Journal of Manufacturing Systems 66 (2023) 92–106

Contents lists available at ScienceDirect

Journal of Manufacturing Systems

journal homepage: www.elsevier.com/locate/jmansys

Technical Paper

A digital twin based framework for detection, diagnosis, and improvement of throughput bottlenecks

Mahesh Kumbhar [a,*], Amos H.C. Ng [a,b], Sunith Bandaru [a]

[a] Division of Intelligent Production Systems, School of Engineering Science, University of Skövde, 541 28, Skövde, Sweden

[b] Division of Industrial Engineering and Management, Department of Civil and Industrial Engineering, Uppsala University, 752 36, Uppsala, Sweden

ARTICLE INFO

Keywords:
Digital twin
Bottleneck detection
Process mining
Factory physics
Utilization
Simulation
Industry 4.0

ABSTRACT

Digitalization through Industry 4.0 technologies is one of the essential steps for the complete collaboration, communication, and integration of heterogeneous resources in a manufacturing organization towards improving manufacturing performance. One of the ways is to measure the effective utilization of critical resources, also known as bottlenecks. Finding such critical resources in a manufacturing system has been a significant focus of manufacturing research for several decades. However, finding a bottleneck in a complex manufacturing system is difficult due to the interdependencies and interactions of many resources. In this work, a digital twin framework is developed to detect, diagnose, and improve bottleneck resources using utilization-based bottleneck analysis, process mining, and diagnostic analytics. Unlike existing bottleneck detection methods, this novel approach is capable of directly utilizing enterprise data from multiple levels, namely production planning, process execution, and asset monitoring, to generate event-log which can be fed into a digital twin. This enables not only the detection and diagnosis of bottleneck resources, but also validation of various what-if improvement scenarios. The digital twin itself is generated through process mining techniques, which can extract the main process map from a complex system. The results show that the utilization can detect both sole and shifting bottlenecks in a complex manufacturing system. Diagnosing and managing bottleneck resources through the proposed approach yielded a minimum throughput improvement of 10% in a real factory setting. The concept of a custom digital twin for a specific context and goal opens many new possibilities for studying the strong interaction of multi-source data and decision-making in a manufacturing system. This methodology also has the potential to be exploited for multi-objective optimization of bottleneck resources.

1. Introduction

In this current age of digitalization and data, manufacturing companies need to adapt rapidly to technological changes brought by Industry 4.0 revolution in order to be competitive and sustainable. However, real factory data has both stochastic and dynamic characteristics due to random events such as machine failures, processing time, and material arrival [1–3], and it needs to be transformed into knowledge for a manufacturing company to move towards being a data-driven organization. Generally, manufacturing systems are categorized as continuous or discrete [2]. In the current paper, we focus on discrete manufacturing systems where data-driven initiatives can be implemented to create knowledge from data.

Manufacturing performance measures organizational objectives and it is mainly governed by repeatability and predictability [2]. Repeatability determines manufacturing consistency and preciseness of output, while predictability refers to forecasting of manufacturing output. The repeatability can be measured by production rate, which is the number of products produced in a specified time. This repeatability is also called *throughput* (TH). Similarly, Manufacturing organizations become competitive by increasing TH and reducing inventory [4]. The author invented the term 'bottleneck' (BT) for a resource whose capacity determines the system TH. Every manufacturing system has BT resources, which can be leveraged to control product flow [4]. Hence, BT detection is essential to achieve long-term sustainability and productivity objectives. Many methods, such as model and data-driven, have been developed to detect the BT, but Roser et al. [5] argued that two-thirds of the BT detected by existing methods differ from what managers of the line believe. Most of the previous methods are based on a single routing manufacturing system, and a BT is a higher utilization resource. However, today's manufacturing systems consist of complex product

* Corresponding author.
E-mail addresses: mahesh.kumbhar@his.se (M. Kumbhar), amos.ng@his.se, amos.ng@angstrom.uu.se (A.H.C. Ng), sunith.bandaru@his.se (S. Bandaru).

https://doi.org/10.1016/j.jmsy.2022.11.016
Received 16 October 2022; Received in revised form 23 November 2022; Accepted 26 November 2022
Available online 13 December 2022
0278-6125/

routing where the arrival rate is different for each resource, and the BT may not be one with the slowest resource [6]. The current digital transformation is enabling organizations in capturing this complex product routing information. However, there is still limited research on complex manufacturing layouts, capturing a real-time stochastic behavior to detect the BT using data-driven methods [7]. A challenge that still exists is: *Can a data-driven methodology be developed to detect BTs in complex manufacturing systems consisting of both processing and assembly operations, and can it provide TH improvement recommendations for decision-making?*

This paper answers the above questions in a robust methodological way by presenting a digital twin (DT) framework for detecting the BT on complex manufacturing systems using utilization-based analytics. Furthermore, detected BT needs diagnosis for TH improvement opportunities. The proposed methodology exploits shop-floor data by creating process-centric event-log from them using process mining which generates abstracted process map. On this map, a manufacturing area is then selected to form a DT for BT detection and TH improvement.

In addition, detecting the BT resource requires insights from data which is generated using production planning, process execution, and asset monitoring. Firstly, high-level production planning confirms the availability of resources. Secondly, at the medium level, resources perform process execution steps. Finally, these resources are monitored and controlled to get the most output from the manufacturing resources. This multi-source information needs data merging, preprocessing, and exploration. Processed data needs validation to confirm current process execution in a manufacturing system.

The key contributions of the paper are:

- A DT framework for BT detection and TH improvement is introduced, which bridges the gap between process and data science.
- Utilization-based BT detection proposed, which detects sole and shifting BT for any complex routing manufacturing system.
- BT diagnosis is investigated to find possible improvement areas for detected BT.
- TH improvement opportunities are analyzed for decision-making.
- The proposed methodology is illustrated by a hypothetical simulation model and a real-world application.

To this end, Section 2 gives an overview of the literature on BT detection and factory physics fundamentals. Section 3 proposes a DT framework consisting of data collection, functional, and user interaction to detect sole and shifting BT for TH improvement recommendations. Section 4 reports DT verification on a real-world application. Furthermore, Section 5 discusses research contributions and implications for industry practitioners. Finally, Section 6 concludes the paper and mentions our future research directions.

2. Literature review

Manufacturing performance is based on the organization's goals, characterized by resource utilization or total lead time. This organization's goal achievement is decided by a resource which is termed a BT [8,9]. A BT that constrains a system to achieve a shorter lead time is defined as cycle time BT, whereas TH BT constraints a system preventing higher resource utilization in terms of TH rate. BT in a system is used to control product flow reflecting long-term utilization [4]. TH is measured using Little's law as $TH = WIP/Average\ Cycle\ Time$. However, these attributes cannot be measured to estimate TH. A system consists of independent machines forming a workstation or a cell and creating a manufacturing line [6]. Resource performance can be estimated using parameters such as buffer level, machine blockage, or starvation [10]. Based on these parameters, different BT methods exist in the literature, which are categorized into model-based and data-driven and are summarized in Table 1. The model-based methods form an analytical or a simulation model representing resource parameters such as machine productivity, reliability, quality, buffer capacity, and flow routing. These parameters measure the manufacturing system's performance. However, data-driven methods use real-time manufacturing data. The model-based methods require manufacturing system understanding and need appropriate statistical distribution for modeling parameters. Although many TH BT methods exist in the literature, they may not be active every time as resources are not utilized 100% of the time. Therefore, an organization can achieve its goals by determining TH BT and improving resource utilization [9,11].

In model-based methods, Hopp and Spearman [6] define BT as a resource having the highest utilization rate, given by $arrival\ rate/departure\ rate$. This definition is valid in all manufacturing systems as validated by Kasemset and Kachitvichyanukul [12] using simulation. Due to the inherent resource variability, a detected BT need not be stationary, in which case it is called a shifting BT [13]. This shifting BT can be measured using queue length variation across workstations. If its value is near one, all workstations have an equal BT probability, shifting between workstations. If its value is near zero, there is only a static BT. The shifting BT is also observed by Zhao and Li [14], who improved a BT station by assigning additional resources, and the primary BT shifted to another resource. Besides machine utilization, WIP inventory can also be used to find a BT [15]. However, it is difficult to measure in real-time; hence machine blockage and starvation time is used. A BT is located upstream of a machine m_{j+1} if blockage time of m_j is less than starvation time of m_{j+1} ($mb_j < ms_{j+1}$), and BT machine is located downstream of m_{j-1} if starvation time of m_j is less than blockage time m_{j-1} ($ms_j < mb_{j-1}$) as show in Fig. 1. In this figure, m_j is a BT where blocking arrow from m_{j-1} and starving arrow from m_{j+1} pointing towards m_j. However, for multiple BT, the severity of blockage and starvation on a workstation is estimated, and the machine that has the highest sensitivity for TH change is identified as a BT in a multi-product serial line with assembly [14,15]. Besides seeing only production BT, quality BT is introduced by Meerkov and Zhang [16], in which a resource affects TH severely. This method assumes Bernoulli reliability and quality models, which were simulated on serial lines with single and multiple inspection stations with constant processing time. However, this approach considers one parameter for detecting BT, but practically, multiple factors affect system performance. To overcome this problem, Chiang et al. [17,18] formulated a method considering the Markovian approach with two machine performance parameters: uptime and downtime. The method detects BT where a machine's uptime and downtime are most critical. This method detects a BT whose increased machine uptime or downtime has the maximum impact on system performance. The arrow method is extended to a flexible manufacturing system (FMS) producing multi-product [19]. Apart from this, some operation research methods were developed. Zhao et al. [20] proposed BT detection using a network model with workstation capacity on the assembly line. This network model is converted into a combinatorial max-flow optimization problem which was solved by the genetic algorithm. To meet the current industry demand for BT detection in FMS, Yan et al. [21] propose shifting BT analysis. This method changes working and idle states using a self-learning method on a knowledgeable manufacturing system. The shifting BT depends on the machine's busy state and buffers in front of a resource, whereas the Roser et al. [22] method detects BT shift due to random failure of a machine. BT can also be controlled using resource variability and is defined as a variation of processing time, demand, and machine failure [13]. BT can be detected and controlled by having a minimum cycle time [23]. In addition, Sengupta et al. [24] proposed inter-departure time method and a resource time broken into working (busy), idle (due to blocked up-stream resource), self failure, and down-stream resource failure or full buffer (blocked-down). A resource blocked-up and blocked-down time are added, and a resource with the lowest blocked-up and blocked-down time is detected as a bottleneck. Similarly, Betterton and Silver [25] developed the inter-departure variance

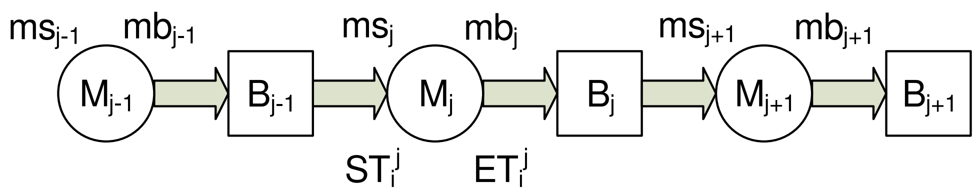

Fig. 1. Serial production system consisting of buffers and machines with blockage and starvation probabilities along with start and end time for ith instance on machine j.

