System integration for predictive process adjustment and cloud computing-based real-time condition monitoring of vibration sensor signals in automated storage and retrieval systems

Sujeong Baek¹

Received: 18 August 2020 / Accepted: 18 January 2021 / Published online: 29 January 2021
© The Author(s) 2021

Abstract

As automation and digitalization are being increasingly implemented in industrial applications, manufacturing systems comprising several functions are becoming more complex. Consequently, fault analysis (e.g., fault detection, diagnosis, and prediction) has attracted increased research attention. Investigations involving fault analysis are usually performed using real-time, online, or automated techniques for fault detection or alarming. Conversely, recovery of faulty states to their healthy forms is usually performed manually under offline conditions. However, the development of intelligent systems requires that appropriate feedback be provided automatically, to facilitate faulty-state recovery without the need for manual operator intervention and/or decision-making. To this end, this paper proposes a system integration technique for predictive process adjustment that determines appropriate recovery actions and performs them automatically by analyzing relevant sensor signals pertaining to the current situation of a manufacturing unit via cloud computing and machine learning. The proposed system corresponds to an automated predictive process adjustment module of an automated storage and retrieval system (ASRS). The said integrated module collects and analyzes the temperature and vibration signals of a product transporter using an internet-of-things-based programmable logic controller and cloud computing to identify the current states of the ASRS system. Upon detection of faulty states, the control program identifies corresponding process control variables and controls them to recover the system to its previous no-fault state. The proposed system will facilitate automatic prognostics and health management in complex manufacturing systems by providing automatic fault diagnosis and predictive recovery feedback.

Keywords Automated storage and retrieval system · Cloud computing · Condition monitoring · Predictive process adjustment · Vibration sensor

1 Introduction

Industrial implementation of automation and digitalization makes manufacturing more complex and diverse. A manufacturing system includes several sub-functions [1, 2], such as planning [3], scheduling [4–7], dispatching [8], machine maintenance [9, 10], and quality control and inspection [11, 12]. In particular, to facilitate the effective operation of a manufacturing process, automated predictive maintenance operations are preferentially performed via fault detection,

diagnosis, and predictions based on sensor signals collected during process execution [9, 10, 13–15]. For example, Lu et al. [16] detected occurrences of bearing faults during operation under harsh conditions (i.e., low signal-to-noise ratio) by applying adaptive stochastic resonance. They installed vibration and acoustic sensors on main shaft bearings and collected corresponding signals. Subsequently, considering enveloping, they fused two types of signals—root mean square and those with peak signal-to-noise ratio—and considered the optimum dominant frequency to identify distinguishable characteristic indices of healthy and faulty system states. To detect faults in wafers, Lee et al. [17] developed denoised autoencoders, which are a type of deep-learning algorithms. The autoencoders were stacked in layers, thereby allowing the output layer of a previous model to be reused as the input layer of the next for pre-training and fine-tuning. Finally, they classified input signals into those corresponding to no-fault and

✉ Sujeong Baek
sbaek@hanbat.ac.kr

¹ Department of Industrial and Management Engineering, Hanbat National University, 34158, Room. 211, Bldg. N4, 125 Dongseodae-ro, Yuseong-gu, Republic of Korea

wafer faults based on fine-tuned parameters pertaining to the trained autoencoder. Post fault detection in a machine or facility, fault diagnosis was performed by way of root cause analysis, and fault identification was achieved through sensor signal analyses, for effective maintenance. In another study, vibration sensor signals were collected and analyzed to assess damage levels in a planetary gearbox [18]. A vibration signal model based on amplitude and frequency modulations was employed to explain vibration signal behaviors when either a ring, planetary, or sun gear was damaged. By calculating the characteristic frequencies of each component using the model and given sensor signals, the status of each gear was classified as non-faulty or as a particular fault type.

Once a user is notified of the occurrence of a faulty state by the included sub-functions of a manufacturing system, informative feedback can be automatically provided to facilitate recovery of the system to its healthy state [19–21]. If this step is included in predictive maintenance, the maintenance strategy is termed “predictive process adjustment” [22–24]. To execute predictive process adjustment in a manufacturing process, users must not only automatically identify the current process status but also handle control parameters as dynamic variables, the values of which must be changed based on user intentions and/or environmental conditions. In particular, real-time monitoring and automatic recovery have an important bearing on product lead times, which, in turn, reflect a company’s productivity.

