Hindawi
Advances in Operations Research
Volume 2023, Article ID 2596069, 12 pages
https://doi.org/10.1155/2023/2596069

Research Article

Monitoring Big Data Streams Using Data Stream Management Systems: Industrial Needs, Challenges, and Improvements

Ahmad Alzghoul

Data Science Department, Princess Sumaya University for Technology, Amman, Jordan

Correspondence should be addressed to Ahmad Alzghoul; a.alzghoul@psut.edu.jo

Received 15 February 2023; Revised 23 March 2023; Accepted 12 June 2023; Published 27 June 2023

Academic Editor: Bui Thanh Khoa

Copyright © 2023 Ahmad Alzghoul. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Real-time monitoring systems are important for industry since they allow for avoiding unplanned system stops and keeping system availability high. The technical requirements for such systems include being both scalable and online, as the amount of generated data is increasing with time. Therefore, monitoring systems must integrate tools that can manage and analyze the data streams. The data stream management system is a stream processing tool that has the ability to manage and support operations on data streams in real-time. Several researchers have proposed and tested real-time monitoring systems which have the ability to search big data streams. In this paper, the research works that discuss the analysis of online data streams for fault detection in industry are reviewed. Based on the literature analysis, the industrial needs and challenges of monitoring big data streams are presented. Furthermore, feasible suggestions for improving the real-time monitoring system are proposed.

1. Introduction

In terms of availability, condition monitoring and fault diagnostics play important roles in increasing the quality of industrial products and systems. Early failure detection and diagnostics (FDD) have the potential to reduce failures of machinery and equipment, prevent major product breakdowns, reduce downtime, and allow maintenance to be scheduled, thus increasing product availability and reducing associated costs. [1–7] The work process of an FDD system generally involves fault detection, diagnostics, evaluation, and response action [8]. Fault detection methods can be divided into three categories: analytical methods, knowledge-based methods, and data-driven methods [9]. If one integrates several fault detection methods, the resulting system is called a *hybrid system* [10].

Analyzing data produced during the time of operation (and during maintenance) is the most common way to detect, predict, and avoid failures [11, 12]. According to the popular magazine The Economist [13], the rate of growth of the generated data is increasing, and the available storage capacity may not be able to cope with all the generated data. This is normally referred to as *big data*, which is simply characterized by the three Vs (volume, velocity, and variety). Furthermore, advanced technologies such as the Internet of Things (IoT) and cyber-physical systems (CPS) are significant sources of big data. As the number of connected devices and sensors continues to increase, so does the amount of data that is generated. An overview of the Internet of Things and its challenges and applications in various fields can be found in [14, 15]. Therefore, monitoring systems have to integrate tools which can analyze the data streams without the need to store it. A data stream is a continuous and ordered sequence of data that arrives in real-time [16]. It addresses the velocity dimension of big data. Several researchers have naturally discussed the issue of data streams in FDD, including [11, 17–26] (a more detailed review of related work is presented in Section 4).

In this paper, a summary of a literature review regarding processing data streams in real time for industrial fault detection is presented. The scope of the paper is the industrial application of the data stream management system (DSMS), industrial needs, and consequent requirements on the design of the future DSMS. Naturally, the author is aware of various technical developments regarding DSMS design, including works in [27–31]. However, such developments are not within the scope of this paper. The author has summarized and deduced the *industrial needs and challenges* that are presented when deploying the DSMS for monitoring big data streams. Furthermore, the author has had a longstanding collaboration with industrial partner companies regarding data stream management and analysis which is further discussed below.

Based on the analysis of the literature review and the author's previous work regarding industrial equipment monitoring with industrial partners, a scalable monitoring system is proposed. The system integrates various aspects of the reviewed monitoring systems as well as functionalities previously developed by the author and is based on industrial needs. The *design rationale*s, i.e., the reasons for including certain functionalities to meet specific industrial needs are summarized. The system (see Section 5), representing part of the paper results, benefits from using *historical data*, the ability to handle *online data streams*, and *forecasted data streams* to answer queries from multiple engineers (examples may include queries regarding *component wear*, *trends* indicating possible future failures, *increased energy usage*, and *increased temperature*). In addition, the proposed monitoring system suggests applying and integrating several fault detection methods, i.e., *analytical methods*, *knowledge-based methods*, and *data-driven methods*, thus enabling the most appropriate query formulation depending on the datasets and the industrial applications where they originate.

The author has a longstanding history in database research and experience from several international and national research projects (SmartVortex (http://www.smartvortex.eu), iStream (https://www.ltu.se/research/subjects/maskinkonstruktion/Forskningsprojekt/iSTREAMS-Storskalig-sokning-av-datastrommar-1.66982?l=en), and SSPI (https://www.ltu.se/research/subjects/maskinkonstruktion/Forskningsprojekt/iSTREAMS-Storskalig-sokning-av-datastrommar-1.66982?l=en)) related to equipment monitoring using data stream management system (DSMS) technology and data stream mining (DSM) in collaboration with mainly three Swedish companies. Section 2 discusses these applications in details. Section 3 presents the research method. Section 4 comprises a literature review and an analysis of the reviewed work. Section 5 presents and discusses the proposed monitoring system, and finally conclusions are presented in Section 6.

2. The Industrial Applications

To achieve products of high quality, industrial companies are interested in searching the data produced during the product's lifecycle. The author had the opportunity to collaborate with three Swedish companies, namely Bosch Rexroth Mellansel AB (BRMAB, formerly Hägglunds Drives AB, http://www.boschrexroth.com), Volvo Construction Equipment (VCE, http://www.volvoce.com), and AB Sandvik Coromant (SC, http://www.sandvik.coromant.com).

Firstly, BRMAB manufactures low-speed, high-torque hydraulic drive systems. They are interested in continuously analyzing their log files, future streams of raw data, and in some cases, in saving a summary of their data streams. In addition, they are interested in applying various fault detection methods to improve their equipment operation.

Secondly, Volvo CE develops, manufactures, and markets equipment for the construction and related industries. They are interested in monitoring sensor data streams including CANBUS data from, for example, wheel loaders or other products. If a deviation is detected a predetermined procedure is applied.

