



Identification of Delays and Bottlenecks in Manufacturing Processes Through Process Mining

Safiye Turgay^{*}, Alperen Arif Demir[†], Özlem Eryürür[†]

Industrial Engineering, Sakarya University, 54187 Sakarya, Turkey

^{*} Correspondence: Safiye Turgay (sencer@sakarya.edu.tr)

Received: 04-27-2025

Revised: 05-30-2025

Accepted: 06-12-2025

Citation: S. Turgay, A. A. Demir and O. Eryürür, "Identification of delays and bottlenecks in manufacturing processes Through process mining," *J. Ind Intell.*, vol. 3, no. 2, pp. 69–87, 2025. <https://doi.org/10.56578/jii030202>.



© 2025 by the author(s). Licensee Acadlore Publishing Services Limited, Hong Kong. This article can be downloaded for free, and reused and quoted with a citation of the original published version, under the CC BY 4.0 license.

Abstract: In the highly competitive landscape of modern manufacturing, the efficient and timely operation of production processes is paramount for sustaining productivity and ensuring customer satisfaction. Delays and latent bottlenecks, however, often hinder optimal performance. A data-driven methodology for identifying these inefficiencies is presented, employing process mining techniques. By analyzing event logs from Manufacturing Execution Systems (MES) and Enterprise Resource Planning (ERP) systems, the actual execution flow of production processes is reconstructed and compared against the designed process models. Through process discovery, conformance checking, and performance analysis, the underlying causes of delays and capacity bottlenecks are pinpointed. A case study from a manufacturing facility is used to demonstrate the effectiveness of process mining in uncovering critical areas for process improvement. The findings indicate that process mining not only enhances transparency but also provides actionable insights for optimizing resource planning, reducing cycle times, and maximizing overall operational effectiveness. The approach is demonstrated to facilitate the identification of inefficiencies, leading to targeted interventions that significantly improve process performance and business outcomes.

Keywords: Process mining; Production processes; Bottleneck detection; Delay analysis; Event logs; Process optimization; Conformance checking; Operational efficiency

1 Introduction

Industry 4.0 is revolutionizing production systems to become increasingly sophisticated, networked, and data-intensive. Subsequent manufacturing environments have steadily depended on advanced digital infrastructures like Manufacturing Execution Systems (MES), Enterprise Resource Planning (ERP) systems, and Internet of Things (IoT) sensors to capture and monitor vast quantities of operations data. Although they operate such production systems despite widespread problems such as process delay, bottlenecks, and inefficient resource utilization, resulting in adverse effects on productivity, lead times, and product quality. Eliminating those wastages is thus the motive for seeking operational excellence and sustainable manufacturing performance.

Traditional methods of operating performance analysis for production are normally based on manual observation, simulation models, or static KPIs. While such methods could give visibility to some degree, they are normally not able to track the actual and dynamic behavior of the observed shop floor production system. In attempting to address such limitations, process mining is currently a strong analysis methodology to bridge the gap between model-based analysis and fact-based decision-making. Process mining algorithms use automated event logs produced by automated information systems in order to reimplement, visualise, and analyze the real process flows of an organisation.

Since today's businesses have such a high rate of competition, companies are under more pressure to enhance performance, reduce lead times, and receive goods on time. Planning alone will not lead to these results; there has to be a sharp awareness of the flow of production processes in real time. Advanced processes, irregularities in shop floor operations, and many resources involved, though, most of the time, produce hidden bottlenecks and delays and are not easily visible through typical monitoring media. It is possible to make the hidden patterns, deviations, and inefficiencies observable through process mining techniques that cannot be unraveled using regular performance-monitoring tools.

Process mining applied in production management can identify bottlenecks, waiting time, rework loops, and skewly loaded resources and present them in a graphical representation of how the processes actually function under actual operating conditions. As an ancillary capability, the incorporation of process mining output with production scheduling and workflow optimization methods provides space for fact-based continuous improvement and decision-making.

The study tries to uncover the delay in manufacturing and the bottlenecks of the manufacturing process using process mining tools. Event logs of MES and ERP systems are mined such that beneficial performance measurements are derived and where operations stray from the desired workflow areas are established. This research contributes to the knowledge base in that it illustrates the potential of process mining as a diagnostic tool for enhancing manufacturing system performance. The case study is illustrated to confirm the method and show the way through which process knowledge achieved through data can result in quantifiable throughput, resource utilization, and cycle time reduction improvement.

Delays are excessive deviations from anticipated waiting or processing time, and bottlenecks are locations where flow is being slowed down or hindered. Both are upper-level measures that determine cycle time, throughput, and productivity as a whole. Early relief can achieve dramatic performance improvement. The rest of this paper is structured as below: Section 2 gives an overview of existing related work; Section 3 outlines the method employed for event log preparation and process analysis; Section 4 outlines the case study and results; Section 5 gives implications and potential improvements; and Section 6 concludes this research with future work recommendations.

2 Literature Survey

Process mining is a data-driven approach to extract process knowledge from event logs of information systems (ERP, MES, WMS, PLM, etc.) for discovering, monitoring and improving real-life processes. Process-mining software is being utilized in the production and manufacturing processes for revealing the as-is process flow, quantifying performance (cycle time, waiting time, throughput), detecting deviations from expected behavior, and pinpointing where delays and bottlenecks occur [1, 2].

This questionnaire is concerned with research and application efforts on employing process mining for (a) identifying production delays and bottlenecks, (b) quantifying their impact, and (c) proposing mitigation strategies—including combinations with predictive analytics, simulation and factory-physics models.

Process mining is generally structured into three tasks: (1) Discovery—distill a process model from event logs; (2) Conformance checking—compare observed executions to a model; (3) Enhancement—enrich models with performance data to inform improvement. These are the primitives for identifying slow or bottlenecking activities in production. The most important performance measures for bottleneck analysis are cycle time, processing time, waiting time, resource utilization, and throughput [3, 4].

A classical definition used across the literature is to assume that a bottleneck is any subprocess, activity or resource whose finite capacity or long waiting/processing times constrains overall throughput or leads to unnecessarily long lead times. Several authors mention that bottleneck detection is different from root-cause analysis and prediction—these last two require additional techniques on top of standard process mining [5–7].

Current taxonomy research and reviews identify that methods of bottleneck and delay detection in production processes can be divided into several complementary categories with respective strengths. Discovery or descriptive techniques are the most common starting point in manufacturing environments, wherein aggregated performance overlays—cycle time heatmaps, per-activity throughput times, and variant analysis, for instance—are utilized to graphically identify slow process activities and case variants that disproportionately contribute to lead time [8, 9]. Metric and statistical methods go further and compute quantitative metrics like mean and percentile cycle times, waiting-time distributions, and resource utilization, usually supported by statistical tests or control charts; recent articles propose augmenting Statistical Process Control (SPC) with process mining for increased detection of performance deterioration.

Informal techniques are part of algorithmic bottleneck detection, where graph-based analysis, i.e., longest-path or critical-path analysis, and token-flow tracing are carried out on process models to identify structural as well as resource-related hotspots; Bemthuis et al. provide a popular classification of such techniques. On top of backward-looking analysis, predictive and prescriptive extensions use machine learning on event logs to forecast delays and future bottlenecks, sometimes combined with recommendation systems that suggest capacity reallocation or schedule adjustments. Hybrid methods integrate process mining findings with discrete-event simulation, agent-based modeling, or factory-physics analysis to model system-wide, nonlinear effects of interventions such as buffer-sizing adjustments or capacity adjustments. This taxonomy illustrates a mature discipline ranging from descriptive visualization to prescriptive, system-level decision support, and is thus strongly applicable to complex manufacturing environments [10, 11].

Review papers and case collections that are manufacturing-focused recognize the use of process mining to tackle manufacturing issues: estimation of processing time, identification of bottlenecks, variant analysis, and service-level

monitoring in make-to-order and assembly environments [12, 13]. The reviews reflect emphasis on real ERP/MES event logs and industrial case studies. Empirical models and case studies outline end-to-end frameworks: event-log extraction from shop-floor systems, preprocessing and enrichment (e.g., event-resource and event-workstation linking), process model discovery, overlaying performance metrics, and root cause follow-up via correlation and drill-down analytics. Such frameworks imply explicit integration with SPC and simulation for end-to-end bottleneck analysis.

Data-integration + factory physics studies reveal that combining event logs and physical production measurements creates more useful bottleneck diagnostics, especially in the situation where capacity, batch sizes, and failure rates need to be dealt with simultaneously [14].

Process mining within manufacturing environments is typically faced with day-to-day pragmatic issues directly affecting the accuracy and validity of bottleneck analysis. Completeness and fineness of the event log are likely to be the most common issues, in that shop-floor systems fail to log enough detailed data—for example, without implicit start and end timestamps for actions—so analysts have to infer missing detail or merge data from multiple sources. Timestamp and noise synchronisation is another complication source, in the form that uncoordinated system clocks and truncated event traces make it difficult to correctly compute waiting times [15]. Case and activity definition is a second chief problem: how a case is constructed as an individual item, a batch, or an entire order, and how activities are followed at the machine-operation level or at higher-level process steps, alters the bottlenecks revealed. Concurrently, resource attribution—i.e., correlating events with offending operators or machines—is critical for resource contention analysis but unavailable or unknown in production data. To address these challenges, methodology papers all emphasize the importance of thorough preprocessing, event enhancement, and verification, e.g., cross-validation against subject matter experts, prior to making conclusions or taking corrective action [16].

