Definitions/applsci-logo-eps-converted-to.pdf

MDPI

Article

Detecting and Processing Anomalies in a Factory of the Future

Linda Feeken [1,*], Esther Kern [2], Alexander Szanto [2], Alexander Winnicki [3,*], Ching-Yu Kao [4], Björn Wudka [5], Matthias Glawe [3], Elham Mirzaei [5], Philipp Borchers [1] and Christian Burghardt [6]

1 German Aerospace Center (DLR), Institute of Systems Engineering for Future Mobility Concepts, 26121 Oldenburg, Germany; philipp.borchers@dlr.de (P.B.)
2 Brandenburg Institute for Society and Security (BIGS), 14482 Potsdam, Germany; esther.kern@bigs-potsdam.org (E.K.); alexander.szanto@bigs-potsdam.org (A.S.)
3 Airbus Operations GmbH, 21129 Hamburg, Germany;
4 Fraunhofer Institute for Applied and Integrated Security, 85748 Garching b. München, Germany; ching-yu.kao@aisec.fraunhofer.de (C.-Y.K.)
5 Department of Energy and Information, Hochschule Technik und Wirtschaft of Applied Sciences, 12459 Berlin, Germany; bjoern.wudka@htw-berlin.de (B.W.)6 *christian.burghardt@de.abb.com*
* Correspondence: linda.feeken@dlr.de (L.F.); alexander.winnicki@airbus.com (A.W.)

Abstract: Production systems are changing in many aspects on the way to a Factory of the Future, including the level of automation and communication between components. Besides all benefits, this evolution raises the amount, effect and type of anomalies and unforeseen behavior to a new level of complexity. Thus, new detection and mitigation concepts are required. Based on a use-case dealing with a distributed transportation system for production environments, this paper describes the different sources of possible anomalies with the same effect, anomaly detection methods and related mitigation techniques. Depending on the identified anomaly, the FoF should react accordingly, such as fleet or AGV reconfiguration, strong authentication and access control or a deletion of adversarial noises. In this paper, different types of mitigation actions are described that support the fleet in overcoming the effect of the anomaly or preventing them in the future. A concept to select the most appreciate mitigation method is presented, where the detection of the correct source of the anomaly is key. This paper shows how various techniques can work together to gain a holistic view on anomalies in the Factory of the Future for selecting the most appropriate mitigation technique.

Keywords: manufacturing; Factory of the Future; resilience; anomaly; detection; mitigation; cyber-attack; threat actors

Citation: Feeken, L.; Kern, E.; Szanto, A.; Winnicki, A.; Borchers, P.; Kao, C.-Y.; Wudka, B.; Glawe, M.; Mirzaei, E.; Burghardt, C. Detecting and Processing Anomalies in a Factory of the Future. *Appl. Sci.* **2022**, *1*, 0. https://doi.org/

Academic Editor: Firstname Lastname

Received: date
Accepted: date
Published: date

1. Introduction

The increasing level of digital networking of the factory landscape and the rising level of automation of its components and other key elements of Industry 4.0 technology offers high potential for increasing productivity, flexibility and sustainability and enables new business models in the Factory of the Future (FoF) [1]. At the same time, increasing connectivity and automation are creating new vulnerabilities in the FoF environment [2], e.g., by cyber-attacks targeting the emerging interfaces [3] and malfunctioning/secured automated systems. In this dynamic environment, resilient manufacturing systems need to be designed and enabled to recover from an undesired state (e.g., as a result of a cyber operation penetrating the infrastructure) and return to a desired state [4]. Hence, increasing the resilience of the FoF is of high importance for the manufacturing sector. To increase the resilience in the networked FoF environment, systems must be able to detect situations where they are in such an undesired state and initiate appropriate actions to return to a desired state. This leads to the requirement to develop methods to detect anomalies, reduce their frequency (avoiding undesired states) and mitigate the effects of anomalies (regaining desired states). One of the challenges in detecting anomalies in the FoF environment is dealing with the increased flexibility: since the FoF environment

is designed to operate very dynamically and is confronted with changing tasks, it is not feasible to derive standard system models whose properties can be compared with monitoring data to identify anomalies [5]. When dealing with cyber threats, existing legacy IoT systems pose a particular challenge as these systems were not originally designed for use in highly connected environments and are thus poorly secured [6,7]. The necessity and importance to mitigate cyber threats is widely emphasized: "Research and industry efforts should provide short-term solutions to mitigate the impact of vulnerabilities. For example, detection and correction of anomalies must be explored, rather than focusing on access control techniques only. As for automotive systems [...], prevention measures may interfere with the production chain and induce downtime: thus, a delicate balance between detection and prevention should be considered" [7]. Not only are security concerns motivating ongoing research, there is also a need for methods to increase resilience to "conventional" anomalies as stated in [8]: "Nevertheless, methods for intrinsic resilience with regard to internal disruptions, such as machine failure or unscheduled downtime, are still lacking". Furthermore, the survey in [9] underlines the need for integrated safety and security concepts in industrial control systems.

This paper contributes to this need by addressing the following aspects:

- Presentation of new methods for detecting anomalies caused by malicious or accidental events for distributed systems in dynamic factory environments.
- Presentation of new approaches to mitigate such anomalies.
- Integration of the presented methods to develop a holistic anomaly handling approach to increase the resilience of the FoF.

All method descriptions are illustrated using the example of a fleet of Automated Guided Vehicles (AGVs) as a key enabler for flexible material flow in Industry 4.0. This study seeks to add value not only to developers of AGVs but also to engineers and researchers in fields related to the presented methods, e.g., in the fields of decision making in distributed systems and anomaly detection with Digital Twins or in networks. The authors do not claim to present a complete approach for dealing with all kinds of anomalies in the FoF environment or in the use case of AGVs but aim to foster the integration of existing and newly developed approaches.

The paper is structured as follows: Section 2 introduces the characteristics of the FoF, emerging threats and related attacks to provide guidance for security-related method descriptions, followed by an overview of related work on anomaly detection and mitigation. Section 3 presents the use-case of fleets of Automated Guided Vehicles for factory logistics, which serves as a recurring example in the following chapters. Section 4 describes a variety of anomalies that can occur in the use-case, highlighting the necessity to complement specific methods to identify and mitigate such anomalies. Section 5 presents appropriate identification methods, followed by methods for the future prevention of identified anomalies and the mitigation of their effect. Section 6 discusses the combination of the presented methods to achieve holistic anomaly management. Eventually, limitations and open points of the presented work as well as future work are outlined.

2. Background and Related Work

The FoF includes highly connected systems in an environment characterized by data processing in real time across them and the associated supply chains. There is a high degree of networking of the manufacturing landscape and the integration of existing technologies and tools such as embedded systems, sensors and other industrial hardware within an ecosystem and with connection to outside systems. The generic term Industry 4.0, which was coined in Germany at the beginning of the 2010s [10], summarizes all these developments as a concept for the networking of the industrial landscape. Technologies, digital methods and processes in the context of a networked FoF, such as additive manufacturing, autonomous machines, collaborative robots, machine learning, augmented reality and big data analytics, offer a wealth of new possibilities for development, cost savings and monitoring as well as new business areas. However, these key elements of flexible automation,

connectivity and intelligence may come with serious vulnerabilities if cybersecurity is not adequately addressed. Therefore, it is necessary to include cybersecurity aspects from the beginning of innovation processes and developments in the industry environment. With regard to the FoF, while conventional anomalies will persist (such as technical failures), new ones will come in addition, i.e., cyber-attacks on operational technology (OT) connected to networks or Automated Guided Vehicles (AGVs), as in the use-case described below. This means additional attack vectors for threat actors (TA) as well as new incentives for TAs. While financial gains are one of the key motivations for cyber-attacks, networked FoF environments offer also novel opportunities for industry espionage.

2.1. Attack Vectors

Cybercrime is a booming business. The sector is becoming increasingly professionalized. This is also proven by the increased number of Cybercrime-as-a-Service (CCaaS) organizations. It has never been easier to run cyber-attacks, even for persons without much technical knowledge. The relevant services can be acquired in the darknet. This can range from (customized) ransomware, encryption and extortion software and the usage of botnets to Distributed-Denial-of-Service (DDoS) attacks. Clients can often also buy personal data for subsequent social engineering attacks. In this case, information about the targeted persons is exploited to make a message look as authentic as possible. The most frequent acquired CCaaS is the development and distribution of malware software of any kind [11]. Essentially, any service that companies can purchase from legitimate managed service providers can also be purchased by CCaaS groups, just for illegal activities.

Ransomware, also known as Ransomware-as-a-Service (RaaS), is probably the most popular subtype of CCaaS. However, even beyond as-a-service offerings, ransomware is booming. Ransomware is a type of malware that encrypts files in the targeted environment. The victims are then asked to pay a ransom for the decryption of the files. There is often now a risk of double extortion, meaning that the attackers not only encrypt the files, but before doing so, they transfer the files to an offsite location. Thereby, they can threaten to publish or leak data if the ransom is not paid [12].

SonicWall states in its 2022 cyber-threat report a volume of 623.3 million ransomware attacks in 2021. This is an increase of 105% from 2020 and 231% compared to 2019 [13]. In its 2021 data breach report, IBM reports average costs of 4.62 million USD. This amount does not include the ransom payments, but includes "only" costs for incident response, notification and loss of business [14].

Phishing is a widely used method to gain access to secured networks. The X-Force Threat Intelligence Index 2022 by IBM Security stated that in 41% of the observed infections with, for example, malware by the X-Force Incident Response in 2021, phishing was the method used to gain initial access [15]. Currently, phishing is the number one method for gaining access, taking over from the exploitation of vulnerabilities [13]. Phishing aims at sending deceptively real communication via email or text message. Most often, the goal is access to passwords or other credentials. Spear-phishing is a targeted phishing campaign against one organization.

A networked FoF means that previously isolated OT environments have more IT-connections within a factory, but also to the outside world. This is a possible vulnerability, in particular in the case of Cyber Supply Chain Attacks (CSCA). Supply chain attacks can be described as the exploitation of "third-party services and software to compromise a final target" [16].

One has to make a distinction between two types of CSCA: targeted attacks and **collateral damage** . In the case of targeted attacks, the connections of a third-party with access to a targeted network are deliberately used. These can be for example external suppliers of a type of software. Lower security measures between trusted parties are specifically exploited to infiltrate the environment of the final target. In the case of collateral damage attacks, the infrastructure of supply chains is exploited to spread malware and increase the attack surface [17].

Even in 2019, 60% of the surveyed organizations of a study conducted by Gartner stated that they worked together with more than 1000 suppliers. Furthermore, 71% declared that their number of suppliers had increased over the last three years [18]. It can be assumed that this number has increased further over the last years. This creates various interfaces and dependencies between organizations that can be exploited and already have been, as the compromise of the SolarWinds software in 2020 showed, as well as the cyber-attack on Kaseya in 2021.

