Available online at www.sciencedirect.com

ScienceDirect

Procedia Computer Science 180 (2021) 492–501

Procedia Computer Science

www.elsevier.com/locate/procedia

International Conference on Industry 4.0 and Smart Manufacturing

CONTEXT: An Industry 4.0 Dataset of Contextual Faults in a Smart Factory

Lukas Kaupp[a,*], Heiko Webert[a], Kawa Nazemi[a], Bernhard Humm[a], Stephan Simons[a]

[a]*Darmstadt University of Applied Sciences, Haardtring 100, 64295 Darmstadt, Germany*

Abstract

Cyber-physical systems in smart factories get more and more integrated and interconnected. Industry 4.0 accelerates this trend even further. Through the broad interconnectivity a new class of faults arise, the contextual faults, where contextual knowledge is needed to find the underlying reason. Fully-automated systems and the production line in a smart factory form a complex environment making the fault diagnosis non-trivial. Along with a dataset, we give a first definition of contextual faults in the smart factory and name initial use cases. Additionally, the dataset encompasses all the data recorded in a current state-of-the-art smart factory. We also add additional information measured by our developed sensing units to enrich the smart factory data even further. In the end, we show a first approach to detect the contextual faults in a manual preliminary analysis of the recorded log data.

Keywords: contextual faults; smart factory; cyber-physical systems; fault diagnosis; anomaly detection;

1. Introduction

In a smart factory cyber-physical systems (CPS) in production lines become more and more integrated and interconnected. Industry 4.0 expedites this transition in manufacturing processes. As a result the environment in which faults have to be analyzed becomes more complex. In addition, the root cause of the error is most likely not found during a single incident within the production line, but rather as a cumulative chain of different signals in different CPS that must be interpreted to find the cause of the error. In order to get a proper interpretation of this error contextual information is required. We state that each CPS forms its own context and consequently the production line consists of multiple contexts and maintains its own. Therefore we classify these types of faults as contextual faults. To our knowledge, there is still no definition of context errors for the Industry 4.0 smart factory domain, but we borrow a first definition of contextual faults from Alos et al. [2]. Alos et al. [2] define "... contextual faults mean that a defective sensor shows invalid values concerning the context of other attributes" in their work about the detection of contextual

* Corresponding author. Tel.: +49.6151.16-30141.
E-mail address: lukas.kaupp@h-da.de

1877-0509
10.1016/j.procs.2021.01.265

faults in an unmanned aerial vehicle (UAV) in 2020. Like a smart factory, an UAV consists of several subsystems and sensors arranged side by side, so that this definition is sufficient to be translated into the smart factory domain. We define a contextual fault as an invalid or unexpected value within the attribute context of the CPS, its surrounded contexts or the global production line context. We have already defined a context, as one or more windows that extend around one or more outliers and include all available sources of information as well as other contexts [10]. Apart from this definition we aim to release a first set of contextual faults, their description and the corresponding dataset with this publication. Correspondingly, we name our dataset CONTEXT.

With the proposed CONTEXT dataset we want to address other challenges that currently exits in open industrial smart manufacturing datasets as well. Most available industrial datasets relate only to individual machines [21, 13, 1, 19, 18, 7, 14, 16, 17, 5, 20] whereas a few relate to a whole production line [15, 12, 8, 3]. A part only takes into account one or two characteristics to be analyzed [21, 13, 15, 7, 12, 16, 3] e.g. energy-consumption, temperature or acceleration. Another major problem with real-world industrial datasets is anonymization and obfuscation [8, 16, 17], where companies can retrieve valuable feedback but no real scientific contribution can be made because the fault cannot be tracked back in the system. Apart from preventing the actual incident from being named, it is not possible to state the origin of the cause in such anonymized datasets.

In respect to the aformentioned challenges, our dataset is open, multivariate and contains many different attribute characteristics. Additionally we developed own sensing units (SU) to enrich the smart factory data with additional information about the production process. We record the dataset in the smart factory at the Darmstadt University of Applied Sciences (DUAS) [22], a production-equal smart factory to build electric relays identical to those found in wind turbines. Each station in the production line of the smart factory encompasses a corresponding state-of-the-art OPC-UA model which holds all the information about the different hardware (e.g. sensors) and software modules (e.g. production software) of the CPS.

We captured three contextual faults in the smart factory. First, a compressor failure is recorded, in which through a lack of air pressure the press and the electrical inspection will slow down the process but will not stop their operations. Second, one of the two shuttles is held in the manual inspection bay so the production time is doubled. Third, the fault is a missing part while assembling. The process behaves normal except some slight changes while assembling and during the pressing operation. As a limitation, our proposed contextual faults address blind spots specific to the smart factory at DUAS. As every production site has its own contextual faults, this shall be seen as an example or extensible list and has no claim to completeness. In the end, we give a short first manual preliminary analysis of the contextual faults to name starting points for further investigation.