Table 1
Summary of literature review.

Author	Method	Metrics used for bottleneck detection
Lawrence and Buss [13]	Queue length	WIP inventory queue
Kuo et al. [15]	Arrow method	Blockage and starvation time
Hopp and Spearman [6]	Factory physics or utilization	Arrival and output rate
Roser et al. [26]	Active period	Processing time contribution in working, repair, set-up
Li et al. [27]	Turning point	Blockage and starvation time
Sengupta et al. [24]	Inter-departure time	Blocked upstream and blocked downstream time
Leporis and Zedenka [28]	Criticality indicator	Utilization, blocking, starving, waiting for operator
Aalst et al. [29]	Process mining	Highest cycle time
Betterton and Silver [25]	Inter-departure time variance	Inter-departure time
Muthiah and Huang [30]	OTE	OEE, quality efficiency, and theoretical processing time
Zhao et al. [20]	Network Model	Cycle time
Roser et al. [5]	Bottleneck walk	Observations WIP inventories
Zhang and Matta [31]	Discrete Event Optimization	Decrease of downtime
Subramaniyan et al. [32]	Active period	Processing time contribution in processing, repair, set-up
Tang [33]	OEE	Availability, efficiency, and quality performance
Subramaniyan et al. [34]	Clustering	Active period
Kahraman et al. [35]	Projected buffer depletion	Buffer level measurement
Kumbhar et al. [36]	Utilization or Busy ratio	Arrival and output rate

BT method on a serial line, and a resource having the smallest inter-departure variance is BT. This method has a few limitations, such as large buffers surrounding a resource having a faster machine with minimum downtime.

However, model-based methods depend on a performance metric, metric combination such as overall equipment effectiveness (OEE) is used to detect a BT [30,33]. Tao et al. [37] modified OEE, and a resource with the lowest standalone OEE is detected as a BT. The method is verified using BT detection on a vehicle assembly plant. Similarly, Muthiah and Huang [30] used OEE to come up overall throughput effectiveness (OTE) ($actualTH/theoreticalTH$). For example, consider a serial line as depicted in Fig. 1. Suppose machine m_{j-1} is producing with less than 100% quality efficiency, then the output from a machine m_j is dependent on output and quality efficiency from the machine m_{j-1}. OTE is calculated with OEE, quality efficiency, and theoretical processing. The method was validated with serial, parallel, and complex systems simulation examples and an industrial case.

Apart from the usual performance parameters of resources, Roser et al. [26] suggested breaking processing time into the active and in-active states. Active state refers to a condition contributing to system TH: working, maintenance, setup, and repair. In-active state refers to waiting. A resource with the highest average active state is referred to as a BT resource. In addition, Roser et al. [22] extended the active period for shifting BT, which occurs for a given time if BT active period overlaps with another resource's active period. No station is the sole BT during this state and BT shifts between overlapped resources. The proposed method is simulated on a flow and job shop production. This method is robust compared to the queue length shifting BT [13], as the active period fluctuates lesser than the queue length. Due to different state data availability, the active period method helps to diagnose BT. Generally, a BT is assumed to be a machine, but it can be an AGV, a robot, or a human worker [38]. In addition, Leporis and Zedenka [28] developed a BT detection having the highest criticality indicator among workstations. This indicator is derived from the cumulative addition of the average difference between utilization, blocking, starving, and waiting for labor. Furthermore, Zhang and Matta [31] developed the BT method using benders decomposition MILP model using discrete event optimization on a flow line. The BT is detected where a minimum decrease in downtime of a machine gives desired increase of TH (by decreasing overall cycle time). The model is compared with the arrow method and converges faster than the arrow method.

Apart from the model and data-driven methods, Roser et al. [5] proposed observing process inventories in a system that is defined as BT walk. It eliminates processes that are not waiting for the parts. A BT is upstream of the starved process if it is waiting for the parts, or a BT is in a downstream process where the process is blocked for parts transport. BT detection is also explored in areas where a buffer is large such as mining operations [35]. The process needs to hold cycle inventory for a batch, and the output rate is lesser than the input rate. So, input to a buffer is possibly a BT and is termed as buffer depletion projection. The minimum of all buffer depletion is detected as a BT.

In recent years, there has been an increasing amount of literature on data-driven BT detection. Li et al. [27] proposed the turning point method using buffer, which absorbs variation, to segment machine-buffer combination. A machine is BT, where an existing process blocks upstream and starves downstream components flow. This BT machine has minimum starvation and blockage time, reversing this magnitude before and after BT. Heuristic "BT index" is introduced if a system has multiple BT, and its magnitude indicates BT criticality. The turning point method can detect local and global BT, whereas the arrow method can detect only local BT [11]. Li [39] extended the turning point on complex production systems consisting of concurrent and closed-loop feedback processes. Instead of comparing blockage and starvation, Li [40] subtracted uptime from upstream blockage or downstream starvation, and the lowest estimate is assigned for a machine. A machine which is having the smallest estimate is detected as BT. The method has been proven analytically and simulated on a serial line that can detect BT in the short-term and long-term. However, this method fails in a complex production system like assembly, feedback loop, and parallel line. To find BT on a complex production line, the production line is broken down into a serial line by considering pseudo station [7] to identify the BT using the turning point. Subramaniyan et al. [32,41] proposed a data-driven BT detection method based on the notion of active period [38] on a serial line and were validated on a real-case study. This method detects a BT with the highest active period of a resource, and BT is diagnosed with respective active period state contributions. Kumbhar et al. [36] extended Hopp and Spearman [6] utilization method by median estimate of $input\ rate/output\ rate$ implemented on a real-case study. This method detects BT in a complex manufacturing environment. Apart from this, unsupervised machine learning is proposed to detect BT [34]. The active period percentage for a combination of production run and machine is variable for time series. On this time series, agglomerative hierarchical clustering with complete linkage was performed with dynamic time wrapping (DTW) as a similarity measure. The Elbow method is used to choose the number of clusters, and a cluster having the highest active period is detected as a BT cluster, and all machines in that cluster are identified as BT. In addition, data-driven BT detection methods are based on real-time data; however, BT estimate does not capture complete process variation. Due to this, there is no consensus among detected BT. Hence a statistical comparison is essential to assess BT method reliability [10]. Simultaneous statistical tests like Holm and BH were used for more than two BT methods, assuming the independence of BT estimates. BT reliability depends on the type of BT method, statistical significance level, number of batches, minimum accuracy, and batch independence assumption. The author has developed a statistical framework with the above factors using the design of experiment, which revealed a level of significance and independence assumptions as the most influencing factors. In addition, Thürer et al. [42] compared the BT detection method on job and flow shop using maximum workload, utilization, active period, inter-departure time variance, and corrected workload. TH is measured which is based on shorter lead time with mean tardiness and tardy percentage indicators. The design of experiment was conducted with shop type as a blocking factor, and the main effects are BT severity, shifting BT parameter, BT location, and BT method. The multiple comparisons were conducted using the Schaffe test to obtain performance differences. Finally, BT accuracy depends on sample size and type of production system [10], and active period and utilization performed well compared to inter-departure time variance [42].

Due to the availability of real-time data, a process map which shows manufacturing status at a particular time is converted into a dynamic map which is termed as process mining [29]. It consists of process discovery, conformance, and enhancement. Process mining can be used with different perspectives, such as control flow, organization, and time. If resources capture events with the timestamp, then time-perspective process mining helps to analyze the as-is process [36]. In process discovery, an actual process is visualized and defined as-is process model. Any variation between the as-is and designed process model is measured with replay fitness. If it is one, then both models conform; else as-is process model is different from expected. This non-conformed model discrepancy can be detected, such as BT, service level, or TH. A BT is identified where the process takes longer than what expected. To quantify this approach, Lorenz et al. [1] implemented process mining in a real manufacturing scenario. The author uses event-log to build an actual process model and found process discrepancies due to BT. In addition, Leng et al. [43] show how a DT for optimization of warehouse stack assignment and packaging. They explain how to extract event-log from warehouse products consisting of different states, such as the movement of a stacker for searching a material from inventory and moving the goods to a destination with respective timestamps.

Every physical asset is converted into a digital asset in the current digitalization age. One way to make a DT is a digital replica of a physical system with real-time operations. It can perform data analytics, simulation, and optimization for decision-making recommendations [44]. There is no single DT for all problems; however, it depends on the defined objective and scope. For example, a DT consists of a CNC machine to track manufacturing stages for operator interaction [44] or a dedicated machining center to optimize the simulation model for optimum cutter orientation and tool-path for the next steps [45]. For example, Friederich et al. [46] developed an automated manufacturing DT that extracts real-time data and develops a simulation model using process mining Petri-nets for asset reliability. In addition, Lugaresi and Matta [47] developed a DT simulation model with no manual intervention using process mining. Yet, both Friederich et al., Lugaresi and Matta [46,47] DT based on object-centric event-log which needs process-centric event-log for complex product routing. Lastly, a DT have been demonstrated by Bambura et al. [48] on a real production system to increase productivity. However, the authors do not demonstrate sole and shifting BT detection, diagnosis, and improvement on any real-world manufacturing systems.

In past literature, TH improvements are performed for specific resource characteristics without BT analysis. For example, system improvement using online maintenance decisions Li et al., Eun et al. [49,50] focused only on failure and improving system performance. Lorenz et al. [1] suggested reducing non-value added activities and improving cycle time for TH [16]. However, cycle and failure time cannot be reduced continuously, and BT analysis using active and in-active periods is helpful to target specific improvement opportunities [51,52]. Roser et al. [51] tried TH improvement by reducing cycle and idle time. With the dynamic characteristic of BT, Lawrence and Buss [13] show BT improvement may cause shifting BT. Hence, BT analysis for TH improvement is crucial to avoid shifting BT.

The practical value of integrating DTs with descriptive, diagnostic, predictive, and prescriptive analytics in manufacturing has been extensively covered in the literature. However, real-world examples of how such frameworks can be implemented are still needed to evaluate the efficacy and usefulness of DTs in shop-floor operations, specifically BT analysis. Based on a review of existing studies [1,53,54], we have identified the following specific research gaps that need to be explored further:

- How can multiple sources of enterprise data from multiple levels, namely production planning (ERP), process execution (MES), and asset monitoring (SCADA), be integrated to gain a unified view of shop-floor operations in complex manufacturing system?
- How can the integrated data be used to build a DT, without the need for expert knowledge which may not be readily available for complex real-world systems?
- How can the DT be combined with methods for descriptive, diagnostic, predictive, and prescriptive analytics in order to detect, diagnose, and improve BTs in complex manufacturing systems?
- and finally, how can a framework that combines all of the above functional elements be implemented to analyze and improve shop-floor operations in a complex real-world setting?

The above research gaps are addressed through the DT framework proposed in this paper. The underlying layer of a DT is the data collection layer which consists of production planning (ERP), process execution (MES), and asset monitoring (SCADA). On top of this, the core entity creates an as-is process map. This complex map needs abstraction, and DT is created to analyze real-time data. It performs BT detection, analysis, and TH improvement recommendations for decision-making. Finally, the result is presented on a dashboard to analyze the current manufacturing system for improvement opportunities.