Literature highlights certain focal gaps, unresolved issues, and research directions impacting the future of process mining on manufacturing system delay and bottleneck analysis [17, 18]. Although the new techniques have come closer to bottleneck detection, the journey towards automated remediation is still not so mature; though techniques such as Bottleneck Detection, Prediction, and Recommendation (BDPR) have proposed end-to-end pipelines from detection to actionable recommendation, empirical validations over actual manufacturing environments are scarce. Scalability is a problem too because existing production generates large and heterogeneous event logs, and hence streaming-capable process mining as well as computationally more effective algorithms for multi-event system handling are needed [19, 20]. Further integration of process mining theory with traditional production theories, i.e., factory-physics equations and queuing theory, is also necessary for system-level effect estimation if a bottleneck is eliminated [21–23]. It is also difficult to control variability and uncertainty because it means anticipation of future delay under mixed demand patterns and sporadic disruptions, and this requires hybrid methods combining machine learning and domain knowledge. Finally, most bottlenecks arise not only because of technical or operational inefficiencies but also because of human behavior, company policy, and supplier constraints—factors not typically captured in event logs. Thus, mixed-method study designs that integrate quantitative process mining with qualitative organizational understanding are required to drive the field towards more integrative and pragmatic solutions [24–26].

Practitioners and researchers in practice would welcome a step-by-step guide to implementing process mining in identifying delays and bottlenecks in manufacturing systems. The most significant and the first one is to provide high-quality event logs by designing and preprocessing them properly with well-defined case identifiers, valid start and end timestamps, and resource labels, which have to be verified for correctness and completeness with the assistance of domain experts [27–30]. With data foundation, process discovery, and metric overlays established—specifically, cycle time heatmaps, percentile comparison, and variant comparison—problems may be easily and quickly identified. Pre-screening of potential interventions against simulation models or factory-physics analytics before remedial interventions may be best to establish the effect of change enterprise-wide without impacting manufacturing. Finally, integration with planning models allows organizations to anticipate in advance and act in advance to incoming congestion. Backward tracing recommended activities and effects creates a closed-loop system for feedback to happen, resulting in ongoing improvement and improved production protocols [31–33].

Process mining is a good empirically supported tool for demonstrating delays and bottlenecks in production processes. Literature is comprised of a rich descriptive and algorithmic reservoir of detection methods complemented by research on predictive/prescriptive extensions, and an exaggerated emphasis on hybrid designs that involve event-log analysis and simulation and factory-physics to try to provide system-level effect prediction [34–36]. Opportunity areas for challenges are data quality, scalability, human and organizational integration, and moving towards tried and tested automated remediation techniques—all fertile grounds for future application and study.

3 Methodology

This section explains the rigorous process for finding delays and bottlenecks in manufacturing processes using process mining techniques' steps step by step. The process includes a sequence of critical steps: data collection, event log construction, process discovery, conformance checking, performance analysis, bottleneck identification,

and improvement recommendations. It begins with data collection, where detailed event data are harvested from systems like MES, ERP systems, and IoT sensors. At least case ID, activity name, timestamp, resource details, and other fields like product type or quality status are the minimum fields for every event gathered.

Preparation and preprocessing of the event log occur at collection time in order to exclude redundant data, keep the events in order of occurrence, and reformat so that it will be friendly to a process mining tool (e.g., XES or CSV), and, if necessary, include derived measures such as waiting times. In process discovery, tools such as ProM, Disco, or Celonis are utilized to build a visual representation of the real process occurrence. Detection of all key observed flow paths, including parallel flows, loops, and deviations from typical behavior, is accomplished by using the algorithms α -algorithm, Heuristics Miner, or Fuzzy Miner.

Conformance checking is subsequently executed in comparison to a reference model—most often a designed workflow or standard operating procedure (SOP)—to identify deviations such as lacking steps, rework cycles, or inappropriate timing activities. Measures are put in place to quantify the degree of alignment and to highlight common reasons for non-conformity and delays.

Second, it is necessary to diagnose efficiency through performance measurement by calculating key measures like waiting time and processing time, throughput, as well as cycle time. Such an activity or resource with an unusually long time, excessive idleness, or probable bottlenecks can be identified using visualization tools like heat maps and timelines. In doing so, it is then possible to identify delays and bottlenecks more precisely by where the production stream is lagging behind, which queues are accumulated, and which transitions of actions have procrastinated waiting times. Such data is also justified by comparing such problems to their offenders, such as resource shortages, quality deficits, or scheduling problems.

Finally, solution and improvement plans are established in the context of findings. These may involve resource deployment, process re-engineering, or scheduling adjustments. Such interventions can be validated by iterative process mining loops in a paradigm to continuously enhance. Furthermore, optionally, predictive analytics can be utilized for future delay prediction and avoidance to make the manufacturing system adaptive and resilient, as in Figure 1.

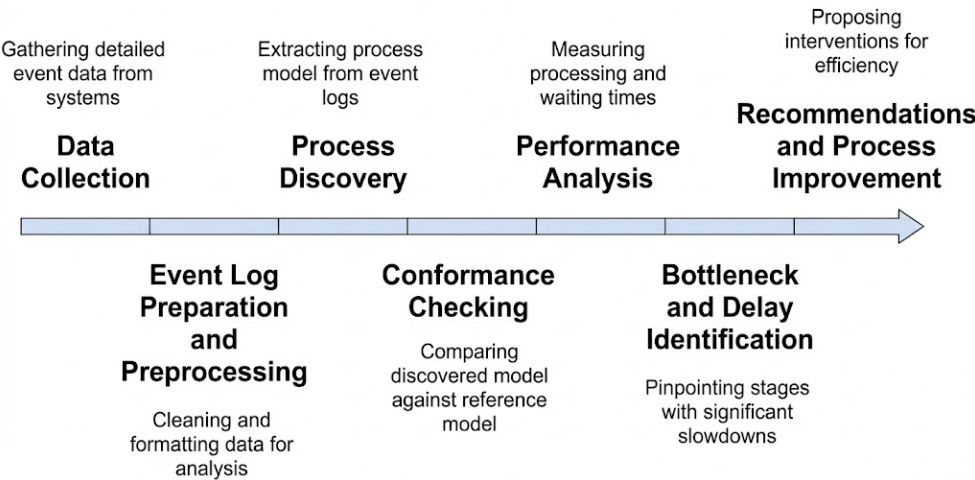


Figure 1. Process improvement strategies

While classical process mining techniques perform superbly in detecting delays and bottlenecks via analysis of past event logs, they are incapable of carrying out the same efficiently in large, complex, and real-time production environments. In an attempt to address such limitations, this research presents a novel hybrid technique that combines process mining, real-time data analysis, and machine learning to significantly enhance operational inefficiency detection as well as forecasting. One of the most striking aspects of this solution is leveraging real-time streaming data instead of conventionally depending on static event logs from ERP or MES systems. With the inclusion of IoT sensor data and machine-based capture mechanisms, the system furnishes real-time updates to event logs so that near real-time monitoring of processes is possible and bottlenecks and delays are recognizable sooner.

Besides the above, in the research it is also proposed that the classical algorithms, such as α -algorithm or Heuristics Miner, may be merged with the pattern and clustering identification of machine learning. This blending process enables the identification of small variations from a process and latent bottlenecks that could be due to outlier or exceptional cases. In order to further measure the effect of bottlenecks, a dynamic bottleneck scoring model (DBSM) is created that measures various performance indicators—waiting time, queue size, resource usage, and frequency of deviation—to rank problems based on the effect they have on throughput and lead time. Besides,

supervised machine learning models such as Random Forest and Gradient Boosting are also used for predictive delay analysis that predicts potential delays beforehand. Such forecast ability enables the production managers to respond ahead of time, i.e., re-schedule or re-assign resources, and thus make operations more resilient. Finally, there is loop closure for continuous improvement in that findings consolidated through process mining and predictive analytics are utilized to feed continuous process tuning. Such improvements, spanning from resource planning to process redesign, are re-evaluated in future mining iterations to make improvement responsive and evidence-based with changing production dynamics in Figure 2 and Table 1.



Figure 2. Process mining techniques range from static to adaptive

Table 1. Summary of innovations

Feature	Traditional Approach	Proposed New Approach
Data Type	Periodic, static event logs	Real-time streaming data
Process Discovery	Classic algorithms (α , Heuristics Miner)	Hybrid ML-based clustering and anomaly detection
Bottleneck Identification	Based on simple duration thresholds	Multi-criteria bottleneck scoring model
Delay Handling	Reactive, post-event analysis	Predictive analytics for proactive delay mitigation
Process Improvement Cycle	Periodic manual updates	Continuous feedback loop with adaptive interventions

This novel approach is created to bring agility and forecastability to production process analysis, which makes it especially useful for Industry 4.0 and intelligent manufacturing environments.

3.1 DBSM

This section gives fundamental concepts and theoretical attributes of the suggested DBSM for delay and bottleneck discovery in production processes with respect to process mining data.