2.2. Threat Actors

Cyber-criminals attempt to penetrate networks by exploiting any accessible vulnerability. In doing so, they primarily pursue two main goals: on the one hand, they want to gain assets (money or valuable data), and on the other hand, they seek to avoid legal consequences by disguising themselves and their activities. A significant part of the financial damage is not even direct, in particular not in industrial networks. Many scams result primarily in indirect costs that can also affect suppliers and other dependent clients, end users and customers and may not become evident until years later. Some of these costs are difficult to measure, as they represent intangible assets such as reputation, customer trust, intellectual property (IP), brand name, insurance premiums and several others. These hidden costs can quickly add up and cause significant damage that is far greater than the measurable direct impact of the actual event [19]. The most common activities of cyber-criminals can be divided into three categories:

- Mass fraud and automated hacking: Use of automated tools (with as little human effort as possible) to monetize large-scale fraud.
- Providers of criminal infrastructures: These actors try to infect as many systems as possible in order to exploit them in a criminal infrastructure (e.g., botnets). They may also sell/rent the exploitation of this infrastructure to third parties.
- Skilled professionals: Expend significant effort to attack individual high-value targets. This type of attack may use specially designed malware with a significant effort, or the attacks are carried out across supply chain partners. High-value targets in an organization are also targeted by email and phone scams, using social engineering skills to extend the attack [20].

Cyber-criminals pose the greatest threat to FoF. However, the range of cyber-crime is very wide, and the potential threats must therefore be considered on an individual basis.

State actors pursue a variety of goals, such as gathering information or supporting national interests (e.g., obtaining financial assets, stealing technological know-how or monitoring dissidents). They employ economic espionage to improve the capabilities of domestic companies. However, strategic sabotage is also a means that nation-state actors use to damage the economy or sabotage military infrastructure. In doing so, they do not make a strict distinction between civilian and military infrastructure. Infrastructures (botnets) are also created to conceal their own identity.

This cyber-operational capability can be established in two ways, which can also be combined. On the one hand, well-positioned intelligence services (financial resources and large numbers of experienced personnel) can carry out such strategic operations. On the other hand, criminal structures and organizations that are not directly attributable to the state can be used for this purpose. Those state-sponsored groups are usually easier to identify but can be more easily denied by the state [21]. These activities by state actors may well pose a threat to companies if they represent a strategic target, e.g., if the company is a world market leader in certain domain or produces unique products/components.

Disgruntled employees who are alienated for various reasons such as salary demands, promotion, appreciation, resignation, etc. and want to harm their company can pose an insider threat. Moreover, employees who are bribed and prove less loyal to their employer as the financial incentives are too lucrative can also pose a major threat. Blackmail can also occur and result in the company being harmed. While disgruntled insiders try to harm

their own company in different ways by retaliation, **mercenary insiders** are paid well for their services.

Preventing insider activity is challenging because individuals need access to trade secrets and systems to perform their duties. However, restrictions on access should thus be automatic with the leaving of the company, as access after dismissal can be used for retaliation. Moreover, depending on the positions they hold, former employees may be well versed in the infrastructure and can circumvent restrictions if necessary. The possible transfer of sensitive data before leaving the company should also be considered, depending on the area of work. Insiders are usually tracked by efficient logging and monitoring systems after their successful activities. Insiders can be abused by other threat actors (e.g., cyber-criminals), but in this case, they are considered threat vectors rather than threat actors.

Industrial espionage by competitors has always been a proven means of gaining access to trade secrets and blueprints in order to save and circumvent the often considerable investments that many companies make in the development of intellectual property. There are countless cases of industrial espionage by competitors that run in the grey area of legality and often beyond.

By the mid-2000s, Deloitte employed a whole team of accountants, veterans and intelligence officers to run undercover operations to gather as much information as possible about competitors and explore how to win prospective clients with insights about their rivals [22]. This unit may have played an important role in 2009, when Deloitte acquired a division from another large consulting firm that held many lucrative federal contracts that was in financial trouble at the time [23].

This example shows that economic competitors have long been a potential threat to companies, and not just since the technical possibilities enabled by the internet for spying and hiding. Nevertheless, cyber-operations open up entirely different possibilities for business intelligence and other active operations to capture trade secrets and other intelligence vital to the competitors. Marketing campaign information for products or the timing of product launches could be exploited to either gain an advantage by better targeting the market or to harm the competitor. Competitors can thereby follow two main approaches: either they rely on competitive intelligence, sometimes also referred to as business or corporate intelligence, which in the original sense means the collection and analysis of all publicly available information about, for example, customers and suppliers of competitors [24], or they rely on corporate sabotage and try, for example, to damage the reputation of an important competitor in various ways (cyber-attacks on IT and OT, data theft, damage to reputation through false claims in public). While competitive intelligence is in a grey area in some areas, corporate sabotage is already illegal. Such threats should not be underestimated by organizations, as competitors may be willing to run offensive operations in the digital fog due to the attribution problem. With good chances of success and a low risk of detection, some competitors may be willing to take a chance. Companies should not lightly ignore these threat actors.

2.3. Anomaly Detection and Mitigation

Anomalies refer to data in a data set that differ from the expected data [25]. An anomaly can either be a a small set of data points or even a single data point that differs significantly from the normally recorded data (point anomalies, e.g., the battery level of an automated guided vehicle (AGV) equals zero, which should never happen), a data instance that is only abnormal in a specific context (contextual anomalies, e.g., a peak in the number of transport tasks at a time in which normally only a low number of tasks is expected) or a set of related data instances, which only in combination differ from expected data (collective anomalies, e.g., quickly repeated messages of an AGV on joining and then again leaving a fleet of AGVs, each message on its own is not abnormal, only the sequence of messages is deviating from the expected data) [26]. Anomaly detection is the problem of identifying anomalies, such that the detected anomalies can be mitigated afterwards. Anomalies such

as unplanned production downtimes cost industrial manufacturers around $50 billion per year [27], and real-world examples for anomalies caused by cyber-attacks with financial losses for the victims are given in Sections 2.1 and 2.2. Hence, it is not surprising that the timely detection of anomalies is already the focus of a large number of research activities [26,28–30]. As already stated in the Introduction, the high dynamics and the high level of connectivity between production systems offer new potential for optimized production but lead at the same time to new challenges in anomaly detection, such as the problem of detecting anomalies in frequently changing factory situations and the detection of cyber-attacks in addition to anomalies in system and process behavior. Recent research trends tackling at least aspects of those challenges include the usage of Digital Twins (DT) for anomaly detection [31,32]. Additionally, artificial intelligence methods are developed to conduct anomaly classification tasks in manufacturing contexts, e.g., in the detection of anomalies in acoustics, such as deviations in the movement of robot arms [33], detection of anomalies in networks and communication [34] and the analysis of time series data [35]. Big Data analysis methods are also applied for anomaly detection [36], e.g., in [37] for quality control and in [38] for energy management. Although there are many methods available for the detection of specific anomalies, there are only few methodologies that address a wide range of anomalies, leading to holistic anomaly detection methods [39–41].

Big factory setups have various kinds of failures that can happen over a day. To withstand failures, new systems have to be robust on the one hand and resilient on the other [42]. Therefore, to increase the resilient behavior of systems and decrease the impact of failures, Gu [8] declared that redundancy and flexibility are major attributes to be built in. For this purpose, mitigation processes are widely used. In general, mitigation means to reduce or dissolve any effects of harm to a system and its environment. The mitigation of a system a can be performed by reconfiguring or adapting systems in response to a system or environmental change. However, reconfiguration is most likely used for system setups with a possible long duration of system downtime and a great diversity of system stages [8]. System stages can be described as the possibility to change system strategies (drive fast, equal wear and tear, etc.) or the replacement of systems by redundant systems. Currently, reconfiguration is split into two major ideas. (1) Global reconfiguration: A set of systems are connected and thus are represented as a fleet. Reconfiguration in this context is changing the behavior for all systems in the fleet to tackle factory goals as a group [43–46]. (2) Local reconfiguration: Each individual system can change its behavior to tackle individual system goals [47–49]. Even if the approaches differ, the reconfiguration process is performed in the same manner. The literature in general has concentrated on MAPE-K [50], where each letter represents one step to overcome system failure by e.g., reconfiguration. MAPE stands for monitoring, analyze, plan and execution; finally, K for knowledge represents data storage, which is used to set and compare monitored data, identify failures by analysis, determine a suitable preventive strategy by planning and implement a new strategy into the fleet or system setup.

3. Use-Case Introduction

In this paper, a fleet of Automated Guided Vehicles (AGVs) on a factory floor is considered as a running example. The factory consists of a set of machines and storage units. Each machine requests material for starting the production process and requires the produced goods and potential for products to be carried away. Those factory-internal logistic requests are collected by a manufacturing execution system (MES) and translated to transport tasks. The transport tasks are sent to a fleet of AGVs. The fleet shall realize goals such as quick transport job fulfillment, avoidance of machine stops due to a lack of material and equal wear and tear of all vehicles. The distribution of the tasks to individual vehicles has a strong impact on the goals' fulfillment. A common approach for task distribution is the usage of a centralized fleet management unit. The central control unit collects status data of each AGV, analyzes them and sends instructions such as the next tasks or operation details to each AGV individually. While a central control unit provides some benefits such

as straightforward development and saving of computational power for the members of the fleet, it also has some disadvantages, such as acting as a single point of failure and leading to the shut down of the whole fleet for system maintenance.

Instead of only following instructions given by a central control unit, AGVs need to coordinate their actions individually. Hence, decentralized fleet management is considered. For this purpose, each AGV is equipped with a communication module that enables the exchange of messages among AGVs via a publish–subscribe communication framework. The task distribution process is negotiated between fleet members through a bidding process. Every transport task is announced by the MES. Each AGV calculates its bid value based on a globally shared bidding algorithm and local parameters such as battery level, number of open tasks already taken over by the AGV and current location in the factory. The bidding procedure can be selected from a catalog of global strategies, varying, e.g., in the number of AGVs included in the negotiation process and the importance of the different goals of the fleet. Additionally, each AGV offers a set of local strategies, which determine details of the AGV behavior in addition to the tasks that it should fulfill. Local strategies in contradiction to global strategy are used to optimize the behavior of each AGV individually. Strategies are realized by adapting and modifying configurations on each individual AGV. To realize that, AGVs are constructed in a Service-Oriented Architecture, which facilitates the exchange system and service information between fleet participants through an orchestration service called SystemDiscovery [51]. SystemDiscovery stores information about local services alongside the information about services, which can also be provided by other fleet members. Those services are used to adapt and generate configurations. Connecting interfaces of different services from various systems for local reconfiguration is called Service-Oriented Reconfiguration (SoR) [52]. To adapt a system, SoR is in charge of blueprints that describe the connection abilities between potential services. Thus, a configuration can be created and optimized at run-time without the integration of hardware or software modules. The possibility for changes in the negotiation process and in the local configuration of AGVs allows the fleet to react dynamically to situations in the factory.

4. Anomalies and Sources

The performance of a fleet of AGVs on a factory floor is strongly linked to the behavior of other components of the factory, such as machines that request transport tasks. Hence, anomalies in the behavior of the fleet can not only have fleet-internal sources but can also arise from external sources. This leads to a great variety of anomalies and their sources in the use-case. Table 1 gives examples of anomalies that are relevant for the fleet of AGVs. The sources are categorized into three types: defects in the hardware or software of the AGVs or external components, changes in processes and cyber-attacks, as introduced in Section 2.1. Additionally, the table shows indicators that support the identification of anomalies.

