Chair of Automation

Doctoral Thesis

Concepts, Methods, and Systems for Machine Data Analysis

Dipl.-Ing. Christopher Josef Rothschedl

May 2020

MONTANUNIVERSITÄT LEOBEN
www.unileoben.ac.at

AFFIDAVIT

I declare on oath that I wrote this thesis independently, did not use other than the specified sources and aids, and did not otherwise use any unauthorized aids.

I declare that I have read, understood, and complied with the guidelines of the senate of the Montanuniversität Leoben for "Good Scientific Practice".

Furthermore, I declare that the electronic and printed version of the submitted thesis are identical, both, formally and with regard to content.

Date 01.05.2020

Signature Author
Christopher Josef Rothschedl

Acknowledgements

First of all, I would like to express my deepest appreciation and gratitude to Paul O'Leary, for his dedication and all the fruitful discussions and illuminating contemplation of ideas and theories. The time spent together on projects was invaluable and the insights gained contributed significantly to how I was able to approach the challenges of my work. Paul played a decisive role in where to focus my professional work and how to utilise the concept of *welcome everything, push away nothing*. Thank you for being an inspiration and for being an integral part of my path.

I would also like to extend my deepest gratitude to Peter Lee, Eva Maria Braun and Alexander Zöhrer, who supported this work significantly by contributing feedback and constructive criticism. Without your honest assessment, comments, and suggestions, the work would not have been of the quality I pursued for it.

A special thank you goes to all my peers and colleagues from the Chair of Automation, Keller, as well as from Sandvik and FLSmidth. A thank you goes to Andreas Körbler, not only but also for hinting that the University of Leoben might be a good fit for my interests. I am glad to have had the chance to gather so many inspiring characters around me and to learn from so many different viewpoints.

My heartfelt appreciation and thankfulness go to my closest family, who always supported my work and intentions. Words alone can never express my gratitude for my fortune of having such inspiring and lovely people around me. You all contributed significantly to my open-mindedness and endurance, which let me pursue my interests and goals with passion. Thanks for bearing with me and my ideas.

Abstract

This thesis is concerned with the means of acquiring data from cyber physical systems, including the required infrastructure, and the use of both classical and novel approaches to derive insights from time series data. The systems discussed are associated with industries which require the establishment of complete data management frameworks to integrate essential domain expertise to gain knowledge.

In a generic manner, the development of a holistic, secure and flexible concept for data life cycle management is presented, for heterogeneous fleets of mobile machines. Apart from providing robustness, such a concept must be capable of adopting changes, such as extensions with subsystems or replacements of components resulting from obsolescence of technology. After a study of the fundamental requirements for hardware, machine interfaces, data handling and storage, as well as data provisioning, an implementation is shown for different machines used in the mining and materials handling industry. To take the insights from these implementation scenarios into account, the concept for a new system for machines of the geotechnical engineering sector is developed and implemented.

The framework for a qualitative data flow is presented, which allows experts to interact with the data of their machines; a step necessary to create added value from time series. It consists of multiple levels of data preparation and presentation methods, to identify elements of interest. Outliers can be highlighted for further investigations, based on rules applied to key performance indicators.

Furthermore, it is shown that true knowledge discovery can be supported significantly by mimicking the mechanisms of the emergence of natural language. This step takes the specific nature of the data into account, since the time series emanate from physical systems and, hence, must abide by the laws of physics. An exemplary evaluation performed in this manner reveals implicit hierarchical structure in the operational data. Only an initial set of language elements are defined as input for a subsequent iterative process. A hierarchy of compounded frequent elements is yielded, the top layer of which reveals the existence of two major sequences that correlate with the two main operation modes. It is shown that the interpretation of the results by domain experts is indispensable for knowledge gain. This is emphasised by the metaphorical capacity exhibited by language-affine evaluation approaches, which are discussed in detail. A model for the emergence of language, based on phenomenological aspects, is proposed to combine the factors of relevance for knowledge discovery.

Index Terms

Cyber Physical Systems; Mining and Materials Handling; Geotechnical Engineering; Domain Expertise; Data Science; Knowledge Discovery; Symbolic Time Series Analysis; Natural Language; Hierarchical Structure; Implicit Structure; Phenomenology

Kurzfassung

Die vorliegende Arbeit handelt von der Erfassung von Daten von cyber-physischen Systemen und der dafür notwendigen Infrastruktur für in dieser Hinsicht unterentwickelte Anwendungsgebiete. Darüber hinaus werden konventionelle sowie neuartige Herangehensweisen zum Erkenntniszugewinn diskutiert.

Die Entwicklung eines profunden, sicheren und flexiblen Konzeptes für die vollständige Abbildung des Datenflusses wird in allgemeiner Form für mobile Maschinen dargestellt. Dabei wird im Detail darauf eingegangen, welche Merkmale und Besonderheiten zu beachten sind, um eine funktionierende Interaktion zwischen Zeitreihen und Fachwissensträgern sicherzustellen. Das entwickelte Konzept muss dabei einerseits belastbar und zuverlässig sein, andererseits muss es auch in der Lage sein, neuartige Komponenten oder Untersysteme zu integrieren, die aufgrund von Erweiterungen oder der Etablierung neuer Technologie notwendig werden. Nach eingehender Betrachtung grundlegender Anforderungen an die Hardware, Maschinenschnittstellen, Datenverarbeitung und -speicherung, sowie auch an deren Zurverfügungstellung, wird eine Implementierung auf verschiedenen Geräten der Bergbau- und Fördertechnik-Sparte beschrieben. Erkenntnisse daraus werden für weitere Implementierungen in der Geotechnik herangezogen.

Weiters wird ein qualitatives Datenflussmodell beschrieben, welches Fachexperten mit Maschinendaten interagieren lässt. Das Modell, bestehend aus mehrstufigen Datenvorbereitungs- und Visualisierungsmethoden, unterstützt bei der Identifizierung von Elementen über eingesetzte Regelsätze, die sich wiederum auf definierte Leistungskennzahlen beziehen.

Der letzte Teil dieser Arbeit beschäftigt sich mit der Frage, welchen Einfluss Mechanismen der natürlichen Sprachentwicklung auf Wissenszugewinn im Rahmen von Zeitreihenanalysen haben können. Dabei wird die besondere Beschaffenheit dieser Daten in Betracht gezogen, da die Daten von Maschinen kommen, die physikalischen Grundgesetzen folgen. Bei einem Versuch werden wenige, grundlegende Sprachelemente einer symbolisierten Form der Zeitreihen eines Bergbaugerätes verwendet, um in iterativer Weise durch Zusammenfassen der häufigsten Elemente implizierte, hierarchische Struktur nachzuweisen. Jene Struktur lässt auf zwei Hauptsequenzen schließen, welche sich wiederum mit den beiden vorhandenen Betriebsmodi decken. Es wird festgestellt, dass die Interpretation der Ergebnisse durch Fachexperten maßgeblich für Wissenszugewinn ist, da die sprachaffinen Assoziationen metaphorischen Gehalt besitzen, dessen Berücksichtigung von Relevanz ist. Zum Abschluss wird ein Modell mit der Absicht präsentiert, die phänomenologischen Aspekte für die Entstehung von Sprache aus Zeitreihen zu beschreiben, um die relevanten Einflussfaktoren für Wissenszugewinn zusammenzuführen.

Schlagwörter

Cyber-physisches System; Bergbau und Fördertechnik; Geotechnik; Phänomenologie; Anwendungsfachexpertise; Datenwissenschaft; Wissensforschung; Natürliche Sprache; Symbolische Zeitreihenanalyse; Hierarchische Struktur; Implizite Struktur

Contents

List of Tables

List of Figures

Chapter 1

Introduction

1.1 Background, Context and Motivation

When Edwin Beard Budding, a British engineer, invented the lawn mower in 1830[1], he probably was not aware of what disruptive impact his invention would have. The prerequisites of sports such as football, rugby or golf rely heavily on lawnmower technology. Budding's invention introduced a paradigm change that benefited the establishment of such multi-billion or even trillion euro sport industries.

Initiatives of digitisation are a modern analogy to the lawnmower invention, they catalyse changes in antiquated paradigms. Dell Inc. revolutionised contemporary supply chain management by introducing a direct sales channel with consumers in connection with a configure-to-order approach. Since their establishment, online banking services contribute significantly to the gaining of momentum of modern societies. Netflix brought classic movie rentals to an end by inaugurating device-independent, on-demand streaming services. The advent of cloud computing facilitates that shift from a conventional way of working to novel approaches by putting emphasis on services rather than on products: users do not need their computer any more to work and to collaborate, they just need *any* device with internet connection. In general, software and computing power became cheaper than material and labour; a fact many new businesses are built on, such as Heliogen, a company operating in concentrating solar power. Hundreds of mirrors are angled to reflect sunlight to a central tower to produce significant levels of heat. Simple and easy to replace mirrors are used instead of highly sophisticated systems; relatively cheap software is used to overcome the issue of expensive material and labour [3]. As well as all these examples bearing the potential of individual, lawnmower-like advances in modern technology, there is another far-reaching field that drives technical progress: data analysis and data intelligence. In many fields of application, such as the development of physical

[1]Edwin Beard Budding from Stroud, Gloucestershire, England, invented the lawn mower in 1830 according to `https://www.oldlawnmowerclub.co.uk/aboutmowers/history`. The US patent USRE8560E from 1879 states on the first page, column two, first paragraph, that Mr. Budding held an English patent with No. 6080, dated August 31st, 1830 [1]. To appreciate and commemorate the impact of his invention, the Golf Course Superintendents Association of America give the annual Edwin Budding award to innovative engineers who help shaping the future of the turf equipment industry [2].

systems, e.g., in the solar power plant described previously, there is often an extensive phase of data evaluation required before innovations can be driven forward. An understanding of how processes work in detail and how a system responds to certain scenarios is vital for the development of novel structures.

For the work presented in the subsequent chapters, it was found that time series emanating from machines with human operators are of particular interest to different industries: mining, bulk materials handling, and construction. Human-operated machines are different from fully-automated machines, since humans are equipped with an entire set of sensors, which additionally sense the environment outside the machine sensor systems, which only observe directly process-related measures. The patterns of these human mechanisms are buried in operational data and might be different from those of fully-automated operation procedures. Three points of interest are identifiable when data from mobile machines is to be evaluated: availability of data, integration of domain expertise, and identification of causal relationships.

Secure and Reliable Availability of Data

A reliable and robust data acquisition infrastructure has to be established prior to any endeavours in the data analytics domain. Regardless of which system and technology is being adopted for this task, it is important that state-of-the-art security measures are undertaken. In the hype of creating something new, certain security aspects are often not addressed in an appropriate manner. This is also often true for how we deal with security in our private lives. For instance, the official Spanish football league, LaLiga, distributed an application for mobile phones for providing football-related information. What most users have not been aware of, was that this piece of software recorded the surrounding noise and was able to identify, whether a football match is being watched. Together with the geolocation feature of the mobile phones, this information has been used to identify bars and public places, where football matches were illegally streamed. The interesting part of this story is the fact that LaLiga explains in their terms of service that the mobile phones the application is installed on are being used to detect fraudulent behaviour. Users confirmed permission for the app to use the phone's microphone and geolocation when it was first started up after installation. [4]

Beside such a disputed way of handling private security, we also need to consider where our data comes from and if we can trust acquired data or its source. Let's have a look at the following example: Simon Weckert, a German artist, wheeled a pull wagon full with 99 mobile phones through the streets of Berlin. All of the devices had Google Maps running. Every street he walked along was indicated as traffic heavy in Google Maps' live traffic prediction functionality, resulting in other Google Maps users being rerouted to avoid the "traffic jam". From a technical point of view, this is a pointer towards how important the quality of the input data is. In an attempt to determine the validity of the input data, we can have a look at the average accuracy of a GPS[2]-enabled mobile phone under open sky conditions, which is a radius of 4.9 m [5]. This accuracy is halved to result in a radius of 9.8 m to compensate for environmental influences on the accuracy. This is done although the video of Mr. Weckert shows him walking through streets where

[2]GPS . . . Global Positioning System

a good GPS connection can be expected. We end up with 99 GPS coordinates, which are distributed within a circle of an approximate diameter of 20 m. It is obvious that 99 vehicles cannot fit in this projected area, especially when they all move at the same time with the same speed into the same direction – also on streets with only one lane in each direction. [6, 7]

The work of Mr. Weckert showcases, how important the input quality of such systems are. Users should not care about the plausibility checks for the ingress of data. However, they rely on these systems to perform their tasks in a proper manner. It is up to the system designers to ensure a sufficient and transparent input validation, prior to any subsequent processing or even provisioning of data and information. This needs to be taken care of for all newly established systems, also – and especially – for data acquisition infrastructures of mobile machines.

Connecting Domain Expertise to Data

In their industry digitisation index, McKinsey assesses the individual rates of digitisation of several industries in a roadmap, as showcased in Figure 1.1. The mining sector exhibits many deficiencies in terms of how a digital mindset can be established across the given metrics. What is even more obvious in this illustration is the position of the construction industry: it is amongst the least digitised. Although this fact is based on many insufficiencies across the used set of metrics, it also implies fertile soil for optimisation and the establishment of initiatives of a digital agenda.

The roadmap showcases the dependencies of the individual sectors when it comes to their digital capabilities. To increase the respective levels of digitisation, domain expertise is required: a specialist of the ICT domain (Information and Communications Technology) might not be of significant support in establishing a digital mindset in the construction industry. Domain experts can additionally act as influential opinion leaders of their sectors and help to collectively push viewpoints amongst their peers, also through their interest groups. The success of running digital initiatives within an industry can be directly dependent on whether domain experts are adopting such new technology.

For the mining and materials handling industry, results of previous work has exhibited the need of involving domain expertise [9]. The conclusions of many evaluations bear out potential of being of poor quality or being based on misinterpretations, which result from insufficient domain expertise. A considerable challenge of endeavours in data analysis of time series from mobile machines is to connect concepts of data science with the knowledge of domain expertise. Without this link, much content and insight remains undiscovered or is falsified by improper interpretation.

Figure 1.1: **McKinsey Global Institute Industry Digitisation Index:** Mining as an industry exhibits many deficiencies in digital metrics, there is much potential for improvement. The construction sector ranks even worse, amongst the least digitised of the list. [8]

Identification of Causal Relationships

In his book, “Spurious Correlations – Correlation does not equal Causation” [10], Tyler Vigen compares the trend of civil engineering doctorates awarded in the United States with the numbers of per capita Mozzarella cheese consumption of Americans. The correlation between both trends is found to be at 95.6 %. The statistical correlation is correct, however, this example clearly misses a causal relationship. In another, more technical, example, an exothermic system with high activation energy is considered: the exothermic model needs to be included, if causality is to be established, since correlation alone will lead to erroneous interpretations. [11]
It is important to identify causal relationships in data sets to derive meaning and to extend the statistical representations. Metaphors can be of fundamental support in attributing additional meaning and to express nuances.

1.2 Contributions

The main contributions of this thesis are:

- The conception and establishment of secure, reliable, and robust data flows for two individual industries to provide data to subject matter experts, whereby the data originates from mobile machines of different types that are operated in remote locations around the globe. One of the key challenges is to handle the significant amount of time series and to provision it to the *right* entity in the *right* environment in the *right* format at the *right* time. The implementation is to be shown for the mining and materials handling industry, whereas the concept for geotechnical engineering, which is to be discussed afterwards, considers insights from the previous implementation.
- The conception and installation of a systematic framework for exploratory data analyses to interconnect domain experts with data to attribute meaning to evaluations and its results to gain fact-based insights.
- The investigation of an approach to mimic mechanisms of natural language when evaluating time series data sets emanating from machines operated by humans and to identify causal relationships instead of or in addition to sole (statistical) correlations. Emphasis is put on the unsupervised detection of structure in the data.

The thesis consists of two main themes: one about a sufficient data availability framework and the other one about novel approaches in evaluating data.
The first part focuses on the whole framework for data availability, which is a significant portion of development in its own right. Only when having such framework available, it is possible to look at new ways of evaluating data: the attention for the second part lies on unsupervised detection of structure, which is important since learning approaches currently have issues with unsupervised forms of data evaluation. Having the data availability framework ready enables completely new approaches to gain insights from the exploration and discovery of structure within time series emanating from the machines and plant discussed.

1.3 Outline

The work presented is structured into three main chapters, which reflect the contributions listed previously. The first chapter will portray in general how infrastructures are established to make data available from mobile machines. Implementations are then showcased for two different industries. Since both businesses come with individual requirements, environments and boundary conditions, the chapter is later split into two sections, one for each industry. The next chapter covers the investigation of an approach to exploratory data analysis using an example from the geotechnical engineering sector of the construction industry. The last chapter discusses how phenomenology and the mechanisms of language can be of benefit when evaluating time series from human-operated machines. Additionally, the significance of metaphors are examined in the context of this work. The closing chapter summarizes the insights of the work and comments on potential future work and extensions.

1.4 Author's Notes

Parts of the presented work have already been peer-reviewed and published; the contributions of the author that are of relevance to this thesis are attached as appendices. The original numbering of the pages of these papers has been altered to fit into the format of this thesis: the figures, tables, and references remain in the original format of the publications. However, they are not listed in the list of tables, in the list of figures or in the bibliography, unless they are cited again in the text of this thesis.

All illustrations used from other works are cited with a respective reference in the caption. If referenced figures have been altered for improved readability or to emphasise certain aspects of relevance, they are marked with an extra *cf.* next to the citation.

Illustrations not specifically cited or attributed with a designated source indication have been created and prepared by the author himself. He holds the original native files and copyright. Some of the illustrations have been also provided to his working group at the Chair of Automation. Photographs without any stated reference have been taken by the author himself with the consent of the owner or respective right holder of the pictured object(s).

In this work, not all trademarked names may be used with the respective trademark symbol. These names are used in a technical context only, without the intention of trademark infringement.

Chapter 2

Infrastructure and Data Availability

2.1 Introduction

This first chapter of the presented work focuses on the requirements for having data available. Here, the term "data" refers to time series emanating from machines operated by humans. Undeniably, discussing the means of acquiring, transferring and provisioning data is linked to the question of *what is a valid source of knowledge*. Since this question, its interpretation and attempts to answer it are of significant importance for what we want to achieve with data, it is being elaborated in chapter 4.

Many globally acting companies in the construction and mining industries have their machine fleet operated in different locations, often with varying and demanding environments. The collection of data for such machines involves more effort and obstacles in contrast to data acquisition for production plants, where pieces of equipment are clustered at a specific location. There are also challenges from an user perspective: numerous user groups have different interests, intentions and skill levels. Hence, it is significant to ensure that the right data is made available in the right format to the right user at the right time. A newly developed concept is required to take care of all the presented subjects and needs to be as flexible as possible to accommodate further advances in functionality and adopt or interface with new technology as well. Additionally, such system must meet the specifications and requirements exhibited by the area of application.

An overview of the contents of this chapter is illustrated in Figure 2.1. How data can be acquired from the existing *data sources* and how these gained time series can be stored and provisioned in a manner, which adds value for the users and insight consumers, is discussed. All parts of the infrastructure and data availability scheme exhibit individual as well as systematic security features, since a secure environment for the whole data flow is integral to the holistic concept.

In the following sections, points of interest for developing such systems are discussed in a generalised manner (see also concepts described in [9]), before developments in two specific industries are elaborated: mining/bulk materials handling and construction/geotechnical engineering. Work for the latter has been started at a later point in time; experience from the development of the first system has been incorporated.

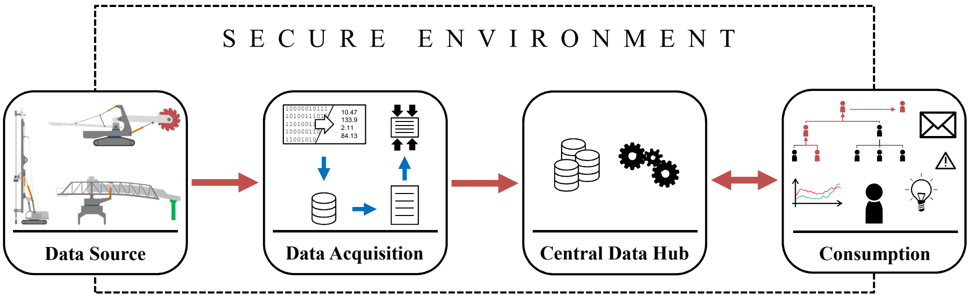


Figure 2.1: **Chapter Overview:** This chapter focuses on sources of data, the means of acquiring data, the requirements for a central data hub as well as how data and derived insights can be used and consumed. Additionally, the fundamental premise of the whole infrastructure and data availability scheme follows a holistic approach that spans a secure environment over it.

2.2 Security

Security is integral to any holistic concept of data acquisition, handling and provisioning. Although the presented work does not particularly focus on the security part, a certain level of awareness is necessary and needs to be raised. Security is not considered an add-on that can be taken care of later, it needs to be part of the initial design considerations already. Alone between 2014 and 2019, the number of notable security incidents in the construction industry increased by factor twelve, while the mining sector exhibited a number four times higher than in 2014. Additionally, also the count of breaches is four times higher for the construction sector [12, 13]. With OT[1] becoming more important in state-of-the-art settings, protection of all devices of an interconnected network, as well as of the network itself, turns out to be mandatory for a system to run reliably and securely. Specifically, in the mining and construction sectors, the machines and the associated data acquisition units are operated outside the operating company's premises. The units are out in the field and connected to public networks, i.e., the internet. Hence, there exist additional system vulnerabilities.

To develop a secure system in the IoT[2] domain, it is of fundamental significance to understand potential threats. According to Siemens, main threads to be considered are as follows [14]:

1. Reducing availability, e.g., via Denial of Service (DoS) attacks;
2. Man-in-the-middle attacks – Circumvention of specific security mechanisms;
3. Intentional maloperation through permitted actions, e.g., password or identify theft;
4. Corporate espionage;
5. Manipulation or falsification of data, e.g., to decrease importance of alarms or notifications;
6. Deletion of operational data and log files, also to potentially delete traces of a cyber attack.

[1]OT . . . Operational Technology
[2]IoT . . . Internet of Things

In addition to this list, support to identify risks in a generalised manner can be found in the interdisciplinary concept of the *CIA Triad*[3], consisting of *Confidentiality*, *Integrity*, and *Availability* [12, 13, 16]. An approach to define the triad terms for vulnerabilities in the context of the presented work can be:

Confidentiality: Data can be compromised, data can be stolen. This can potentially happen on the data acquisition device, at the centralised data hub, or during data transfer via (public) networks.

Integrity: The system, or parts of it, bear the potential of being manipulated. Data can be modified to intentionally feed wrong data into the central data hub. If data needs to be altered or processed, it is required to do this in a traceable manner.

Availability: Once a system is operative and well accepted among the users, it becomes essential for performing tasks reliably. A vulnerability is given if the system as a whole, or any of its services, are not available in an unrestricted manner. This is also true for time-critical communication and notifications when they are suppressed. Additionally, missing data is to be avoided.

An additional requirement for such systems is the **Authority** part, which is sometimes implicitly mentioned when confidentiality is discussed: for a system, it needs to be ensured that only authorised users have access [17]. Based on the individual clearance level, different users have diverse access rights, which need be administrated and maintained. The *Federal Office for Information Security of the Federal Republic of Germany* summarises the security requirements for projects in the IoT domain with these objectives [18]: *security by design*, *security in deployment* (integration and individual modifications), and *secure operation*. Support to implement those essential requirements is given in the guideline *VDI/VDE 2182 – IT-Security for Industrial Automation* [19]. This set of instructions describes a process model to implement appropriate security measures for a project in the IoT domain. The approach is process-oriented and puts emphasis on the entire life cycle of such a project, which also involves the collaboration between vendors, integrators and asset owners. Even if all the three stakeholders would be of the same company, the guideline can be of support to identify action points, create checklists and execute mandatory security-related initiatives.

In addition to the above, the paradigm of *Low-Code* platforms can be of supplementary support to reach a more secure system. Low-Code environments offer a clearly defined and prepared development environment for pieces of software, in IT as well as in OT. The fundamental premise is to ensure that developers focus on the usage of already implemented and tested code parts and functions to significantly reduce the amount of custom code. This does not only ensure a lower effort required to build out certain pieces of software, it also decreases the creation of additional vulnerabilities or bad-practice solutions, since the core functionality of the re-used implementations was already tested thoroughly. [20]

[3]Although the original source for the concept of the *CIA Triad* seems to be not identifiable with appropriate certainty, the underlying concepts of it were already of importance millennia ago, in a military context, as can be found in the commentaries of Julius Caesar in *Commenttarii De bello Gallico* [15].