González-Prieto et al. [25] detected open-switch-type faults in an induction motor drive by analyzing the corresponding machine model and energy conversion process. They monitored phase current signals and performed online determination of fault indices. Post detection of fault occurrences, they controlled motor drive parameters (motor speed and current) to recover the system to its healthy state. Mani and Sivaraman [26] employed a fuzzy decision-making system to classify fault types. Based on predefined fault detection rules, their proposed fuzzy classifier computed output variables to classify the current system state as healthy or faulty. The output variables determined an appropriate control action along with an input signal. The performance of this system was validated using a state space model for a continuous stirred tank reactor process.

Although it is important to provide appropriate feedback automatically to systems to facilitate their timely recovery to their healthy state, real-time health management of a manufacturing system is yet exclusively limited to fault detection/alarming and/or root cause analysis. This is because without robust models validated using relevant mathematical principles, it is difficult to execute automatic adjustment of controls on complex systems comprising several machine elements. To overcome this limitation, this paper proposes a system integration technique to perform predictive process adjustment in manufacturing systems via remote real-time condition monitoring of vibration sensors. We developed an

automated storage and retrieval system (ASRS) for validating the proposed framework using a motion control module of an industrial controller (i.e., a programmable logic controller (PLC)). In the proposed system, cloud computing is introduced for automatic data collection and real-time monitoring to facilitate fault state analysis using different fault diagnosis techniques. The use of cloud computing and the internet-of-things (IoT)-based PLC enables adjustment of the relevant control parameters of the system as well as application of various statistical and machine-learning-based fault detection methods, without the need for any complicated settings.

The remainder of this paper is organized as follows. Section 2 describes the ASRS developed in this study as a typical automation facility for implementing the proposed predictive process adjustment strategy. Operating sequences within a system, as well as an equipped controller, data acquisition system, and attached sensors are described in detail. Section 3 describes the proposed framework for predictive process adjustment of the constructed system, including details concerning faulty state detection and real-time system recovery. Section 4 presents and discusses the experimental results obtained using the proposed system. Finally, the conclusions drawn from this study and the scope for future research are presented in Section 5.

2 Description of the existing automated storage and retrieval system

In this study, an existing ASRS was modified to develop the proposed real-time remote predictive process adjustment framework. As illustrated in Fig. 1, the existing ASRS comprised five major components—(i) product entrance table, (ii) linear transporter (feeder with two threads and AC motors), (iii) product storage, (iv) industrial controller (PLC), and (v) human–machine interface (HMI). When products on a pallet are placed on the ASRS entrance table, a proximity sensor automatically detects the pallet and sends a signal to the industrial controller, which actuates two DC motors and corresponding threads to extend a sliding feeder of the linear transporter. The feeder locates the appropriate position for pallet handling. Subsequently, the feed rotates 180° to face the product storage, and the linear transporter moves to the predefined HMI-managed product storage position. In cases involving product retrieval/dispensing, the operator first selects the stored product via the HMI. Subsequently, the linear transporter moves to the determined position and product entrance table to facilitate product delivery to the operator or next manufacturing process. The product storage comprises 15 shelves, and each shelf is fitted with a proximity sensor to detect whether a pallet is stored therein. The information received by all proximity sensors can be monitored in real time on the HMI.

Fig. 1 Existing ASRS for storage and retrieval of ordered products for users or subsequent manufacturing process

A PLC module (developed by Mitsubishi) was used in this study as the main ASRS controller. As depicted in Fig. 2, the PLC included two card types for controlling and communicating with the ASRS actuators. A digital input–output (DI/DO)

communication card was used for receiving digital input signals from proximity sensors attached to the product entrance table, storage shelves, and reed switches attached to the feeder (for sliding and rotating), as well as for controlling feeder

Fig. 2 Conventional ASRS control and communication architecture

extension, retraction, and rotation. Moreover, motion control cards were used for controlling the operation of two AC servomotors along two orthogonal axes (i.e., vertical and lateral movements) as well as receiving built-in encoder signals. The HMI was connected to the PLC via Ethernet to facilitate simultaneous reception of information provided by any sensor or actuator. In addition, the HMI generated a control signal from a graphical user interface to retrieve any pallet from the product storage or control any single actuator.