3. A digital twin framework

A digital replica of a physical system that uses sensor data for monitoring, controlling, predicting, and performing what-if scenario analysis is called a digital twin (DT) [44]. Leng et al. [55] compare the simulation model and a DT from the perspective of a smart manufacturing system. Any digital model can be categorized as a digital replica, a digital shadow, or a DT. A digital replica, as the name suggests is a digital representation of a physical object in the digital world. A digital shadow approximates a physical system in a mathematical form. Finally, the DT emulates actual functional properties of the lifecycle of a physical system, and can be managed by real-time data coming from a physical system. The authors also provide an extensive literature review on product lifecycle focusing on physical equipment properties such as functional, structural, behavioral, controller, intelligence, and performance level. Across each level, DT models were developed in the industry. Our current study focuses on the integration of intelligence and performance-level DT. In addition, Leng et al. [56] propose an open architecture of a production line to reconfigure physical elements for optimizing operational decisions using an ensemble algorithm. Their proposed methodology is implemented on an assembly line to evaluate system reliability and performance. The system performance is measured with different attributes such as TH, so that BT can be determined and the cycle time can be reduced. However, it is necessary first to find the BT in a complex manufacturing system and determine the cause based on the SCADA system.

A DT could be a piece of equipment, a facility, or a manufacturing environment defined as an observable manufacturing element (OME). For example, in a manufacturing organization, a group of resources forms OME. The framework of a DT is shown in Fig. 2 and consists of entities: physical system, data collection and system control, core, and user. An OME executes operational activities by adding value to incoming raw materials. This OME generates data and is controlled by an operator or automated control through PLCs. Core entities include creating event-log with asset monitoring, DT using process mining, and decision-making knowledge in the form of visualization. In conclusion, a DT is a digital replica that helps in decision-making.

3.1. Observable manufacturing element

A physical system entity is an OME that is a study's focus. It can be a machine, product, personnel, process, environment, or entire physical system. This OME performs process execution on incoming material. The process execution consists of instructions, which generate sensor data with event types. In addition, it interacts with data collection and device control entities.

3.2. Data collection and device control entity

A domain that captures data and controls OME is the data collection and device control entity. It bridges OME working with system control by operating personnel. It includes the following functional elements (FEs):

- **Data Collection FE:** An OME under consideration is controlled and managed by a hierarchy of production planning orders. These orders are in the form of data for OME control and monitoring. These hierarchical orders include planning, execution, and monitoring. First, the planning level allocates resources for a planned order. Subsequently, the allocated resources execute a sequence of activities by adding value to WIP, and processed material follows the next resource which is captured as process execution. Finally, every resource is monitored and controlled using sensors [57].
- **Data Pre-processing FE:** Data generated during operation requires initial data pre-processing like cleaning, integration, and transformation. First, data cleaning involves removing default initialization steps, treating missing instances, and data standardization. Data integration combines and consolidates data coming from the hierarchy of production planning. Finally, data needs adequate variable transformations.
- **Controlling FE:** An OME is controlled by operating personnel or automated control through PLCs. For example, reactive maintenance is performed by maintenance personnel.
- **Actuation FE:** An OME is activated using inherent functioning, which takes input from operating personnel or automated control through PLCs.

3.3. Core entity

In the core entity, BT detection and TH improvement continuously analyzes the production process to detect BT with the help of event-log from a manufacturing system. Depending on the severity of detected BT, possible improvement opportunities are determined and suggested for operator decision-making. A user could visualize results such as resource utilization, asset monitoring contributions for a detected BT, and TH output after a what-if scenario analysis. Therefore, a DT is continuously enhanced by the daily production process and becomes more comprehensive to make any decision on a shop-floor for possible TH improvement. This core entity consists of event-log, DT, asset monitoring, process mining, simulation, and analytics.

- **Event-log FE:** Event-log record OME production execution. For example, an event consists of operation steps performed on a resource, and one step is represented by a case ID. A case ID can trace process execution consisting of event types and timestamps. Typical data is captured as object-centric. If this is true, it needs transformation into process-centric event-log representing one case ID with a sequence of resources. If different case IDs exist, they should be consolidated according to a primary case ID [29]. Aggregation of cases forms event-log. It has additional features like case ID, activity, and timestamp. In addition, data quality checks were performed on event-log, such as data trustworthiness, validation of process execution steps, and coherent semantic structure. In conclusion, these logs help to discover an actual production execution.
- **Process Mining FE:** Process mining consumes the event-log to create process discovery, validate process conformance, and enhance existing processes. The process discovery helps to produce a dynamic process map without prior domain knowledge [29]. Furthermore, this time-perspective process discovers flow routing useful for creating an analytical or a simulation model [2]. Moreover, it helps to generate complex OME routing for validation with the designed system. As a result, it is helpful to enhance the existing system if any discrepancies are observed in the process map.
- **Digital Twin FE:** Process mining generates a dynamic map representing resource dependencies and interaction, which is captured by the manufacturing system. An in-scope target area is selected from the process map to form a DT. As DT works on real-time data, it does not consider any statistical distribution for processing, failure, and repair time. Therefore, a DT represents a digital replica of an OME showing process activities with disturbances.

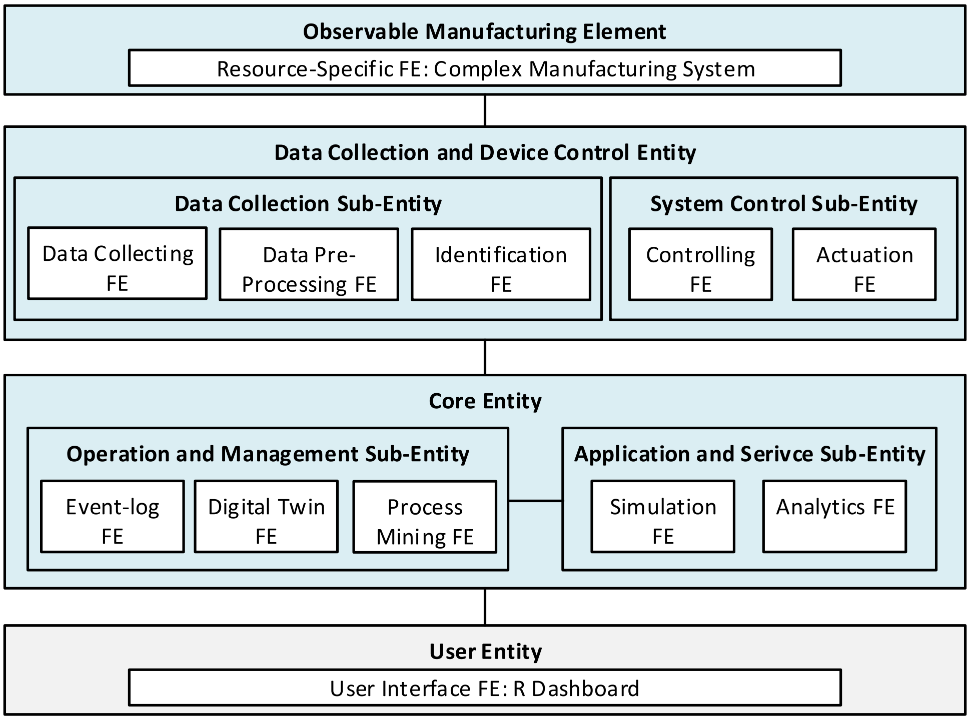

Fig. 2. Digital twin consisting of observable manufacturing system, data collection and device control entity, core entity and user entity.
Source: Adapted from [44].

- **Analytics FE:** The Analytics FE analyzes an OME with the help of a DT. Analytics is performed by a sequence of steps such as describing existing OME, diagnosing current behavior, and decision-making recommendations.

Descriptive Analytics: The descriptive analytics summarizes event-log by estimating the utilization of resources to achieve the desired goal [7]. Utilization of a resource is defined as $u = r/c$, where r is the input rate and c is the capacity of a resource [9,36], which applies to complex flow manufacturing routing. An input rate is calculated as the inter-arrival rate, and the capacity of a resource is determined by processing time. Here, processing time refers to actual process time, including setup, failure, and other possible distractors.

Fig. 1 shows on an activity j, an instance i starts at ST_i start time and completes at ET_i end time with processing time PT_i ($PT_i = ET_i - ST_i$) and inter-arrival time IAT_i ($IAT_i = ST_i - ST_{i-1}$). The utilization of a resource is calculated as a measure of busyness denoted by busy ratio (BR) and is given by:

$$\text{Busy ratio } (BR) = \left(\frac{Input\ Rate}{Output\ Rate}\right) = \left(\frac{\frac{1}{IAT_{i+1}}}{\frac{1}{PT_i}}\right) = \left(\frac{ET_i - ST_i}{ST_{i+1} - ST_i}\right) \tag{1}$$

The resource BR is random and independent, and the ratio is estimated over a given period for all resources. A resource with the highest mean utilization constrains the system's output which is identified as a BT. If one or two resources have the highest mean BR, they need further diagnosing to determine the cause, which is explained in diagnostic analytics.

Diagnostic Analytics: Diagnosing resource BR is determined by monitoring individual resources using the asset monitoring. Typically, a resource that performs operations can stay in different states like processing, failure, repair, setup, waiting for personnel or waiting for material. An effective processing time for a case is obtained by aggregation of processing, failure, repair, and setup time [26]. This effective processing time is called an active period. On the other hand, the inactive period consists of waiting for material and operator, which are primary TH improvement opportunities.

Prescriptive Analytics: Based on the resource diagnosis, the busiest resources are detected and targeted for TH improvement opportunities. These resources were analyzed by active and in-active periods by exploring the diagnostic capability of the data and reducing sequentially in-active period, failure, and cycle time.

3.4. User entity

DT's results are analyzed by a user using the user interface. First, a user can check the data coming from different sources in a manufacturing system. Then, a user can select a period from the uploaded sheet and visualize a descriptive analysis of BR using a box-plot. Box-plot helps in identifying potential BT resources, and a user can choose more than one resource for break-down analysis to see stochastic behavior of BR. Furthermore, BR contributions over a selected period help determine the cause of BR and TH improvement actions.

3.5. An illustrative example

The methodology is verified using a hypothetical serial product mix simulation model consisting of five machines, four buffers, and two sources, as shown in Fig. 3. Machine processing times were assumed to be uniform, and time to failure and repair were kept constant. The simulation was performed for 12 hours in FACTS Analyzer 2.0 [58]. `Source1` generates two products (A and B) at the rate of $2:1$ for the first six hours, after which `Source2` generates products (A and B) at a $1:2$ production rate. The processing times for product variants are different on machines `M2` and `M4`, as shown in Fig. 3.

