Throughput bottleneck detection in manufacturing: a systematic review of the literature on methods and operationalization modes

Taylor & Francis
Taylor & Francis Group

Production & Manufacturing Research

An Open Access Journal

ISSN: (Print) (Online) Journal homepage: https://www.tandfonline.com/loi/tpmr20

Throughput bottleneck detection in manufacturing: a systematic review of the literature on methods and operationalization modes

Anders Skoogh, Matthias Thürer, Mukund Subramaniyan, Andrea Matta & Christoph Roser

To cite this article: Anders Skoogh, Matthias Thürer, Mukund Subramaniyan, Andrea Matta & Christoph Roser (2023) Throughput bottleneck detection in manufacturing: a systematic review of the literature on methods and operationalization modes, Production & Manufacturing Research, 11:1, 2283031, DOI: 10.1080/21693277.2023.2283031

To link to this article: https://doi.org/10.1080/21693277.2023.2283031

Published online: 28 Nov 2023.

PRODUCTION & MANUFACTURING RESEARCH
2023, VOL. 11, NO. 1, 2283031
https://doi.org/10.1080/21693277.2023.2283031

Taylor & Francis
Taylor & Francis Group

Throughput bottleneck detection in manufacturing: a systematic review of the literature on methods and operationalization modes

Anders Skoogh[a], Matthias Thürer[b], Mukund Subramaniyan[a,c], Andrea Matta[d] and Christoph Roser[e]

[a]Department of Industrial and Materials Science, Chalmers University of Technology, Gothenburg, Sweden; [b]Department of Mechanical Engineering, Chair of Factory Planning and Intralogistics, Chemnitz University of Technology, Chemnitz, Germany; [c]Insights & Data, Capgemini AB, Gothenburg, Sweden; [d]Department of Mechanical Engineering, Politecnico di Milano, Milan, Italy; [e]Department of Management Science and Engineering, Karlsruhe University of Applied Sciences, Karlsruhe, Germany

ABSTRACT

Throughput is an important parameter to evaluate production system performance. It is typically constrained by one or more resources referred to as 'throughput bottlenecks'. To start improvement actions, the first step is to identify throughput bottlenecks. Consequently, several bottleneck detection methods were developed in the literature. But this literature remains largely unstructured, which makes it difficult for practitioners to select an appropriate method. To generate clarity and to consolidate the field, a systematic literature review was conducted. The review identified 14 different bottleneck detection methods that are classified according to the information used: queue states, process states, or combined queue and process states. It further identified three different modes used to operationalize the different bottleneck detection methods: *gemba* walk, discrete event simulation, and data science. This study further presents important research issues, identifies contingency factors for method application, and discusses important guidelines for the choice of operationalization mode in practice.

ARTICLE HISTORY
Received 23 August 2023
Accepted 6 November 2023

KEYWORDS
Throughput bottlenecks; theory of constraints; production control; operations management

1. Introduction

Increasing throughput is one of the important goals of many manufacturing companies and a key objective of Lean Production and Operational Excellence (Hopp & Spearman, 2008). One way to increase throughput is to resolve the throughput bottleneck (Roser et al., 2015, Pehrsson et al., 2016, Wu et al., 2016)), which can be defined as the workstation (or resource) that has the largest impact on overall system performance. This study focuses on these throughput bottlenecks, which will also be referred to simply as bottlenecks. In order to resolve the bottleneck, one first needs to identify the bottleneck.

CONTACT Anders Skoogh anders.skoogh@chalmers.se Department of Industrial and Materials Science, Chalmers University of Technology, Gothenburg 41296, Sweden

Failure to identify the bottleneck correctly will lead to wasted improvement efforts. Quickly and accurately identifying throughput bottlenecks is specifically important in contexts where throughput bottlenecks shift, causing throughput fluctuations across production runs. For example, a throughput fluctuation between 475 and 800 production batches was reported in a real-world semiconductor manufacturing line (Wang et al., 2019). Practitioners need to correctly identify throughput bottleneck to control these fluctuations and eventually improve throughput.

In response to this practical need, scholars and practitioners have been developing different scientific methods to identify throughput bottlenecks in production systems, and demonstrated possible operationalization modes of these methods (Betterton & Silver, 2012 , Kuo et al., 1996, Li, 2018). Over the last 30 years, this has resulted in a broad range of academic literature. Meanwhile, the authors of this study observed that practitioners are increasingly showing interest in implementing the different throughput bottleneck detection methods in real-world factories. But before implementation, they need to identify different scientific methods from the vast academic literature, and carefully analyze the methods in the specific context of their factory. Practitioners are left largely alone in this endeavor, which motivated this study.

Although there have been previous literature reviews, these reviews remain rather restricted. For example, (Li et al., 2009) focuses on analytical methods that construct recursive equations based on Markovian and Bernoulli assumptions. How well these methods fit the real-world production system depends on the extent to which the Markovian and Bernoulli assumptions are satisfied, which is something that can only be empirically determined i.e. by using real-world production system data. Unfortunately, there are only few attempts to validate these assumptions and to study the impact of deviations. In contrast, a large number of throughput bottleneck detection methods were developed that directly use production system state information, which makes them more relevant for practice. Meanwhile, (Subramaniyan et al., 2021) focuses on the implementation method, reviewing how bottleneck detection methods can be implemented using artificial intelligence. Bottleneck detection methods itself are not reviewed. There exist only some convenience-based literature surveys on detection methods that use system state information. For example (Betterton & Silver, 2012; Roser & Nakano, 2015; Yu & Matta, 2016), (Yong-Cai Wang, Qian-Chuan Zhao, Wang et al., 2005) (Rocha & Lopes, 2022), and (Lima et al., 2008) identified a subset of available bottleneck detection methods to be included in their simulations. According to the authors' best of knowledge, no systematic, transparent, and thorough literature review of different bottleneck detection methods has been conducted to date.

This study systematically reviews the literature and synthesizes the current knowledge on throughput bottleneck detection methods. This provides three main contributions. First, it classifies existing throughput bottleneck detection methods into three categories based on the information about the production system it uses (information derived from queue state, process state, and system state). Second, it classifies the operationalization modes of these throughput bottleneck detection methods based on how a method can be implemented on the shop floor (*gemba* walk, discrete event simulation (DES) approach, and data science approach). Third, it provides a range of promising future research directions and practical recommendations that can support further advancement of the throughput bottleneck research field.

The remainder of this article is structured as follows. In Section 2 an overview of the fundamentals of throughput bottlenecks using an illustrative serial production line is first provided. The methodology behind the systematic literature review is then presented in Section 3. In Section 4, results of the analysis of the set of articles identified through the systematic search process are presented. In Section 5 results are discussed, areas for future research highlighted, and guidance for application in practice provided. The conclusions are summarized in Section 6.

2. Fundamentals of throughput bottlenecks

This section outlines the authors understanding of throughput bottlenecks in production systems. A production system consists of resources, such as machines, robots, resources for material transportation, and humans, and inventory buffers. All must work together to produce products from raw material. Consider an example of a serial production line with two shifting bottlenecks that has five machines (M1, M2, . . . , M5) and four limited input/output buffers (B1, B2, . . . B4), as shown in Figure 1.

In the real world, every machine in the production system exhibits stochastic behavior that is caused by stochastic random events, such as breakdown, variations in the processing times, or variations in the setup times (Wu et al., 2016). These events itself have different time durations. As the processes are connected in the production system (Goldrat & Cox, 1990), every time a random event occurs on a machine, its effects may be propagated to the upstream and downstream machines in the form of blockage and starvation. For example, an undesirable event on M3 May cause M1 and M2 to be blocked from delivering the products to M3. Similarly, M4 and M5 May become starved. This affects the dynamics of the whole production system (Li et al., 2011). As a result, individual machine throughput fluctuates over time, causing the final throughput from the production system to fluctuate. Eliminating throughput bottlenecks reduces these fluctuations and increases the throughput of the production system. This is also called the 'law of bottlenecks' (Schmenner & Swink, 1998)(p.101).

(Goldrat & Cox, 1990) further argues that stochastic effects of a set of machines in the production system create larger fluctuations in the system throughput than other machines. These sets of machines are called 'throughput bottlenecks' (Goldrat & Cox, 1990). In general, there will be less inventory downstream of a bottleneck than upstream of a bottleneck. But there will be no increase in upstream inventory if buffers are not limited and thus blocking introduced. With limited buffers upstream blocking may occur leading to a loss in production rate at upstream machines and consequently increased

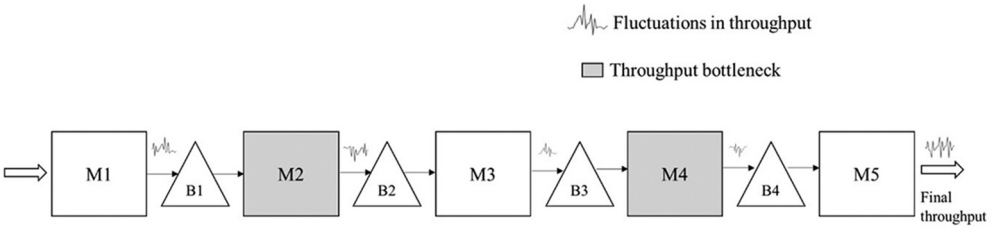

Figure 1. Illustration of throughput bottlenecks in a serial production line.

inventory build-up. Downstream of the bottleneck starvation will always occur, leading to a reduction in inventory.