Different anomalies can cause the same effect but can result from different sources. In general, it is not sufficient to just focus on mitigating the effect of an anomaly without searching for the source of it. In this section, different sources for anomalies with the same effect are sketched. In a fleet of AGVs on a factory floor, it is noticed that one of the AGVs is not acting as expected but is only moving slowly or comes to a complete stop. Looking into the monitoring data of this AGV shows that it is still communicating with the fleet, as it is still operational. In particular, the AGV still participates in bidding processes for new transport jobs. All of the other AGVs are acting normally. As a consequence, the malfunctioning AGV is collecting pending transport jobs in its local queue over the time without being able to fulfill them. In the worst case, this forces the production to stop, since machines can run out of material. Other negative consequences of the anomaly depend on its source, e.g., a corrupted AGV could also send inadequate bids to other AGVs. In the following, three different sources for the effect of the non-moving AGV are presented.

Table 1. Overview of possible anomalies classified by anomaly indicators and sources.

		Anomaly Source		
		Defects	**Process Changes**	**Cyber-Attack**
Anomaly Indicator	**Process indicator**	No transport jobs from broken machine	High number of transport jobs from suddenly high active machine	Malicious or dangerous manipulated process data
	Component indicator	Lower driven distance of AGV with broken self-localization module	Length of local queue of each robot longer than usual due to unexpected transport request peak	Deviation of AGV-internal software list
	Communication indicator	No messages received from AGV with broken WIFI module	High number of messages from MES received	Wrong agent (e.g., AGV) publishes transport job

4.1. Unrecognized AGV Failure

As a first example for the source of an anomaly that can cause an AGV to stop, a set of well-known anomalies that could already appear in traditional factories without a direct link to cyber-attacks is considered: the malfunctioning AGV is suffering from an unrecognized failure of its hardware or software. The specifics of such an anomaly can vary, e.g., the motor of the AGV might be broken, the sensors for self-localization on the factory floor might be blocked or defective, the locally stored data on the factory map might be lost or the AGV might repeatedly try and fail to exchange goods with a machine. Since such anomalies are already well understood and mitigation methods include adding redundant sensors, integration of additional degradation modes in case a component or service is unavailable or adding timeouts for the process of exchanging goods, the detailed identification of the source of the system failure is out of the scope for this paper. Instead, the focus is on the general identification and mitigation of situations in which the AGV is not moving while its controller does not react to the system failure, such that the AGV still participates in the fleet communication as if it were operational.

4.2. AGV under Cyber-Attack

In this section, a cyber-attack is considered as a possible anomaly source that results in an immobilized AGV that is still able to communicate. For the sake of simplicity, the following assumptions are made: first, the attacker has already obtained network access to the AGV fleet, e.g., by a phishing attack (see Section 2.1). Second, prior knowledge of a vulnerability in the firmware of the targeted AGVs allowed the attacker to prepare a weaponized payload that compromises AGVs when deployed, handing over control to the attacker. Both these assumptions are realistic in an industrial context, especially due to the accessible fleet design that allows dynamic addition or removal of AGVs. These assumptions make it possible to focus directly on the relevant anomaly as well as its detection and mitigation. When trying to understand the source of the anomaly in the context of a cyber-attack, it is also relevant to discuss potential threat actors and their motivations. Due to the complex and proprietary nature of industrial systems as well as their significance with regard to critical infrastructure and national security, the majority of active threat actors in this environment constitute nation-states or state-sponsored groups [53]. As opposed to ordinary cyber-criminals who primarily seek monetary profit, nation-state interests comprise espionage, sabotage and the acquisition of offensive capabilities against critical infrastructure (see Section 2.2 for more details). In this context, the above-mentioned cyber-attack against an individual AGV in the fleet is likely to constitute only a minor part of a larger attack. For instance, immobilizing an AGV but keeping its communication capacities operational could allow attackers to learn about anomaly detection and incident

response systems and processes that are in place to protect the fleet. This knowledge could then be used for further attacks against more significant targets.

4.3. Adversarial Attack

Whether in reality or in theory, AGVs have sensors to sense different information. For example, camera sensors can sense pictures, and some sensors can learn location or voice information. These sensors receive real-world information, such as obstacles. The device should stop moving or bypass obstacles when it encounters an object. When the device encounters a stop sign, it should stop immediately and adjust its route. At this time, a safety problem arises. If the image received by the sensor has been intentionally modified, this modification would be invisible to humans, but it would lead the AGV to make an incorrect decision—for example, speeding up when it is time to stop. As another example, if someone deliberately adds some insignificant noise to the obstacle, this could cause the AGV to directly hit the obstacle. These are the safety concerns that should be considered in the industrial environment. This kind of noise is called adversarial noise and cannot be detected by humans but will change the noise predicted by the AGV. Data with this kind of noise are called adversarial examples. This is expressed more formally below.

Definition: Adversarial Examples Let f be a trained neural network used for classification tasks. Let H be a human oracle with the same classification capabilities. Assume that for a given legitimate input x, the following equation holds: $f(x) = H(x)$ Let x' be a modified version of x that is close to x, i.e., $||x' - x|| \leq \epsilon$, for all small ϵ, $\epsilon \in \mathbb{R}$. Then, x is an adversarial example, if the following holds:

$$H(x) = H(x') \bigwedge f(x) \neq f(x'). \tag{1}$$

In the real world, the way to implement adversarial attacks is to use adversarial noise, such as an adversarial sticker, attached to one or more AGVs. When the AGV's sensor transmits these artificial noises to the decision-making system, it will make incorrect decisions. These prediction errors can produce runaway AGVs or disabled AGVs, which in turn lead to many safety and security problems. Hence, this adversarial attack is considered a kind of anomaly. In this context, the following two possible scenarios are introduced: the first one is that there is no obstacle ahead, but the AGV thinks one or several obstacles are present because of an adversarial attack. The AGV stops and does not move (forward) for a long time. In this situation, the AGV should communicate with other AGVs to find or update a new feasible path to complete the task.

The second case is the opposite: an obstacle is in front of the AGV, but the attacker makes the AGV think that there is no obstacle (see Figure 1). This will cause the AGV to go forward and hit the object directly (with high speed). At this moment, the AGV may be damaged due to this unexpected collision. The damage means that one or more components of the AGV stop working. If the motor is damaged or the wheels are blocked as consequence of the collision, the adversarial attack can lead to the anomaly described in Section 4.1.

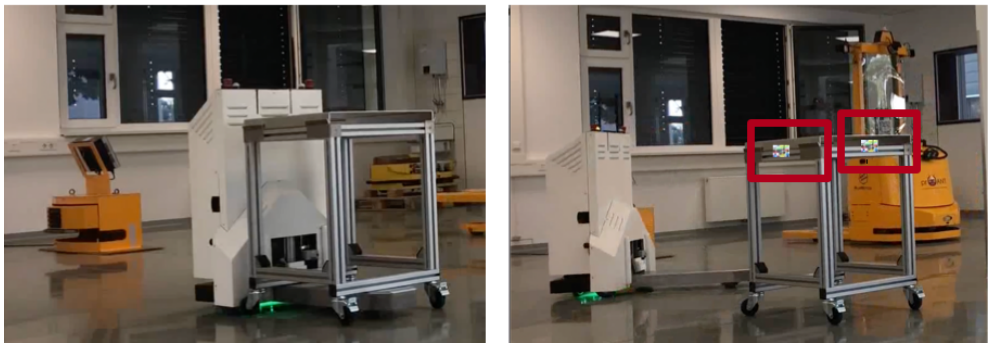

Figure 1. An example of an adversarial attack. On the left, the normal situation is shown. On the right figure, one object has adversarial patch(es), which results in the anomaly. More specifically, the AGV cannot detect the object correctly.

5. Methodology for Anomaly Detection and Mitigation

In order to enable the FoF to be resilient with respect to undesired behavior, such as a barely moving AGV, appropriate reactions as mitigation are required. As illustrated above, the origins for the same undesired FoF behavior can wildly differ. Since the adequacy of a mitigation can depend on the source of the observed anomaly, there is not only the need for mitigation methods for increasing the resilience of the FoF but also for anomaly detection methods that are capable to differentiate between the origin of undesired FoF behavior. In the following, anomaly detection methods are presented (Section 5.1), followed by a set of mitigation methods (Section 5.2). The integration of the presented methods is discussed in Section 6. As a running example, the undesired AGV behavior sketched in Section 4 is used in all method descriptions.

5.1. Anomaly Detection

The anomalies shown before differ in their source, characteristics and area to which they belong. Indicators to identify the anomaly can be assumed either in the data-lake as part of the communication or directly on a specific robot or attached system. As the parameters of the anomaly can be assumed to be very different depending on the source and area of the anomaly, and also the necessary parameters to solve the anomaly depend on the goal of mitigation (e.g., optimize operation or mitigate a cyber-attack), different anomaly detection methods are needed. This chapter presents anomaly detection methods developed as part of the use-case to address the aforementioned anomalies.

5.1.1. Unrecognized Failure in AGV

To detect anomalies such as a non-moving AGV that is still communicating as a normal working device (cf. Section 4.1), fleet behavior logs are used, received via the data lake or directly from AGVs. The following KPIs from these logs are of interest to detect a non-moving AGV:

Finalized Tasks per Hour

If an AGV is not moving, it does not execute transport jobs as usual. Hence, the number of finalized transport tasks will be lower than before, and a stopped AGV can be detected by comparing the actual number of finalized tasks with the expected number. The number of finalized transport tasks can be observed on a global as well as an AGV-individual level.

Driven Distances of Each AGV

If the driven distances of each AGV are observed at every time, less driven distances of the stopped AGV will be detected compared to the the other AGVs. If the task bidding strategy is configured such that the driven distances are equalized over all AGVs, the stopped AGV with this KPI value can be detected. Furthermore, it is possible to compare the driven distances to the expected amount.

Length of Local Queue of Each AGV

As no tasks for the stopped AGV will be finalized, the length of the local queue of open transport tasks will increase over and over again and become much larger than queues of other AGVs. If the task bidding strategy is configured such that the local queue is equalized over all AGVs, the stopped AGV with this KPI value can be detected. Furthermore, it is possible to compare the length of the local queues to the usual or expected length.

Battery Status

When the AGV is not driving any more but is still active, the battery discharging will be slower than usual. Hence, the observation of the discharging velocity is another indicator of an unusually moving AGV.

To indicate an anomaly, it has to be known for each AGV which of those KPIs is unexpected. Hence, the expected KPI values are needed at each time to compare them to the actual ones. To obtain the expected KPIs, a Digital Twin of the AGVs can be used to simulate their behavior in a production environment. The anomaly detection itself can be realized with detection methods such as an adapted version of classic control charts [54] that includes the expected KPIs over time as a mean value [55]. Quality control charts have the advantage that they allow fluctuations of the KPIs around the expected value. Only if the deviation of expected and actual KPIs are very excessive and warning limits are exceeded is an anomaly triggered.