Table 2 shows the mean values of BR for all machines by source and it includes the result of ANOVA test for significant differences between the means at $\alpha = 0.05$ level of significance. As the p-value of less than $alpha$ indicates, at least one BR mean is significantly different from others. Subsequently, the Tukey–Kramer post-hoc comparison test was

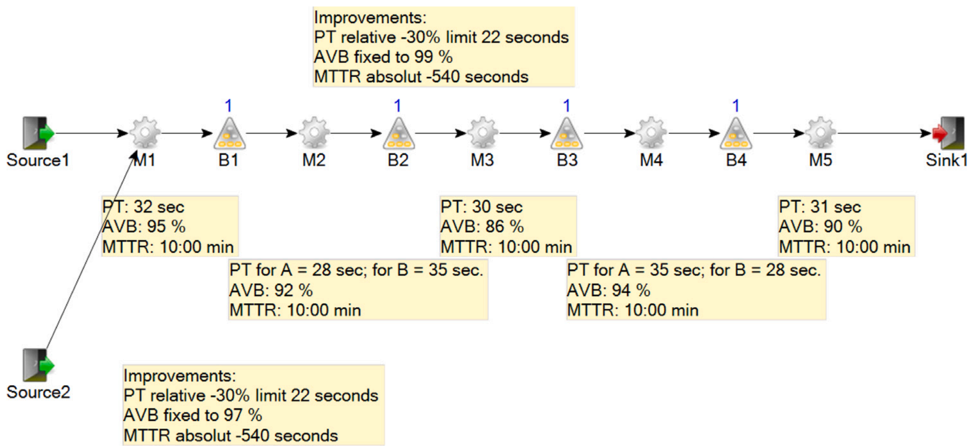

Fig. 3. FACTS Analyzer model for serial flow line with product mix.

Table 2
Mean values of busy ratios (BR) for all machines before and after source change. Bold values indicate the highest observed BR for each phase of the simulation horizon.

Machine	Mean BR (Source1)	Mean BR (Source2)
M1	0.975	0.966
M2	0.929	**0.987**
M3	0.928	0.931
M4	**0.984**	0.931
M5	0.944	0.957
ANOVA Test	0.00*	0.00*

*Indicates significant difference in BR means across cells at 0.05 level of significance.

Table 3
Tukey–Kramer pairwise difference tests between all pairs of BR means.

H_0	Source 1		Source 2	
	t value	p value	t value	p value
M2–M1 ≤ 0	−6.605	1	3.236	0.005*
M3–M1 ≤ 0	−6.757	1	−5.196	1
M4–M1 ≤ 0	1.246	0.522	−5.275	1
M5–M1 ≤ 0	−4.472	1	−1.265	1
M3–M2 ≤ 0	−0.158	0.998	−8.443	1
M4–M2 ≤ 0	7.834	1e−04*	−8.523	1
M5–M2 ≤ 0	2.122	0.124	−4.510	1
M4–M3 ≤ 0	7.984	1e−04*	−0.077	0.995
M5–M3 ≤ 0	2.278	0.087	3.941	0.000*
M5–M4 ≤ 0	−5.704	1	4.019	0.000*

*Indicates significant pair of BR means at 0.05 level of significance.

performed to identify machines with significantly higher BRs. This test was selected to compare the means of unequal samples from machines. The one-tailed higher difference in mean comparison than zero was tested with a 0.05 level of significance.

Table 3 shows the results of the post-hoc analysis, which revealed a statistically significant difference between machine pairs. Interestingly, this table was not only showing sole BT but also detected shifting BT. A significant difference in BR means for `Source 1` having (t = 7.834, t = 7.984) for `M4 - M2` ≤ 0, `M4 - M3` ≤ 0 pair, respectively. This shows that a pair `M4 - M2` was BT pair from the `Source1`, and after comparing with BR mean as seen in Table 2, `M4` was a BT. Similarly, there was significant difference of BR means for `Source 2` having (t = 3.941, t = 4.019, t = 3.236) for `M5 - M3` ≤ 0, `M5 - M4` ≤ 0, `M2 - M1` ≤ 0 pair respectively. From this `Source 2`, pairs `M5 - M3`, `M5 - M4`, and `M2 - M1` were possible BT pairs. After comparing with BR mean in Table 2, machine `M2` had highest BR among other cells. Hence, it can be concluded that BT shifted from machine `M4` to `M2` during the period.

4. Results from a real manufacturing facility

The proposed TH BT detection and improvement methodology was tested and verified in a fully-automated manufacturing facility in Gothenburg, Sweden. The facility manufactures different assembled bearings with in-house and out-source processed components by automated machines, cells, and transport systems. The facility produces in batches with a make-to-stock manufacturing strategy. In the following sections, the DT framework was implemented in the manufacturing facility having OME, data collection and device control, core and user entity.

To adhere to confidentiality requirements with the manufacturing company, the data and product flow were masked during the presentation of the below results.

4.1. Observable manufacturing element

The manufacturing system is consisted of complex components routing having seven operations and two assembly machines, as shown in Fig. 4. Materials to be processed on machines start from a source, transported via AGVs using containers. Next, the containers and AGVs follow a product variant path. Finally, the assembly machine assembled processed components to form FG.

4.2. Data collection and device control entity

The production system generated information through a sequence of activities. This information flow was categorized into enterprise planning, production execution, and process monitoring. At an enterprise level, customer demand is aggregated at a product variant and confirmed dependencies on raw materials and resource availability. Subsequently, an order is released to a manufacturing line. This mapping of resources to a customer demand is logged as an entry into the ERP system. Afterward, the manufacturing line executes an order from ERP to produce individual components for the final product variant. Finally, processed components are combined to produce a finished product with the help of containers and transport systems. These transport activities are recorded into MES. Machine-level states and events are monitored and recorded at the micro-level in SCADA. All these hierarchical data are shown in Table 4 to analyze information flow across the system.

The production order with `BatchId` “10” producing `material` “FG”, is split by `materials` “Part 1”, and “Part 2”. Each individual `material` processes on a different `activity`. Here, activity refers to a resource. Containers are carrying material across `activities`. For example, on `activity` “M4”, input `material` supplied using `ConatinerId` “C155” and processed material leaves via `ContainerId` “C156”.

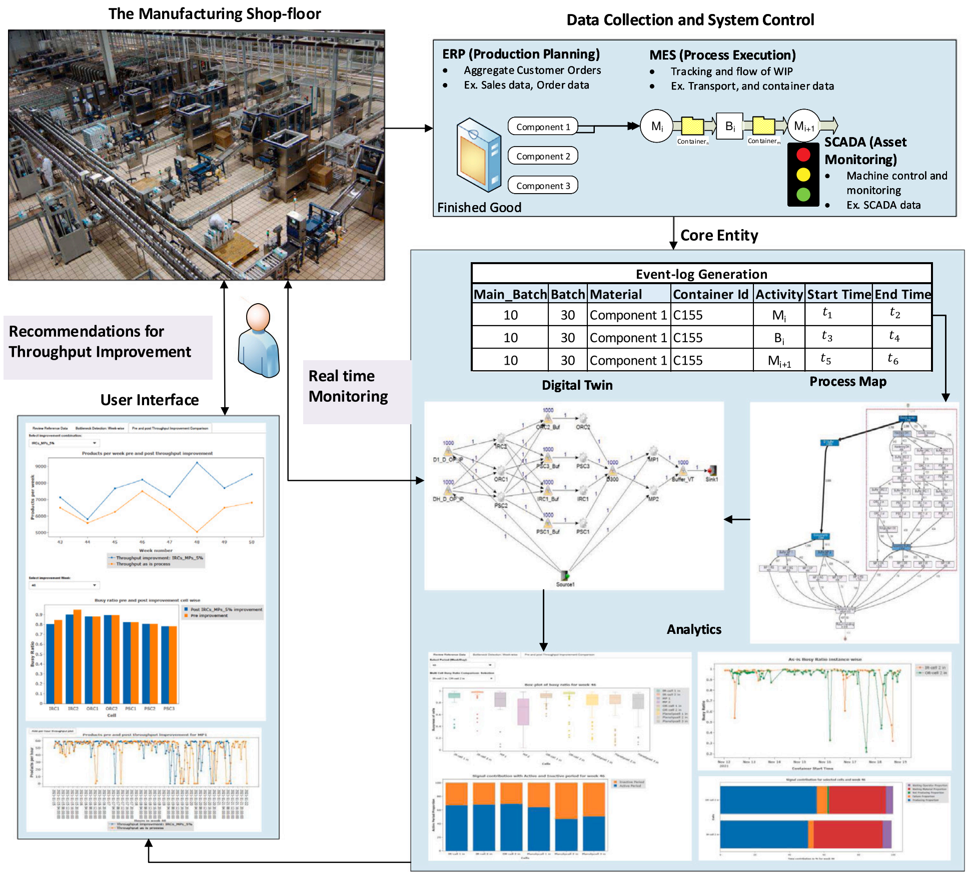

Fig. 4. Digital twin for complex manufacturing system (OME) consisting of data collection and system control, core entity, and user interface entity. The image at the top left is licensed under CC BY 3.0 [59].

Container transfer material from one machine to the next using AGVs. For example, in Table 4 for `Part 1` activity involved are “M1”, “B1”, and “M2” with respective `EventType`. `EventType` describes overall process execution steps like container Id created, assigned, arrived, and left.

Resources are being monitored and recorded in SCADA, as shown in Table 4. It captures combinations of `Light` and `Light Indicator` from an activity. The distinct values of `Light Indicator` and `Lights` are (`Blinking`, `Solid Light`, `No Light`), (`Green`, `Blue`, `Yellow`, `White`, `Stop`, `Stopping`) respectively.

4.3. Core entity

As the name suggests, the core entity is the core of BT detection and TH improvement. It includes the following components as shown in Fig. 4:

Event-log Generation: The ERP and MES data are combined to generate event-log. It can trace components using containers and transport systems. Generated logs are transformed from object-centric (`Batch`) to process-centric (`MainBatch`) to trace a product variant. A `MainBatch` is comprised of many components with `Batch` and `ContainerId` required to form a product variant.

Process Mining: The process-centric logs is given as input to process mining software to create a dynamic process map [29]. The generated process map is shown in Fig. 5, showing the actual product routing in a manufacturing system; however, it is too complex to understand, as every instance of event-log is shown in the process map. Therefore, the process map is analyzed, and a particular area is selected for further analysis. If OME resource paths are stable for an extended period, the abstracted process map remained the same and needs periodic revision.

Table 4
Organizational data across all levels of process execution.

ERP (Production Planning)

Activity	Event Type	Batch Id	Time	Material	Container Id
A1	100	10	t_1	FG	C150
A1	110	10	t_2	FG	C151
M4	100	30	t_3	Part 1	C155
M4	110	30	t_4	Part 1	C156
M5	100	60	t_5	Part 2	C180
M5	110	60	t_6	Part 2	C181

MES (Process Execution)

Activity	Event Type	Container Id	Time	Material
M1	20	C180	t_3	Part1
M1	21	C180	t_4	Part1
B1	20	C180	t_5	Part1
B1	21	C180	t_6	Part1
M2	20	C180	t_7	Part1
M2	21	C180	t_8	Part1

SCADA (Asset Monitoring)

Activity	Light	Light Indicator	Time
M1	●	Blinking	t_3
M1	●	Solid Light	t_4
M1	●	Blinking	t_7
M1	○	No light	t_8

DT: DT is an abstracted dynamic process map representing activities and interconnection among them. It is a digital replica of the actual OME, which continuously analyzing real-time event-log as shown in Fig. 6.