2.3 Sources of Data

Prior to any data collection there have to be physical phenomena identified, which one wants to observe, and which one wants to acquire data for. Such observations can be accomplished by using sensing systems – sensors – that convert a physical process, e.g., a temperature or inclination angle, into an electrical signal. This signal is captured by a control system, usually a *programmable logic controller (PLC)*, either by accepting the electrical signals directly or by receiving a message on a field bus. The latter would imply the sensor's capability of converting the electrical signal, induced by the observation, to a message containing the value, which is being send on a field bus, e.g., a CAN[4]-based protocol. In this case, the message can be received and interpreted by the PLC and the value can be used for further processing on the CPU[5]. Otherwise, the electrical signal is converted by a analogue-digital converter to yield a value, a number, processable by the CPU. The same concept works in the other direction when CPU commands are send to actors, e.g., hydraulic valves or electrical drives. Before the control values are received by the actors, they are processed by the CPU, which implies that these values are available as well as the sensor signals. All current system values are therefore known for each read/write cycle of the PLC.

Since many modern machines are designed to be controlled with programmable logic controllers, data can be acquired by connecting to the particular PLC(s). A connection between them and a data acquisition unit can be established via field buses or Ethernet-based buses. PLC data is accessed by those units via the selected bus. This results in having data available in a number format for data collection. However, additional data is required to add layers of information: metadata [9]. Potential forms of metadata include, but are certainly not limited to:

- Channel names;
- Units;
- Value ranges (valid, warning, and error ranges);
- Sensor and actor specifics, e.g., cylinder geometries;
- Spatial data, such as GPS locations;
- Project and machine information.

If data is read from an existing machine PLC, it will be necessary to preserve the absolute timestamp of the original controller or the relative deviation in comparison to the master time of the data acquisition unit. NTP[6] services running on both devices, the machine controller and the data acquisition unit, can be used to ensure the times are synchronised with the appropriate precision.

[4]CAN . . . Controller Area Network
[5]CPU . . . Central Processing Unit
[6]NTP . . . Network Time Protocol [21]

2.4 Data Acquisition Units

To collect data from any source available, data acquisition units are required. The specification profiles of these devices vary much, depending on their area of application and intended use. In some scenarios, those devices also host local applications and inherit other functions, e.g., visualisation or user interaction and support tasks.

Data acquisition units are required to run reliably and, depending on the application, need to run without mandatory user interference. They must be able to accept data via bus protocols or from sensors directly connected to the data acquisition devices; also hybrid versions – combining both options – are reasonable for many cases.

A data forwarding mechanism is part of the device as well, since the collected data needs to be transported to a (centralised) data collection hub. This can be accomplished by transferring data via portable storage media or, more conveniently for most applications but not for all operating conditions and environments, by transferring data via a network connection to the collection hub using private or public networks. The latter can be accomplished by transferring files containing acquired data, resulting in intermittent data sets. However, data transfer via network connection bears the potential of streaming data channels as well. Depending on how the collected data is being processed, this may or may not create benefit. Intermittent data sets can be stitched together at the data handling location if required for further processing. The means of how data is transferred to data handling hubs is entirely separated from how a data acquisition system is communicating with the original data source, i.e., a PLC. The latter communication should be active as soon as both devices are in a run state. It is a benefit to have the data acquisition unit booting faster into an operational state to cover all the values from the entire online time of the associated machine controller.

It is reasonable for many applications to design a data acquisition device with the capability of executing edge computing tasks. Especially when looking at the technologies expected to keep research and development entities busy for the upcoming years, it seems that *Edge Analytics* play a major role, see Figure 2.2. Devices bearing the potential to cope with the required computing power, resources and security mechanisms can be used for additional edge analytics tasks besides data acquisition. This is of particular interest when a data acquisition unit is operated in an environment without a stable network connection: safety-related evaluations required during execution can run on the unit, as well as calculations necessary for monitoring production quality.

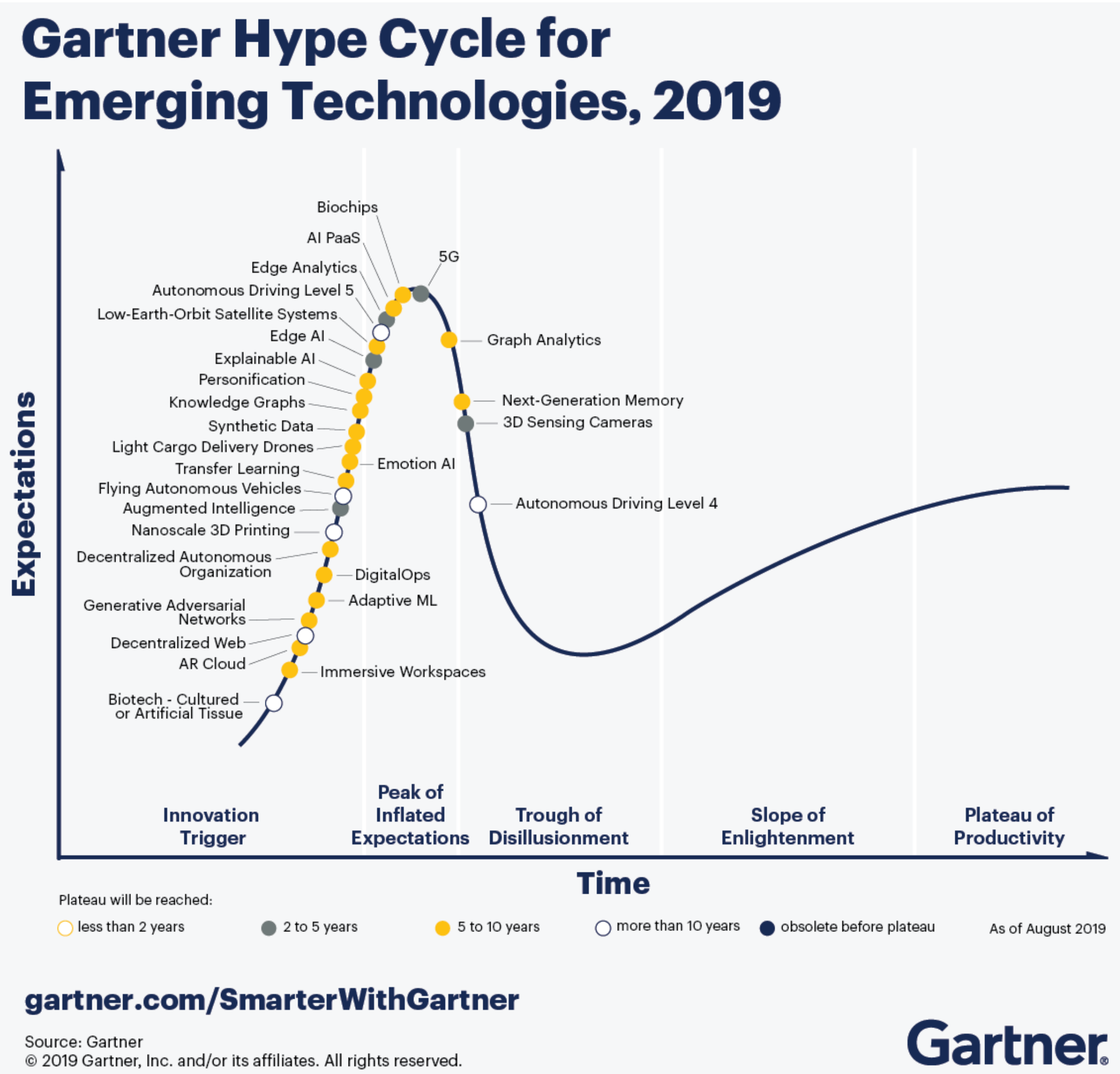


Figure 2.2: **Gartner Hype Cycle for Emerging Technologies 2019:** *Edge Analytics* just entered the phase *Peak of Inflated Expectations*. It is expected that the technology development will be concentrated for the next 2 to 5 years, according to Gartner's analysis. [22]

2.5 Data Handling and Storage

If a data acquisition unit provides data in a file format to a centralised data hub, the file needs to be run through an ingestion process to verify it is from a valid source, its sufficient quality, its authority, and its integrity. Given a file with sufficing content, it is forwarded to automatic pre-processing services. If the file does not meet the specified criteria, it is quarantined and a designated user needs to check it manually, before it can go through the ingestion process again or can be marked for removal.

After ingestion, the file itself is archived and its content is stored in a database. If the collected data is not present as values of a uniform sampling interval[7], the data needs to be restored to exhibit uniform time distances between the sampling points. This is important for further processing using the methods developed and available at the Chair of Automation.

Computations can be triggered automatically, e.g., force calculations based on cylinder pressures or statistical evaluations. The results obtained are also stored in a database. Additionally, separate metadata is merged with the file content and both are stored in a combined manner: for example, when sampled trace data comes in as numbers and the channel names are available as metadata from a separate source only, the data is merged and stored in a format where the channel names label the corresponding column or field of values.

It is also necessary to have several storage containers or databases, based on the necessity of access. Data of older projects or machines are more likely to be accessed less frequently than data from a current project or from a heavily utilised machine.

2.6 Data Provisioning

There are many interest groups for the data collected by a centralised data hub. However, the availability and simple access to it is vital for its acceptance among the users. Many different use cases are present, a selection is listed here:

R&D (Data Science)	demands access to all data available;
Regional Mangement	is interested in having an overview of all the machines and/or projects of a region;
Project Management	needs insights on a particular project;
Equipment Management	requires information about the entire machine fleet;
Design and Engineering	uses data of selected projects or machines to fulfil expectations and to improve machines and processes;
Site Operations	is depending on data of a particular project and the machines assigned to the specific site.

[7]This is the case with data that is collected on-change rather than at a fixed sampling rate. The on-change collection of data establishes an additional level of compression, since it is dependent on the machine utilisation. Ship loaders for instance are usually operated for a limited time, e.g., loading a vessel, and are then in idle mode until the next bulk carrier arrives.

There are different methods of how data can be accessed by the individual interest groups. Data can be processed automatically to produce summary tables of defined time ranges or of a fleet of similar-type machines. Additionally, during exploratory data analysis, it is of benefit to automatically generate results from an initial set of analyses to create a basis for further work. This is of particular interest for the first involvement of domain expertise. Automatic document synthesis is of support to quickly produce results: All automatically generated data can be provided in PDF[8], or editable text documents, or other formats, which can be downloaded from the data hub or can be send out via mailing lists. Notifications can be distributed via e-mail as well. However, the most important means of provisioning data are data-on-demand services, with which users can query data from the data hub for specific analyses.

2.7 Creating Added Value

Building up a flexible but robust, globally-scaled infrastructure to make data available to specific groups of interest is fundamental to generating added value for both, research entities and industrial applications. During the work in the mining and geotechnical sector, certain points of interest have been identified and are listed as follows [9, 23, cf.]:

Condition Monitoring: The array of potential solutions for questions arising in the fields of condition monitoring, as well as of preventative maintenance, is significantly broadened by the use of data analysis. Data can also be used to identify, which components or assembly groups require additional or particular attention. The system response to external loads represented by operational data supports the understanding how a machine operates. The MTBF[9] can be increased (improved) with techniques used in condition monitoring and in preventative maintenance.

Commissioning Support: During machine assembly, it is of support for the commissioning engineers to have data of the machine available. Unit tests of assembly groups or parts of those can be conducted promptly and can be verified with manageable effort. Especially when a machine is fully assembled and controlled function tests are undertaken, the acquired system response behaviour to external loads is stored to gain a load profile. Such profile is characteristic for the particular machine and can be used during operation to monitor, whether the current load profile matches the original one. The machine will require attention, if unwanted or unknown response patterns are exhibited. In general, this helps increasing efficiency and identifying unexpected system responses.

Fleet Management: Results and insights obtained for a machine can be of use when analysing other machines of similar design and application. However, this is a field requiring a significant amount of data sets from many machines to ascertain, which insights can be projected to other machine instances. For example, two bucket-wheel excavators of the same design, digging the same material in one mine, have

[8] PDF . . . Portable Document Format
[9] MTBF . . . Mean Time Between Failures

been found to exhibit the same characteristics in many aspects. In another example, two ship loaders of similar design, operated beside one another, demonstrated significantly different response behaviours to similar load scenarios. This justifies future research to be conducted on the matter of ergodicity[10] of such machines and systems. Additionally, data can be provided to Computerised Maintenance Management Systems (CMMS), offering support in planning and tracking maintenance tasks for machines.

Automatic Operations Recognition: Time series emanating from machines can be symbolised to form a stream of symbols, rather than a stream of numbers. These symbols can be attributed a length, which characterises their occurrence duration and facilitates time-less sequence comparison. Additionally, it is also possible to compound symbols and sequences. This results in a pattern of how a machine is operated. Labels for the symbols add meaning to single movements, as well as to their compounded sequences. In this manner, the different processes can be classified within the time series to recognise machine operations automatically. Also process time analyses can be conducted in a simple manner, since the run-length of the symbols is known from their attributes.

Engineering Feedback: For a machine manufacturer it is of importance to know, how its products are performing when exposed to real operational environments. To improve future iterations of machine design, it is necessary to get reliable and direct feedback from operation for each machine. Systematic issues can also be identified by inter-machine comparisons, as previously described for *fleet management*.

Incident Analysis: Especially for machines in the mining and bulk materials handling industry, it is of significant importance to ensure operation without interruptions. As a consequence, entire supply lines for subsequent machines or processes are affected immediately; problems in this domain bear immense financial risk. Incidents leading to downtime or other issues require semi-immediate attention, depending on the potential risk consequences. Hence, it is important to analyse time series of such machines in a timely manner to quickly identify the potential root cause, and, to take further action to avoid repetition.

Claim and Warranty Management: The financial risk at hand involved into issues as described for *incident analysis* is imminent to its originator. Root cause analyses based on operational data can potentially support in identifying the responsible party, data analysis can provide critical input for associated warranty or liability claims or lawsuits.

Logistics Optimisation: Mid- to long-term planning of activities in mining and materials handling bear the potential of optimisation. Besides efficiency increases, optimisation can also be achieved by managing machine utilisation to decrease maintenance efforts and to extend life span of machine components. For instance, a bucket-wheel

[10] The definition for *ergodic* is given by: *"Of or relating to a system or process whose overall statistical properties can be determined by analysis of a sufficiently large sample of the states of one of its constituents or instances averaged over time."*, as retrieved from `https://ahdictionary.com/word/search.html?q=ergodic` on 2020-02-12. Further reading in a scientific context is provided in [24].

reclaimer in a stockyard or a bucket-wheel excavator in a mining environment are usually operated to fulfil requirements of subsequent plants or processes. The better the demands of those processes and their schedule are known, the more effort can be put into distributing machine utilisation. Both types of machines can slew their superstructure against the undercarriage, since a slew bearing is connecting both. The life span of this bearing can be expanded by an evenly distributed load profile along its circumference. Hence, logistics planning has an effect on machine life span.

Operational Efficiency Optimisation: Production processes of machines operated by humans often bear the potential of optimisation in terms of identifying subprocesses or tasks that take longer than anticipated. Additionally, complex operations can hold a number of unknown subprocesses or processes, which are invisible at a first screening of data. Those parts all add up to lost time, which is subject to reduction once their root causes and characteristics are able to be determined.

Reporting: Machines are used to perform tasks, i.e., in the mining or materials handling business, or to install products, i.e., in geotechnical engineering. In either application, reports are required to justify the performed work. On the one hand such documentation is required for invoicing, on the other hand it is needed for quality assurance. There are many distinct kinds of reports; many of them can be generated automatically before they are approved if necessary and can be distributed to the recipients.

Notifications: Values exceeding defined thresholds, or follow an unusual pattern, can trigger notification mechanisms. Notifications are distributed to users with appropriate privileges. These notifications can be the trigger of a decision-making chain, which ensures a timely acknowledgement and initiation of further actions: the scheme follows a hierarchical structure of recipients, where the next higher entity will be notified in case a timely response of the current notification receiver is pending for a time span exceeding a pre-defined limit.

From an industry perspective, added value generation is important for short- and mid-term initiatives. Of course, corporate interest groups also have long-term ideas for the beneficial use of advanced data analytics, however, without instruments to accelerate the return on investment (ROI), it is tougher to promote such projects internally. Many of the above listed instruments are of support for the industry to achieve short- to mid-term goals, such as simplified *reports* and *notifications*, *commissioning support* or *incident analysis*. Moreover, such initiatives help to build a framework that allows working on more sophisticated opportunities, such as: *automatic operations recognition*, *engineering feedback*, or *operational efficiency optimisation*.

2.8 Development of a Data Acquisition System for Application in the Mining and Bulk Materials Handling Industry

Initially, the development of the presented system has been conducted for use cases of the company *Sandvik Mining and Construction Materials Handling Gmbh & Co KG*, hereinafter referred to as *Sandvik*. For a limited time, operation and maintenance of the system was also performed by *FLSmidth GmbH*, the company which continued the Sandvik subsidiary after its divestment. The data acquisition system discussed in this section is referred to as *System I*. This part of the work presented is concerned with the development of *System I* until June 2018.

The author was responsible for requirements gathering, the conception of the whole system including the data hub, the installation and deployment of the hardware, the entire software development on the local data acquisition device, as well as for data analysis; the implementation of data flow services on the data hub have only been accompanied and were carried out mainly by colleagues at the Chair of Automation.

2.8.1 Background, Status Quo and Motivation

Previous work has been performed on time series of bulk materials handling machines as discussed in detail in [9]. Findings of these analyses justified further work on how to acquire data in a structured and managed manner for this kind of machines. No standard has been established prior to the development of a new means of acquiring the data from such machines. Data used for previous work has been gathered from different kinds of data collection systems, often recorded in a manual manner or by extracting data from databases of historical data and log files.

The following part should provide an overview to describe the fleet of machines on which the developed system was run on, and their operational environments in an appropriate manner. The machines equipped with the developed system are of these different kinds of mining and bulk materials handling machines:

- Ship Loader – A-Frame Type;
- Ship Loader – Slew/Luff/Shuttle Type;
- Bucket-Wheel Excavator;
- Mobile Primary Crushing Systems;
- Belt Wagon;
- Tripper Spreader.

The data acquisition system was developed for Sandvik, an original equipment manufacturer (OEM). As a mining and materials handling supplier, it is neither the owner nor the operator of the machines the system is implemented on. This is of particular interest when it comes to data ownership, accessibility and the instruments of added value.

For Sandvik it was a necessity to build a data acquisition system, which is additionally capable of running a unique HMI[11] software. This interface was initially planned to be the top layer of all existing systems of a particular machine and should as well be able to act as the supervising software of many machines of a plant, providing capabilities of a SCADA[12] system. Hence, an already existing hardware unit, an edge computer, and a pre-selected software development environment were used for the data acquisition unit; the entire software has been implemented on this type of device. This was also the main reason for the selected size and resource specification of the edge device. The already existing Windows-based edge computer was a B&R APC 910, as development environment Evon XAMControl was used. This environment is based on the Microsoft® .NET framework, all the programming for *System I* was performed using C#.

2.8.2 Concept for Entire Data Life Cycle Management

The fundamental overview of the presented data acquisition system is illustrated in Figure 2.3. It is shown how data is transferred from the local data acquisition device in the field (on the machine) to the main processing server within the central data hub. From there, it is then provided to users of different interest groups. Once the data is collected at field level by a local data acquisition device, it is forwarded to base servers via SFTP[13] upload. For this, files containing the data are uploaded at a fixed interval, usually once a day. The capacity of the SFTP servers is managed by a load balancing service, which distributes all incoming files over the available computing resources. During the ingestion process of the files, they are checked for sufficient integrity and quality. In a next step, data is archived and made available to the main processing server. It acts as the central point of intelligence, where domain expertise[14], metadata and physical models are merged with the data from a machine. The outcome can be consumed via different services:

Notification Services are automatically checking the operational data to be within a valid value range. In case of any deviations or detected patterns of interest, defined decision-making chains are served to enforce a timely response to issues bearing the potential of immediate risk. If the first level of notification receivers does not respond within a specified time, the message will follow an escalation scheme to reach another person with the same or higher privileges to ensure a timely acknowledgement; further actions might be triggered.

Remote Data Access Services enable a user to access data independent from his or her location and used equipment. By providing data via the REST[15] paradigm, the end user can interact with the main data hub by using a web browser on any mobile device. This supports engineers who request results only retrievable from the main data hub, while being directly at the machine's location. Additionally, users without specific software tools to access the data analysis services can retrieve results for standardised enquiries.

[11] HMI . . . Human-Machine Interface
[12] SCADA . . . Supervisory Control and Data Acquisition
[13] SFTP . . . SSH File Transfer Protocol or Secure File Transfer Protocol
[14] Domain experts are often also referred to as subject matter experts (SME).
[15] REST . . . Representational State Transfer

Data Analysis Services are functions and methods implemented based on the outcome of an exploratory phase. Once a method can be reliably designed to accept a set of input parameters to produce specific plots or return results in form of values, it can be encapsulated or containerised and made available for end users. Usually, such functions are called on demand.

E-Mail Reporting Services deliver predefined reports and results to end users via e-mail. Data analysis services are automatically called to derive insights for the content of the material forwarded as attachment of the e-mail messages.

Development Services represent the interface for data scientists to the main processing server. Trained scientists can access time series data in different formats relevant for their analysis purposes, for instance in binary MATLAB® format. This enables a seamless integration of data in the individual local environments the users have established for an exploratory analysis phase. For them, it is also possible to test analysis modules on a dedicated part on the main processing server if required.

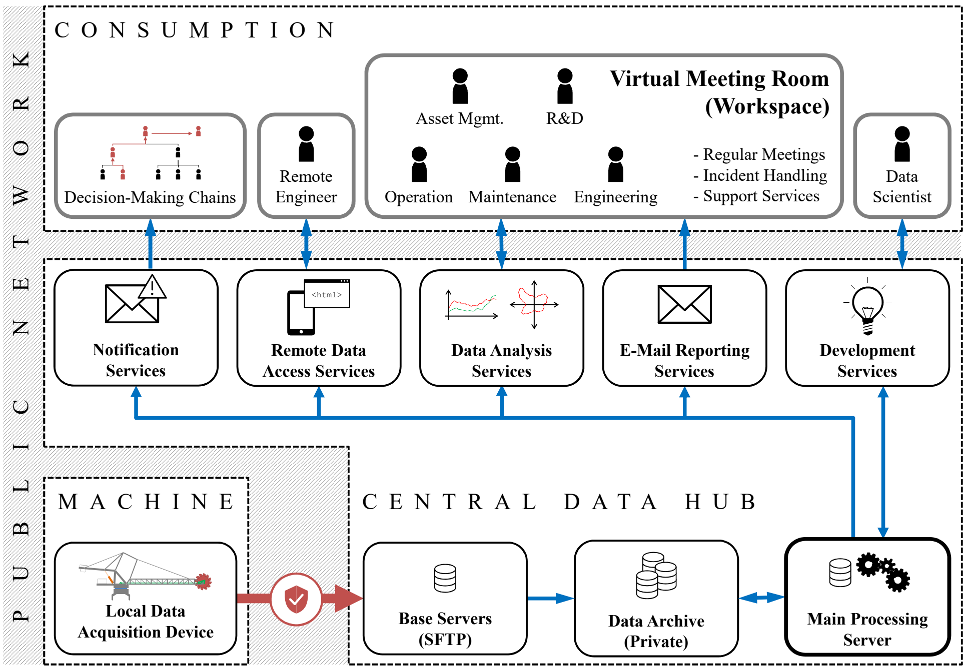


Figure 2.3: **Overview *System I*:** The entire data acquisition system consists of a data source, data management systems and provision services. Acquired data from the machine network is uploaded file-wise to SFTP servers, where the individual files are ingested and stored for further processing and archiving purposes. Additional data is brought in by domain experts, metadata and physical models. An exploratory analysis branch (development services) is served with the stored data from the main processing server. A second branch provides services for notifications, remote data access, data analysis and e-mail reporting. Many of the offered services are used as input for a virtual meeting room, where users of different disciplines can collaborate efficiently.

The concept of a virtual meeting room can be utilised as a workspace for collaboration of different interest groups. The users being part of this virtual workspace can also be of different companies, e.g., the manufacturer of the machine, suppliers, contractors, or the operating entity. It can be used for regular meetings or on-demand incident handling after abnormal occurrences. This is intended to be an instrument to ensure a smooth and flexible interaction of users with data services. For instance, a generalised report could be generated and issued to the participants of a virtual meeting room. During the discussions of the meeting, new results may be of interest; they can be acquired immediately to discuss further. This guarantees that the data and information flow is not bound to specific persons but rather to generic services which allow domain-level users to interact with data in the most beneficial way.