3 Proposed automatic predictive process adjustment framework

Figure 3 illustrates the proposed real-time condition monitoring and automatic process control framework for automated manufacturing systems. The proposed predictive process adjustment strategy can be divided into two phases—offline analysis and online monitoring/feedback. This study focuses on the development of the online remote-monitoring and feedback phase; offline analysis can be configured based on simple signal analyses or expert knowledge-based rules. However, because it functions as an independent module, the offline analysis phase can be easily replaced by complex fault detection, diagnosis, or prediction algorithms.

In accordance with the predictive process adjustment strategy, the architecture for online condition monitoring and process

control adjustment can be described as illustrated in Fig. 3. When an automated manufacturing system is powered on and completes initialization, it begins executing assigned tasks. If any fault detection or prediction rule is activated during system operation in the condition-monitoring phase, the corresponding fault state is identified, and an appropriate adjustment is automatically performed. After the said adjustment, the manufacturing system continues operation in its healthy state.

It is possible to generate several fault detection or prediction rules along with corresponding process adjustments. As already mentioned, any fault detection/prediction rules and resolution strategies can be employed in the offline fault analysis phase. From expert knowledge-based models to complex signal-processing and machine-learning-based fault detection/prediction methods, any fault diagnosis approach with historical time-series data can be used, because current sensor signals can be compared against adjustable variables through cloud computing and the IoT-based PLC. In other words, adjustment #1 is performed using a statistical distance-based approach to detect fault type 1, whereas adjustment #2 is performed from a deep neural network to detect fault types 2 and 3 together.

At this point, the proposed automatic predictive process adjustment is conducted. The adjustment commands derived from the fault detection models are sent through the IoT gateway (via Ethernet communication) to the main PLC (via Modbus). Then, the corresponding parameters (to the given adjustment commands) in the main PLC are re-optimized and the actuators are re-controlled for preventing fault states in the system.

Fig. 3 Proposed maintenance strategy for predictive process adjustment using sensor signals collected from automation systems

Table 1 Specifications of the installed analog sensor

Sensor model	Power requirement	Connection	I/O
QM42VT2	DC 3.6–5.5 V or DC 10–24 V	Cable with a 5-pin M12/Euro-style male quick disconnect	Vibration and temperature via RS-485 Modbus

In addition, we could analyze sensor signals, which were stored in local or cloud databases, using interlocking application programming interfaces (APIs), and could thereby provide informative feedback to control process parameters or logic.

4 Experimental results and discussion for proposed automatic predictive process adjustment framework

4.1 Experimental setup for remote real-time predictive process adjustment system of ASRS

Because the existing automation system receives digital input signals comprising zeros and ones, we installed additional analog sensors (QM42VT2; Banner Engineering Corp.; MN, USA), the details of which are listed in Table 1. Sensors installed on AC motors corresponding to each axis of the linear transporter can measure various types of data simultaneously (maximum of 25 variables), such as temperature, velocity, and peak acceleration. Among several raw signals, we measured root mean square (RMS) data for each axis (X - and Z -axes) to obtain refined and pre-processed vibration signals, based on the specifications summarized in Table 2. In addition, temperature signals from the sensors were simultaneously collected. After measurement of vibration, analog signals were exported and collected using an IoT-based PLC (Hans Turck GmbH & Co. KG; Germany) prior to being transferred to an IoT gateway (Hans Turck GmbH & Co. KG; Germany), as depicted in Fig. 4. In addition, the collected datasets were stored in a cloud database as an $n \times 4$ array, where n indicates the length of the time series in either a normal or an abnormal state, and four rows indicate two vibration sensors on two axes, respectively (X - and Z -axes), as shown in Table 3.