Our contribution in this work is three-fold. We give a definition of contextual faults in a smart factory (1). We name examples and use cases for contextual faults (2). We describe and release a novel smart factory dataset for fault diagnosis (3).

2. Exisiting Datasets

Smart factory datasets are rare, especially if related to Industry 4.0. Most data sets cover individual machines, not an entire production line. In case the focus relies on a complete production line, only one or two characteristics (e.g. temperature or energy consumption) are recorded. A minor part is related to faults, the major part is related to wear. Wear can also be seen as a non-deterministic fault on the long run, but wear-related faults stop the production resulting in an major issue. Anonymization or obscuration are a major issue on real-word industrial datasets. Datasets currently available are fairly simple. Most of the time the following datasets consist of individual attribute characteristics, whereby real-world datasets are far more complex and of several characteristics (e.g. integers, floats, messages)[9].

2.1. Individual Machines

Individual machine datasets are the most common among the industrial setting. The UCI Machine Learning Repository [6] contains a lot of different datasets. Two datasets by Seabra Lopes et al. [21] and Schneider et al. [20] suit our setting. Seabra Lopes et al. [21] released a multivariate time-series of force and torque measurements of robot execution failures while assembling parts. Schneider et al. [20] released a dataset about condition monitoring a test-

rig of hydraulic systems. The NASA maintains 16 dataset repositories in their prognostic data repository[1] related to aerospace. Not all of them address an industrial setting. Agogino et al. [1] provide a milling datasets that contains different speeds, feeds and depth of cuts with different levels of wear. Lee et al. [13] use accelerometers in three test-to-failure experiments on different bearings. In addition to the NASA repository, the Prognostics and Health Management Society (PHM Society[2]) covers different industrial datasets in their challenges provided during their conferences. Saxena et al. provide two datasets that contain turbofan engine degradation simulation data [19, 18]. In 2009 the PHM Society released a dataset about fault detection and magnitude estimation for a generic gearbox utilizing accelerometer sensors [7]. In 2018 the PHM Society [5] published a data set concerning an ion mill etch tool, which is used during a wafer manufacturing processes. The dataset consists of faults, regarding flowcool, ion mill chambers, as well as wear in the electric grids or the ion chambers. The error prediction in both location and time was the major challenge, as well as scheduling downtimes, regarding maintenance on ion mills. Real-world data is even harder to obtain and only a few datasets on Kaggle[3] refer to manufacturing. Magalhães Oliveira discloses data about a flotation plant in a mining operation [14]. Data encompass silica concentration that has to be reduced for a higher purity of the ore and lower environmental impacts. No health data is given on the machinery itself. The last two relevant datasets were released from the company inIT [16, 17] and refer to an article about self-organizing maps published by Birgelen et al. [4]. The first dataset is about the wear of a cutter in a cutting assembly during operation [16, 4]. The second dataset contains data about a production plant with anonymized features. In addition the dataset consists of the chosen features (also anonymized) included other components of the production plant by hand [17, 4]. Revealing the problem of open industrial datasets that real-world data is either anonymized or obscurified to retain intellectual property or to take security concerns into consideration, but foremost not to give a competitor an advantage in the market.

2.2. Production Lines

McCann et al. [15] do not reveal the sensor types that are used to surveil a semi-conductor manufacturing process and neither the names or position of the sensors are mentioned. The smart energy data published by the RWTH Aachen University consists of data about the energy consumption of a smart factory [12]. The energy profiles were recorded during the EU-funded FINESCE project[4] and contain the different energy consumption of different stations within the smart factory. As part of the data challenge at the IEEE BigData 2016 conference, the company Bosch [8] released a fully anonymized dataset on a production line that produces chocolate soufflés. It is not possible to identify the purpose of a single station nor the process which they are involved in. But in return this is the most complete industrial production line dataset. Bandeira de Mello Martins et al. [3] published a dataset on a production line that covers the energy consumption of different parts of a poultry feed factory in Brazil. As a result, the dataset contains only the voltage values of the different involved stations.

