Journal of Physics: Conference Series

PAPER • OPEN ACCESS

Metric Based Dynamic Control Charts for Edge Anomaly Detection in Factory Logistics

To cite this article: Patrick Eschemann *et al* 2022 *J. Phys.: Conf. Ser.* **2352** 012010

Journal of Physics: Conference Series **2352** (2022) 012010 doi:10.1088/1742-6596/2352/1/012010

Metric Based Dynamic Control Charts for Edge Anomaly Detection in Factory Logistics

Patrick Eschemann[1], Philipp Borchers[2✉], Dennis Lisiecki[3], Jan Elmar Krauskopf[1]

[1] OFFIS - Institute for Computer Science
[2] DLR e.V. - Institute Systems Engineering for Future Mobility
[3] Carl von Ossietzky Universität Oldenburg, dennis.lisiecki@uni-oldenburg.de
✉ *corresponding author*

E-mail: patrick.eschemann@offis.de, philipp.borchers@dlr.de✉, dennis.lisiecki@uni-oldenburg.de, jan.krauskopf@offis.de

Abstract. The optimization of transport logistics in production environments is a holistic task for the factory of the future. Autonomous guided vehicles that perform transport jobs in factories are facing this challenge and have to detect, react and prepare to unforeseen changes and anomalies in the production system. Due to data protection concerns, details like production plans are often not available for an external transportation system. Hence the anomaly detection has to be based on self-collected and observed data of the transport system like occurred transport needs or the evolution of internal metrics. In this paper we infused a production system with manufacturing process anomalies and demonstrate a detection based on the observation of transport needs to overcome the gab caused by restricted information. For that detection we extended classic control charts to work with expected values based on learned dynamic production characteristics. The system sets a tolerance field as narrow as possible around dynamically determined values, resulting in an average precision of 95% for detection unusual number of transport jobs.

1. Introduction

As the level of automation within manufacturing companies increases, the call for automated logistics becomes more urgent, as these are able to act independently without constant supervision. Automated Guided Vehicles (AGVs) are increasingly used to replace partial or entire logistics structures within a factory. This leads to new technical and organizational challenges as the management of a robot fleet requires supervision. By transporting material within a factory, robots produce and observe information. The presented concept in this paper shall enable robots to process these information in a reasonable manner.

Dynamics due to production fluctuations in factories are plant specific and can change within short time intervals. Example situations are an unexpected order of certain products, a machine or robot failure and other situations with potential to significantly disrupt the operating condition of the factory. These situations directly affect the utilization of machines as well as the amount of transport tasks and are therefore considered as abnormal factorial circumstances or anomalies in general.

The contribution of this paper is an approach for anomaly detection to be integrated in an AGV fleet. Anomaly detection describes techniques to discover uncommon or unpredicted behavior in datasets [1]. The fleet receives transport jobs published by a Manufacturing Execution System (MES). Each individual AGV of the fleet calculates a bid value to decide which member is best suited to take over a transport job. Operating systems of transport robots typically collect continuously data via sensors [2]. While executing the job, the AGVs collect data about driven distance, required time, start and endpoint, which can be used for analysis without conflicting confidentiality requirements of companies. This has to be considered when the fleet of AGVs is developed and sold by another company than the one which will be using them.

By processing data robots shall be enabled to develop a context awareness that helps to mitigate and reduce the impacts of anomalies.

The operating systems of AGVs furthermore implement configurations that determine the behavior / strategy of the robot. The reconfiguration on the level of the operating system offers possibilities to adapt the behavior. This can help in mitigating anomalies. Most configurations allow to change the priorization of goals, the battery management, speed parameter and the transport order distribution among the robots. Changes in the configuration of the robots affect the factory globally, in contrary to configurations changes on the machine level which are limited to local and slower impacts. Therefore this approach focusses on adaption of the robot configuration and not the machine configuration. The machine configuration is set in a way to process incoming material as fast as possible independently of the state of the factory. For the fastest anomaly mitigation, an automated adaption of robot behavior of the AGVs to the abnormal situation would be ideal. However the AGVs, as a third party asset, are expected to have limited access to confidential data of the factory. Therefore, the fleet should be able to process and analyze data that is generated or observed by themselves.

The processing of this data with an anomaly detection trigger automatically a reconfiguration component. This can for example influence the bidding process of the robots to mitigate anomalies and shorten the recovery phase after an anomaly. The concept of the strategy change is introduced in [3].