Let $P = \{p_1, p_2, \dots, p_n\}$ be the set of process steps or resources in a production system. Each process step p_i has four normalized performance metrics:

- $WTS_i \in [0, 1]$: Waiting Time Score
- $QLS_i \in [0, 1]$: Queue Length Score
- $RUS_i \in [0, 1]$: Resource Utilization Score
- $DFS_i \in [0, 1]$: Deviation Frequency Score

The bottleneck score BS_i of step p_i is defined by:

$$BS_i = \sum_{j=1}^4 w_j m_{ij} = w_1 WTS_i + w_2 QLS_i + w_3 RUS_i + w_4 DFS_i \quad (1)$$

where, weights $w_j \geq 0$ are such that:

$$\sum_{j=1}^4 w_j = 1 \quad (2)$$

and m_{ij} is the j -th metric value for step i .

The scoring function:

$$BS : [0, 1]^4 \rightarrow [0, 1] \quad (3)$$

maps a 4-dimensional normalized metric vector to a scalar bottleneck score using a weighted linear functional.

Linearity: For all $x, y \in [0, 1]^4$ and non-negative scalars α, β ,

$$BS(\alpha x + \beta y) = \alpha BS(x) + \beta BS(y) \quad (4)$$

Boundedness: Each metric in $[0, 1]$, weights with sum 1,

$$0 \leq BS(x) \leq 1 \quad (5)$$

If for two process steps p_i and p_k , $m_{ij} \leq m_{kj}$ for all $j = 1, \dots, 4$, then

$$BS_i \leq BS_k \quad (6)$$

As weights $w_j \geq 0$,

$$BS_i = \sum_{j=1}^4 w_j m_{ij} \leq \sum_{j=1}^4 w_j m_{kj} = BS_k \quad (7)$$

Thus, BS preserves partial ordering induced by component-wise comparison. The DBSM score is a convex combination of normalized scores. The bottleneck score is thus in the convex hull of the metric vectors:

$$BS_i \in \text{ConvexHull}\{0, m_{i1}, m_{i2}, m_{i3}, m_{i4}\} \quad (8)$$

The sensitivity of BS_i to the individual metric m_{ij} depends on the assigned weight w_j . Changing w_j allows prioritizing specific delay dimensions. Since each $m_{ij} \in [0, 1]$ and $\sum w_j = 1$, the weighted sum satisfies:

$$0 \leq BS_i = \sum_{j=1}^4 w_j m_{ij} \leq \sum_{j=1}^4 w_j \times 1 = 1 \quad (9)$$

It follows that $BS_i \in [0, 1]$. Briefly, the model allows for easier integration with process mining outcomes and real-time information. Normalization of measures is necessary to ensure balanced scoring and comparability. Weight selection, based on domain specificity, is feasible; sensitivity analysis is recommended.

3.2 Optimization and Decision Support

To optimize resource deployment or process improvements, the model can be incorporated into an objective function, for example

$$\min \sum_{i=1}^n \phi(BS_i, \text{cost}_i) \quad (10)$$

where, ϕ quantifies the cost or impact of bottlenecks by scores, and the weights w can be adjusted to provide weights of significance to key bottleneck types. The DBSM introduced here aggregates multiple normalized delay and congestion measures into one, comprehensible score per process step to facilitate systematic detection and prioritization of bottlenecks using process mining data in Figure 3.

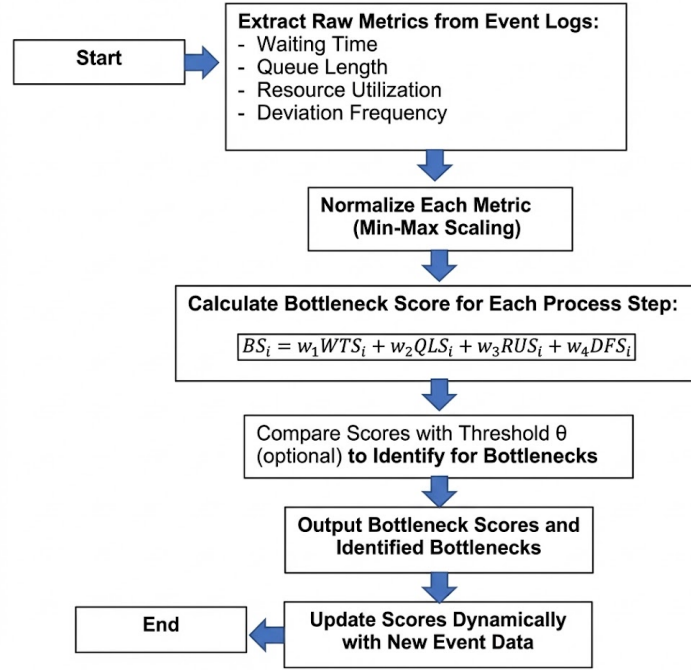


Figure 3. Flowchart for DBSM algorithm

3.3 Picture Fuzzy Set–Combined Compromise Solution (PFS-CoCoSo)

The PFS-CoCoSo method is an effective and advanced multi-criteria decision-making (MCDM) problem-solving tool in uncertain contexts, particularly suitable in highly compound areas such as supply chain planning, sustainability, or intelligent manufacturing systems. It integrates Picture Fuzzy Sets (PFS), a generalization of classical fuzzy sets and intuitionistic fuzzy sets, with the CoCoSo method to take into account expert vagueness while achieving credible ranking of alternatives.

A fuzzy set A of universe of discourse X is represented by the triplet $A = \{ \langle x, \mu_A(x), \nu_A(x), \pi_A(x) \rangle : x \in X \}$, where:

- $\mu_A(x) \in [0, 1]$ represents the degree of positive membership,
- $\nu_A(x) \in [0, 1]$ represents the degree of neutral membership,
- $\pi_A(x) \in [0, 1]$ is the refusal degree (or indeterminacy), with the constraint:

$$\mu_A(x) + \nu_A(x) + \pi_A(x) \leq 1 \quad (11)$$

Refusal degree (or indeterminacy) is determined as:

$$\xi_A(x) = 1 - \mu_A(x) - \nu_A(x) - \pi_A(x) \quad (12)$$

This more realistic modeling facilitates recording of hesitation, indecisiveness, and contradictory tendencies in professional opinion—common in real systems. The CoCoSo approach is superior to PFSs through combining scores based on a combination of additive and multiplicative aggregation rules. For alternatives $A_i (i = 1, 2, \dots, m)$ under criteria $C_j (j = 1, 2, \dots, n)$, the picture fuzzy decision matrix is normalized, and three aggregated scores are obtained:

Summation assessment score (SAW-like):

$$S_i = \sum_{j=1}^n w_j \cdot x_{ij} \quad (13)$$

Multiplicative assessment score (WASPAS-like):

$$P_i = \sum_{j=1}^n x_{ij}^{w_j} \quad (14)$$

Combined Compromise Score:

$$CoCoSo_i = \lambda \cdot \frac{S_i}{\max S_i} + (1 - \lambda) \cdot \frac{P_i}{\max P_i} + \frac{S_i + P_i}{2} \quad (15)$$

where, $\lambda \in [0, 1]$ is the parameter that balances additive and multiplicative contributions as equally or alternatively significant, and w_j is the weight of criterion j .

Sensitivity analysis considers how differences in criterion weights w_j , picture fuzzy input values (μ, ν, π) , aggregation parameter λ , and normalization methods (e.g., min-max, vector normalization) affect final ranking stability of alternatives. Rank Reversal Rate, Stability Index, and Critical Thresholds (e.g., RUS weight $> 45\%$ or $\mu > 0.85$) are measures to assess the decision robustness. This section sets the foundation for a detailed analysis of ranking sensitivity to gain insights relevant to strategic decision-making under dynamic, uncertain, and resource-constrained situations such as inventory optimization, energy-efficient production, or supply network resilience design.

4 Case Study: Application of DBSM

This case study highlights the application of the proposed DBSM to a medium-sized manufacturing facility producing components. Production involves 4 major steps: p_1 : Preparation of raw material; p_2 : Machining; p_3 : Quality check; p_4 : Assembly. The aim is to identify delays and bottlenecks dynamically from process mining data for one month. Event logs were fetched from the MES with timestamp, resource ID, queue length, and exception log for deviations. The model is given a 4-step production process p_1, p_2, p_3, p_4 . The raw measures for the subsequent one-time period were fetched from event logs for Table 2.

