Data Acquisition and Monitoring System for Legacy Injection Machines

Bruno Silva*‡, João Sousa*†, Guillem Alenya§
*School of Techology and Management, Polytechnic of Leiria, Portugal
Email: {bruno.s.silva, jcsousa}@ipleiria.pt
†INESC Coimbra, DEEC, Polo II, University of Coimbra, Portugal
‡Vipex, Leiria, Portugal
§Institut de Robotica i Informatica Industrial CSIC-UPC, Barcelona, Spain
Email: galenya@iri.upc.edu

***Abstract*—Nowadays, companies must embrace the concept of Digitalization and Industry 4.0 to remain competitive in the market. The reality is that most of them do not have their industrial devices prepared to access their data on a real-time basis. As most companies do not have the possibility to renew all their legacy devices and because these devices are still very productive, a retrofit solution is of high interest. In this work, we propose an affordable procedure that allows data collection and monitoring of older injection machines, as a contribution towards legacy devices integration. The developed system neither requires additional proprietary modules, nor contractual annual fees for different devices, sharing the same interface across different machine manufacturers and also contributing to uniform data collection. Evaluation was carried out in a real shop floor, monitoring the injection parameters for different machine models, validating the effectiveness of the developed system.**

***Index Terms*—Injection Machines, Digitalization, Industry 4.0, Monitoring Systems.**

I. Introduction

The world is constantly evolving and today with the spread of the concept of Digitization and Industry 4.0, companies have to keep up with the evolution to remain competitive in the market. Regarding this, it is necessary to find the best trade-off between innovation and the possible investment turnover, as not all companies are equipped with recent equipment with standardized communication protocols for data collection and transmission like the OPC-UA, and are unable to replace the entire shop floors with new equipment. Therefore, it is necessary to adapt the existing equipment on the factory floor [1] [2] so that it is possible to have the process data in real time to be further processed to draw conclusions that can improve performance [3], production [4], maintenance schedules, among others.

Another difficulty that companies encounter in this evolution is not only the investment in the equipment itself but also in the licenses of the software contracted to third parties for the collection and data manipulation. These license fees are commonly paid on an annual basis and often expensive to smaller companies.

This work was partially supported by the Spanish State Research Agency through the María de Maeztu Seal of Excellence to IRI MDM-2016-0656 and by FCT—Portuguese Foundation for Science and Technology under project grant UIDB/00308/2020.

The injection processes are quite variable and for that, the analysis of the data in real time can be an asset in the identification or prediction of possible failures in the produced parts [5] [6], which leads to a reduction of the rejected parts [7], or in the interpretation of the events that occurred, which helps the maintenance teams [8].

Concerning injection machines, the older ones usually do not support protocols like OPC UA with Euromap77 information model to describe them, as it happens with the recently released machines [9]. Some of these older injection machines, support other communication mechanisms, frequently with a required extra fee.

As most injection machines that do not have real time data available have a USB output to generate a file for printing, a low cost system (around tens of euros instead of hundreds/thousands, which are typically the prices of the interfaces provided by the manufacturer) was created. This system allows to collect, process and display data in almost real time. This system and its development are presented in this article. The real time here means access to the injection process data between each cycle of parts injection, it is not entirely in real time but it is only on the lapse of a cycle. In this case, it ends up not making a big difference because many of the repercussions in this type of processes are not reflected in a single cycle but in a combination of several.

The developed system solves the problem of data collection in equipment that do not have this characteristic of factory pre-configuration (without adding extras). So with a low-cost system we can collect, process and view data in real-time [10] using open-source software and using the recording feature via the USB port provided by the brands without additional cost. Therefore, it is not necessary to purchase extra modules or contractual licenses for the different equipment annually. Another focus on the development of this system was the possibility of using the same interface on several different machine brands, which allows uniformity in data collection.

The system functionality was evaluated in the shop floor for several months on machines from three different brands (Engel, Negri Bossi and Tederic) and the experimental results prove that the system behaves correctly.

Thus, the system developed and presented in this article contributes to:

- Low-Cost: Make possible to the companies to have a simple way of adapting their equipment to the current concept of Industry 4.0 without having to make large additional investments;
- Flexibility: Wireless based solution, which allows the system to be flexible. It is not necessary to make major changes to the physical network structures of the shop floor systems;
- Versatility: The concept of this system can be used in any equipment that exports data via USB, but it is not just for injection machines. As the software is developed using a general purpose programming language (Python in this case), it allows using different known libraries for data processing/machine learning such as *Numpy*, *Pandas*, among others.