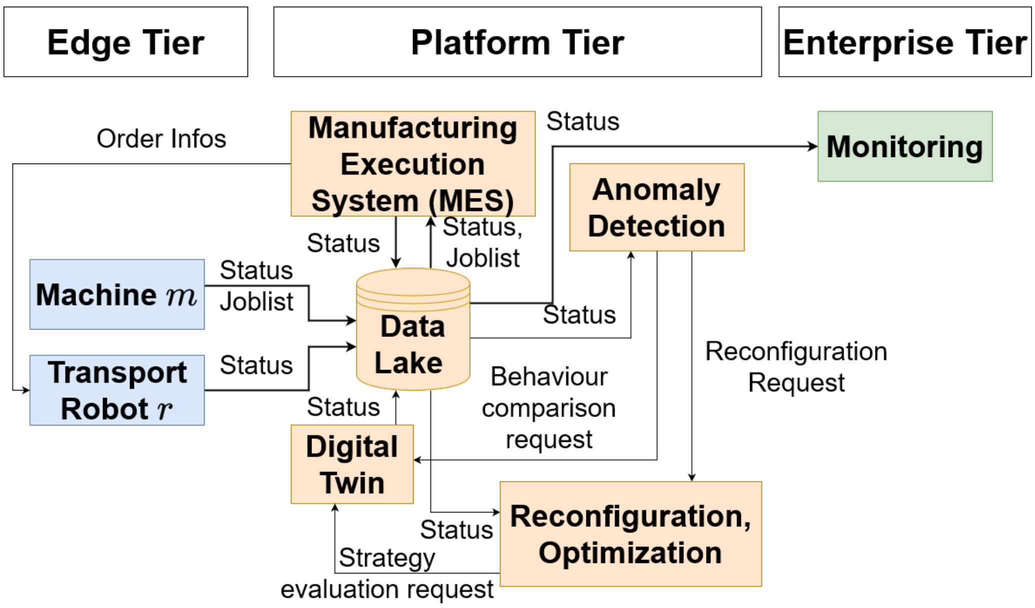

Figure 1. Architecture for the Optimization of an AGV Fleet.

This work implements parts of the requirements of an overarching architecture which is based on the "industrial internet reference architecture", precisely the three-tier architecture pattern, which distinguishes edge, platform and enterprise tiers [4]. The architecture illustrates the context of this work in the underlying european project Cyberfactory#1 [5], see Figure 1.

The tiers are dependent parts of an overall factory structure. The edge-tier describes all facilities and autonomous working devices that are located in the factory hall. The platform-tier is superordinate of the edge-tier as it monitors all systems of the edge tier and platform tier. The enterprise-tier monitors all data that is distributed from the edge-tier and the platform-tier, [6].

The edge and platform tiers are investigated in [7] and [3]. The present article elaborates further in platform tier, specifically in the "Anomaly Detection" component.

The work is organized as follows. Section 2 provides an overview of related work on this topic. Section 3 introduces a three step approach to anomaly detection, which consists of a simulation, a learning component and the anomaly detection itself. Section 5 shows results of experiments and Section 6 concludes the paper with an outlook for future work.

2. Related work

In a recent survey Musa and Bouras presented a structured overview of research directions on anomaly detection [1]. They identified five main application domains for anomaly detection. These are intrusion detection, fraud detection, industrial damage detection, image processing and the public health domain. The authors state that wear and tear damage as the main cause of anomalies in the industrial damage detection requires an early detection to prevent further growth and losses. Further they identified three types of anomalies, the point anomaly, contextual anomaly and the collective anomaly. Point anomalies are characterized as significant deviation from the average or normal distribution of the entire dataset. They state that these are typically easy to detect. Contextual or conditional anomalies are determined by deduction. A certain data point is an anomaly because of its context. To detect these kind of anomalies domain knowledge is required. Collective anomalies are not individual data points of the dataset, but a collective set of the entire dataset. Anomalies can be detected in a supervised, semi-supervised or unsupervised mode. The supervised mode requires a labeled normal and anomal dataset, whereas the semi-supervised mode only requires the labeling of the normal data. Unsupervised modes are applied when it is not possible to label the data in a senseful way. To detect anomalies instrinsic properties are used like densities or distances.

A common tool for detecting process anomalies are quality control charts [8]. These are also known as Shewhart charts which were introduced by Shewhart to monitor manufacturing processes. With control charts, it can be checked, if a value deviates too much from its statistical mean over time. We provide an example of a classical control chart in Figure 2.