Since there are fluctuations, i.e. changes over time, (Roser et al., 2002a) and (Subramaniyan et al., 2016) further distinguished between three types of throughput bottleneck: momentary throughput bottleneck, average throughput bottleneck, and shifting throughput bottleneck. These types of throughput bottlenecks are based on the time period of reference. Machines that are throughput bottlenecks at a specific time instance are called momentary throughput bottlenecks. Machines that are throughput bottlenecks for a major time interval are called average throughput bottlenecks. Meanwhile, time intervals where the momentary throughput bottleneck is shifting from one machine to another machine are called shifting time periods, and both machines are called shifting throughput bottlenecks. Shifting bottlenecks are mainly due to the inherent variability in the duration of stochastic random events and the actions taken by practitioners to resolve the throughput bottleneck. Both change the production system dynamics giving rise to new throughput bottlenecks in the production system (Li et al., 2011). To a certain degree every station may become the momentary bottleneck. But managers are very often interested in the average bottleneck, which can be defined as the station that has the most significant impact on the throughput during a time period (Roser et al., 2015).

The identification and elimination of throughput bottlenecks is a continuous process by practitioners that needs to be pursued until the required throughput levels are reached. But to resolve throughput bottlenecks, one needs appropriate methods to identify throughput bottlenecks quickly and reliable, specifically in dynamic systems and when bottlenecks shift. But there is no universal definition of a bottleneck. Authors use different definitions, and this leads to different measures and methods. In response, this article provides a review of state-based bottleneck detection methods and their operationalization modes.

3. Methodology – systematic review of the literature

A systematic review of the literature (following (Tranfield et al., 2003)) has been conducted. The three subsections below outline the approach adopted for sourcing, screening, and analyzing of the articles. The screening process of the articles is also summarized in Figure 2.

3.1. Sourcing the articles

To ensure the sample is representative, and to avoid any partiality of the authors, a systematic sampling procedure was applied. There are, arguably, three major abstract and citation databases: Google Scholar, Scopus, and the Web of Science. Google Scholar was excluded because of its low data quality, which raises questions about its suitability for research. Meanwhile, Scopus has a broader coverage than the Web of Science. In general, the number of journals in the Web of Science not covered by Scopus is about 5% and the number of Scopus articles not covered by the Web of Science is about 50%. Scopus was therefore chosen. Scopus was queried in October 2022 using the terms: 'bottleneck detection', 'bottleneck identification', 'bottleneck diagnosis', 'bottleneck

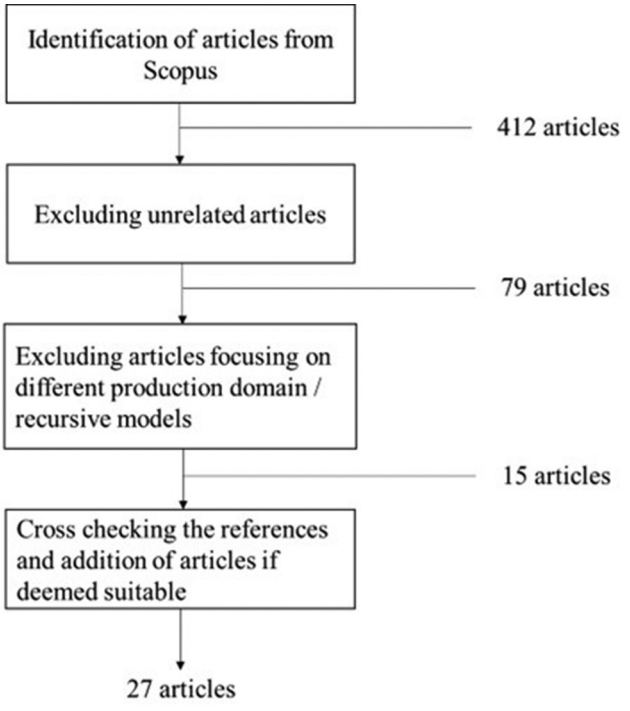

Figure 2. Systematic sampling process.

prognosis', and 'bottleneck prediction'. The term 'bottleneck' was not used as a search term since this resulted in an unmanageable number of results. To still capture a broad range of articles, the search was broadened by considering terms related to detection, such as identification, diagnosis, prognosis, and prediction. The title, abstract, and keywords of articles were searched, whilst the document type was limited to 'articles' and 'articles in press' to ensure the quality of the sources held in the database. Only peer-reviewed articles were considered. Note that it is recognized that there is also literature in form of books and white papers; however, it is assumed that relevant methods that are presented in books are also published as an article. The search was further restricted to articles from the Engineering, Decision Sciences, and Business Management fields, and articles in English. There was no restriction on the year of publication. For this search, a total of 412 articles was retrieved.

3.2. Screening the articles

The original sample of 412 articles was reduced to 79 articles by excluding unrelated articles, for example, related to traffic congestions, computer science, or genetics. After reading these articles, a further 64 articles were excluded because of the following reasons. First, articles did not focus on the manufacturing domain, which is the focus of this study. Second, articles were concerned with recursive mathematical models (an extension of the work reported in (Li et al., 2009)), did not propose new throughput bottleneck detection methods or operationalization modes, or used optimization

Table 1. Distribution of publications across journals/conferences.

Journal/Conference Name	Count
Proceeding of the Winter Simulation Conference	5
International Journal of Production Research	4
Journal of Manufacturing Systems	2
Computers and Industrial Engineering	2
Production and Operations Management	2
Journal of Manufacturing Systems and Engineering	2
Other journals (8 different journals)	8
Other conferences (2 different conferences)	2

techniques to optimize the performance of bottlenecks after identification. Third, duplicate publications were produced as conference and journal articles. To ensure that relevant articles were not missed, the references in the 15 remaining articles were cross-checked. From this process, 12 additional relevant articles were retrieved. Most were conference articles, not included in the initial search to ensure the quality of the database. Since these articles were highly cited and the conferences subject to a peer review process, they were included in the review. This approach of supplementing the set of articles that had been mechanically retrieved helped to ensure that the list of articles was complete. The final sample of analyzed full papers was thus 27 articles. Table 1 presents the distribution of journals and conferences where the 27 articles of the sample were published. Finally, note that only the articles that are referred to directly in this study are listed in the references at the end of this article, but a full reference list is available from the corresponding author upon request.

3.3. Analyzing the articles

This stage involved extracting and documenting information from each of the 27 articles. To minimize subjectivity, the authors: (i) cross-checked results; and, (ii) conducted regular meetings to resolve any emerging inconsistencies in interpreting the results. The major research vehicle was content analysis. As a template for data collection, a simple matrix was used where for each article (row) two questions (column) were asked: What bottleneck detection methods are used? How are they operationalized? Results from this analysis process will be presented next.

4. Results

Bottleneck detection methods that emerged from the systematic literature review will first be introduced. The focus then shifts on how these methods were operationalized.

4.1. Bottleneck detection methods

Out of the sample, 14 articles presented new bottleneck detection methods. The remaining articles mainly focused on new operationalization procedures for existing methods or compared different methods. One way to classify bottleneck detection methods is according to the bottleneck definition adopted by an author. For most studies bottleneck

definition and measure used to detect the bottleneck overlap. This is typically the case if the bottleneck is simply defined in terms of a measure, such as utilization, work in queue, or the time a station is active. While this is practical, other studies provide rigorous mathematical definitions of bottlenecks. Since it is difficult to directly apply these definitions, indirect methods based on measurable data are then introduced, and it is shown how these methods approach the mathematical definition. This study is motivated by a practical need. The classification is therefore focused on the type of measurable data used. The 14 different bottleneck detection methods are classified into three distinct categories: methods that focus on the queue state, methods that focus on the process state, and methods that focus on both queue and process state (i.e. the system state). These categories are based on real-world practice, where it is common for practitioners to identify throughput bottlenecks by observing queues, the processes at stations, or by combining queue and process-based observations. Each category will be discussed next before a critical discussion of the different categories is presented.

4.1.1. Bottleneck detection methods focusing on the queue state

There are two types of throughput bottleneck detection methods proposed in the literature that use queue information: (1) queue length method, and (2) waiting time method.

The *Queue Length Method* measures the queue lengths $Q_{s,t}$ of each station s, with the momentary bottleneck being the station with the maximum queue length at a given time instant, that is $max(Q_1, Q_2, \ldots, Q_n)$ at that time instant, with n stations (Lawrence & Buss, 1994), assuming that the queue limit is never reached. The average throughput bottleneck would be the station that on average has the maximum queue length for a given time period.