The paper is organized as follows. In Section II are presented some of the works done in the area and how they can relate to our work. Section III resumes the functioning of the presented system. Section IV is concerned with the development details about the software and hardware developed to collect the data from the injection machines. Section V presents the evaluation of the system in two relevant use-cases: the gathering of the data and a glimpse of the potential benefits, and the normalization across different machines. To conclude, the last section presents some conclusions and future work to be developed.

II. Related Work

In the literature, there are some related works dealing with real-time injection process monitoring.

In [11] Karbasi *et al.* present a monitoring system with pressure and temperature sensors added to the mold cavity. A LabView setup was used to do the data acquisition and sensor calibration on the software side, using a PC to read the sensors and send the data into a spreadsheet file. On the hardware side, they used an interface card (PC and data acquisition), a data acquisition card to collect analog signals from sensor amplifiers and a power supply to feed the temperature and pressure amplifiers. This means the need to use a lot of hardware, which implies a high cost and complexity. In comparison, we aim to a simpler and easy to setup system. Additionally, in this article the data collection of the injection process is focused on the mold side, which presents a different approach to our work.

Zhao et al. present a different approach to injection process monitoring [12]. In this study, electrical sensors such as thermocouples and displacement and pressure probes are installed in the injection molding machine, not in the mold, for collecting the pressure, position, temperature and time during plastic molding. Three different data collection cards are used to pretreat and transform the analog and discrete signals to digital signals, respectively. The digital signals are read and saved by a supervising computer running Windows OS that also serves as a data processor. In this article, the data from the injection process is collected on the machine side, similarly to our study, but the machine is instrumented which is a disadvantage compared to our presented system that uses the machine sensor reading.

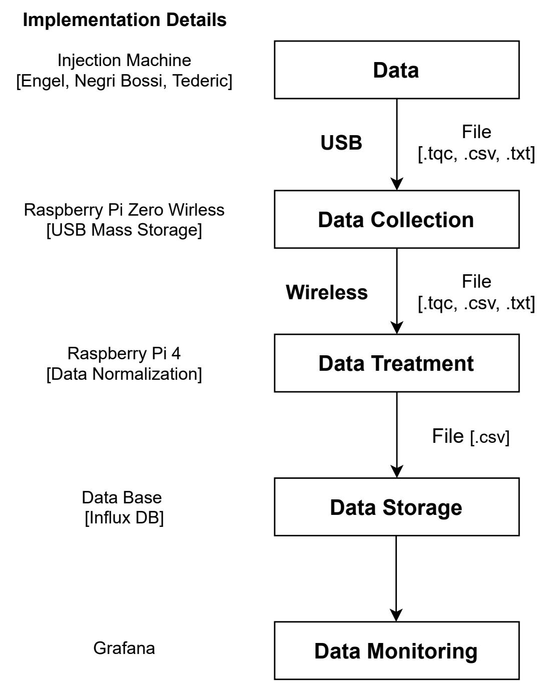

Fig. 1. System Architecture

In [13] Hong *et al.* an interface is presented which is not only used for collection but also for control. The data collection implementation is identical to that presented by Karbasi *et al.* [11]. They also use signal amplifiers, data acquisition system, data interface and a PC running Python scripts to apply machine learning algorithms. This article presents an AI-based control system to control the hot runner and barrel temperature and holding pressure. In this article we can observe a possible future work to be done with the data collected from our system (in this case the AI-based control using the injection machine parameters).

There are many studies done in this area based on simulations of the injection processes [14] [15] [16]. It is also possible to find studies done in real environments, like ours, but most of them take into account only one injection machine [17] [18] [10]. One of the advantages of our system in relation to the works presented is that it allows to monitor more than one machine as will be presented later in this article.

As for the implementation costs of the systems of the studies presented in this chapter, they are superior in relation to our system due to the need for instrumentation, whether of the machine or the mold, the use of external monitoring systems, signal converters, charge amplifiers, among others.

III. Developed System

This section presents the overall design of the system and the description of the different stages.