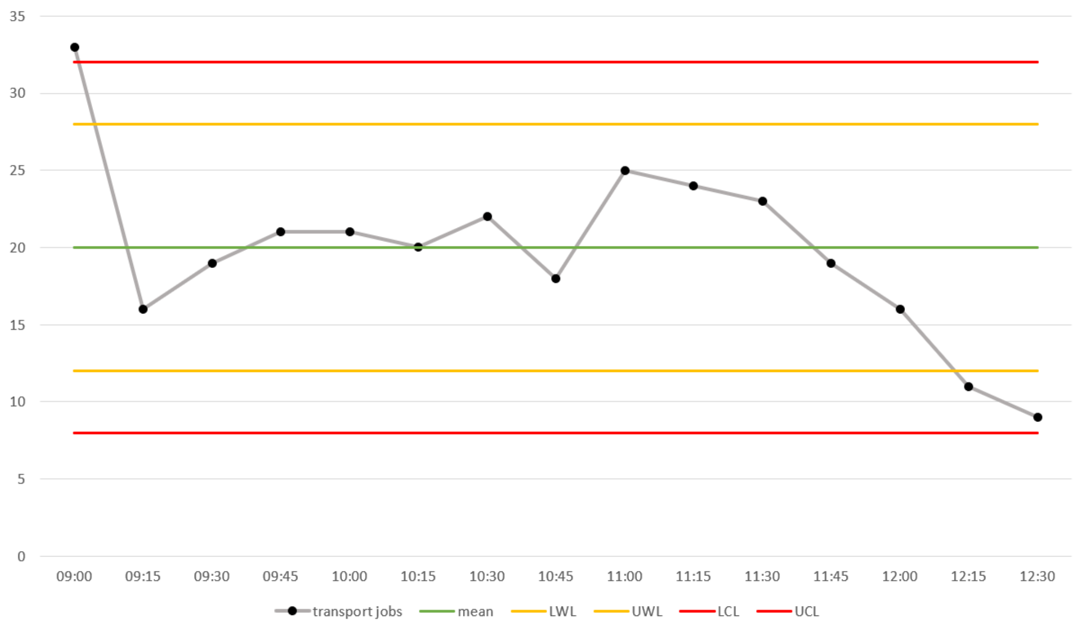

Figure 2. Classic control chart with dynamically changing number of transport jobs.

The y-axis reflects the current number of the observed value, the time is plotted over the x-axis in 15 minute intervals. The green line reflects the statistical mean μ of the observed value based on normally distributed historical data with a standard deviation σ of 4. The yellow lines form the upper and lower warning limits (UWL, LWL) respectively. They are calculated according to the the formula: $WL = \mu + 2 * \sigma$. If the data is normally distributed, then 95.45% of the observed values are between both warning limits. The red lines are the upper and lower control limit (LCL, UCL), according to $CL = \mu + 3 * \sigma$. 99.73% of the observed values are in this range. Typically there are defined rules to detect an anomaly before the red lines are breached, like:

(i) The current value is outside of the control limits.
(ii) Seven consecutive values are above or below the mean (so-called run).
(iii) Two out of three consecutive values are beyond the warning limit.
(iv) Seven consecutive values follow a non expected trend.

Examples on the utilization of control charts for industrially used robots in manufacturing environments are given in [9], [10]. Furthermore the authors in [11] use control charts for fault detection in manufacturing robots. They monitor the wear and tear of a robot arm by logging the vibrations in a control chart in order to monitor faults at early stages. For this, constant upper and lower bounds are specified in the used control charts. However, if we monitor the number of outputs from a machine in a factory, then the expected number of outputs may vary over the course of a day, e.g. due to a lower production efficiency during nightshifts. Hence, we consider control charts with fixed limits to be unsuitable for anomaly detection in our factory setting.

A possible approach to apply a dynamic upper and lower limit are Bollinger bands. The upper and lower limits of Bollinger bands are calculated by applying a standard deviation to a moving average. Bollinger bands are mainly used by traders and investors to indicate the current volatility and momentum of markets in order to decide on market entry and exit points [12]. In case of manufacturing, the authors in [13] utilize Bollinger bands to detect anomalies in patterned fabric during manufacturing. Their success rate in anomaly detection is around 98%. However, the authors state that their approach has trouble with detecting smaller anomalies in their setting.

The authors in [14] used quality control charts to detect failures and break downs of a machine an a supply chain. Similar to the use case of our paper, the outcome of the machine has to be delivered other locations and other factory entities are relying on the machine. The quality control charts are part of an integrated quality and maintenance model with respect to the effect and costs for the overall factory. In contrast to the scope of this work, the authors observed the machine directly from a factory perspective and not from a transportation system' viewpoint where production details are not available.

Another alternative to anomaly detection are *neuronal networks*. Basically, a neural network fits a function with training data via a process called backpropagation. The fitted network can then approximate the correct output to given input data [15]. The authors in [16] feed digital sensor data to a neural network in order to classify anomalies in manufacturing processes which may predict factory downtime. They utilize two different neural network based models in order to detect anomalies. To address the issue of collecting anomalous data in a non-anomalous environment, they utilize unsupervised learning with an autoencoder type of neural networks. This approach produced a missclassification error of 1.79 % on their test data.

In the next section, an adaption of the control chart is presented, which is usable in dynamically changing environment.

3. Anomaly Detection Concept

The concept for anomaly detection used in this article is embedded in a three layered optimization architecture presented in Figure 3.