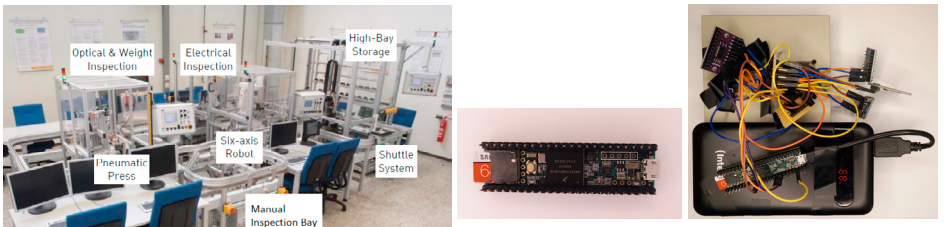
3. The Smart Factory

The smart factory at the DUAS produces electric relays (Figure 1a). One relay component consists of a socket, on which a relay and a protection module is mounted. These types of relays are often used for processes in wind turbines. The smart factory manufacturing process is equal to production. Therefore, the smart factory is an optimal testbed for current Industry 4.0 smart factory-related research questions, as the proposed contextual faults, where the context of multiple stations is necessary to determine the reason for an issue. The smart factory consists of a high-bay storage (with a three-axis robot), a six-axis robot assembly station, a press, an electrical inspection, a manual inspection bay and everything is interconnected through a monorail shuttle-system (with two shuttles). The high-bay storage comprises pallets with not assembled relays. The assembly station will mount the relays with the help of the six-axis robot. Furthermore, the press will assure the electrical connection between the socket and the relay by alignment of the parts by force. In addition the electrical inspection recognizes the type of the relay and uses different test routines

[1] https://ti.arc.nasa.gov/tech/dash/groups/pcoe/prognostic-data-repository
[2] https://www.phmsociety.org
[3] https://kaggle.com
[4] http://www.finesce.eu

to assure the electrical functionality of the produced relays. At the time of recording, the optical and weight inspection machine (Figure 1a) is out of order and not included in the dataset and use cases. All stations have corresponding OPC-UA models. Each station has several safety sensors in place to protect the personnel.

But even with these optimal test conditions, issues occur in production. There are blind spots in surveillance which do not trigger an automatic stop of the production leading in the worst case to incomplete or damaged relays. These uncovered contextual faults can be recognized if contextual knowledge is applied. We present three use cases where contextual faults exist that are not covered by any safety routine. Therefore, no error is thrown and the production will not be stopped. A detection of the contextual faults will result in a faster and more reliable production of the electronic relays, because counter measures can be taken.



(a) Structure of the smart factory (modified from [22])

(b) Arduino Processing Unit (Teensy 3.6)

(c) Prototype of the Shuttle Sensing Unit

Fig. 1: a) shows the smart factory with all its stations (modified from [22], added manual inspection bay). b) depicts the used Arduino board (Teensy 3.6) to capture and process the external sensor data. c) presents the assembled prototype of the shuttle sensing unit, that features a 10.000 mAh battery, an Arduino processing unit, pull-up resistors, an I^2C-multiplexer (TCA9548A) and three sensors (BMP280, MPU9250, TSL2591) on a pallet

3.1. Use Cases

The proposed use cases are chosen to visualize contextual faults and address specific blind spots in the smart factory at DUAS. These use cases should be not taken as a complete list of possible contextual faults, rather than an extensible list of faults that can appear also in other production sites. Every production site has introduced its own contextual faults and surveillance blind spots either hardware-related or programmatically.

3.1.1. Missing Pressure

At normal speed the production time in the smart factory is around four minutes, from outsourcing the pallet to store the finally mounted and successfully tested relay. The six-axis robot, the press and the electrical inspection use air pressure to operate. The robot uses pressure to grab its tool and operate the gripper to assemble all necessary components into the socket, including the relay. The press uses air pressure to gain velocity and press the relay part into the relay socket. In the end, the electrical inspection uses air pressure to move the testing device on top of the socket in order to start the inspection process. Each of these stations is affected due to a leakage in the pressure system. The process will be slowed down, through a lower movement speed of each station. Furthermore, the shuttles are forced to wait until the operation is completed. In order to detect this kind of contextual fault, data of all involved stations is necessary, because no station or measurement alone will lead to the combined inferred cause of an air pressure leakage.

3.1.2. Shuttle Drop Out

Every pallet with a non-assembled relay will be placed on one of two shuttles. The monorail system interconnects all stations. If a shuttle is missing during the production process, the product throughput is reduced by a factor of two, while the production time gets doubled. Furthermore, the reasons for a missing shuttle vary, but at every situation the shuttle waits in the manual inspection bay and awaits a signal form the personnel that the error that cannot be resolved by the machinery is resolved by hand in order to queue back into the production cycle. Both shuttles will remain in the system and send a positive feedback that they are in use. Only the amount of stations in operation and the time between active stations will vary. This brings up another contextual fault, that can only be recognized due to interactions between different stations and which is not directly covered by any thrown error.

3.1.3. Missing Parts

Shuttle speeds may vary during a production cycle. A station may be in use and the shuttle has to throttle its speed or to stop and in return accelerate speed to reach the station. At each time a part could fall of the pallet (even as the pallet obtains engravings) or a part is already missing in the High-Bay storage. As a result, the robot will grasp air and place invisible parts in the relay socket. The press, as well as the electrical inspection operates normally. The electrical inspection will testify the relay as faulty and send it to the manual inspection bay. As the optical & weight inspection is currently out of order, the missing part fault is hard to detect. The resulting contextual fault can again only be unveiled if all stations and quirks are taken into consideration in order to distinguish faults, caused by a missing part or by electrical faults of the relay socket.