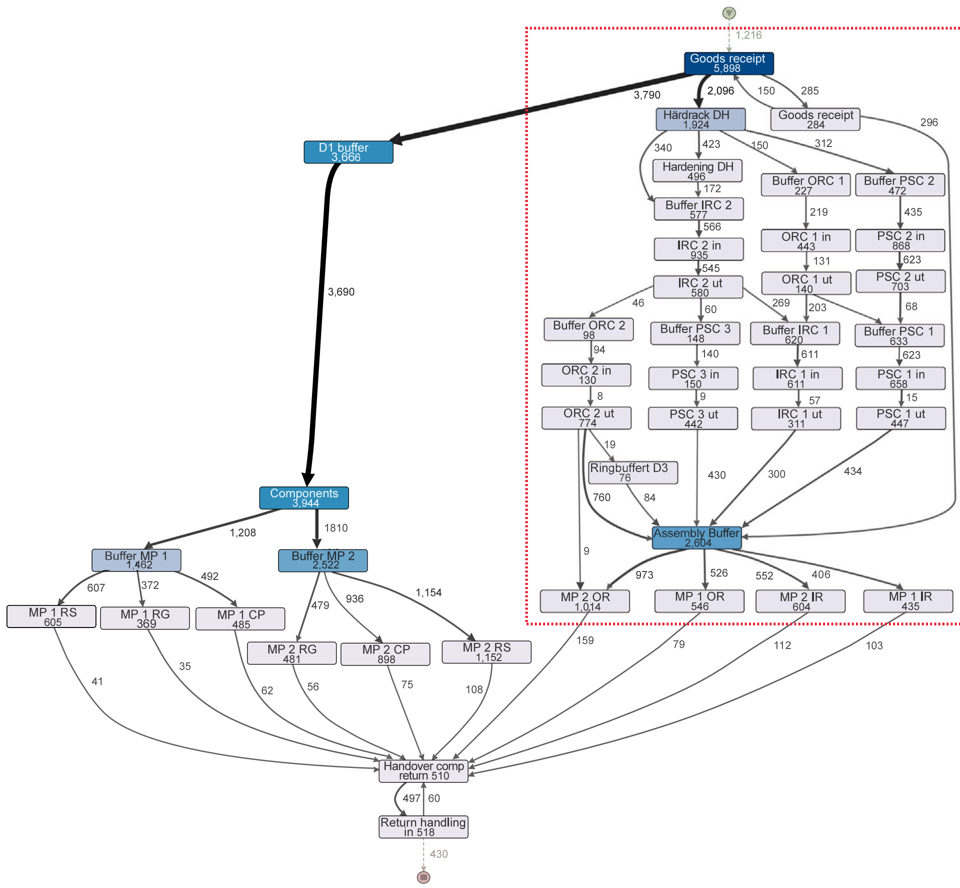

Fig. 5. Process map for manufacturing system generated using event-log using DISCO.

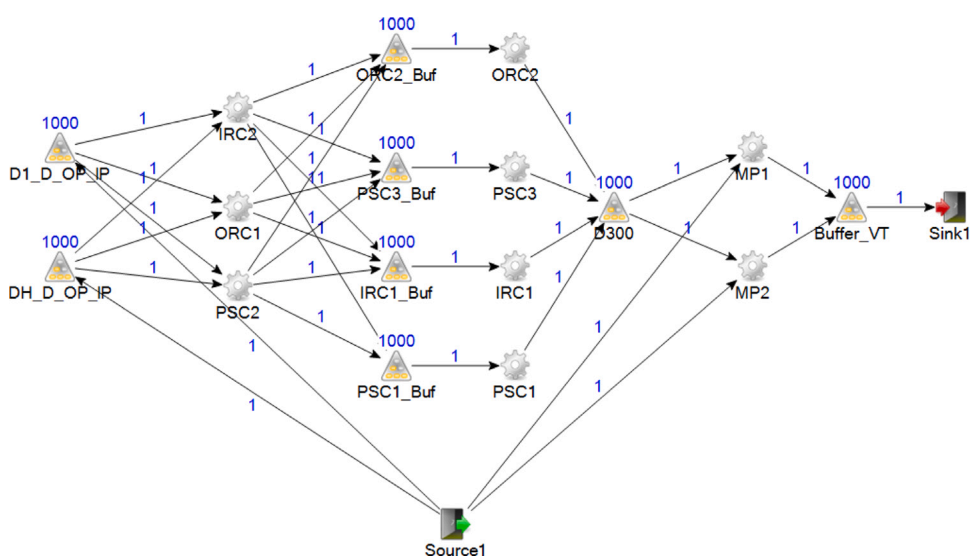

Fig. 6. Digital twin in FACTS software.

Analytics: DT is analyzed using analytics with real-time production. Descriptive, diagnostic, and prescriptive analytics are performed on the data to understand, analyze, and summarize the real-time process.

The event-log passing through DT are summarized by estimating activity utilization. A utilization-based BT detection is implemented on generated event-log. The mean BR for each cell is shown in Table 5. For weeks 43 to 47 cell IRC2 was busiest, and after week 47 cell ORC2 was busiest compared to other cells, which indicates that multiple cells were constraining the system output over the considered period.

Analysis of BR means for multiple cells is performed week-wise using the ANOVA test to measure the equality of mean BR of cells. The test result shows that BR significantly differs cell-wise on a weekly basis. To further analyze which cells are most affected in the ANOVA result, post-hoc analysis is performed to determine a significant cell pair using the Tukey–Kramer test as shown in Table 6. The one-tailed test is

Table 5
Mean values of busy ratios (BR) for all machines in each week. Bold values indicate the highest observed BR within that week.

Cell	Week 43	Week 44	Week 45	Week 46	Week 47	Week 48	Week 49
IRC1	0.89	0.91	0.92	0.84	0.92	0.83	0.95
IRC2	**0.92**	**0.94**	**0.97**	**0.95**	**0.95**	0.94	0.96
ORC1	0.85	0.90	0.93	0.89	0.90	0.82	0.91
ORC2	0.84	0.90	0.95	0.89	0.88	**0.95**	**0.96**
PSC1	0.83	0.81	0.88	0.82	0.80	0.88	0.84
PSC2	0.85	0.80	0.82	0.81	0.72	0.87	0.84
PSC3	0.84	0.80	0.84	0.78	0.79	0.82	0.74
Test ANOVA	0.00*	0.00*	0.00*	0.00*	0.00*	0.00*	0.00*

*Indicates a significant difference in BR means across cells at 0.05 level of significance.

Table 6
Tukey–Kramer pairwise difference tests between all pairs of BR means for weeks 46, 47, and 48.

H_0	Week 46		Week 47		Week 48	
	t value	p value	t value	p value	t value	p value
IRC2–IRC1 ≤ 0	2.88	0.03 *	1.16	0.75	3.53	0.00*
ORC1–IRC1 ≤ 0	1.45	0.59	−0.85	1.00	−0.45	1.00
ORC2–IRC1 ≤ 0	1.63	0.47	−1.47	1.00	4.12	0.00*
PSC1–IRC1 ≤ 0	−0.68	1.00	−4.76	1.00	1.69	0.43
PSC2–IRC1 ≤ 0	−0.98	1.00	−7.34	1.00	1.33	0.65
PSC3–IRC1 ≤ 0	−2.04	1.00	−4.85	1.00	−0.33	1.00
ORC1–IRC2 ≤ 0	−1.79	1.00	−2.18	1.00	−5.70	1.00
ORC2–IRC2 ≤ 0	−1.67	1.00	−2.94	1.00	0.68	0.94
PSC1–IRC2 ≤ 0	−4.02	1.00	−6.38	1.00	−2.95	1.00
PSC2–IRC2 ≤ 0	−4.10	1.00	−8.9	1.00	−3.06	1.00
PSC3–IRC2 ≤ 0	−5.13	1.00	−6.34	1.00	−6.06	1.00
ORC2–ORC1 ≤ 0	−0.18	1.00	−0.64	1.00	6.87	0.00*
PSC1–ORC1 ≤ 0	−2.59	1.00	−4.47	1.00	3.20	0.01*
PSC2–ORC1 ≤ 0	−2.76	1.00	−7.36	1.00	2.50	0.09
PSC3–ORC1 ≤ 0	−3.97	1.00	−4.54	1.00	0.20	1.00
PSC1–ORC2 ≤ 0	−2.87	1.00	−4.35	1.00	−4.00	1.00
PSC2–ORC2 ≤ 0	−3.01	1.00	−7.5	1.00	−3.95	1.00
PSC3–ORC2 ≤ 0	−4.25	1.00	−4.39	1.00	−7.48	1.00
PSC2–PSC1 ≤ 0	−0.42	1.00	−3.77	1.00	−0.43	1.00
PSC3–PSC1 ≤ 0	−1.74	1.00	−0.54	1.00	−3.36	1.00
PSC3–PSC2 ≤ 0	−1.21	1.00	2.95	0.03*	−2.54	1.00

*Indicates significant pair of BR means with 0.05 level of significance.

conducted with $\alpha = 0.05$, considering higher mean BR. On `Weeks 43, 44`, and `45` this test is insignificant and on `Weeks 46, 47`, and `48`, it is significant as shown in Table 6. On `Weeks 46` cell `IRC2 - IRC1` ≤ 0 is significant ($t = 2.88,\ p = 0.03$). With this significant pair, `IRC2` cell is BT with 95% confidence level, as it has the highest BR mean as shown in Table 5. On `Week 47`, `PSC3 - PSC2` ≤ 0 is significant ($t = 2.95,\ p = 0.03$), but as BR mean of `PSC2` and `PSC3` is lower compared to other cells, these cells are not BT. Furthermore, on `Week 48`, there is significant difference between the pair of cells for `ORC2 - IRC1` ≤ 0, `ORC2 - IRC1` ≤ 0, `IRC2 - IRC1` ≤ 0, and `PSC1 - ORC1` ≤ 0 ($t = 6.87,\ p = 0.00$ and $t = 4.12,\ p = 0.00$ and $t = 3.53,\ p = 0.00$ and $t = 3.20,\ p = 0.00$). Of all these significant pairs, from Table 5, BR means of cell `ORC2` is the highest. A BT cell `IRC2` shifted to `ORC2` and is able to detect a shift in BT using utilization-based method.

The detected BT on `Week 46` is diagnosed using light signal data of cells. Before this, SCADA data is mapped with event-log using respective cell, `Start Time` and `End Time`. Apart from SCADA status, "Waiting for material" is derived when the particular cell is waiting for material. For a selected week, effective and ineffective processing times are analyzed. Further exploration of the BR cause is determined by respective contributions such as waiting for material, waiting for an operator, and failure, as shown in Fig. 7.

Fig. 7 shows the BT cell `IRC2` and `ORC2` for `week 46`. The diagnostic analysis of signal data shows that the cell is not producing during its time horizon. The signal diagnosis shows that the cell is waiting for material and operator.