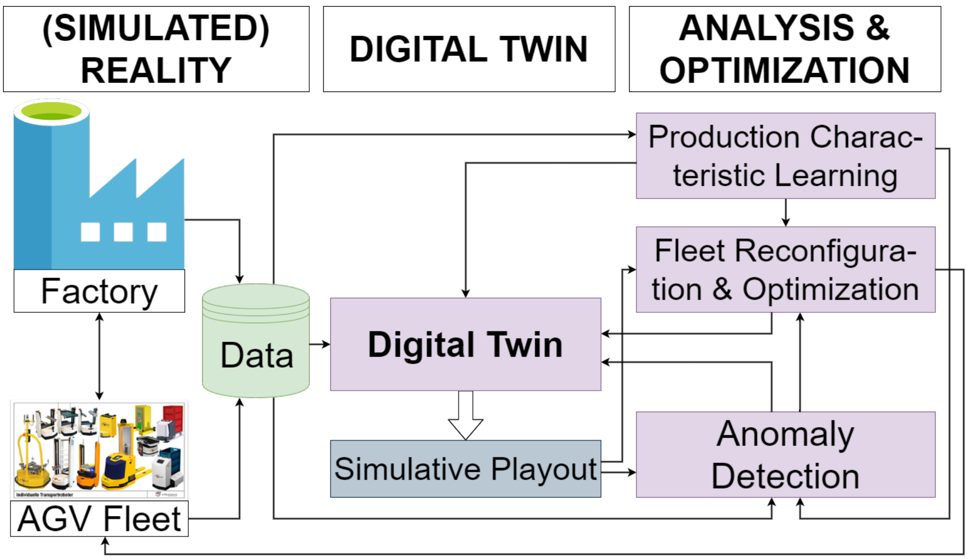

Figure 3. Architecture for the Optimization of AGV Fleet Behaviour with embedded Anomaly Detection [7].

The "Simulated Reality" layer consists of an AGV fleet transporting products in a factory as shown in Figure 4. We consider a factory that has sources to store the resources (e.g. sources q_1 and q_1 in Figure 4) and sinks to store the final products (e.g. sinks s_1 and s_2 Figure 4). Additionally there are machines to process resources into products. Note that machines can produce products for other machines. If a transport of a product is possible, like a machine needs and a warehouse can offer products, a manufacturing execution system (MES) detects the possibility and decides to publish a new transport task between machines and warehouses. Then the MES publishes corresponding transport requests to the transportation system (TS), in our case the AGV fleet for execution. During this production and transportation process, a lot of data is available. This data includes transport job updates, like timestamps, when tasks has started or finished as well as the evolution of robot Key Performance Indicators (KPI), like battery level or driven distances. In [3] we presented an approach to model and simulate the interplay of a production process and transport robots in order to get those data.

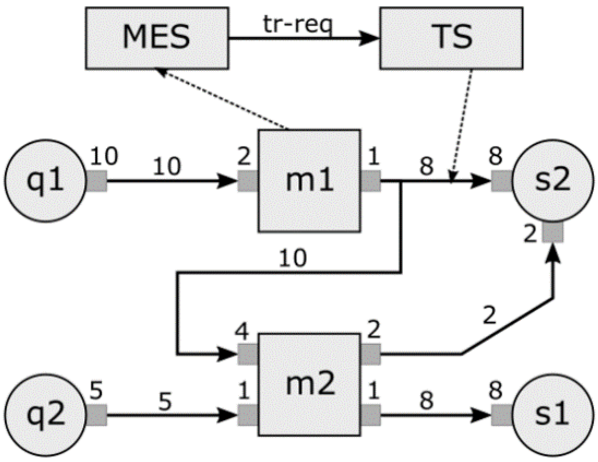

Figure 4. Schematic overview of an exemplary factory.

To predict the future behavior of the real environment, the optimization architecture in Figure 3 has a "Digital Twin" (DT) layer for digital representation components, that are taking current data or data from the near past to update its internal status. Furthermore, the component applies results of analysis of transport data from the "Analysis & Optimization" layer to improve the prediction. The outcome of the DT is a prediction of the future behavior of the robot fleet and its environment.

In the "Analysis & Optimization" layer components for analyzing historic data, detecting anomalies and the reconfiguration the AGV fleet are located. In the "Production Characteristic Learning" component the historic production data are taken to search for regular patterns and dependencies in the occurrence of transport tasks. The resulting production dynamics represent the knowledge about the production process and are provided to the other components for simulation purposes of transport dynamics. This component and different production dynamic prediction methods was elaborated in [7]. The component "Anomaly Detection" is capturing the work and methods we are considering in this article. It compares the prediction results from the digital twin with the data from the near history to detect an anomaly and raise an alert. Such alerts are used by the "Fleet Reconfiguration & Optimization" component to prepare the fleet.

3.1. Anomaly and Metrics

For anomaly detection the AGV fleet is limited to transport and AGV related data. Hence, anomalies caused in the production process can be only noticed via variations from the regular transport request behavior. Anomalies caused by robots (e.g. a malfunction of the driving engine) can be noticed via AGV KPIs like the driven distances or the battery level. In Table 1 we listed a set of anomalies that are relevant for the AGVs and added the metric to detect the anomaly.