Thirdly, Sandvik Coromant is an engineering group involved in tooling, materials technology, mining, and construction. Sandvik Coromant is interested in analyzing streams of sensor readings from a mill. For example, they need to compare the expected power consumption of the mill using a mathematical model to the measured power consumption in order to detect any abnormal behavior. The next section discusses the research method of this work.

3. Research Method

This research work aims at reviewing and analyzing online data stream fault detection systems to identify industrial needs and challenges of monitoring big data streams and also, to provide feasible improvements to the reviewed real-time monitoring systems.

Concerning the literature review, previous monitoring systems which have the ability to analyze data streams in real time, were reviewed. The literature review was conducted using search terms such as "data stream management system," "monitoring system," "fault detection," and "data stream mining". Then, the industrial needs and challenges that researchers previously identified were deduced and summarized (see Section 4.1). Furthermore, the basis for the research presented in this paper is a longstanding collaboration (including collaborative development and workshops) with the industrial partner companies through several projects. The collaboration with the industrial partner companies has given the researcher good insights into the challenges within the different industries (more elaboration is provided in Section 4).

Based on the analysis of the literature review industrial needs and challenges were identified. Also, for each reviewed work, the fault detection method (i.e., analytical, knowledge-based, data-driven, or hybrid) and the data source (i.e., real-time data stream or forecasted data stream) were pointed out and used in a matrix. A matrix row represents one or more reviewed works, while a column represents the fault detection method or the data source (see Table 3 in Section 4.2). Based on the analysis of the matrix and the identified challenges, feasible suggestions to improve the DSMS-based real-time industrial fault detection were identified (further discussed in the last part of Section 4), and a DSMS-based monitoring system was proposed (see Section 5).

4. Literature Review and Analysis

Several researchers discuss the issue of searching data streams for fault detection and/or prediction. [11, 17–26, 32] developed a monitoring system based on data stream mining and DSMS technology and tested it on data collected from hydraulic motors. [32] developed a vehicle data stream (VEDAS) mining system for real-time vehicle health monitoring and driver characterization. [32] used a data stream management system to control the raw data stream generated by the monitored vehicle. The monitoring system proposed by [32] involved two modes: monitoring mode and training mode (algorithm training). [20] used a data stream mining method based on principal component analysis (PCA) and a binary support vector machine for cutting tool condition monitoring. [18] built and tested a monitoring system which is based on forecasted data streams for system fault prediction. [19] proposed a framework that combines fault detection, isolation, and correction (FDIC). The proposed FDIC used both model-based and classification-based methods for fault detection. A comparison between a knowledge-based and a data-driven method in analyzing data streams for system fault detection was made by [33]. A general approach for defining the correct behavior of the monitored equipment either analytically or statistically using a stream validator (SVALI) was validated in [25]. SVALIs support the use of data-driven and analytical-based fault detection methods through the *learn-and-validate* and the *model-and-validate* functions, respectively. [34] proposed and tested a general method to manage the problem of concept drift when using one-class, data-driven models for condition monitoring [34]. Gu et al. [35] proposed an online *failure detection system* using the IBM System S stream processing. The authors used the stream-based decision tree classifier as a fault detection method. Wheel loader slippage detection for Volvo Construction Equipment was tested and validated based on both knowledge-based and data-driven based models [36]. A distributed framework for streaming anomaly detection in embedded systems was proposed in [37]. They examine the effectiveness of their method using data from two sources: autonomous vehicle and advanced driver-assistance system (ADAS) platform. It was shown that the proposed method was able to detect anomalies with low latency. The authors in [36] used a DSMS-based framework and on-board sensor technology for clutch slippage detection and diagnosis. The Gaussian mixture model (GMM) and the logistics regression classifier were used for online anomaly detection while the diagnosis was done using case-based reasoning [38]. A visualization component was integrated into a DSMS in [39]. They implemented an operator that supports industrial analytics applications by executing query-based visualization methods over the data streams. The authors in [40] proposed an algorithm that can handle multiple concurrent data streams, which can be used for detecting contextual outliers. The performance of the algorithm was tested on real-world and synthetic datasets and showed a good performance. A hybrid deep learning classifier was used in [41] to detect the concept drifts in streaming data. The proposed approach is able to handle the time and memory constraints. A data-driven model was utilized in [42] to detect faults based on large-scale data sets from Metro do Porto subsystems.

The reviewed papers relate to a variety of industrial applications and have resolved several industrial needs and challenges. The industrial needs and challenges, identified through a focus on collaboration with industrial partners and a literature review, are discussed in Section 4.1. Further, the analysis of the applied fault-detection method is presented in Section 4.2.

4.1. Industrial Needs and Challenges. A summary of the arguments, challenges, and applications is presented in Table 1. Table 1 shows that real-time monitoring systems are needed for many applications such as monitoring heavy diesel engines, missile defense systems, vehicles, tooling machines (e.g., cutting), spacecraft, the steel industry, the metal industry, hydraulic systems, and milling processes. Furthermore, real-time monitoring systems were found to be required for other applications rather than industry processes or machine monitoring, such as software, driver characterization, and drinking water networks.

Several industrial needs were presented in the reviewed papers. Most of the papers agreed on the need for real-time monitoring systems to increase the availability of industrial systems and decrease the consequences of system failures. Based on Table 1, the industrial needs can be summarized in List 1 as follows:

4.1.1. List 1

(i) Use resources efficiently

(ii) Manufacture products of high quality

(iii) Increase production efficiency

(iv) Increase product and process availability

(v) Save on costs and time includes the following:

  (1) Save the costs and impact the readiness of the schedule-based preventive maintenance
  (2) Reduce the consequences of equipment failures in terms of time and cost
  (3) Reduce unscheduled machine down time
  (4) Minimize system down time for maintenance and reparations
  (5) Foresee failures and take maintenance action in advance of actual failures

(vi) Address the restrictive safety and environmental regulations

(vii) Achieve optimal performance of machining processes

(viii) Increase the efficiency of monitoring

(ix) The need for resource-constrained monitoring of time-critical data streams where central collection of data is an expensive proposition

(x) Vehicle health monitoring is an area of interest for NASA in terms of subsystems on the spacecraft

(xi) Need for online cutting tool condition monitoring

(xii) Reduce the consumption of communication resources in distributed data stream processing

Table 1: A summary of the reviewed papers.