Assembly activity performed on `MP1` and `MP2` assumed availability of all components. Cells are targeted for improvement based on activity BR, and higher BR resources are selected for improvement opportunities. Cells `IRC2`, `MP1` and `MP2` are selected for 5% improvement, which is measured in terms of products per week. It is observed that the BR of cells decreased by the improvement initiatives and is reflected on the products per hour plot coming from `MP1` and `MP2`.

Non-parametric one-tailed Wilcoxon signed-rank test is performed to compare TH differences after improvement. As the p-value 0.0039 ($< \alpha = 0.05$), it can be concluded that there is a statistically significant difference between pre- and post-throughput improvements. In other words, the difference between post- and pre-throughput improvement was greater than zero.

4.3.1. Comparison with active period method

As summarized in Table 1, multiple BT detection methods exist in the literature. Among these, only the active period method has some resemblance to our current approach in terms of how SCADA data (asset monitoring) is processed, which makes a fair comparison possible to some extent. Therefore, we have chosen to only compare these two methods. Other BT detection methods either require sources of data that are currently not available from the manufacturing facility considered in this study, or involve certain assumptions whose real-world effects can only be understood through a more elaborate study. Therefore, we defer comparisons with other methods to a future study.

The proposed utilization-based method is compared with the active period method calculated using the following equation, proposed by Roser et al. [26] and used by Subramaniyan et al. [32].

$$\text{Active Period} = 100 * \left(\frac{\textit{Active Period}}{\textit{Total Period}}\right) \tag{2}$$

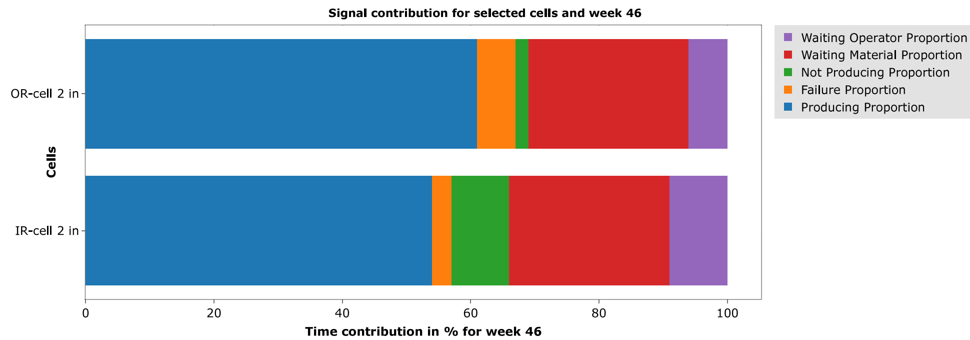

Fig. 7. Diagnostic analytics for the bottleneck resources.

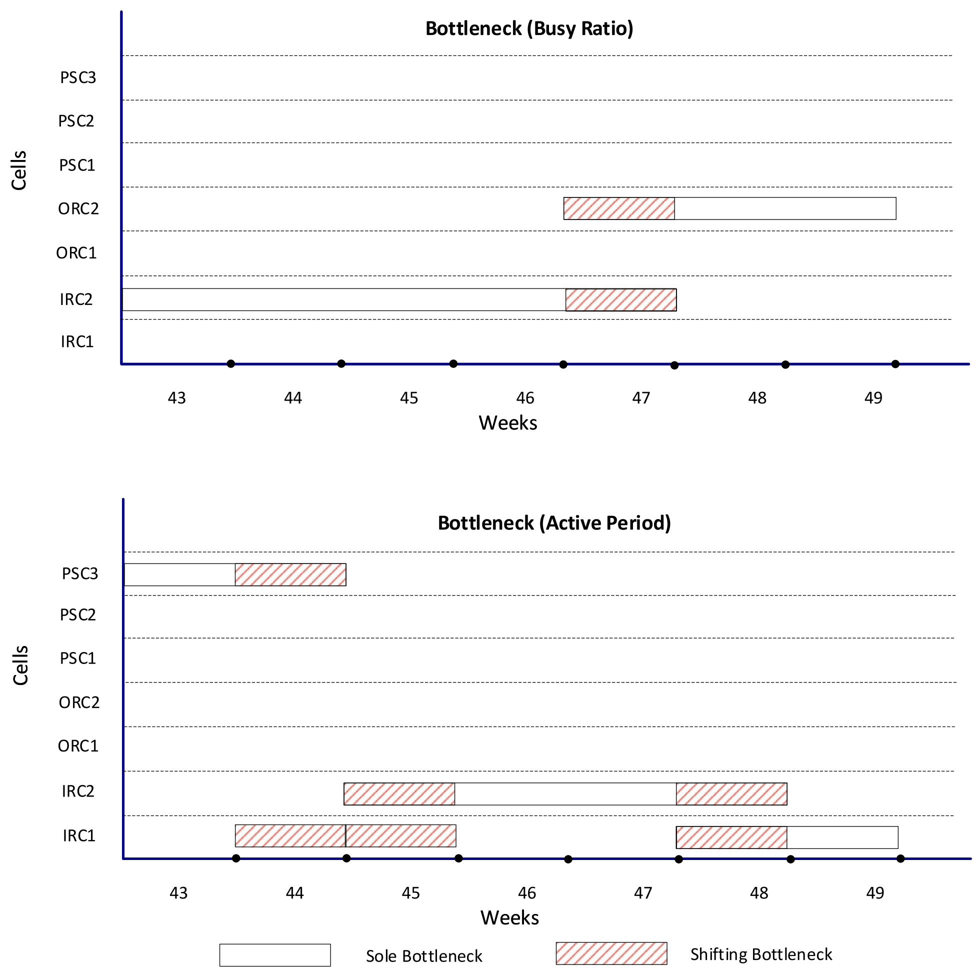

Fig. 8. Sole and shifting bottleneck comparison according to Busy Ratio and Active Period method.

Here, active period refers to 'Producing', 'Failure', and 'Not Producing' categories of the signal data, whereas the total period is the sum of the active and inactive periods. Fig. 8 shows sole and shifting BT weekly for the two methods. It reveals that the active period fluctuates more than the utilization-based BR method. The upper graph shows that according to our approach, Cell IRC2 is first BT, and it shifts to ORC2 during the week 47. The lower graph shows that according to the active period method, the BT shifts across cells PSC3, IRC1, IRC2 and again IRC1. The difference between the two methods can be explained from the fact that the active period method only utilizes SCADA data (asset monitoring), while our proposed approach also incorporates ERP data (production planning) as well as MES data (process execution). Consider a situation when a particular cell has failed, but there are no jobs to be processed on it. Since, the active period method only relies on SCADA data, it sees that cell as a BT, even during periods when it does not need to be operational. This is the reason why BT, as identified by this method, constantly shifts. On the other hand, our utilization-based approach, is capable of levering

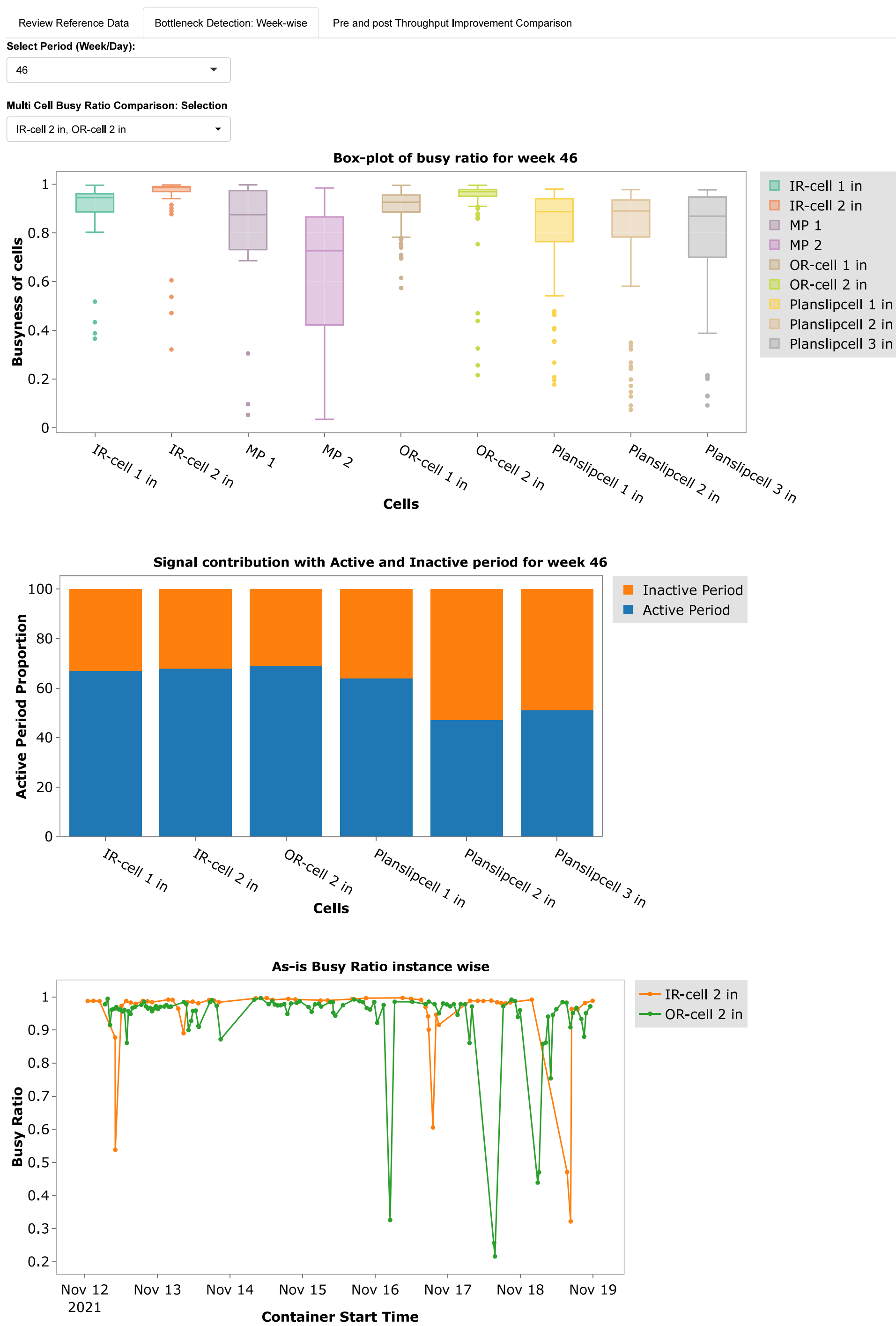

Fig. 9. Dashboard interface showing descriptive analytics for bottleneck detection.

MES data to determine exactly how long the machine is engaged in actual production, which makes it much more reliable in detecting BT. The SCADA data is utilized for diagnosis of the detected BT.

4.4. User entity

An interactive dashboard is developed using the Shiny package in R [60], which takes event-log data in CSV format. The data can be reviewed in the tab `Review Reference Data` as shown in Fig. 9. On the `Bottleneck Detection` tab, a user can select any period (weeks or days) within the uploaded file. BR descriptive analytics are presented as a box plot. If all BR medians are less than one, then it can be said that all cells are not fully busy. The instance-wise BR can be analyzed using `As-is Busy Ratio instance wise`, which shows that BR is fluctuating as indicated by the trend line. For the selected week, diagnostic analysis can be performed to determine state contribution indicating cell BR cause as shown in Fig. 7. Furthermore, after analyzing the cause of BT, `Pre-` and `Post-Throughput Comparison` tab helped to determine TH improvement possibilities as shown in Fig. 10. The selected combination shows a plot for `Products per week`, indicating a significant improvement opportunity. The tab also gives insights about `Busy ratio pre-` and `post-improvement`.