The data flow itself is presented at a higher level of detail in Figure 2.4. A machine is operated at a remote site. It is equipped with sensors and actuators, which are hard-wired to a machine PLC, the central control unit of the machine. All channels of sensors, e.g., pressure transducers or encoders; of actuators, e.g., cylinder valves or electrical drives; and even of user input devices, such as joysticks or buttons, are available at this central control device. The local data acquisition device reads data from this PLC via an Ethernet-based protocol, such as OPC[16]. The data acquisition device stores the values it receives in a local database, which is used as a data buffer. The relevant data is exported to individual files, compressed, and uploaded via SFTP over a public network, i.e., the internet, to the data cluster on a daily basis. There, the receiving services are monitoring specific file locations for any new incoming files. If a new file exists, it will be forwarded to ingestion services, where all data is checked for quality and integrity. It is validated that the values received are within a valid range and of right format and type. This is ensured by providing appropriate metadata via the receiving services. If the file passes the checks, the data will be stored in different formats, such as binary MAT for direct use with MathWorks MATLAB®, CSV[17] for general applications, Apache HBase[18], in the hierarchical format HDF5[19], and others as required. The implemented concept stores data in MAT, CSV, HBase and HDF5; however, the first two were mainly used for analysis during the exploratory phase. If a data file fails the checks during ingestion, it will be quarantined in a protected area where manual interaction is required by users with elevated privileges. Notifications are sent out via e-mail to inform about these files and why they failed ingestion. The reason for failing the checks might be a defect sensor or, in a worse scenario, a security breach. Provision services, such as notifications, reporting, evaluation requests and data access, let the user interact with the data.

The several data manipulation steps executed on the local data acquisition device are demonstrated in Figure 2.5. Two physically separated Ethernet ports ensure an isolation between the network connection to the machine and the network connection to the central data hub. Machine data is read from the machine PLC via a read-only connection, e.g., OPC with disabled write access, via Ethernet port 1. A listening interface service

[16]OPC . . . previously Object Linking and Embedding (OLE) for Process Control, now Open Platform Communications [25]

[17]CSV . . . Comma-Separated Values

[18]HBase . . . Database of the Apache Hadoop framework; HBase is a non-relational database, modelled after Google Bigtable

[19]HDF5 . . . Hierarchical Data Format, version 5

is set up to read values from the machine's controller. This interface considers values which change in comparison to their previous readings (on-change data acquisition). The runtime software on the device is checking the data for validity based on the file format specified for each channel. The values are forwarded into a buffer that writes the data to a Microsoft® SQL database. New data is attributed with a minimum keep time, which is usually set to a couple of months – in this particular case it was set to six months. Since the local database has limited memory, a recurring cleanup routine removes data with an expired keep time, in the case memory space is required.

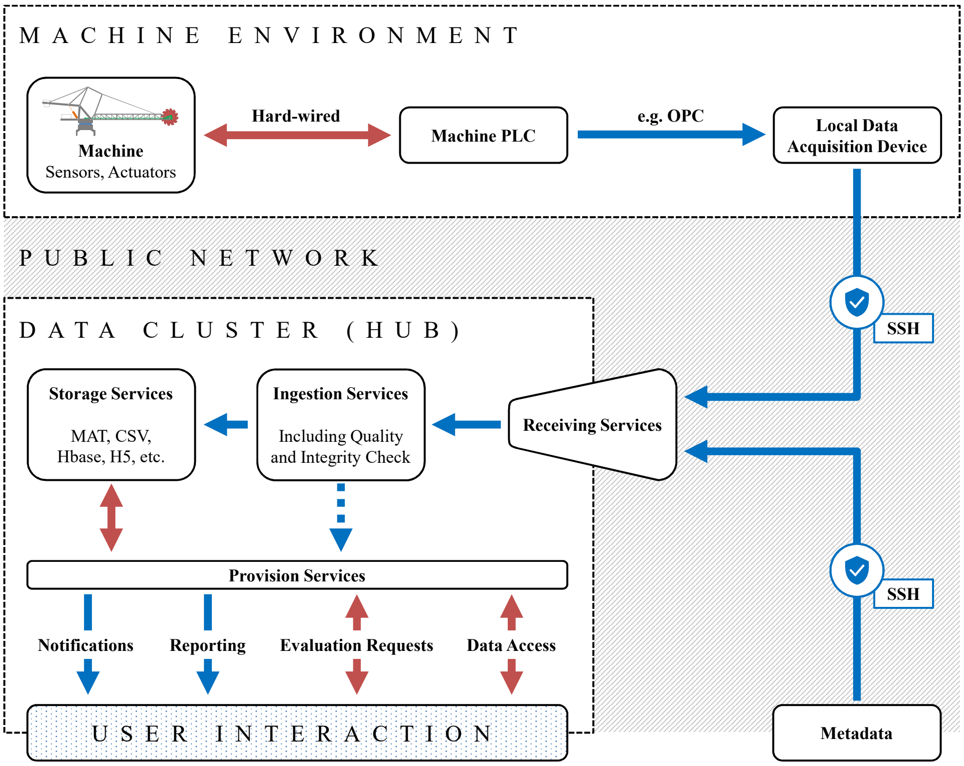


Figure 2.4: **Data Flow and Ingestion:** Data channels on a machine PLC are gained from its sensors and actuators. The local data acquisition device can access those channels by using a protocol such as OPC. After buffering the data locally, the data acquisition device sends it as individual files to the receiving services of the data cluster. This is done via a secured SFTP connection. Additionally, metadata necessary for the ingestion checks and for further data provisioning is provided via a secured connection to the data cluster. An internal service listens on specific file locations on the SFTP server (part of the receiving services) and forwards any newly uploaded file to the ingestion services. There, the file content is checked for sufficient integrity and quality, prior to being stored in different formats serving diverse purposes. Several provision services let user interact with the data.

After a full day is concluded, an autonomously running service exports values of the data channels from the database to CSV files; one file per day. The exported numbers are stored as on-change data in the following format:

```
1337.1508, 2016-06-23T17:31:24Z, 958923768235094, 7
<value>, <timestamp_UTC>, <channel_id>, <channel_type>
```

The files are saved to a designated folder on a second drive. An upload service looks for newly generated files and compresses them in a lossless manner. The file size after compression is approximately at 10% of the original size, which is beneficial in case the internet connection is unstable, the transfer quality is insufficient or there are other limitations. Memory on the data acquisition device is provided by a solid state drive (SSD) and a CFast drive. The solid state drive hosts the Microsoft® Windows operating system, as well as the SQL database. The exports of the database – the CSV files and their compressed versions – are stored on the CFast drive, a variant of CompactFlash memory with higher transfer rates.

The compressed files are transferred to the central data hub by using an SFTP service. Each local data acquisition device uses dedicated, unique credentials. The connection to the SFTP server is secured via SSH, providing transport, authentication and connection layer security. The security device, as seen in the figure, is connected to Ethernet port 2 of the data acquisition device. The model MGUARD RS 4000 from Phoenix Contact was used as security device. It is acting as modem for mobile internet connections, as a firewall and a VPN[20] end point. The latter is especially of interest for remote support entities, which can access the local data acquisition device over an active VPN tunnel connection. For security reasons, only the security device can initiate VPN connections to a central VPN hub. It is configured to initiate a connection as soon as it has an active mobile internet connection. However, it could also be initiated on demand, e.g., when a service technician on-site starts the VPN initiator. Once a VPN tunnel is established via IPsec, a remote engineer can access the local device. The IPsec VPN tunnel uses X.509v3[21] certificates in a public key infrastructure. To establish a secure tunnel, following cryptographic algorithms were used [27]: AES256 (symmetric) and SHA2-512 (hash function).

In addition to the security given by the utilisation of SFTP for file transfer, several layers of implicit security are part of the concept. As presented in Figure 2.6, the numeric values are transferred to the central data hub entirely separated from metadata. The latter is provided to the data hub using another channel, e.g., another SFTP connection or encrypted e-mail. Because of this separation, a file from the data acquisition system cannot be used without the metadata: merging both does not happen prior to processing on the central data hub. A file holding only the numerical data cannot be processed without the associated metadata, because of the missing channel attributes.

A system used for the ingestion of data emanating from multiple machines must also be capable of treating the incoming data individually. To run the services on the central data hub in an efficient and resource-aware manner, all the processes are operated within a

[20] VPN . . . Virtual Private Network

[21] X.509v3 is a standard for certificates used for public key infrastructures. [26]

virtual environment. A separate virtual machine is used for each machine data is received from. This avoids any interference between the digital versions (digital twins) of the machines.

In case of any issues, the owner of the machine with the data acquisition system installed should be able to disconnect it from its machine network directly without the need of assistance by the machine manufacturer. This is also true for the second network line, which is established for communication with entities which are not part of the machine network. The connection of both network lines can be interrupted separately by pulling the plug of either Ethernet port. The requirement for this necessity was specified by a client of Sandvik, who requested this as a part of an emergency plan in case of a system intrusion during a cyber attack.

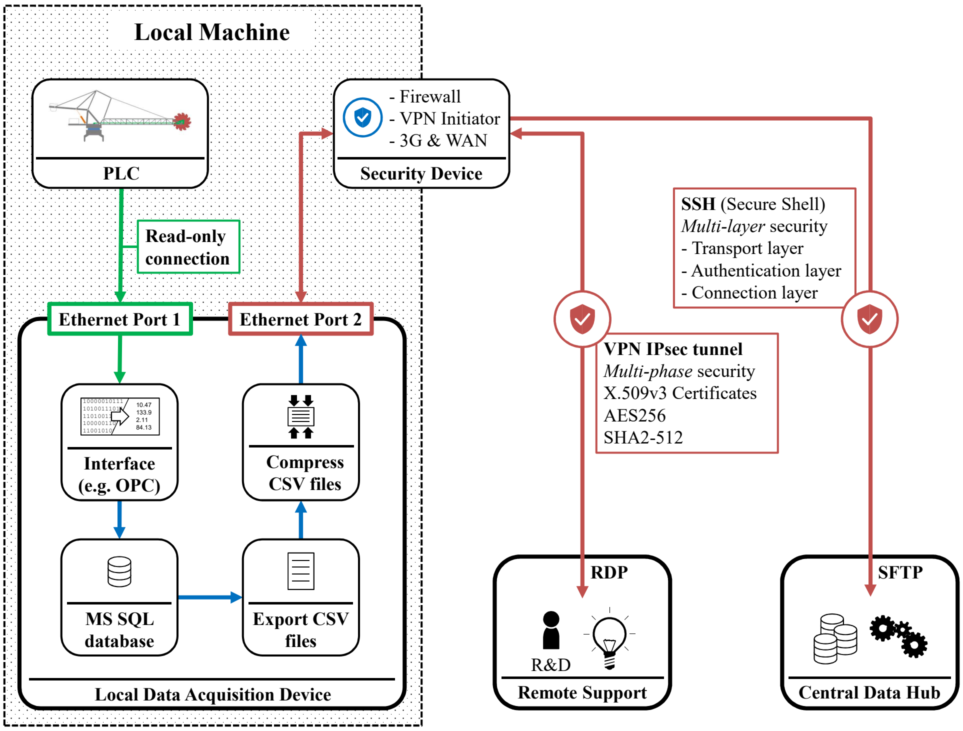


Figure 2.5: **Local Data Handling and Transfer:** The local data acquisition device receives data from the machine PLC via Ethernet port 1. An interface listens on this port for incoming data via OPC. The values are stored on-change in the SQL database. The data is exported to files on a daily basis, which are then compressed before they are uploaded to the central data hub via SFTP. This is done via a security device, which is connected to the second Ethernet port of the data acquisition device. Besides providing modem functionality, the security device is also acting as a firewall and VPN end point. The latter is used for a VPN tunnel that is established for remote support.

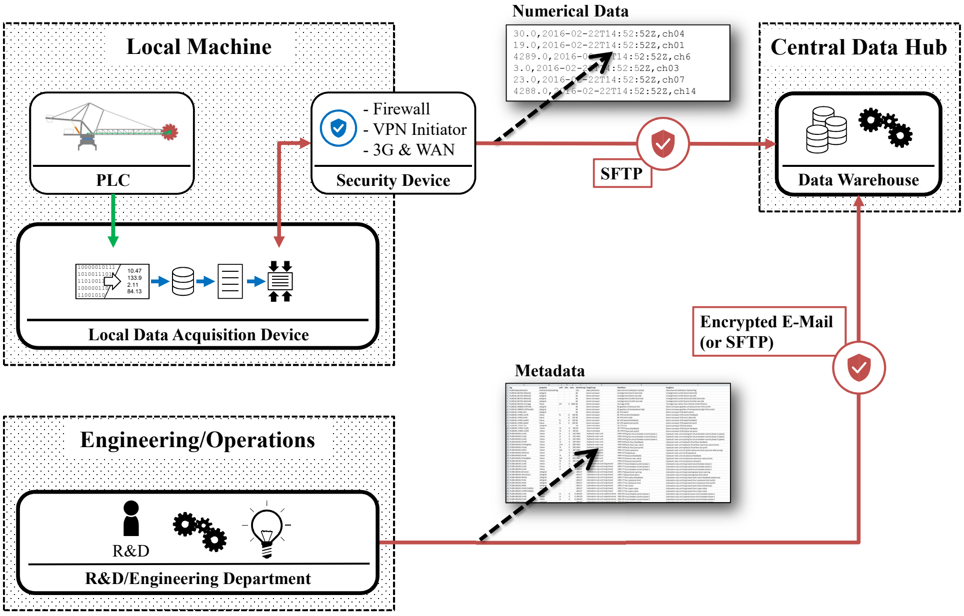


Figure 2.6: **Separation of Numerical Data and Metadata:** The files transferred from the local data acquisition device only contain numeric values. Metadata from domain experts is provided via a separate transmission channel. This adds an implicit level of security.

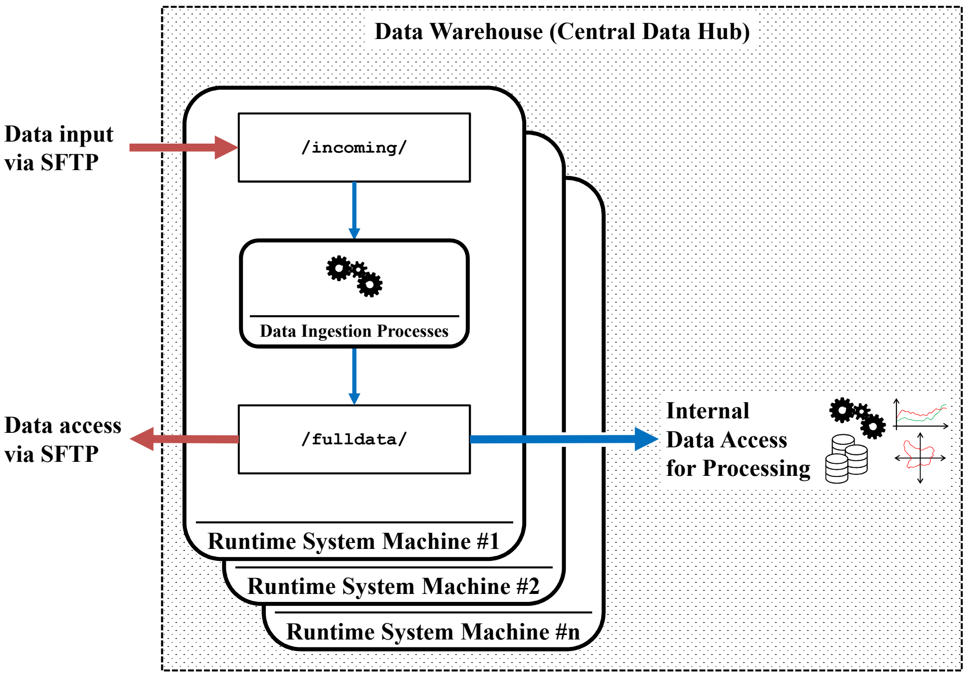


Figure 2.7: **Multiple Virtual Environments for Multiple Machines:** All the server-side processes and services run in a virtual computing environment. An individual virtual machine is maintained for every machine data is received from. This avoids any interference of the digital clones of the machines.

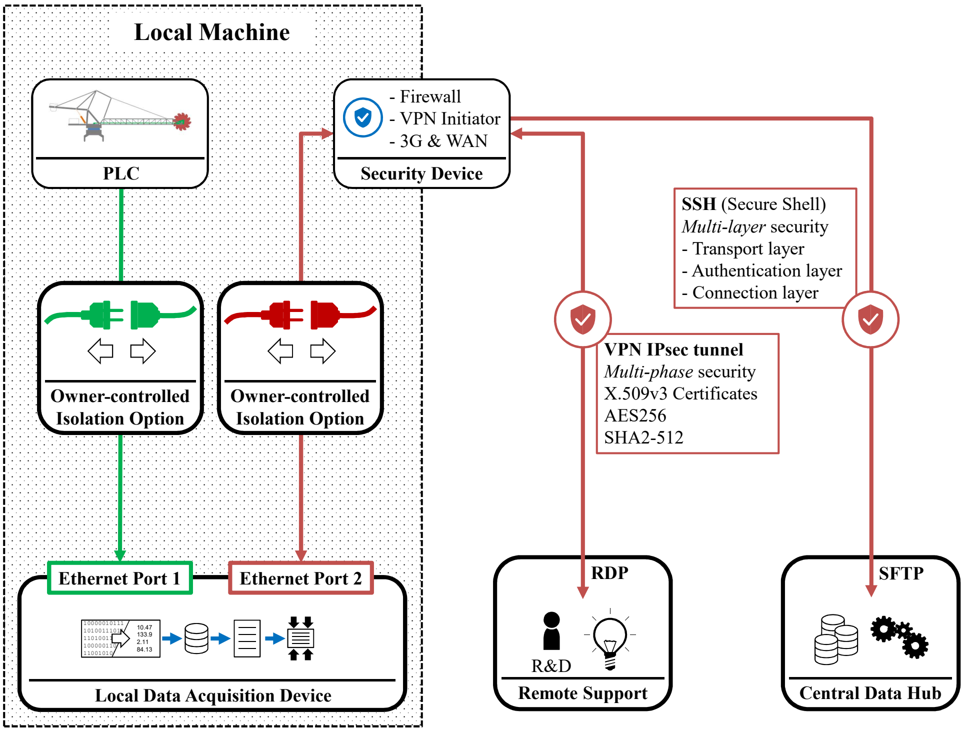


Figure 2.8: **Isolation Option for Owner:** At all times, the connection between the data acquisition unit and the machine PLC can be interrupted by unplugging *Ethernet Port 1*. To not interfere with data acquisition but to disconnect the device from sending data to an end point not located on the machine owner's premises, there is the option to uncouple the network connection from *Ethernet Port 2*.

2.8.2.1 Memory Handling

With increasing amount of data acquired, it is necessary to manage memory appropriately. The goal of handling memory in this section's context is to have a reasonable level of data redundancy persistent. Meaning, acquired data is only deleted, when the acquiring device is running out of memory. However, under no circumstances should existing data prevent newly acquired data from being stored.

2.8.2.1.1 Memory Handling on Local Data Acquisition Device As previously mentioned, the local data acquisition device has a solid state drive for the operating system, the runtime software and the database. A separate CFast drive holds the database exports – the CSV files – as well as their compressed counterparts.

All values stored to the database are attributed with a specific keep time. A recurring cleanup routine, implemented as a stored procedure, deletes all data which keep time has expired, in case memory space is needed. The time is specified to a certain limit, so that the maximum available memory space is not exceeded by the storage amount of to-be-kept values.

For the files on the second drive, the CFast drive, a memory watchdog service has been implemented. The watchdog checks the available memory within a specified interval. If it falls below a certain limit, e.g., 10 % of free memory left, a clean-up process starts to delete the oldest files on the disk. At first the oldest export files are removed, before any of the compressed files are deleted. The removal process is executed until the amount of available memory is again above a specified threshold, e.g., 15 % of free memory. With this hysteresis-like approach, the access cycles on the second drive are limited to a minimum.

2.8.2.1.2 Memory Handling on Provisioning End Point Memory handling on the server end point of the concept is not so centred around memory space, it is more concerned with how data is accessed efficiently; to minimise download times. To let users retrieve data from a central data hub and to apply advanced methods of data analysis as described in a later part of this work in more detail, it is beneficial to organise the data in a contiguous manner. The content for this *contiguous data model* is stored in a data archive on the server. It contains all the data available for the respective machine or project. A direct consequence of establishing a contiguous data model is the need for efficient access of the data for someone querying results from a local entity. The longer the data acquisition is active, the more data is part of the contiguous data model. Especially during an exploratory phase of data analysis, it is a prerequisite, to ensure a smooth and efficient way of accessing the necessary values. The approach for an adequate access for a data scientist is organised in three levels, as illustrated in Figure 2.9: Once data has been downloaded to the user's computer, the portions for the last three days of a particular machine is held in random access memory (RAM), while data spanning about three weeks is stored onto the local drive, a hard disk (HDD) or solid state drive (SSD). All other data remains on the server and is provided to the user via network each time it is demanded. The latter has been shown to be significantly slower [28]. This approach supports adequate access times across all data available, resulting in faster retrieval times for more frequent queries [9].

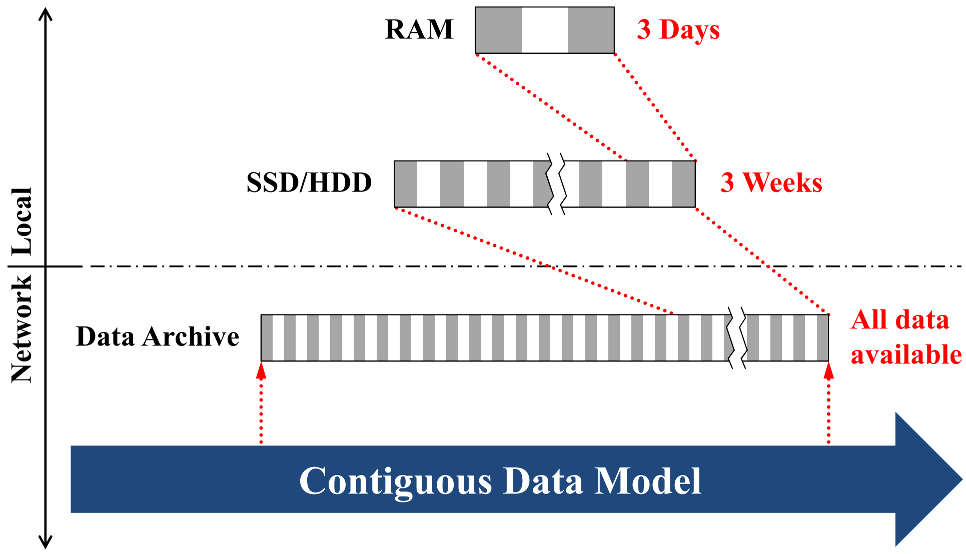


Figure 2.9: **Three-Level Caching Methodology for Data Access** for a data scientist during exploratory work to values behind a contiguous model façade is organised in three distinct steps: for the previous three days in RAM, for the last three weeks on a local drive – all other data resides in the data archive on the server on network level and is only enquired on demand. Such a three level caching methodology ensures efficient retrieval times for frequently accessed data. [9]

2.8.2.2 Data Reconstruction: From On-Change Data to Contiguous Data

On the receiving end of the system, data is made available in the form of a contiguous data model. With this, an additional meta level is introduced, which acts as a façade between data queries by users or subsequent processes and the input format of the data. This separation of the output from the input is illustrated in Figure 2.10. File-based data is ingested into the model for every calendar day. However, data is often required for different time spans, e.g., for daily evaluations with a timeshift (Δt between UTC and local time), for shift or weekly summary reports, or for evaluations of a materials handling process spanning multiple days – such as loading a vessel.

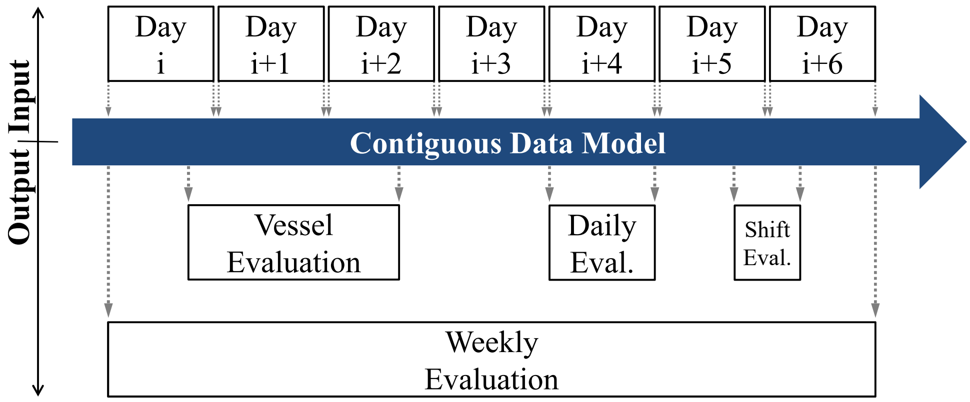


Figure 2.10: **Contiguous Data Model With Output Separated from Input:** This model establishes a façade between data input and output; the input format is entirely separated from the output. Queries of data for varying time spans are attainable, since an intermediate query handling process returns data for the enquired time span.