References	Arguments	Challenges	Application areas
[19]	(i) A need of automatic fault detection for large and complex systems (ii) Address the restrictive safety and environmental regulations	(i) Large dimensionality of monitored variables (ii) High sampling rates (iii) Nonstationary patterns (iv) False alarms	Automotive engine test benches. Heavy diesel engine (caterpillar)
[26]	(i) Take maintenance action in advance of actual failures (ii) Minimize downtime and use resources efficiently (iii) Decrease costs and impact readiness of schedule-based preventive maintenance	Physical models of the covered structures in normal and anomalous states are unavailable or of limited fidelity	Missile defense system structural components
[23, 32]	The need of resource-constrained monitoring of time-critical data streams where central collection of data is an expensive proposition	Monitoring fleet of vehicles and associated data streams in a resource-constrained environment	Vehicle (ford taurus car) and driver characterization
[22]	(i) Achieve optimal performance of machining process (ii) Need of online cutting tool condition monitoring (iii) Cost saving	Real-time monitoring	Cutting tool machines
[24]	Vehicle health monitoring is an area of interest for NASA in terms of vital subsystems on the spacecraft	(i) Analyze large, complex, multivariate time-series in near-real time (ii) The dynamics of the system cannot be modeled	Spacecraft
[43]	(i) Increase the efficiency of monitoring (ii) Minimize system down time for repair and maintenance	(i) Limited expert knowledge (ii) Fault patterns not predefined (iii) Fault patterns cannot be simulated	Steel industry (metal sheet forming processes in rolling mills)
[44]	(i) Reduce unscheduled machine down time (ii) Decrease repair costs (iii) Increase production efficiency	(i) Curse of dimensionality (ii) Ideal time lag estimation (iii) Inclusion of output (error) feedback (iv) Structure identification (linearity versus non-linearity) (v) Parameter estimation	Metal industry and car engines
[22]	(i) Detect deviations and monitor machine health status (ii) Save damage costs	(i) Fast-arriving data from multiple sensors (ii) Rapid online and real-time analysis	General framework for machine monitoring
[11]	(i) Manufacture products of high quality (ii) Reduce the consequences of equipment failures in terms of time and cost	Monitor data stream in real time	Hydraulic systems
[17]	(i) Detecting failures at an early stage or foreseeing them before they occur is crucial for machinery availability (ii) Data prediction can reduce the consumption of communication resources in distributed data stream processing	(i) Real-time monitoring (ii) Failures may occur suddenly (in short time)	Hydraulic systems
[42]	Processing data streams from controllers and sensors is critical for monitoring the functional product in use	Scale up data analysis for handling huge amounts of equipment	Milling
[34]	Increasing product and process availability	Ability to search data streams while dealing with concept drift	Hydraulic systems
[33]	Increase the availability of industrial companies' products	Monitor data stream in real time	Hydraulic systems

Table 1: Continued.

References	Arguments	Challenges	Application areas
[35]	To achieve predictive failure management for fault-tolerant data stream processing	Providing lightweight failure prediction in an online and streaming setting	Software
[45]	The need of highly sophisticated supervisory and control schemes to satisfy a certain degree of performance when unfavorable conditions are occurring in critical infrastructure systems (CIS)	(i) Analytical models are not applicable (ii) Real-time monitoring	Drinking water network
[36, 38]	To deliver quality services for industrial equipment by continuously monitoring its behavior	(i) On-board condition monitoring (ii) Real-time sensor analysis (iii) Distributed data sources	Volvo CE wheel loaders
[37]	Provide a framework and taxonomy of anomaly symptoms for low latency online anomaly detection	(i) Real-time anomaly detection in embedded system	Autonomous vehicle/Advanced driver-assistance systems (ADAS)
[40]	Detecting outliers in multiple concurrent data streams	(i) Parallel processing for outlier detection in data streams	Detecting contextual outliers
[39]	Analyzing data streams in industrial processes and industrial cyber-physical systems	(i) Provide scalable capability to visualize the results from the analysis of data streams to support industrial needs	Industrial analytics applications
[41]	A method to handle nonstationary and dynamic data streams where the distributions are altered with the time	(i) Real-time applications with time and memory constraints	Applied on standard datasets from literature
[42]	Utilizing data-driven models for anomaly detection in the industrial area	(i) Large-scale data sets	Metro do porto subsystems

Table 2: A summary of the challenges that an industrial company may face when implementing the required monitoring systems.

Challenges	Industries	Technical development
Large dimensionality of monitored variables	x	
High sampling rates		x
Searching data streams in a resource-constrained environment	x	x
Handling huge amounts of equipment	x	x
Real-time monitoring		x
Expert knowledge is limited-fault patterns not predefined	x	
The dynamics of a system cannot be modeled (i.e., complex system)		x
Fault patterns expensive to simulate	x	
False alarms	x	x
Addressing concept drift		x
Model parameter estimation		x
Failures may occur suddenly (in short time)	x	x

(xiii) Processing data streams from controllers and sensors is critical for monitoring the functional product in use

List 1 shows that there is a need for real-time monitoring systems in many applications. However, Table 1 shows that industrial companies face numerous challenges in implementing the required monitoring systems. In Table 2 challenges are listed depending on who—industry or technical development—is concluded to be most responsible for addressing the challenge.

In order to resolve the challenges of large dimensionality, high sampling rates, handling huge amounts of equipment, and real-time monitoring, a developed monitoring system must be scalable. The use of a stream processing system such as DSMS can resolve the issue of scalability.

The ability to implement multiple fault detection methods overcomes the challenges of complex systems, the limitations of expert knowledge, and fault patterns that are not predefined or that cannot be simulated. For example, these challenges can be resolved by using an appropriate data-driven method. The concept drift challenge can be addressed by using an incremental algorithm or updating the fault detection model based on certain cases; see, for example, [34].

Reference [32] discussed how to search data streams in a resource-constrained environment. They proposed a framework which uses on-board PDA-like devices to run data stream mining and DSMS. They suggested using approximate algorithms to handle the limited computing power and memory. A central control station is also used to support the PDA in certain cases (i.e., not all the time to avoid the high communication cost and save energy).

