

Automatic Generation of Improvement Suggestions for Legacy, PLC Controlled Manufacturing Equipment Utilizing Machine Learning

Koehler, W. & Jing, Y.

Published PDF deposited in Coventry University's Repository

Original citation:

Koehler, W & Jing, Y 2020, Automatic Generation of Improvement Suggestions for Legacy, PLC Controlled Manufacturing Equipment Utilizing Machine Learning. in J Beyerer, A Maier & O Niggemann (eds), Automatic Generation of Improvement Suggestions for Legacy, PLC Controlled Manufacturing Equipment Utilizing Machine Learning. Springer Verlag, pp. 93-102, 5th International conference Machine Learning for Cyber Physical Systems , Berlin, Germany, 12/03/20.

https://dx.doi.org/10.1007/978-3-662-62746-4_10

DOI 10.1007/978-3-662-62746-4_10

ISBN 978-3-662-62745-7

ISBN 978-3-662-62746-4

Publisher: Springer

This chapter is licensed under the terms of the Creative Commons Attribution 4.0 International License (<http://creativecommons.org/licenses/by/4.0/>), 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 license and indicate if changes were made.

The images or other third party material in this chapter are included in the chapter's Creative Commons license, unless indicated otherwise in a credit line to the material. If material is not included in the chapter's Creative Commons license 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.



Automatic Generation of Improvement Suggestions for Legacy, PLC Controlled Manufacturing Equipment Utilizing Machine Learning

Wolfgang Koehler⁰⁰⁰⁰⁻⁰⁰⁰²⁻⁸⁰⁴⁶⁻³⁵⁰¹ (✉) and Yanguo Jing⁰⁰⁰⁰⁻⁰⁰⁰¹⁻⁹⁵⁸¹⁻⁴²¹⁵

School of Computing, Electronics and Mathematics, Faculty of Engineering, Environment and Computing, Coventry University
Priory Street, Coventry, UK, CV1 5FB
koehlerw@coventry.ac.uk

Abstract. The manufacturing industry and, for this research, the automotive manufacturing industry specifically, is always on the lookout for opportunities to improve production throughput with a minimum of investment. Identifying these opportunities often requires the observation of the current production process by experts. This paper is the continuation of the previous work 'Automated, Nomenclature Based Data Point Selection for Industrial Event Log Generation'. One of its aims is to provide strategies that can be used to pre-process an in-depth, slightly flawed industrial equipment log to allow for further analysis. The pre-processing is achieved by identifying the flaws, removing the non-value added events and a heuristic methodology to cluster the log into individual sequences. Expert knowledge then is encoded into engineering features to extend the log matrix and prepare it for machine learning model generation for identification of the complete cases. To derive value from the available data, the sequences are plotted into Gantt charts, and eight hypotheses are introduced that allow for automated annotations within this chart to highlight potential areas of improvement. Application of the framework to real life logs, obtained from stations considered bottlenecks within the evaluated automotive body shop, lead to the discovery of improvement potential between two and twelve seconds per cycle.

Keywords: Industrial Logs · Process Mining · Case Clustering

1 Introduction

This research aims to devise an automated framework that will, provided with the code of the programmable logic controller (PLC), monitor the desired production equipment and generate a Gantt chart of its actual sequence while highlighting areas of improvement. The proposed framework has been structured into three distinct approaches.

© The Author(s) 2021

J. Beyerer et al. (Hrsg.), *Machine Learning for Cyber Physical Systems*, Technologien für die intelligente Automation 13,
https://doi.org/10.1007/978-3-662-62746-4_10

The first function required is automated, nomenclature based data point selection and equipment log generation, as described in detail in the authors' previous publication [1]. Here the goal is the collection of start and end timestamps for all motions within a production cell. The relevant tags to be monitored are, based on their nomenclature, extracted from the PLC program and stored within a SQL database. The monitoring is done with a centralized workstation utilizing an Open Platform Communication (OPC) server. The necessary OPC groups and items are automatically generated. Changes within the status of the PLC tags will trigger an event which logs those changes in the database. In order to evaluate the quality of the obtained data, a quality matrix was devised and applied. The evaluation showed that the records' completeness was above 96% for real-life equipment data.