The contiguous data model is realised by an intermediate query handling process, which accepts queries similar to the following format:

```
getData(<channel_ID>, <timestamp_UTC_from>, <timestamp_UTC_to>)
```

The entire data collection and provision framework was not only installed to accommodate data in a controlled manner from local data acquisition units, but also to ingest other sources of data. As apparent in Figure 2.11, historical data from data archives can also fit into the generalised data model. There was the necessity to explore data already available by a variation of sources, especially for early initiatives in exploratory data analyses. For instance, exports from data collected by a third party on-site monitoring system or by a logging function on the machine control system (both counting to *Data Archive* in the figure) were used for an initial exploratory phase. Acquired data from the local data acquisition unit are also fed into the same data model, regardless of the input format: file-based as the main means of data transfer, direct database access via VPN as an option, as well as live data access in a related form of streaming as discussed in section 2.8.2.4. All the available input formats are processed individually on the central data hub to fit into the data model. Hence, if another source format is added to the system, only a pre-processor needs to be implemented to take over the tasks of preparing and converting the data into the desired data model format; in case of the described local data acquisition unit, one of the main tasks is the conversion from on-change data to full table. [9]

The metadata the operational data is attributed with is delivered to the ingestion process in the same manner, also regardless of the input source. Information regarding the machine and its sensor setup or about the rule sets for quality and integrity checks is brought into the system by using simple Microsoft® Excel files in XLS (XLSX) format. The format decision for the metadata is of particular interest, since it enables domain experts to provide the required data in a format native to them. For the ingestion process it is of no importance, in which format the metadata is brought in, as long as it is prepared in a manner it can be interpreted correctly. Hence, there is only the need for an appropriate metadata handling process, which ensures that the data contained in the Excel files can be merged with the associated operational data.

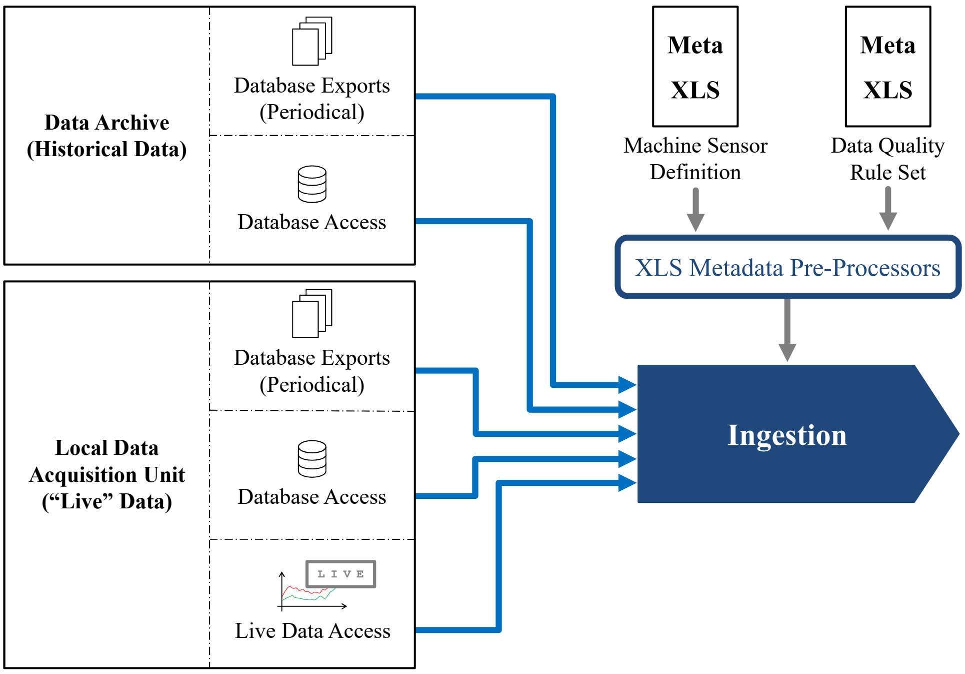


Figure 2.11: **Processing Data from Different Sources:** The presented framework is able to accommodate data from a variety of source formats, ranging from third party data archives to formats natively provided by the developed local data acquisition unit. The metadata is prepared by domain experts in an Excel format, making data scientists obsolete for this task. All system-specific transformation and handling of the metadata is performed by pre-processors. [9, cf.]

As pointed out previously, the local data acquisition unit acquires and delivers data to the ingestion process in an on-change format. To conform with the contiguous model, the raw data needs to be reconstructed to full table. The goal of this step is to build a table-like data structure, which holds all the available data channels as individual columns and a leading column with all the timestamps. For the latter it is an additional requirement to ensure a unique time step length between the timestamps, resulting in a uniform sampling rate. With this it is ensured to have a value for each channel for each timestamp. The reconstruction process is illustrated in Figure 2.12. On the left, there is a set of on-change data in the form of a long list, where one line corresponds to a channel value that changed at a particular time, while on the right there is the output in form of a "full table". The first of the two intermediate steps showcases, how the on-change data resembles a fragmented table with many voids in between. The solid blue line indicates the values gained from a calibration routine, which runs at least once a day on the local data acquisition unit. This guarantees for a certain timestamp to have values available for all monitored channels. Zero-order hold (ZOH) and first-order hold (FOH) models are used to reconstruct data [29]. The aforementioned calibration values ensure that the ZOH and FOH models for both value types, Boolean as well as analogue values, are applied in a manner they produce reliable results.

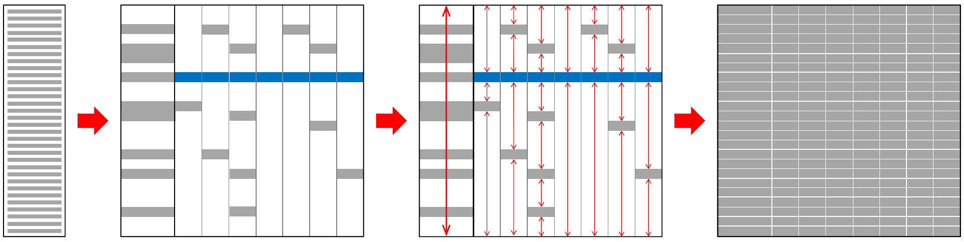


Figure 2.12: **Reconstruction of Full Table Data:** On-change data contained in the files of the local data acquisition unit can be found on the left as a list of timestamp-value pairs. In a first step, a table is generated, which has a leading column of timestamps with a unique step length constituting a uniform sampling rate throughout the table. Each channel has its own column containing corresponding values at the particular timestamps. In a second step, the gaps are closed using ZOH and FOH models to ensure the generation of a *full table*, containing a value per channel and per timestamp. [9]

2.8.2.3 Integration of Subsystems

Previous work has shown that significant attention needs to be addressed to particular assembly groups of mining equipment [9, 29–31]. Particularly, the slew bearing of a ship loader or a bucket-wheel excavator is a vulnerable component crucial to the machine's ability to operate normally. In many cases it is required to install additional sensors to monitor specific aspects of an assembly group or main component. To accommodate such independent monitoring units, the communication with, as well as the data transfer from these units, is part of the concept of the data acquisition system: they are considered subsystems.

An example is given by a monitoring system, which observes the vibration response of a slew bearing. As shown in Figure 2.13, the subsystem extends the already existing local data acquisition device by a subsystem PLC, as well as a network switch to connect the new device properly to the existing network. The additional devices are operated on the second Ethernet line of the data acquisition device, which is physically separated from the original machine network to avoid any potential interference. The supplementary PLC is used to communicate with four piezoelectric acceleration sensors, which are mounted on the inside of a slew bearing. Two of such bearings have been equipped with this subsystem, the inner diameter of those components ranges between approximately 5.2 m and 6.6 m. All four sensors of each subsystem deliver raw acceleration values to the subsystem PLC, two sensors are additionally providing temperature information. The values are captured at a sampling rate of 2.5 kHz, which is significantly faster than the one of the local data acquisition device (1 Hz). This sampling rate exhibits the physical limit of the used subsystem PLC. The PLC consists of a B&R CP 1585 CPU of the X 20 series, with a CM 4810 high-speed analogue input module and an AT 2222 temperature input module. The software is designed to not exceed a CPU utilisation of 70 % to avoid missing data in case of overutilisation. In idle mode, the control and status variables are actively read and written, the two temperature values are processed and provided to the local data acquisition device and watchdog services are running, which observe local memory space as well as the health of the connection to the local data acquisition device. If the subsystem PLC is in acquisition mode after the control variable has been set, the values of all four vibration sensors are written to a buffer array, each holding 60,000 samples per channel. Once an array is filled with values, another process writes its content to a file in an asynchronous manner. Hence, each file contains 24 seconds of data. While the file is written, the next array is filled with another 60,000 samples. There are six buffer arrays in total to ensure a continuous data acquisition. During operation of two units on two different machines, and also during operation within a controlled test environment, it has been shown that usually three buffer arrays are busy either capturing values or feeding values to the file writing process. The maximum number of concurrent busy arrays was observed to be four. The acquisition process stops after the control value has been reset, however, it is delayed to write the content of the last file entirely; all files exhibit the same number of entries. In case the acquisition process is triggered again while data is still added to the currently active array, the whole process is restarted without interruption. To keep processing time at a minimum, a written file follows this format:

```
34.2585,29.9359,18.1281,36.2899
<sensor_value_1>,<sensor_value_2>,<sensor_value_3>,<sensor_value_4>
```

Only the values are written as rows with comma-separated values. Each row represents one sample of four readings. The time between the samples is always 400 μs, otherwise the system would produce an error to prevent any misinterpretation of data. Another software piece observes the folder to where the files are stored on the PLC and compresses the files. The newly created size-reduced files are then put into a final folder, where they await downloading. The original files are removed.

When a capture process is started, the local data acquisition device provides its current timestamp, which is then used as a prefix for the generated files on the subsystem. Every filename consists of the timestamp prefix, the subsystem ID and a sequential ID to indicate in which order files have been created. This allows a later reconstruction of the values and they can be sufficiently attributed to the operational data provided by the local data acquisition device. Detailed analyses of the monitored values are outside the scope of this work, however, the concept and implementation of a subsystem that is able to integrate sensors with the necessity of a higher sampling rate is.

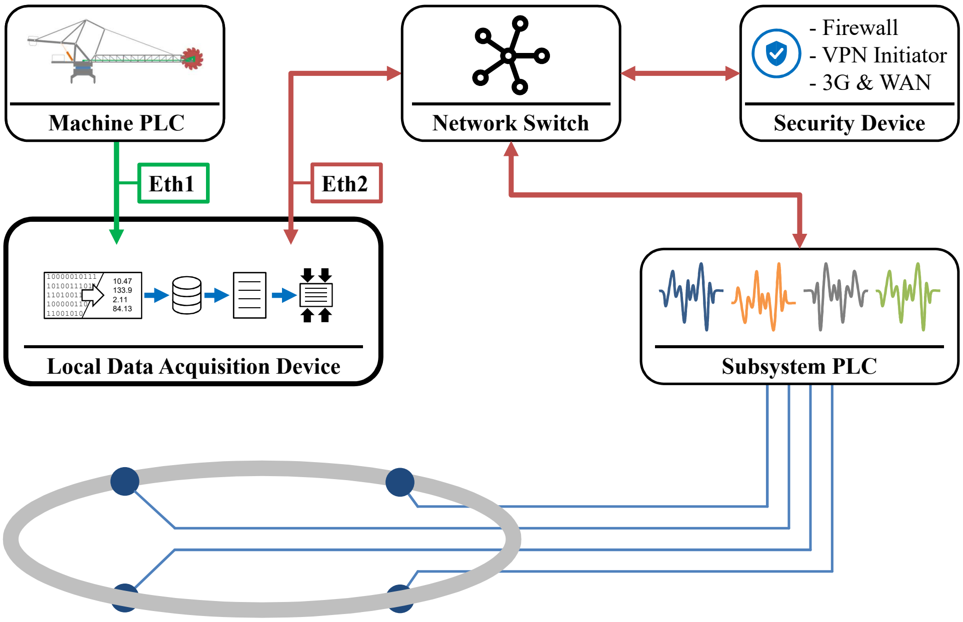


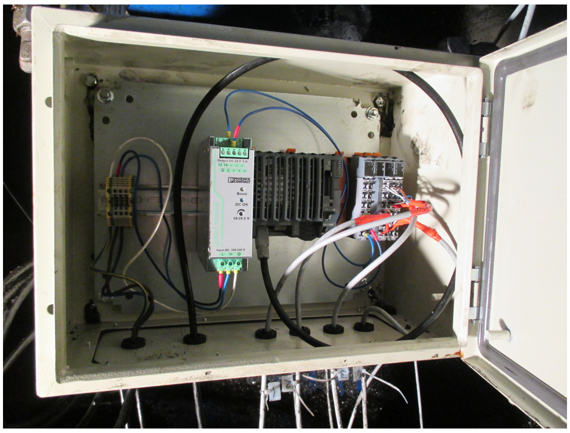
Figure 2.13: **Subsystem Slew Bearing Monitoring Overview:** The monitoring abilities of the already existing local data acquisition device are extended by the installation of a subsystem PLC. This additional component is connected to the second physical network line of the data acquisition device by using a network switch. This ensures that the added device cannot interfere with the machine network connected to the first Ethernet line of the data acquisition device. Four sensors installed on the inner side of a slew bearing are connected to the subsystem's PLC and deliver raw acceleration values the PLC is sequentially saving to files.

The subsystem PLC is running autonomously and separately from the rest of the data acquisition. Communication between the subsystem and the existing system is limited to control and health monitoring purposes, which is performed via the binary protocol of OPC UA[22]. The variables used on the local data acquisition device are listed in Table 2.1. In addition to parameters indicating the current status of both involved devices, there are two life tick variables: *lifeTickLDAD* is a counter running on the local data acquisition device to provide a heartbeat-like signal to the subsystem; and *lifeTickSubSys*, which reads a similar value from the subsystem. If the change rate of any of both variables exceeds specified timeouts, the connection is reset and restarted from either end. A message is sent out to notify about such issues, especially if they exhibit a recurring nature. Additionally, the reset mechanism is bound to other measures to avoid an infinite restart loop in case of an interrupted connection or a permanent issue. The files containing the acquired raw values of the sensors are provided to the local data acquisition device via a private FTP service, which runs on the subsystem's PLC. When a file is downloaded correctly, it is removed from the subsystem PLC to free up memory space.

The installed enclosure with the subsystem PLC and the necessary auxiliaries, such as terminals and a power supply unit, can be seen with the lid opened in Figure 2.14a. It is installed within the slew deck of a bucket-wheel excavator, next to where the sensors

[22]OPC UA . . . Open Platform Communications Unified Architecture [32]

have been mounted. All four of the acceleration sensors have been screwed onto metal pads, which have been glued to the inner surface of the slew bearing by using a two-component epoxy adhesive. One of the sensors can be seen in its final mounting position in Figure 2.14b; the surface area around the sensor has been cleaned for the application of the adhesive.



(a) Components of the Enclosure

(b) Mounted Sensor

Figure 2.14: **Subsystem Slew Bearing Monitoring Installation:** The subsystem PLC with auxiliaries is installed in an enclosure within the slewdeck of a mining machine. Four sensors are mounted on the inner circumference of the slew bearing of the same machine. Mounting pads have been glued to the metal surface with epoxy adhesive to not void the warranty of the slew bearing. The sensors are screwed to these pads to ensure sufficient contact with the bearing. [9]

Tag Name	**Data Type**	**Description**
busyAcq	Boolean	Flag indicating that subsystem is currently acquiring data
busyWrite	Boolean	Flag indicating that subsystem is currently busy writing to the file system
comOpcUaOk	Boolean	Status indication for the OPC UA connection (from OPC UA driver of the client on the LDAD)
cycleActive	Boolean	Status flag indicating a currently active acquisition cycle
errFlag	Boolean	Error flag of the subsystem, indicating a currently active error awaiting acknowledgement
errMsg	String	Elaborate text description of the currently active error on the subsystem
fileCountCycle	UInt32	Number of files generated on the subsystem during the active or last cycle
fileCountTotal	UInt32	Total number of files generated during uptime of the subsystem
fileNamePrefix	String	Prefix for all file names, including the local data acquisition device (LDAD) timestamp at start time
lifeTickLDAD	UInt64	Running counter of LDAD
lifeTickSubSys	UInt64	Running counter of subsystem
resetErr	Boolean	Reset/acknowledge latest error flag and message of the subsystem
start	Boolean	Start trigger for the subsystem; can be triggered automatically, but is usually started by a machine movement that might have an influence on the slew bearing; acquisition is stopped when this trigger is reset to *FALSE*
statusOkLDAD	Boolean	Status indication of the LDAD as information for the subsystem
statusOkSubSys	Boolean	Status indication of subsystem

Table 2.1: **OPC UA Variables for Subsystem**, used on the local data acquisition device to control and monitor the slew bearing monitoring subsystem

2.8.2.4 Database Mirroring – Implementation Test

As pointed out earlier, the local data acquisition device was built based on an edge computer, because a unified HMI was supposed to run on it as well. A local HMI can only access the data contained by the local database, which is limited to the available amount of memory space. Additionally, if there is no connection to the local node in the field or if it is not stable enough to support an efficient online interaction, an engineer cannot establish a connection to the local device. The same problem exists when HMI variables are streamed by using an appropriate protocol, such as MQTT[23] or OPC UA. To overcome these issues, a server node has been set up to create the ability to run the HMI on all data available for a particular machine. For this, the data of the local data acquisition device needs to be mirrored to the server node. This ensures that the server node always holds a copy of the latest data; in case of an interrupted connection, the data that has been mirrored at last is accessible on the server node. A trial run has been undertaken to test such a database mirroring approach between the local data acquisition device and a server running in a virtual environment. The principle is illustrated in Figure 2.15.

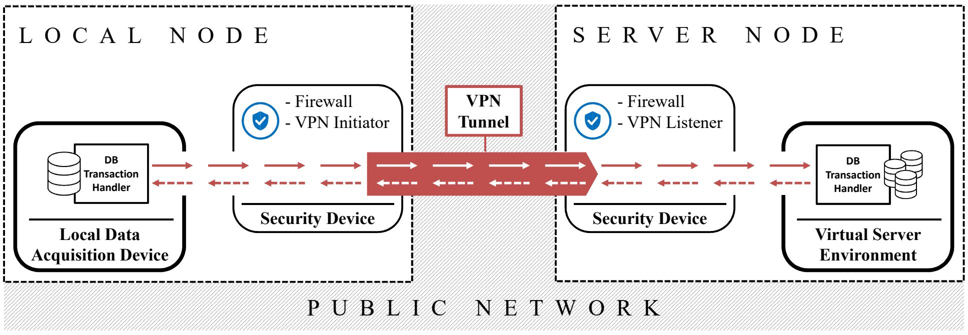


Figure 2.15: **Database Mirroring:** Values are read from the runtime database of the local data acquisition device. In a next step, the values are serialised to OPC UA strings, so that a receiving service at a virtual server environment can read the data. The values are then written to the server databases. The OPC UA connection is encapsulated in a VPN tunnel, which exhibits security measures similar to the other connections the security device establishes.

Both end points are connected via a VPN tunnel, which is initiated by the security device existing at the local node. A VPN listener at the server node waits for the initiation of the tunnel. The same security standards as used for the previously described tunnels apply for this VPN tunnel. Once the connection is stable, a mirroring service reads data from the local data acquisition device's database, serialises it to strings and forwards it to the virtual server environment by using OPC UA variables. At the server node, the string variables are read and the data values are stored in the server database. This process is considered being a passive form of mirroring, since data is only written to the server entity, following a write once read many paradigm (WORM).

Since Sandvik's intention of additionally running HMI software on the acquisition device

[23]MQTT . . . Message Queuing Telemetry Transport

was eventually dismissed, the implementation of this database mirroring approach was limited to a trial run for two machines spanning half a year.

2.8.3 Implementations of *System I*

In the following section, a selection of implementations of *System I* are described. The first two examples are of the materials handling sector, whereas the others are counting to the mining sector. Characteristics of each machine and implementation specifics are discussed. Local data acquisition devices have been installed on all machines, a central data hub as previously specified has been established at the Chair of Automation. The server hardware of the data hub is physically located in Leoben. The machines' operation locations are distributed over the world, exhibiting the truly global nature of the whole system.

2.8.3.1 A-Frame Ship Loader

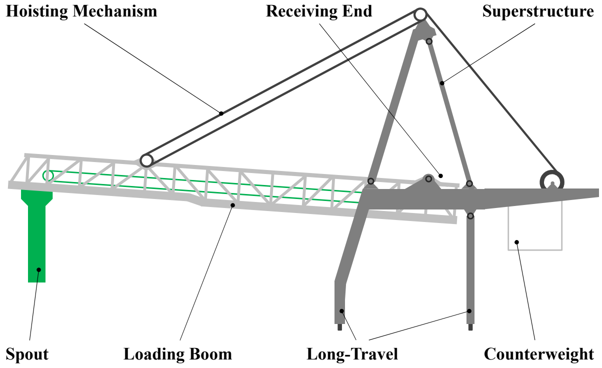
The land transport of bulk materials from mines, such as iron ore or coal, is usually accomplished by trains on tracks. To overcome longer distances on water, huge bulk carrier vessels are used. Track-mounted ship loaders are used to fill the material into the storage compartments of the bulk carriers. [33]

The type of ship loader illustrated in Figure 2.16b, a so-called *A-frame ship loader*, consists of a superstructure in the shape of a capital "A". The machine can move along tracks. The loading boom receives bulk material from a tripper car, which elevates the material from the port-level conveyor to the level of the boom conveyor. The hoisting mechanism, consisting of an electrically-driven winch and steel ropes, lifts the boom up and down. To precisely fill material into the vessel holds and to avoid unwanted spilling, a spout is mounted at the tip of the loading boom. To ensure vessels can approach and depart the loading area without obstructions, the loading boom can be lifted entirely. There are variations of this kind of ship loader with telescopic loading booms, so that the outreach is increased to serve a higher variety of vessel sizes.

This machine is controlled via an Allen-Bradley PLC, which is capable of providing signal tags over OPC. To read the variables, an OPC server application is installed on the data acquisition device. As Table 2.2 shows, one third of the signals acquired are of an analogue nature. No subsystem has been installed on this machine.

The internet connection could not be considered stable in the area of operation of this machine. During heavy winds or storms, the connection was interrupted temporarily, sometimes for several days. In addition to that, a hardware defect of the security device made it necessary to exchange it with local support.

(a) A-Frame Ship Loader [34] (typical)



(b) A-Frame Ship Loader Principle Sketch

Figure 2.16: **A-Frame Ship Loader:** This kind of machine consists of a superstructure in the shape of an A, with a loading boom attached to it. This boom can be lifted up and down by a rope-winch hoisting mechanism. A tripper car (not shown in 2.16b) elevates the material from the conveyor on the ground to the conveyor on the boom. The spout at the tip of the boom ensures that the material is filled into the vessel holds without unwanted spilling.

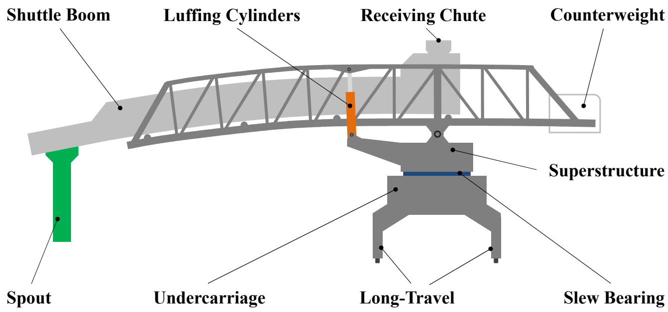
Signals from Machine PLC			**Additional Signals**	**Location**
$\sum$	**Analogue**	**Boolean**		
228	76	152	none	North America

Table 2.2: Signal Overview of A-Frame Ship Loader

2.8.3.2 Slew/Luff/Shuttle Ship Loader

As illustrated in Figure 2.17b, a slew/luff/shuttle ship loader has a compact superstructure. This type is as well mounted on tracks and can travel along those. The loading boom of this machine can be lifted up and down via hydraulic luffing cylinders. Again, a tripper car elevates material from the port conveyor to the receiving chute of the ship loader, which sits atop of the machine. Since a slew bearing is mounted between the undercarriage and the superstructure, the latter can be rotated about the vertical central axis of the machine. Additionally, the outreach is significantly increased by a shuttling mechanism on the boom, to serve a higher variety of bulk carriers. At the tip of the loading boom there is a spout, guaranteeing the material being loaded into the designated vessel hold in a controlled manner.