Next, the applied fault detection methods found in the literature are reviewed and analyzed in Section 4.2.

4.2. The Applied Fault Detection Methods. Researchers have used a variety of fault detection methods based on the available resources for their applications. Stream processing systems have also been used to manage and control data streams online. Table 3 shows the fault detection methods and types of data that were used in the reviewed papers.

Table 3 shows that most of the monitoring systems developed to analyze data streams are based on data-driven methods. The data-driven methods are cheap, easy, and fast to implement compared to other fault detection methods [33]. Furthermore, fault detection functions that are based on data-driven methods can be updated automatically. The matrix presented in Table 3 shows that several gaps or research opportunities exist for which solutions are needed to be developed and tested. Few papers discuss the use of knowledge-based, analytical-based, or hybrid models. Furthermore, only one identified paper discussed the analysis of forecasted data streams for system fault prediction. The forecasted data stream was analyzed using a data-driven model. However, at the time of writing, there was no work identified that explored the analysis of forecasted data streams using the knowledge-based, analytical-based, or hybrid models used in conjunction. Also, no work proposed or discussed how to combine or utilize all the characteristics presented in Table 3. Thus, in this paper, a monitoring system which combines the characteristics presented in Table 3 is proposed. The next section presents and discusses the proposed monitoring system.

5. The Proposed Monitoring System

Based on the challenges presented in Section 4.1, the gaps identified in Section 4.2, and the insights gained from the collaborative development and workshops with several industrial partner companies, a monitoring system is proposed. The proposed monitoring system is capable of overcoming the challenges in Section 4.1, covering the gaps identified in Section 4.2, and providing the components that are important for engineers.

The proposed monitoring system, its features, and the connection between them are presented in Figure 1. The data stream management system (DSMS) holds most of the components due to its ability to manage data stream, implement multiple functions and queries, and provide interfaces and connections between multiple components.

The following subsections describe and discuss the functionality and usefulness of the proposed monitoring system in more detail.

Table 3: A comparison of the applied fault detection methods of the reviewed papers presented in the form of a matrix.

References	Data-driven model	Knowledge-based model	Analytical-based model	Hybrid model	Real-time data	Forecasted data
[11, 23, 32, 34, 35]	√				√	
[26] and others e.g. [21, 24]	√				√	
[19]	√		√	√	√	
[25]			√		√	
[17]	√				√	√
[34, 36]	√	√			√	

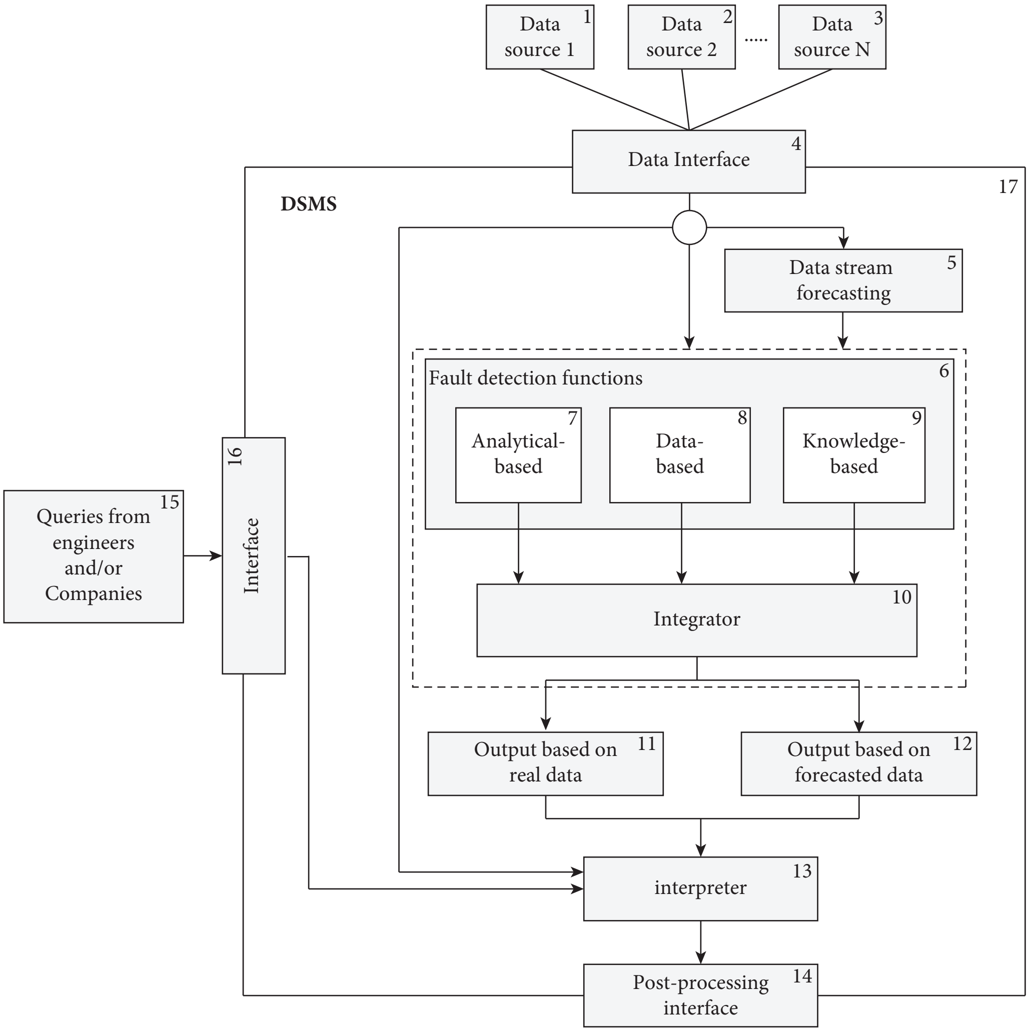

Figure 1: Proposed monitoring system.