3.2. Sensing Units as Additional Source of Information

We developed additional Sensing Units (SUs) and attached them to each station and one shuttle (Figure 1c) to gain access to additional information about the process and therefore to encounter the contextual faults. A SU is composed of an Arduino Processing Unit (Teensy 3.6), an I^2C-multiplexer (TCA9548A) and up to three groups of three sensors (light, pressure and a 9-axis depth of field (DOF) sensor). The light sensor (TSL2591) records infrared, full-spectrum and human-visible light in ranges of 188uLux to 88.000 Lux. Moreover, the pressure sensor (BMP280) measures pressure (300...1100 hPa) and temperature (-40...85°C). Lastly, the 9-axis DOF sensor (MPU9250) consists of a 3-axis gyroscope (±250, ±500, ±1000, ±2000 dps), 3-axis accelerometer (±2g, ±4g, ±8g, ±16g) and a 3-axis magnetometer (±4800µT). In order to avoid programmatically introduced errors, each SU is constructed in the same way, only the amount of sensor groups per SU vary. Additionally, we employ the SUs in order to compensate lost information (e.g. packets, commands or events) due to heavy-load on the machinery while recording the use cases.

3.3. Setup and Course of the Recordings

The recording has been done under real-world circumstances. Therefore limitations of bandwidth exist, lost packages and events may occur, as well as out of order events may appear. The OPC-UA models of the stations emit 33.089 variables that are possible subscription endpoints. During the recordings we subscribe to all 33.089 variables at the same time and listen only on data change events to reduce system load and minimize the chance of faults that appear through lost signals. We captured the normal operation mode (ground truth) as follows:

We let build sixteen electric relays, eight proven as good and eight tested faulty in the electrical inspection. The high-bay storage obtains all the not assembled parts on pallets. A three-axis robot places the pallets onto one of two shuttles if an order is placed. The monorail system connects all stations. After the successful placement of the pallet, the shuttle drives to the assembly station. The six-axis robot will assemble all parts on the pallet and build the relay. Next, the shuttle arrives at the press where the relay gets fixed so that all electric contacts within the relay are tightly coupled. Normally, the shuttle would check the relay socket in the optical and weight inspection, but as this station is out of order for the time of the recordings the shuttle will pass through the station and stop at the electrical inspection. The electrical inspection will read the data matrix code (DMC) on the side of the relay socket and determine the right routine to proof the expected electrical properties of the relay. If the test passes successfully the pallet is placed back by the three-axis robot in the high-bay storage, otherwise the shuttle will pass through the high-bay storage and the assembly station to enter the manual inspection bay for further investigation of the problem. Only if the DMC is readable or consists of a valid code for a testing routine the relay is tested, otherwise the relay will be automatically flagged as faulty and head also to the manual inspection bay.

After we record ground truth, we capture data for each of the aforementioned use cases. In order to simulate a pressure leakage we slowly open a valve at the last pressure affected station (electrical inspection). Every two relays the pressure will be lowered until a minimum operating level of 1.9 bar is left, after ten assemblies the valve is closed and pressure moves back to a normal operating level. Next, the shuttle drop out is simulated after the first three successful assemblies. The shuttle enters the manual inspection bay and do not queue back before the faulty cycle begins. Lastly, the missing parts use case is recorded. In the faulty cycle (eight relays tested faulty) missing parts are introduced, so there will be slightly changes in values while assembling and press and no abnormal behavior distinguished from ground truth (enters manual inspection bay as usual after testing faulty).

4. Data

This section describes the recorded data and initial findings. Starting with a brief overview about the structure of the recorded log files and found limitations. Followed by a short description of interesting aspects in the recorded data of each use case. Given the size of the proposed data set, only parts of it can be shown in this publication. We manually examines first characteristics of each use case. Later a study is conducted to find these correlations and interconnections automatically.

4.1. Data Structure per Run

The data structure differentiates between OPC-UA and SU recordings. In OPC-UA 33.089 different subscription endpoints exist. Table 1 shows all available data types and their respective node count. Furthermore, Table 1 exhibits multiple findings. First, the majority of recorded nodes are Boolean values and numbers. Second, the data type String is used in multiple and questionable ways. The String data type can hold everything from messages, bytes to arrays and numbers. Third, we have found Boolean values as well as stringyfied Boolean arrays transmitted using the Boolean data type. Subsequent studies have to take countermeasures.

Table 2 represents the current log file format that consists of date, nodeid, value and dtype (data type). Date contains a DateTime string. Furthermore, the string unveiled by nodeid contains multiple information separated through a dot. The naming scheme is as follows. First, a namespace is given. Followed by the station name. Next, multiple software as well as hardware modules are named; again dot separated. Whereas, the last chunk is the variable of the module. As a result of this naming convention, it is possible to assign and track each value through the whole smart factory. In the end, the value of the data changed event and its data type is tracked. For all data changed events only one log file exists.