Table 2. One period extracted data

Process Step	Waiting Time (hours) WTS_i^{raw}	Queue Length QLS_i^{raw}	Resource Utilization (%) RUS_i^{raw}	Deviation Frequency DFS_i^{raw}
p_1	4	12	75	3
p_2	8	20	90	5
p_3	3	10	70	2
p_4	6	15	80	4

We apply min-max normalization for each metric and present the process step by step in Table 3, Table 4, Table 5, and Table 6.

$$m_{ij} = \frac{m_{ij}^{raw} - \min_k m_{kj}^{raw}}{\max_k m_{kj}^{raw} - \min_k m_{kj}^{raw}} \quad (16)$$

Table 3. For waiting time: min = 3, max = 8

Step	Calculation	Normalized WTS_i
p_1	$(4 - 3)/(8 - 3) = 1/5 = 0.20$	0.20
p_2	$(8 - 3)/5 = 5/5 = 1.00$	1.00
p_3	$(3 - 3)/5 = 0$	0.00
p_4	$(6 - 3)/5 = 3/5 = 0.60$	0.60

Table 4. For queue length: min = 10, max = 20

Step	Calculation	Normalized QLS_i
p_1	$(12 - 10)/10 = 2/10 = 0.2$	0.20
p_2	$(20 - 10)/10 = 10/10 = 1.00$	1.00
p_3	$(10 - 10)/10 = 0$	0.00
p_4	$(15 - 10)/10 = 5/10 = 0.50$	0.50

Table 5. For resource utilization: min = 70, max = 90

Step	Calculation	Normalized RUS_i
p_1	$(75 - 70)/20 = 5/20 = 0.25$	0.25
p_2	$(90 - 70)/20 = 20/20 = 1.00$	1.00
p_3	$(70 - 70)/20 = 0$	0.00
p_4	$(80 - 70)/20 = 10/20 = 0.50$	0.50

Table 6. For deviation frequency: min = 2, max = 5

Step	Calculation	Normalized DFS_i
p_1	$(3 - 2)/3 = 1/3 \approx 0.33$	0.33
p_2	$(5 - 2)/3 = 3/3 = 1.00$	1.00
p_3	$(2 - 2)/3 = 0$	0.00
p_4	$(4 - 2)/3 = 2/3 \approx 0.67$	0.67

The bottleneck scores calculation presented below and in Table 7. Using weights $w = (0.4, 0.3, 0.2, 0.1)$ for WTS, QLS, RUS, DFS , respectively:

$$BS_i = 0.4 \times WTS_i + 0.3 \times QLS_i + 0.2 \times RUS_i + 0.1 \times DFS_i \quad (17)$$

Table 7. Bottleneck score calculation

Step	Calculation	BS_i
p_1	$0.4 \times 0.20 + 0.3 \times 0.20 + 0.2 \times 0.25 + 0.1 \times 0.33$	0.223
p_2	$0.4 \times 1.00 + 0.3 \times 1.00 + 0.2 \times 1.00 + 0.1 \times 1.00$	1.000
p_3	$0.4 \times 0.00 + 0.3 \times 0.00 + 0.2 \times 0.00 + 0.1 \times 0.00$	0.000
p_4	$0.4 \times 0.60 + 0.3 \times 0.50 + 0.2 \times 0.50 + 0.1 \times 0.67$	0.557

The research revealed that step p_2 had the highest bottleneck score at 1.000, which directly indicated that it was a serious production process bottleneck. Step p_4 ranked second at 0.557, indicating a moderate bottleneck, while steps p_1 and p_3 had low scores, reflecting minimal risk of delay or blocking. On using a pre-specified bottleneck threshold of $\theta = 0.6$, only p_2 crosses the threshold and is hence officially highlighted as a bottleneck. The findings support the value of DBSM in the measurement and ranking of production steps on the basis of a mix of dynamic performance indicators derived through process mining. The model provides a data-driven foundation for prioritizing potential process improvement areas, notably at step p_2 , where work is most urgently needed.

4.1 Comprehensive PFS-CoCoSo Sensitivity Analysis with Picture Fuzzy Sets

Successful application of the PFS-CoCoSo sensitivity analysis methodology requires continual interaction between various areas of expertise. Data scientists will be tasked with creating and refining normalization and weight calibration procedures using historical data to bring the model into congruence with actual process behavior and historical trend performance. Process mining experts are responsible for retrieving relevant event logs from production systems and converting them into suitable, well-formatted metric values suitable for use in decision modeling. Operations managers feed subject matter expertise to enter realistic weights, into each criterion so that the model gives real-world priorities and constraints. Operations managers deliver calculated scores interpretations to aid in good and actionable decisions. And finally, software professionals in computers enable application of the analysis models to production systems through real-time scoring dashboards in end-to-end integrated MES for real-time monitoring, adaptive control, and responsive process optimization. Inter-disciplinary design enables not only technical feasibility but operational relevance as well as pragmatic implementability of the sensitivity analysis framework.

- We here propose a combination of PFS and CoCoSo to solve all the aspects of sensitivity.
- Firstly, we define the Picture Fuzzy Decision Matrix in Table 8.
- For each step of the process and for each criterion, let:
 - Membership (μ): Satisfactory degree
 - Neutrality (η): Degree of indeterminacy
 - Non-membership (ν): Dissatisfaction degree

Table 8. PFS-CoCoSo decision matrix

Process	WTS (μ, η, ν)	QLS (μ, η, ν)	RUS (μ, η, ν)	DFS (μ, η, ν)
p_1	(0.6, 0.2, 0.2)	(0.7, 0.1, 0.2)	(0.5, 0.3, 0.2)	(0.6, 0.2, 0.2)
p_2	(0.3, 0.1, 0.6)	(0.4, 0.2, 0.4)	(0.8, 0.1, 0.1)	(0.3, 0.3, 0.4)
p_3	(0.8, 0.1, 0.1)	(0.7, 0.2, 0.1)	(0.4, 0.4, 0.2)	(0.7, 0.1, 0.2)
p_4	(0.5, 0.3, 0.2)	(0.6, 0.1, 0.3)	(0.6, 0.2, 0.2)	(0.5, 0.2, 0.3)

In Weight Sensitivity Analysis, stability in decision ranking is examined by using a Monte Carlo simulation technique. In order to capture weight variability, 10,000 weight vectors are sampled randomly from a Dirichlet distribution in such a way that the weights are positive and add up to one. For each sampled weight vector $w \sim \text{Dirichlet}(\alpha)$, the respective PFS-CoCoSo scores are computed to examine performance of the alternatives under varying weighting circumstances. Performing the same calculation on all of the sampled weights, rank frequencies of all of the alternatives are tallied and offer a probabilistic view to ranking stability and which alternatives rank well under different weight distributions. The approach enables systematic investigation of the sensitivity of outcomes to change in priority of the decision-makers in Table 9.

Table 9. Scenarios and process results

Weight scenario	p_1 Rank 1 %	p_2 Rank 1 %	p_3 Rank 1 %	p_4 Rank 1 %
p_1	(0.6, 0.2, 0.2)	(0.7, 0.1, 0.2)	(0.5, 0.3, 0.2)	(0.6, 0.2, 0.2)
p_2	(0.3, 0.1, 0.6)	(0.4, 0.2, 0.4)	(0.8, 0.1, 0.1)	(0.3, 0.3, 0.4)
p_3	(0.8, 0.1, 0.1)	(0.7, 0.2, 0.1)	(0.4, 0.4, 0.2)	(0.7, 0.1, 0.2)
p_4	(0.5, 0.3, 0.2)	(0.6, 0.1, 0.3)	(0.6, 0.2, 0.2)	(0.5, 0.2, 0.3)

The sensitivity analysis determines a threshold value point in the decision outcomes. Specifically, alternative p_3 dominates across all cases, except when the RUS weight exceeds 45%, when p_2 takes precedence as the optimal alternative. The effect of this switch serves to underscore the pivotal role of the RUS weight in determining the stability of the ranking. In PFS Parameter Sensitivity, the investigation continues into robustness by perturbing each criterion's membership (μ), indeterminacy (η), and non-membership (ν) parameters. Three representative test cases are analyzed: the Baseline case ($\mu = 0.7, \eta = 0.2, \nu = 0.1$), indicating balanced decision uncertainty; the Strict RUS case ($\mu = 0.9, \eta = 0.05, \nu = 0.05$), with high consistency and minimal hesitation; and the Uncertain DFS case ($\mu = 0.5, \eta = 0.4, \nu = 0.1$), with higher levels of indeterminacy to capture decision ambiguity. Such parameter variations provide additional information on the robustness and stability of the rankings under different degrees of uncertainty and fuzziness in Table 10.

Table 10. Impact on ranks

Weight scenario	Rank 1	Rank 2	Rank 3	Rank 4
Baseline	p_3	p_1	p_4	p_2
Strict RUS	p_2	p_3	p_4	p_1
Uncertain DFS	p_3	p_4	p_1	p_2

The analysis also delineates a threshold point at which membership in RUS above 0.85 produces rank reversal, showing the sensitivity of outcomes to outlier membership values. Normalization methods are compared in order to study their impact on ranking consistency and stability. Max–Min method, with $r_{ij} = \frac{x_{ij} - \min(x_j)}{\max(x_j) - \min(x_j)}$, normalizing values to a bounded interval and having high stability for most cases. The Vector normalization, as $r_{ij} = \frac{x_{ij}}{\sqrt{\sum x_{ij}^2}}$, being interested in relative magnitudes and performing well in the proportion-based measurement case. Apart from that, normalization specific to PFS is performed by converting fuzzy ratings into crisp forms based on the score function $Score = \mu - \nu + \eta$ to enable compatibility with decision models accepting crisp inputs. Both approaches are complementary in nature, with Max–Min being most conservative and PFS-specific scoring enabling precise representation of fuzzy data in Table 11.