The *Waiting Time Method* measures the waiting time (measured in time units) of jobs in a station queue (Roser et al., 2001). This method can be used to identify average throughput bottlenecks. The station with the maximum average waiting time is the throughput bottleneck station, assuming that the queue limit is never reached.

While the above two methods appear similar, they are different. The queue length is an aggregate measure whereas the waiting time is associated with individual flow items.

4.1.2. Bottleneck detection methods focusing on the process state

This category includes detection methods that focus on the information about the process states to identify the throughput bottlenecks. A state represents an activity performed on a station e.g. producing, repair, setup, etc. A timeline representing process states is shown in Figure 3.

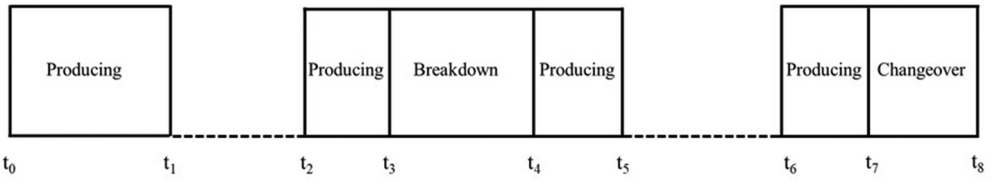

Figure 3. Process states of the station during a production run (adapted from (Roser et al., 2001)).

Nine different throughput bottleneck detection methods have been proposed in the literature that uses process state information: (1) utilization method, (2) active period percentage method, (3) average active period method, (4) longest current active period method, (5) shifting bottleneck method, (6) inactive period method, (7) inter-departure time variance method; (8) arrow method, and (9) turning point method. Each of these methods is described below.

The *Utilization Method* measures the utilization U_s, with the bottleneck being the station with the largest utilization, that is $max(U_1, U_2, \ldots, U_n)$, with n stations (Hopp et al., 2007). The utilization method includes the producing and breakdown states of a station.

The *Active Period Percentage Method* presented by (Roser et al., 2001) distinguishes between a station being active, i.e. station states when the station is not waiting for products from another station (e.g. producing, down, or setup, as illustrated in Figure 3 above), and a station being inactive, i.e. station states when the station is waiting for jobs from another station (e.g. blockage, starvation, or waiting). To determine the bottleneck the active period percentage is calculated as the time a station is active during a given time period divided by the time period. The station with the largest active period percentage is identified as the throughput bottleneck.

The nomenclature of the active period percentage and utilization method are not standardized in the literature. For example (Roser et al., 2002a), uses the term utilization percentage to refer to the active period percentage method. But referring to the active period percentage as utilization or workload percentage can be misleading. Utilization percentages follow the busy concept of a station as defined by (Law & Kelton, 1991). They only include a subset of the active states, mainly producing and down. But reflecting on the original definition of the active state by (Roser et al., 2001), the active state should include all activities towards increasing the production system throughput, such as repair, service and set up activities. (Lima et al., 2008) includes this aspect and defines the utilization percentage of a machine as the percentage of time the machine is working, and the active period percentage as the percentage of time the machine is active.

The *Average Active Period Method* also uses the active state information of a station. But rather than calculating the percentage, it calculates the average length of the active durations for each station in the production system. The station with the longest average active period is considered to be the average bottleneck, as this station is the least likely to be interrupted by other processes and thus dictates the overall system output (Roser et al., 2001). The active period method records the active periods a_s for the analyzed time horizon for each station s, that is $\{a_{s,1}, a_{s,2}, \ldots a_{s,t}\}$ being $a_{s,i}$ the duration of the active period i of station s. It then calculates the average duration for a station s, being the station with the highest average active duration the throughput bottleneck.

The *Longest Current Active Period Method* is a method that specifically focuses on momentary bottlenecks (Roser et al., 2002a, Subramaniyan et al., 2016). At any given time, the process with the longest active period is the momentary bottleneck for this method.

The *Shifting Bottleneck Detection Method* calculates the time periods in which a station is a momentary or a shifting bottleneck using the active period method. The momentary bottleneck is the station with the longest uninterrupted active duration. Shifting

bottlenecks occur when there is a large overlap between the two longest active periods of stations. There is no clear definition on how overlap is defined, and the method relies on parametrization. The throughput bottleneck is then identified as the station that was the longest time momentary of shifting bottleneck (Roser et al., 2002b). also presented a method that seeks to link process state (working, repair, etc.) to bottleneck occurrence, thereby enhancing this method.

Above introduced methods that use the different active states (such as producing, downtime, etc.) of the stations to identify throughput bottlenecks. The next set of methods focuses on the inactive states such as blocking and starvation states.

The *Inactive Period Method* identifies the bottleneck as the station with the minimum of the sum of the blocked state time and the starved state tim (Sengupta et al., 2008) e, that is station b is the bottleneck if $TB_b + TS_b < TB_{b-1} + TS_{b-1}$ $(1 < b < n)$ and $TB_b + TS_b < TB_{b+1} + TS_{b+1}$ $(1 < b < n)$, with TB_s being the blockage time for the station s over a time period, and TS_s being the starvation time for station s over a time period. It is further assumed that b is a natural value. Note that (Sengupta et al., 2008)use the inter-departure time as measure to operationalize the procedure, which is also the focus of the next method.

The *Inter-departure Time Variance Method* identifies the station with the smallest work-in-process inter-departure time variance as the bottleneck (Betterton & Silver, 2012). Since the bottleneck is argued to have a higher active time than other stations, it will cause upstream stations to be blocked and downstream stations to be starved. The increased blocking and starving at non-bottleneck stations will cause their inter-departure time variance to be larger, and the lower blocking and starving at the bottleneck will cause its inter-departure time variance to be smaller (Betterton & Silver, 2012).

In above studies the definition of the bottleneck and the measure/method used to identify the bottleneck overlap. Bottlenecks are defined in terms of utilization and active/inactive periods of stations. A mathematical rigorous approach for defining bottlenecks in serial production lines was adopted by (Kuo et al., 1996, Chiang et al., 2000), who used a partial derivative notation to reflect dependence on two or more independent variables, and (Li et al., 2009) (Li, 2009), who used the delta notation that is mostly used in limit expressions to approach zero and evaluate the slope. According to definition, authors outline different detection methods based on measurable data that result in identifying the same bottleneck as identified by the mathematical definition.

The *Arrow Method* draws arrows pointing left or right to show which stations have a higher frequency of being blocked and starved compared to adjacent stations (Kuo et al., 1996, Chiang et al., 2000). (Kuo et al., 1996) calculated the frequency in terms of time slots meaning that the arrow method measures the duration of starvation and blockage in a time period. If the frequency of blocking of station s is greater than the frequency of starvation for station $s + 1$, then the bottleneck is downstream of station s. If the frequency of starvation of station s is greater than the frequency of blocking for station s-1, then the bottleneck is upstream of station s. This is indicated by arrows, with the bottleneck being the station that has arrows pointing towards it from both sides.

The *Turning Point Method* proposed in (Li et al., 2009) and (Li, 2009) identifies the bottleneck station as the station that experiences a change (turning point), i.e. a scenario where the blockage is higher than starvation should change to a scenario where starvation is higher than the blockage. So, station b is the turning point during a time period if:

$$(TB_i - TS_i) > 0 : i \in [1, \ldots, b-1], b \neq 1, b \neq n$$

$$(TB_i - TS_i) < 0 : i \in [b+1, \ldots, n], b \neq 1, b \neq n$$

$$TB_b + TS_b < TB_{b-1} + TS_{b-1}, b \neq 1, b \neq n$$

$$TB_b + TS_b < TB_{b+1} + TS_{b+1}, b \neq 1, b \neq n$$

In addition:

b=1: $(TB_1 - TS_1) > 0$ and $(TB_2 - TS_2) < 0$ and $TB_1 + TS_1 < TB_2 + TS_2$
b=n: $(TB_{n-1} - TS_{n-1}) > 0$ and $(TB_n - TS_n) < 0$ and $TB_{n-1} + TS_{n-1} > TB_n + TS_n$

For the special case that no turning point can be found, i.e. where each station's starvation is higher than its blockage, the first station is considered to be the bottleneck; else if the station's blockage is higher than its starvation, the last station is the bottleneck (Li et al., 2009) (Li, 2018) later extended this method by proposing that, for a typical serial production line with n stations and n-1 buffers, station b is the bottleneck during a time period if the following relations hold:

For the general case $1 < b < n$:

$TB_i - TS_i > 0$ for $i < b$ and $TB_i - TS_i < 0$ for $i > b$;
$TUP_b - TB_b < TUP_{b-1} - TS_{b-1}$ or $TUP_b - TS_b < TUP_{b+1} - TB_{b+1}$;
For the special case $b = 1$:
$TB_1 - TS_1 > 0$ and $TB_2 - TS_2 < 0$ and $TUP_1 < TUP_2 - TB_2$;
For the special case $b = n$:
$TB_{n-1} - TS_{n-1} > 0$ and $TB_n - TS_n < 0$ and $TUP_{n-1} - TS_{n-1} > TUP_n$;

Where TUP_s, TB_s and TS_s are the up, blockage, and starvation time, respectively, for station s over the analyzed time horizon.