(a) Slew/Luff/Shuttle Ship Loader



(b) Slew/Luff/Shuttle Ship Loader Principle Sketch

Figure 2.17: **Slew/Luff/Shuttle Ship Loader:** A slew/luff/shuttle ship loader consists of a compact superstructure, to which a loading boom is attached; the luffing movement is performed by hydraulic cylinders. The outreach can be considerably extended by a shuttling boom, which can be extended and retracted. There is a slew bearing mounted between the superstructure and the undercarriage, allowing the machine to rotate about its central, vertical axis. The complex movement of the centre of gravity as a direct consequence of the sophisticated kinematics is assumed to have an impact on the life span estimation of the slew bearing [9]. At the tip of the boom a spout is installed, with which the loading process of a bulk vessel can be accomplished in a controlled manner.

Similar to the previously discussed ship loader, this one is also operated by using an Allen-Bradley PLC. The protocol used for reading the variables is therefore OPC again. Due to its complex abilities to move, this machine exhibits a rather considerable number of analogue input signals – they add up to over two third of all monitored signals (see Table 2.3).

The position of the centre of gravity (COG) varies significantly, since the kinematics of this type of ship loader are complex due to its shape and degrees of movement freedom. Previous work indicated that the position of the centre of gravity has an impact on the life span of the slew bearing; especially changes of the COG position are of direct relevance to life span estimations for slew bearings [9]. For this reason, particular focus has been set onto the slew bearing of this machine and its monitoring. The data acquisition system is extended in this implementation to include a subsystem for acquiring data of accelerometers mounted on the inner ring of the slew bearing. These sensors deliver raw acceleration values at four locations along the inner circumference of the bearing and temperature values for two of the four vibration sensors (see Table 2.3: 4+2 additional signals). Besides the temperature values, which are read constantly, the acquisition of the acceleration values is triggered by the movement of any main assembly group: relocation of the machine along the rails; slewing, luffing, or shuttling of the boom. This was the first installation of such subsystem. Since the local data acquisition device monitors data using the on-change paradigm and the acceleration values are only captured when the machine is operating, the amount of data necessary to be transferred is kept to a minimum. This is of particular interest, because the internet connection within the operation location was not capable of constantly transferring an immense amount of data.

Signals from Machine PLC			**Additional Signals**	**Location**
$\sum$	**Analogue**	**Boolean**		
150	105	45	4+2	North America

Table 2.3: Signal Overview of Slew/Luff/Shuttle Ship Loader

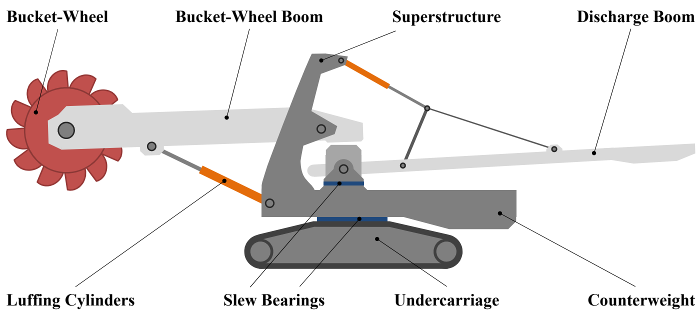
This machine holds a double pioneering role: The first local data acquisition device has been installed on this machine before it was later equipped with the first deployed subsystem.

2.8.3.3 Bucket-Wheel Excavator

A bucket-wheel excavator (BWE) is used to mine minerals[24] which do not exceed specific characteristics [35]. The system is installed on a compact-type BWE that is “mining” heterogeneous overburden with a significant range of material properties. It is required to remove this kind of material, so that other bucket-wheel excavators can mine the lignite previously covered by the overburden. The denomination “compact” is used because the counterweight of the machine is located underneath the discharge boom [36]. The particular machine is located in the Eastern part of Europe.

The BWE, as illustrated in Figure 2.18, consists of a rotating bucket-wheel which is attached to the tip of the main boom of the machine, the bucket-wheel boom. This boom is connected to a C-shaped frame. Mineral loaded onto the bucket-wheel boom conveyor by the bucket-wheel falls through a chute onto the conveyor of the discharge boom. The material is then loaded onto another conveyor belt, usually the one of a receiving bridge of a belt wagon. The bucket-wheel excavator is able to raise and lower (“luffing”) both booms up and down hydraulically and individually. Additionally, those booms are able to be slewed about the central vertical axis, as well separately from each other. Crawlers mounted on the undercarriage ensure the mobility of the machine.

(a) BWE in operation [11]



(b) BWE Principle Sketch

Figure 2.18: **Bucket-Wheel Excavator:** The BWE consists of a bucket-wheel boom, to which a rotating bucket-wheel is attached. Minerals mined from the ground get onto the conveyor belt on the bucket-wheel boom via a chute, which supports the unloading of the buckets of the wheel. The material is then transferred onto a conveyor mounted on the discharge boom, which loads it then on a subsequent machine, i.e., a belt wagon. Both booms can be lifted up and down via hydraulically-driven cylinders. All other drives are electrical. A slew bearing connects the superstructure with the undercarriage and permits the superstructure to rotate against the latter. Additionally, the discharge boom can be rotated separately from the bucket-wheel boom.

Since the bucket-wheel is mining material from the solid and heterogeneous ground, high and varying digging forces introduce significant tilting moments onto the slew bearing;

[24] The following definition for *mineral* is used: *“...any of various naturally occurring homogeneous substances (such as stone, coal, salt, sulfur, sand, petroleum, water, or natural gas) obtained usually from the ground”*, as retrieved from `https://www.merriam-webster.com/dictionary/mineral` on 2020-01-06.

especially the oscillations of this moment have an unfavourable impact on the life span of the slew bearing [9]. Hence, the implementation of a data acquisition system on this machine requires as well the consideration of a subsystem gathering data from sensors installed onto the slew bearing: the specifications are similar to the ones used for the vibration sub system described in section 2.8.3.2.

The local data acquisition device reads its data via OPC from an ABB PLC the machine is equipped with. In more detail, an OPC server is part of the HMI system of the machine, which is also from ABB. This allows the local data acquisition device to read data without the need of installing an additional OPC server application on it.

Potentially, this machine can provide at least as many channels as the ship loader in section 2.8.3.2. However, as evident from Table 2.4, the total number of monitored values is far below the previously discussed installations. The reason for the reduced count of signals was the intention of the client to limit the available variables, to restrict analyses to certain assembly groups of the machine. Although a holistic approach is preferred from a data science point of view, the available channels were of support to further develop *System I*.

Signals from Machine PLC			**Additional Signals**	**Location**
$\sum$	**Analogue**	**Boolean**		
104	64	40	4+2	Eastern Europe

Table 2.4: Signal Overview of Bucket-Wheel Excavator

For this particular installation, an additional modem was used to enable a 4G connection on the security device. This has become necessary, since experience of the client with network quality was significantly higher than with 3G. After a few months, the device stopped working. An investigation by the modem vendor yielded no root cause, because the logging system of the device failed. According to the vendor, the hardware itself did not exhibit any defect of the modules. The device was replaced, however, it was not mounted on the BWE again. It was found that the existing 3G connection on the security device was sufficient for all data transferring and remote access tasks.

The operation location of the BWE exhibited another speciality: All the machines of the mine, including the BWE with the data acquisition device, have been usually stopped and shut down in the late afternoon of working days. This is a consequence of heavy power grid usage, which forced mine operations to adopt their activities accordingly. This was mainly an issue for the drives and therefore the machine was only unable to move, with the control system and local data acquisition system still running. However, these power shut downs often provoked power fails of these systems as well. The software architecture and the hardware used were able to cope with this situation. Nonetheless, it remained unclear, if this was responsible for the defective 4G modem.

A significant advance in creating added value for a client has been achieved by the establishment of *expert review sessions*, constituting a novel approach of communication between the client and the manufacturer. Such sessions are usually planned to be organised in fixed intervals. The expert group is constituted by: subject matter experts of both parties to provide domain expertise of the client and of the machine manufacturer, as well as a data engineer who retrieves and prepares manual and semi-automatic results from the central data hub. Automatically generated results are also of support during these sessions. Whenever the data engineer or key user is unable to sufficiently fulfil requests of the expert group, external data scientists can be consulted, e.g., from universities. The main tasks of the expert review session members consist of comparing current load profiles with design profiles, analysis of response behaviour of the machine when it is exposed to different load scenarios, investigating machine utilisation, and, in general, advancing the exploration of data sets prepared as symbolised time series to identify future points of interest for the development of pre-generated evaluations. In case of an incident that requires immediate attention, such a constituted group of collaborating domain and data experts can be of significant benefit as well.

Towards the end of the observation period it had been decided to extend it to equip this machine with additional sensor systems for the slew bearing. The intention behind the application of supplementary sensor systems was to gather more data from the slew bearing. During a long-term trial run, the information and knowledge derived from these systems help to determine which system is the most suitable for the task. All the sensor systems discussed can be considered as subsystems and can be seamlessly integrated into *System I*.

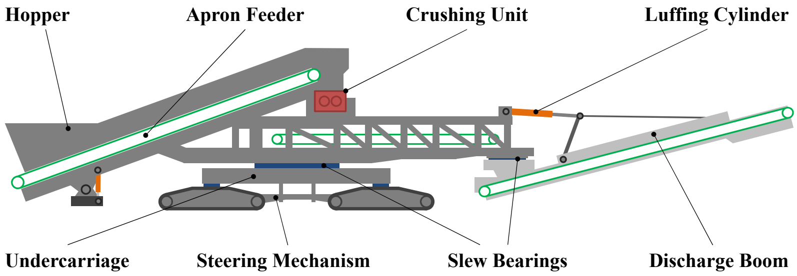
2.8.3.4 In-Pit Crushing and Conveying System

For this application, *System I* is used for a number of machines operated for the same iron ore mine in South America, designed as an *In-Pit Crushing and Conveying (IPCC)* system, as described in [37, 38]. In comparison to conventional mining of iron ore by *drilling and blasting*, the IPCC approach is truckless: the blasted material is directly loaded via a large-scale excavator onto a mobile crushing machine, located next to the pit face (mine face). The material is sized by this primary crusher and, hence, can be directly loaded onto a conveyor belt. The gap between the crushing machine and the belt can be bridged with a mobile belt wagon. Since the primary crushing of the material can be performed within the immediate proximity to the pit face, the process is more efficient than the conventional truck-based approach, especially because of the semi-continuous nature of the material processing and handling of IPCC. The material is conveyed to a centralised (secondary) crushing and screening hub, where the ore is separated from the waste material. Valuable ore is brought to further processing and overland transportation, while the remaining waste is transported to a dump, also in a continuous manner on conveyor belts. At the end of the belts, the waste is being backfilled in designated areas by using material dumpers, called *spreaders*. So, trucks (and fuel) are neither required for the transportation of iron ore, nor for dumping waste material. This makes the IPCC approach a more efficient and sustainable approach for specific applications [38]. However, the additional effort for conveyor belt shifting needs to be considered additionally.

2.8.3.4.1 Mobile Primary Crushing Systems This kind of machines is also referred to as *mobile sizing rig*, used in IPCC systems. Such a machine is considered fully mobile, since it can travel on crawlers mounted on the undercarriage. The machine can accommodate different kinds of crushers: sizers, double-roll, or hybrid crushers. The material is comminuted to a size that allows further material transportation via conveyor belts located near the pit face. The varying distance between the several stages of blasting are bridged by link conveyor machines; usually mobile belt wagons, which can extend the reach of the discharge end of the machine. Because of this, the conveyor belts are not required to be shifted to a new location so often, which further reduces operation efforts.

Figure 2.19b shows the main components of a mobile primary crushing machine, which has been equipped with *System I*. A large-scale excavator loads material next to the pit face into the hopper of the sizing rig. From there, an apron feeder (steel chain conveyor) transports the material to the top of the machine, where it is fed into the crushing unit. After the material is crushed into a specified maximum grain size, determined by the crushing unit's design, the material falls onto a conveyor belt to transport the material to the discharge boom, where another conveyor belt loads it either on the main conveyor line of the mine, or, more commonly, on the receiving boom of a belt wagon. The machine can move with crawlers and a steering mechanism: the crawler drives, as well as all the other drives of this machine, are designed as electric drives; the steering mechanism is based on hydraulics, as well as the luffing cylinder of the discharge boom. The rig can be rotated against the undercarriage to gain further flexibility. Additionally, the discharge boom can be slewed to extend the range of the discharge unit. Once the machine is moved to a location it resides at for operation, an outrigger attached to the bottom of the hopper is extended to support the machine at the hopper end.

(a) Mobile Primary Crushing System during relocation [23]



(b) Mobile Primary Crushing System Principle Sketch

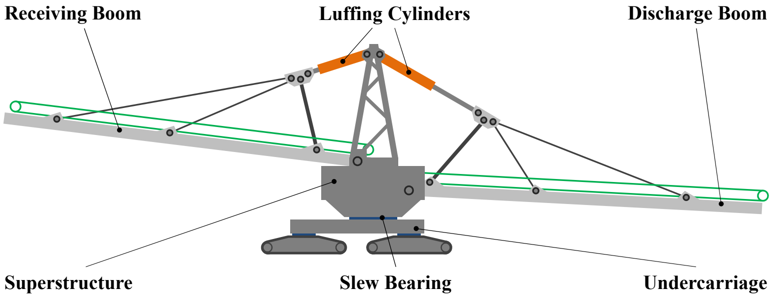
Figure 2.19: **Mobile Primary Crushing System:** Mined material is loaded into the hopper, an apron feeder (metal chain conveyor) supplies it to a crushing unit. Once the material is processed by this unit, it is transported to the discharge end of the machine via conveyor belts mounted on it. To extend the outreach of the discharge boom, it can be slewed as well as lifted up and down. The undercarriage consists of crawlers and a steering mechanism to gain flexibility during relocation. Once the machine reaches a designated location for operation, a support structure under the hopper (an outrigger) is extended to increase stability. All drives are electrical.

Signals from Machine PLC			Additional Signals	Location
$\sum$	Analogue	Boolean		
595	247	348	none	South America

Table 2.5: Signal Overview of Mobile Primary Crushing Unit

2.8.3.4.2 Belt Wagon A crawler-based belt wagon allows to bridge the gap between a mining machine, in this case a mobile sizing rig, and a conveyor belt, usually a mine conveyor. The type equipped with *System I* is used to transfer pre-crushed material gained from a mobile sizing rig (also equipped with the data acquisition system) to a conveyor belt, which conveys the material to the central crushing and screening facilities. As Figure 2.20b illustrates, the belt wagon consists of two separate booms: a receiving boom and a discharge boom. Both can be slewed about the vertical central axis and can be lifted up and down individually with hydraulically-driven luffing cylinders. The whole superstructure of the machine can rotate against the undercarriage, since a slew bearing is mounted in between. The tip of the receiving boom is designed to take up material loaded from the sizing rig. The discharge end is compatible with the conveyor belt frame, so that material can be loaded onto the mine conveyor line without uncontrolled spilling. Only electrical drives are used.

(a) Mobile Belt Wagon [39]



(b) Mobile Belt Wagon Principle Sketch

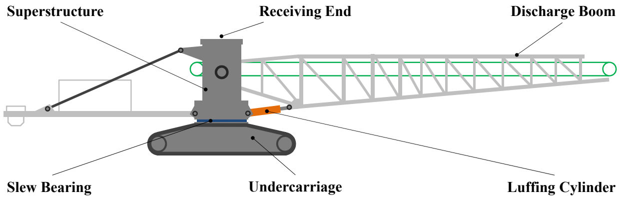
Figure 2.20: **Mobile Belt Wagon:** Consisting of a receiving boom and a discharge boom, the belt wagon bridges distances between a mining machine, e.g., a mobile sizing rig, and a conveyor belt on the ground. Hydraulic luffing cylinders lift the booms up and down. A slew bearing between the undercarriage and the superstructure allows the rotation of the machine. The belt wagon can be relocated by using its crawlers. The drives used for the belts and the crawlers are electrical.

Signals from Machine PLC			Additional Signals	Location
$\sum$	Analogue	Boolean		
376	159	217	none	South America

Table 2.6: Signal Overview of Mobile Belt Wagon

2.8.3.4.3 Tripper Spreader Once the ore of the mined material is separated from the waste, the latter is used to backfill part of the already mined areas. The waste is conveyed in a continuous manner to the dump area, where a mobile tripper spreader is used to distribute the material on a designated area. Mobile tripper cars are used to elevate material from ground conveyor level to the receiving end of a mobile tripper spreader. Part of such a tripper car can be seen on the left in Figure 2.21a, as it is delivering material to the mobile tripper spreader. The illustration in Figure 2.21b showcases the different assembly groups of the spreader. Crawlers of the undercarriage enable the machine to relocate itself to the area of operation. The superstructure can be rotated against the undercarriage, since they are connected via a slew bearing. A conveyor belt on the discharge boom allows the material, which is loaded onto the spreader via a tripper car at the receiving end, to be transported to the tip of the boom to be then dumped onto the designated area. A slewing motion of the whole superstructure enables an appropriate distribution of the backfilled material. As well as the belt wagon and the mobile sizing rigs, this machine is electrically driven. Hydraulics are used for the luffing mechanism.

(a) Mobile Tripper Spreader, part of the material-supplying mobile tripper car can be seen on the left [40] (typical)



(b) Mobile Tripper Spreader Principle Sketch

Figure 2.21: **Mobile Tripper Spreader:** A mobile tripper car is used to elevate waste material from a mine conveyor on the ground to the level of the receiving end of the spreader. The material is then dumped onto a designated area via a conveyor belt on the discharge boom, which can be lifted up and down hydraulically. A slew bearing ensures the superstructure can rotate about the central vertical machine axis. For relocation purposes, crawlers on the undercarriage are used. They are electrically driven, as well as the other drives used on this machine.

Signals from Machine PLC			**Additional Signals**	**Location**
$\sum$	**Analogue**	**Boolean**		
541	192	349	none	South America

Table 2.7: Signal Overview of Mobile Tripper Spreader

The local data acquisition device of *System I* has been initially installed for a mobile sizing rig, a mobile belt wagon, and a mobile tripper spreader. All three machines have been erected and commissioned at the same time. Data has been already captured for the early commissioning phase to benefit from data-driven support. The significantly high numbers of monitored signals, as listed in tables 2.5, 2.6 and 2.7, witness the necessity of capturing a reasonable amount of data during the commissioning phase. This data has proven to be beneficial to identify issues at an early stage of commissioning. *System I* added value as well in incident analysis during the initial ramp-up phase of production. In addition to the first sizing rig, another three machines of similar design have been assembled. Three additional data acquisition devices have been installed to monitor the data of those machines as well. The insights gained from the commissioning stage and early phase of operation of the first machine were of support for finalising preparations for operation of the three remaining sizing rigs.

The controllers of all the machines are connected via fibre optic cables to a centralised control cabin. Therefore, the six data acquisition devices have been installed in this cabin as well. The signals of the ABB controllers of the machines are provided to the data acquisition devices via a surrogate OPC server, which is provided by ABB as well. This server builds up a façade for the listening entities, so that the physical structure of the distributed controllers is projected onto a hierarchical tree of folders and nodes for the OPC tags.

The mobile internet connection on-site is unstable and not capable of establishing a VPN tunnel or transferring files. The client did not allow data transmission in a tunnelled manner by using their on-site wired internet access. These circumstances made it necessary to implement a service running on the local data acquisition device, which copies files from local database exports to USB flash drives. Those files on these drives need to be manually uploaded to the SFTP end of the data hub. This was planned to be performed once a day by an on-site engineer. In reality these uploads happened one to two times a week, which was usually still fitting the purpose.

2.8.4 Challenges, Limitations and Discussion

The operation locations of the machines equipped with the hardware for *System I* are distributed around the world. Hence, the system needs to exhibit a truly global nature, capable of collecting data from the distinct sources. Additionally, data from third-party systems can be ingested into the central data hub to provide the ability to run the same evaluations on time series collected by different kinds of acquisition devices.

The observed machines are operated in remote locations, implying requirements and limitations such as:

Connectivity: Although the used security device is capable of connecting to the internet of an existing on-site network via Ethernet, this option has not been provided by any of the operating companies. Mobile network connections represent a reasonable alternative to transfer data and to establish remote access via VPN tunnels. If no network connection is available at all, it will be necessary to provide data on USB memory drives: data downloaded from these drives is uploaded to the ingestion services of the data hub by a local user. Such a scenario bears the potential of delayed data ingestion for subsequent processes.

Watchdog Services: Remote operation locations require the installed data acquisition devices to run reliably and without user interaction. Watchdog services run on the deployed devices to ensure that all observed processes are running as expected, connections such as VPN tunnels are established and remain active, and subsystems are reachable and stay responsive. These watchdogs trigger subsequent pieces of software, which attempt to revive failing processes or runtimes of components.

Notifications: Devices and procedures exhibiting unusual patterns or exceeding thresholds, cause notifications to be sent out to engineers responsible for the operation of the local data acquisition devices. This is necessary to raise awareness for devices or processes that require attention or even maintenance actions; also tracking of those occurrences becomes more transparent. Even bugs in the software can be identified more easily, when issues provoke notifications to be sent out.

Reachability: Local maintenance personnel is required to solve issues on the data acquisition device in case of an unstable or non-existing internet connection inhibiting VPN tunnels from being established, or a hardware defect which cannot be fixed via remote access. Since auxiliary components became defective during operation at two different occasions, this is a threat that can only be solved by a concept based on redundant components. If data acquisition of a machine is crucial for operation or for dependent subsequent processes, it will be necessary to adapt the underlying concept of the system to overcome such issues.

To many clients, a device that acquires operational data of their machines and forwards it to the manufacturer's servers is novel ground. This circumstance becomes visible when operating companies restrict the channels to be observed based on their opinions and fears, rather than on the necessity for the projected output. Data ownership is a related topic: Who is allowed to manipulate data and in which manner? What happens with the stored data when a party withdraws from a contract for data acquisition? Clauses in contracts can be of support in handling such issues, however, there was no standard available across the industry or in related fields at the time of deployment of the data acquisition devices.

An unexpected social problem arose from the nature of the data: The time series acquired by *System I* allow not only to identify the performance of the machine, but also of its human operators. Exceptional care must be taken when analysing data in a manner that performance of operators can be concluded from it. Besides adding value by identifying which operators require training and also which colleagues can train them, such evaluations bear the potential of being misused to pursue interests of another nature.

One of the main advantages in using the concept of *System I* is the approach to not interfere with the existing control system of the machines. Avoiding any physical modification of the existing PLC is mandatory to not void any warranties or to not incur any liabilities. For some machine controllers, it is a requirement to flag the signals to be monitored in the runtime logic: this is performed either by the operating company's own engineers or by a consultant who adapts the software to provide the desired signals over the specified protocol.

2.9 Development of a Data Acquisition System for Application in the Geotechnical Engineering Sector

This section elaborates on the concept of a data acquisition framework for the geotechnical engineering sector within the construction industry. The development process was started in the mid of 2018 and was still ongoing in early 2020 when this thesis has been prepared. The works for this framework have been initiated for use cases of *Keller Grundbau Ges.m.b.H.* and affiliated companies, hereinafter referred to as *Keller*. The data acquisition system presented in this section is referred to as *System II*.

Keller is currently developing a state-of-the-art data acquisition and provisioning system; the author is participating in developing the hardware part and is responsible for advances in the data analysis domain, based on data gathered from Keller's machine fleet. The server portion of the concept has been accompanied by the author, however, he did not directly work on implementing the services.

Since *System II* is still under development, an intermediate solution has been established together with the *Chair of Automation of the University of Leoben*, referred to as *CoA*, and the company *eSENSEial Data Science GmbH*, referred to as *EDS*. This became necessary to collect data in a controlled manner to ensure the provision of a continuous stream of time series without interruptions.

2.9.1 Background, Status Quo and Motivation

The urge to establish a new means of data acquisition has its roots over 10 years ago, when Keller initiated investigations to develop a novel control and data acquisition system. The works have been performed on a KB 0 rig [41]. The newest version of this machine is discussed in detail later in section 2.9.4.1.