The second step is machine learning based pre-processing. The obstacle to overcome is clustering the event log data into cases as the raw data do not contain a reliable case identifier. Case clustering was achieved with the part present status within the station and a heuristic approach that allows for the identification of case-related setup, load/unload and reset events. Next, five hypotheses were formulated to create additional features for the data set based on expert knowledge. After tagging the trace classes manually, six different machine learning algorithms were applied with cross-validation. More details can be found in paragraph 3.1.

In chapter 3.2, an expert knowledge-based, heuristic generation of improvement suggestions is introduced. The eight hypotheses postulated were implemented using Python and applied to event logs of four real-life framing re-spot stations. A sequence chart for every style, based on the pre-processed event log was plotted. The issues found were automatically annotated within the same chart, and the findings for the four stations summed up.

2 Related Works

Plant floor systems, as described by Lee [2], were the first step towards the autonomous observation of manufacturing processes. They are logging critical parameters of the process which are used to create KPI (key points of interest) charts and to highlight potential bottlenecks. Next cyber-physical systems started to emerge. Their goal, to create a digital clone of the real-life production equipment, which can be used to create simulations and derive predictions, was also documented by Lee [3]. Jaber et al. [4] showed that predictions regarding required maintenance could also be obtained by applying machine learning techniques to vibration sensor data. The results could help to move the time of preventive maintenance closer to the predicted time of failure thus realising additional savings. Banerjee et al. [5] propose a similar approach. Instead of using vibration sensors, which normally are not an integral part of manufacturing equipment, they are utilising the already available sensors for fault detection.

Processes can not only be found in manufacturing but also for business transactions. Van der Aalst [6] started at the beginning of this century the develop-

ment of the research field of Process Mining. The aim is to discover the underlying process model of such business transactions based on logged transaction data. Hu et al. [7] realised that the proposed algorithms might also be beneficial for the discovery of process models within 'flexible manufacturing systems'. They proclaim that the derived model not only allows for validation of the actual process against the design intent but also can be used for further process improvements. These improvements were mainly focused upon the resources available. The literature review did not reveal any additional attempts to apply Process Mining algorithms to industrial equipment logs until 2014 when Son et al. [8] presented their research into discovering process models for the product flow from the first step of manufacturing to final shipment. Due to a lag of detailed data, these models, however, cannot be used to enhance the performance of the individual machines involved. Yahya's research [9] was also focused on the path of the product through manufacturing. He noted that the granularity needs to be chosen and the process model customised to the analysis' goal. Yang et al. [10] propose the enhancement of such high-level production data with the help of unstructured data like emails. Although the technical approach is presented in detail, it remains unclear what added benefits such approach yields.

Farooqui et al. [11] realised that the implementation of additional code within industrial robot programs allow them to record more details relating to their work sequence. They are proposing to apply Process Mining algorithms to the resulting log to discover a matching process model which is helpful for maintenance work and also supports decision making. Brzywczy et al. [12] also see the benefits of utilising low-level machine data for their research. They indicate that one of the significant hurdles to overcome is the grouping of activities into cases. According to their work, this is best achieved through knowledge-based identification of the beginning and the end of a case. Nowaczyk et al. [13] are tapping into the 'wisdom of the crowd' by evaluating groups of peers. Deviates one of the observed systems from the behaviour of the remaining, similar systems, it can be concluded that maintenance is required.