Table 1. Anomalies and Metrics

Nr.	Anomaly Description	Metric
1	Robots detect an anomal increase or decrease of global or local transport jobs.	$\frac{\text{Number of jobs}}{\text{Time interval}}$
2	Robots do not process transport orders. Productivity decreases.	$\frac{\text{Completed Jobs}}{\text{Number of jobs}} * \frac{1}{\text{Time interval}}$
3	Single robot handles significantly more (or less) jobs than usual or compared to the rest of the fleet.	$\frac{\text{Completed Jobs}}{\text{Time interval}}$
4	Time to process a certain job type takes longer than usual.	$\frac{\text{Transport Job}}{\text{Time interval}}$
5	Distances to process a certain job type are longer than usual.	$\frac{\text{Driven distance}}{\text{Transport job}}$
6	Robot recharges its battery more frequently or spends more time overall than usual at the charging station.	$\frac{\text{Charging time per week}}{Robot}$

For the experiments in the following chapters, the focus was on the recognition of an unusual number of jobs according to anomaly 1.

3.2. Dynamic Control Charts

A metric based dynamic control limit approach can be used to identify the above anomalies as deviations between expected and observed factory behavior. As mentioned in Section 2 the Shewart chart works for controlled tasks with normally distributed observed data but shows several disadvantages. When it is used to detect anomalies within a factory environment, the data is dynamically distributed due to delayed deliveries, shift changes, machine breakdowns, maintenance breaks, capacity adjustments etc. Therefore the classic control chart detects a high number of false alarms without any adaption.

In the example of Figure 2 the work in the factory starts at nine o'clock, leading to an increased number of transport jobs at short notice. Accordingly, an anomaly is detected because the number of transport jobs in this first 15 minute time interval is beyond the warning limit. This increase is belonging to expected behavior of the factory, as it occurs regularly at every day. Further anomalies are triggered by the rule set towards midday. However the decreasing amount of transport tasks in this example factory is due to lunchtime and a change of shift. This example shows that there are expected trends within the daily manufacturing process.

In the following, the adaption of the control chart concept is presented, which enables a more accurate anomaly detection for the defined metrics. It incorporates one limit for detecting unusual numbers of transport jobs which is similar to the warning limit of the standard approach. There are no additional rule sets, except that calculated negative values are treated as zero. Anomalies are detected only, when the current value is outside of the anomaly limits. Since the transport order amount is not expected to be normally distributed, the mean value as basis for determining the anomaly limits (AL) is dynamically determined. The quality of the approach strictly depends on the calculation of this dynamic lead value. To determine it as basis for calculating the limits there are two options. On the one hand the weighted moving average of previous transport jobs and the mean standard deviation can be used to calculate the borders (moving average), on the other hand the expected values provided by the DT can be used.

With the first option the decrease of transport jobs towards midday due to shift change or a work break does not lead into an anomaly detection. The control limits adapt to the trend as they are calculated by using the mean value of the weighted sum of a number of last transport jobs according to $\mu = (x_1 * a + x_2 * b + x_3 * c)/3$ and the mean standard deviation of historic data $AL = \mu + 2 * \sigma$. The larger the number of past values considered, the slower the lead value and therefore the limits adjusts. Generally the optimal number can be found experimentically so that it fits to the targeted factory.

The architecture in Figure 3 shows that the anomaly detection is connected to the DT which supplies expected values based on the learned production characteristics from history data. Using an expected value from the digital twin as the second option can represant the dynamic behavior due to peaks that repeatedly occur to certain times (shift change, start-up time). For calculating the control limits the mean standard deviation of the current timeframe is used. This approach leads to time dependent detection ranges, see Figure 5.