5.1.2. AGV under Cyber-Attack

The Cyber Kill Chain [56] provides a comprehensive model of the general phases of a cyber-attack. Monitoring for actions and indicators of each phase provides the opportunity to detect and disrupt ongoing attacks at different stages—an approach also known as defense-in-depth. This section discusses possible detection capabilities at each individual stage. As previously assumed, the attacker has already gone through the *Reconnaissance* and *Weaponization* phases, i.e., they have access to the AGVs and possess a ready-to-deploy exploit. Detection thus revolves around the subsequent phases of *Delivery*, *Exploitation*, *Installation* and *Command and Control*. Network traffic to and from AGVs can be minored to detect communication patterns outside the defined baseline, indicating potential *Delivery* attempts. As normal AGV behavior is highly predictable, the corresponding network traffic constitutes a useful detection source with low false-positive rates. As most cyber-attacks involve privilege escalation as part of the *Exploitation* phase, the Operating System (OS) of AGVs can be monitored for creations and modifications of users and corresponding rights. Under normal operating conditions, there would be close to no modifications of accounts and privileges; therefore, any modification is worth inspecting. Continuous validation of the software baseline of AGVs enables the detection of the unauthorized *Installation* of attack tools and malware an attacker would need for lateral movement. AGVs have clearly defined tasks, and any installation of additional software should thus be investigated. Furthermore, the firmware update mechanism of AGVs is a direct entry point for attackers and provides a significant opportunity for compromise if unsigned over-the-air (OTA) firmware updates are accepted. In this case, firmware updates would need to be monitored so that each update attempt is detected and validated. In addition, equipping AGVs with hardware switches that would require flipping to enable firmware updates would limit this attack path, provided the switches are returned to their default position after legitimate updates. Attackers may further attempt to introduce rogue devices pre-equipped with malware or malicious firmware or configurations. However, simply detecting the connection of new AGVs to the network is not directly indicative of an attack, as dynamic fleet constellation is an intended design feature and not an anomaly. Therefore, it is necessary to also validate newly introduced AGVs, either by verifying their software and configuration baseline using hashes or by equipping legitimate AGVs with certificate-based digital identities prior to deployment and verifying those. Finally, monitoring the physical state of AGVs, e.g., location and task assignment, makes it possible to notice

deviating AGV behavior that could be indicative of foreign *Command and Control*. In the case of the anomaly where an AGV cannot move but still communicates, this is the attack stage that has already been reached if a cyber-attack constitutes the source of the anomaly.

5.1.3. Anomaly Information Exchange

The earlier sub-chapters present different anomaly detection methods and how they can detect anomalies that might look similar in their effect based on different inputs. All of these methods use their own data sources, tools, algorithms and protocols to analyze an anomaly. To gain a holistic view of an anomaly and to be able to understand the real source of an anomaly leading to the proper mitigation, an exchange of anomaly information needs to be enabled. General requirements for such an anomaly information exchange are the capability to store all relevant anomaly information by all connected detection capabilities. In addition, the anomaly information must be readable for all available mitigation methods in order to have all relevant information available to define proper mitigation methods.

In the anomaly detection approach, the available robot fleet data base is used, which was designed originally to store information on robot fleet performance, such as performed jobs, driven distances, etc. To address the need for anomaly representation, two new categories were implemented in the database. The first category, anomaly information, provides general information on possible anomalies without having a real occurrence of the respective anomaly. The second category represents an anomaly occurrence, referencing an anomaly and adding the case-related information. The included information in both categories is shown in Figure 2.

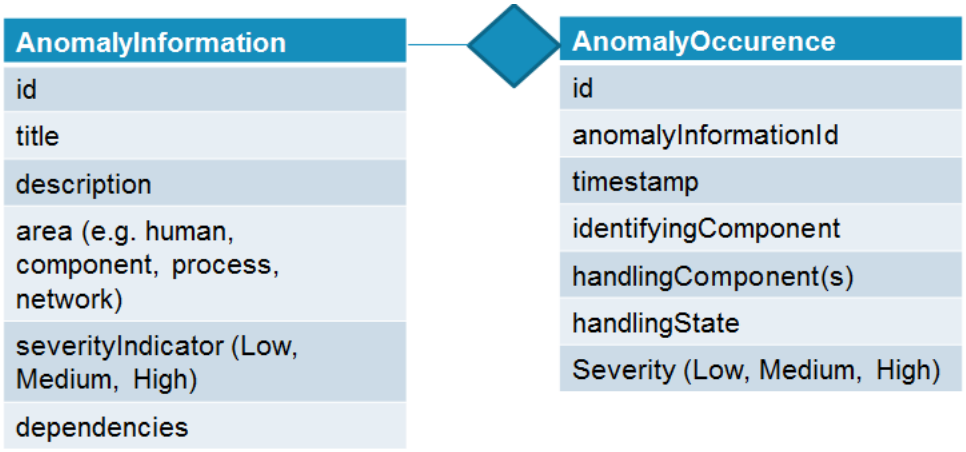

Figure 2. Anomaly classes.

5.2. Mitigation

Once the anomaly is detected within the FoF, following the defined process in the previous section, a proper mitigation action should be initiated to either remove the malfunction behavior or reduce its negative impact to the minimum. In general, depending on the identified anomaly, the FoF behavior should react accordingly. Therefore, different types of mitigation actions are required to address different types of anomalies within the FoF. The so-called adaption can for instance be planned and deployed through reconfiguration, either locally at the level of each AGV or if necessary globally at the level of the fleet. Furthermore, from the security perspective, all the deployed mitigation actions would be worth nothing if AGVs have no solid and trusted computing base. Thus, avoiding malicious instructions that could lead to an undesired or dangerous AGV state and mitigation against adversarial and cybersecurity attacks is essential to ensure AGVs' trustworthiness and improve their security. Within this section, different types of mitigations addressing the above-mentioned anomalies will be described.

5.2.1. Reconfiguration of AGV Services

Within the overall mitigation process, service oriented reconfiguration (SoR) facilitates adaption within a single system. Mitigation can therefore be triggered through service failures, cyber attacks or an adversarial attack. For the purpose of simplicity ,only two AGVs (AGV1 and AGV2) are assumed to be operating together in the factory. Figure 3 illustrates the service setup in normal operation of AGV1. We assume that one specific service of the robot-mapping configuration of AGV1, which consists of the three services Laser scanner, Adaptive Monte Carlo landmark (AMCL) and Local-cost-map, is under attack and thus is misbehaving.

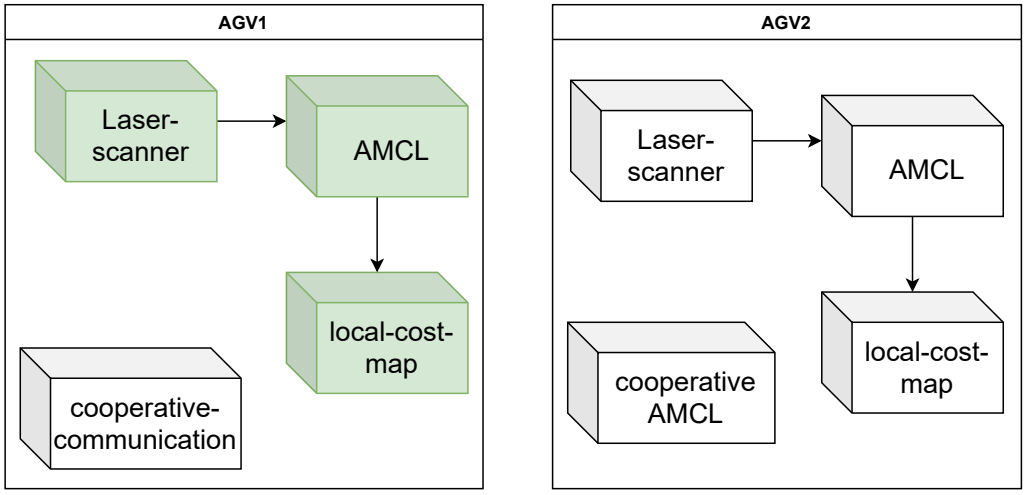

Figure 3. Service and communication setup in normal operation of AGV1 and AGV2 (green: services used by AGV1).

In this example, it is assumed that the AMCL-service is not available due to an adversarial attack. Thus, AGV1 is not able to navigate in the factory layout. AMCL uses the data from laser scanner as an input to calculate the location of the AGV and its surrounding objects. AMCL data are then used to the allocate the AGV and the identified objects in a preset local-cost-map, which is used as a basis for the navigation of the AGV. By identifying the misbehavior of AGV1, a trigger with an identification of the misbehaving service is deployed to the reconfiguration. By setting a trigger, reconfiguration is used to find a replacement for the AMCL service and thus restructure the current configuration. To create new configurations, SoR uses data from a service catalog that indicates all available services within a SoS and blueprints. Blueprints are templates of different configurations that are constructed by services. A configuration can be deployed by using different types of services that handle general functionalities such as sensing, acting or computing. To continue the interrupted navigation process, the AMCL service must be replaced. Therefore, a cooperative AMCL service running on AGV2 is used instead of the local AMCL (attacked on AGV1). In order to use the cooperative AMCL service, a cooperative communication across system boundaries has to be started. The cooperative communication service enables the communication of the AGV1 laser scanner and AGV2 cooperative AMCL service. With successful verification, the planned configuration can be deployed to AGV1 and AGV2. The new created configuration now consists of the following:

- Laser scanner running locally on AGV1,
- A cooperative communication-service that sends and receives all necessary messages,
- An AMCL-service provided by AGV2 and
- Local-cost map running on AGV1.

Figure 4 illustrates the configuration and the service communication after reconfiguration in the case of the adversarial attack on AMCL.

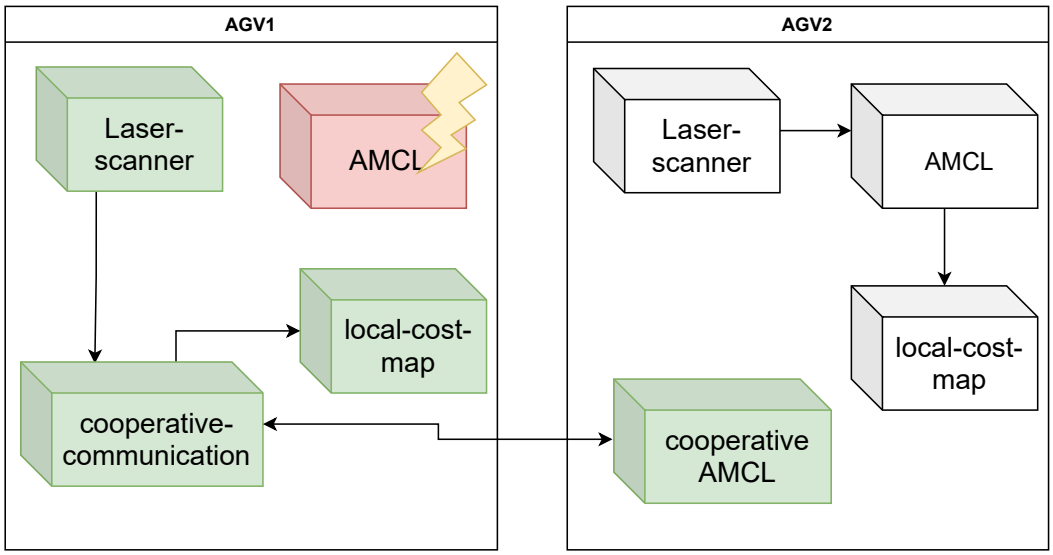

Figure 4. Service and communication setup after local reconfiguration is conducted (red: attacked service, green: services used by AGV1).