As we can see in Fig.1, the system consists of five stages. The availability of data from the injection process (data), the collection of data on each individual machine (data collection), the data treatment from different machine brands (data treatment), the storage of data in a physical database (data storage) and real-time data monitoring and visualization (data monitoring). It is also possible to observe how data is transferred between systems and the type of files that are exchanged between each phase.

The details of each step related with the system concept are described below (implementation details are presented in the next section):

- **Data** - The data made available by different brands of machines comes in different formats (.txt, .tqc, .csv). These formats are imposed by the manufacturers of each machine and cannot be changed. Regarding the number of available parameters, Negri Bossi allows to collect only 6 parameters at a time from all available parameters. Tederic and Engel allow to export all the parameters they monitor.
- **Data Collection** - The injection parameter data for each cycle is stored in the collection system connected to each machine and sent, cycle by cycle, to the data processing unit.
- **Data Treatment** - In this stage, the data of the different machines are normalized so that the order of writing in the database is always the same (normalized data set). This is because, as mentioned, there are machine brands that provide more parameters than others and thus, identical parameters will be saved in the same order in the database. The storage order of these variables is fixed and chosen based on the parameters that the shop floor injection engineers evaluate when there is some variation in the process. This facilitates access and processing of the data.
- **Data Storage** - The storage of previously normalized data allows having an organized structure to interact with monitoring systems, automatic data processing systems (data science and machine learning algorithms), among others. With this, this system allows not only to visualize the data in almost real time, but also the historical records that potentially allow to correlate defects in parts and problems that may have occurred and can be detected within the injection datalog.
- **Data Monitoring** - In terms of the system's tangible gains, this is one of the most valuable for the company's performance. This is because, having a graphical access to the collected data, allows not only process engineers to analyze the different behaviors of the machines to correct the processes, but also for the common operators. Having a screen next to each machine with the respective parameters, the operators can see sudden differences in the parameters and alert those responsible for the process or maintenance about any problems that may have occurred. This minimizes the response of different teams to reduce downtime, which consequently reduces the company's monetary losses.

If we want to add this system to a machine that does not have a file extension like the ones mentioned (.tqc, .csv or .txt), it is just necessary to adapt the data processing script. This flexibility is one of the advantages of this system.

Another advantage of the developed system is the possibility of adapting and scaling the system because it is fully developed by us (developers on the factory floor). This leads to greater autonomy and less dependence on the availability of providers to implement the different needs that may arise.

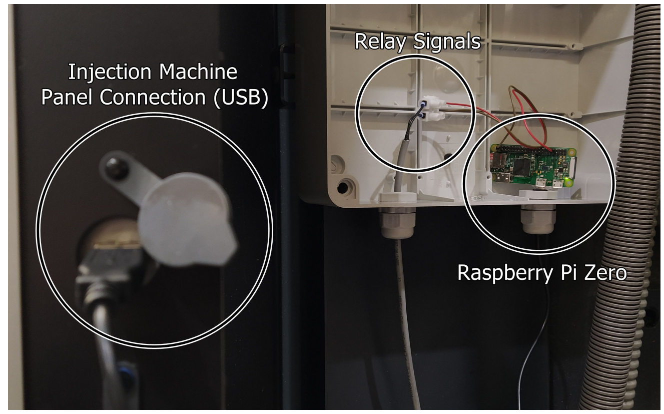

Fig. 2. Real system connected to the injection machine.

IV. Implementation Details

The great challenge of this project was to find a way to have a system where a machine could write data via USB between each injection cycle but which could also make the data available on another platform so that it could be saved and observed at the same time. In that way, a pen drive cannot be used because it does not allow reading and writing at the same time. In essence, we need a device that can work similarly to USB-OTG technology.

In the next two subsections, the developments made regarding the hardware and software are presented.

A. Hardware

Figure 2 shows the developed system connected to the injection machine, in this case, Negri Bossi 330 Tones Bi Injection Machine (NB330BI). Regarding the hardware, one restriction was to only use equipment already available on the market so it was not necessary to develop any specific component from scratch, which makes this system easy to reproduce.