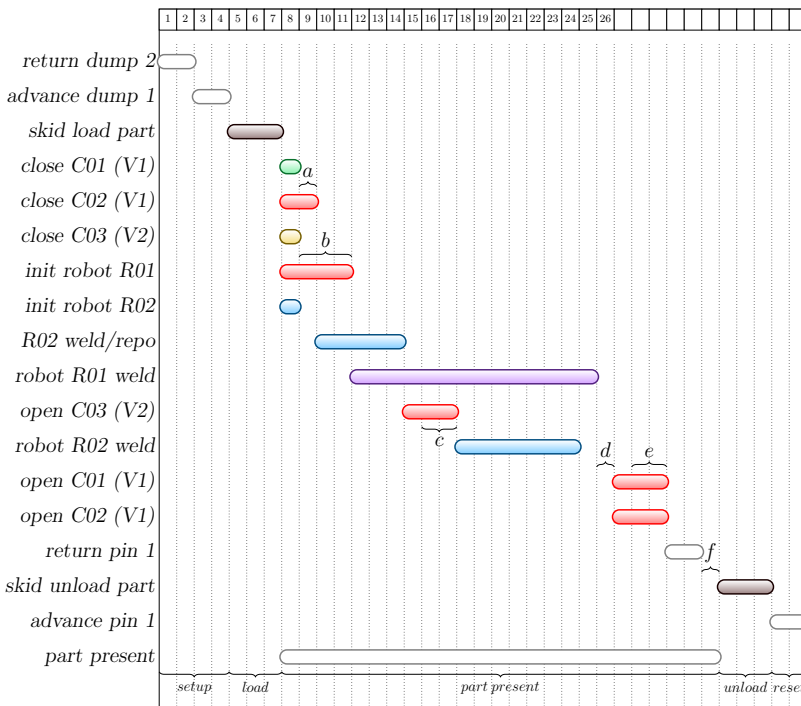
3 Hypothesis

3.1 Log Pre-Processing

As shown in figure 1, a machine sequence can be split up into five distinct sections. A part being present in the machine is a signal common to all manufacturing equipment. Therefore the load event must be the event just before the part is present. Analog the unloading activity is observed while the part no longer is present in the machine. In cases where a machine manufactures multiple different parts, a setup, just prior to the load, might be required. If the same events always happen before a load event for a given part type, then it can be reasoned that those activities must be setup related. Finally, some machines require some additional motions so that the part can be unloaded. In figure 1 this is shown as pin 1 returning. This event has to be reset before the next part

is loaded. Therefore a reset event is present if the same activity can be found after each unload event. Identifying the five sections described above allows for case identification within the equipment log.

Several hypotheses were devised that allow conclusions regarding the completeness of a case within an event log for automated production equipment. Within every cycle, there must be a load and an unload event. If a part is just passing through a station, it is even possible that those two activities are the only activities. Most non-robotic activities within the sequence have opposing motions. A typical example in figure 1 would be the closing and opening of clamp C01. If one of those two actions is missing, that could be an indicator for the logs incompleteness. Although robotic events do not have opposing motions, it is expected that a process follows the robots initiation and vice versa. Since log completeness is expected to be at a high level, it can be assumed that the most occurring trace class is complete. These knowledge-based rules can be used to annotate the log, and after manually tagging example logs, a machine learning model can be created which allows for the classification of the remaining cases.



3.2 Automated Improvement Potential Detection

Excessive manual cycles: During production, the equipment typically is in automatic mode unless a problem occurs that requires manual intervention. The machines within an automotive body shop often are specified to provide an up-time of 80+%. If excessive manual cycles are recorded daily, it can be concluded that there is a systematical problem which needs to be addressed.

Identical units: Several pneumatic cylinders are often connected to a single solenoid. The grouped cylinders typically have the same bore and stroke and therefore should require the same time to advance and return. Setup can impact the synchronous movement of the units. This fact can be found in the event log. An example is shown in figure 1 where cylinders C01 and C02 are attached to the same valve, but their closing time is different. This improvement potential is marked in red and labelled with (a). With $\Delta\tau_{Se}$ being the duration of a station event, $\Delta\tau_{Se'}$ being the duration of an equivalent event triggered by the same solenoid and $\lambda\tau(Se_{ref})$ the mean duration of an identical reference event, setup problems are present if

$$\Delta\tau_{Se} \neq \Delta\tau_{Se'} \vee \Delta\tau_{Se} \neq \lambda\tau(Se_{ref}) \quad (1)$$