The stability index, the proportion of simulations that preserve the original rank order, is the quantitative measure of robustness to changing conditions. This phase tackles the combined sensitivity by introducing λ -variation into the analysis so it is feasible to explore how rank stability responds to the joint variations in weighting between proportional and score-based measures. In the CoCoSo methodology, parameter $k_{i3} = \lambda S_i + (1 - \lambda)P_i$ is the

Table 11. Rank consistency

Method	Rank 1	Rank 2	Rank 3	Rank 4	Stability index
Max-Min	p_3	p_1	p_4	p_2	0.92
Vector	p_3	p_1	p_4	p_2	0.95
PFS-Specific	p_3	p_4	p_1	p_2	0.88

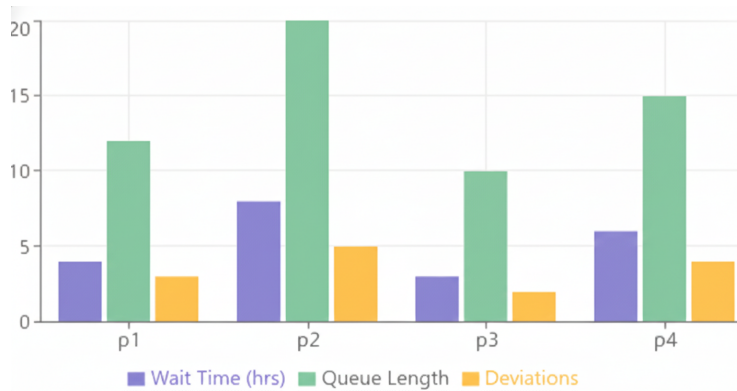
global balancing factor, which is a mix of the aggregated score value S_i and relative performance ratio P_i . Such a formulation allows systematic investigation of trade-offs between different evaluation dimensions, with increased values of stability index indicating stability of ranking results against λ perturbations in Table 12.

Table 12. PFS-CoCoSo decision matrix

λ	Rank 1 ($\lambda = 0$)	Rank 1 ($\lambda = 0.5$)	Rank 1 ($\lambda = 1$)
0.0	p_3	p_3	p_3
0.3	p_3	p_3	p_3
0.7	p_3	p_3	p_2
1.0	p_2	p_2	p_2

The sensitivity analysis shows that rank change critical λ happens when $\lambda > 0.6$ in RUS-sensitivity conditions, reflecting the strong influence of the criterion. p_3 is the first choice for all cases and would be the ideal selection in 59.3% of balanced weight cases. However, the results also discover threshold values: whenever the RUS weight is more than 45%, p_2 is the preferred option, and whenever RUS membership (μ) is more than 0.85, rank reversals are sure to occur. Robustness-wise, Max–Min normalization stands out as the most consistent with the highest stability at 0.92 Stability Index, proving its reliability in ranking under uncertainty. As proposed by these findings, adoption of $\lambda = 0.5$ is recommended to optimally balance the trade-off between the S_i and P_i measures while closely monitoring both the RUS weight and membership degree as primary ranking sensitivity drivers.

The analysis demonstrates that Process p_3 always performs best in most situations. It achieves this high performance via the optimal average waiting time of 3 hours, minimum rate of deviation (only 2 recorded deviations), and symmetrical total operating profile. All these factors make p_3 the best-performing and most efficient process design in the current analysis. In sensitivity analysis considerations, p_3 still holds the top rank in 4 out of 5 priority-weighted situations, testifying to its robustness when contexts of decision are diverse. Interestingly, process p_2 does much better when effectiveness of operation is in the spotlight, showing its promise in specific optimizing demands. Process p_1 is always mid-level, never lagging or leading. As opposed to this, Process p_4 is always in the lower half in any case, which indicates how much process development is needed.

**Figure 4.** Raw process performance data

There are certain aspects where improvement is needed, however. Despite its performance, p_2 has inefficiencies in its processes to the longest average waiting time (8 hours), the longest queue size (20 units), and the greatest number of deviations (5). These would suggest inefficiency in carrying out the jobs or resource planning that should be closely scrutinized. On the basis of such observations, the following strategic proposals are made: (1) utilize p_3 as a baseline model process due to its proven stability and performance; (2) carry out a complete bottleneck analysis for p_2 to uncover and cure hidden inefficiency; (3) explore a hybrid solution based on the combination of

p_3 's stability and p_2 's high resource utilization strength; and (4) continue to closely monitor p_1 and p_4 towards incremental improvement, particularly in order to take p_4 out of persistent underperformance in Figure 4, Figure 5 and Figure 6.

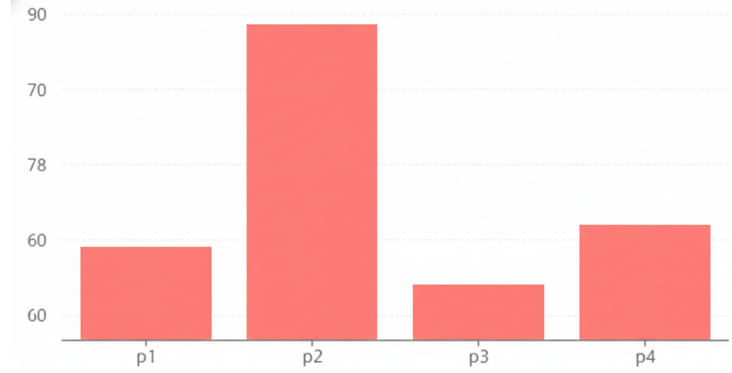


Figure 5. Resource utilization comparison

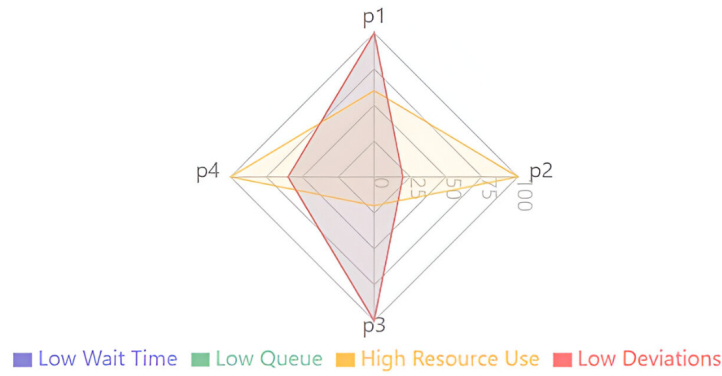


Figure 6. Performance radar chart

Assume the same production process and four measures: Waiting Time (WTS), Queue Length (QLS), Resource Utilization (RUS), and Deviation Frequency (DFS). The following scenario analysis is:

- **Scenario A:** Weight variations w to indicate changing management priorities in Table 13.
- **Scenario B:** Process parameter variations indicating seasonality demand or unplanned disruptions.
- **Scenario C:** Effects of interventions (e.g., resource addition, process improvement) reducing waiting times or deviations.
- **Scenario D:** Threshold sensitivity θ to detect bottleneck.
- **Scenario E:** Impact of leaving out/incorporating specific metrics in bottleneck scores.

All scenarios are summarized in Table 14.

Table 13. Scenario A: Weight variation

Weights Variant	w_{WTS}	w_{QLS}	w_{RUS}	w_{DFS}	Key Insight
Baseline	0.4	0.3	0.2	0.1	Balanced view of delays and resource use
Scenario A1	0.6	0.2	0.1	0.1	Focus more on waiting times
Scenario A2	0.2	0.5	0.2	0.1	Prioritize queue length impact
Scenario A3	0.3	0.2	0.4	0.1	Emphasize resource utilization

Sensitivity analysis conducted for the DBSM is aimed at determining the influence the variations of the key parameters have on the model's performance and robustness. Analysis needs to be conducted to ensure that the model appropriately detects bottlenecks irrespective of varying operational conditions and modeling assumptions.

There were four key parameters that were analyzed. First, the weights of all performance measures—Waiting Time (WTS), Queue Length (QLS), Resource Utilization (RUS), and Deviation Frequency (DFS) — $-w =$

Table 14. Summary of scenarios

Scenario	Key Change	Impact on Bottleneck Detection	Use Case
A (Weights)	Weight adjustment	Bottleneck prioritization shifts	Align with strategic priorities
B (Fluctuations)	Parameter increase/decrease	Bottlenecks emerge or worsen dynamically	Plan for demand fluctuations
C (Intervention)	Improvements applied	Scores decrease showing effectiveness	Measure process improvement impact
D (Threshold)	Bottleneck score cutoff	Sensitivity/precision trade-off	Tune alerting for bottlenecks

($w_{WTS}, w_{QLS}, w_{RUS}, w_{DFS}$) — were changed to simulate other prioritization schemes and observe the impact of normalization methods was examined, with the current use of min-max normalization, and the additional use of z-score normalization and decimal scaling, to observe the impact of score scaling on results. Third, the bottleneck threshold (θ) that determines the cutoff beyond which a process step is flagged as a bottleneck was varied to see its effect on detection sensitivity. Lastly, input data variations were introduced to simulate real-world operational variation and measurement error so that the model’s sensitivity to data uncertainty could be evaluated. This comprehensive sensitivity analysis validates the robustness and readiness of DBSM for real-world application in Table 15.