To solve this problem, we tested the use of a system that could read and write at the same time. For this, a Raspberry Pi zero (RPi0) with wireless functionality was used to emulate as a mass storage device. As this Raspberry Pi has wireless functionality, it was possible to send this data to another device to be processed and stored. This means that is necessary to place a RPi0 per injection machine.

In order not to overload the computational power of the collection devices, a Raspberry Pi 4 (RPi4) with 8 GB ram was used to process and store the data.

At the moment, the system is running on five machines (5 RPi0's) and only one RPi4 is used to process the data. This makes the system cheaper. The RPi4 may still be able to process data from more machines, but it is a matter of balancing the required computational power and the cost. We note that the data flow depends on the injection cycle, the shorter the injection cycle, the greater the data flow per machine.

Regarding the collection cycle by cycle, it was necessary to place two relays on each machine to replicate the signal referring to the automatic operation mode and the signal that represents a new injected part. When the two signals are at high logic level it signals that we have a new cycle and this give the order to the RPi0 to send the data to the RPi4.

In machines that do not have the possibility to append the date and time of each injection cycle, a real time clock (RTC) was placed in the system to guarantee the correct time and date so that later it can be compared with data process and the time at which specific events occurred.

B. Software

In this section, the scripts that were developed for the different constituents of the system (data collection and treatment) are presented.

1) Raspberry Pi Zero Wireless: In this case, the developed scripts are simpler. These are bash scripts that allow the read of a digital signal (new machine cycle signal for data collection order) and copy the file with the process data generated by the machine at each new cycle.

The file copy is made through a folder shared between the RPi0 and the RPi4. As these devices have the wireless functionality, a wireless communication network was created at the factory that allows interconnecting all devices. Thus, through the assigned IPs, there is a shared folder between each of the collection devices and the data processing board (RPi4).

2) Raspberry Pi 4: On the factory floor where the system test was carried out, there are three brands of machines that have the functionality to export a file with the process data via USB. The brands of the machines are Engel, Negri Bossi and Tederic.

Each of these brands exports the file in a different extension. Negri Bossi exports the process data in a file of the type (.tqc), Engel in a file of the type (.csv) and Tederic in a file of the type (.txt).

As the cycle-by-cycle writing of the file is imposed by the machine, it was necessary to implement the data processing system for each of the different brands (this implies a different script for each brand).

In this data treatment, the Python language was used to develop the scripts that facilitate the implementation of some of the necessary functionalities. With the Python language it is possible to process the data with functionality features like pandas and numpy to prepare the data effectively not only for placement in databases, but for future use of machine learning algorithms [19].

Although the treatment of the different files is done differently, the reasoning and the final implementation is always the same. Basically, the written file in the shared folder by the data collection system is opened, the last written cycle is read and saved in an open source time series database (Influx DB[1]).

This part of the system works as a data normalization API, as each of the brands provides a number of different process variables per file. Engel and Tederic have no limit on the number of the variables that can be acquired, but Negri Bossi using the .tqc file has a limit of six variables per file. Therefore, in this case we have to choose which variables we want to see within that limit.

This data can also be sent to a cloud storage, for example, AWS[2] from Amazon or Azure[3] from Microsoft, in order to facilitate the work of processing data through machine learning algorithms. The use of computing and storage resources as a service from the service provider, will further avoid the need of powerful computing devices in the presented system side.

So far we have referred to the form of data collection, treatment and storage. However, without having the possibility of observing the data cycle by cycle, it is not possible to correlate problems in production with the process variables of the machines. An open source version of Grafana software was used for this propose, allowing a real-time data visualization.

V. Experimental evaluation

This chapter presents two experiments: first the example of the monitorization of several parameters from a single machine, and second, an example of comparison of data gathered from different machines. Both experiences include a practical example of a simple problem that occurred in an injection process and that was solved thanks to the analysis of the data collected in the injection machine where the system was connected.

First experiment results can be seen in Fig.3, where it is possible to observe the appearance of the created monitoring interface for the NB330BI machine.

In this machine, six parameters were being monitored, namely: injection time, plasticization time, cycle time, cushion, closing force and maximum injection pressure. Due to the size of the image and in order to be perceptible, only the parameters that have undergone relevant changes appear here and allowed us to understand the cause of the problem. In this particular case, the parameters in which the highest deviations were observed are the injection time, the plasticization time and the cushion. The injection pressure is also displayed to maintain the correct dimensions of the image (although variations are observed, their impact is residual).