4.1.3. Bottleneck detection methods focusing on the system state

Three throughput bottleneck detection methods proposed in the literature were found that use a combination of queue state and process state information. These methods were categorized as focusing on the system state. They are (1) bottleneck walk method, (2) bottleneck index method, and (3) sensitivity of the system method.

The *Bottleneck Walk Method* is similar to the arrow method described above but adds information on the queue state (Roser et al., 2015). It provides a different means to identify the same bottleneck as identified by the mathematical definition. For the process states (waiting, starved, and blocked), the following three rules are applied: (i) whenever the process is waiting, it cannot be the bottleneck; (ii) if a process is waiting for parts (starved), then the bottleneck must be upstream; and, (iii) if a process is waiting for transport (blocked), then the bottleneck must be downstream. For all other process states (working, breakdown, set-up, maintenance, scheduled break, etc.) the station may be the bottleneck and the queue state needs to be consulted.

There are three rules for the queue states: (i) if the buffer between two processes is full or rather full, the bottleneck is probably downstream (where the parts go to); (ii) if the buffer is empty or rather empty, the bottleneck is probably upstream (where the parts came from); and if the buffer is neither rather full nor rather empty but somewhere in the middle the bottleneck direction is unknown. Probably means that it is not certain. To reliably find the momentary bottleneck, one would have to take the first derivative of inventories. i.e. it is not important if the buffer is large or small, but rather if it is getting larger or smaller. However, this is difficult to observe reliably, and in practice above assumptions works well according to (Roser et al., 2015). Arrows are used to indicate the direction of the bottleneck, with the momentary bottleneck being the station that has arrows pointing towards it from both sides. To indicate the average bottleneck, the bottleneck walk needs to be replicated in periodic time intervals. In this sense the bottleneck walk is different from the arrow method, which focuses on time durations and thus average bottlenecks.

The *Bottleneck Index Method* uses the utilization of a station and the number of jobs in the buffer preceding the station to calculate the bottleneck index (Huang et al., 2019). This calculation is performed for every station. The station with the highest bottleneck index is identified as the throughput bottleneck.

The *Sensitivity of The System Method* uses the ratio of change in the system state to determine the bottleneck (Kuo et al., 1996, Chang et al., 2007). The ratio is defined by the production rate p_s and queue Q_s of each station in the system divided by the change in the production rate of each station. A station i is a bottleneck if it has the system's largest ratio, that is:

$$\frac{\partial P_R(p_1, p_2, \ldots, p_n, Q_1, Q_2, \ldots, Q_{n-1})}{\partial p_i} > \frac{\partial P_R(p_1, p_2, \ldots, p_n, Q_1, Q_2, \ldots, Q_{n-1})}{\partial p_b}, \quad i \neq b$$

Even though the sensitivity of the system method can be considered a definition of a throughput bottleneck, it is also classified as a throughput bottleneck detection method in this paper. This is because, in real-world practice, practitioners can test improvements on every machine to identify the throughput bottleneck. Note that (Li et al., 2009) (Li, 2009) introduced a similar definition but using the delta notation.

4.1.4. Summary of bottleneck detection methods

Table 2 summarizes the 14 different bottleneck detection methods identified through the systematic literature review. It can be observed that only the queue length, the longest current active period method, and the bottleneck walk can be used to identify momentary throughput bottlenecks.

The two methods that focus on the queue state described in Section 4.1.1 are similar, and consequently share similar drawbacks. First, differences in the number of products in different queues may be small or non-existent if the measure used is very discrete, such as the number of jobs or batches (Roser et al., 2001). In this case, no clear identification of the bottleneck can be obtained. Second, if the size of the queue in front or after a station (input or output buffer) is limited, then the method may become inaccurate. If a method uses average queue states, then it will be difficult to detect shifting bottlenecks.

Table 2. Summary of different bottleneck detection methods identified from the literature review.

Focus	Name	Description	Capability
Queue State	Queue Length Method (Lawrence & Buss, 1994)	The station whose queue has the largest number of jobs is the bottleneck	Momentary bottleneck, average bottleneck
	Waiting Time Method (Roser et al., 2001)	The station for which jobs in the queue have the longest waiting times is the bottleneck	Average bottleneck
Process state	Utilization Method (Hopp et al., 2007)	The station with the highest utilization is the bottleneck	Average bottleneck
	Active Period Percentage Method (Roser et al., 2001)	The station with the highest active state time duration in a time period is the bottleneck	Average bottleneck
	Average Active Period Method (Roser et al., 2001)	The station with the longest average active duration is the bottleneck	Average bottleneck
	Shifting Bottleneck Detection Method (Roser et al., 2002a)	The station being the longest time momentary of shifting bottleneck during a time period is the bottleneck	Average bottleneck
	Longest Current Active Period Method (Roser et al., 2002a, Subramaniyan et al., 2016)	At any given time, the process with the longest active duration is the bottleneck	Momentary bottleneck
	Inactive Period Method (Sengupta et al., 2008)	The station with the minimum sum of the blocked state time and the starved state time is the bottleneck.	Average bottleneck
	Inter-departure Time Variance Method (Betterton & Silver, 2012)	The station with the smallest work-in-process inter-departure time variance is the bottleneck	Average bottleneck
	Arrow Method (Kuo et al., 1996)	Uses blocking and starvation information to indicate whether the bottleneck is upstream or downstream	Average bottleneck
	Turning Point Method (L. Li et al., 2009) (Li, 2009)	In addition to the condition that a station's blockage and starvation are smaller than its neighboring stations, the bottleneck station should be a 'turning point'.	Average bottleneck
Queue and Station (or System) State	Bottleneck Walk Method (Roser et al., 2015)	Uses blocking and starvation information together with information on buffer inventory levels to indicate whether the bottleneck is upstream or downstream	Momentary bottleneck, average bottleneck
	Bottleneck Index Method (Huang et al., 2019)	Uses a composite measure of utilization and the number of jobs in the queue to identify the bottleneck.	Average bottleneck
	Sensitivity of the System Method (Chang et al., 2007)	The station with the largest ratio of change in the system state is the bottleneck	Average bottleneck

Meanwhile, also all the methods that focus on the process state described in Section 4.1.2 share similar shortcomings. First, differences across stations may be small, especially at high utilization levels. In this case, no clear bottleneck can be identified. Second, if the system contains queues with no explicit limit, then the method may become inaccurate since no blocking occurs. If the methods rely on long term averages, such as the utilization method, then the results may become incorrect for shifting bottlenecks.

The *Bottleneck Walk Method* presented in Section 4.1.3 overcomes the major shortcoming of methods that use process state information if there is no blocking, i.e. a station upstream of the bottleneck that produces 100% of the time but is only filling its downstream queue would be identified as the bottleneck. The bottleneck walk method identifies this station as working; but since its downstream queue is full, the bottleneck is likely to be downstream of this station. However, it remains largely inconclusive for

a broad set of process states (working, breakdown, set-up, maintenance, scheduled break, etc.) where a station may be the bottleneck.

Finally, the *Sensitivity of The System Method* is arguably the best method since it directly reflects the definition of a bottleneck in terms of the system state. However, it has one major shortcoming: it is based on counterfactuals. In other words, one would have to systematically introduce changes and evaluate the impact using experiments and what-if analysis. This is different from the other measures discussed in this section, which are purely observational.

Overall, it can be concluded that there is limited scientific evidence reported in the literature on the suitability of different throughput bottleneck detection methods for different production contexts. There are contingency factors that determine applicability that should be considered by managers when choosing a method. These will be discussed further in Section 5. Next, this study discusses how bottleneck detection methods were operationalized in the literature.

4.2. Operationalization of bottleneck detection methods

There are different ways to classify operationalization approaches. In this paper, the approaches used in the literature to operationalize bottleneck detection method were classified into three categories: *gemba* walk, DES approach, data science approach. These categories were again chosen based on real-world practice. Each category will be discussed next before a critical discussion of the different categories is presented.

4.2.1. Gemba walk

This approach is based on human shop floor observations. Ideally, all the 14 throughput bottleneck detection methods can be implemented manually, i.e. by manually collecting the shop floor data and subsequent manual analysis of that data. However, out of all the methods in the existing literature, only the bottleneck walk (as proposed by (Roser et al., 2015)) was reportedly operationalized using a *gemba* walk. Two types of observations are collected during the bottleneck walk: machine activities and queue information. Thereafter, these two observations are used to identify the momentary and shifting throughput bottlenecks.

4.2.2. DES approach

DES models allow for the analysis of the time-dependent behavior of production systems, which is often too complex for manual analysis. In this context, a DES model of a production system is built using simulation software such as GAROPS Analyser (Roser et al., 2001), Extend (Faget et al., 2005), Simul8 (Lima et al., 2008), or Arena (Yu & Matta, 2016). The model is run for the desired time interval and the necessary data on the queue and machine states are extracted. This data is then analyzed to identify the throughput bottleneck in the production system.