The IoT gateway sends monitored signals to both an IoT master and a cloud-based condition-monitoring server. The

IoT master acts as a data repository and decision maker by collecting data relevant to the status of the entire system in real time from the main PLC and IoT gateway. As illustrated in Fig. 5a, we set the communication environment between the main IoT gateway and PLC with a regular cyclic data send/receive protocol and a fixed IP address. Sensor signal acquisition and message-sending for feedback control were set as shown in Fig. 5b.

Therefore, operators can remotely determine the status of the entire ASRS using the supervisory control and data acquisition (SCADA) interface in real time. Whereas HMI provides information acquired exclusively from the main PLC in the local network, the cloud-based condition-monitoring server displays sensor signals collected by the IoT-based PLC in a remote network. As depicted in Fig. 6, it is possible to perform real-time monitoring and collect relevant signals automatically in the SCADA interface in local and cloud environments. In particular, the current temperature and two vibration measurements on a sensor in the ASRS are recorded in the SCADA interface.

To establish communication between heterogeneous machines in the distributed ASRS control system, two appropriate communication protocols—Ethernet and Modbus—were considered in this study, as depicted in Fig. 5 [27, 28]. Ethernet is a common communication protocol, which helps to establish connection between existing internal networks as well as with the Internet (worldwide web) [29, 30]. In this study, Ethernet was used to facilitate intercomponent communication. Examples include the communication between the main PLC and HMI as well as that between the IoT gateway and IoT-based PLC. In addition, Ethernet communication allows for establishing connection between the local controller or data acquisition system and condition-monitoring applications (for example, between the IoT master and SCADA as well as between the IoT gateway and cloud server). On the other hand, Modbus is a serial data transmission protocol,

Table 2 Specifications of the measured sensor signals

Measured signal		Z-axis RMS acceleration	X-axis RMS acceleration	Temperature
Unit		g	g	°C
I/O range	Minimum	0	0	−327.68
	Maximum	65.535	65.535	327.67
Modbus register alias address		45,211	45,212	45204
Modbus register address		42,406	42,456	40043

Fig. 4 Newly designed control and communication architecture for ASRS to perform predictive process adjustment; Ethernet (yellow arrows) and Modbus (black arrows) connections are used to establish and facilitate communication between components

such as the RS-485 and RS-232 [31–33]. Owing to its master–slave (i.e., question–answer) architecture, it can receive signals easily from connected components and provide a control message in response. Because Modbus allows heterogeneous controllers to communicate easily with each other over Ethernet or serial cable and enables inexpensive construction [32, 33], it was employed in this study to send control messages and receive system status information from the main PLC and IoT gateway, respectively. Finally, as depicted in Fig. 7, the existing ASRS was reconfigured to perform predictive process adjustment based on the proposed control and communication architecture.

4.2 Real-time condition monitoring and feedback control for reliable ASRS operation

To demonstrate the utility of the proposed predictive process adjustment, we developed a real-time condition-monitoring and automatic process control system for the ASRS. A simple model, shown in Fig. 8, was generated for online condition monitoring and automatic adjustment. Similar to analyzing a simple statistical control chart and adopting expert knowledge, the following fault prediction rules were set forth: if

either the current vibration sensor signal measured from the *X*-axis thread is larger than or equal to 1G or the current vibration signal of the *X*-axis motor itself is larger than or equal to 0.8G, the loads placed on the feeder of the linear transporter are excessive (considered as a faulty state). If the current status is not adjusted, a fault will occur soon. Therefore, to preempt the occurrence of a fault, such as sudden termination of the ASRS operation, the main PLC controls relevant process functions (e.g., reduction of motor speed).