Keller operates a fleet of various machines of different types and sizes. They produce rigs and equipment on their own and also procure third-party products. Due to the diversity of the fleet, there are many different types of data acquisition systems in operation. Internal surveys showcased that 27 unique systems of 15 different vendors are currently used within Keller. The illustration in Figure 2.22 provides an overview about how value is currently created from many different data sources. The individual types of data acquisition systems (DAQ) installed on the machines get their data handled by separate systems, which range from local computers to cloud services and from on-premise servers to servers in remote or unknown locations. Another dimension of issues is added by the fact that the support for those systems is organised in a diversified manner by many resources, such as field engineers, superintendents, IT personnel, project managers, research and development persons and others. The support is not well organised and, hence, bears the potential of being an incubator for untracked errors and poor knowledge sharing.

The circumstances that originate from the diverse landscape of data collection create the need of a uniform system, which is capable of being installed on machines without any data recording capabilities and can replace the assortment of various existing acquisition units. Furthermore, an entire platform is planned to offer a data ecosystem, which users can utilise for applications across the company and around the globe. However, the global nature of such a project involves many stakeholders, which implies the necessity of many complex decision-making processes to derive common ground for standardised functionality and service provisioning within Keller globally.

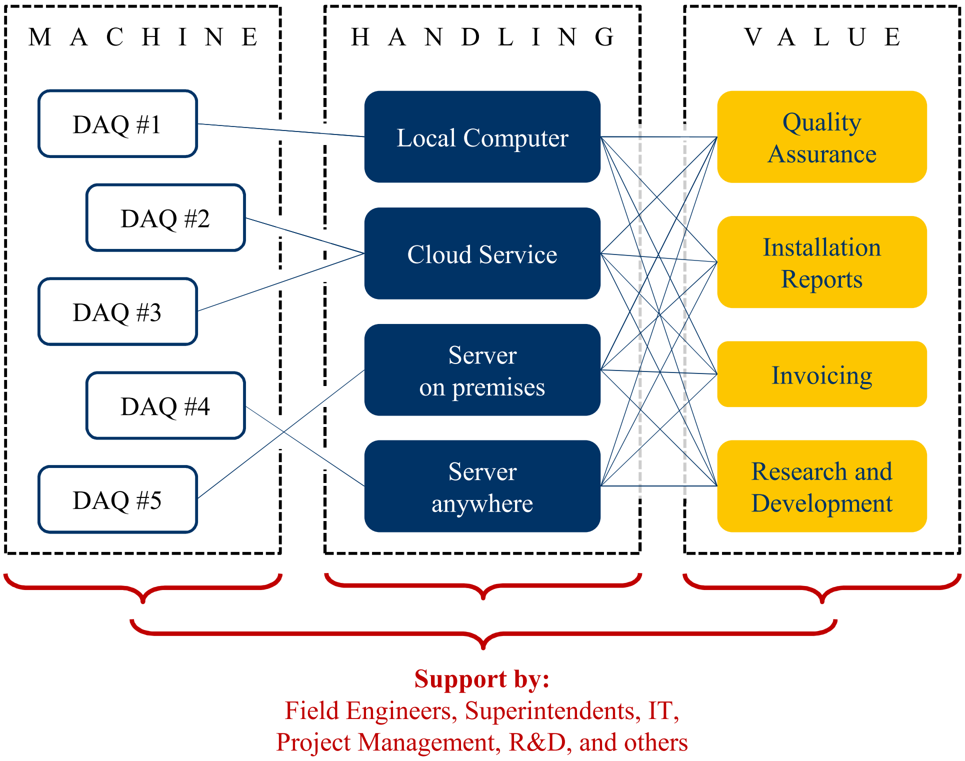


Figure 2.22: **Status Quo of Keller's Data Acquisition Systems:** Data from machines is used for many applications, such as quality assurance purposes, installation reports, invoicing, or R&D efforts. Currently, the input for these value-creating initiatives comes from numerous diverse systems. The data from these systems come from handling entities on local computers, in cloud environments or on servers: on-premise, in remote or unknown locations. Individual data acquisition units (DAQ) are used to collect data on machine level. Support for the entities of the processes is provided by an array of different resources, such as field engineers, site engineers (superintendents), IT supporters, project managers, R&D personnel, and others.

Since *System II* is still under development, an overview of the concept is given to have a comparison to the system established and proven for the mining and materials handling industry. The insights gained from the development of *System I* were incorporated into the design of *System II*. For the initial exploratory analyses of time series, data is used from the native data acquisition units that Keller has installed on their self-produced machines.

2.9.2 Concept for Entire Data Life Cycle Management

System II is based on a local data acquisition concept that is capable of collecting data from different sources. Figure 2.23 illustrates the used components. A device based on Microsoft® Windows[25] is considered as the core of the entire local data acquisition device. It receives the data via OPC UA from an interface that connects to the existing control system of the respective machine. Only one interface is used for a machine, based on the required protocol for the connection. There are three versions available:

- **PLC:** Sensors of a machine can be directly hard-wired to a PLC. Signals from protocols such as CAN or OPC UA can be additionally accommodated as well. The PLC is used to provide all incoming signals to the subsequent device – the Windows-based device – via OPC UA, taking over the task of a signal gateway.
- **Ethernet:** This solution allows the integration of machine controllers that provide channels via Ethernet-based protocols, e.g., Modbus TCP or OPC UA. The Ethernet solution is only considered to be a software service, no additional hardware is needed.
- **CAN:** A considerable number of machine manufacturers provide their data via a CAN protocol. A dedicated hardware gateway accepts data from a CAN source and maps the channels to OPC UA signals. The used CAN gateway is a small-scale PLC, which is only utilised for this conversion task.

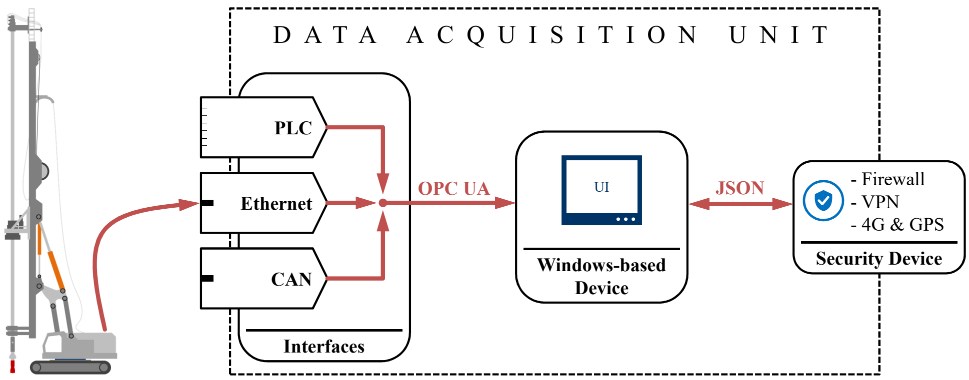


Figure 2.23: **Data Acquisition with *System II*:** Data is read from the control system of a machine by one of the required interfaces: *PLC* for a variety of protocols and for additional hard-wired sensors, *Ethernet* for Ethernet-based protocols, e.g., OPC UA or Modbus, and *CAN* for machines providing data via CAN bus. The used interface converts the input to tags readable via OPC UA. A software running on a Windows-based device reads those OPC UA tags and buffers them locally. After an element has been installed in the ground, this device sends a file in JSON format to a cloud environment via a security device. Besides modem capabilities, the latter offers low-precision GPS location data.

Once the data is available on the Windows-based device, which is a rugged industrial tablet, it is buffered and provided to a cloud-based end point via a security device. This

[25] The use of Microsoft® Windows benefits the integration into the Microsoft® Azure platform, which this unit communicates with.

security device offers firewall and VPN tunnelling capabilities. The internal modem uses 4G technology for mobile network connections and provides low-precision GPS location. The time series are provided to the cloud in a file format based on JSON[26]. Besides the file containing data of an installed element, there is a rig provisioning file, also in JSON format, which provides design data for the elements to be produced, such as name, design depth and design location. Hence, the Windows-based device of the data acquisition unit must be capable of bidirectional communication. JSON is used as a format for all files that are in use for *System II*, including configuration files. One of the major advantages of the JSON format is that the data is saved in attribute-value pairs: interpreters do not need to know about the structure of the file prior to loading it, the structure is contained in the file itself. Additionally, the data stays human-readable. In contrast, CSV files are used for Keller's own data acquisition systems, which are still actively used until a production version of *System II* is deployed on a global scale.

The fundamental structure of a JSON file for transferring data of an installed element is listed as follows:

```
{
    "header":{...}
    "trace":{
        "parameters":[...],
        "units":[...],
        "data":[
            {...}
        ]
    },
    "event":{
        "parameters":[...],
        "units":[...],
        "data":[
            {...}
        ]
    },
    "elementComplete":true
}
```

The main objects used within the file are `header`, `trace`, as well as `event`. The file closes with a Boolean variable `elementComplete` to indicate that the file has been written entirely, since the file is not closed before production has been finished for the element. To maintain clarity and to provide a reasonable overview of the different parts, they are looked at in more detail individually.

As can be seen in the following listing, the header part holds data relevant for the entire element, such as project and element identification, information about used equipment, as well as design data for the produced element.

[26]JSON . . . JavaScript Object Notation

```
{
    "projectId":4,
    ...
    "elementName":"1",
    "header":{
        "rigEquipmentNumber":"1113333",
        ...
        "targetMaxDepth":10.9
    }
}
```

The trace data object corresponds to the time series acquired during production of the element. It consists of three main parts: `parameters` for an order-relevant list of the used channels, `units` to specify the respective SI units, and `data`, which contains the time series data for the parameters and in the units provided. For the latter, the timestamp is an object separate from an array of values. Each entry in the `data` object resembles one sample of all available channels. An example of the structure is given:

```
{
    "trace":{
        "parameters":[
            "depth",
            ...
            "amperage"
        ],
        "units":[
            "m",
            ...
            "A"
        ],
        "data":[
            {
                "timeStamp":"2019-07-09 14:10:11.185",
                "values":[
                    0.000000,
                    ...
                    0.000000
                ]
            },
            {
                "timeStamp":"2019-07-09 14:10:12.185",
                "values":[
                    0.140000,
                    ...
                    0.000000
                ]
            }
        ]
    }
}
```

In case there are different sampling rates required, another trace object would be used in parallel to the existing one.

In addition to header information and trace data, there is the option to include an `event` object. It has a structure similar to the one of the trace object, however, it contains relevant data per event occurrence rather than per fixed sampling interval. A listing for this object is provided:

```
{
    "event":{
        "parameters":[
            "eventTypeId",
            "eventTypeName",
            "eventDesc"
        ],
        "units":[
            "",
            "",
            ""
        ],
        "data":[
            {
                "timeStamp":"2019-07-09 14:10:11.185",
                "values":[
                    "1",
                    "Start Activity",
                    "Element Start"
                ]
            },
            {
                "timeStamp":"2019-07-09 14:10:12.185",
                "values":[
                    "4",
                    "Stop Activity",
                    "Element Complete"
                ]
            }
        ]
    }
}
```

Data Acquisition on machine or site level is only one piece of the concept. As illustrated in Figure 2.24, there are two more major segments required: data handling, as well as consumption services. A cloud gateway is used for accepting data from the local data acquisition units of the machines in the field. Since the adopted cloud environment is based on Microsoft® Azure technology, Azure IoT Hub offers such gateway services. In case of an unstable or not existing internet connection, data can be provided by copying them from the Windows-based device and uploading them directly to the cloud environment; since all the necessary information is contained in the file, this can be performed in a single drag-and-drop manner to reduce the effort on the upload side for the users. After the JSON file arrived, ingestion services check quality and integrity of both, the file itself and its content. Data and information from other sources, such as project details of

ERP[27] systems or metadata provided by domain experts, is merged with the incoming file. Storage services manage and distribute the file and its content to the respective storage locations: trace data is put into a non-relational database, while aggregate data and header information per element are stored in a conventional SQL database. The separation of both different kinds of data benefits an efficient data access methodology across the variety of provision services. They pull data from all available sources within the cloud environment to fulfil the requests of the enquiring consumption service.

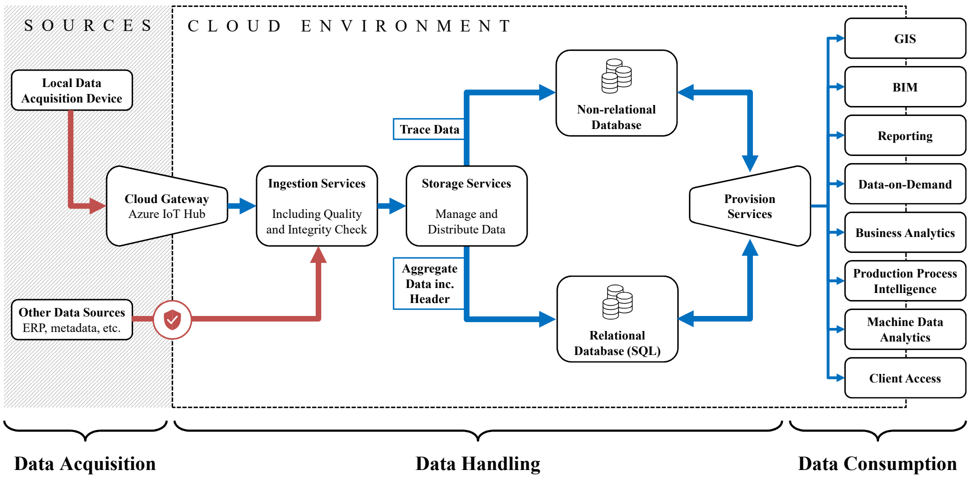


Figure 2.24: **Data Flow and Ingestion:** Data emanating from a machine is collected by a local data acquisition device. It sends files containing the time series to the ingestion services in a cloud environment via a designated cloud gateway. Additional metadata from other sources is also provided to the ingestion process, where all data is combined as well as checked for quality and integrity. Once a file passed this procedure, it is forwarded to storage services, where its contents are divided into trace data, which is stored in a non-relational database, and aggregate data. The latter is pushed into a relational database. Data is pulled from both databases as needed for the several data consumption services. Supervising provision services manage the data input for those means of data consumption.

The provision services for data consumption consist of a significant array of individual items creating added value. Major use cases are showcased in Figure 2.24, which are:

- **GIS:** Geographic Information System, used for manipulating and visualising spatial data;
- **BIM:** Building Information Modelling, offers capabilities of digital twins of construction sites in a passive manner;
- **Reporting:** Installation as well as production summary reports can be generated in a centralised and standardised manner;
- **Data-on-Demand:** Users of distinctive interest groups within the company can enquire data as they need it for specific tasks;

[27]ERP . . . Enterprise Resource Planning

- **Business Analytics:** Insights for business development and management levels can be provided via tools such as Microsoft® Power BI;
- **Production Process Intelligence:** Research and development efforts, such as the work elaborated in the upcoming chapters, can be of benefit in developing tools for gaining insights on the processes during production;
- **Machine Data Analytics:** Besides analytics focused on the production processes, benefit can be generated by analysing the operation of the machines;
- **Client Access:** After certain portions of data have been approved, reports and other information can be provided to clients via direct access to the cloud environment.

2.9.3 Intermediate Solution for Acquiring Data in a Continuous Manner

System II has not yet been deployed for use within a production environment. Current analyses can only be performed on data sets provided by already running data acquisition systems. Files generated by Keller's own systems are used for exploratory evaluations. These files exhibit disadvantages in terms of sampling rate, which is fixed at 1 Hz, as well as channel availability and providing element-wise data only. Together with EDS, an intermediate solution has been established, which is based on a *Kunbus RevPi Core 3* as a small-scale data acquisition device. It accepts signals via OPC UA in a continuous manner at a sampling rate of 10 Hz. The sampling rate is limited by the update rate of the dedicated PLC tasks, not by the RevPi itself. For this solution the data is send to an environment similar to what has been described for *System I*. Data is buffered locally and sent securely to the receiving services via the existing internet connection of the rig, whenever it is available. This intermediate solution is used for two jet grouting rigs, as discussed in the upcoming section, and is in preparation for machines used for vibro ground improvement.

2.9.4 Techniques and Machines of Relevance for the Implementation of *System II*

2.9.4.1 Jet Grouting

Jet Grouting, also called Soilcrete® within Keller, is a technique of geotechnical engineering that involves in situ mixing of soil with grout to produce an element that either transmits load, creates a sealed[28] barrier to prevent uncontrolled draining, or remediates soil pollutions with additional reactants. The jet grouting technique is used to produce geotechnical elements for a wide range of applications [43, 44]:

- **Stabilisation of Structures and Buildings:** Underpinnings, deep foundations, tunnelling support, foundation enhancement and extension, and lateral support structures;

- **Sealing Structures:** Cut-off walls and panels, bottom seals, vaulted slabs, sealing membranes, dam core seals, pile wall infill, and sealing of joints and gaps between structures;

- **Environmental Geotechnics:** Soil contaminated with toxic substances can be treated with Keller's Halocrete® technique. A suitable neutralising reactant is mixed with grout to trigger a chemical reaction during the grouting phase. Based on the characteristics of the contamination, the reactant used can be an oxidising or reducing agent. For instance, remediation of chlorinated hydrocarbons pollutions is achieved by using a potassium permanganate reactant. It is also possible to enclose contaminated soil with jet grouting elements to prevent leakage into surrounding, non-contaminated soil.

Different geometrical shapes of the elements can be produced: full columns, partial columns (half, quarter, et cetera), or lamella-shaped elements, e.g., for sealing walls. The absolute and relative locations, as well as the orientations of the installed elements in the ground, are of fundamental importance for the proper function of the geotechnical system the individual elements are part of.

Prior to considerations of which shapes and designs are appropriate for a particular application, the soil conditions determine which techniques are suitable. Although jet grouting exhibits the widest range of application of grouting techniques – from loose sediments to stiff clay, see Figure 2.25 – it has problems with highly coarse soil; it cannot be properly mixed with the grout to create an element with sufficient characteristics. The function principle of the technique is based on mixing soil with appropriate grout to form a product with new specifications. Still, of all grouting techniques it offers the highest versatility in application. [43, 45, cf.]

[28] It must be pointed out that the term "sealed" refers to the technical interpretation of water stop capabilities. Permeability coefficients of 10^{-10} m/s can be reached [42], indicating a virtually impervious barrier. This is true for the individual element, however, it needs to be taken into account that the system permeability lies above the values of the individual elements.

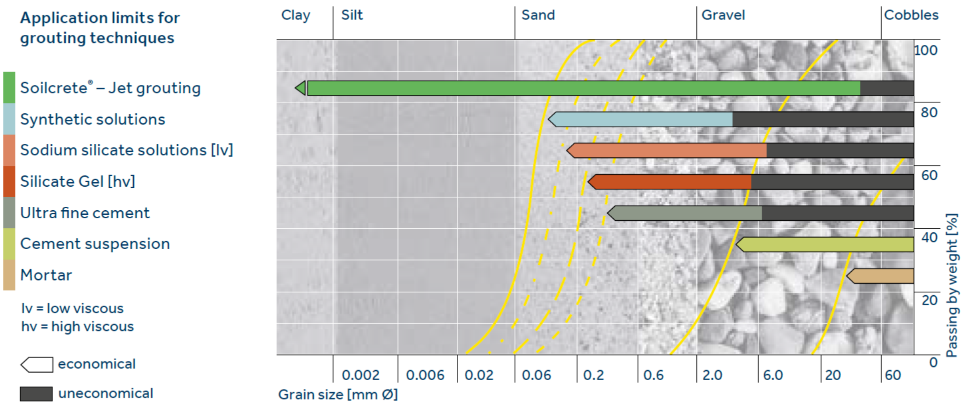


Figure 2.25: **Application Limits for Grouting Techniques:** In comparison to other established grouting techniques, jet grouting exhibits a wide area of application in soils ranging from loose sediments to clay where it adds economic value. [43, 45, cf.]

An overview of the jet grouting process is illustrated in Figure 2.26. Apart from the drilling rig (only its drill string is visible in the figure), there are several additional pieces of equipment required. Compressors and pumps are necessary to provide air and water at the specified pressures and flow rates. Cement or bentonite is sourced to mixing units from silos, additives are blended with the silo material and water. The resulting grout – or cement suspension – is pumped via a high-pressure pumping station over hoses to the drill string. The hoses are connected to the drill string via connections on a swivel, which is mounted on the top end of the drill string. Depending on the Soilcrete® process used, air, water and grout exit via nozzles at the bottom end of the drill string. The nozzles are either directly installed on the drill bit or on a monitor at the end of the drill string, depending on the chosen configuration of machine-related specifics and of the to-be-produced elements in the present soil conditions. First of all, a hole is drilled at the designated location to produce a jet grouting element (*Phase 1 – Drilling*). GPS systems can be of assistance to position the rig. Flush drilling is used to stabilise the borehole and to prevent it from collapsing. Grout with related or also similar characteristics is used to flush. The end depth of each drill has to consider the lengths of the drill bit, the reamer and the position of the nozzles of the monitor. After the final bore depth has been reached, the drill string is continuously rotated and pulled up while either a high-pressure grout or water jet cuts the soil around the borehole (*Phase 2 – Jetting*). This erosion of the soil is followed by bringing in grout via the grout jet itself or via an additional outlet on the bottom end of the monitor (*Phase 3 – Grouting*). The emerging turbulences benefit a sufficient mixture of soil with grout to result in a Soilcrete® element. Production planning is necessary to ensure an appropriate installation order of adjacent elements. Depending on the system the elements are part of, new columns can be either installed fresh in fresh or fresh in firm (*Phase 4 – Extension*). The installation sequence of the elements is vital to a functioning geotechnical system, especially when secondary structures, such as sheet pile walls, are part of the system. The backflow from all phases is collected and can be treated to recycle the water used in it.

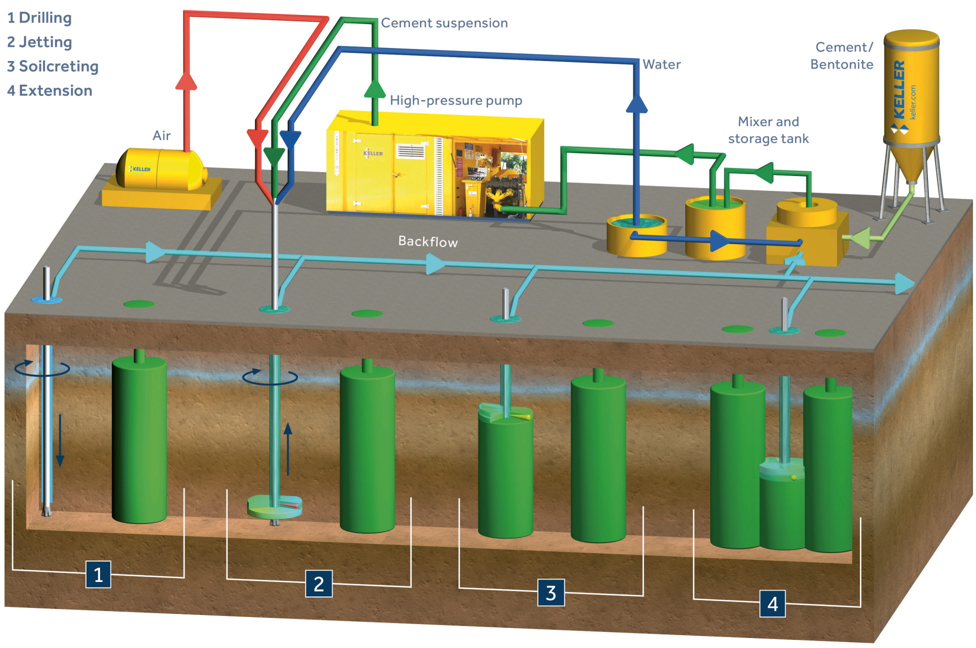
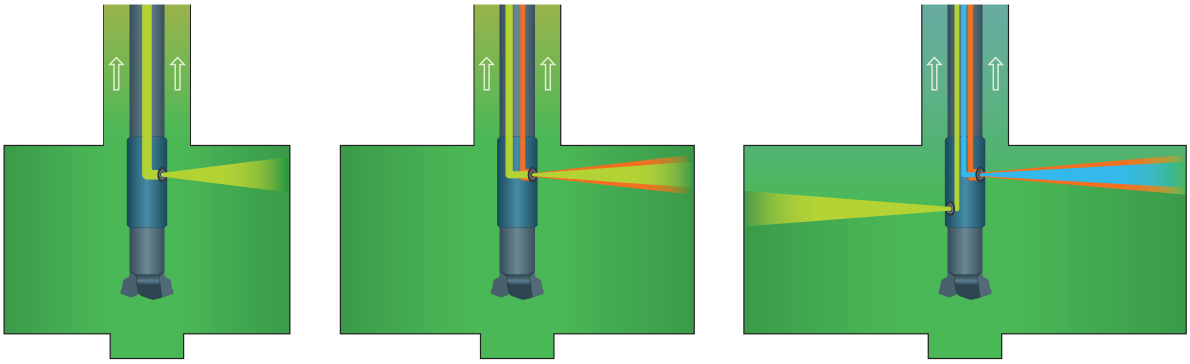


Figure 2.26: **Jet Grouting Process:** Several pieces of equipment are required on a jet grouting site. Apart from the drilling rig (not shown in the figure; only drill string visible), there is the need for a high-pressure pumping station: it is used to deliver grout (cement suspension) via the drill string down the hole to the nozzle(s) at high pressure. Besides the rig operator, there is a second person operating the pumping station and the grout mixing and supplying units; the cement suspension is mixed directly on-site. Air and water are also provided at required pressures and flow rates. The production process itself starts with drilling a borehole down to the necessary depth by using cement suspension at low pressure rates to ensure proper flush drilling (*Phase 1 – Drilling*). In a next step, the soil is cut with either grout or water jets at high pressures, depending on the used process (*Phase 2 – Jetting*). For this, the drill string is rotated and gradually pulled out. While the soil is eroded, cement suspension is brought in by either the grout jet itself or via another outlet (*Phase 3 – Grouting*). The turbulences during the erosion ensure adequate mixing of the soil with the grout. The produced elements are often part of a geotechnical system consisting of multiple elements (*Phase 4 – Extension*). [43]

Different subtechniques are used for Soilcrete®. The main ones used by Keller are discussed in Figure 2.27. They differ in how the most efficient erosion can be accomplished in respective soils to produce elements with expected characteristics. The bottom end of a drill string with a monitor with high-pressure water jets is pictured in Figure 2.28.