Simulation is widely used to evaluate the different throughput bottleneck detection methods in the literature. For example, (Lima et al., 2008) use DES to demonstrate how the utilization method, queue length method, and waiting time method can be operationalized. Similarly, (Roser et al., 2001) use DES to demonstrates how the active period percentage method and the average active period method can be

operationalized. Meanwhile, (Roser et al., 2002a, Roser et al., 2003) and (Roser & Nakano, 2015) use DES to demonstrate how the shifting and sole active period method can be operationalized, whilst (L. Li et al., 2009) and (Li, 2018) demonstrate with DES how the turning point method can be operationalized. Finally (Chang et al., 2007), shows how the sensitivity of the system method can be operationalized using DES.

There also exists a broad set of studies using generalized simulation models to compare bottleneck detection method performance. Meanwhile (Rocha & Lopes, 2022), assessed the performance of 11 bottleneck prediction methods using a DES model of a real-life production line. In general, real-world industrial cases using DES models for analyzing throughput bottlenecks are scarce (Kuo et al., 1996) present some indication that the arrow method was implemented via DES in an automotive component plant, whilst (Faget et al., 2005) analyzed the throughput bottlenecks in the body shop production system at Volvo Cars Corporation in Sweden via the active period method by building a DES model of the production system in Extend simulation software.

4.2.3. Data science approach

Most recent literature focuses on the development of data science approaches to support the detection of throughput bottlenecks. Data science refers to a multi-disciplinary approach to extract meaningful insights from (potentially) large amounts of real-life data. It is distinguished from the Gemba walk in terms of size of data and from simulation in terms of being real-life data rather than data created through a model. This includes procedures where production system data is directly collected from the shop floor, and then analyzed using data processing and learning techniques (such as statistics, machine learning, deep learning, graphical models etc. (Hutson, 2017)). For example (Yu & Matta, 2016), propose a statistical framework using hypothesis testing techniques that can be coupled with any throughput bottleneck detection method based on process states to identify the throughput bottleneck directly from real-time data (Subramaniyan et al., 2020). propose an unsupervised machine learning-based clustering framework, that can also be coupled with any process state-based bottleneck detection method to identify throughput bottleneck clusters (Subramaniyan et al., 2016) and (Subramaniyan et al., 2018) propose statistical algorithms to operationalize the average active period and active period percentage methods using data science methods, whilst (Subramaniyan et al., 2016) propose a matrix-based data-driven algorithm to operationalize the shifting bottleneck detection method.

Meanwhile (Subramaniyan et al., 2018), and (Subramaniyan et al., 2019) propose a data science approach based on statistical techniques and the active period method to predict future throughput bottlenecks (i.e. the expected throughput bottlenecks for a future time period) and prescribe actions on them (Li et al., 2011) uses statistical techniques to predict the throughput bottlenecks using the turning point method as proposed by (Li et al., 2009). (Huang et al., 2019) uses neural networks to predict throughput bottlenecks using the bottleneck index method (Cao et al., 2012) uses adaptive neuro-fuzzy inference systems (ANFIS) to predict throughput bottlenecks using the utilization method.

4.2.4. Summary of operationalization approaches of bottleneck detection methods

Although being the simplest method, the *gemba* walk is one of the least reported in the literature (Roser et al., 2015). report several advantages of using the *gemba* walk for operationalizing the bottleneck walk method. First, momentary throughput bottlenecks can be found quickly just from manual observation without any complicated calculations involved. Second, the bottleneck walk is more accurate, especially in discrete flow production systems. Finally, as observations are collected on the shop floor, where problems occur, determining the root causes of the throughput bottlenecks is facilitated.

The above highlights a first important constraint for the *gemba* walk as operationalization approach: there should be no complicated calculations involved. Simple methods, just observing and recording simple data points, such as the queue length method, or methods focusing on direct observable system states, such as blocking and starvation, are the most amenable to this approach of operationalization. Meanwhile, for long production lines with several machines, it is time consuming to walk along the line to detect the throughput bottlenecks, whilst it may be challenging with machines that have extremely small cycle times and in environments where the throughput bottleneck shifts across machines frequently. Finally, human observations are prone to errors and can lead to wrong identification of throughput bottlenecks.

Similar, DES is one of the least reported approach for operationalization in the literature, although simulation is widely applied to compare bottleneck detection methods in generalized shop models. Generalized models are relatively easy to build and do not require data from real-life shops. They thus provide a unique platform to test, verify, and validate different throughput bottleneck detection methods. But although DES model-based analysis provides substantial value for practitioners, some limitations impede the wide application of DES for operationalizing bottleneck detection methods in practice. First, building simulation models that accurately represent a real-life production system, is very costly and time-consuming. Second, the results of the simulation model are highly dependent on the level of detail included when building the model. It is challenging to simulate all possible noises and factors to mimic the real-world production system. As a result, the outputs from simulation models could easily be misinterpreted by practitioners. Finally, it is difficult to keep the simulation model updated. But new simulation software that allows for easy data integration is likely to overcome many obstacles, whilst updating frequency and noises are typically less in more repetitive production contexts.

With the limitations of the *gemba* walk and DES, data science approaches are seen as an important alternative for throughput bottleneck detection in the literature. The use of data science approaches is enabled by recent advances in information and communication technologies (ICT) in manufacturing, which allow for the collection and the management of a larger amount and a wider variety of real-time production system data (Wuest et al., 2016). At the same time, advances in the field of data science (e.g. machine learning, statistics, etc.) offer a large and increasing number of techniques that can be used to handle and process these large amounts of real-time data (Jordan & Mitchell, 2015). Many of these techniques can easily be implemented using freely available software (such as R and Python), which offers a large potential for developing data science approaches for real-time data processing (Wuest et al., 2016). But data science remains limited in terms of causal analysis. Data science can predict the

probability that a station will be the bottleneck, but it is not able to evaluate counterfactuals. But only evaluating counterfactuals allows for precisely identifying the bottleneck according to, for example, the mathematical definition given in (Kuo et al., 1996). Some modelling needs to be reintroduced into the data analysis process. These and other emerging future research directions will be discussed next as part of Section 5.

5. Discussion

5.1. Linking bottleneck detection and operationalization mode

This study identified 14 different throughput bottleneck detection methods from the existing literature. This largely extends the sets of methods identified by (Betterton &

Table 3. Throughput bottleneck detection methods and operationalization modes.

		Operationalization modes		
Focus	Method	Gemba Walk	DES	Data science
Queue State	Queue Length		(Lima et al., 2008)	
	Waiting Time		(Lima et al., 2008)	
Process state	Utilization		(Lima et al., 2008)	Cao et al., 2012) (Yu & Matta, 2016) (Subramaniyan et al., 2020)
	Active Period Percentage		(Roser et al., 2001Lima et al., 2008)	(Subramaniyan, Skoogh, Salomonsson, Bangalore, Gopalakrishnan, et al., 2018) (Subramaniyan, Skoogh, Salomonsson, Bangalore, & Bokrantz, 2018) (Subramaniyan et al., 2019) (Yu & Matta, 2016) (Subramaniyan et al., 2020)
	Average Active Duration		(Roser et al., 2001)	(Subramaniyan, Skoogh, Gopalakrishnan, & Hanna, 2016) (Yu & Matta, 2016) (Subramaniyan et al., 2020)
	Longest Current Active Period		(Roser et al., 2002a)	(Subramaniyan, Skoogh, Gopalakrishnan, Salomonsson, et al., 2016) (Yu & Matta, 2016) (Subramaniyan et al., 2020
	Shifting Bottleneck Detection		(Roser et al., 2002a, Roser & Nakano, 2015)	(Subramaniyan, Skoogh, Gopalakrishnan, Salomonsson, et al., 2016) (Yu & Matta, 2016) (Subramaniyan et al.,2020)
	Inactive Period		(Sengupta et al., 2008)	(Yu & Matta, 2016, Subramaniyan et al., 2020)
	Inter-departure Time Variance		(Betterton & Silver, 2012)	(Yu & Matta, 2016, Subramaniyan et al., 2020)
	Arrow		(Kuo et al., 1996Chiang et al., 2000)	(Yu & Matta, 2016, Subramaniyan et al., 2020)
	Turning Point		(L. Li et al., 2009) (Li, 2009Li, 2018)	L. Li et al., 2011) (Lai et al., 2018) (Yu & Matta, 2016) (Subramaniyan et al., 2020
Queue and Station (or System) State	Bottleneck Walk	(Roser et al., 2015)		
	Bottleneck Index			(Huang et al., 2019)
	Sensitivity of the System		(Chang et al., 2007)	

Silver, 2012) and (Roser & Nakano, 2015). Table 3, links bottleneck detection methods and operationalization mode used in the sample.