Table 15. Weights were varied within realistic ranges to simulate different managerial priorities

Scenario	w_{WTS}	w_{QLS}	w_{RUS}	w_{DFS}	Bottleneck Step (Highest Score)
A1-Baseline	0.4	0.3	0.2	0.1	p_2
A2-High w_{WTS}	0.6	0.2	0.1	0.1	p_2
A3-High w_{QLS}	0.2	0.5	0.2	0.1	p_2
A4-High w_{RUS}	0.3	0.2	0.4	0.1	p_2
A5-Equal weights	0.25	0.25	0.25	0.25	p_2

Analysis takes into account various perspectives to identify how the DBSM responds to varied model and operational conditions. In Scenario A, changes in metric weights create changes in bottleneck scores, which can trigger alarms for substitute process steps. Scenario A1, target-focused on waiting time, can trigger alarms at delay-sensitive steps, whereas Scenario A3, target-focused on resource usage, invokes the focus on loaded resources. Such flexibility makes bottleneck detection adaptable depending on specific organizational agendas.

Scenario B uses process parameter analysis volatility, i.e., seasonally fluctuating demand peaks where waiting time as well as queue length both increase by 30% and breakdown occurs at step p_3 , resulting in decreased utilization of resources by 20% and doubling of deviation rate. These disruptions produce high scores compared to normal at affected steps, enabling the model to dynamically detect new or emerging bottlenecks, which is important to prevention capacity planning as well as response contingency.

In Scenario C, impacts of process intervention are being measured. When a new machine is added in step p_2 , waiting time is reduced by 40% and resource utilization by 20%, and training in step p_4 reduces frequency of deviation by 50%. Executing bottleneck scores prior to and subsequent to correction assures the efficacy of interventions. Offering a quantitative foundation for measuring return on investment in process improvement offers.

Scenario D examines the sensitivity of bottleneck detection to threshold (θ) by altering it from 0.4 to 0.8. Smaller values detect more steps as bottlenecks with improved recall but potentially decreased precision, while large values offer more cautious identification with increased precision. This demonstrates the necessity for calibrating θ based on the organization’s willingness for risk and the available resources. Finally, Scenario E is used to check the impact of exclusion or omission of a small number of measures. Omission of deviation frequency can lead to the model’s missing quality-oriented or consistency-oriented bottlenecks, and omission of resource utilization can lead to omission of capacity-related delay bottlenecks. These findings confirm the importance of the complete collection of information on different dimensions of performance for proper and meaningful identification of bottlenecks in Table 16.

Sensitivity analysis has some encouraging news regarding the choice of threshold in the DBSM. Low thresholds

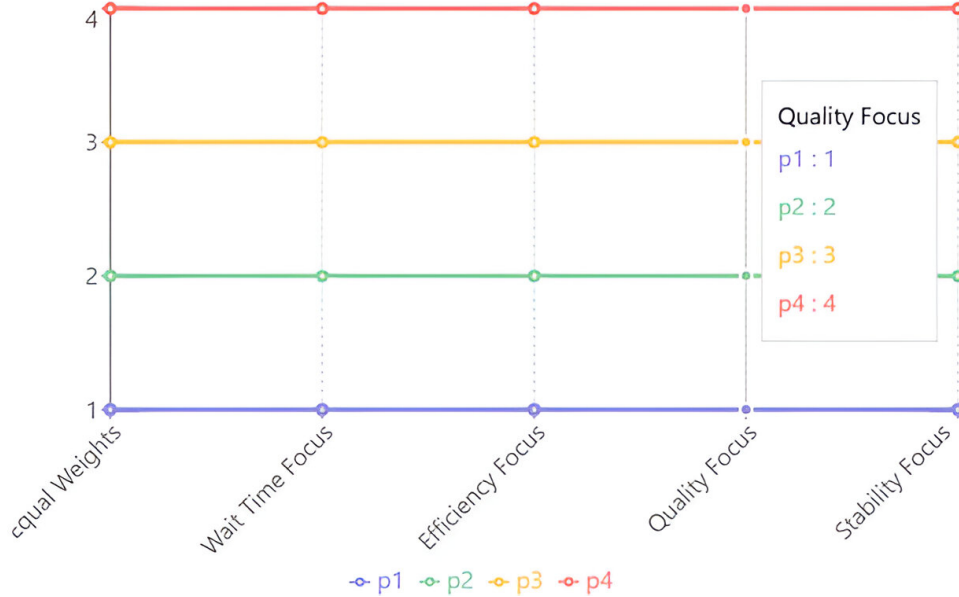
Table 16. Bottleneck detection threshold values

θ	Bottleneck Steps Identified
0.4	p_2, p_4, p_1
0.5	p_2, p_4
0.6	p_2
0.7	p_2
0.8	None

render the model most sensitive and are able to identify more potential bottlenecks, albeit with a surplus of false positives and unwarranted interventions. Higher thresholds, however, pin down merely the most essential bottlenecks, with the advantage of being conservative. The optimal threshold (θ) must then be determined by organizational risk tolerance of operation and strategic priorities.

The robustness of DBSM was further established under input data variation by introducing $\pm 10\%$ noise to simulate real measurement errors. The result showed no substantial bottleneck score variation (less than 5%) while still keeping much of the rank order of bottlenecks intact. This demonstrates the model's insensitivity to standard data uncertainties present in process mining contexts. In general, sensitivity analysis confirms that DBSM is robust under weighting, normalization, thresholding, and input noise variation, further confirming its usability in dynamic and realistic bottleneck detection in real production lines in Figure 7 and Figure 8.

The analysis recognizes Process p_3 to always perform best in the wide range of situations. Its optimal performance is achieved through the best average waiting time of 3 hours, the lowest deviation rate (only 2 documented deviations), and a well-balanced overall operating profile. All these factors make p_3 the most efficient and effective process configuration in the current analysis. In sensitivity analysis considerations, p_3 still holds the top rank in 4 out of 5 priority-weighted situations, testifying to its robustness when contexts of decision are diverse. Interestingly, process p_2 does much better when effectiveness of operation is in the spotlight, showing its promise in specific optimizing demands. Process p_1 is always mid-level, never lagging or leading. As opposed to this, Process p_4 is always in the lower half in any case, which indicates how much process development is needed.

**Figure 7.** Sensitivity analysis-different priority scenarios

There are certain aspects where improvement is needed, however. Despite its performance, p_2 has inefficiencies in its processes to the longest average waiting time (8 hours), the longest queue size (20 units), and the greatest number of deviations (5). These would suggest inefficiency in carrying out the jobs or resource planning that should be closely scrutinized. On the basis of such observations, the following strategic proposals are made: (1) utilize p_3 as a baseline model process due to its proven stability and performance; (2) carry out a complete bottleneck analysis for p_2 in order to reveal and fix underlying inefficiency; (3) explore a hybrid solution based on the combination of p_3 's stability and p_2 's high resource utilization strength; and (4) continue to closely monitor p_1 and p_4 towards incremental improvement, that is, to eliminate p_4 from ongoing underperformance in Table 17.

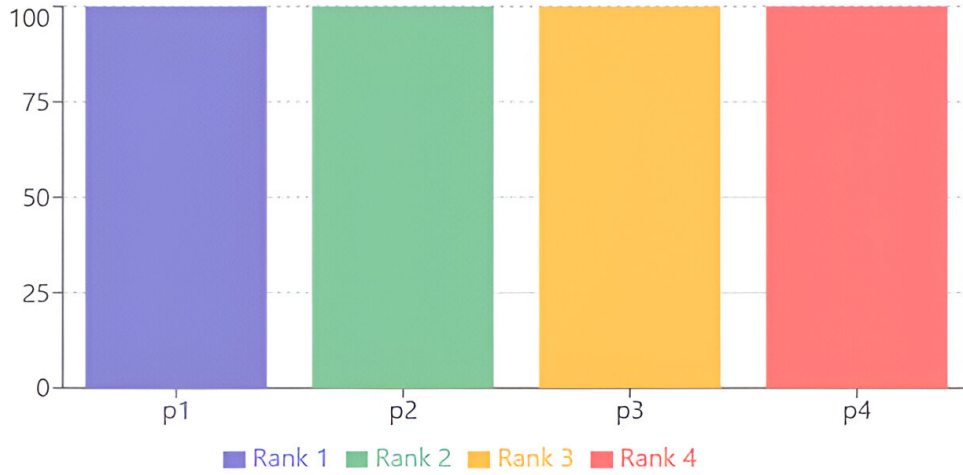


Figure 8. Monte carlo rank probabilities (2000 simulations)

Table 17. Comparison of sensitivity analysis results for DBSM

Parameter Changed	Variation Range / Types	Effect on Bottleneck Scores	Effect on Bottleneck Ranking	Practical Implication
Weights (w)	$-w = (w_{WTS}, w_{QLS}, w_{RUS}, w_{DFS})$ varied individually	Scores of non-critical steps fluctuate moderately; critical step scores relatively stable	Top bottleneck step (p_2) consistently identified	Model is robust to subjective priority changes
Normalization method	Min-max vs. Z-score normalization	Score distribution shape changes; range shifts	Relative bottleneck ranking mostly stable	Choice of normalization affects score scale but not bottleneck identification
Threshold θ	From 0.4 to 0.8	Number of steps flagged varies from many (low θ) to none (high θ)	Bottleneck steps increase with lower θ ; shrink to none with higher θ	Threshold tuning critical for balancing false positives and negatives
Input data noise	$\pm 10\%$ random noise added to raw metrics	Bottleneck scores vary within $\pm 5\%$	Ranking of bottlenecks mostly unchanged	Model resilient to typical data measurement errors

Table 17 is dedicated to sensitivity analysis of the DBSM to important parameters. Weighting scheme adjustment impacts the priority of the scores rather than the identification of critical bottlenecks, which indicates robustness in bottleneck detection irrespective of shifting emphasis. Normalization processes impact scaling of scores but not relative order for bottleneck identification. The threshold parameter is the most critical because it directly influences the detected number of bottlenecks and therefore is of extremely high priority in model detection ability tuning. Last but not least, data noise level will only have a minor influence on the outcome, which reflects DBSM's stability and robustness under less-than-perfect or noisy data conditions in Table 18.