Opposing motions: If a motion in one direction takes longer than into the opposing direction, a setup problem is present as well. A nomenclature based algorithm can identify which activities are opposing motions. The open events for C01 and C02 in figure 1 take longer than their corresponding closing events. Therefore the potential improvement is labelled with (e). Let $\Delta\tau_{Se}$ be the duration of station event Se and $\Delta\tau_{\overline{Se}}$ the duration of the events opposing motion then the setup is correct if:

$$\Delta\tau_{Se} = \Delta\tau_{\overline{Se}} \quad (2)$$

Double triggers: If there are programming errors, an equipment motion may be started, interrupted and restarted again. Such behaviour causes increased cycle time and is responsible for excessive mechanical wear. In the log, this can be identified by an event which has a start timestamp but no complete timestamp followed shortly after by another event for the same activity that has both timestamps. Since events can happen twice within a case, the detection algorithm has to consider that the opposing motion did not happen in between. If the start timestamp of a station event Se is defined as $\tau_s(Se_n)$, the complete event as $\tau_c(Se_n)$ and the opposing motion of that event as \overline{Se}_n then a double trigger is present if

$$\tau_s(Se_n) \neq \emptyset \wedge \tau_c(Se_n) = \emptyset \quad (3)$$

is followed by an identical event with

$$\tau_s(Se_{n+x}) \neq \emptyset \wedge \tau_c(Se_{n+x}) \neq \emptyset \quad (4)$$

as long as

$$Se_{n+1} \dots Se_{n+(x-1)} \neq \overline{Se_n} \quad (5)$$

Bouncing motions: The term 'bouncing motion' was coined for a motion that reaches its end position but, due to the mechanical setup, bounces back so that it needs to be triggered once again to arrive at the stop position. In the event log, this can be identified by an event with start and complete timestamps followed shortly after by again the same event with start and complete timestamps without the opposing motion being recorded in between. Double triggers and bouncing motions manifest themselves in figure 1 similar to the opposing motion hypothesis mentioned previously (figure 1 (c)). However, the underlying data allow the discovery of the actual root cause. Based on above definitions a bouncing motion can be detected if

$$\tau_s(Se_n) \neq \emptyset \wedge \tau_c(Se_n) \neq \emptyset \quad (6)$$

is followed by an identical event with

$$\tau_s(Se_{n+x}) \neq \emptyset \wedge \tau_c(Se_{n+x}) \neq \emptyset \quad (7)$$

as long as

$$Se_{n+1} \dots Se_{n+(x-1)} \neq \overline{Se_n} \quad (8)$$

Gaps: In the automotive body shop domain, there should be no period within a sequence, where there is no motion occurring. Considering that for this experiment, a variance of $\sim 100\text{ms}$ within the timestamps was found, it can be concluded that any gap $> 200\text{ms}$ marks an area of possible improvement. Gaps can be caused either by programming errors or by external circumstances which are not recorded. A typical example of a gap is marked with the letter (d) within figure 1. Gaps can be detected by splitting up the timeline t of a case into bins. Then the number of events that fall within one such bin are counted and represented by ξ_t . Based on these definitions, a gap is present if

$$\sum_{t=x}^{x+200\text{ms}} \xi_t = 0 \quad (9)$$

with x being any value between the start timestamp $\tau_s(Se_i)$ of the incoming event Se_i and the start timestamp $\tau_s(Se_o)$ of the outgoing event Se_o of a case.

