al. (2009) gives less information on the non-bottlenecks of the production system. Moreover, the accuracy of the turning point method is less when there are small stoppages in the production line (for example, when the stoppage time is less than the machine cycle time) due to the buffers, whereas the active period method of shifting bottlenecks also takes this into consideration when detecting the bottlenecks. Lastly, the algorithm facilitates interaction with production personnel in terms of the type and the number of machines needed to be analysed, thus making the algorithm more flexible to the end-user, which is one of the criteria as mentioned by Davenport et al. (2012).

The information on bottlenecks from the application of the data-driven algorithm is extremely helpful to production and maintenance engineers when allocating resources on a real-time basis for bottleneck machines and avoids spending time on improving the non-bottleneck machines. For example, prioritising the maintenance activities on the bottleneck machines will reduce the waiting time for repairs (Gopalakrishnan et al., 2014; Langer et al., 2010), while buffers could be increased both upstream and downstream of the bottleneck machines to avoid blockage and starvation of the machines (Lawrence & Buss, 1994). On the other hand, the performance of the non-bottleneck

machines can be reduced in order to match the overall production rate, thus improving the balance losses in the production system. Information on average bottlenecks of the previous production shift or day can be used at the traditional planning meetings which take place before the next production shift or day to frame strategies to mitigate the bottleneck. This can help effectively utilise finite maintenance and production resources by allocating them to the bottleneck machines (Li, Ambani, et al., 2009; Li & Ni, 2009). Furthermore, the algorithm can be used as an integrated part of the Computerised CMMS or PMDSS (Li, Ambani, et al., 2009; Li & Ni, 2009) in order to give better preventive maintenance to bottlenecks and to prioritize reactive repairs to the momentary bottlenecks of the system in order to increase the throughput (Gopalakrishnan et al., 2014). These types of integration can enhance decision-making by individuals (Santana, 1995) and help them to frame effective strategies to mitigate the bottleneck.

Considering the growth foreseen in the availability of the real-time information in production systems as indicated by Selbst et al. (2011), the developed data-driven algorithm for the shifting bottleneck detection method can be used to aid effective decision-making by exposing the relevant information on bottleneck. The companies, when using the voluminous real-time data from the MES, can understand the production constraints at a more granular level, thus improving the visibility and reducing the uncertainty in detecting the bottlenecks at the operational level. Thereby, the resources can be allocated to the bottleneck machines on a real-time basis and improvement activities can be prioritised to improve the productivity and robustness of the production system.

8. Future work

The algorithm presented in this paper provides descriptive information on bottlenecks using historical data. Future work would consist of testing the algorithm over other manufacturing companies' MES data sources in order to investigate the generalisability of the algorithm and to explore the limitations. After descriptive algorithms, the natural way forward is to develop predictive and prescriptive algorithms to predict the bottlenecks by using machine performance metrics and other product related data and to prescribe necessary mitigating actions to improve the throughput.

9. Conclusion

In this paper, the data-driven shifting bottleneck detection algorithm was developed, verified and validated. The algorithm can detect the current bottleneck, average bottlenecks and the non-bottlenecks by using the data obtained directly from MES systems. This work addresses the problem of locating the right bottleneck in quicker time and suggests a practical data-driven approach to accomplish that end. The algorithm is tested on two automotive manufacturing production lines to identify the benefits and the prerequisites of the algorithm. It requires only the different machine state information and the time stamps for input. The algorithm can be used over big data sets of log files of different machines as collected by MES and identifies the different bottlenecks in the production system. The paper shows that it can be helpful for production and maintenance engineers in order to allocate resources such as buffers and manpower, and to prioritize such improvement activities as cycle time reduction, setup time reduction and preventive and reactive maintenance on bottleneck machines. The application of this algorithm is expected to result in increased robustness and efficiency of production systems.

Acknowledgements

The research project "Streamlined Modelling for Decision Support for Fact-based Production Development (StreaMod)" is funded by VINNOVA [grant number 2013-04726], Swedish Agency for Innovative Systems. The authors would also like to thank all members of the StreaMod project included in the study. This work has been performed within Sustainable Production Initiative and the Production Area of Advance at Chalmers. The support is greatly appreciated.

Funding

This work was supported by VINNOVA [grant number 2013-04726].