In contrary, the SU log file format is rather simple. Its datetime field is a DateTime formatted string. TCAp and sGrp are Integer. All other columns are in Float format. An example is given in Table 3. The TCAp column describes the port number of the I^2C-multiplexer on which the sensor group is attached, whereas the sGRP column is only a counter which identifies the particular sensor group. Each log line represents the measurement of one sensor group. The naming convention is the sensor type, underscore, the measured unit. Each station and one shuttle has its own SU, consequently its own log file. As a result, one dataset consists of six log files per use case.

Table 1: Datatypes of all available 33.089 OPC-UA Variables with Examples

Data Type	Node Count	Example Values
Boolean	17874	True, "[False,]"
Byte	4918	11
ByteString	406	b'\xff...'
DateTime	5	2020-07-09 16:05:11.795000
Double	1	0.0
Float	1248	3.1233999729156494
Int16	3003	4
Int32	1822	16
Int64	6	2103635700381566
SByte	34	"[32, 32,..]"
String	93	V3.0, "['+40,,', '', '', '', '']"
UInt16	2655	2
UInt32	1024	3

4.2. Naïve Strategy to Data Alignment

In this production-equal setup, the recorded data is independently recorded throughout the smart factory. Furthermore, the dataset for each run consists of the log file of each SU plus the OPC-UA recordings. As a matter of fact, every run needs to be aligned. This misalignment needs to be considered within any downstream analysis. Each SU is started right after another and the timestamps between the SUs do not vary deeply. A strategy is only required for the OPC-UA log file. Our proposed naïve strategy is to use the timestamp of the first activity of the High-Bay storage as a starting point. Additionally, we subtract the travel-time of a shuttle from the found timestamp. A shuttle needs

about 30 seconds from order placement until arrival at the High-Bay storage. In the end, we use the date provided through the SU data to align the OPC-UA data and correct the timestamps correspondingly. This can be done because the OPC-UA recording is started seconds before the first order is placed. Another naïve strategy is to synchronize the timestamp by aligning the operational spikes between a particular SU and the corresponding actions in the OPC-UA recording of a station.

Table 2: OPC-UA Log File Structure with Examples

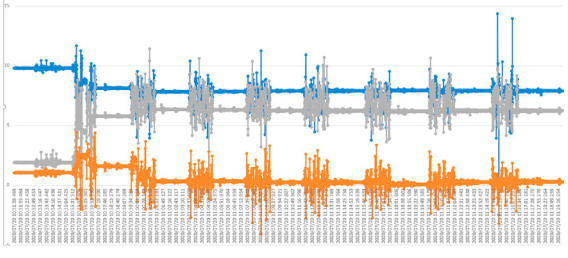
Columns	Data Type	Example Values
date	DateTime	2020/07/01 16:15:40:647
nodeid	String	”ns=6;s=Station 20 OpenC.St20 Soft-SPS.SAP_Fehlercodes_PCo_iDB.Start”
value	Values for Data Types Tab. 1	False
dtype	Data Types Tab. 1	Boolean

Table 3: Combined Log File Structure of the different Sensing Units with Examples

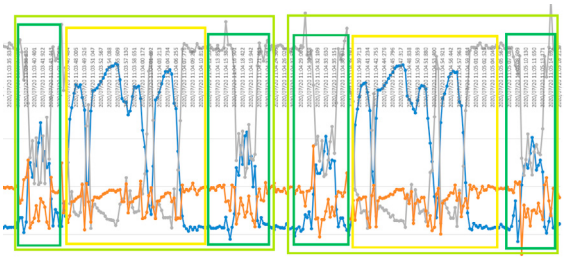
Columns	Data Type	Example Values
datetime	DateTime	2020/07/01 16:15:40:647
TCAp, sGRP	Integer	7, 0
TSL_IR, TSL_Full, TSL_Vis, TSL_LUX, BMP_TempC, BMP_Pa, BMP_AltM, MPU_AccelXMss, MPU_AccelYMss, MPU_AccelZMss, MPU_GyroXRads, MPU_GyroYRads, MPU_GyroZRads, MPU_MagXuT, MPU_MagYuT, MPU_MagZuT, MPU_TempC	Float	41.00, 256.00, 215.00, 29.47, 27.57, 99359.20, 87.82, 5.72, -6.28, -5.15, -0.00, -0.01, 0.00, 21.58, 29.34, -28.94, 30.40

4.3. Normal Operation

Normal operation is recorded to obtain the ground truth for all subsequent use cases. It consists of the data about eight successful tested relays and eight tested faulty relays. The recorded dataset is ~64 mb in size, which translates to ~490.000 log events. First, the high-bay storage places a pallet with components on a shuttle for each assembly job. Afterwards, the robot starts to assemble each relay. Figure 2a depicts the output of one sensor group of the corresponding SU. Furthermore, each of the activity blocks can be split into two parts. In the first cycle the robot picks up its tool and starts mounting all components on the pallet according to the deposited order. The second cycle may vary depending if a second shuttle awaits assembly, otherwise the robot will put back its tool and restart the procedure in case if a new shuttle with a new order arrives (Figure 2b). Next, the press will be activated and join the parts together. In the end, the electrical inspection will test all relays. Note that only the relays with a DMC, indicating a suitable testing routine, will be inspected in the end.