The high pressures and flow rates of the grout are achieved by high-pressure plunger pump stations. Long hoses are usually used, since these stations are not mobile in terms of simple relocation without additional equipment. Limited accessibility for works carried out within buildings or in areas with restricted access require many metres of hosing. Additionally, it is necessary for the drill rigs to possess a sufficient amount of flexibility to install all elements on sizeable sites – for this, long supply hoses are required. The pump needs to compensate for the loss of pressure due to the flow resistance of the hoses and bends, before the grout can exit the supply line via the nozzle at the tip of the drill string. Grout pressures of approximately 400 bar at flow rates of around 700 l/min are established in jet grouting; those numbers correspond to the value readings at the pumping station [46].



(a) **Soilcrete® S:** A single grout jet is used with an exit velocity of up to 200 m/s. This jet cuts the soil and mixes it with the grout. Soilcrete® elements of small and medium diameters can be achieved.

(b) **Soilcrete® D:** The grout jet is additionally shrouded by air to increase the erosion efficiency, resulting in increased jet grouting column sizes.

(c) **Soilcrete® T:** This process is mainly used for installing elements in cohesive soils. In contrast to the S and D processes, an air shrouded water jet with up to 200 m/s exit velocity cuts the soil. Grout is injected simultaneously through a nozzle at a pressure ranging around 15 bar. A variation of this process utilises a high-pressure grout jet instead.

Figure 2.27: **Soilcrete® Processes and Monitor Configurations:** There are different configurations for the monitor available for jet grouting, the main variations are S, D, and T. In the figures the grout is illustrated in green, air in red and water in blue. Backflow consisting of a mixture of grout and soil (and water for version T) exits through the annular space between the drill string and the borehole wall. [43, cf.]

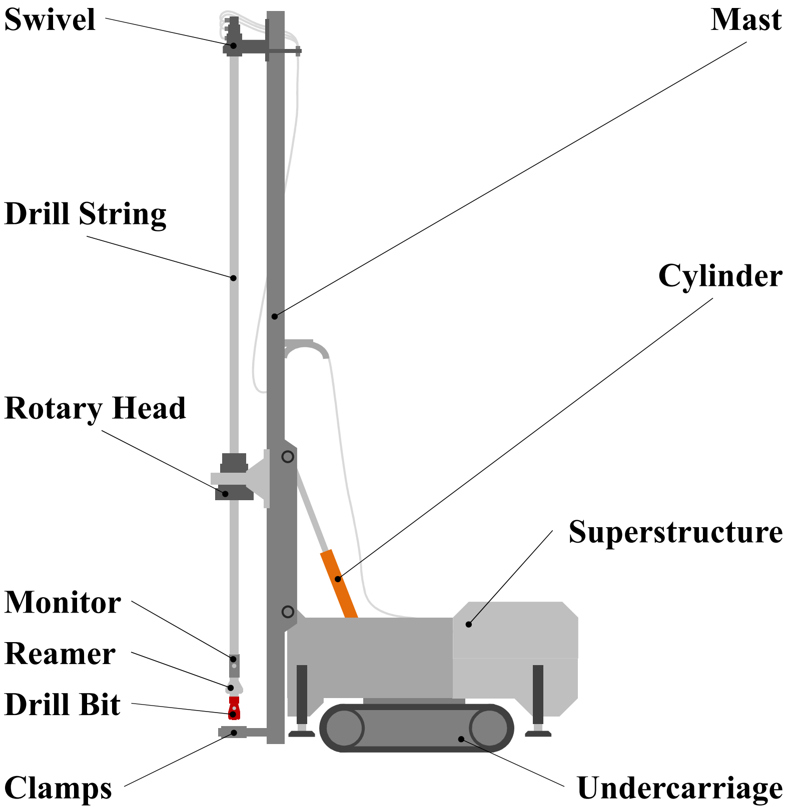
Down-the-hole (DTH) hammers are used for the efficient penetration of hard soil layers. Depending on the specific characteristics of such layers they can be driven by compressed air or water. The supply chambers of the drill string are used for driving the hammer. In many cases, the hammers have nozzles installed to make the use of a monitor obsolete; some hammers also have reamers.

Figure 2.28: **Jet Grouting Drill Bit with Jet Nozzles** (*Courtesy of Keller*)**:** The figure shows the location of the jet nozzles on the monitor at the bottom end of a drill string. Water exits the nozzles at high pressure. This is only one out of many different configurations of jet nozzle size, number and spatial distribution, as described in Figure 2.27.

Keller's fleet of jet grouting machines consists of different types and sizes of drilling rigs. The two machines of interest for this work were developed, designed and produced by the equipment department of Keller and have their own data acquisition system installed. As pointed out previously, those data collection systems capture data only during the production of an element and the operator is in control of when data is acquired and for how long.

The first machine is a KB 0-5, as shown in Figure 2.29: "KB 0" denominates the type, while the "5" indicates the version.

(a) Jet Grouting Rig KB 0-5 (*Courtesy of Keller*)



(b) Jet Grouting Rig KB 0-5 Principle Sketch

Figure 2.29: **Jet Grouting Rig KB 0-5**, a small-scale rig used for jet grouting. A rotary head is mounted on the mast, the feed mechanism pushes the head up or down to enable a drill string to penetrate the ground. Rotation as well as feed is controlled hydraulically, such as all other major functions of the machine, e.g., crawlers, clamps and mast positioning cylinders. The hydraulic pump is powered by a Diesel engine; an electrical version without a combustion engine is available. The rig operator stands next to the machine and has the option of a remote control. Media required to perform jet grouting jobs, such as water and grout, are pumped from a separate central high-pressure pumping station; air comes from an additional compressor if needed.

The KB 0-5 rig is part of the small-scale jet grouting fleet of Keller. The operator is standing next to the machine to control it, either with the HMI and joysticks at the side of the machine or with a remote control. The drill string consists of drill pipes which are screwed together. They come in various lengths. The drill string can be extended for a deeper reach by adding pipes during drilling. The drill string can go through the rotary head, a clamping mechanism on the inside of the head holds the drill string during drilling. Another clamping mechanism at the bottom end of the mast supports the drill string during adding or removing drill pipes. Especially during retracting the drill string from the hole – as this is the case during jetting/grouting – it is mandatory to hold the bottom part of the drill string, before a pipe connection is loosened with the use of the rotary head. At the top end of the drill string there is a swivel, with which the hoses for grout, air and water are connected to the respective chamber of the drill pipes. The rotary head can travel up and down the mast by using a hydraulic feed mechanism. The inclination of the mast is adjusted by a cylinder. A Diesel-driven hydraulic unit controls all main actors, such as the rotary head, the feed mechanism, the mast cylinder, the clamps, as well as the crawler drives. A fully electrical version of this machine has been developed as well.

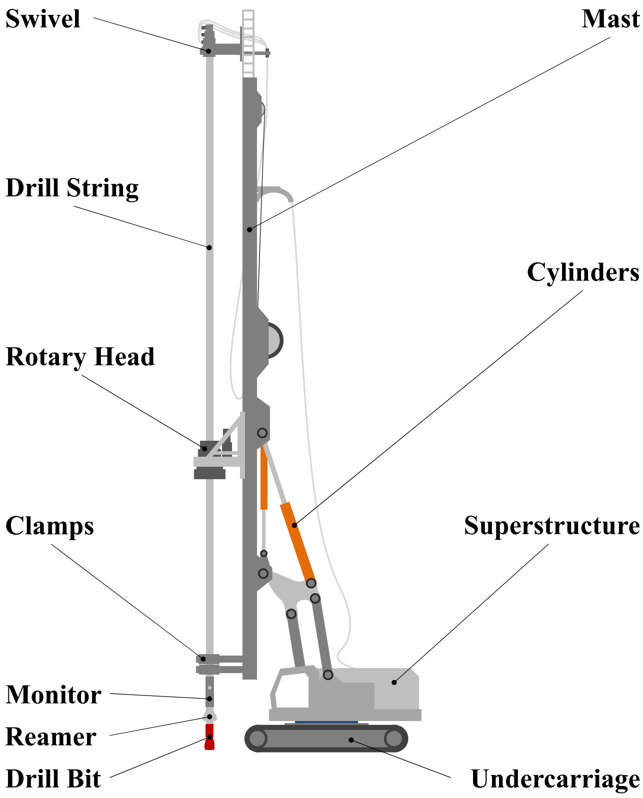
An overview of the available and used channels for this machine is given in Table 2.8. A total of 25 channels is accessible on the machine PLC via OPC UA, whereas 80% provide analogue values. Additional 10 channels contribute metadata, such as mode statuses and control system time references.

Signals from Machine PLC			**Additional Metadata Channels**
$\sum$	**Analogue**	**Boolean**	
25	20	5	10

Table 2.8: Signal Overview of KB 0-5

The drilling rig KB 6 of version 2 is shown in Figure 2.30. This rig is the biggest of Keller's jet grouting fleet at the time this thesis has been completed. Version 3 is available as well to incorporate improvements and modifications. The working principle is similar to the KB 0, however, the kinematics are more sophisticated and the reachable depth is significantly deeper. Additionally, the drill string can be longer to keep the necessity for adding or removing pipes during the production process low or eliminates it at all. Since the base carrier machine is comparable to the one of an excavator, the operator can sit in a cabin to control the rig. A Diesel engine is used for the hydraulic unit that drives all cylinders of the kinematic system, the rotary head, the clamps, the slewing mechanisms as well as the crawlers of the base machine.

(a) Jet Grouting Rig KB 6-2 (*Courtesy of Keller*)



(b) Jet Grouting Rig KB 6-2 Principle Sketch

Figure 2.30: **Jet Grouting Rig KB 6-2**, currently the biggest rig Keller produces for its entities: power (rotation and feed), reachable depth and drill rod diameter excels the specifications of the smaller rig, a KB 0. The kinematics system allows for more flexibility and increases productivity on appropriate sites. This machine is as well powered by a Diesel engine. The rig operator is sitting in a cabin. Media required for jet grouting (water, cement suspension, air) is as well pumped from separate units.

The KB 6-2 provides 28 channels over OPC UA as shown in Table 2.9, 23 of those are of analogue nature. Again, a sum of additional 10 signals is available for collecting metadata during runtime.

Signals from Machine PLC			**Additional Metadata Channels**
$\sum$	**Analogue**	**Boolean**	
28	23	5	10

Table 2.9: Signal Overview of KB 6-2

Both machines, the KB 0 as well as the KB 6, have an existing data acquisition system installed, however, it comes with the limitations mentioned in section 2.9.1. Since *System II* is still in development, the intermediate solution has been installed on both rigs to gather time series in a continuous manner throughout the uptime of the machine.

2.9.4.2 Vibro Ground Improvement

Another important technique for Keller is Vibro Ground Improvement. It is mainly used for foundations on soils which exhibit low bearing capacity [47]. The subtechniques mainly used, *Vibro Compaction* as well as *Vibro Replacement*, cover a wide range of different soils – as visible in Figure 2.31. One of the most significant differences to the jet grouting technique is that elements installed by vibro techniques are only capable of bearing loads when they are part of a system; only clusters of elements can be used to transmit loads.

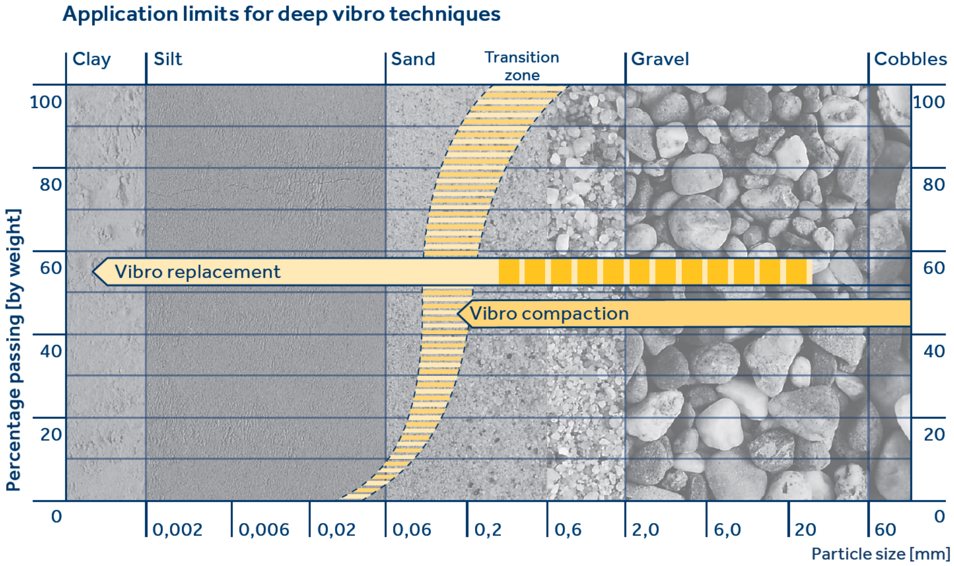


Figure 2.31: **Application Limits for Vibro Ground Improvement:** The main vibro techniques, *Vibro Compaction* and *Vibro Replacement*, cover a wide range of application. Although vibro replacement can be used for more coarse soils as well, it is not the most efficient subtechnique for this soil (dotted line). The limits for vibro compaction are more defined, since a soil with a high amount of fine particles cannot be compacted in a reasonable manner. [45, 47, cf.]

Although both being part of vibro ground improvement, the two subtechniques follow fundamentally different working principles (see [48]). The process of the first subtechnique, vibro compaction, is illustrated in Figure 2.32: three steps showcase the distinct phases from left to right. During *penetration*, the vibrator is lowered into the ground, supported by water[29] exiting the bottom end of the vibrator to benefit forming an annular gap. If needed, the vibrator is oscillated up and down (*Phase 1*). When reaching the bearable layer (the end depth), the water flow is reduced. The next step, *compaction*, is carried out by a step-wise lifting of the vibrator; at each lifting step the surrounding soil is compacted (*Phase 2*). The increased power consumption of the vibrator indicates an increase in soil density at the respective step. The power ratings of the used vibrators reach values of 200 kW and beyond: significant effort is put into research about the relation of the energy brought in by the vibrator and the actual compaction of the ground [49]. A crater

[29] Also air is used instead of water for specific applications [49].

develops around the hole as a direct consequence of a more compact soil – it is backfilled with material similar to the original soil (*Phase 3*). Usually, a void up to 10 % of the treated soil volume is created and needs to be backfilled. [45]

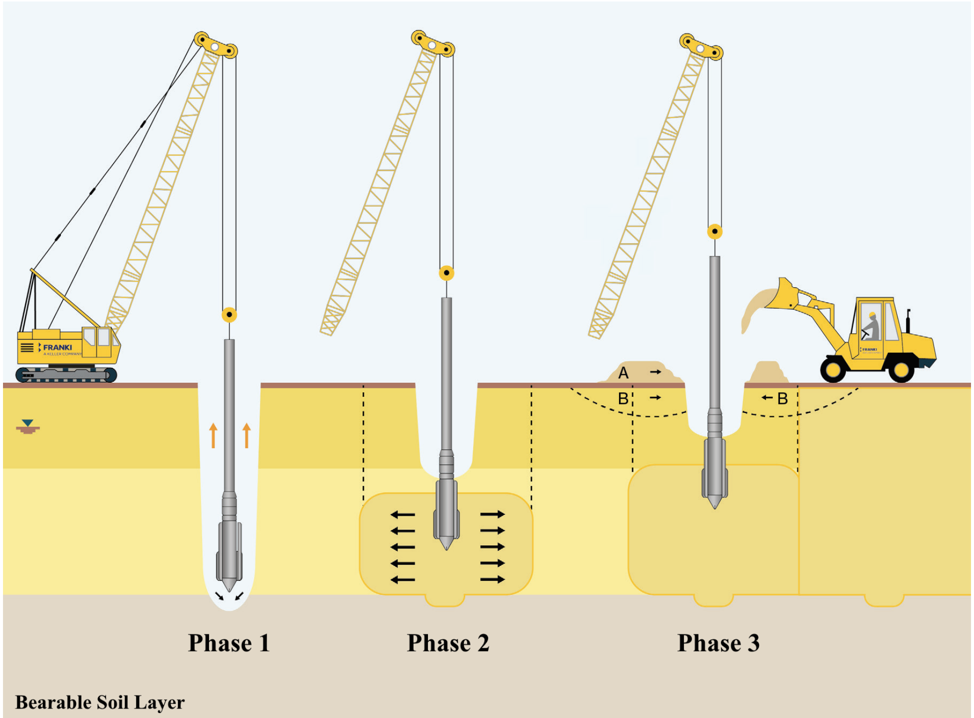


Figure 2.32: **Vibro Compaction Process:** In a primary step, the vibrator *penetrates* the ground with the support of water, which exits at the bottom (*Phase 1*). When the bearable soil layer is reached, the water flow is reduced and the vibrator is started to be lifted step-wise. During this *compaction* phase, the soil around the vibrator is getting denser (*Phase 2*). The crater which develops around the hole is backfilled with material with same characteristics as the original soil (*Phase 3*). [45, 47, cf.]

Vibro compaction can be performed with a rig specifically manufactured by Keller, called *vibrocat*, or with a free-hung vibrator on a crane, such as shown in Figure 2.32.

In contrast to the aforementioned, the process phases of vibro replacement are showcased in Figure 2.33. The used vibrator is different from the one used for vibro compaction in many aspects: it is attached to a hollow vibrator string and it is hollow itself to form a chamber, so that filling material (gravel) can be feed through the string and the vibrator to the bottom. This is the reason why this process is also called "bottom feed" [45]. During the *penetration* phase, the vibrator is lowered into the ground with the support of pressured air that flows through the chamber of the string and vibrator to ensure that soil is not clogging the chamber exit at the tip of the vibrator (*Phase 1*). The gravel material is loaded into the skip of the machine via a wheel loader. This skip can travel along the mast independently from the vibrator string. It is used to fill material into the hollow chamber of the string at the top of it, using an air chamber lock mechanism. Usually, several loads of skips are used for a produced element. The gravel is exiting the string at the tip of the vibrator after it has been lifted a specified height: the material fills the now existing cavity resulting from the retraction of the string. Afterwards, the vibrator is pulled down to compact the loose material (*Phase 2*). Such compaction steps are repeated, until the full design length of the element has been reached. One skip load serves several compaction steps. To ensure a proper load distribution across all installed elements, the top layer needs to be compacted specifically or a load distribution layer has to be installed (*Phase 3*).

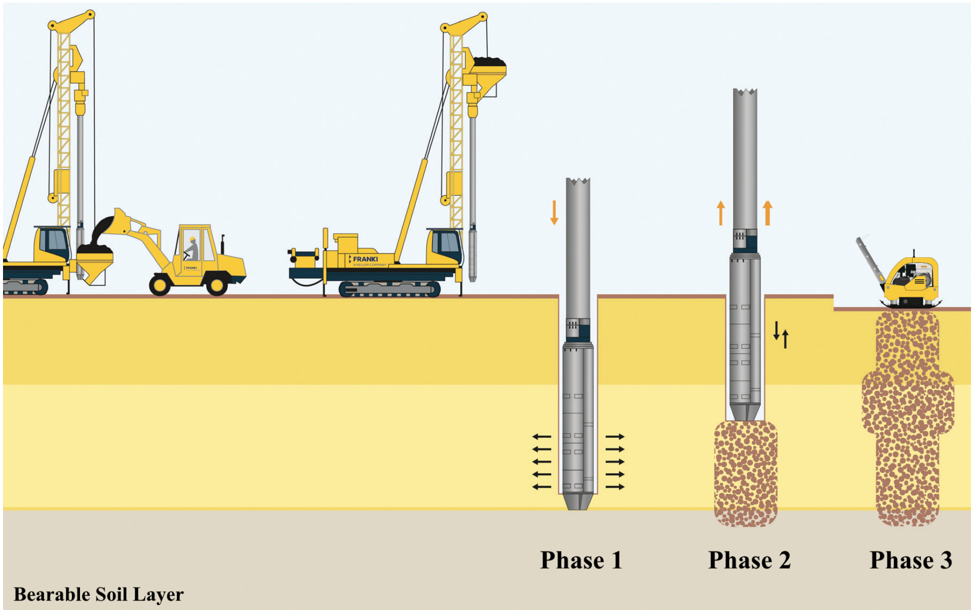
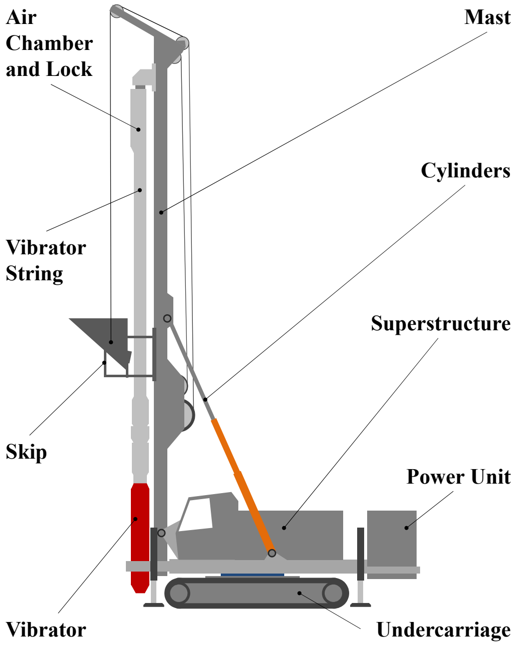


Figure 2.33: **Vibro Replacement Process:** At first, the vibrator penetrates the ground until bearable soil is reached (*Phase 1*). The whole vibrator string is hollow, so that filling material (gravel) can travel from the lock system at the top to the bottom of the vibrator. The gravel material is filled into this chamber by a skip, which can travel along the mast. A wheel loader fills the skip with material. Pressured air is used to avoid soil from clogging the bottom outlet of the vibrator. After reaching the end depth, the vibrator string is retracted by a defined step height; the gravel fills the cavity. Then the vibrator is pushed down again to compact the loose gravel. This compaction step is repeated to form an element (*Phase 2*). Finally, the top layer is compacted or a load distribution layer is used (*Phase 3*). [45, 47, cf.]

A machine such as illustrated in Figure 2.34 is used to install elements with the vibro replacement technique. This so-called “vibrocat” also bears the potential to mount equipment required for performing vibro compaction. The machine is driven hydraulically as well as with individual, electrically-driven winch systems for the vibrator string and the skip. The considerable amount of power needed to drive the vibrator comes from a separate, Diesel-driven power unit. It is directly installed on the back of the machine and additionally serves as a counterweight for the heavy vibrator equipment and mast.

(a) Vibrocats (*Courtesy of Keller*)



(b) Vibrocat Principle Sketch

Figure 2.34: **Vibrocat:** This particular machine is equipped with a vibrator, vibrator string, air chamber and lock, as well as with a skip for the filling material. Such configuration is used to perform the vibro replacement technique. The vibrocat is mainly hydraulically driven, electrical drives are used for the vibrator pull-down and skip winches. Additional power needed to drive the vibrator itself comes from a separate power unit, which is mounted at the rear of the machine instead of a dedicated counterweight.