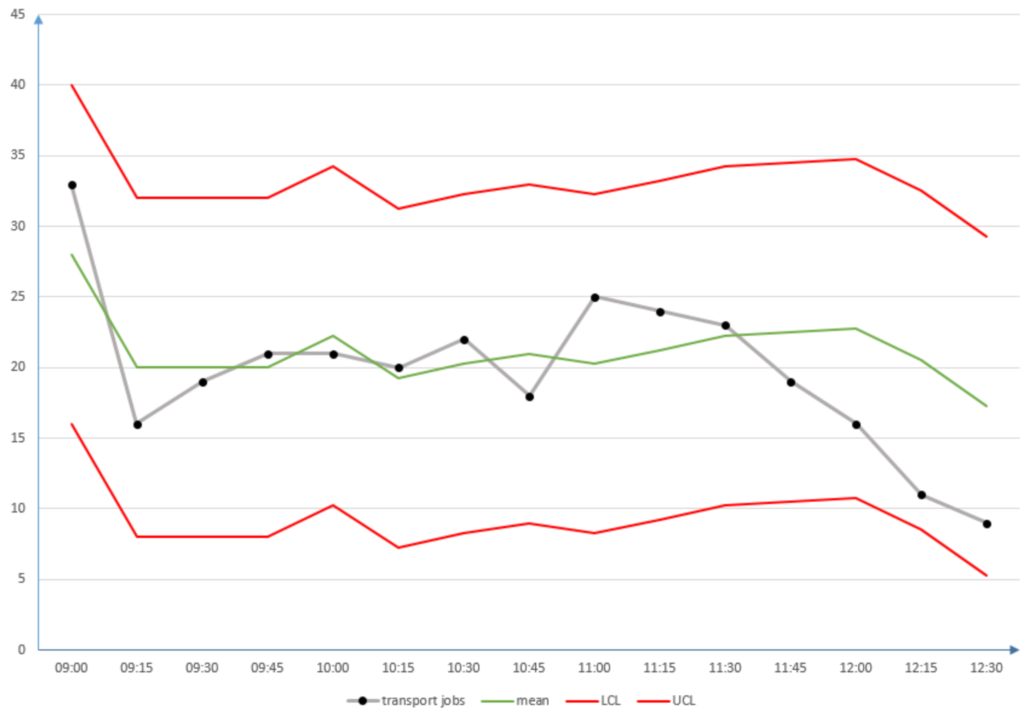

Figure 5. Adapted Control Charts for Factory Environments.

Each option has advantages and disadvantages. The weighted moving value cannot represent spontaneous peaks e. g. at the start of a workday. Furthermore an attacker can take advantage of the fact that the weighed moving average has no upper or lower limitation. It is possible to slowly increase the number of transport task to arbitrary numbers. In case of fluctuations in production, this control chart adapts dynamically depending on the number of values that are considered in the average value calculation.

If a low number of transport jobs is expected, the lower control limit value can be negative. In this case it is set to zero, so that during these times only too many but not too less transport jobs can be detected. However this approach has theoretically no upper or lower limit.

Comparing both approaches shows better results when using the expected values of the digital twin to create the control chart. An open topic is the combination of both options for further improvement.

4. Simulation & Anomaly detection

For the experiments in this article we used a factory with four machines and one warehouse that prints and folds magazines (Figure 6). We uses tree AGVs to transport products according to the production flow, that is indicated by red arrows in Figure 6.

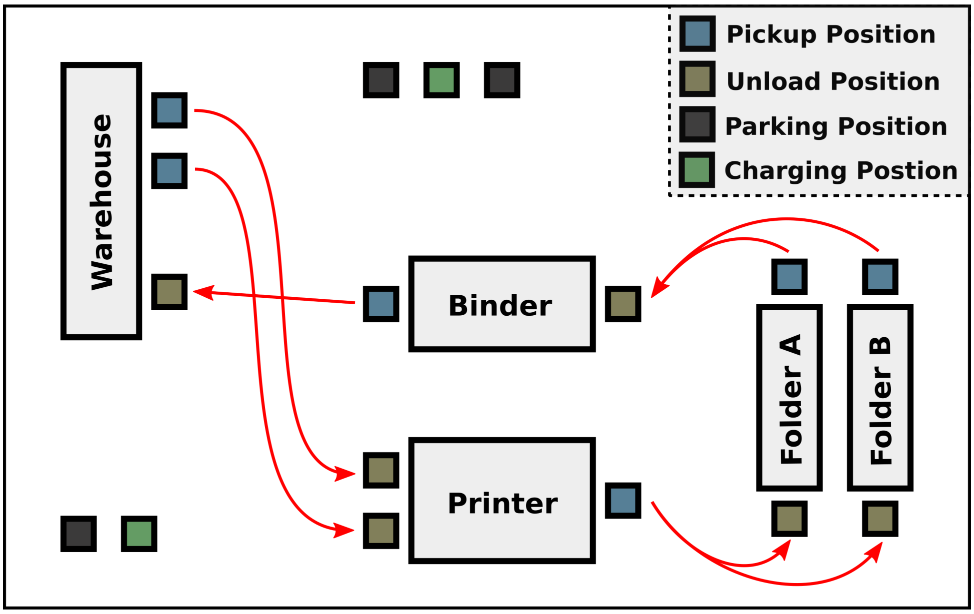

Figure 6. Schematic Overview of used factory including transport flow of products.

We simulated the production and transport interplay for 20 days to get a clean data set for transport needs under normal working conditions. From these data we calculated the average amount of transport request for every 15 minute in the week. Together with the standard deviation of the number of requests, we have the expected mean line and limits for the dynamic control charts.

To produce anomalies, regardless of whether they were caused by a hacker attack or a system failure, we simulate 10 production days, where the production times of the machines are manipulated to the fastest or slowest possible times at single hours in the 10 days. When machines spontaneously producing significantly slower or faster then under normal conditions, the resulting transport requests are expected to be higher or lower respectively.

From the transport data resulting from this 10 days run with anomalies, we again take the number of transport requests for every 15 minutes in these day. Then we compare the number of tasks with the expected ones from the clean data set in the dynamic control chart for every day. To avoid a detection of anomalies caused by the simulation start, where the transport need is usually higher, we start the detection at the second day [17].