Regarding the problem, quality operators identified defects in some produced parts and alerted the injection technicians.

[1]https://www.influxdata.com
[2]https://aws.amazon.com
[3]https://azure.microsoft.com

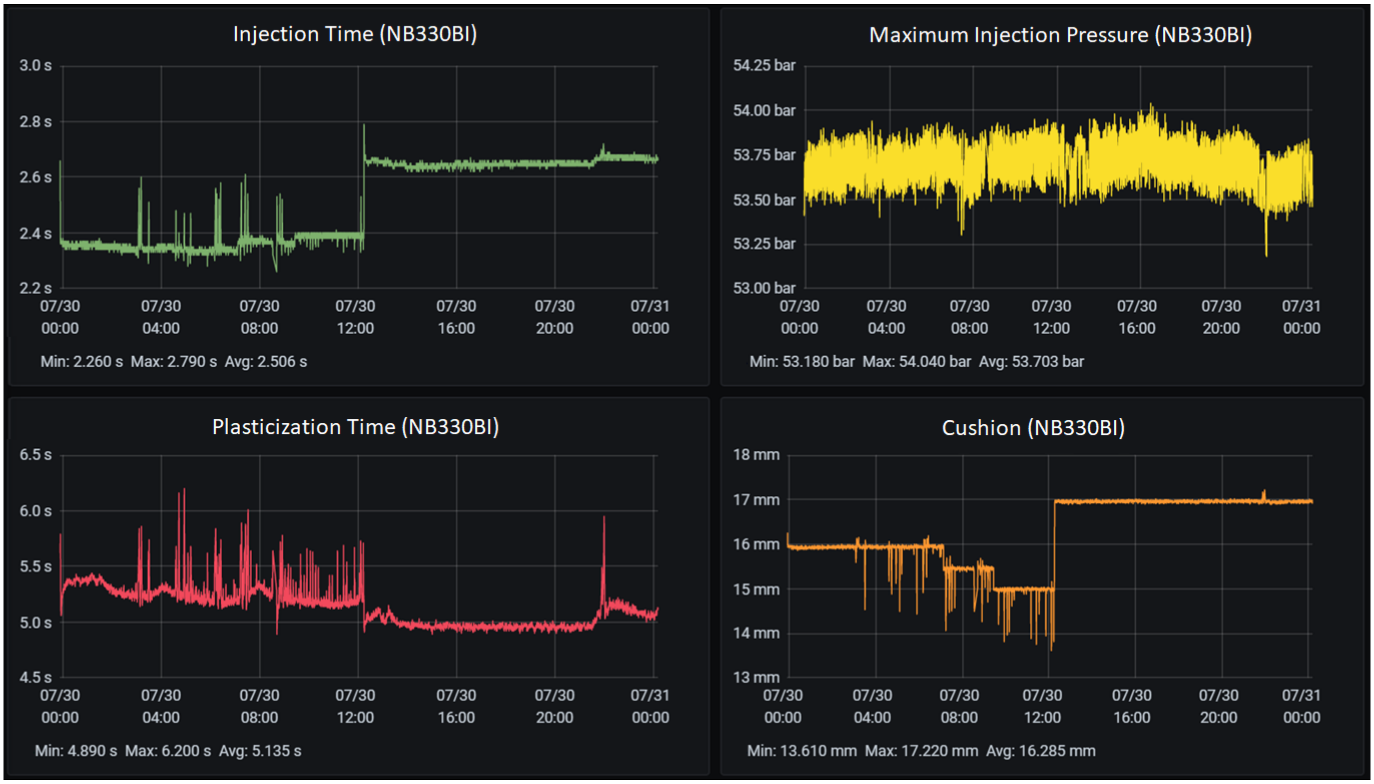

Fig. 3. Example of a data visualization interface.

It was concluded that the injection process was unstable, as it is possible to observe at the beginning of the different graphs shown in Fig.3.

The technicians tried to make adjustments without yet understanding the cause of the problem, but began to observe that over time the value of the cushion decreased, which caused the material in the machine's spindle to decrease. It is possible to observe this decrease in the value of the cushion over time in the graph related to this variable. Although this is not an article related to the injection technique and in order to assist in the analysis, the cushion represents the amount of material that is present in the spindle after a part is injected. This amount is measured by the value of the distance between the nozzle and the site where the inner part of the spindle stops, which retains the material after injecting. The shorter the distance, the less material remains after each injection.