Author details

Mukund Subramaniyan¹
E-mail: mukunds@chalmers.se
ORCID ID: <http://orcid.org/0000-0002-2787-7262>
Anders Skoogh¹
E-mail: anders.skoogh@chalmers.se
ORCID ID: <http://orcid.org/0000-0001-8519-0736>
Maheshwaran Gopalakrishnan¹
E-mail: mahgop@chalmers.se
ORCID ID: <http://orcid.org/0000-0001-5102-6559>
Hans Salomonsson²
E-mail: hans.salomonsson@chalmers.se

ORCID ID: <http://orcid.org/0000-0002-9615-7410>

Atieh Hanna³

E-mail: atieh.hanna@volvo.com

Dan Lämku⁴

E-mail: dan.lamku@volvocars.com

¹ Department of Product and Production Technology, Chalmers University of Technology, Gothenburg 41296, Sweden.

² Department of Computer Science and Engineering, Chalmers University of Technology, Gothenburg 41296, Sweden.

³ Manufacturing Research and Advanced Engineering, Volvo Group Trucks Operations, Gothenburg 40508, Sweden.

⁴ Virtual Methods and IT, Volvo Car Corporation, Gothenburg 40531, Sweden.

Citation information

Cite this article as: An algorithm for data-driven shifting bottleneck detection, Mukund Subramaniyan, Anders Skoogh, Maheshwaran Gopalakrishnan, Hans Salomonsson, Atieh Hanna & Dan Lämku, *Cogent Engineering* (2016), 3: 1239516.

References

- Bean, R., & Kiron, D. (2013). Organizational alignment is key to big data success. *MIT Sloan Management Review*, 54.
- Betterton, C. E., & Silver, S. J. (2012). Detecting bottlenecks in serial production lines—a focus on interdeparture time variance. *International Journal of Production Research*, 50, 4158–4174. doi:10.1080/00207543.2011.596847
- Chang, Q., Ni, J., Bandyopadhyay, P., Biller, S., & Xiao, G. (2007). Supervisory factory control based on real-time production feedback. *Journal of Manufacturing Science and Engineering*, 129, 653. doi:10.1115/1.2673666
- Davenport, T. H., Barth, P., & Bean, R. (2012). How “big data” is different. *MIT Sloan Management Review*, 1, 43–46.
- Faget, P., Eriksson, U., & Herrmann, F. (2005). Applying discrete event simulation and an automated bottleneck analysis as an aid to detect running production constraints. In M. E. Kuhl, N. M. Steiger, F. B. Armstrong, & J. A. Joines (Eds.), *Proceedings of the 2005 Winter Simulation Conference* (pp. 1401–1407). Orlando, FL: IEEE. <http://dx.doi.org/10.1109/WSC.2005.1574404>
- Fowler, J. W., & Rose, O. (2004). Grand challenges in modeling and simulation of complex manufacturing systems. *Simulation*, 80, 469–476. doi:10.1177/0037549704044324
- Goldrat, E. M., & Cox, J. (1992). *The goal: A process of ongoing improvement*. Great Barrington, MA: North River Press.
- Gopalakrishnan, M., Skoogh, A., & Christoph, L. (2014). Simulation based planning of maintenance activities by a shifting priority method. In A. Tolk, S. Diallo, I. Ryzhov, L. Yilmaz, S. Buckley, & J. A. Miller (Eds.), *Proceedings of the 2014 Winter Simulation Conference* (pp. 2600–2608). Piscataway, NJ: IEEE
- Hopp, W., & Spearman, M. L. (2000). *Factory physics: Foundations of manufacturing management*. New York: McGraw Hill.
- Langer, R., Li, J., Biller, S., Chang, Q., Huang, N., & Xiao, G. (2010). Simulation study of a bottleneck-based dispatching policy for a maintenance workforce. *International Journal of Production Research*, 48, 1745–1763. doi:10.1080/0020754080255576
- Lawrence, R. S., & Buss, A. H. (1994). Shifting production bottlenecks: Causes, cures and conundrums. *Production and Operations Management*, 3, 21–37.
- Leemis, L. M. (2004). Building credible input models. In R. G. Ingalls, M. D. Rossetti, J. S. Smith, & B. A. Peters (Eds.), *Proceedings of the 2004 Winter Simulation Conference* (pp. 29–40). Piscataway, NJ: Institute of Electrical and Electronics Engineers.
- Lengyel, L. (2015). Validating rule-based algorithms. *Acta Polytechnica*, 12, 59–75.
- Li, L., & Ni, J. (2009). Short-term decision support system for maintenance task prioritization. *International Journal of Production Economics*, 121, 195–202. <http://dx.doi.org/10.1016/j.ijpe.2009.05.006>
- Li, L., Ambani, S., & Ni, J. (2009). Plant-level maintenance decision support system for throughput improvement. *International Journal of Production Research*, 47, 7047–7061. doi:10.1080/00207540802375705