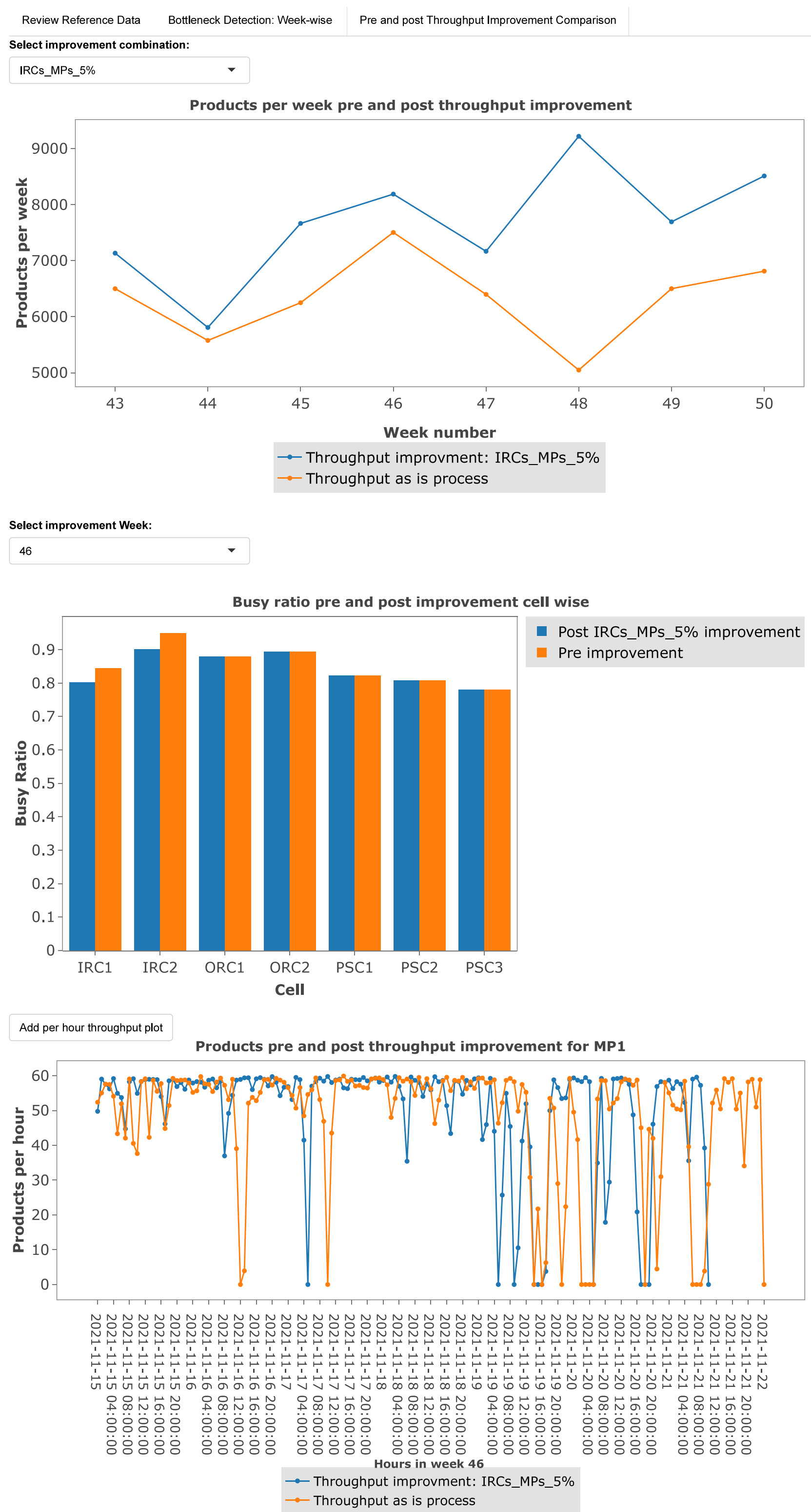

Fig. 10. Dashboard interface showing throughput improvement opportunities for decision-making.

5. Discussions

In this section, we discuss the research contributions and practical implications of our proposed framework.

5.1. Research contributions

This paper contributes to the literature on BT analysis methods by proposing a framework for DT based detection, diagnosis, and improvement of TH BT. This novel approach eliminates the need of expert knowledge for building a simulation model and perform BT analysis. Instead, this paper introduces how existing shop-floor data can be utilized to generate to use the event-log that can in turn be used to build a DT through process mining, thus eliminating the need for manual modeling of complex industrial systems.

With the widespread adoption of digitalization, there is an increasing emphasis on the development on DTs for process and operations throughout the value chain. The approaches proposed in this paper, starting from using shop-floor data, including data processing, integration, model development, and TH improvement, contribute to methodological research on practical implementation of DT based BT analysis. The framework's effectiveness is also demonstrated by applying it to a real application involving complex routing for assembling different variants of ball bearings. The proposed BT analysis approach has provided the company with valuable insights into various improvement opportunities on the shop-floor.

5.2. Implications for industry practitioners

The proposed DT first analyzes event-log which generate an abstract process map. Then, this abstracted dynamic process map is able to detect sole and shifting BT using the utilization-based method, which is further used for the activity's BR diagnosis. Finally, the proposed DT can be used to perform TH improvement experiments, and it is observed that post-improvement activity significantly affects OME TH. From the above, below are implications for industrial practitioners.

- Customers' ever-changing demands are fulfilled by utilizing the existing resources optimally. However, managers need to know which resource impacts the manufacturing system's productivity. Here, BR-based BT detection detects sole and shifting BT impacting a minimum TH improvement of 10% in the existing system, thus encouraging DT adoption, and impacting the bottom line and sustainability. However, more scenarios and experiments are needed to validate the generalizability of the proposed methodology.
- The digital transformation of manufacturing opens new possibilities to utilize the wealth of enterprise information obtained from production planning, process execution, and asset monitoring data. This information into knowledge can be converted using describing, diagnosing, predicting, and optimizing a manufacturing system.
- The use of event-log helps to close the gap between the value of data across all manufacturing resources to make any simulation or DT model. Enterprise, by default, captures object-centric event-log, which needs to convert into process-centric to track all components of an assembled product for the generation of a dynamic process map.
- An end user working with an existing system needs to know a BT resource and its cause. Subsequently, they should take necessary improvement actions based on the cause to increase TH of an existing system

6. Conclusions and future research

Detecting and managing BT is essential for decision-making in a complex manufacturing system. A DT framework is developed for determining BT, analyzing BT, finding TH improvement opportunities, and recommending necessary actions. Importantly, the utilization-based BT method can detect both sole and shifting BT using simple calculations. This framework highlights how a wealth of multi-source data generated by organizational resources be utilized using data merging, feature generation, and detailed analysis on a given objective, helping to bridge the gap between process and data science for productivity and sustainability improvement in organizations. The framework is verified in a real-world application and helps the industry's long-term goal of sustainability. The framework is automated using open-source software, and the dashboard is created for the organization to track, diagnose, and improve the BT resources in real time.

Therefore, understanding process and data science, as well as their interaction, appears rewarding for organizational long-term sustainability objectives. This may be important as further expectations of return on industrial digitalization.

The framework proposed in this paper currently implements descriptive and diagnostic aspects of bottleneck analysis, while also allowing the study of various what-if improvement scenarios. In the future, we intend to incorporate predictive analytics into this framework by implementing machine learning methods that can predict order quantities and failure events. This could potentially allow an early detection of shifting bottlenecks. A further extension involves incorporating prescriptive analytics through the use of multi-objective optimization methods to generate Pareto-optimal solutions containing different sets of improvements. Here, for any desired level of throughput, a decision maker will be able to pick a solution with minimal number of improvements.

Declaration of competing interest

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

Acknowledgments

The authors acknowledge the financial support received from KK-stiftelsen (The Knowledge Foundation, Stockholm, Sweden) for the research project 'TOPAZ - Towards Prescriptive Analytics in Virtual Factories through Structured Data Mining and Optimization' under grant 20200011.

References

[1] Lorenz R, Senoner J, Sihn W, Netland T. Using process mining to improve productivity in make-to-stock manufacturing. Int J Prod Res 2021;59(16):4869–80. http://dx.doi.org/10.1080/00207543.2021.1906460.

[2] Gershwin S. Manufacturing systems engineering. Prentice Hall; 1993.

[3] Xu K, Yang H-D, Zhu C, Jin X, Fan B, Hu L. Deep extreme learning machines based two-phase spatiotemporal modeling for distributed parameters systems. IEEE Trans Ind Inf 2022. http://dx.doi.org/10.1109/TII.2022.3165870.

[4] Goldratt EM, Cox J. The goal: A process of ongoing improvement. Routledge; 2016.

[5] Roser C, Lorentzen K, Deuse J. Reliable shop floor bottleneck detection for flow lines through process and inventory observations: The bottleneck walk. Logist Res 2015;8(1). http://dx.doi.org/10.1007/s12159-015-0127-2.

[6] Hopp WJ, Spearman ML. Factory physics: Foundations of manufacturing management. Waveland Press; 2011.

[7] Lai X, Shui H, Ding D, Ni J. Data-driven dynamic bottleneck detection in complex manufacturing systems. J Manuf Syst 2021;60(July):662–75. http://dx.doi.org/10.1016/j.jmsy.2021.07.016.

[8] Li J, Meerkov SM. Production systems engineering. Springer Science & Business Media; 2008.

[9] Wu K. An examination of variability and its basic properties for a factory. IEEE Trans Semicond Manuf 2005;18(1):214–21. http://dx.doi.org/10.1109/TSM.2004.840525.

[10] Yu C, Matta A. A statistical framework of data-driven bottleneck identification in manufacturing systems. Int J Prod Res 2016;54(21):6317–32. http://dx.doi.org/10.1080/00207543.2015.1126681.

[11] Li L, Chang Q, Ni J. Data driven bottleneck detection of manufacturing systems. Int J Prod Res 2009;47(18):5019–36. http://dx.doi.org/10.1080/00207540701881860.

[12] Kasemset C, Kachitvichyanukul V. Simulation-based procedure for bottleneck identification. In: Asian simulation conference, vol. 5, 2007, p. 46–55. http://dx.doi.org/10.1007/978-3-540-77600-0_6.

[13] Lawrence SR, Buss AH. Shifting production bottlenecks: Causes, cures, and conundrums. Prod Oper Manage 1994;3(1):21–37. http://dx.doi.org/10.1111/j.1937-5956.1994.tb00107.x.

[14] Zhao C, Li J. Analysis and improvement of multi-product assembly systems: An application study at a furniture manufacturing plant. Int J Prod Res 2014;52(21):6399–413. http://dx.doi.org/10.1080/00207543.2014.948576.