From the 14 bottleneck detection methods identified, nine focus on the process state. These nine are also the most operationalized via data science approaches. An explanation is that the real-time production system data created by ICT is often machine data. The increased digitalization of the shop floor often focuses on monitoring the station activities, which makes it relatively easy to obtain the process states and associated timestamps as event log data sets from manufacturing execution systems (Subramaniyan et al., 2018, Subramaniyan et al., 2018). This has given rise to the development of data science procedures to identify throughput bottlenecks. Most operationalizations use active period-based methods, such as active period percentage, average active period, and shifting bottleneck method. There are two main reasons. First, active periods of a station account for all the events that cause blockage and starvation at other stations, thus enabling deeper insights on throughput bottlenecks (see e.g. (Subramaniyan et al., 2019)). Second, active periods are simpler to observe than potential blocking and starvation at other stations in the system.

Meanwhile, bottleneck methods that focus on the queue state are often operationalized using DES. This can be explained by queue-related measures being standard measures in common simulation software. The same holds for other modeling approaches, such as standard queueing models (Kuo et al., 1996). In contrast, measuring active periods can become quite complex using standard simulation software. Finally, the only method of which a manual implementation was reported in the literature, is the bottleneck walk.

Table 3 maps the research field on bottleneck detection. It highlights that a broad set of methods has been developed, which have been operationalized using several different approaches. However, it also highlights several research gaps through its empty cells. Other potential venues for future research will be discussed next.

5.2. Future research directions

Out of the insights gained during the review process, and the authors' own practical experience of identifying throughput bottlenecks in manufacturing industries for several years, a list of promising future research directions is next provided.

5.2.1. Comprehensive throughput bottleneck detection methods

Existing throughput bottleneck detection methods focus on process and queue states. Although these states reveal useful information on the dynamics of a production systems, they may not fully capture the system's dynamics, because there may be several other contextual factors that influence the location of throughput bottlenecks. This includes:

- *Different production system resources*: A production system may consist of several resources such as stations, buffers, transport systems, human workers, robots, and machines. The existing throughput bottleneck methods in the literature use the information only from a subset of these resources to identify throughput bottlenecks. To fully capture and understand the production system dynamics, new

throughput bottleneck detection methods are required that considers the information from all production resources.

- *Product information*: Existing methods consider an environment where there are limited product types, and each product type has similar cycle time profiles (e.g. a mass production environment). There is also the assumption that the future product types and product mix will be the same as the historical product types. However, in modern production systems, there is a risk that these assumptions are violated. To reliably detect throughput bottlenecks in environments with variable product mix, further research is required to explore how the existing throughput bottleneck detection methods can be combined with product information. For example, how can the active period method (which currently uses process state information) include product mix information to identify throughput bottlenecks?
- *Accounting of supply chain information*: Merely increasing the throughput of a production system without the inclusion of supply chain information may result in inventory buildups. Throughput bottlenecks can be better managed if they are identified based on a combination of process states, queue states, and contextual supply chain information, such as customer due dates, incoming raw material information, and backlog information. Research efforts are required to develop throughput bottleneck detection methods that also include supply chain information.
- *Quality information*: The existing methods identify throughput bottlenecks from a capacity perspective. It is inherently assumed that there are no quality issues, and good products are always produced in the system. As a result, when the existing methods are used by practitioners on the shop floor, there might be a risk of increasing the throughput of defective products. To mitigate such risks, practitioners need to include a quality perspective into bottleneck detection. Future research efforts are needed to develop methods that combine station, queue, and quality information to identify throughput bottlenecks, and eventually increase the throughput of good products.
- *Complementary outcomes*: More demands for complementary outcomes, such as sustainability, innovation, and security, are emerging (Van Wassenhove, 2019). Bottleneck detection need to reflect this since otherwise demand may become the bottleneck. For example, (Silva et al., 2021) include energy efficiency consideration into a bottleneck detection framework that extends mere throughput bottleneck detection. More research is needed to further extent this line of research given that complementary outcomes appear to gain in importance.

5.2.2. Transient state of production systems

Most of the existing literature focused on developing methods to identify throughput bottlenecks in a steady-state production system. In contrast, the applicability of these methods during the transient behavior (e.g. when the production system starts for a new shift) remains relatively unexplored. Transient analysis is also important in real-world practice as transient behavior affects throughput. Although (Roser et al., 2002a) qualitatively argue that the shifting bottleneck method can also be used in a transient state, more research is required to determine the suitability of different methods.

5.2.3. Applicability to complex production systems

The existing literature provides a large set of throughput bottleneck detection methods. These methods have been demonstrated and proved effective in detecting throughput bottlenecks mainly on serial production lines. However, modern production lines have more complex flows (e.g. with parallel machines, re-entrant production lines, shared buffers, split flows, moving assembly, etc. (Owen & Huang, 2007)). There is only limited support for practitioners to decide which throughput bottleneck detection methods are best suited for these more complex contexts. More research is required on comparing different bottleneck detection methods, and on studying the applicability of the different methods in production systems with different structures and different flows.

It is also important to prove the applicability of different throughput bottleneck detection methods in production systems that produce a high variety of products that may require varied operation sequences, such as a line flow in which different products uses different sequence of operations, different production system configurations (e.g. line flow, job shops, lines with closed loops, rework loops, etc.) and a combination of production system configurations (e.g. a production system where machining and painting are performed in a consecutive sequence), which is common in many modern real-world factories. The existing throughput bottleneck detection methods were mostly proven to work in production systems in which the stations have the same type of operations. For example (Li et al., 2009), demonstrates the applicability of the turning point method in an assembly production system, in which assembly operations are performed at all stations. Similarly (Subramaniyan et al., 2020), demonstrated the data science approach of clustering the machines on a machining production line. Studying the applicability of throughput bottleneck detection methods for varied operation sequences consequently requires further research. This will also help to scale the level of analysis of the different methods, from detecting the bottlenecks in the production system to detecting the bottlenecks in the entire factory.

5.2.4. Validation of throughput bottleneck detection methods

When developing throughput bottleneck detection methods, it is common for researchers to use DES to verify and validate the different bottleneck detection methods (as seen in Table 3). Thereafter, researchers argue that the throughput bottleneck detection methods will also work in the real-world. But how can one be sure that the developed methods identify the right throughput bottlenecks in the real-world? One way to answer this question is to implement the different bottleneck detection methods in practice and to assess the results. For example, when practitioners resolve the throughput bottlenecks in the real world, one can assess if the actual production system throughput has increased. None of the existing literature provides real-world validation of methods. Future research activities are needed to implement the methods and study their effects.

5.2.5. Stochasticity and shifting bottlenecks

Most bottleneck detection methods were developed for stable contexts. But companies in most need of bottleneck detection methods are often companies with shifting bottlenecks. Bottlenecks are a stochastic phenomenon and identifying bottlenecks quickly and reliably is of utmost importance. While the shifting bottleneck method explicitly considers this dynamicity, it remains largely unknown how well other methods perform in contexts with shifting

bottlenecks. Future research should evaluate the performance of alternative methods, including the evaluation of response time (quickness of detecting a shift) and accuracy.

5.2.6. Digital Twins for throughput bottleneck detection

Data science was the most followed approach for operationalization in practice. This was motivated by advances in technology. But advances in technology also affect simulation. A specific form of real-time modeling that has received recent attention is the so-called digital twin (e.g (Shao & Helu, n.d..), (Lugaresi & Matta, 2021)). The use of digital twins recognizes the need to respond to emerging problems quickly, which makes it specifically suitable for shifting bottlenecks. For example, a digital twin of the production system can automatically analyze the throughput bottlenecks from the real-time data sets, predict the expected dynamics using data science, examine the different scenarios of eliminating the bottlenecks, and prescribe actions to resolve these bottlenecks in the real world. Such a type of digital twin can continuously evolve using the real-time data of the production system and shop floor engineers' feedback. Future research is required to develop such a twin, and to evaluate how it can be operationalized for throughput bottleneck detection and control.

5.3. Practical implications

5.3.1. Guidelines for selecting a throughput bottleneck detection method

Table 2 provides a set of throughput bottleneck detection methods. Choosing a suitable throughput bottleneck detection method is highly dependent on the structure of the production system and the available production system information. For example, a rigorous mathematical definition can and should be chosen in serial production lines, and methods implemented that are based on this definition. However, in high-variety make-to-order shops with complex routing such rigorous definition may not be possible and a simpler more intuitive method needs to be adopted. In general, the following guidelines apply.

Methods based on queue states: These throughput bottleneck detection methods can only be applied in production systems that have intermediate buffers between the stations. The selection of a particular method depends on the capacity of the buffer. If the buffer limit is commonly reached, then the waiting time method is more suitable. If the buffer limit is never reached, then the queue length method can be used.

Methods based on processes states: These throughput bottleneck detection methods can be applied for production systems with or without intermediate buffers. The selection of method is highly dependent on the level of complexity that is considered feasible. In general, the active period methods (including the active period percentages, average active period, longest current active period and shifting bottleneck detection method) are more complex than other methods. This is because they consider all stochastic events that influence the production system throughput, and all the information about these stochastic events needs to be available. In contrast, the turning point method uses only the blockage and starvation information. The required level of complexity is best judged by practitioner experience and their assessment of the feasibility of a method. Meanwhile, the inter departure time method may identify non-bottlenecks instead of bottlenecks if the coefficient of variation is the same for bottlenecks and non-bottlenecks (Thürer et al., 2021). If there is blocking, then the inter departure time method may identify a blocked

station as a bottleneck station given the time delay that is necessarily introduced because the inter departure time is calculated based on historical data.