- Li, L., Chang, Q., & Ni, J. (2009). Data driven bottleneck detection of manufacturing systems. *International Journal of Production Research*, 47, 5019–5036. doi:10.1080/00207540701881860
- Nishimura, M. (1997). Strategic inventory control for stable production. *Proceedings of 1997 IEEE International Symposium on Semiconductor Manufacturing*, A1–A4. doi:10.1109/ISSM.1997.664480
- O'Donovan, P., Leahy, K., Bruton, K., & O'Sullivan, D. T. J. (2015). Big data in manufacturing: A systematic mapping study. *Journal of Big Data*, 1, 20.
- Pete, C., Julian, C., Randy, K., Thomas, K., Thomas, R., Colin, S., & Wirth, R. (2000). *Crisp-Dm 1.0. CRISP-DM Consortium (CRISPMWP-1104)*. Chicago, IL: SPSS.
- Roser, C., Nakano, M., & Tanaka, M. (2001). A practical bottleneck detection method. In B. A. Peters, J. S. Smith, D. J. Medeiros, & M. W. Rohrer (Eds.), *Proceedings of the 2001 Winter Simulation Conference* (pp. 949–953). Washington, DC: IEEE. <http://dx.doi.org/10.1109/WSC.2001.977398>
- Roser, C., Nakano, M., & Tanaka, M. (2002a). Shifting bottleneck detection. In E. Ycesan, C. H. Chen, J. L. Snowdon, & J. M. Charnes (Eds.), *Proceedings of the 2002 Winter Simulation Conference* (pp. 1079–1086). San Diego, CA: IEEE. <http://dx.doi.org/10.1109/WSC.2002.1166360>
- Roser, C., Nakano, M., & Tanaka, M. (2002b). Tracking shifting bottlenecks. In *Japan-USA Symposium on Flexible Automation* (pp. 745–750). Hiroshima.
- Roser, C., Nakano, M., & Tanaka, M. (2003a). Constraint management in manufacturing systems. *JSME International Journal Series C*, 46, 73–80.
- Roser, C., Nakano, M., & Tanaka, M. (2003b). Comparison of bottleneck detection methods for AGV systems. In S. Chick, P. Sanchez, D. Ferrin, & D. J. Morrice (Eds.), *Proceedings of the 2003 Winter Simulation Conference* (pp. 1192–1198). New Orleans, LA: IEEE. <http://dx.doi.org/10.1109/WSC.2003.1261549>
- Santana, M. (1995). Managerial learning: A neglected dimension in decision support systems. *Proceedings of the Twenty-Eighth Hawaii International Conference on System Sciences*, 4, 82–91. doi:10.1109/HICSS.1995.375741
- Sengupta, S., Das, K., & VanTil, R. P. (2008). A new method for bottleneck detection. In S. J. Mason, R. R. Hill, L. M.önch, O. Rose, T. Jefferson, & J. W. Fowler (Eds.), *Proceedings of the 2008 Winter Simulation Conference* (pp. 1741–1745). Miami, FL: IEEE. <http://dx.doi.org/10.1109/WSC.2008.4736261>
- Skoogh, A. (2011). *Automation of input data management* (PhD Diss.). Chalmers University of Technology, Göteborg.
- Subramaniyan, M. (2015). *Production data analytics—to identify productivity potentials* (Master Thesis). Chalmers University of Technology, Göteborg.
- Zelbst, P. J., Green, Jr, K. W., Sower, V. E., & Abshire, R. D. (2011). Radio frequency identification technology utilization and organizational agility. *Journal of Computer Information Systems*, 52, 24–33.



© 2016 The Author(s). This open access article is distributed under a Creative Commons Attribution (CC-BY) 4.0 license.

You are free to:

Share — copy and redistribute the material in any medium or format

Adapt — remix, transform, and build upon the material for any purpose, even commercially.

The licensor cannot revoke these freedoms as long as you follow the license terms.

Under the following terms:

Attribution — You must give appropriate credit, provide a link to the license, and indicate if changes were made.

You may do so in any reasonable manner, but not in any way that suggests the licensor endorses you or your use.

No additional restrictions

You may not apply legal terms or technological measures that legally restrict others from doing anything the license permits.

Cogent Engineering (ISSN: 2331-1916) is published by Cogent OA, part of Taylor & Francis Group.

Publishing with Cogent OA ensures:

- Immediate, universal access to your article on publication
- High visibility and discoverability via the Cogent OA website as well as Taylor & Francis Online
- Download and citation statistics for your article
- Rapid online publication
- Input from, and dialog with, expert editors and editorial boards
- Retention of full copyright of your article
- Guaranteed legacy preservation of your article
- Discounts and waivers for authors in developing regions

Submit your manuscript to a Cogent OA journal at www.CogentOA.com