Local reconfiguration enables recovery behavior upon failure in services either due to hardware or software failures as well as cyber-attacks to reduce the impacts on the factory operations. However, it should be noted that service-oriented reconfiguration is only applicable when the failure is associated with the services. Thus, in case there is a degradation of the system that cannot be mitigated by the reconfiguration of services, global reconfiguration must be deployed as a complementary mitigation process.

5.2.2. Global Reconfiguration

The objective of the global reconfiguration method is to find a sequence of configurations for each AGV of the fleet such that the overall fleet behavior is optimized. In contrast to the reconfiguration of AGV services (Section 5.2.1), all AGVs of the fleet are in general involved in a reconfiguration with this method. In this use-case, each (global) configuration is encoded as one bidding procedure that determines how the fleet distributes the transport tasks and when AGVs charge their batteries. This method can be used to support the mitigation of effects of anomalies, as in Sections 4.1 and 4.2, or as a fallback mitigation method if the reconfiguration of AGV services (Section 5.2.1) does not lead to a solution.

The tackled reconfiguration problem in this use-case is related to Multi-Robot Task Assignment (MRTA) problems. In MPTA problems, the challenge is to decide which robot of a group of robots should take over which task in order to achieve the overall objectives of the group. The specific problems vary, among others, in the complexity of tasks, assumptions on knowledge on tasks, dependencies between robot schedules and capabilities of robots [57,58]. However, MRTA solutions cannot be applied directly in this use-case, since here, the possible configurations of AGVs are already fixed and implemented and cannot be changed. The same holds for purely game-theoretical approaches, in which additionally extensive formal models are required that are not available in this use-case and would become too complex to be computationally feasible to handle. Nevertheless, game-theoretical ideas on decision making are taken over in this reconfiguration approach. For this, formal complex models are replaced by predictions and the usage of Digital Twins.

For the rest of this section, it is assumed that it is not possible to make the misbehaving AGV operational again by a reconfiguration of AGV services (Section 5.2.1). This is the case when additional resources are needed, e.g., maintenance or repair staff when there was a AGV failure, as in Section 4.1, or a human worker that can clear the floor around the AGV if it is stuck between real obstacles, as in Section 4.3. The goal of the reconfiguration process is then to find a configuration that enables the remainder of the fleet to handle the predicted transport tasks without requiring human intervention at once.

The detection of an anomaly triggers the four-step global reconfiguration: as a first step, a prediction of future transport task emergence is made. Data such as timestamps, start locations and destination locations of previous transport tasks are collected and serve as training data for learning methods for predictions of future tasks. A comparison of different learning methods applicable for this use-case can be found in [59]. The resulting list of transport tasks is further analyzed in the second step for the pre-selection of possible reconfiguration candidates. This step is performed to reduce the amount of required computational power in the next step. Based on the predicted transport task emergence and the current AGV performance, it is checked which of the implemented configurations are applicable in the current situation. In the running example, anomaly detection points to one AGV as the origin of unexpected evolving KPIs. Hence, only configurations that exclude this AGV from the fleet are considered as reconfiguration candidates. Additionally, the pre-selection process can make use of the results of previous reconfiguration recommendations. If the current situation is similar to a previous situation, the previously recommended reconfiguration should be taken into account as the next reconfiguration. The set of previous solutions could be completed with other possible randomly chosen configurations to avoid the problem of repetitively neglecting promising reconfiguration alternatives. Each of the collected reconfiguration candidates can itself consist of a sequence of reconfigurations. For example, if a configuration for the next three hours is requested, a reconfiguration candidate could consist of one configuration for each hour, hence being a sequence of three configurations. In the third step, the prediction of transport task emergence (step 1) is combined with a Digital Twin of the fleet. The Digital Twin is used for calculating predictions of the fleet performance for each of the chosen reconfiguration candidates (step 2). As starting point, the Digital Twin uses the current AGV situation, including the current AGV status and detected anomalies. The result of this step is a set of fleet key performance indicator (KPI) evolutions as predictions of the effect of different configurations. In the fourth step, those predictions are used in a decision making process. The purpose of the decision making is to pick one of the reconfiguration candidates as a result of the whole reconfiguration process. Due to the multiple objectives of the fleet and potential conflicts between them (e.g., fast task fulfillment can conflict with low driven distances for avoidance of wear), deciding which fleet performance is better than another is already a challenge. Additionally, the evolution of performance indicators has to be taken into account; hence, a decision cannot simply be made on a single snapshot of the indicator valuations. Decision criteria taken from the theory of multi-objective optimization problems (MOOP, [60]) can be used to decide which fleet behavior is best for a single point in time. A modification of well-known MOOP solutions for dynamic system behavior can be applied here to choose one of the performance predictions (step 3) and hence one of the configurations. In the running example, it is assumed to be desired to have one recommended configuration for each of the next three hours after identifying the anomaly and to choose one of the simplest MOOP solutions as a decision criterion: the weighted sum method, in which a weighted sum of all KPIs should be minimized. To be able to apply this criterion, the fleet performance predictions are processed, such that for each pair of prediction and KPI, one value representing the KPI valuation in this prediction is calculated. The weighted sum is then determined using all values of the same prediction. The sequence of configurations that corresponds to the prediction with the lowest weighted sum is then chosen as the recommended reconfiguration. The reconfiguration process is repeated at the latest two hours after the last one, i.e., before the last element of the latest proposed sequence of configurations is applied. Thereby, the risk of running in a local fleet optimum that might be followed by highly undesired fleet performance is reduced.

5.2.3. AGV under Cyber-Attack

This section discusses a set of high-level mitigations that would either directly address or help to prevent each of the previously described cyber-attack phases. The mtigation of *Delivery* attempts to an AGV can be performed with strict enforcement of defined network

traffic using a firewall, where any pattern outside a baseline is automatically blocked. However, if the attacker uses legitimate channels to deliver a malicious payload, e.g., over the air firmware updates, this could not be detected or mitigated based solely on packet filtering. Instead, the firewall would need to have application layer capabilities to inspect and validate all data going through such legitimate channels. In addition, proper input validation within the software of an AGV would block code injection attempts towards any exposed service, including unsigned firmware delivery. With regard to *Exploitation* , least-privilege should be enforced within the software and OS of AGVs, hard-coded user accounts should be disabled, and all default passwords must be changed. This would directly prevent the exploitation of user accounts for malicious purposes and contain the damage in case they are compromised. Application whitelisting would further help to contain damage by preventing the unauthorized *Installation* and execution of software. However, the mitigation of malicious *Command and Control* can prove difficult once the attacker has established a foothold in the AGV system. As a last defense, the control software of an AGV should be protected by strong authentication and access control. Depending on its sophistication, this software could further detect malicious instructions that could lead to an undesired or dangerous AGV state and block their execution. In summary, both the detection and mitigation possibilities outlined above can be automated to a certain degree, depending on the complexity of the actual systems as well as the availability of capable tools. Automation would further allow detection capacities to be paired with adequate response actions, enabling the prevention of AGVs being compromised in the first place. However, for an effective defense-in-depth approach, the AGV firmware is of critical importance and must constitute a solid and trusted computing base. Otherwise, all mitigations built on top of it would be severely affected in the case of being compromised.

5.2.4. Mitigation against Adversarial Attack

Preventing adversarial attacks is a problem that must be solved urgently. There are many studies addressing adversarial attacks, but almost none of them can be applied to the factory setting. The reason is that the factory environment is too complex with too many variables, making traditional defense methods almost impossible to achieve, such as adversarial learning. In addition, although there is a large amount of data in the factory, there are very few clearly labeled data, which greatly increases the difficulty of the used training data sets and makes most supervised-based defense methods unsuitable; thus, an unsupervised or semi-supervised method has to be used. In this project, effective and useful defense methods for FoF are provided. It is assumed that the FoF uses artificial intelligence methods or will use artificial intelligence methods, that there are real-time monitoring systems in the factory, and that the AGVs have sensitive vision sensors. First, the suggested method is used by restarting the AGV and looking for other suitable systems, such as charging stations. This is the first step of defense: try to initialize the parameters of the AGV again. This method is usually very effective for anomaly detection, and this method is very simple, so it is recommended as the first step of defense or the pre-step of the adversarial defense. Next, it is intended to offer two more specific defenses.

Ensemble Learning

Usually, an artificial intelligence model is used to help in decision making. Adversarial attacks can attack this single model, generating adversarial samples. However, ensemble learning uses two or more models. Using multiple different models is called ensemble learning. In practice, ensemble learning gives better accuracy and stronger robustness. Predicting the results will make the attack difficult, and the attack cost will be greatly increased. Furthermore, ensemble learning can not only increase the difficulty of the attack but also improve the accuracy of the model prediction, which is a win–win solution. Although this method increases the difficulty of the attack and reduces the probability of being attacked, it does not mean that the system will not be attacked at all. If the attacker

has enough time and rich resources, the attack can still be successful, so in the next section, a second method is provided.

Remove Adversarial Noise

In the real world, adversarial samples are usually added by adding noise stickers/patches to the target object. The studied method in the context of this work requires only training samples without labels and training an up-stream model, which can reconstruct the image of the adversarial parts. This method is divided into two steps: the first step is training a up-stream model that can reconstruct the image with patches, while the second step is to remove the adversarial parts by setting pixels all to zero (called hollowed out part) and using the previous up-stream model to reconstruct the hollowed out part. The attacked image is therefore modified to the correct image, which allows devices using artificial intelligence to predict it correctly again. The concept is shown in Figure 5. The method is divided into two parts. In the first part, an up-steam model needs to be trained using collected AGV images. In the second part, adversarial patches are hollowed and fed into the previously constructed up-steam model. In the end, a clear image is generated.