5.1. Data Source and Data Interface. The terms data source and data interface refer to boxes numbers 1–3 and box number 4 in Figure 1, respectively. The lifecycle data of a product is important and useful for product monitoring. Metadata, historical data, and data collected in the design phase (e.g., the specification of the product's material such as max stress and temperature) can be saved in the DSMS's local memory if needed. The DSMS can control and manage the data generated during the operation and maintenance phases of the product. The data generated by products in use can have differing formats based on specific applications. For example, in the case of BRMAB Company, the data stream may arrive in the form of compressed log event files, produced from a variety of measurement points with multiple clocks. Therefore, an interface was needed to make the BRMAB data readable by the DSMS. Furthermore, metadata is an important issue when monitoring several machines. Metadata provides extra information about the arrived data. In the case of BRMAB, the metadata provided information about the name and place of the collecting unit, the number and names of the measured parameters, and the sampling rate of every parameter. The active mediators object system (AMOS) [47, 48] is an example of "mediato"' software that can be used to combine data from many different data sources, support the integration with other systems, and act as an intermediate level between data sources and their use in applications and by users.

5.2. DSMS: Data Stream Management Systems. According to [49], data stream management systems (box number 17 in Figure 1) represent an extension to the database management systems (DBMSs) that have the ability to manage and support operations on data streams. The structure of a general data stream management system is presented in Figure 2. Data stream management systems have the ability to handle data generated at high frequency in a fleet. Several papers, such as [11, 17, 18, 32, 50], showed the ability and usefulness of DSMS technology when integrated with the monitoring system.

Every DSMS has a query language which can be used to handle data streams and define queries. The DSMS's query language can be used to implement the data stream forecasting function (i.e., Box 5 in Figure 1), the fault detection functions (Box 6), the integrator (Box 10), the interpreter

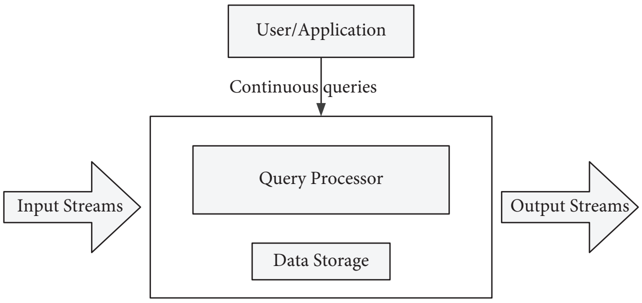

Figure 2: Data stream management system structure.

(Box 13), queries from engineers and companies (Box 15), and interfaces (Boxes numbers 4, 14, and 16). The implemented queries are executed by the DSMS's query processor, which applies continuous queries over the input data stream and streams the output to the user or to a temporary buffer.

The input streams in Figure 2 refer to the data streams generated from the data sources, i.e., Boxes 1–3 in Figure 1. The output streams in Figure 2 are refer to the output at the postprocessing interface in Figure 1 (i.e., Box 14). A DSMS commonly has a local data storage which can be used for several purposes, such as temporary working storage, stream synopses, and metadata [16].

5.3. Data Stream Forecasting. Data stream forecasting (Box 5 in Figure 1) uses historical data and/or one or more current data streams to forecast the future data stream. Data stream forecasting can be used for long- or short-term prediction. [17] proposed a fault prediction system based on data stream forecasting. They used a variety of data stream prediction approaches based on linear regression for short- and long-term prediction. System fault prediction was achieved by applying the forecasted data on a fault detection function. The results of [17] showed good performance in short-term prediction.

A query can be implemented to apply the data stream forecasting method. The query will then be applied continuously over the arrived data stream through the DSMS's query processor.

Data stream forecasting is useful as it can help in detecting failures early and increases the response time, which is important, especially in cases of failures which occur suddenly or in a short time (such as seizures in the case of BRMAB). Using data stream forecasting for fault prediction allows the support system to gain a longer reaction time to handle early warnings of impending failures and thus increase system availability. In addition, the communication consumption in distributed data stream processing can be reduced through the use of data prediction [51].

5.4. Fault Detection Functions. The fault detection function (Box 6 in Figure 1) concerns the identification of fault occurrence, referring specifically in this paper to industrial equipment faults or failures. According to [9], fault detection methods can be classified into three categories: analytical-based methods, knowledge-based methods, and data-driven methods. The three categories of fault detection methods have differing advantages and disadvantages and may not be applicable in all cases [9]. For example, data-driven methods are applicable for both small and complex systems, whereas analytical-based methods are applicable for small systems [9]. On the other hand, analytical-based methods achieve higher accuracy than data-driven methods in detecting failures [9]. Knowledge-based methods are good at detecting preknown failures but not in detecting new types of failures, while data-driven methods are good at detecting new types of failures [52]. A comprehensive study and comparison between knowledge-based and data-driven methods can be found in [33].

In data-driven methods, data stream mining algorithms are useful in monitoring systems which concern data streams. These algorithms have the ability to extract patterns from continuous and fast-arriving data streams. A review of data stream mining algorithms and their application in monitoring systems was conducted by [11].

With regards to the three industrial companies and applications developed by the author, the methods used were of varying types, due to reasons which are discussed below. An analytically based method was used with data from AB Sandvik Coromant [42], whereas data-driven methods were used with data from both Volvo CE and BRMAB [11]. The author was also able to develop a knowledge-based fault detection model in the case of BRMAB [33]. A comparison between the results of applying the knowledge-based and data-driven methods for the BRMAB case can be found in [33]. According to [9], analytical-based methods achieve higher accuracy than other methods. In general, one might conclude that it is wise to consider using an analytically based method first if possible. In the case of AB Sandvik Coromant it was possible to develop such a model. In the case of BRMAB, it was not possible to use an analytically based method due to the complexity of hydraulic motors; therefore, both a data-driven method and a knowledge-based method were used. In the case of Volvo CE a data-driven method was used.

In monitoring systems, it is common to use each fault detection method separately. However, a number of researchers, as discussed in Section 5.5, have integrated individual fault detection methods. Section 5.5 discusses the significant role of the integrator and how it could help improve the performance of the monitoring system.