Machine-learning-based fault detection by sensor signal analyses is usually helpful when any mathematical and physical operation principles are not clear or when faulty states rarely occur [34]. Among the various machine-learning-based fault detection models summarized in Table 4, in this study, two statistical distance-based fault detection models (i.e., individual charts with limit checking of absolute values) were employed to detect two fault types individually, based on the following equations:

$$UCL = \bar{\bar{X}} + \frac{3}{d_2\sqrt{n}} \times (\bar{R}), LCL = \bar{\bar{X}} - \frac{3}{d_2\sqrt{n}} \times (\bar{R})$$

where $\bar{\bar{X}}$ is the mean value of the overall measurements in the normal states, $\bar{R}$ is the mean value of the differences between

Table 3 Sample of the data in the collected dataset

Time	Vibration sensor 1 on <i>X</i> -axis	Vibration sensor 1 on <i>Z</i> -axis	Vibration sensor 2 on <i>X</i> -axis	Vibration sensor 2 on <i>Z</i> -axis
00:00:00	0.014	0.010	0.010	0.007
00:00:01	0.014	0.010	0.012	0.007
00:00:02	0.014	0.009	0.011	0.008

Fig. 5 Communication settings between IoT gateway and PLC:
(a) Ethernet channel setting and
(b) read and write settings of regular cyclic communication in PLC

No	Protocol	Com.	C/S	Port	Address	Ethernet...
000	MODBUS	TCP	Client	502	192.168.0.201	
001	MODBUS	TCP	Client	502	192.168.0.211	
002	MODBUS	TCP	Client	502	192.168.0.221	
003	User Defined	TCP	Client	7171	192.168.0.71	
004	User Defined	TCP	Client	7171	192.168.0.72	
005	User Defined	TCP	Client	102	192.168.0.61	
006	User Defined	TCP	Client	102	192.168.0.181	
007	User Defined	TCP	Client	102	192.168.0.191	
008	MELSEC	TCP	Client	7000	192.168.0.91	
009						
010						
011						
012						
013						

(a)

	Enable	Operation	Device type	Address	Read/write length	Data type	ETOS address	ETOS bit location	ETOS status address
1	☒	Block Read	D	4220	10	Word	%MB5800	0	%MB5710
2	☒	Block Write	D	4200	10	Word	%MB5850	0	%MB5720
3	☐	Block Read	D	0	1	Word	%MB0	0	%MB0

(b)

Fig. 6 SCADA interface for the integrated ASRS

Fig. 7 Advanced ASRS for storage and retrieval of ordered products for users or subsequent manufacturing process

the maximum and minimum measurements in a sample, n is the sample size, and d_2 is the constant value determined by the sample size.

Fig. 8 Sample model for automatic predictive process control in ASRS

Using two control limits, a fault state is detected when the following equation is not satisfied [35]:

$$\alpha \times LCL \leq x_t \leq \alpha \times UCL$$

where x_t is the current measurement and α is the constant that corresponds to the confidence level. This distance-based fault detection is usually beneficial when the measurements collected during the no-fault state are clear, robust, and their density is high. In this condition, it is quite straightforward to consider statistical outliers as fault indicators and determine the correct adjustment commands to return to a normal state, with a relatively simple detection model.

Therefore, we first collected sensor signals during no-fault states (for each fault type) and constructed multivariate statistical models using the corresponding dataset. The developed models (i.e., fault detection rules 1 and 2) represent the healthy status for the corresponding fault type. With suitable statistical confidence levels, we generated appropriate threshold values for each sensor signal to detect or predict faulty states based on each rule.

This implies that while control and measurement signals are measured periodically, as depicted in Fig. 9, a condition-monitoring application always compares current vibration signals with predefined prediction rules. Figure 10a presents a ladder diagram that is used to perform real-time condition

Table 4 Machine-learning-based fault detection approaches using sensor signal analysis [34]

Fault detection approach	State information (labeled or not)	Gradient relationship	Popular decision criteria	Popular model
Anomaly detection	Unsupervised	Usually yes	Distance or density	Clustering
Distance-based fault detection	Supervised	Yes	Distance from the normal states	Statistical process control chart, statistical projection
Fault pattern identification	Supervised	No	Similarity to the known fault patterns	Classification

monitoring and control parameter adjustment. Prediction and adjustment rules are defined using dynamic variables (such as D4200, D4213, and M4000 observed in the current signal and control parameter windows in Fig. 9), which can be easily replaced to suit other fault detection, prediction, and resolution approaches.