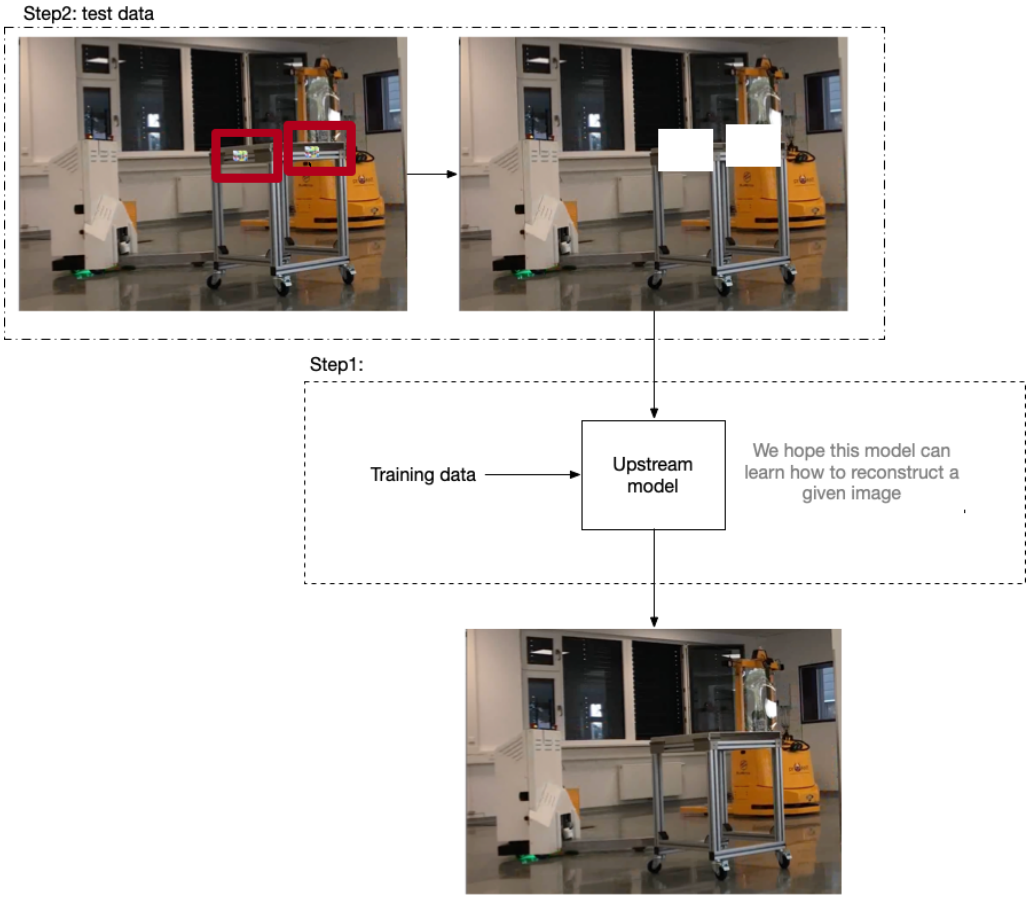

Figure 5. This figure represents the reconstruction of the image with patches.

6. Discussion

6.1. Integration of the Methods

This paper illustrates that anomalies can be caused by multiple sources. If an AGV simply stops moving, the behavior can be based on hardware or software failures in the AGV, obstacles on the factory floor, cyber-attacks or adversarial attacks. In order to address the anomaly, it is necessary to identify the source. Several methods to identify the source of the anomaly (Section 5.1) were presented, which jointly can provide a holistic view of anomaly detection. Depending on the source of the anomaly, different mitigation

methods can support the fleet in overcoming the effects of the anomaly or preventing similar anomalies in the future. Figure 6 summarizes the anomaly detection and mitigation methods described in this paper. During the operation of the provided use-cases, the fleet of AGVs and the components of the factory, such as machines and MES, produce data (e.g., communication data and monitoring data), which are collected in a central data warehouse or data lake. The various anomaly detection methods request the type of data that is relevant for the specific method. For example, for the identification of a cyber-attack (Section 5.1.2), the network communication data is analyued, whereas the process anomaly identification method (Section 5.1.1) is fed with data on transport tasks and AGV status monitoring data. If one of the anomaly detection methods identifies an anomaly, an alarm and information on the detected anomaly are sent to the data lake. This triggers the various presented mitigation methods.

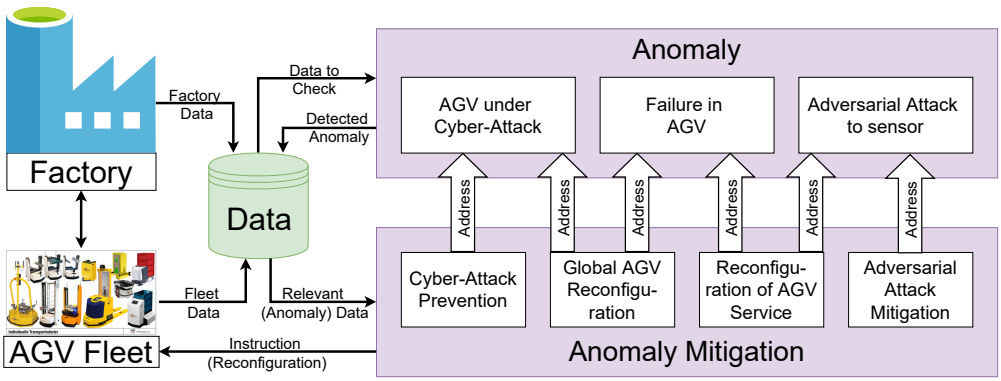

Figure 6. Overview for detection and mitigation of different anomalies.

Depending on the anomaly at hand, one or more mitigation methods start to calculate mitigation measures. For example, an identified cyber-attack or adversarial attack can trigger the reconfiguration of AGV services (Section 5.2.1), the global reconfiguration method (Section 5.2.2) and the cyber-attack mitigation method (Section 5.2.3), leading in combination to a proposal for how to make the factory less vulnerable to this kind of attack (e.g., using additional firewalls) and how to mitigate the effect of the attack immediately. If it is possible to identify a new service configuration that enables the attacked AGV to become operational again, this configuration is deployed in the AGV. If this is not possible, the global reconfiguration serves as a fallback solution, e.g., by removing the attacked AGV from the fleet and reconfiguring the remaining fleet to cope with the new situation.

Figure 7 illustrates the anomaly detection and mitigation approach for the use-case with the non-moving robot in more detail. The considered causes for the interrupted robot (broken robot, cyber-attack, adversarial noise) have different effects: if the AGV is the target of a cyber-attack, the network traffic would be different from usual; an adversarial attack will affect the availability and outcome of services. In the case of a defective drive engine, only certain KPIs such as distances covered will be affected. Hence, the AGV's KPIs, the services and the traffic data are the kinds of data that must be monitored by the anomaly detection methods from Section 5.1. Depending on the source of the anomaly, different detection methods will raise an anomaly alert, as each technique has dedicated data parts to monitor. A mitigation decision plan is used to select one of the possible mitigation methods (cf. Figure 6), where each mitigation candidate is checked sequentially for application. Assuming that, for instance, a driving engine of one AGV is broken and the *failure in AGV* detection is triggered, then, the *Global AGV Reconfiguration* and the *Reconfiguration of AGV Service* are possible mitigation candidates. In this case, the *service oriented reconfiguration* will fail, because another AGV could not replace the service provided by the broken driving engine. As there is no adversarial attack, the *Adverarial Attack Mitigation* is not applicable. Finally, the defective AGV will be excluded from the transport service via a global reconfiguration by excluding it from the task distribution process. In the last step of

the anomaly detection and mitigation approach, the resulting mitigation method will be sent to the AGV fleet for execution.

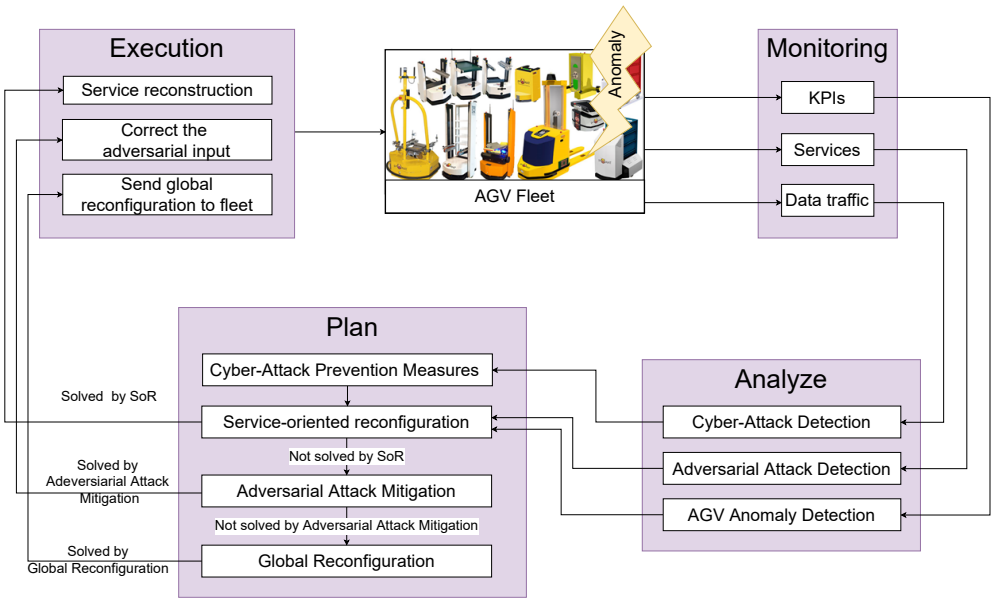

Figure 7. Anomaly detection and mitigation approach applied on the broken AGV example.

6.2. Benefits, Limitations and Future Work

This paper presented ideas on integrating methods for the anomaly detection and mitigation of different types (cyber-attacks, adversarial attacks, system failures) within the context of Industry 4.0. This is a step towards a holistic approach to dealing with anomalies in the FoF environment. The anomaly detection and mitigation approach from Figure 7 already comprises anomalies with dedicated sub-data, sub-detection and sub-mitigation methods. Additionally, more types of anomalies can be added to approach the complexity of Industry 4.0. In this regard, it is important that novel sub-methods are well integrated. The detected anomalies should be distinguishable to identify what needs to be mitigated. Furthermore, the mitigation sub-methods must fit into the decision plan so that the suitability of the individual methods for the respective situation can be extracted. Moreover, the presented methods for anomaly detection and mitigation alone exceed the state of the art.

The application of the Cyber Kill Chain as a field-validated method to a new industrial use case of growing relevance regarding future factories introduces conceptually and operationally validated security considerations to the presented AGV fleet. The subsequent discussion of targeted cyber-threat scenarios has resulted in corresponding security measures tailored to AGVs and their proposed use-cases, thus providing a list of opportunities for detecting cyber-related anomalies and protecting AGVs from cyber-attacks. Since there is a limited amount of academic work dealing with the cyber-security of industrial AGVs in FoF, this contribution shows that security in future factories is indispensable and provides specific suggestions for detecting cyber-anomalies and addressing threats to AGVs with corresponding mitigations.

For the detection of process anomalies, data on KPIs of AGVs are used to derive an adapted version of classical control charts for anomaly detection. Those charts consider the increasing dynamicity of FoFs by integrating dynamic mean values instead of static means, such as in the literature. With the non-static mean, it is possible to consider the predicted KPIs that could be received from a Digital Twin run as a prediction of dynamic behavior.

The mitigation method of global reconfiguration provides a backup plan for dealing with anomalies if more specialized mitigation methods such as local reconfiguration do not provide sufficient mitigation solutions. This improves the resilience of a fleet of AGVs and helps to address the research gaps for resilient, sustainable material handling stressed

in [61]. To the best of the authors' knowledge, this is the first study to address the subject of adversarial attacks in Industry 4.0. The two proposed mitigation methods should therefore provide initial guidance and raise awareness of this subject in the future.