5.5. Integrator. The integrator component (Box 10 in Figure 1) concerns the way in which the various fault detection methods will be incorporated. The integrator can be seen as a predefined function that produces an output based on the applied fault detection methods. For example, [52] used Bayesian ranking inference to integrate a knowledge-based and a data-driven method. They used a knowledge-based method to detect preknown fault types, while the data-driven method was used to detect preknown and unknown fault types. Using expert system technology, Leung & Romagnoli 53 integrated multivariate statistical process control (MSPC) into knowledge-based fault diagnosis, leading to a more accurate diagnosis. Norvilas et al. 54 used the G2 real-time KBS shell to integrate multivariate statistical process monitoring (MSPM) techniques and knowledge-based systems (KBS). The MSPM was used for fault detection, while the KBS was responsible for determining the response actions to the detected faults and sending the corresponding alert messages to the operator. These examples showed the importance of integrating multiple fault detection and diagnosis methods. The integrator (Box 10 in Figure 1) is intended to be formulated into a CQ which will be applied to the arriving data streams.

Note that in the proposed monitoring system of Figure 1, there are two separate sources of data for the fault detection function component. The two input sources are the online real-time data stream and the forecasted data stream. The CQs of the fault detection function (including the integrator) will be applied to both sources of data in parallel, producing two types of outputs: output based on real data and output based on forecasted data.

5.6. Queries from Engineers. The proposed monitoring systems can be used to answer other queries ("questions") from engineers (Box 15 in Figure 1), provide information needed for the industrial company, and provide several services on request. [18], discussed how the services can be obtained using data stream mining and DSMS technology. They proposed an approach has the potential to significantly support continuous availability awareness in industrial systems.

The engineers' queries can be written using the DSMS's query language. The interpreter through the DSMS's query processor can then be applied to the engineers' queries continuously over the input data stream. The query output can be visualized through a GUI, dashboard, or used in the interpreter component of the proposed monitoring system, see Figure 1. Examples of such queries and how to visualize them can be found in [49]. They determined key queries (supporting industry partner engineers), such as monitoring and predicting the customer/operator usage and monitoring system/product performance, and showed how to visualize such tasks.

5.7. Interpreter. The interpreter component (Box 13 in Figure 1) is basically a set of queries which are executed by the DSMS's query processor. It receives several data stream inputs such as the outputs from the fault detection system (output based on real data and output based on forecasted data) and the monitored data stream, and reads queries from engineers. In addition, the interpreter may use information saved in the local memory of the DSMS.

The interpreter has several tasks. It has to interpret the output from the fault detection function and issue the corresponding response action. For example, if there is an uncritical problem such as the cooler functionality not being efficient, then the interpreter has to send an alert message to the operator. If there is a critical fault which may damage the machine, then the interpreter has to take a response action such as switching off the machine and also send a corresponding alert message. The interpreter may also answer engineers' queries, as in [49] which require the outputs from the fault detection function, the monitored data, and/or information saved in the local memory. The outputs of the interpreter are then passed to a postprocessing interface.

The variant information that the interpreter receives can be utilized to gain several advantages such as follows:

(i) Allows using one fault detection method or more to monitor the different system components

(ii) Increases the accuracy and robustness of the prediction by comparing the output of different fault detection methods

(iii) Allows short- or long-term fault prediction by utilizing the forecasted data

(iv) The forecasted data might be utilized to detect the concept drift

(v) Increases the ability to answer the various engineers' queries

5.8. Interfaces. Interfaces (Boxes 4, 14, and 16 in Figure 1), are used to facilitate the interactions between the different components. For example, the interface between the data source and the DSMS is used to make the data format readable by the DSMS. On the other hand, a graphical user interface makes the interaction between the engineers or operators, and the monitoring system more user-friendly (Boxes 14 and 16 in Figure 1). [18] discussed the importance of developing a flexible GUI which is able to visualize the engineer queries and the corresponding results.

The postprocessing interface (Box 14, Figure 1) can be used to visualize the answers to the queries and the real and expected values of various parameters, to send alert messages and take necessary actions, and to create a customized dashboard. A proof-of-concept of such a component was implemented and evaluated in [39]. The authors showed how to integrate query-based visualization methods into a DSMS to visualize the query results of industrial data streams.

6. Conclusions

In this paper, a summary of a literature review regarding processing data streams in real time for industrial fault detection is presented. The scope of the paper is the industrial application of data stream analysis, industrial needs, and consequent requirements on the design of the future DSMS. For this paper, the author has summarized and deduced the *industrial needs and challenges* that other researchers have previously identified. Most importantly, the author has had a longstanding collaboration with industrial partner companies regarding data stream managing and analysis.

Furthermore, suggestions for improving monitoring systems were discussed. Then a DSMS-based monitoring system was proposed to show how the suggestions can be implemented. The proposed monitoring system benefits from integrating multiple fault detection methods, i.e., analytical methods, knowledge-based methods, and data-driven methods, and using historical data, online real-time data streams, and forecasted data streams.

Data Availability

No underlying data were collected or produced in this study.

Disclosure

This work is based on conclusions and tests from the author's previous research (which was funded by the SSPI(SSF) and the iStreams (VINNOVA) projects).

Conflicts of Interest

The authors declare that they have no conflicts of interest.

References

[1] K. Alhamad, M. Alardhi, and A. Almazrouee, "Preventive maintenance scheduling for multicogeneration plants with production constraints using genetic algorithms," *Advances in Operations Research*, vol. 2015, Article ID 282178, 12 pages, 2015.

[2] I. Alsyouf and A. Alzghoul, "Soft computing applications in wind power systems: a review and analysis," in *Proceedings of the European Offshore Wind Conference and Exhibition*, Stockholm, Sweden, March, 2009.

[3] A. Alzghoul, A. Jarndal, I. Alsyouf, A. A. Bingamil, M. A. Ali, and S. AlBaiti, "On the usefulness of pre-processing methods in rotating Machines faults classification using artificial neural network," *Journal of Applied and Computational Mechanics*, vol. 7, no. 1, pp. 254–261, 2021.

[4] J. D. Andrews and T. R. Moss, *Reliability and Risk Assessment*, Professional Engineering Publishing Limited, Bury St Edmunds, UK, 2002.

[5] M. Cheshmberah, A. Naderizadeh, A. Shafaghat, and M. Nokabadi, "An integrated process model for root cause failure analysis based on reality charting, FMEA and DEMATEL," *International Journal of Data and Network Science*, vol. 4, no. 2, pp. 225–236, 2020.