Station blocked: A particular case of the above described external circumstances, is the station being blocked. A blockage is caused by the next station not being ready to receive the completed part. In that case, the event data will show a gap before the unload event. A blocked condition has been highlighted within figure 1 with the letter (f). Let $\Delta\tau_p(Se_o)$ be the distance from the outgoing station even Se_o to the second last event then a blocked condition exists if

$$\Delta\tau_p(Se_o) > 0 \quad (10)$$

Special Event - Robot Initiation: The duration of the robot initiation process was found to be varying substantially. During this timeframe, the robot receives its program number and a start signal which triggers it to move to a pounce position. Typically this routine takes a maximum of two seconds what leads to the assumption that a robot initiation lasting more than two seconds is suspicious. Such a situation is shown in figure 1 with the letter (b). With $\Delta\tau_{Rinit}$ being the duration of a robot initiation event a reason for suspicion is present if

$$\Delta\tau_{Rinit} > 2sec. \quad (11)$$

4 Evaluation

4.1 Log Pre-Processing

To prepare the log for further processing the activities stemming from a double trigger or bouncing motion event, as described in chapter 3.2 were combined by merging the start timestamp of the first with the end timestamp of the second record. Next, the first activities, along with other, incomplete log items were removed. Python algorithms were developed to identify the five sections defining a case.

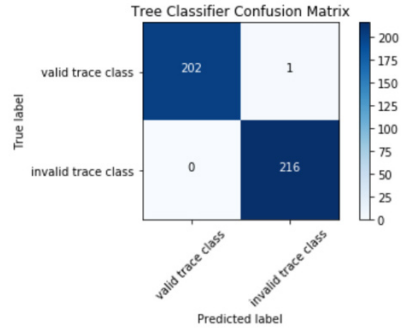
The five hypotheses introduced in section 3.1 created the basis for Python algorithms that can add engineering features to an industrial log. For 'load/unload present', 'station in bypass' and 'robot initiate & process present' a binary value of 0 or 1 was chosen. For the remaining features 'most occurring' and 'opposing motions present' a percentage, expressed as value between 0 and 1, was used. Manual tagging was performed for 500 random cases within the log available. Various machine learning algorithms, included in the Python scikit-learn package [14], were applied with cross-validation to the resulting matrix. Table 1 shows that a simple decision tree classifier, after tuning the hyperparameters, already achieves a 99% accuracy with +/-1% deviation. The associated confusion matrix, shown in figure 2, also exhibits no false positives.

4.2 Automated Improvement Potential Detection

For evaluation purposes, reasoning based algorithms for all of the above hypotheses (chapter 3.2) were implemented using Python and applied to event logs of

Table 1. Accuracy Of Classifier Models

	accuracy without hyper parameter tuning	accuracy with hyper parameter tuning
gradient boosting classifier	94% +/- 8%	99% +/- 1%
random forest classifier	93% +/- 8%	99% +/- 1%
decision tree	93% +/- 8%	99% +/- 1%
gaussian naive bayes	87% +/- 5%	87% +/- 5%
support vector classifier	91% +/- 8%	95% +/- 3%
logistic regression	91% +/- 12%	95% +/- 2%
k-nearest neighbors	90% +/- 7%	99% +/- 2%

**Fig. 2.** The Confusion Matrix

four real-life framing re-spot stations. The issues automatically discovered for the four stations are summed up in the table 2.

Table 2. Evaluation Results For Four Body Shop Re-spot Stations

	station 1	station 2	station 3	station 4
number of cycles recorded	1157	1181	1185	1184
excessive manual cycles	< 0.5%	< 0.5%	< 0.5%	< 0.5%
time differences for identical units	0 sec.	0.2 sec.	0.2 sec.	0 sec.
time differences for opposing motions	1 sec.	2.3 sec.	1.2 sec.	1.9 sec.
double triggers events (# of times)	7 (88)	13 (1033)	9 (203)	11 (2045)
bouncing motions events (# of times)	0	2 (104)	0	0
gaps	0.8 sec.	6.4 sec.	10.7 sec.	0.4 sec.
station blocked	0.3 sec.	5.5 sec.	0.4 sec.	0.2 sec.
difference typical to fastest variant	0 sec.	0 sec.	0 sec.	10 sec.