[15] Kuo CT, Lim JT, Meerkov SM. Bottlenecks in serial production lines: A system-theoretic approach. Mathematical Problems in Engineering 1996;2:233–76. http://dx.doi.org/10.1155/S1024123X96000348.
[16] Meerkov SM, Zhang L. Product quality inspection in Bernoulli lines: Analysis, bottlenecks, and design. Int J Prod Res 2010;48(16):4745–66. http://dx.doi.org/10.1080/00207540903032874.
[17] Chiang SY, Kuo CT, Meerkov SM. Bottlenecks in Markovian production lines: A systems approach. IEEE Trans Robot Autom 1998;1689(1965):4043–4. http://dx.doi.org/10.1109/70.681256.
[18] Chiang SY, Kuo CT, Meerkov SM. DT-bottlenecks in serial production lines: Theory and application. IEEE Trans Robot Autom 2000;16(5):567–80. http://dx.doi.org/10.1109/70.880806.
[19] Zhao C, Li J, Huang N. Efficient algorithms for analysis and improvement of flexible manufacturing systems. IEEE Trans Autom Sci Eng 2016;13(1):105–21. http://dx.doi.org/10.1109/TASE.2015.2434054.
[20] Zhao D, Tian X, Geng J. A bottleneck detection algorithm for complex product assembly line based on maximum operation capacity. Math Probl Eng 2014;2014(1). http://dx.doi.org/10.1155/2014/258173.
[21] Yan HS, An YW, Shi WW. A new bottleneck detecting approach to productivity improvement of knowledgeable manufacturing system. J Intell Manuf 2010;21(6):665–80. http://dx.doi.org/10.1007/s10845-009-0244-3.
[22] Roser C, Nakano M, Tanaka M. Shifting bottleneck detection. In: Winter simulation conference, vol. 2, 2002, p. 1079–86. http://dx.doi.org/10.1109/WSC.2002.1166360.
[23] Wu K, Zheng M, Shen Y. A generalization of the theory of constraints: Choosing the optimal improvement option with consideration of variability and costs. IISE Trans 2020;52(3):276–87. http://dx.doi.org/10.1080/24725854.2019.1632503.
[24] Sengupta S, Das K, VanTil RP. A new method for bottleneck detection. In: Winter simulation conference. 2008, p. 695–702. http://dx.doi.org/10.1109/WSC.2008.4736261.
[25] Betterton CE, Silver SJ. Detecting bottlenecks in serial production lines - A focus on interdeparture time variance. Int J Prod Res 2012;50(15):4158–74. http://dx.doi.org/10.1080/00207543.2011.596847.
[26] Roser C, Nakano M, Tanaka M. A practical bottleneck detection method. In: Winter simulation conference, vol. 2, 2001, p. 949–53. http://dx.doi.org/10.1109/wsc.2001.977398.
[27] Li L, Chang Q, Ni J, Xiao G, Biller S. Bottleneck detection of manufacturing systems using data driven method. In: IEEE international symposium on assembly and manufacturing. 2007, p. 76–81. http://dx.doi.org/10.1109/isam.2007.4288452.
[28] Leporis M, Zedenka K. A simulation approach to production line bottleneck analysis. In: International conference cybernetics and informatics. 2010, p. 1–10.
[29] Aalst Wvd, Adriansyah A, Medeiros AKAd, Arcieri F, Baier T, Blickle T, et al. Process mining manifesto. In: International conference on business process management. Springer; 2011, p. 169–94. http://dx.doi.org/10.1007/978-3-642-28108-2_19.
[30] Muthiah KM, Huang SH. Overall throughput effectiveness (OTE) metric for factory-level performance monitoring and bottleneck detection. Int J Prod Res 2007;45(20):4753–69. http://dx.doi.org/10.1080/00207540600786731.
[31] Zhang M, Matta A. Data-driven downtime bottleneck detection in open flow lines. In: 2018 IEEE 14th international conference on automation science and engineering. 2018, p. 1513–8. http://dx.doi.org/10.1109/COASE.2018.8560403.
[32] Subramaniyan M, Skoogh A, Salomonsson H, Bangalore P, Gopalakrishnan M, Sheikh Muhammad A. Data-driven algorithm for throughput bottleneck analysis of production systems. Prod Manuf Res 2018;6(1):225–46. http://dx.doi.org/10.1080/21693277.2018.1496491.
[33] Tang H. A new method of bottleneck analysis for manufacturing systems. Manuf Lett 2019;19:21–4. http://dx.doi.org/10.1016/j.mfglet.2019.01.003.
[34] Subramaniyan M, Skoogh A, Muhammad AS, Bokrantz J, Johansson B, Roser C. A generic hierarchical clustering approach for detecting bottlenecks in manufacturing. J Manuf Syst 2020;55(February):143–58. http://dx.doi.org/10.1016/j.jmsy.2020.02.011.
[35] Kahraman MM, Rogers WP, Dessureault S. Bottleneck identification and ranking model for mine operations. Prod Plan Control 2020;31(14):1178–94. http://dx.doi.org/10.1080/09537287.2019.1701231.
[36] Kumbhar M, Ng AH, Bandaru S. Bottleneck detection through data integration, process mining and factory physics-based analytics. In: 10th Swedish production symposium. IOS Press; 2022, p. 737–48, URL https://ebooks.iospress.nl/doi/10.3233/ATDE220192.
[37] Tao F, Zhang H, Liu A, Nee AY. Digital twin in industry: State-of-the-art. IEEE Trans Ind Inf 2019;15(4):2405–15. http://dx.doi.org/10.1109/TII.2018.2873186.
[38] Roser C, Nakano M, Tanaka M. Comparison of bottleneck detection methods for AGV systems. In: Winter simulation conference, vol. 2, 2003, p. 1192–8. http://dx.doi.org/10.1109/wsc.2003.1261549.
[39] Li L. Bottleneck detection of complex manufacturing systems using a data-driven method. Int J Prod Res 2009;47(24):6929–40. http://dx.doi.org/10.1080/00207540802427894.
[40] Li L. A systematic-theoretic analysis of data-driven throughput bottleneck detection of production systems. J Manuf Syst 2018;47:43–52. http://dx.doi.org/10.1016/j.jmsy.2018.03.001.
[41] Subramaniyan M, Skoogh A, Gopalakrishnan M, Salomonsson H, Hanna A, Lämkull D. An algorithm for data-driven shifting bottleneck detection. Cogent Eng 2016;3(1). http://dx.doi.org/10.1080/23311916.2016.1239516.
[42] Thürer M, Ma L, Stevenson M, Roser C. Bottleneck detection in high-variety make-to-Order shops with complex routings: An assessment by simulation. Prod Plan Control 2021;1–12. http://dx.doi.org/10.1080/09537287.2021.1885795.
[43] Leng J, Yan D, Liu Q, Zhang H, Zhao G, Wei L, Zhang D, Yu A, Chen X. Digital twin-driven joint optimisation of packing and storage assignment in large-scale automated high-rise warehouse product-service system. Int J Comput Integr Manuf 2021;34(7–8):783–800. http://dx.doi.org/10.1080/0951192X.2019.1667032.
[44] Shao G. Use case scenarios for digital twin implementation based on ISO 23247. J Res NIST 2021. http://dx.doi.org/10.6028/NIST.AMS.400-2.
[45] Zhu Z, Xi X, Xu X, Cai Y. Digital twin-driven machining process for thin-walled part manufacturing. J Manuf Syst 2021;59(December 2020):453–66. http://dx.doi.org/10.1016/j.jmsy.2021.03.015.
[46] Friederich J, Francis DP, Lazarova-Molnar S, Mohamed N. A framework for data-driven digital twins for smart manufacturing. Comput Ind 2022;136:103586. http://dx.doi.org/10.1016/j.compind.2021.103586.
[47] Lugaresi G, Matta A. Automated manufacturing system discovery and digital twin generation. J Manuf Syst 2021;59(January):51–66. http://dx.doi.org/10.1016/j.jmsy.2021.01.005.
[48] Bambura R, Šolc M, Dado M, Kotek L. Implementation of digital twin for engine block manufacturing processes. Appl Sci 2020;10(18):6578. http://dx.doi.org/10.3390/app10186578.
[49] Li L, Ambani S, Ni J. Plant-level maintenance decision support system for throughput improvement. Int J Prod Res 2009;47(24):7047–61. http://dx.doi.org/10.1080/00207540802375705.
[50] Eun Y, Liu K, Meerkov SM. Production systems with cycle overrun: Modelling, analysis, improvability and bottlenecks. Int J Prod Res 2021. http://dx.doi.org/10.1080/00207543.2021.1968528.
[51] Roser C, Nakano M, Tanaka M. Throughput sensitivity analysis using a single simulation. In: Winter simulation conference, vol. 2, 2002, p. 1087–94. http://dx.doi.org/10.1109/wsc.2002.1166361.
[52] Subramaniyan M, Skoogh A, Salomonsson H, Bangalore P, Bokrantz J. A data-driven algorithm to predict throughput bottlenecks in a production system based on active periods of the machines. Comput Ind Eng 2018;125:533–44. http://dx.doi.org/10.1016/j.cie.2018.04.024.
[53] Subramaniyan M, Skoogh A, Bokrantz J, Sheikh MA, Thürer M, Chang Q. Artificial intelligence for throughput bottleneck analysis – State-of-the-art and future directions. J Manuf Syst 2021;60(August):734–51. http://dx.doi.org/10.1016/j.jmsy.2021.07.021.
[54] Mahmoodi E, Fathi M, Ghobakhloo M. The impact of industry 4.0 on bottleneck analysis in production and manufacturing: Current trends and future perspectives. Comput Ind Eng 2022;108801. http://dx.doi.org/10.1016/j.cie.2022.108801.
[55] Leng J, Wang D, Shen W, Li X, Liu Q, Chen X. Digital twins-based smart manufacturing system design in industry 4.0: A review. J Manuf Syst 2021;60:119–37. http://dx.doi.org/10.1016/j.aei.2022.101676.
[56] Leng J, Chen Z, Sha W, Lin Z, Lin J, Liu Q. Digital twins-based flexible operating of open architecture production line for individualized manufacturing. Adv Eng Inform 2022;53:101676. http://dx.doi.org/10.1016/j.aei.2022.101676.
[57] De Ugarte BS, Artiba A, Pellerin R. Manufacturing execution system - A literature review. Prod Plan Control 2009;20(6):525–39. http://dx.doi.org/10.1080/09537280902938613.
[58] Ng AHC, Bernedixen J, Moris MU, Jägstam M. Factory flow design and analysis using internet-enabled simulation-based optimization and automatic model generation. In: Winter simulation conference, (no. 3):2019, p. 2181–93. http://dx.doi.org/10.1109/WSC.2011.6147930.
[59] Tittenberger P. Mengniu production line. 2009, URL https://commons.wikimedia.org/wiki/File:Mengniu-production-line.jpg.
[60] Chang W, Cheng J, Allaire J, Sievert C, Schloerke B, Xie Y, et al. Shiny: Web application framework for R. 2021, R package version 1.7.1, URL https://CRAN.R-project.org/package=shiny.