[6] M. Ram and S. B. Singh, "Availability and cost analysis of a parallel redundant complex system with two types of failure under preemptive-resume repair discipline using gumbel-hougaard family copula in repair," *International Journal of Reliability, Quality and Safety Engineering*, vol. 15, no. 04, pp. 341–365, 2008.

[7] N. Ren, Y. Wang, and S. Gan, "A condition-based maintenance policy (CBM) of repairable multi-component deteriorating systems based on quality information," *International Journal of Reliability, Quality and Safety Engineering*, vol. 27, no. 01, Article ID 2050002, 2020.

[8] D. G. Arseniev, B. E. Lyubimov, and V. P. Shkodyrev, "Intelligent fault detection and diagnostics system on rule-based neural network approach," in *Proceedings of the 2009 IEEE Control Applications, (CCA) & Intelligent Control, (ISIC)*, St. Petersburg, Russia, July, 2009.

[9] L. H. Chiang, E. Russell, and R. D. Braatz, *Fault Detection and Diagnosis in Industrial Systems*, Springer Verlag, Berlin, Germany, 2001.

[10] V. Venkatasubramanian, R. Rengaswamy, S. N. Kavuri, and K. Yin, "A review of process fault detection and diagnosis: Part III: process history based methods," *Computers and Chemical Engineering*, vol. 27, no. 3, pp. 327–346, 2003.

[11] A. Alzghoul and M. Löfstrand, "Increasing availability of industrial systems through data stream mining," *Computers and Industrial Engineering*, vol. 60, no. 2, pp. 195–205, 2011.

[12] R. Isermann, *Fault-diagnosis Applications: Model-Based Condition Monitoring: Actuators, Drives, Machinery, Plants, Sensors, and Fault-Tolerant Systems*, Springer, Berlin, Germany, 2011.

[13] K. Cukier, "Data, data everywhere: a special report on managing information," Economist Newspaper, London, UK, 2010.

[14] H. Baziyad, V. Kayvanfar, and A. Kinra, "The Internet of Things—an emerging paradigm to support the digitalization of future supply chains," in *The Digital Supply Chain*, pp. 61–76, Elsevier, 2022.

[15] P. Dudhe, N. Kadam, R. Hushangabade, and M. Deshmukh, "Internet of Things (IOT): An Overview and its Applications," in *Proceedings of the 2017 International Conference on Energy, Communication, Data Analytics and Soft Computing (ICECDS)*, Chennai, India, August, 2017.

[16] L. Golab and M. T. Özsu, "Issues in data stream management," *ACM Sigmod Record*, vol. 32, no. 2, pp. 5–14, 2003.

[17] A. Alzghoul, M. Löfstrand, and B. Backe, "Data stream forecasting for system fault prediction," *Computers and Industrial Engineering*, vol. 62, no. 4, pp. 972–978, 2012.

[18] A. Alzghoul, M. Löfstrand, L. Karlsson, and M. Karlberg, "Data stream mining for increased functional product availability awareness," in *Functional Thinking for Value Creation*, J. Hesselbach and C. Herrmann, Eds., Springer, Berlin, Germany, pp. 237–241, 2011.

[19] P. Angelov, V. Giglio, C. Guardiola, E. Lughofer, and J. M. Luján, "An approach to model-based fault detection in industrial measurement systems with application to engine test benches," *Measurement Science and Technology*, vol. 17, no. 7, pp. 1809–1818, 2006.

[20] C. Karacal, S. Cho, and W. Yu, "Sensor stream mining for tool condition monitoring," in *Proceedings of the 2009 International Conference on Computers and Industrial Engineering*, pp. 1429–1433, Troyes, France, July, 2009.

[21] S. C. Karacal, "Data stream mining for machine reliability," in *Proceedings of the IIE Annual Conference and Exhibition*, San Juan, Puerto Rico, May, 2006.

[22] S. C. Karacal, "Mining machine data streams using statistical process monitoring techniques," in *Proceedings of the IIE Annual Conference and Exhibition*, Cancún, Mexico, June, 2007.

[23] H. Kargupta, V. Puttagunta, M. Klein, and K. Sarkar, "On-board vehicle data stream monitoring using mine-fleet and fast resource constrained monitoring of correlation matrices," *New Generation Computing*, vol. 25, no. 1, pp. 5–32, 2006.

[24] B. Matthews and A. N. Srivastava, "Comparative analysis of data-driven anomaly detection methods," in *Proceedings of the JANNAF Conference on Propulsion Systems, Proceedings of the Joint Army Navy NASA Air Force Conference on Propulsion*, Orlando, FL, USA, December, 2008.

[25] C. Xu, D. Wedlund, M. Helgoson, and T. Risch, "Model-based validation of streaming data," in *Proceedings of the 7th ACM international conference on Distributed event-based systems*, USA, June, 2013.

[26] R. Youree, J. Yalowitz, A. Corder, and T. Ooi, "A multivariate statistical analysis technique for on-line fault prediction," in

Proceedings of the 2008 International Conference on Prognostics and Health Management, Denver, CO, USA, October, 2008.

[27] E. Bauleo, S. Carnevale, T. Catarci, S. Kimani, M. Leva, and M. Mecella, "Design, realization and user evaluation of the SmartVortex Visual Query System for accessing data streams in industrial engineering applications," *Journal of Visual Languages and Computing*, vol. 25, no. 5, pp. 577–601, 2014.

[28] M. Hirzel, R. Soulé, S. Schneider, B. Gedik, and R. Grimm, "A catalog of stream processing optimizations," *ACM Computing Surveys*, vol. 46, no. 4, pp. 1–34, 2014.

[29] M. Ivanova and T. Risch, "Customizable parallel execution of scientific stream queries," in *Proceedings of the 31st International Conference on Very Large Data Bases*, Trondheim, Norway, August, 2005.

[30] T. Risch, V. Josifovski, and T. Katchaounov, *Functional Data Integration in a Distributed Mediator System*, Springer, Berlin, Germany, 2003.

[31] E. Zeitler and T. Risch, "Massive scale-out of expensive continuous queries," *Proceedings of the VLDB Endowment*, vol. 4, no. 11, pp. 1181–1188, 2011.

[32] H. Kargupta, "Vedas: a mobile and distributed data stream mining system for real-time vehicle monitoring," in *Proceedings of the Fourth SIAM International Conference on Data Mining*, pp. 300–311, Lake Buena Vista, FL, USA, April, 2004.