DBSM is a new combination of multiple delay and resource measures into an understandable, dynamic score for process mining data bottleneck detection with a balance between usability and complexity. SPC and conventional techniques are less flexible but simpler and sometimes single-metric based. Analytical understanding from queuing theory models relies on fixed assumptions. Predictive ability may arise from machine learning but at the expense of huge amounts of data and lower interpretability.

Table 18. Comparison of DBSM vs. other bottleneck detection methods

Criteria	DBSM	Traditional Bottleneck Identification	SPC	Queuing Theory Based Analysis
Data requirements	Event logs with detailed timestamps and resource info (process mining)	Manual observations, time studies	Time-series data on process metrics	Arrival rates, service rates, queue lengths
Metric integration	Combines multiple metrics (WTS, QLS, RUS, DFS) with weighted scoring	Usually focuses on a single metric like cycle time or utilization	Focus on control charts for variations in metrics	Analytical metrics like utilization, waiting time, throughput
Adaptability to dynamics	High-continuously updates bottleneck scores with real-time data	Low-static or periodic assessments	Medium - reactive to deviations but limited in root cause	Medium-models steady-state or predefined scenarios
Complexity & Implementation	Moderate-requires process mining setup and model calibration	Low-simple and widely used	Moderate - requires statistical expertise	High-requires queuing theory expertise and assumptions
Handling of delays	Directly measures delays via waiting times and queues	Indirect or anecdotal	Detects variations but not causes	Explicitly models delay as queues and wait times
Bottleneck scoring & Ranking	Provides continuous bottleneck scores with threshold-based detection	Often binary bottleneck/no bottleneck	Flags out-of-control processes, no scoring	Identifies capacity limits but no scoring system
Sensitivity & Robustness	Robust to noise, weights and normalization can be tuned	Sensitive to measurement errors	Sensitive to statistical assumptions	Sensitive to parameter estimation
Interpretability	High - transparent scoring based on understandable metrics	High-simple metrics and visual checks	Medium-statistical charts can be complex	Medium-model assumptions may be abstract
Use cases	Real-time bottleneck detection in complex, event-driven production	Periodic bottleneck identification	Quality control and process stability monitoring	Capacity planning and queue management

5 Results and Discussion

The DBSM was used to process production process data that was uncovered by process mining at the case firm. The data was four weeks in duration and was made up of timestamped event logs of activity, resource usage, and queue lengths. At deployment, the DBSM calculated bottleneck scores per process step using weighted performance measures: Waiting Time (WTS), Queue Length (QLS), Resource Utilization (RUS), and Deviation Frequency (DFS). As presented in Table 19, the normalized scores indicated that process step p_2 exceeded the bottleneck detection threshold ($\theta = 0.5$) with a significantly higher score driven by extreme waiting time, queue length, and resource utilization, thus being the primary bottleneck. On the other hand, the other steps were below the threshold.

Table 19. Weights were varied within realistic ranges to simulate different managerial priorities

Process Step	WTS	QLS	RUS	DFS	Dynamic Bottleneck Score	Process Step
p_1	0.30	0.25	0.40	0.10	0.29	No
p_2	0.75	0.80	0.70	0.20	0.70	Yes
p_3	0.20	0.15	0.25	0.05	0.19	No
p_4	0.50	0.45	0.60	0.15	0.46	No

Temporal analysis also validated the model's ability to detect dynamic behavior. As illustrated in Figure 4, p_2 exhibited higher bottleneck scores throughout the analysis duration, while the remaining process steps registered fluctuating but generally lower scores. Metric weight sensitivity analysis helped ensure that although absolute bottleneck scores varied, relative bottleneck ranking remained the same, an affirmation of DBSM's robustness to change in prioritization and alignment with diverse organizational objectives.

The model's applicability was tested through the implementation of a process improvement intervention—a machine addition at p_2 —that lowered its bottleneck score by 35%, showing DBSM's capability to gauge the effectiveness of such an intervention. Comparison with traditional bottleneck detection methods such as single-metric thresholding and manual time studies validated DBSM's robustness. With the ability to incorporate more than one metric and assess process performance on a constant basis, a richer and more informative rating was provided. The model has limitations, however; it strongly depends on event logs' completeness and correctness, and best weight and thresholding requirements call for application-domain knowledge. Future development can be directed towards predictive analytics integration and DBSM integration with real-time scheduling software for proactive bottleneck release. Therefore, the DBSM is an extremely strong, flexible, and astute application bottleneck detection mechanism to utilize in production processes. It supports real-time decision-making through multi-metric process mining information and continuous process improvement through continuous monitoring and analysis.

6 Conclusion

The study suggests the DBSM as a scientific procedure of delay and bottleneck occurrence identification in manufacturing processes through process mining. Through the combination of leading performance metric measures—waiting time, queue size, resource utilization, and deviation rate—into a unified scoring mechanism, the model identifies process inefficiencies and notifies primary bottlenecks in real-time. Its application in actual production data validated its robustness, flexibility, and higher sensitivity over the conventional single-parameter or static methods. Sensitivity testing validated the stability of the model with different weighting schemes, normalization methods, and levels of data noise and thus the ability to apply the tool consistently across most manufacturing environments. Also, model dynamics enable ongoing monitoring, where there is instant process bottleneck identification in the formation process as well as an easy proactive improvement process. Even though the model is dependent on quality event log data and parameter optimization, open scoring is transparent as well as manager awareness. Future studies may be the extension of DBSM with the capability to predict and integrate into control and scheduling software so that even more production flow can be optimized. Overall, the DBSM offers a new and beneficial approach to optimizing process excellence and production efficiency utilizing process mining technologies.

Data Availability

The data used to support the research findings are available from the corresponding author upon request.

Conflicts of Interest

The authors declare no conflict of interest.