(a) Activity in robot-sensors (X/Y/Z-Axis, sGrp 1) normal production

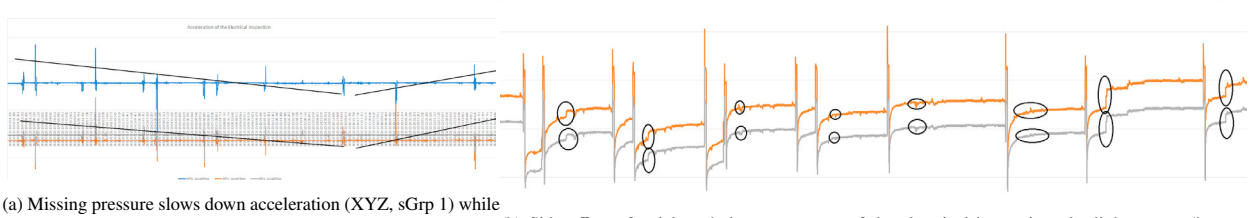


(b) Activity of two assemblies (yellow) and back-placing the tool four times (dark green)

Fig. 2: Assembly station operates normally

4.4. Missing Pressure

Each use case in common is that contextual faults are not recognized by the smart factory routines. But, with the employed SUs we can measure the side effects additionally to the OPC-UA data. As already discussed in Section 3.1.1 the process will be slowed down by missing pressure, causing interference in the production flow of the robot, the press, as well as the electrical inspection. The recorded dataset encompasses ~63mb of data, which translates to ~480.000 log events. The series of measurements started with a pressure of 4.0 bar for the electrical inspection and the assembly station. The press has its own pressure feed mechanism, which limits the global pressure to a station-internal pressure level of 3.0 bar at maximum. Every two assemblies the pressure is lowered. After the first two assemblies the pressure is lowered by hand to 3.8 bar pressure (the press remains at a pressure of 3.0 bar). Next, the loss of pressure after every two assemblies lies between 0.5-0.8 bar (the pressure of the press also starts to decrease, when the margin of the global pressure falls lower than 3.0 bar). We reestablish the pressure after the smart factory reached the minimum operation level of 1.9 bar. The minimum level was reached after ten assemblies. Figure 3 shows the effect of the low pressure to the electrical inspection machine. A downwards trend can be identified in the acceleration data (Figure 3a). Additionally, an unexpected change in the movement speed of the station can be recognized by the light sensors. As faster the inspection machines moves the quicker the light sensor will change between lighted and shadowed states, which can be seen as spikes in the data. While the movement slows down the spike tends to transform into a curve, because the sensor gets measured multiple times with a minor delta between measurements. As the re-establishment of the pressure takes place after the 10th assembly, the spike can be seen again. Figure 3 confirms also that not every of the sixteen relays is tested electrically (e.g. because of a faulty DMC).



(a) Missing pressure slows down acceleration (XYZ, sGrp 1) while operation. Trend is colored in black.

(b) Side effect of quick and slow movement of the electrical inspection, the light sensor (human visible light and full-spectrum, sGrp 1) adapts either more quickly or slowly in a curve

Fig. 3: Effects of low pressure to the inspection machine

4.5. Shuttle Drop Out

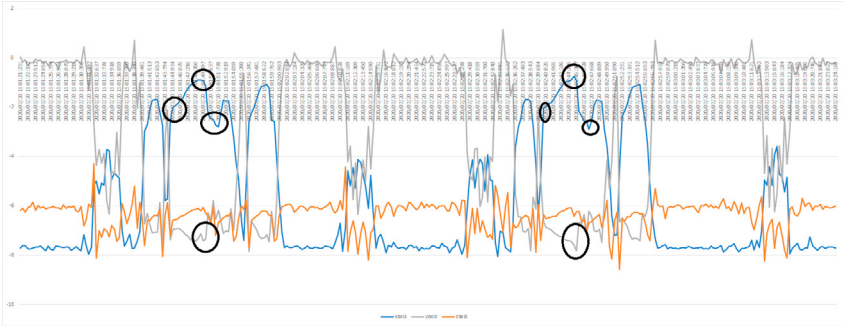
A shuttle drop out will decrease the production rate by a factor of two (Section 3.1.2). Consequently, the production time for the amount of relays is doubled. The recorded dataset is about ~74 mb in size, which equals to about ~565.000 log events. Figure 4 depicts the activity of the gyroscope that measures 3-axis in radients per second. Through the pressing process the sensors positioned at the press arm will shook and the gyroscope measures the radients per second per shake. The time between activity is doubled after the shuttle drop out and return to normal after the shuttle is queued back.