5 Conclusion And Future Works

In a typical automotive body shop, one can almost always find a few stations which do not meet their expected throughput. This shortcoming might be due to long cycle times or increased maintenance activities. These stations are holding back the output of the body shop and therefore are termed 'bottlenecks'. Removing those few bottlenecks can increase the output of the body shop as a whole.

In this paper, hypotheses were formulated that allow for the automated generation of equipment logs and the subsequent discovery of hidden manufacturing potential. As proof, the prepositions were encoded and applied to real life equipment logs taken from bottleneck stations within an automotive body shop. The insight gained was automatically marked within Gantt charts. The results presented show that there is, for the evaluated stations, an improvement potential

ranging from 2.1 seconds to 12.5 seconds. These findings lead to the conclusion that the analysis effort is worthwhile, even if it is assumed, that the current production process is well understood.

It is believed that more discoveries are possible. Therefore research will continue to focus on potential information gain based on more elaborate Process Mining techniques initially developed for business process analysis.

References

1. Koehler, W., Jing Y.: Automated, Nomenclature Based Data Point Selection for Industrial Event Log Generation. *International Conference on Intelligent Data Engineering and Automated Learning*, 31–40 (2018)
2. Lee, J.: E-manufacturing—fundamental, tools, and transformation. *Robotics and Computer-Integrated Manufacturing*. 501-507 (2003)
3. Lee, J., Bagheri, B., Jin, C.: Introduction to cyber manufacturing. *Manufacturing Letters*, 8. 11-15 (2016)
4. Jaber, A.A., Bicker, R.: The state of the art in research into the condition monitoring of industrial machinery. *Int. J. of Current Engineering and Technology*. 1986-2001 (2014)
5. Banerjee, T.P., Das, S.: Multi-sensor data fusion using support vector machine for motor fault detection. *Information Sciences*. 96-107 (2012)
6. Van der Aalst, W.M.: *Process Mining: data science in action*. Springer Berlin Heidelberg. (2016)
7. Hu, H., Li, Z. and Wang, A.: Mining of flexible manufacturing system using work event logs and petri nets. In *International Conference on Advanced Data Mining and Applications* 380-387 (2006)
8. Son, S., Yahya, B., Song, M., Choi, S., Hyeon, J., Lee, B., Jang, Y. and Sung, N.: Process Mining for manufacturing process analysis: a case study. In *Proceeding of 2nd Asia Pacific Conference on Business Process Management* (2014)
9. Yahya, B.N.: The development of manufacturing process analysis: lesson learned from Process Mining. *Jurnal Teknik Industri* 95-106 (2014)
10. Yang, H., Park, M., Cho, M., Song, M. and Kim, S.: A system architecture for manufacturing process analysis based on big data and Process Mining techniques. In *2014 IEEE International Conference on Big Data* 1024-1029 (2014)
11. Farooqui, A., Bengtsson, K., Falkman, P. and Fabian, M.: From factory floor to process models: A data gathering approach to generate, transform, and visualize manufacturing processes. *CIRP Journal of Manufacturing Science and Technology* (2018).
12. Brzychczy, E. and Trzcionkowska, A.: Creation of an Event Log from a Low-Level Machinery Monitoring System for Process Mining Purposes. In *International Conference on Intelligent Data Engineering and Automated Learning* 54-63 (2018)
13. Nowaczyk, S., Sant’Anna, A., Calikus, E. and Fan, Y.: Monitoring Equipment Operation Through Model and Event Discovery. In *International Conference on Intelligent Data Engineering and Automated Learning* 41-53 (2018)
14. Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V. and Vanderplas, J.: Scikit-learn: Machine learning in Python. *Journal of machine learning research* 2825-2830 (2011)

Open Access This chapter is licensed under the terms of the Creative Commons Attribution 4.0 International License (<http://creativecommons.org/licenses/by/4.0/>), 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 license and indicate if changes were made.

The images or other third party material in this chapter are included in the chapter's Creative Commons license, unless indicated otherwise in a credit line to the material. If material is not included in the chapter's Creative Commons license 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.