[33] A. Alzghoul, B. Backe, M. Löfstrand, A. Byström, and B. Liljedahl, "Comparing a knowledge-based and a data-driven method in querying data streams for system fault detection: a hydraulic drive system application," *Computers in Industry*, vol. 65, no. 8, pp. 1126–1135, 2014.

[34] A. Alzghoul and M. Löfstrand, "Addressing concept drift to improve system availability by updating one-class data-driven models," *Evolving Systems*, vol. 6, no. 3, pp. 187–198, 2014.

[35] X. Gu, S. Papadimitriou, S. Y. Philip, and S. P. Chang, "Online failure forecast for fault-tolerant data stream processing," in *Proceedings of the 2008 IEEE 24th International Conference on Data Engineering*, pp. 1388–1390, IEEE Computer Society, Massachusetts, NW Washington, USA, April 2008.

[36] C. Xu, E. Källström, T. Risch, J. Lindström, L. Håkansson, and J. Larsson, "Scalable validation of industrial equipment using a functional DSMS," *Journal of Intelligent Information Systems*, vol. 48, no. 3, pp. 553–577, 2017.

[37] S. Kauffman, M. Dunne, G. Gracioli, W. Khan, N. Benann, and S. Fischmeister, "Palisade: a framework for anomaly detection in embedded systems," *Journal of Systems Architecture*, vol. 113, Article ID 101876, 2021.

[38] E. Källström, T. Olsson, J. Lindström, L. Håkansson, and J. Larsson, "On-board clutch slippage detection and diagnosis in heavy duty machine," *International Journal of Prognostics and Health Management*, vol. 9, 2020.

[39] M. Ram and S. B. Singh, "Visualisation of numerical query results on industrial data streams," in *Proceedings of the New trends in database and information systems: ADBIS 2022 short papers, doctoral consortium and workshops: DOING, K-gals, MADEISD, MegaData, SWODCH*, turin, Italy, September, 2022.

[40] M. Cheshmberah, A. Naderizadeh, A. Shafaghat, and M. Karimi Nokabadi, "A GPU algorithm for detecting contextual outliers in multiple concurrent data streams," in *Proceedings of the 2021 IEEE international conference on big data (big data)*, Orlando, FL, USA, December, 2021.

[41] D. K. Talapula, A. Kumar, K. K. Ravulakollu, and M. Kumar, "A hybrid deep learning classifier and Optimized Key Windowing approach for drift detection and adaption," *Decision Analytics Journal*, vol. 6, Article ID 100178, 2023.

[42] N. Davari, B. Veloso, R. P. Ribeiro, and J. Gama, "Fault forecasting using data-driven modeling: a case study for Metro do Porto data set machine learning and principles and practice of knowledge discovery in databases," in *Proceedings of the international workshops of ecml pkdd 2022*, Grenoble, France, September, 2023.

[43] F. Serdio, E. Lughofer, K. Pichler, T. Buchegger, and H. Efendic, "Data-driven residual-based fault detection for condition monitoring in rolling mills," *IFAC Proceedings Volumes*, vol. 46, no. 9, pp. 1530–1535, 2013.

[44] F. Serdio, E. Lughofer, K. Pichler, T. Buchegger, and H. Efendic, "Residual-based fault detection using soft computing techniques for condition monitoring at rolling mills," *Information Sciences*, vol. 259, pp. 304–320, 2014.

[45] J. Quevedo, C. Alippi, M. A. Cuguero et al., "Temporal/spatial model-based fault diagnosis vs. hidden Markov models change detection method: application to the Barcelona water network," in *Proceedings of the 21st Mediterranean Conference on Control and Automation*, pp. 394–400, IEEE, Platanias, Greece, June 2013.

[46] C. Xu, D. Wedlund, M. Helgoson, and T. Risch, "Model-based validation of streaming data:(industry article)," in *Proceedings of the 7th ACM International Conference on Distributed Event-Based Systems*, Orlando, FL, USA, October, 2013.

[47] G. Fahl, T. Risch, and M. Sköld, *AMOS: an Architecture for active mediators*, Universitetet i linköping, Linköping, Sweden, 1993.

[48] M. H. Staffan Flodin, V. Josifovski, T. Katchaounov, T. Risch, S. Martin, and E. Zeitler, "Amos ii release 14 user's manual. uppsala database laboratory," 2012, http://www.it.uu.se/research/group/udbl/amos/doc/amos_users_guide.html.

[49] G. Hébrail, "Data stream management and mining," *Mining Massive Data Sets for Security*, vol. 19, pp. 89–102, 2008, https://www.google.jo/books/edition/Mining_Massive_Data_Sets_for_Security/4EjpqmwhD7UC?hl=en&gbpv=1.

[50] A. Alzghoul, *Mining Data Streams to Increase Industrial Product Availability*, Luleå University of Technology, Luleå, Sweden, 2013.

[51] L. Tian, A. Li, and P. Zou, "Research on prediction models over distributed data streams," in *Web Information Systems – WISE 2006 Workshops*, L. Feng, G. Wang, C. Zeng, and R. Huang, Eds., vol. 4256, pp. 25–36, Springer, Berlin, Germany, 2006.

[52] F. Chih-Min and L. Yun-Pei, "A Bayesian framework to integrate knowledge-based and data-driven inference tools for reliable yield diagnoses," in *Proceedings of the 2008 Winter Simulation Conference*, Miami, FL, USA, December, 2008.

[53] D. Leung and J. Romagnoli, "An integration mechanism for multivariate knowledge-based fault diagnosis," *Journal of Process Control*, vol. 12, no. 1, pp. 15–26, 2002.

[54] A. Norvilas, A. Negiz, J. DeCicco, and A. Çinar, "Intelligent process monitoring by interfacing knowledge-based systems and multivariate statistical monitoring," *Journal of Process Control*, vol. 10, no. 4, pp. 341–350, 2000.

[55] S. Kimani, M. Leva, M. Mecella, and T. Catarci, "Visualization of multidimensional sensor data in industrial engineering," in *Proceedings of the 2013 17th International Conference on Information Visualisation*, London, UK, July, 2013.