Fig. 4: Production during a shuttle drop out (gyroscope activity). Production output is cut in half.

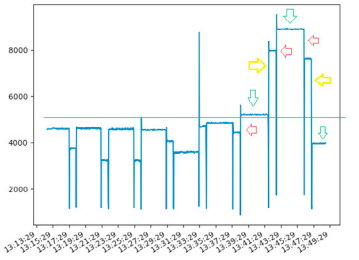
4.6. Missing Parts

The contextual fault of missing parts involves multiple stations to find the hint towards a reason of these anomalies. The recorded dataset is about ~63 mb, consisting of ~476.000 log events. Figure 5 shows the relationship between

the robot assembly station and the press to determine the missing parts. Meanwhile assembling, short spikes indicate an empty grip (Figure 5a), but for validation of the results, the information of e.g. the press is also needed. We were unable to validate the results using the SU of the press only, because the measured acceleration is not accurate enough to draw conclusions towards the three missing parts. Additionally, we use the OPC-UA data of the press, and its through OPC-UA propageted model, to conclude that each time a missing part is present the actual pressure level starts to rise (Figure 5b). The red arrows indicate the situations where no relay was under the press arm. The green arrows present the rising pressure levels, while the yellow arrows represent the situation where a relay was present under the press arm. In detecting the contextual fault of missing parts, the broken parts can be automatically flagged for redeployment to minimize waste and probably reuse the other parts of the pallet after the missing part is replaced.



(a) Missing parts robot assembly between 10th (not missing) and 11th (missing relay) assembly job. The difference is marked with black cycles



(b) Press OPC-UA data "M2 Ventil Ist-Druck" start rising with missing parts

Fig. 5: Effects of missing parts to the robot assembly station and the press station

5. Conclusion

While CPSs get more and more interconnected the chance of contextual faults will rise. We publish our CONTEXT-dataset (around 2 mio. log events) to show that contextual faults exist in the smart factory and need to be addressed in future research. Most of the industrial datasets currently available refer to individual machines and not to a complete production line. In contrary, the datasets that do refer to complete production lines are either anonymized or obscurified or focus on one or two features only. Therefore, sufficient data is not available. Anonymization is a major issue for valid scientific research, because neither the reason nor the cause can be named or tracked throughout the smart factory.

Our Industry 4.0 production-equal dataset is neither anonymized nor obscurified to gain explainability in experiments and encourage new findings. Additionally, we did a first preliminary manual analysis of the recorded contextual faults to present the complexity which contextual faults are offering. Furthermore, we developed SUs to save additional information about the process and its surroundings. The usage of SUs proved auxiliary during the first analysis, where the contextual faults cannot be detected without extra hardware. Consequently, the SUs help to supplement incomplete data or compensate either hidden or missing data. Through the production-equal smart factory the logs encompass out of order events, missing events or data types that are not accurately implemented. Missing events are introduced due to network latency and bottle necks during the heavy recordings throughout the smart factory. Nevertheless, to the best of our knowledge, there is no other production-equal dataset with the same advanced characteristics and data types for the Industry 4.0 smart factory domain with explainable faults.

Our preliminary analysis of the contextual faults scratched only on the surface of the dataset. Many trailing relationships in the data remain hidden, unravel opportunities for future research. We believe that our CONTEXT dataset may be useful in the areas of outlier, anomaly or novelty detection, development of virtual twins as well as machine learning and deep learning tasks. Furthermore, we want to encourage other researchers to publish their contextual faults in their smart factories. We conduct multiple studies that investigate contextual faults further in future. A study is planned to identify and extract contextual knowledge automatically. Another study will focus on visualization of contextual faults to the personnel. We plan to release each upcoming study with additional uncensored open data from

the smart factory. Along with this publication we publish our recorded dataset [11]. May our dataset help to unveil new research questions, test new algorithms and draw conclusion for the upcoming era of real fully-integrated smart factories.

Acknowledgments

This work was conducted within the research group on Human-Computer Interaction and Visual Analytics (https://vis.h-da.de). The presentation of this work was supported by the Research Center for Applied Informatics.