Based on the information mentioned above, the injection technicians realized that over time there was less material in the spindle after each injected part and they went to see where leakage problems could be. They observed that the spindle ring (connection between the mold and the spindle) was not sealing correctly.

After realizing the problem, the injection technicians intervened in the process in order to solve it. This intervention can be seen in Fig.3 from 12 o'clock. Here the parameters were stable and the parts produced with defects did not appear again.

Regarding the second experiment, we focus on showing the potential of visualizing the process parameters of the different machines simultaneously. As before, errors were caused in the injection processes (spindle temperature variation, disconnecting water circuits, among others) in two different machines. These machines are working with the same material (in this case, low density polyethylene) to understand if the variation of the different parameters were identical. The tests with error provocation were the same and performed in the same order.

In Fig.4 it is possible to observe the variation of the plasticization time. This variable was one of the chosen among the others because it is one of the variables with the greatest variation and because it is different from the one analyzed in the previous example.

The tests were carried out on different days and in parts with different cycle times (hence the variations on the x-axis of the graph). Despite this, it is possible to observe that the behavior of this parameter was identical in both cases. Even if the behavior shown is identical, further testing is needed in the future to see if there is a pattern of variation and correlation.

These examples are a proof that a lot of potential work can be done in this field because there is still a lot to explore. The causes of problems are not always the same, the variables that change can vary from problem to problem and from material to material, among other things.

In this case, the data was used to help in identifying a problem, but in the future, machine learning techniques can be used to predict failures before they occur and help with predictive maintenance and thus reduce the time that machines

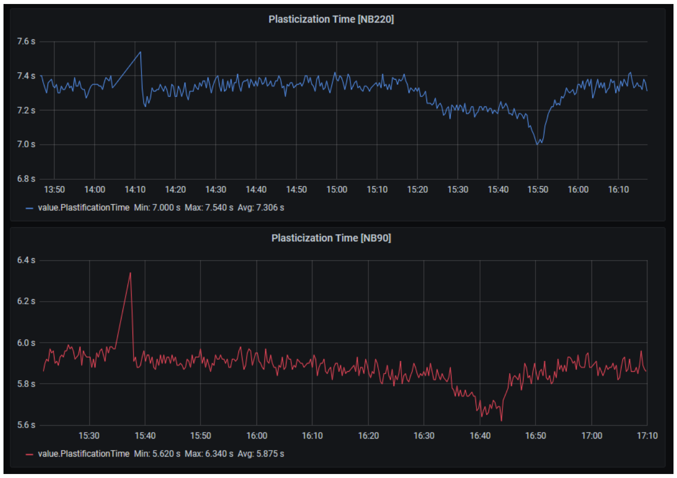

Fig. 4. Plasticization time of NB90 and NB220 machines.

are stopped.

VI. Conclusion and Future Work

With the system developed and presented in this article, we can solve a problem that many industries face today, which is obtaining data in real time from machines who do not have the latest communication protocols (legacy devices) and at a cost that can be supported. This system is a low-cost, flexible and versatile alternative for implementing digitization in factories with injection machines.

The described system is actually working on different machines on a factory floor. To show its interest, we have described two experiments involving a single machine and the comparison of two machines. Some problems that caused defective parts were solved with the help of the data collected and presented by the developed system. Some of these problems are presented in this article in order to explain the potential that it can add to its use in the plastic industry.

With the success implementation of this system, a new stage now appears, which is the treatment and drawing of conclusions about the data obtained. One of the great future goals is to understand the injection process (correlation between the variables) and to understand how these variations of the machine parameters can be related to the condition of the parts produced.

This work opens the possibility of evaluating work batches a-posteriori, but also to on-line monitor the evolution of the process to, for example, raise alarms to the operators or even adjust automatically some parts of the process. This is a challenge with many aspects to be worked on because the injection process is a complex process, the correlations of the variables can vary depending on the type of material, the size of the part, the capacity of the machines, among others. It will also be interesting to develop algorithms that allow, through the analysis of injection parameters, to identify problems before they occur, perform predictive maintenance and solve other challenges that may arise.