Consequently, if a faulty state is detected while monitoring and analyzing online signals, the controller changes the relevant control parameters in accordance with the corresponding adjustment rule. For example, as depicted in Fig. 10b, if the current RMS values of the vibration signal pertaining to the *X*-axis motor exceed the upper control limit, the main PLC allows for a reduction in the speed of the linear transporter motor. Consequently, the

linear transporter operation becomes stable, and the RMS value of its vibration signal approaches zero.

To validate the effectiveness of the proposed process adjustment, we simulated fault states arbitrarily by controlling the external environments (e.g., excess load). As soon as the pre-developed model detected a fault state, adjusted control parameter values were sent, via memory variables in the IoT gateway, to the main PLC. Subsequently, the actuators moved slowly compared with the conventional movement to avoid sudden stopping of the system (e.g., arising from an action such as movement to an unavailable zone or clog at the certain point), and the system was returned to normal operation within a reasonably short time (e.g., less than a minute).

Fig. 9 Example of real-time monitoring window showing system control parameters and measurements

Fig. 10 Sample model for automatic predictive process control of ASRS: **(a)** ladder diagram for predictive process control and **(b)** real-time monitoring status

(a)

(b)

5 Conclusion

In this study, a system integration architecture was proposed for remote real-time condition monitoring and predictive process adjustment using collected sensor signals in an automated manufacturing system. An IoT-based PLC and cloud computing were introduced to collect vibration sensor signals automatically and reflect corrections made to the related control parameters in real time. In addition, cloud computing enabled application of several fault-detection methods, such as statistical and machine-learning-based approaches. The utility

of the proposed architecture was demonstrated via the development of an ASRS along with an online condition-monitoring and automatic and predictive process adjustment application for research and pedagogical purposes. The proposed predictive process adjustment technique can be employed by monitoring and analyzing vibration signals pertaining to an AC servomotor of the liner transporter in the developed ASRS. If faulty states are detected/predicted based on the predefined fault analysis rules, corresponding adjustments can be automatically performed to avoid sudden and unknown terminations in the manufacturing system. All fault

analysis results and adjustment rules are managed as dynamic control logic variables. Through cloud computing, these variables can be easily replaced to suit complex maintenance models.

The proposed architecture exhibits the potential for widespread practical application in several automated manufacturing systems equipped with industrial process controllers. The system developed in this study employed only two externally manufactured PLCs along with an additionally installed sensor. Therefore, the arbitrarily simulated fault states are relatively simple and, consequently, a simple statistical distance-based fault detection and process adjustment can always easily detect and fix the fault state. However, it is necessary to identify more system response and reliability measures when complex fault states occur, and to also compare the performance using different types of fault detection models, not only statistical distance-based fault detection models, but also fault pattern identification approaches such as dynamic deep learning-based models. In addition, we plan to extend the flexibility of the proposed system so that it can be used in combination with (i) various industrial controllers—such as Raspberry Pi (Raspberry Pi Foundation), CompactRIO (NI), and industrial computers—and (ii) different types of sensors.

Author contribution S. Baek conceived and designed the proposed system, performed the experiments, analyzed the results, and wrote the paper. The ASRS system is the result of the “Innovation Leading University Program for the Fourth Industrial Revolution,” supported by the Ministry of Education and National Research Foundation of Korea. The staff of FM Solution provided immense support for developing the proposed system.

Funding This work was supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIT) (No. NRF-2019R1G1A1097478).

Data availability Not applicable.

Compliance with ethical standards

Conflict of interest The funding sponsors had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript, or in the decision to publish the results.

Code availability Not applicable.