In Figure 7 the anomaly detection with the dynamic control charts is presented for one working day. The x-axis is divided in 2 hour time increments, the y-axis displays the number of transport jobs with the range $[0, 8]$. According to the chart there are higher numbers of transport requests than expected in the morning between 5:00 and 6:00 as well as in the evening at 23:00. Over the overall anomaly run, we observed 23 15-Minute intervals where the upper warning limits is breached, and 59 respected upper warning limit breaches.

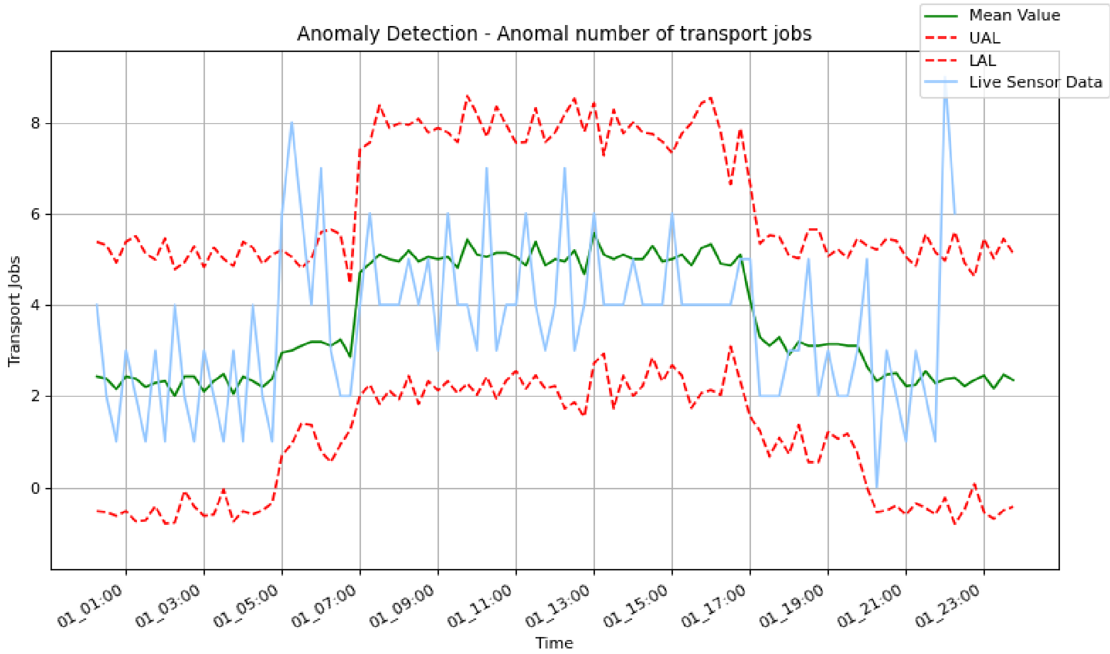

Figure 7. Exemplary day of the simulation where anomalies were detected.

5. Evaluation

To evaluate the anomaly detection results from Section 4, we compared the number of interval breaches, that indicates the detection of an anomaly, with the number of production time modifications, that is the source of the anomaly in the simulation. In the simulation run with anomaly, we infused 22 anomalous events, 13 fastest production times and 9 slowest production. As the anomalous events hold for an one our, we count them as detected, if at least one warning limit breach in one corresponding 15 Minute interval is detected.

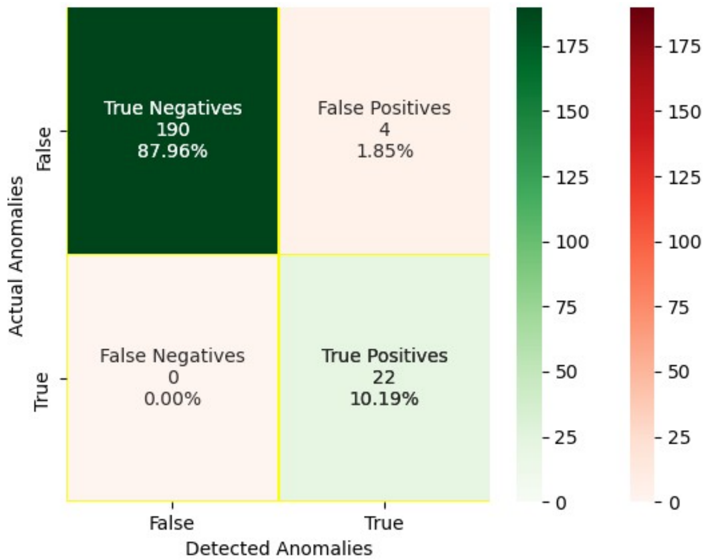

Figure 8. Confusion Matrix at the Example of an Anomal Increase of Transport Jobs.

To evaluate the anomaly detection, we used the confusion matrix in Figure 8, that shows the performance of the concept. The rows correspond to the actual anomalies that were triggered by the production time manipulation. The columns correspond to the detected anomalies. The diagonal from top left to bottom left shows the correct classifications coloured by the green heatmap. The diagonal from top right to bottom right shows the number of false classifications coloured by the red heatmap. All anomalies were detected by the algorithm. Therefore the percentage of true anomalies that were correctly classified (sensitivity) is 100 %. Furthermore the percentage of non anomalous events that were correctly classified (specificity) is $\sim$ 98 %. However the anomaly detection raised 4 false alarms within 9 observed days, leading to a positive predictive value (precision) of $\sim$ 95 %.