References

[1] J. Rosas, V. Brito, L. B. Palma, and J. Barata, "Approach to adapt a legacy manufacturing system into the IoT paradigm," *International Journal of Interactive Mobile Technologies*, 2017.

[2] A. Martins, H. Costelha, and C. Neves, "Shop Floor Virtualization and Industry 4.0," *2019 IEEE International Conference on Autonomous Robot Systems and Competitions (ICARSC)*, 2019.

[3] M. Charest, R. Finn, and R. Dubay, "Integration of artificial intelligence in an injection molding process for on-line process parameter adjustment," in *12th Annual IEEE International Systems Conference, SysCon 2018 - Proceedings*, 2018.

[4] C. Hopmann, A. Ressmann, M. Reiter, S. Stemmler, and D. Abel, "A self-optimising injection moulding process with model-based control system parameterisation," *International Journal of Computer Integrated Manufacturing*, 2016.

[5] P. Zhao, H. Zhou, Y. Li, and D. Li, "Process parameters optimization of injection moldingusing a fast strip analysis as a surrogate model," *International Journal of Advanced Manufacturing Technology*, 2010.

[6] R. Bernardete, "Support vector machines for quality monitoring in aplastic injection molding process," *IEEE Transactions on Systems, Man,and Cybernetics, Part C (Applications and Reviews)*, 2005.

[7] O. Ogorodnyk, O. V. Lyngstad, M. Larsen, and K. Martinsen, "Application of feature selection methods for defining critical parameters in thermoplastics injection molding," in *Procedia CIRP*, 2019.

[8] C. Park, D. Moon, N. Do, and S. M. Bae, "A predictive maintenance approach based on real-time internal parameter monitoring," *International Journal of Advanced Manufacturing Technology*, 2016.

[9] P. Pierleoni, L. Palma, A. Belli, and L. Sabbatini, "Using Plastic Injection Moulding Machine Process Parameters for Predictive Maintenance Purposes," in *Proceedings of International Conference on Intelligent Engineering and Management, ICIEM 2020*, 2020.

[10] S. Kumar, H. S. Park, and C. M. Lee, "Data-driven smart control of injection molding process," *CIRP Journal of Manufacturing Science and Technology*, 2020.

[11] K. Hamidreza and H. Reiser, "Smart mold: Real-time in-cavity data acquisition," 2006.

[12] P. Zhao, H. Zhou, Y. He, K. Cai, and J. Fu, "A nondestructive online method for monitoring the injection molding process by collecting and analyzing machine running data," *International Journal of Advanced Manufacturing Technology*, 2014.

[13] H. S. Park, D. X. Phuong, and S. Kumar, "AI based injection molding process for consistent product quality," in *Procedia Manufacturing*, 2019.

[14] P. Nagorny, M. Pillet, E. Pairel, R. Goff, J. Loureaux, M. Wali, and P. Kiener, "Quality Prediction in Injection Molding," *2017 IEEE International Conference on Computational Intelligence and Virtual Environments for Measurement Systems and Applications (CIVEMSA)*, 2017.

[15] M. Meiabadia, A. Vafaeesefatb, and F. Sharific, "Optimization of Plastic Injection Molding Process by Combination of Artificial Neural Network and Genetic Algorithm," *Journal of Optimization in Industrial Engineering 13 (2013) 49-54*, 2013.

[16] A. Goyal, S. Sharma, and V. Sharma, "Quality Improvement of Plastic Injection Molded Product Using Doe and Taguchi Techniques," *International Journal of Recent advances in Mechanical Engineering 4(2):93-103*, 2015.

[17] K.-C. Ke and M.-S. Huang, "Quality prediction for injection molding by using a multilayer perceptron neural network," *Polymers*, vol. 12, no. 8, 2020.

[18] Y. Zhang, J.-Y. Chen, C.-C. Tseng, and M.-S. Huang, "Quality Indexes Design for Online Monitoring Polymer Injection Molding," *Advances in Polymer Technology*, 2019.

[19] M. Saleh Meiabadi, A. Vafaeesefat, and F. Sharifi, "Optimization of Plastic Injection Molding Process by Combination of Artificial Neural Network and Genetic Algorithm," *Journal of Optimization in Industrial Engineering*, 2013.