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain

permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

References

1. Basir R, Qaisar S, Ali M, Aldwairi M, Ashraf MI, Mahmood A, Gidlund M (2019) Fog computing enabling industrial internet of things: state-of-the-art and research challenges. *Sensors* 19:4807. <https://doi.org/10.3390/s19214807>
2. Huh J, Pham Van HP, Han S, Choi H-J, Choi S-K (2019) A data-driven approach for the diagnosis of mechanical systems using trained subtracted signal spectrograms. *Sensors* 19:1055. <https://doi.org/10.3390/s19051055>
3. Jung K, Choi S, Kulvatunyou B, Cho H, Morris KC (2017) A reference activity model for smart factory design and improvement. *Prod Plan Control* 28:108–122. <https://doi.org/10.1080/09537287.2016.1237686>
4. Chekired DA, Khoukhi L, Mouftah HT (2018) Industrial IoT data scheduling based on hierarchical fog computing: a key for enabling smart factory. *IEEE Trans Ind Inf* 14:4590–4602. <https://doi.org/10.1109/TII.2018.2843802>
5. Goryachev A, Kozhevnikov S, Kolbova E, Kuznetsov O, Simonova E, Skobelev P, Tsarev A, Shepilov Y (2012) “Smart factory”: intelligent system for workshop resource allocation, scheduling, optimization and controlling in real time. *Adv Mater Res* 630:508–513. <https://doi.org/10.4028/www.scientific.net/AMR.630.508>
6. Cardin O, Trentesaux D, Thomas A, Castagna P, Berger T, El-Haouzi HB (2017) Coupling predictive scheduling and reactive control in manufacturing hybrid control architectures: state of the art and future challenges. *J Intell Manuf* 28:1503–1517. <https://doi.org/10.1007/s10845-015-1139-0>
7. Wang C, Jiang P (2018) Manifold learning based rescheduling decision mechanism for recessive disturbances in RFID-driven job shops. *J Intell Manuf* 29:1485–1500. <https://doi.org/10.1007/s10845-016-1194-1>
8. Ortiz MA, Betancourt LE, Negrete KP, De Felice FD, Petrillo A (2018) Dispatching algorithm for production programming of flexible job-shop systems in the smart factory industry. *Ann Oper Res* 264:409–433. <https://doi.org/10.1007/s10479-017-2678-x>
9. Korodi A, Anitei D, Boitor A, Silea I (2020) Image-processing-based low-cost fault detection solution for end-of-line ECUs in automotive manufacturing. *Sensors* 20:3520. <https://doi.org/10.3390/s20123520>
10. Choi J, Lee SJ (2020) Consistency index-based sensor fault detection system for nuclear power plant emergency situations using an LSTM network. *Sensors* 20:1651. <https://doi.org/10.3390/s20061651>
11. Baek W, Kim DY (2020) An in-process inspection system to detect noise originating from within the interior trim panels of car doors. *Sensors* 20:630. <https://doi.org/10.3390/s20030630>
12. Zeng J, Chang B, Du D, Wang L, Chang S, Peng G, Wang W (2018) A weld position recognition method based on directional and structured light information fusion in multi-layer/multi-pass welding. *Sensors* 18:129. <https://doi.org/10.3390/s18010129>
13. Gajjar S, Kulahci M, Palazoglu A (2018) Real-time fault detection and diagnosis using sparse principal component analysis. *J Process Control* 67:112–128. <https://doi.org/10.1016/j.jprocont.2017.03.005>
14. Hsu C, Liu W (2020) Multiple time-series convolutional neural network for fault detection and diagnosis and empirical study in