Methods based on system states: These throughput bottleneck detection methods can also be applied for production systems with or without intermediate buffers. These methods are more complex compared to the methods based on the process states since they require both queue and station information to detect throughput bottlenecks. These methods can be more useful for production systems that have complex flows such as split flows, parallel flows etc.

Overall, it must be noted that none of the throughput bottleneck detection methods is perfect in a rigorous sense. Every production system is different and generalizing the bottleneck detection methods is challenging. Practitioners need to use their domain knowledge of the specific production system to cautiously evaluate the different methods and only then select a method.

Table 4. Operationalization modes: suitability, advantages and challenges.

Modes	When to use	Advantages	Challenges
Gemba Walk	• Suitable for non-digitalized production system. • Can be used to detect throughput bottlenecks when practitioners are present on the shop floor. • Needs relatively stable production system. • Can identify momentary and average bottlenecks.	• Allows for directly observing the production system dynamics and identifying throughput bottlenecks. • Facilitates quick decisions. • Checks and aligns the practitioner's perception on throughput bottlenecks with reality.	• Considers limited stochastic events. • Needs extensive manual efforts and is time consuming. • Limited possibilities to detect shifting bottlenecks in real-time. • Lack of predictive capabilities.
DES	• Suitable for both non-digitalized and digitalized production systems. • Production system data should be available (either in digital format or manual recorded observations). • Suitable for back-office analysis of throughput bottlenecks. • Suitable to identify average throughput bottlenecks.	• Considers all types of stochastic events. • Can support "what if" analysis, i.e. test interventions to resolve throughput bottlenecks before implementation.	• Time consuming to build simulation models. • Requires high input data quality. • Need of extensive manual efforts and expertise (e.g. programming) to build and verify simulation models. • Difficult to keep the model updated in dynamic contexts.
Data science	• Suitable for digitalized production systems. • Digital production system data should be available (e.g. event log data). • Back-office analysis of the throughput bottlenecks. • Suitable to identify momentary, average and shifting bottlenecks.	• Considers all stochastic events (even disturbances in seconds time scale). • Can identify the throughput bottlenecks in real-time. • Enables quick decisions. • Can predict future throughput bottlenecks. • Has the capability to automatically learn about changing production system dynamics (i.e. without much manual efforts compared to DES).	• Requires high input data quality. • There may be a bias in the input data. • Limited capability to test interventions before implementation in the real-world.

5.3.2. Guidelines on selecting an appropriate operationalization mode

After selecting an appropriate throughput bottleneck detection method, an operationalization mode to operationalize the method needs to be determined. The three different operationalization modes considered in this study are summarized in Table 4 together with their suitability, advantages, and challenges.

If there is no digital production system data available, then *gemba* walks are the first choice. In general, the *gemba* walk is a good choice if bottlenecks need to be detected less frequently, i.e. are stable and less likely to shift. Even though Table 3 may suggest that only the bottleneck walk falls under the *gemba* walk, in real-world practice, most of the bottleneck detection methods can also be operationalized using the *gemba* walk. However, during the *gemba* walk only a limited number of stochastic events can be taken into account. The results may therefore not always be accurate.

To increase the accuracy of the results, DES models of the production system can be used. DES constitutes a unique approach that enables the simultaneous consideration of a range of stochastic events. DES also provides more insights on which stochastic events are truly important in influencing the behavior of the throughput bottlenecks and which are presumably less so. Moreover, using DES practitioners can test different interventions to resolve throughput bottlenecks and select the best strategy for implementation. Despite the benefits of DES, the creation of appropriate models is often challenging as it is very time consuming to construct the model and to keep the model updated with the changes taking place in the real-world production system.

Data science approaches are more suitable for back-office monitoring of a completely digitalized production systems. They can directly analyze real-time data and provide real-time insights to practitioners. This allows for effective decisions to resolve bottlenecks quickly. Data science approaches also have the capability to predict the throughput bottlenecks before a production run (e.g. production shift) so that practitioners can start their production run with a complete understanding of the expected throughput bottleneck stations for that production run. This can help them to act proactively. However, data science approaches cannot identify causes and remain entrenched in probabilities. For the same reason they remain limited in terms of evaluation of potential interventions. A summary of data science approaches together with detailed guidelines for implementation can be found in (Subramaniyan et al., 2021).

6. Conclusion

The importance of bottlenecks is widely recognized in the literature and in practice. Bottleneck detection is the first step in bottleneck management leading to a large literature proposing different bottleneck detection methods. However, to-date no comprehensive review of the literature on throughput bottleneck detection has been conducted. This leaves practitioners alone in their task to choose an appropriate bottleneck detection method and its operationalization for their shop. In response, a systematic literature review was conducted to consolidate the field. A total of 14 bottleneck detection methods were identified, which can be classified according to the measure applied. This provides a comprehensive set of methods and significantly extends existing reviews that only provided limited sets. Meanwhile, three modes of operationalization were identified. The literature was mapped along bottleneck detection method and operationalization mode, and a series of important research issues

outlined. This includes the consideration of other constraining resources, such as workers, tooling, or transportations, and the development of a digital twin-based method for bottleneck detection and control. Meanwhile, this review identified contingency factors for method application and discusses important guidelines for the application of the different operationalization modes. This guides practitioners on which method and operationalization to adopt in their shop.

Disclosure statement

No potential conflict of interest was reported by the author(s).

Funding

No funds, grants, or other support was received

ORCID

Anders Skoogh http://orcid.org/0000-0001-8519-0736
Andrea Matta http://orcid.org/0000-0003-3902-2007

References

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*(15), 4158–4174. https://doi.org/10.1080/00207543.2011.596847

Cao, Z., Deng, J., Liu, M., & Wang, Y. (2012). Bottleneck prediction method based on improved adaptive network-based fuzzy inference system (ANFIS) in semiconductor manufacturing system. *Chinese Journal of Chemical Engineering*, *20*(6), 1081–1088. https://doi.org/10.1016/S1004-9541(12)60590-4

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* (3), 653. https://doi.org/10.1115/1.2673666

Chiang, S.-Y., Kuo, C.-T., & Meerkov, S. M. (2000). DT-bottlenecks in serial production lines: Theory and application. *IEEE Transactions on Robotics and Automation*, *16*(5), 567–580. https://doi.org/10.1109/70.880806

Faget, P., Erkisson, 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*, Orlando, Florida, USA, (pp. 1401–1407).

Goldrat, E. M., & Cox, J. (1990). *The goal: A process of ongoing improvement (Third revi)*. North River Press.

Hopp, W. J., Iravani, S. M. R., & Shou, B. (2007). A diagnostic tree for improving production line performance. *Production and Operations Management*, *16*(1), 77–92. https://doi.org/10.1111/j.1937-5956.2007.tb00167.x

Hopp, W. J., & Spearman, M. L. (2008). *Factory physics : Foundations of manufacturing management* (2nd ed.). Long Grove, Illinois, US: Waveland Press.

Huang, B., Wang, W., Ren, S., Zhong, R. Y., & Jiang, J. (2019). A proactive task dispatching method based on future bottleneck prediction for the smart factory. *International Journal of Computer Integrated Manufacturing*, *32*(3), 278–293. https://doi.org/10.1080/0951192X.2019.1571241

Hutson, M. (2017). AI glossary: Our, in so many words. *Science (New York, NY)*, *357*(6346), 19. https://doi.org/10.1126/science.357.6346.19

Jordan, M. I., & Mitchell, T. M. (2015). Machine learning: Trends, perspectives, and prospects. *Science*, *349*(6245), 255–260. https://doi.org/10.1126/science.aaa8415

Kuo, C.-T., Lim, J.-T., & Meerkov, S. M. (1996). Bottlenecks in serial production lines: A system-theoretic approach. *Mathematical Problems in Engineering*, *2*(3), 233–276. https://doi.org/10.1155/S1024123X96000348

Law, A. M., & Kelton, D. W. (1991). *Simulation modeling & analysis*. McGraw Hill.