References

[1] Agogino, A., Goebel, K., 2007. Milling Data Set. Moffett Field, CA. URL: http://ti.arc.nasa.gov/project/prognostic-data-repository.
[2] Alos, A., Dahrouj, Z., 2020. Detecting contextual faults in unmanned aerial vehicles using dynamic linear regression and k-nearest neighbour classifier. Gyroscopy and Navigation 11, 94–104. doi:doi:10.1134/S2075108720010046.
[3] Bandeira de Mello Martins, Pedro, Barbosa Nascimento, V., de Freitas, A.R., Bittencourt e Silva, P., Guimarães Duarte Pinto, R., 2018. Industrial Machines Dataset for Electrical Load Disagreggation. IEEE DataPort. doi:doi:10.21227/CG5V-DK02.
[4] von Birgelen, A., Buratti, D., Mager, J., Niggemann, O., 2018. Self-organizing maps for anomaly localization and predictive maintenance in cyber-physical production systems. Procedia CIRP 72, 480–485. doi:doi:10.1016/j.procir.2018.03.150.
[5] Bonatakis, J., Chokor, A., Propes, N., 2018. PHM Data Challenge 18. Philadelphia, Pennsylvania, USA. URL: https://www.phmsociety.org/events/conference/phm/18/data-challenge.
[6] Dua, D., Graff, C., 2017. UCI machine learning repository. URL: http://archive.ics.uci.edu/ml.
[7] Goebel, K., Sandborn, P., et al., 2009. PHM09 Challenge Data Set Gearbox. San Diego, USA. URL: https://www.phmsociety.org/competition/PHM/09/apparatus.
[8] Joshi, J., et al., 2016. 2016 IEEE International Conference on Big Data Manufacturing Data Challenge: Production Line Performance: Reduce manufacturing failures (Robert Bosch GmbH). IEEE, Piscataway, NJ. URL: https://www.kaggle.com/c/bosch-production-line-performance.
[9] Kaupp, L., Beez, U., Hülsmann, J., Humm, B.G., 2019. Outlier detection in temporal spatial log data using autoencoder for industry 4.0, in: MacIntyre, J.D., Iliadis, L.S., Maglogiannis, I.G., Jayne, C. (Eds.), Engineering applications of neural networks. Springer, Cham, Switzerland. volume 1000 of *Communications in Computer and Information Science*, pp. 55–65. doi:doi:10.1007/978-3-030-20257-6_5.
[10] Kaupp, L., Nazemi, K., Humm, B.G., 2020a. An industry 4.0-ready visual analytics model for context-aware diagnosis in smart manufacturing (in press), in: Proceedings of the 24th International Conference Information Visualisation (IV). IEEE, pp. 337–346. doi:doi:10.1109/IV51561.2020.00064.
[11] Kaupp, L., Webert, H., Nazemi, K., Humm, B., Simons, S., 2020b. The context dataset containing contextual faults of a smart factory. doi:doi:10.5281/ZENODO.4034867.
[12] Kolen, S., Otten, J., 2015. FINESCE Data Repository in FIWARE Lab Data. Aachen, Germany. URL: https://data.lab.fiware.org/dataset/smart_energy_data-_aachen__cologne_smart_factory.
[13] Lee, J., Qiu, H., Yu, G., Lin, J., 2007. Bearing Data Set. Moffett Field, CA. URL: http://ti.arc.nasa.gov/project/prognostic-data-repository.
[14] Magalhães Oliveira, E., 2007. Quality Prediction in a Mining Process: Explore real industrial data and help manufacturing plants to be more efficient. Av. Marginal, 156, Jaguariúna – SP - CEP 13820-000, Brasil. URL: https://www.kaggle.com/edumagalhaes/quality-prediction-in-a-mining-process.
[15] McCann, M., Johnston, A., 2008. SECOM Data Set: Data from a semi-conductor manufacturing process. URL: https://archive.ics.uci.edu/ml/datasets/SECOM.
[16] Niggemann, O., Schüller, P., et al., 2018a. IMPROVE project. One Year Industrial Component Degradation: Degration of a cutting blade. Lemgo, Germany. URL: https://www.kaggle.com/inIT-OWL/one-year-industrial-component-degradation.
[17] Niggemann, O., Schüller, P., et al., 2018b. IMPROVE project. Production Plant Data for Condition Monitoring: Prediction of the condition of an important component. Lemgo, Germany. URL: https://www.kaggle.com/inIT-OWL/production-plant-data-for-condition-monitoring.
[18] Saxena, A., Goebel, K., 2008a. PHM08 Challenge Data Set. Moffett Field, CA. URL: http://ti.arc.nasa.gov/project/prognostic-data-repository.
[19] Saxena, A., Goebel, K., 2008b. Turbofan Engine Degradation Simulation Data Set. Moffett Field, CA. URL: http://ti.arc.nasa.gov/project/prognostic-data-repository.
[20] Schneider, T., Klein, S., Bastuck, M., 2018. Condition monitoring of hydraulic systems Data Set. Eschberger Weg 46, 66121 Saarbrücken. URL: https://archive.ics.uci.edu/ml/datasets/Condition+monitoring+of+hydraulic+systems.
[21] Seabra Lopes, L., Camarinha-Matos, L.M., 1999. Robot Execution Failures Data Set. Monte da Caparica, Portugal. URL: https://archive.ics.uci.edu/ml/datasets/Robot+Execution+Failures.
[22] Simons, S., Abé, P., Neser, S., 2017. Learning in the autfab – the fully automated industrie 4.0 learning factory of the university of applied sciences darmstadt. Procedia Manufacturing 9, 81–88. doi:doi:10.1016/j.promfg.2017.04.023.