- semiconductor manufacturing. *J Intell Manuf.* <https://doi.org/10.1007/s10845-020-01591-0>
15. Said M, Abdellafou KB, Taouali O (2020) Machine learning technique for data-driven fault detection of nonlinear processes. *J Intell Manuf* 31:865–884
16. Lu S, Zheng P, Liu Y, Cao Z, Yang H, Wang Q (2019) Sound-aided vibration weak signal enhancement for bearing fault detection by using adaptive stochastic resonance. *J Sound Vib* 449:18–29. <https://doi.org/10.1016/j.jsv.2019.02.028>
17. Lee H, Kim Y, Kim CO (2017) A deep learning model for robust wafer fault monitoring with sensor measurement noise. *IEEE Trans Semicond Manufact* 30:23–31. <https://doi.org/10.1109/TSM.2016.2628865>
18. Feng Z, Zuo MJ (2012) Vibration signal models for fault diagnosis of planetary gearboxes. *J Sound Vib* 331:4919–4939. <https://doi.org/10.1016/j.jsv.2012.05.039>
19. Wen-ju C, Hui L (2008) Aircraft electric braking system based on parameter fuzzy automatic adjustive PID. *Electron Opt Control* 15: 39–42
20. He R, Chen G, Dong C, Sun S, Shen X (2019) Data-driven digital twin technology for optimized control in process systems. *ISA Trans* 95:221–234. <https://doi.org/10.1016/j.isatra.2019.05.011>
21. Ely GR, Speersad CC (2009) A comparative study of metamodeling techniques for predictive process control of welding applications. In: *Proceedings of the ASME 2009 international manufacturing science and engineering conference*, vol 43611, pp 395–407
22. Kim D-Y, Park J-W, Baek S, Park K-B, Kim H-R, Park J-I, Kim H-S, Kim B-B, Oh H-Y, Namgung K, Baek W (2020) A modular factory testbed for the rapid reconfiguration of manufacturing systems. *J Intell Manuf* 31:661–680. <https://doi.org/10.1007/s10845-019-01471-2>
23. Baek S, Oh H, Baek W, Kim D-Y (2016) Trend of predictive maintenance technology for facilities and products using sensor information. In: *Proceedings of the Society of 2016 computational design and engineering conference*, Republic of Korea, pp 25–26
24. Ishak S, Ahmad A (2017) Improving VIGV predictive monitoring by developing a failure mode virtual sensor. In: *Proceedings of. Comput Ind Eng, Malaysia, 2017 7th IEEE International Conference on Control System*, pp 144–149
25. González-Prieto I, Duran MJ, Rios-Garcia N, Barrero F, Martín C (2018) Open-switch fault detection in five-phase induction motor drives using model predictive control. *IEEE Trans Ind Electron* 65: 3045–3055. <https://doi.org/10.1109/TIE.2017.2748052>
26. Mani G, Sivaraman N (2017) Integrating Fuzzy based fault diagnosis with constrained model predictive control for industrial applications. *J Electr Eng Technol* 12:886–889. <https://doi.org/10.5370/JEET.2017.12.2.886>
27. Czwick C, Kugler S, Greven J, Anderl R (2019) Communication protocol application for enhanced connectivity of sensors, machines and systems in additive manufacturing and production networks. In: *IFIP Advances in Information and Communication Technology Proceedings of the IFIP 16th international conference on Product LifeCycle Management*, pp 136–145. https://doi.org/10.1007/978-3-030-42250-9_13
28. Hinchy EP, O'Dowd NP, McCarthy CT (2019) Using open-source microcontrollers to enable digital twin communication for smart manufacturing. *Procedia Manuf* 38:1213–1219. <https://doi.org/10.1016/j.promfg.2020.01.212>
29. Wang ZP, Xiong GZ, Luo J, Lai MZ, Zhou W (2000) A hard real-time communication control protocol based on the Ethernet. *Real Time Syst* 1:6
30. Joelianto E, Hosana (2008) Performance of an industrial data communication protocol on ethernet network. In: *Proceedings of the 2008 5th IFIP international conference on Wireless and Optical Communications Networks, Indonesia*, pp 1–5
31. Hittanagi KN, Ramesh M, Kumar KR, Mahadeva SK (2017) PLC based DC drive control using Modbus RTU communication for selected applications of sugar mill. In: *Proceedings of the 2017 international conference on circuits, controls, and communications (CCUBE), India*, pp 80–85
32. Kulik V, Kirichek R (2018) The heterogeneous gateways in the industrial Internet of Things. In: *Proceedings of the 2018 10th international congress on ultra modern telecommunications and control systems and workshops, Russia*, pp 1–5
33. Xiao G, He Y, Huang Y, Liu Y, Dai W, Quan L (2019) Integration of heterogeneous system information in machining-detection logistics for titanium alloy blade with adaptive belt grinding. *Int J Comput Integr Manuf* 32:900–917. <https://doi.org/10.1080/0951192X.2019.1644479>
34. Baek S, Kim D-Y (2019) Abrupt variance and discernibility analyses of multi-sensor signals for fault pattern extraction. *Comput Ind Eng* 128:999–1007. <https://doi.org/10.1016/j.cie.2018.06.019>
35. Isermann R (2006) *Fault-diagnosis systems*. Springer, Berlin, Heidelberg

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.