Lawrence, R. S., & Buss, H. A. (1994). Shifting production bottlenecks: Causes, cures and conundrums. *Production and Operations Management*, *3*(1), 21–37. https://doi.org/10.1111/j.1937-5956.1994.tb00107.x

Li, L. (2009). Bottleneck detection of complex manufacturing systems using a data-driven method. *International Journal of Production Research*, *47*(24), 6929–6940. https://doi.org/10.1080/00207540802427894

Li, L. (2018). A systematic-theoretic analysis of data-driven throughput bottleneck detection of production systems. *Journal of Manufacturing Systems*, *47*, 43–52. https://doi.org/10.1016/j.jmsy.2018.03.001

Li, J., Blumenfeld, D. E., Huang, N., & Alden, J. M. (2009). Throughput analysis of production systems: Recent advances and future topics. *International Journal of Production Research*, *47*(14), 3823–3851. https://doi.org/10.1080/00207540701829752

Li, L., Chang, Q., & Ni, J. (2009). Data driven bottleneck detection of manufacturing systems. *International Journal of Production Research*, *47*(18), 5019–5036. https://doi.org/10.1080/00207540701881860

Lima, E., Chwif, L., & Barreto, M. R. P. (2008). Metodology for selecting the best suitable bottleneck detection method. In S. J. Mason, R. Hill, L. Mönch, O. Rose, T. Jefferson, & J. W. Fowler (Eds.), *Proceedings of the 2008 Winter Simulation Conference*, Miami, Florida, US, (pp. 1746–1751).

Li, L., Qing, C., Xiao, G., & Ambani, S. (2011). Throughput bottleneck prediction of manufacturing systems using time series analysis. *Journal of Manufacturing Science and Engineering*, *133*(2), 1–8. https://doi.org/10.1115/1.4003786

Lugaresi, G., & Matta, A. (2021). Automated manufacturing system discovery and digital twin generation. *Journal of Manufacturing Systems*, *59*(January), 51–66. https://doi.org/10.1016/j.jmsy.2021.01.005

Owen, J. H., & Huang, N. (2007). Local improvements that degrade system performance: Case studies and discussion for throughput analysis. *International Journal of Production Research*, *45*(10), 2351–2364. https://doi.org/10.1080/00207540600791616

Pehrsson, L., Ng, A. H. C., & Bernedixen, J. (2016). Automatic identification of constraints and improvement actions in production systems using multi-objective optimization and post-optimality analysis. *Journal of Manufacturing Systems*, *39*, 24–37. https://doi.org/10.1016/j.jmsy.2016.02.001

Rocha, E. M., & Lopes, M. J. (2022). Bottleneck prediction and data-driven discrete-event simulation for a balanced manufacturing line In *Procedia computer science* (pp. 1145–1154. Vol. 200). Elsevier BV. https://doi.org/10.1016/j.procs.2022.01.314

Roser, C., Lorentzen, K., & Deuse, J. (2015). Reliable shop floor bottleneck detection for flow lines through process and inventory observations: The bottleneck walk. *Logistics Research*, *8*(1), 1–9. https://doi.org/10.1007/s12159-015-0127-2

Roser, C., & Nakano, M. (2015). A quantitative comparison of bottleneck detection methods in manufacturing systems with particular consideration for shifting bottlenecks. *IFIP International Federation for Information Processing 2015*, *1*, 273–281. https://doi.org/10.1007/978-3-319-22759-7

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). IEEE. https://doi.org/10.1109/WSC.2001.977398

Roser, C., Nakano, M., & Tanaka, M. (2002a). Shifting bottleneck detection. In E. Yucesan, C.-H. Chen, J. L. Snowdon, & J. M. Charnes (Eds.), *Proceedings of the 2002 Winter Simulation Conference* (Vol. 2). https://doi.org/10.1109/WSC.2002.1166360

Roser, C., Nakano, M., & Tanaka, M.(2002b). Throughput sensitivity analysis using a single simulation. *Proceedings of the Winter Simulation Conference* (Vol. 2, pp. 1087–1094). San Diego, California, US.

Roser, C., Nakano, M., & Tanaka, M. (2003). Comparison of bottleneck detection methods for AGV systems. In S. Chick, S. PJ, & D. Ferrin (Eds.), *Proceedings of the 2003 Winter Simulation Conference*, New Orleans, Louisiana, US, (pp. 556–564).

Schmenner, R. W., & Swink, M. L. (1998). On theory in operations management. *Journal of Operations Management*, *17*(1), 97–113. https://doi.org/10.1016/S0272-6963(98)00028-X

Sengupta, S., Das, K., & VanTil, R. P. (2008). A new method for bottleneck detection. In S. J. Mason, R. R. Hill, L. Monch, O. Rose, T. Jefferson, & J. W. Fowler (Eds.), *Proceedings of the 2008 Winter Simulation Conference*, Miami, Florida, US, (pp. 1259–1267).

Shao, G., & Helu, M. (n.d.). Framework for a digital twin in manufacturing: Scope and requirements. *Manufacturing Letters*. https://doi.org/10.1016/j.mfglet.2020.04.004

Silva, G. V., Thomitzek, M., Abraham, T., & Herrmann, C. (2021). Bottleneck reduction strategies for energy efficiency in the battery manufacturing. In *Procedia CIRP* (Vol. 104, pp. 1017–1022). Elsevier BV. https://doi.org/10.1016/j.procir.2021.11.171

Subramaniyan, M., Skoogh, A., Bokrantz, J., Sheikh, M. A., Thürer, M., & Chang, Q. (2021). Artificial intelligence for throughput bottleneck analysis – state-of-the-art and future directions. *Journal of Manufacturing Systems*, *60*, 734–751. https://doi.org/10.1016/j.jmsy.2021.07.021

Subramaniyan, M., Skoogh, A., Gopalakrishnan, M., & Hanna, A. (2016). Real-time data-driven average active period method for bottleneck detection. *International Journal of Design & Nature and Ecodynamics*, *11*(3), 428–437. https://doi.org/10.2495/DNE-V11-N3-428-437

Subramaniyan, M., Skoogh, A., Gopalakrishnan, M., Salomonsson, H., Hanna, A., & Lämkull, D. (2016). An algorithm for data-driven shifting bottleneck detection. *Cogent Engineering*, *3*(1), 1–19. https://doi.org/10.1080/23311916.2016.1239516

Subramaniyan, M., Skoogh, A., Muhammad, A. S., Bokrantz, J., Johansson, B., & Roser, C. (2020). A generic hierarchical clustering approach for detecting bottlenecks in manufacturing. *Journal of Manufacturing Systems*, *55*, 143–158. https://doi.org/10.1016/j.jmsy.2020.02.011

Subramaniyan, M., Skoogh, A., Salomonsson, H., Bangalore, P., & Bokrantz, J. (2018). A data-driven algorithm to predict throughput bottlenecks in a production system based on active periods of the machines. *Computers and Industrial Engineering*, *125*, 533–544. https://doi.org/10.1016/j.cie.2018.04.024

Subramaniyan, M., Skoogh, A., Salomonsson, H., Bangalore, P., Gopalakrishnan, M., & Sheikh Muhammad, A. (2018). Data-driven algorithm for throughput bottleneck analysis of production systems. *Production and Manufacturing Research*, *6*(1), 225–246. https://doi.org/10.1080/21693277.2018.1496491

Subramaniyan, M., Skoogh, A., Sheikh Muhammad, A., Bokrantz, J., & Turanoğlu Bekar, E. (2019). A prognostic algorithm to prescribe improvement measures on throughput bottlenecks. *Journal of Manufacturing Systems*, *53*, 271–281. https://doi.org/10.1016/j.jmsy.2019.07.004

Thürer, M., Ma, L., Stevenson, M., & Roser, C. (2021). Bottleneck detection in high-variety make-to-order shops with complex routings: An assessment by simulation. *Production Planning and Control*, 1–12. https://doi.org/10.1080/09537287.2021.1885795

Tranfield, D., Denyer, D., & Smart, P. (2003). Towards a methodology for developing evidence-informed Management knowledge by means of systematic review* introduction: The need for an evidence- informed approach. *British Journal of Management*, *14*(3), 207–222. https://doi.org/10.1111/1467-8551.00375

Van Wassenhove, L. N. (2019). Sustainable innovation: Pushing the boundaries of traditional Operations Management. *Production and Operations Management*, *28*(12), 2930–2945. https://doi.org/10.1111/poms.13114

Wang, L. C., Chu, P. C., & Lin, S. Y. (2019). Impact of capacity fluctuation on throughput performance for semiconductor wafer fabrication. *Robotics and Computer-Integrated Manufacturing*, *55*(March 2017), 208–216. https://doi.org/10.1016/j.rcim.2018.03.005

Wang, Y.-C., Qian-Chuan Zhao, D.-Z.-Z., & Zheng, D. (2005). Bottlenecks in production networks: An overview. *Journal of Systems Science and Systems Engineering*, *14*(3), 347–363. https://doi.org/10.1007/s11518-006-0198-3

Wuest, T., Weimer, D., Irgens, C., & Thoben, K. (2016). Machine learning in manufacturing: Advantages, challenges, and applications. *Production and Manufacturing Research*, *4*(1), 1–23. https://doi.org/10.1080/21693277.2016.1192517

Wu, K., Zhou, Y., & Zhao, N. (2016). Variability and the fundamental properties of production lines. *Computers and Industrial Engineering*, *99*, 364–371. https://doi.org/10.1016/j.cie.2016.04.014

Yu, C., & Matta, A. (2016). Data-driven bottleneck detection in manufacturing systems: A statistical approach. *International Journal of Production Research*, *54*(21), 6317–6322. https://doi.org/10.1080/00207543.2015.1126681