The respective system (here, the fleet of AGVs) needs to be prepared properly to put the presented methodology for anomaly handling into practice. The first step is to decide which methods for anomaly detection and mitigation should actually be implemented. For example, when an already existing fleet of AGVs is equipped with the means to increase its resilience a posterior, it would require an unreasonable amount of effort to change the software architecture of the AGVs to a service-based architecture, and the implementation of the local reconfiguration for anomaly mitigation might be out of scope. Instead, other methods for increasing the redundancy of AGV components might be considered and integrated into the overall implementation. In the following, high-level requirements for implementing the presented methods are summarized. The process anomaly detection method (autorefsec:detectionUnrecognizedFailure) and the global reconfiguration method (autorefsubsec:globalReconfiguration) both have the benefit that they can be integrated in an existing fleet of AGVs, as long as the AGVs can communicate with each other and to additional infrastructure (namely, a required Digital Twin and a monitoring component). Both methods do not require complete insights into other factory components. Instead, they rely on predictions of behavioral aspects of their context that can be derived from observations made by the AGV. It is particularly beneficial if the fleet does not have access to restricted sensitive production data such as machine configurations or detailed production schedules. As a downside for the application of both methods, it is a challenge to provide a Digital Twin of the fleet of AGVs, in which the context is reflected by predictions. In [59], it is shown that the prediction of transport tasks can be performed, e.g., with random forest methods. The construction and validation of a realistic Digital Twin is out of the scope of this study. The global reconfiguration method assumes additionally that the AGVs can select between already implemented configurations (e.g., determining how the transport tasks are distributed among the fleet). The same configurations need to be implemented in the Digital Twin.

In general, reconfiguration methodologies are focused to prevent factory architectures on restricting performance due to, i.e., changed factory goals, misbehaving systems etc. It is most likely sufficient to adapt system fleets to compensate for a change in the factory environment. However, the efficiency of a fleet depends on every individual system. In case of a failure of an individual AGV, usually a reconfiguration would exclude the failed system in the first place. Thus, the fleet has to assume all jobs of the excluded AGV and also needs to perform more jobs over the whole downtime, which leads to an increased workload for all remaining AGVs in the fleet. Therefore, a service-oriented reconfiguration (Section 5.2.1) can react locally at the individual AGV architecture by reconnecting service interfaces across system boundaries. Nevertheless, the use of SoR is not limited to preventing system downtime due to failed services; even more, it has the ability to extend the abilities of an individual system [51]. For future integration, SoR has to be extended by two essential functions. (1) The integration of an efficient service selection method. By using SoR in a fleet of only two systems, the number of available services is predictable, especially if both systems are equally constructed. When integrating more systems, it is easy to notice that the number of services will also increase, which on the one hand has to be included in the configuration selection process and on the other hand will increase the time to reconfigure the system. (2) Integrating an optimization algorithm is mandatory to not only replace services of the same type but even more to extend the AGV abilities and thus increase the performance and safety of each AGV.

The tailored application of the Cyber Kill Chain provides a global view on possible threat scenarios against AGVs and shows opportunities for disrupting cyber-attacks in each of their phases. In addition, the results are non-exhaustive and can be enriched with further threats and mitigations in case of different or more specific use-cases. However, implementing the suggested security measures is limited by the availability of capable

security tools dedicated to AGVs. Furthermore, the proposed automation, monitoring and response activities are highly dependent on the budget and skill-set of a company and will likely be out of reach for most small and medium-sized enterprises (SMEs) working with AGVs. Even with the proper budget, skills and tools, security always comprises a significant operational human aspect that co-determines the effectiveness of any technical measure. Therefore, the presented security considerations require corresponding standards, procedures and guidelines to address the human factor with regard to AGV security. This topic constitutes an aspect for future work to explore. When using AI-related methods, we do not need to build a simulator, namely Digital Twins; only the collected data from the real world are required. However, collecting as much clean data as possible is also a challenge because more data can give us more accurate results. This issue needs to be addressed soon. As shown in Figure 6, data exchange is a crucial element in anomaly processing, not only between operational systems but also between components of anomaly detection and mitigation. A data lake is one possibility to provide all systems and components with the required data whenever needed. This prevents the need to store the same data multiple times in various components. In this paper, it was not investigated how the full potential of data lakes can be used for storing and providing unstructured data, as well as the benefits of a growing number of anomaly detection and mitigation methods, which will be considered as future work.

From a cyber-security perspective, it is difficult to identify malicious intent within the technically valid commands an attacker issues to an AGV. As long as these commands are syntactically valid and controlled variables remain within defined constraints, malicious commands are indistinguishable from legitimate ones. However, historical operational process data could be leveraged to address this shortcoming of traditional threat detection that relies solely on syntax and static variable boundaries. Production systems supported by AGVs usually follow a strictly defined process that requires specific commands to be issued at corresponding process steps. This implies that the order and often also the parameters of commands are highly predictable, and one can expect recurring patterns. Therefore, past process data contain baselines of regular process flows that could be used to train an AI system to detect malicious anomalies, even if they were triggered by technically legitimate commands. As sketched in Section 5.2.2, predictions on transportation tasks were already used to feed a Digital Twin of the fleet for predicting other AGV parameters. It remains open to check how the combination of predictions with Digital Twins for detecting anomalies in the presented use-case performs in comparison with purely AI-based methods. To account for naturally occurring process deviations, this AI would need to be trained on large data sets to contain representative normal process behavior. It would ideally keep learning while in operation to acquire new process behavior. As computational resources become widely available in interconnected factories, historical process data provide significant opportunities for effective anomaly detection in future production systems.

Author Contributions: Conceptualization, M.G. and L.F.; software, C.B.; writing—original draft preparation, L.F., E.K., A.S., A.W., C.-Y. K., B.W., M.G., E.M. and P.B.; writing—review and editing, A.S. and L.F.; visualization, A.W., C.-Y.K. and P.B.; All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the German Federal Ministry of Education and Research grant number 01 | S18061C.

Institutional Review Board Statement: Not applicable

Informed Consent Statement: Not applicable

Conflicts of Interest: The authors declare no conflict of interest. The funders 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.

Abbreviations

The following abbreviations are used in this manuscript:

AMCL	adaptive Monte Carlo landmark
AGV	Automated Guided Vehilce
CCaaS	Cybercrime-as-a-service
CSCA	Cyber Supply Chain Attacks
DDoS	Distributed-Denial-of-Service
FoF	Factory of the Future
IP	intellectual property
KPI	key performance indicator
MES	manufacturing execution system
MOOP	multi-objective optimization problem
MRTA	Multi Robot Task Assignment
OS	operating system
OT	operational technology
RaaS	Ransomware-as-a-Service
SoR	Service-Oriented Reconfiguration
TA	threat actor

References

1. Mohamed, M. Challenges and benefits of industry 4.0: An overview. *Int. J. Supply Oper. Manag.* **2018**, *5*, 256–265.
2. Vaidya, S.; Ambad, P.; Bhosle, S. Industry 4.0–A Glimpse. *Procedia Manuf.* **2018**, *20*, 233–238. In Proceedings of the 2nd International Conference on Materials, Manufacturing and Design Engineering (iCMMD2017), 11–12 December 2017; MIT Aurangabad: Maharashtra, India. https://doi.org/10.1016/j.promfg.2018.02.034.
3. Ahanger, T.A.; Aljumah, A. Internet of Things: A comprehensive study of security issues and defense mechanisms. *IEEE Access* **2018**, *7*, 11020–11028.
4. Sheffi, Y. Resilience: What it is and how to achieve it. *Retrieved Oct.* **2008**, *1*, 2013.
5. Choi, S.; Youm, S.; Kang, Y.S. Development of scalable on-line anomaly detection system for autonomous and adaptive manufacturing processes. *Appl. Sci.* **2019**, *9*, 4502.
6. Wu, Y.; Dai, H.N.; Tang, H. Graph neural networks for anomaly detection in industrial internet of things. *IEEE Internet Things J.* **2021**, *9*, 9214–9231.
7. Quarta, D.; Pogliani, M.; Polino, M.; Maggi, F.; Zanchettin, A.M.; Zanero, S. An Experimental Security Analysis of an Industrial Robot Controller. In Proceedings of the 2017 IEEE Symposium on Security and Privacy (SP), San Jose, CA, USA, 22–24 May 2017; pp. 268–286. https://doi.org/10.1109/SP.2017.20.
8. Gu, X.; Jin, X.; Ni, J.; Koren, Y. Manufacturing system design for resilience. *Procedia Cirp* **2015**, *36*, 135–140.
9. Kriaa, S.; Cambacedes, L.P.; Bouissou, M.; Halgand, Y. A survey of approaches combining safety and security for industrial control systems. *Reliab. Eng. Syst. Saf.* **2015**, *139*, 156–178.
10. Siepmann, D.; Graef, N. Industrie 4.0–Grundlagen und Gesamtzusammenhang. In *Einführung und Umsetzung von Industrie 4.0*; Roth, A., Ed.; Springer Gabler: Berlin, Germany, 2016; pp. 17–82. https://doi.org/10.1007/978-3-662-48505-7_2.
11. Laufenburg, R. Cybercrime-as-a-Service. 2021. Available online: https://www.pcspezialist.de/blog/2021/09/15/cybercrime-as-a-service-caas/ (accessed on 19 May 2022).
12. Panda Security. 73 Ransomware Statistics Vital for Security in 2022. 2022. Available online: https://www.pandasecurity.com/en/mediacenter/security/ransomware-statistics/ (accessed on 31 May 2022).
13. SonicWall. 2022 SonicWall Cyber Threat Report. 2022. Available online: https://www.sonicwall.com/2022-cyber-threat-report/ (accessed on 28 July 2022).
14. IBM Security. Cost of a Data Breach Report 2021. 2021. Available online: https://www.ibm.com/downloads/cas/OJDVQGRY (accessed on 28 July 2022).
15. IBM Security. X-Force Threat Intelligence Index 2022. 2022. Available online: https://www.ibm.com/downloads/cas/ADLMYLAZ (accessed on 28 July 2022).
16. Symantec. Internet Security Threat Report. Vol. 24. 2019. Available online: https://docs.broadcom.com/doc/istr-24-2019-en (accessed on 28 July 2022).
17. Kern, E.; Szanto, A. Cyber Supply Chain Attacks. Forthcoming. Brandenburgisches Institut für Gesellschaft und Sicherheit. BIGS Policy Paper. 10.
18. Bryan, J. A Better Way to Manage Third-Party Risk. 2019. Available online: https://www.gartner.com/smarterwithgartner/a-better-way-to-manage-third-party-risk (accessed on 31 May 2022).
19. Smith, Z.M.; Lostri, E.; Lewi, J.A. The Hidden Costs of Cybercrime. 2020. McAfee. Available online: https://www.mcafee.com/enterprise/en-us/assets/reports/rp-hidden-costs-of-cybercrime.pdf (accessed on 28 July 2022).