In judging the overall performance of this anomaly detection, there is a difficulty caused by the determination of the absolute number of verified anomalies. Follow-up anomalies are not considered in the assessment. Depending on the severity of the anomaly, we observed in the simulation data, that the factory operations require 30-120 minutes to return to a steady state. The fluctuations caused by this effect, trigger further anomalies that have not been programmed into the data set. However, only intended anomalies were matched when the matrix was set up. This means that the detection may also represent anomalies that occur as a result of mitigation.

6. Conclusion & Outlook

In this article, an approach for detecting production process anomalies by observing transport requests with dynamic control charts was introduced. It is based on a extension of control charts to make them deployable in dynamically changing environments such as different factories.

If the predicted transport requests are low in general, the lower limit of the control chart can be negative or zero. Because a negative amount of transport requests is not possible, the lower limit of the control chart is useless. Hence anomalies that affect lower transport requests cannot be detected, only upper anomaly breaches are detectable.

The prediction of expected transport tasks currently bases on the statistical analysis of the transport volume in the history. For an improvement of the prediction, a Digital Twin can be used, as it is sketched in the optimization architecture in Figure 3. It is planned to equip a digital twin with machine learning models to forecast factory runs, based on observed transport data. In [7], we already presented applicable methods and successfully utilized neural networks and tree-based prediction methods to predict the time between delivery and request of new transport jobs. Resulting from that work the XGBoost-method [18] is one of the most promising candidates.

The resulting prediction-improved Digital Twin can support and improve the anomaly detection precision, for example a more accurate mean value in the dynamic control charts or a prediction of anomalies that will happen in the future. Furthermore it supports the reconfiguration and optimization components with a more accurate comparison of possible mitigation, reconfiguration and optimization measures.

Acknowledgments

This work received funding from the BMBF through the ITEA CyberFactory#1 project under grant agreement No 01IS18061C.

References

[1] Abu Musa T and Bouras A 2022 *Anomaly Detection: A Survey* pp 391–401 ISBN 978-981-16-2101-7
[2] Moorthy R L, Hock-Guan W, Wing-Cheong N and Chung-Piaw T 2003 *International Journal of Production Economics* **83** 309–324
[3] Eschemann P, Borchers P, Feeken L, Stierand I, Zernickel J S and Neumann M 2020 *Modelling and Simulation 2020 - The European Simulation and Modelling Conference, ESM 2020* 208–215
[4] Lin S W, Miller B, Durand J, Joshi R, Didier P, Chigani A, Torenbeek R, Duggal D, Martin R, Bleakley G *et al.* 2015 *Industrial Internet Consortium (IIC), Tech. Rep*

[5] gGmbH B 2022 @online cyberfactory#1 URL https://www.cyberfactory-1.org
[6] Glawe M, Feeken L, Wudka B, Kao C Y, Weinhold T and Szanto A 2020
[7] Borchers P, Lisiecki D, Eschemann P, Feeken L, Hajnorouzi M and Stierand I 2021 *Modelling and Simulation 2021 - The European Simulation and Modelling Conference, ESM 2021* 49–55
[8] Koutras M, Bersimis S and Maravelakis P 2007 *Methodology and Computing in Applied Probability* **9** 207–224
[9] Pippin C and Christensen H 2012 *2012 15th International Conference on Information Fusion* pp 390–395
[10] Jaber A 2017 *International Journal of Electrical and Computer Engineering (IJECE)* **Vol 7**
[11] Jaber A and Bicker R 2016 *American Journal of Engineering and Applied Sciences* **9** 251–263
[12] Bollinger J 2002 *Bollinger on Bollinger Bands* (McGraw-Hill Education) ISBN 9780071373685 URL https://books.google.de/books?id=FLlxAz85iysC
[13] Ngan H Y T and Pang G 2006 *Optical Engineering* **45** 087202
[14] Hajej Z, Nyoungue A C, Abubakar A S and Mohamed Ali K 2021 *Applied Sciences* **11** ISSN 2076-3417 URL https://www.mdpi.com/2076-3417/11/9/4192
[15] Goodfellow I, Bengio Y and Courville A 2016 *Deep Learning* (MIT Press) http://www.deeplearningbook.org
[16] Liu J, Guo J, Orlik P, Shibata M, Nakahara D, Mii S and Takáč M 2018 *2018 13th World Congress on Intelligent Control and Automation (WCICA)* pp 175–180
[17] Liebl F 2018 *Simulation* (Oldenbourg Wissenschaftsverlag)
[18] Chen T and Guestrin C 2016 *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* URL http://dx.doi.org/10.1145/2939672.2939785