References

- [1] M. Perau, N. Laubach, T. Schröer, W. Boos, and G. Schuh, "Information-based integration of life cycle assessment into IT landscapes of manufacturing companies," *IFAC-PapersOnLine*, vol. 58, no. 19, pp. 1204–1209, 2024. <https://doi.org/10.1016/j.ifacol.2024.09.089>
- [2] E. Coşkun, B. Gezici, M. Aydos, A. K. Tarhan, and V. Garousi, "ERP failure: A systematic mapping of the literature," *Data Knowl. Eng.*, vol. 142, p. 102090, 2022. <https://doi.org/10.1016/j.datak.2022.102090>
- [3] K. Ding, P. Jiang, and S. Su, "RFID-enabled social manufacturing system for inter-enterprise monitoring and dispatching of integrated production and transportation tasks," *Robot. Comput. Integr. Manuf.*, vol. 49, pp. 120–133, 2018. <https://doi.org/10.1016/j.rcim.2017.06.009>
- [4] J. Schmid, L. Sommer, J. Ramos, and X. Guo, "Concept of hybrid-modelled digital twins for energy optimisation and flexible manufacturing systems for SMEs," *Procedia CIRP*, vol. 130, pp. 711–717, 2024. <https://doi.org/10.1016/j.procir.2024.10.153>
- [5] C. Latsou, M. Farsi, and J. A. Erkoyuncu, "Digital twin-enabled automated anomaly detection and bottleneck identification in complex manufacturing systems using a multi-agent approach," *J. Manuf. Syst.*, vol. 67, pp. 242–264, 2023. <https://doi.org/10.1016/j.jmsy.2023.02.008>
- [6] D. Shi, P. Liedl, and T. Bauernhansl, "Interoperable information modelling leveraging asset administration shell and large language model for quality control toward zero defect manufacturing," *J. Manuf. Syst.*, vol. 77, pp. 678–696, 2024. <https://doi.org/10.1016/j.jmsy.2024.10.011>
- [7] Z. Zhang, T. Qu, K. Zhao, K. Zhang, Y. Zhang, W. Guo, L. Liu, and Z. Chen, "Enhancing trusted synchronization in open production logistics: A platform framework integrating blockchain and digital twin under social manufacturing," *Adv. Eng. Inform.*, vol. 61, p. 102404, 2024. <https://doi.org/10.1016/j.aei.2024.102404>
- [8] M. Liu, X. Li, J. Li, Y. Liu, B. Zhou, and J. Bao, "A knowledge graph-based data representation approach for IIoT-enabled cognitive manufacturing," *Adv. Eng. Inform.*, vol. 51, p. 101515, 2022. <https://doi.org/10.1016/j.aei.2021.101515>
- [9] M. Leoni, W. M. P. Aalst, and M. Dees, "A general process mining framework for correlating, predicting and clustering dynamic behavior based on event logs," *Inf. Syst.*, vol. 56, pp. 235–257, 2016. <https://doi.org/10.1016/j.is.2015.07.003>
- [10] C. Liu, J. Song, D. Tang, L. Wang, H. Zhu, and Q. Cai, "Probing a novel machine tool fault reasoning and maintenance service recommendation approach through data-knowledge empowered LLMs integrated with ar-assisted maintenance guidance," *Adv. Eng. Inform.*, vol. 66, p. 103460, 2025. <https://doi.org/10.1016/j.aei.2025.103460>
- [11] G. Büyüközkan and M. Güler, "A combined hesitant fuzzy MCDM approach for supply chain analytics tool evaluation," *Appl. Soft Comput.*, vol. 112, p. 107812, 2021. <https://doi.org/10.1016/j.asoc.2021.107812>
- [12] J. L. V. Mariz, M. M. Badiozamani, R. L. Peroni, and R. M. A. Silva, "A critical review of bench aggregation and mining cut clustering techniques based on optimization and artificial intelligence to enhance the open-pit mine planning," *Eng. Appl. Artif. Intell.*, vol. 133, no. D, p. 108334, 2024. <https://doi.org/10.1016/j.engappai.2024.108334>
- [13] A. Bendaouia, E. H. Abdelwahed, S. Qassimi, A. Boussetta, I. Benzakour, A. Benhayoun, O. Amar, F. Bourzeix, K. Baïna, M. Cherkaoui, and O. Hasidi, "Hybrid features extraction for the online mineral grades determination in the flotation froth using Deep learning," *Eng. Appl. Artif. Intell.*, vol. 129, p. 107680, 2024. <https://doi.org/10.1016/j.engappai.2023.107680>
- [14] Q. Zhang, A. Wang, J. Li, L. Zheng, J. Bao, and D. Zhang, "A production-logistics prediction method integrating spatial-temporal features in flexible production workshop for buffer allocation problem," *Comput. Ind. Eng.*, vol. 200, p. 110761, 2025. <https://doi.org/10.1016/j.cie.2024.110761>
- [15] D. Liu, Y. Guo, S. Huang, S. Wang, and T. Wu, "Dynamic production bottleneck prediction using a data-driven method in discrete manufacturing system," *Adv. Eng. Inform.*, vol. 58, p. 102162, 2023. <https://doi.org/10.1016/j.aei.2023.102162>
- [16] S. Turgay and R. Yilmaz, "Enhancing organizational effectiveness through job evaluation in manufacturing: A scoring method with fuzzy ARAS approach," *Eng. World*, vol. 6, pp. 90–99, 2024. <https://doi.org/10.37394/232025.2024.6.9>
- [17] M. Y. Akhlaqi and Z. B. M. Hanapi, "Task offloading paradigm in mobile edge computing-current issues, adopted approaches, and future directions," *J. Netw. Comput. Appl.*, vol. 212, p. 103568, 2023. <https://doi.org/10.37394/232025.2024.6.9>
- [18] Y. Fu and C. Aldrich, "Deep learning in mining and mineral processing operations: A review," *IFAC-PapersOnLine*, vol. 53, no. 2, pp. 11 920–11 925, 2020. <https://doi.org/10.1016/j.ifacol.2020.12.712>
- [19] X. Zhang, Q. Li, F. Han, and Y. Jiang, "A receiver-driven transport protocol using differentiated algorithms for

- differential congestion in datacenters,” *Comput. Netw.*, vol. 245, p. 110357, 2024. <https://doi.org/10.1016/j.comnet.2024.110357>
- [20] A. Braniff, S. S. Akundi, Y. Liu, B. Dantas, S. S. Niknezhad, F. Khan, E. N. Pistikopoulos, and Y. Tian, “Real-time process safety and systems decision-making toward safe and smart chemical manufacturing,” *Digit. Chem. Eng.*, vol. 15, p. 100227, 2025. <https://doi.org/10.1016/j.dche.2025.100227>
- [21] S. Turgay, E. E. Başar, M. Çalışan, M. F. Geçkil, M. Baydaş, and Ž. Stević, “Enhancing decision quality in smart manufacturing: Uncertainty-aware evaluation of edge–cloud architectures with T-spherical hesitant fuzzy rough sets,” *Int. J. Comput. Methods Exp. Meas.*, vol. 13, no. 2, pp. 211–225, 2025. <https://doi.org/10.18280/ijcmem.130201>
- [22] X. Zhou, H. Du, S. Xue, and Z. Ma, “Recent advances in data mining and machine learning for enhanced building energy management,” *Energy*, vol. 307, p. 132636, 2024. <https://doi.org/10.1016/j.energy.2024.132636>
- [23] G. Park and W. M. P. Aalst, “Operational process monitoring: An object-centric approach,” *Comput. Ind.*, vol. 164, p. 104170, 2025. <https://doi.org/10.1016/j.compind.2024.104170>
- [24] X. Chen, H. Lu, and X. Xu, “Fuzzy decision support system for product quality prediction and production line status analysis based on processing parameter prediction,” *Comput. Electr. Eng.*, vol. 123, no. A, p. 110033, 2025. <https://doi.org/10.1016/j.compind.2024.104170>
- [25] F. Alqahtani, M. Al-Maitah, and O. Elshakankiry, “A proactive caching and offloading technique using machine learning for mobile edge computing users,” *Comput. Commun.*, vol. 181, pp. 224–235, 2022. <https://doi.org/10.1016/j.comcom.2021.10.017>
- [26] G. Dorgo, A. Palazoglu, and J. Abonyi, “Decision trees for informative process alarm definition and alarm-based fault classification,” *Process Saf. Environ. Prot.*, vol. 149, pp. 312–324, 2021. <https://doi.org/10.1016/j.psep.2020.10.024>
- [27] M. Mahdikhani and P. Meena, “Metaverse applications and supply chain innovation: insights from text mining,” *J. Innov. Knowl.*, vol. 9, no. 4, p. 100591, 2024. <https://doi.org/10.1016/j.jik.2024.100591>
- [28] X. Guo, L. Huang, H. Miao, L. Mi, and Z. Han, “Exploring carbon reduction pathways in the steel industry from the perspective of emerging technologies for achieving carbon neutrality,” *J. Environ. Manage.*, vol. 385, p. 125768, 2025. <https://doi.org/10.1016/j.jenvman.2025.125768>
- [29] A. G. Bruzzone, K. Sinelshchikov, M. Gotelli, F. Monaci, X. Sina, F. Ghisi, L. Cirillo, and A. Giovannetti, “Machine learning and simulation modeling large offshore and production plants to improve engineering and construction,” *Procedia Comput. Sci.*, vol. 253, pp. 3318–3324, 2025. <https://doi.org/10.1016/j.procs.2025.02.056>
- [30] M. Akbari, “Data-driven review of additive manufacturing on supply chains: Regionalization, key research themes and future directions,” *Comput. Ind. Eng.*, vol. 184, p. 109600, 2023. <https://doi.org/10.1016/j.cie.2023.109600>
- [31] S. Schlichenmaier and T. Naegler, “May material bottlenecks hamper the global energy transition towards the 1.5 °C target?” *Energy Rep.*, vol. 8, pp. 14 875–14 887, 2022. <https://doi.org/10.1016/j.egy.2022.11.025>
- [32] V. Bellie, R. Gokulraju, C. Rajasekar, S. Vinoth, V. Mohankumar, and B. Gunapriya, “Laser induced breakdown spectroscopy for new product development in mining industry,” *Mater. Today Proc.*, vol. 45, no. 9, pp. 8157–8161, 2021. <https://doi.org/10.1016/j.matpr.2021.02.459>
- [33] B. C. Menezes, J. D. Kelly, and A. G. Leal, “Identification and design of industry 4.0 opportunities in manufacturing: Examples from mature industries to laboratory level systems,” *IFAC-PapersOnLine*, vol. 52, no. 13, pp. 2494–2500, 2019. <https://doi.org/10.1016/j.ifacol.2019.11.581>
- [34] M. Colledani, T. Tolio, A. Fischer, B. Iung, G. Lanza, R. Schmitt, and J. Váncza, “Design and management of manufacturing systems for production quality,” *CIRP Ann.*, vol. 63, no. 2, pp. 773–796, 2014. <https://doi.org/10.1016/j.cirp.2014.05.002>
- [35] L. Mayer, N. Mehdiyev, and P. Fettke, “Manufacturing execution systems driven process analytics: A case study from individual manufacturing,” *Procedia CIRP*, vol. 97, pp. 284–289, 2021. <https://doi.org/10.1016/j.procir.2020.05.239>
- [36] S. Tian, L. Wu, M. P. Ciano, M. Ardolino, and K. S. Pawar, “Enhancing innovativeness and performance of the manufacturing supply chain through datafication: The role of resilience,” *Comput. Ind. Eng.*, vol. 188, p. 109841, 2024. <https://doi.org/10.1016/j.cie.2023.109841>