20. Fortinet. Global Threat Landscape Report. A Semiannual Report by FortiGuard Labs. 1H 2021. 2021. Available online: https://www.fortinet.com/content/dam/fortinet/assets/threat-reports/report-threat-landscape-2021.pdf (accessed on 28 July 2022).
21. FireEye Mandiant Services. M-Trends 2020 Special Report. 2020. Available online: hhttps://www.mandiant.com/sites/default/files/2021-09/mtrends-2020.pdf (accessed on 28 July 2022).
22. Dunn, P. Deloitte and the Ethics of Corporate Espionage. In Proceedings of the International Association for Business and Society, Hong Kong SAR, China, 6–10 June, 2018; Number 29, pp. 65–70. https://doi.org/10.5840/iabsproc2018297.
23. Javers, E. Accountants and Spies: The Secret History of Deloitte's Espionage Practice. 2016. Available online: https://www.cnbc.com/2016/12/19/accountants-and-spies-the-secret-history-of-deloittes-espionage-practice.html (accessed on 31 May 2022).
24. Porter, M.E. *Competitive Strategy: Techniques for Analyzing Industries and Competitors*. 1980. University of Michigan Free Press: Ann Arbor, MI.
25. Davis, N.; Raina, G.; Jagannathan, K. A framework for end-to-end deep learning-based anomaly detection in transportation networks. *Transp. Res. Interdiscip. Perspect.* **2020**, *5*, 100112. https://doi.org/10.1016/j.trip.2020.100112.
26. Chandola, V.; Banerjee, A.; Kumar, V. Anomaly detection: A survey. *ACM Comput. Surv. CSUR* **2009**, *41*, 1–58.
27. IndustryWeek in collaboration with Emerson How Manufacturers Can Achieve Top Quartile Performance. 2016. Available online: https://partners.wsj.com/emerson/unlocking-performance/how-manufacturers-can-achieve-top-quartile-performance/ (accessed on 19 July 2022).
28. Kamat, P.; Sugandhi, R. Anomaly detection for predictive maintenance in industry 4.0-A survey. In E3S Web of Conferences; EDP Sciences, Les Ulis, France 2020; Volume 170, p. 02007.
29. Sharma, B.; Sharma, L.; Lal, C. Anomaly detection techniques using deep learning in IoT: A survey. In Proceedings of the 2019 International Conference on Computational Intelligence and Knowledge Economy (ICCIKE), Dubai, United Arab Emirates, 11–12 December 2019; pp. 146–149.
30. Landauer, M.; Onder, S.; Skopik, F.; Wurzenberger, M. Deep Learning for Anomaly Detection in Log Data: A Survey. *arXiv* **2022**, arXiv:2207.03820.
31. Trauer, J.; Pfingstl, S.; Finsterer, M.; Zimmermann, M. Improving Production Efficiency with a Digital Twin Based on Anomaly Detection. *Sustainability* **2021**, *13*, 10155.
32. Bécue, A.; Maia, E.; Feeken, L.; Borchers, P.; Praça, I. A New Concept of Digital Twin Supporting Optimization and Resilience of Factories of the Future. *Appl. Sci.* **2020**, *10*, 4482. https://doi.org/10.3390/app10134482.
33. Duman, T.B.; Bayram, B.; İnce, G. Acoustic Anomaly Detection Using Convolutional Autoencoders in Industrial Processes. In *Proceedings of the 14th International Conference on Soft Computing Models in Industrial and Environmental Applications (SOCO 2019)*; Martínez Álvarez, F., Troncoso Lora, A., Sáez Muñoz, J.A., Quintián, H., Corchado, E., Eds.; Springer International Publishing: Cham, Switzerland, 2020; pp. 432–442.
34. Münz, G.; Li, S.; Carle, G. Traffic Anomaly Detection Using K-Means Clustering. In *GI/ITG Workshop MMBnet*; University of Bamberg Press, Germany 2007; Volume 7, p. 9.
35. Hsieh, R.J.; Chou, J.; Ho, C.H. Unsupervised online anomaly detection on multivariate sensing time series data for smart manufacturing. In Proceedings of the 2019 IEEE 12th Conference on Service-Oriented Computing and Applications (SOCA), Kaohsiung, Taiwan, 18–21 November 2019; pp. 90–97.
36. Windmann, S.; Maier, A.; Niggemann, O.; Frey, C.; Bernardi, A.; Gu, Y.; Pfrommer, H.; Steckel, T.; Krüger, M.; Kraus, R. Big data analysis of manufacturing processes. *J. Phys. Conf. Ser.* **2015**, *659*, 012055.
37. Stojanovic, L.; Dinic, M.; Stojanovic, N.; Stojadinovic, A. Big-data-driven anomaly detection in industry (4.0): An approach and a case study. In Proceedings of the 2016 IEEE International Conference on Big Data (Big Data), Washington, DC, USA, 5–8 December 2016; pp. 1647–1652.
38. Zhang, C.; Wang, Z.; Ding, K.; Chan, F.T.; Ji, W. An energy-aware cyber physical system for energy Big data analysis and recessive production anomalies detection in discrete manufacturing workshops. *Int. J. Prod. Res.* **2020**, *58*, 7059–7077.
39. Hollerer, S.; Kastner, W.; Sauter, T. Towards a threat modeling approach addressing security and safety in OT environments. In Proceedings of the 2021 17th IEEE International Conference on Factory Communication Systems (WFCS), Linz, Austria, 9–11 June 2021; pp. 37–40.
40. Novak, T.; Treytl, A.; Palensky, P. Common approach to functional safety and system security in building automation and control systems. In Proceedings of the 2007 IEEE Conference on Emerging Technologies and Factory Automation (EFTA 2007), Patras, Greece, 25–28 September 2007; pp. 1141–1148.
41. Antón, S.D.; Schotten, H.D. Putting together the pieces: A concept for holistic industrial intrusion detection. In *Proceedings of the ECCWS 2019 18th European Conference on Cyber Warfare and Security*. Academic Conferences and Publishing Limited: Reading South Oxfordshire, UK 2019; p. 178.
42. Bauer, D.; Böhm, M.; Bauernhansl, T.; Sauer, A. Increased resilience for manufacturing systems in supply networks through data-based turbulence mitigation. *Prod. Eng. Res. Dev.* **2021**, *15*, 385–395. https://doi.org/10.1007/s11740-021-01036-4.
43. Hu, M.; Liao, Y.; Wang, W.; Li, G.; Cheng, B.; Chen, F. Decision Tree-Based Maneuver Prediction for Driver Rear-End Risk-Avoidance Behaviors in Cut-In Scenarios. *J. Adv. Transp.* **2017**, *2017*, 7170358.

44. Xia, W.; Goh, J.; Cortes, C.A.; Lu, Y.; Xu, X. Decentralized coordination of autonomous AGVs for flexible factory automation in the context of Industry 4.0. In Proceedings of the 2020 IEEE 16th International Conference on Automation Science and Engineering (CASE), Hong Kong, China, 20–21 August 2020; pp. 488–493.
45. Herrero-Perez, D.; Matinez-Barbera, H. Decentralized coordination of autonomous agvs in flexible manufacturing systems. In Proceedings of the 2008 IEEE/RSJ International Conference on Intelligent Robots and Systems, Nice, France, 22–26 September 2008; pp. 3674–3679.
46. Yao, F.; Keller, A.; Ahmad, M.; Ahmad, B.; Harrison, R.; Colombo, A.W. Optimizing the scheduling of autonomous guided vehicle in a manufacturing process. In Proceedings of the 2018 IEEE 16th International Conference on Industrial Informatics (INDIN), Porto, Portugal, 18–20 July 2018; pp. 264–269.
47. Salehie, M.; Tahvildari, L. Towards a Goal-Driven Approach to Action Selection in Self-Adaptive Software. *Softw. Pract. Exp.* **2012**, *42*, 211–233. https://doi.org/10.1002/spe.1066.
48. Rosa, L.; Rodrigues, L.; Lopes, A.; Hiltunen, M.; Schlichting, R. Self-Management of Adaptable Component-Based Applications. *IEEE Trans. Softw. Eng.* **2013**, *39*, 403–421. https://doi.org/10.1109/TSE.2012.29.
49. Mauro, J.; Nieke, M.; Seidl, C.; Yu, I.C. Context Aware Reconfiguration in Software Product Lines. In Proceedings of the Tenth International Workshop on Variability Modelling of Software-Intensive Systems, VaMoS '16, Salvador, Brazil, 27–29 January 2016; pp. 41–48. https://doi.org/10.1145/2866614.2866620.
50. Sinreich, D. An architectural blueprint for autonomic computing. In *Technical Report*; IBM: New York, US 2006.
51. Siefke, L.; Sommer, V.; Wudka, B.; Thomas, C. Robotic Systems of Systems Based on a Decentralized Service-Oriented Architecture. *Robotics* **2020**, *9*, 78. https://doi.org/10.3390/robotics9040078.
52. Thomas, C.; Mirzaei, E.; Wudka, B.; Siefke, L.; Sommer, V. Service-Oriented Reconfiguration in Systems of Systems Assured by Dynamic Modular Safety Cases. In *Communications in Computer and Information Science*; Springer International Publishing: Basel, Switzerland, 2021; pp. 12–29. https://doi.org/10.1007/978-3-030-86507-8_2.
53. Understanding and Mitigating Russian State-Sponsored Cyber Threats to U.S. Critical Infrastructure., **2022** Available online: https://www.cisa.gov/uscert/ncas/alerts/aa22-011a (accessed on 21 March 2022).
54. Koutras, M.V.; Bersimis, S.; Maravelakis, P.E. Statistical Process Control using Shewhart Control Charts with Supplementary Runs Rules. *Methodol. Comput. Appl. Probab.* **2007**, *9*, 207–224. https://doi.org/10.1007/s11009-007-9016-8.
55. Eschemann, P.; Borchers, P.; Lisiecki, D.; Krauskopf, J.E. Metric Based Dynamic Control Charts for Edge Anomaly Detection in Factory Logistics. In Proceedings of the 3nd International Joint Conference on Automation Science and Engineering (JCASE 2022, 14.-16. October 2022, Chengdu, China, to appear).
56. The Cyber Kill Chain. **2011.** Available online: https://www.lockheedmartin.com/en-us/capabilities/cyber/cyber-kill-chain.html (accessed on 21 March 2022).
57. Korsah, G.A.; Stentz, A.; Dias, M.B. A comprehensive taxonomy for multi-robot task allocation. *Int. J. Robot. Res.* **2013**, *32*, 1495–1512. https://doi.org/10.1177/0278364913496484.
58. Gerkey, B.P.; Matarić, M.J. A Formal Analysis and Taxonomy of Task Allocation in Multi-Robot Systems. *Int. J. Robot. Res.* **2004**, *23*, 939–954. https://doi.org/10.1177/0278364904045564.
59. Borchers, P.; Lisiecki, D.; Eschemann, P.; Feeken, L.; Hajnorouzi, M.; Stierand, I. Comparison of Production Dynamics Prediction Methods to Increase Context Awareness for Industrial Transport Systems. In *Proceedings of the European Simulation and Modelling Conference 2021, ESM 2021; 7–29 October 2021, Rome, Italy*; 2021; pp. 49–55.
60. Hwang, C.L.; Masud, A.S.M. Methods for Multiple Objective Decision Making. In *Multiple Objective Decision Making—Methods and Applications: A State-of-the-Art Survey*; Springer: Berlin/Heidelberg, Germany, 1979; pp. 21–283. https://doi.org/10.1007/978-3-642-45511-7_3.
61. Li, R.; Tian, X.; Yu, L.; Kang, R. A systematic disturbance analysis method for resilience evaluation: A case study in material handling systems. *Sustainability* **2019**, *11*, 1447.