For this thesis, data is collected from a vibrocat that installs elements based on the vibro replacement technique, as described in Figure 2.34. This particular version of the machine is controlled by a legacy PLC system, a B&R 2005, which in its used configuration does not offer a direct connection via OPC UA over Ethernet. Hence, the data used in chapter 3 has been provided by Keller's own data acquisition system, which offers time series in a CSV format. In total there are 17 channels of analogue data contained in these files. The newest development iteration of this rig is available since late 2019, it is controlled by a modern B&R X 20 – a PLC capable of providing data via OPC UA.

2.9.5 Challenges, Limitations and Discussion

The development of *System II* benefited to a reasonable extend from insights and experience gained during conception, operation, and data consumption within the framework of *System I*. The establishment of the new system for Keller is a global effort covering a wide range of different requirements and restrictions. Until it is available in a productive environment, data is gathered by using the currently existing data acquisition systems. Collected data is then provided to the analysis processes via temporary interfaces. Those allow the ingestion of data from different sources. Additionally, a temporary solution is used by a third-party company, EDS, to make data available in a continuous manner without interruptions between the individual elements that have been produced, as it is the case with an element-bound data acquisition. Exploratory analyses, as discussed in the upcoming chapter, are independent from the source of the data by using individual interfaces. However, some analyses cannot be performed without the presence af a continuous stream of time series, such as full time evaluations or relocation analyses. The JSON format specified for *System II* is also element-based in the first version, however, it comes with the ability of continuous data collection: files will then cover specific time ranges instead of elements and are concatenated later during ingestion into the data hub (in this case the cloud environment).

Of significant importance is the fact that the novel hardware concept is centred about the software running on the Windows-based device, the tablet. This paradigm allows the placement of centralised data acquisition intelligence on this device. In case of a hardware defect of this tablet, it can be exchanged temporarily with an arbitrary computer, which offers the same operating system as well as an Ethernet connection. No intelligence other than providing data values to the tablet is required on a PLC, in case it is needed to accommodate additional sensors for certain applications; the PLC only acts as a signal gateway. Maintenance and support efforts can be bundled and can mainly focus on the software part of the Windows-based device. In general, all components used for the local data acquisition unit are simple to be exchanged, since the interfaces are defined: the Windows-based tablet must offer Ethernet connection and has to accept values over OPC UA as a minimum requirement, the security device must be able to establish a secure and stable connection for communication with the cloud services and provide GPS location data in a dedicated format, and a PLC or gateway must provide OPC UA channels via Ethernet from sources such as additional sensors or field buses. A reasonably high level of flexibility is gained as a consequence.

Reliable and accurate reporting is a key requirement for production processes within the geotechnical engineering sector, since the products themselves can usually neither be observed during nor after production. Jet grouted elements can be theoretically excavated, however, this only happens for trials or when the elements are part of a construction pit wall, where their partly excavation is part of the system. Additionally, there are only indirect ways to measure production quality for vibro ground improvement installations: the quality is assured by a continuous collection of all signals relevant to the production of an element [48].

The presented data acquisition system exhibits a high degree of flexibility in terms of data acquisition units and their conception, interfaces to machines and other systems, availability of network connections, and access of subsequent processes to data in the cloud environment regardless of location on the globe.

2.10 Chapter Conclusions

Although both systems had different boundary conditions and requirements to fulfil, certain key aspects arose during the development of both: *security*, *flexibility*, *connectivity*, and *data access*.

Security is integral to systems within the IoT domain. A considerable amount of effort needs to be put into a multi-layer secure architecture, already starting at an early stage of the development process to avoid fundamental changes to the concept afterwards.

Apart from being secure, an infrastructure is required to be *flexible* enough to be installed on a variety of machine types in different operating environments. The security concept must foresee the adaptability to changing requirements. To keep such systems fit for the upcoming years of operation, it is beneficial to reduce a system's dependency on the components used, so that new devices can be used as replacements. Focusing the intelligence of a data infrastructure in software supports this paradigm significantly, since the support and further development is concerned only with the ability of the software platform rather than the limitations of the hardware.

An often neglected fact of the operation of such infrastructures is that a stable and always available *connection* to the data handling environment, be it a private data centre or a cloud, cannot be seen as granted. For *System I* it was observed that bad weather conditions and machine shutdowns delayed data transmission at several occasions. Additionally, there was no internet connection available for the machines of the IPCC mining system. The rigs used in geotechnical engineering also encounter issues with a stable internet connection, especially when working inside buildings for smaller rigs, in deep excavation pits, or in locations with low to no stable (mobile) internet reception. The possibility of ingesting files manually into the data handling environment is required.

Building up a façade for the users between data input and output of a system is fundamental to letting the users interact with data provisioning without having to know anything about the data ingestion process. The domain experts can utilise a continuous data model to query from and to base their procedures on. Hence, developments at the provisioning end of the infrastructure system are able to be performed in a more generalised manner. A simple and robust *access* to data is vital for any system to add value in the long run. The more users can collaborate by using such a system as a platform, the more domain expertise can be brought in.

Chapter 3

Conventional Approach for Machine Data Evaluation

Once data is available for further analyses, it is necessary to establish a robust and simple way of working for domain experts to derive insights from the time series. The procedure described in this chapter focuses on an example for data of vibro ground improvement, however, it has been built out in a generic manner to simplify an adoption for other techniques and domains. The exploratory analyses are performed with Mathworks® MATLAB. The analysis of the vibro data is organised in four phases: preprocessing, point[1]-wise evaluations, aggregated site-wise assessments, and spatial representations of the evaluated time series. A defined and clear folder structure has been found to be a key aspect for analysing a considerable amount of data. This is especially true for team efforts, where several analysts work on the same data repository. Additionally, a paradigm where the code is brought to the data is of support when conquering big sets of data. A structure used for the analysis of vibro data is illustrated in the directory tree in Figure 3.1. The root folder `kellerVibro` contains three subfolders:

- `genericMetadata` consists of MATLAB code defining all necessary paths for the files and methods;
- `mCode` provides methods in form of MATLAB code; the methods cover preprocessing tasks (`csvProcess`), key performance indicators (KPI) evaluations (`pointKPI` as well as `overlap`), and the procedure of plotting all available time series data of a point;
- `sites` holds time series in raw format and evaluation results per site; the site-related metadata is contained in `siteMetadata`, while the evaluation results are stored in `pointData` and in `overlap`.

The time series used for these analyses are contained in point-wise CSV files. Since the preprocessing step is designed as a generic stage, it can also handle other files formats,

[1] *Point* in this context refers to an *element*, which is produced by a rig. For the presented analysis, files containing time series for a single element – a point – are used, which cover a pre-production, a penetration, a compaction, and a post-production phase.

such as the JSON format described in the previous chapter – it is just necessary to extend the read mechanism to handle the additional file format(s).

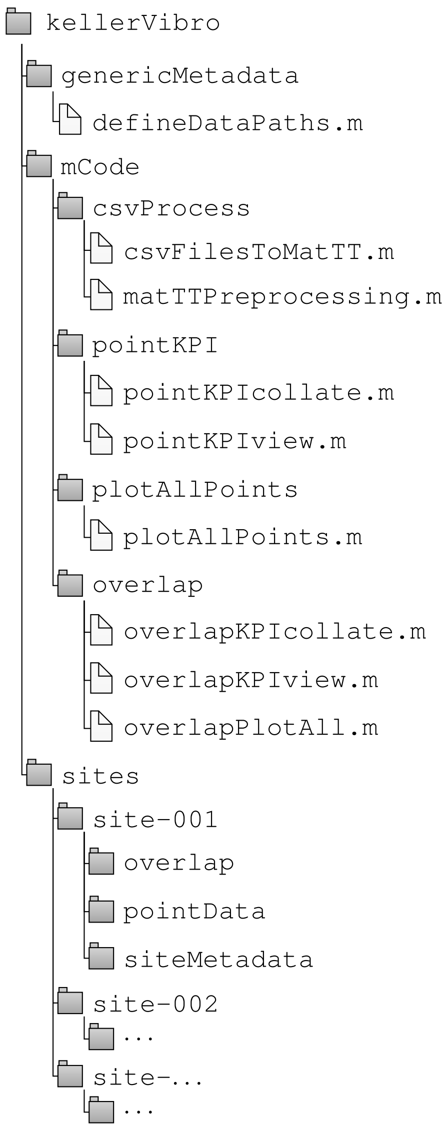


Figure 3.1: **Directory Tree for Vibro Analysis:** The MATLAB code mandatory to analyse the data is organised in three subfolders: `genericMetadata` for metadata related to the analysis project and the associated paths, `mCode` for all MATLAB methods and functions, and `sites` for point data (input), site-related metadata and evaluation results.

The following sections describe the steps necessary for the evaluations. The data flow figures illustrate the transition from input to output data: The path to the input data is printed on the left bottom, while the output path can be found on the right bottom side of each figure. The MATLAB code used to generate the output data from the input is given in the top middle part of each flow diagram. The leading dots of each path denote the root path as listed in Figure 3.1.

3.1 Preprocessing – Read From Source

Prior to any analysis, the time series need to be available in an appropriate data structure. The initial processing step is to read data from a source file. This is considered an individual step to ensure that further analysis steps are independent from the input format. If another input format needs to be read in, the parser can be extended to handle other formats accordingly. The general data flow is showcased in Figure 3.2. The source files of this preprocessing step contain time series and metadata in tabular CSV format. After the content of a file has been parsed, the metadata is stored in a structure, while the time series are kept in a timetable format: each row corresponds to a specific timestamp, each column represents an individual signal channel. When the method `csvFilesToMatTT` is executed, the respective site (its storage location according to the directory tree previously discussed) needs to be specified; all point files of the site are processed automatically.

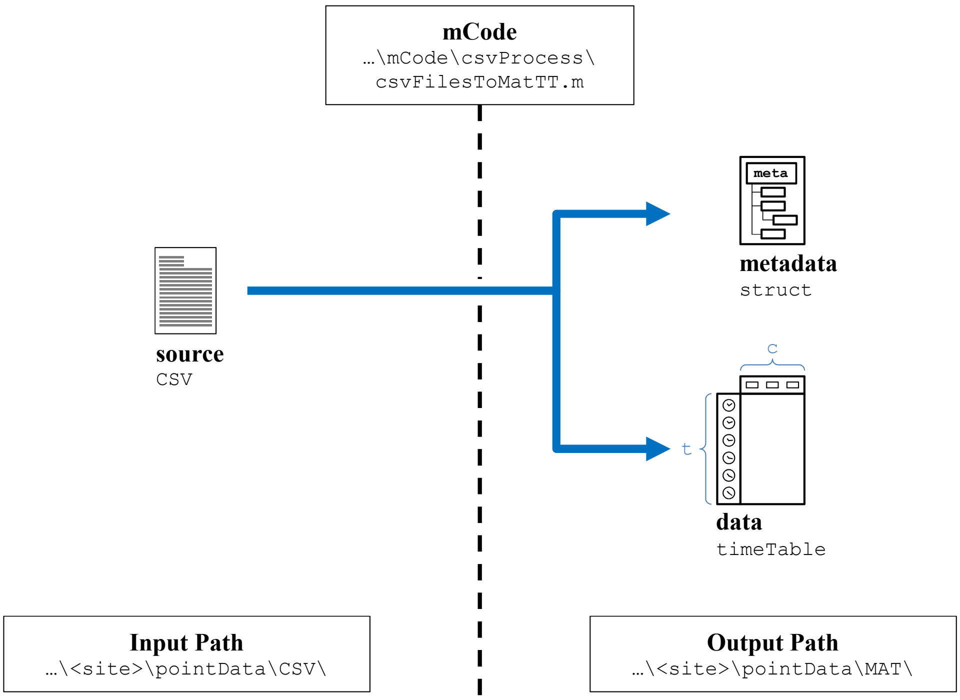


Figure 3.2: **Preprocessing:** A preprocessing step is mandatory to organise data from tabular CSV files in a structured manner. After the file contents are read in, **metadata** is stored in an appropriate structure and the time series are saved to a timetable format, **data**. All files of a site are processed automatically.

3.2 Level 1 Processing – Per Point

Once the data is made available in processable formats with the initial, previous step, the time series are used to produce summaries and derived calculations. As showcased in Figure 3.3, the data in form of a timetable is used to create a variety of intermediate results. A channel-wise statistical summary, called **dataSummary**, is generated. Each time series per point can be segmented into several individual phases, such as the penetration and the compaction phase, as well as the time ranges prior the penetration and after the compaction phases. The indices of the start and end timestamps of all these phases are saved in a **phases** table.

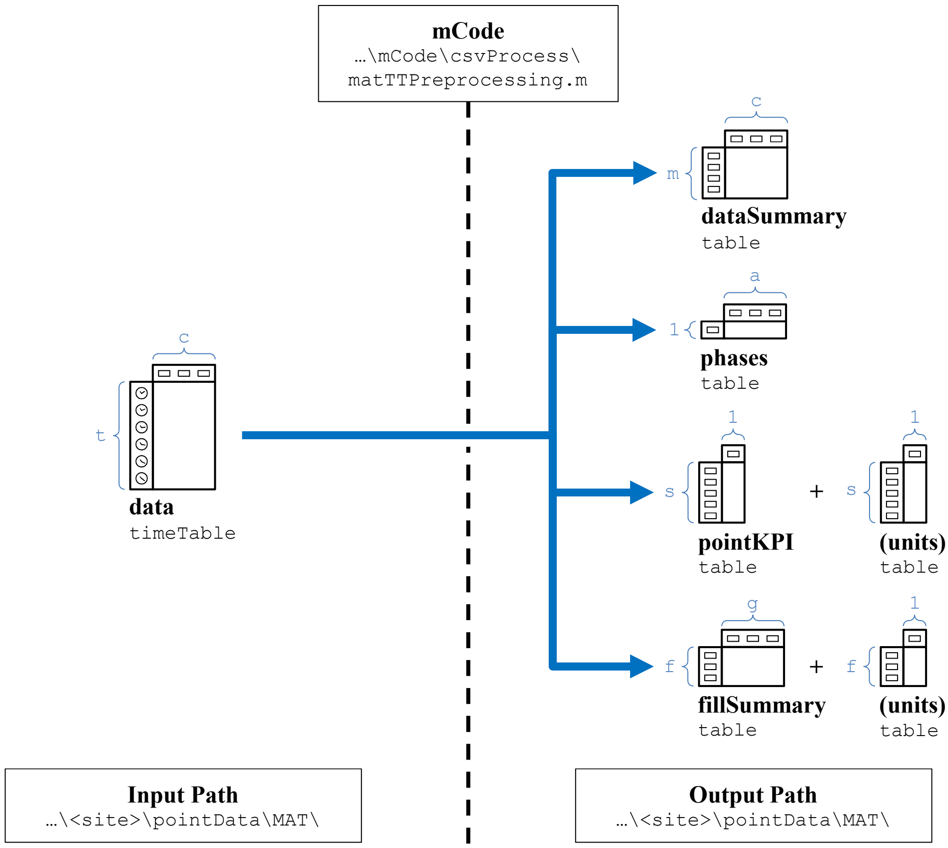


Figure 3.3: **Summaries and KPI Calculations:** A summary table containing statistical measures is produced from the preprocessed timetable **data**, called **dataSummary**. The indices of the timestamps indicating the individual production phases of a point are saved in a separate table, **phases**. A list of KPI is calculated per point and is saved in a **pointKPI** table. Calculations regarding the skip fills during the compaction phase are quality measures of the compaction phase; they are saved in a **fillSummary** table.

Key performance indicators for the point are calculated and stored in a table called **pointKPI**; the corresponding units are additionally kept as metadata. One of the specifics of the vibro technique is the activity of several skip fills during the compaction phase. To summarise fill-related calculations, the corresponding data is saved in a **fillSummary** table.

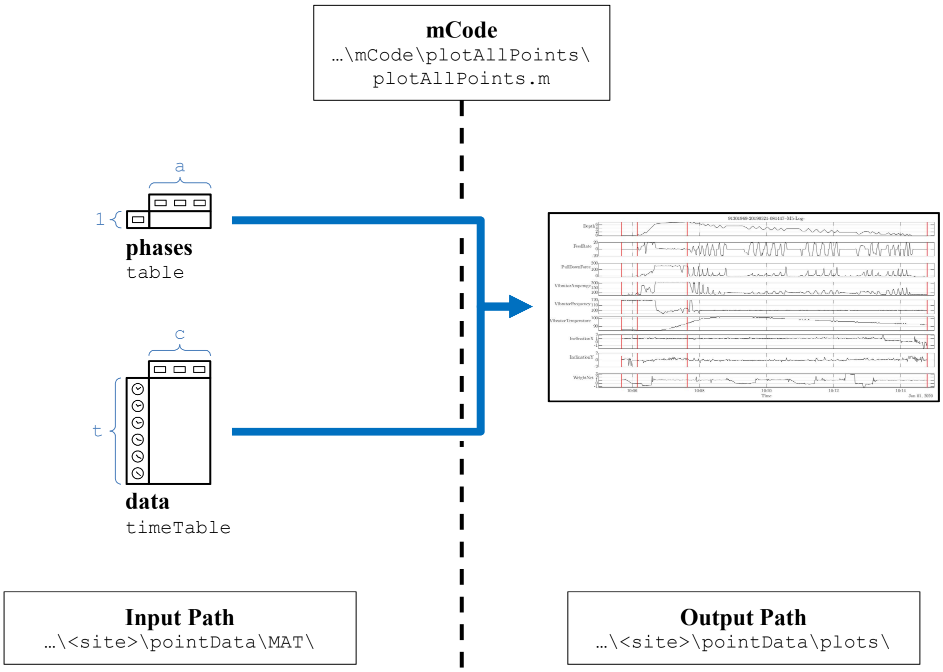


Figure 3.4: **Plot All Points with Phase Markers:** The **data** timetable is used together with the **phases** table to produce a plot per point with highlighted phases. The channels used for the plots can be selected.

The information in the **phases** table is used, together with the timetable **data**, to generate plots of the point data with highlighted phase start and end markers. This `plotAllPoints` procedure processes all files of a site and outputs one plot per point, as shown in Figure 3.4. The channels that are listed on each plot can be selected after the procedure has been started. An example plot view is shown in Figure 3.5. The phase transitions are marked in red. These plots are of support when a quick view of an element, a point, is required.

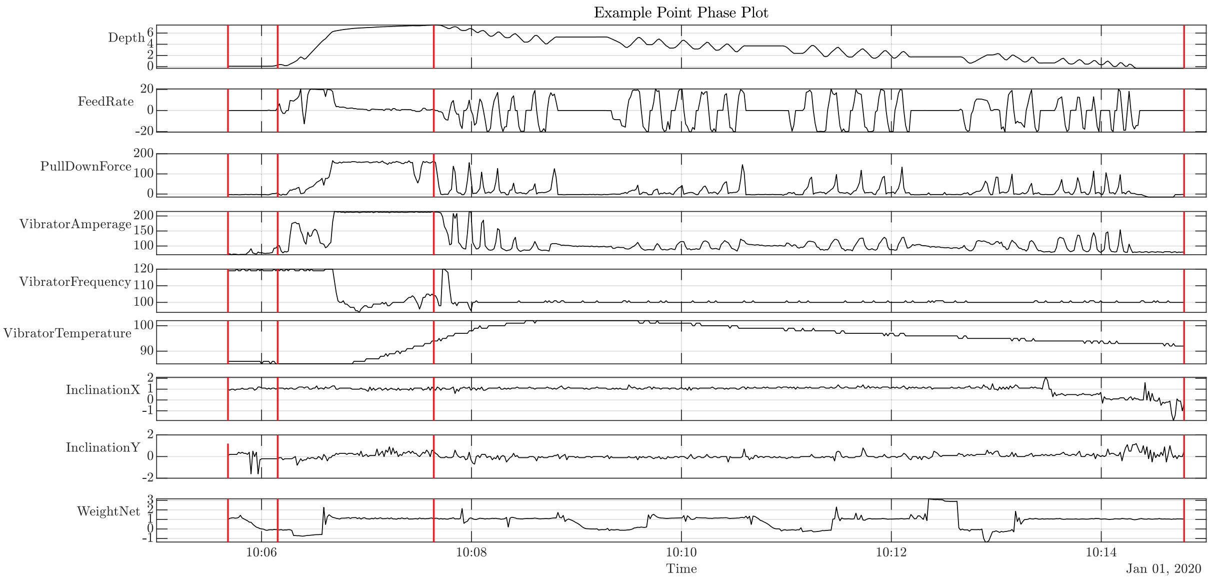


Figure 3.5: **Example Plot of Single Point with Markers for Phases:** The selected channels are plotted against time, the time range limits of the different phases are highlighted with red lines.

The **data** timetable and **phases** table are additionally used for generating KPI necessary for the overlap evaluation. An *overlap* in terms of vibro ground improvement is a quality measure during the compaction phase: the vibrator is lifted up for the gravel to fill the void underneath it, before it penetrates the ground again to compact the fill material in the hole – the ratio between those two single steps, the difference of the relative depths, is considered to be the *overlap*. The procedure in Figure 3.6 showcases the derived results from the input data: An **overlapTable** timetable contains information about every single overlap (duration and depth difference) and associated timestamps. The **compacting** table contains indices of start and end timestamps of a compaction step, as well as its length. Derived KPI for all overlaps are stored in a separate **overlapKPI** table.

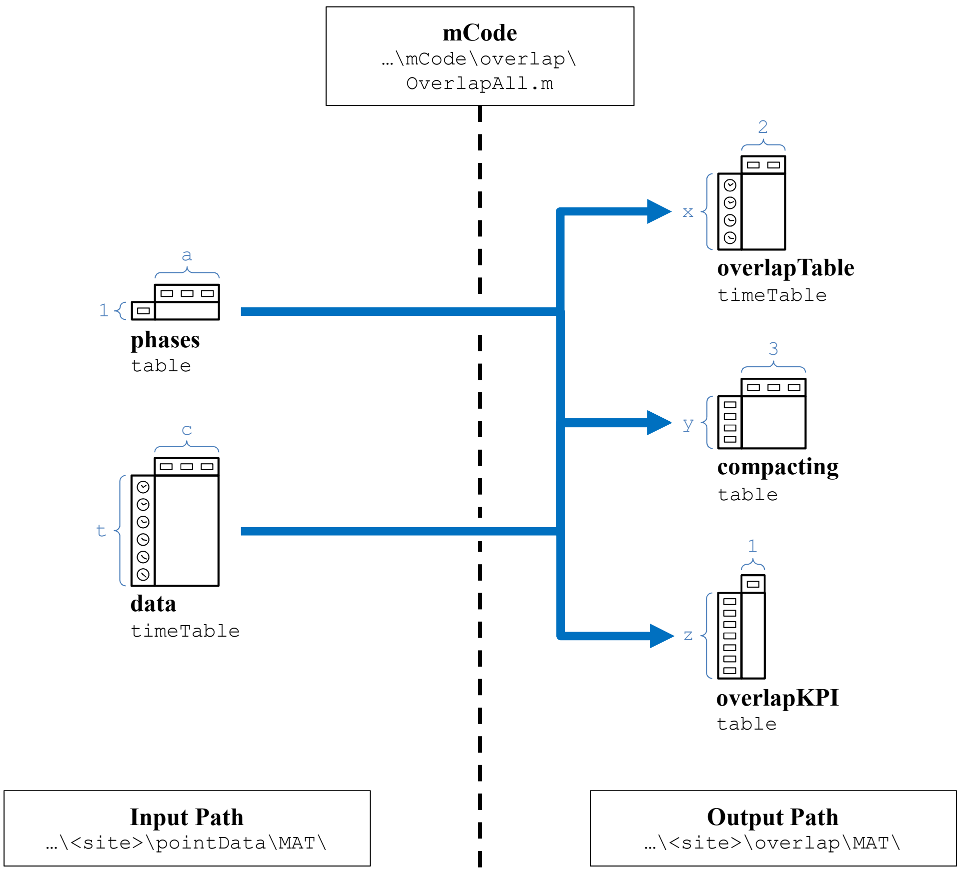


Figure 3.6: **Calculation of "Overlap" KPI per Point:** The timetable **data** is used together with the **phases** table to produce an **overlapTable** timetable, which holds information about the individual overlaps, a **compacting** table containing start, end and duration information of each compaction step, as well as an **overlapKPI** table, holding relevant key performance indicators.

3.3 Level 2 Processing – Per Site

In addition to evaluations based on single points, data of several points per site can be aggregated to gain supplementary insights. The preceding steps facilitate the conglomeration of individual point-related values. The aggregated data is of particular interest for the process of identifying points with abnormal characteristics per conglomeration, per site. The underlying data flow is showcased in Figure 3.7. After the execution of the method, a site is selected for the evaluation – the corresponding folder contains the individual **pointKPI** table data. The procedure collates all the KPI values of the input tables to a new **pointsKPI** table. It holds all KPI values of all points, each column corresponds to the values of one point.

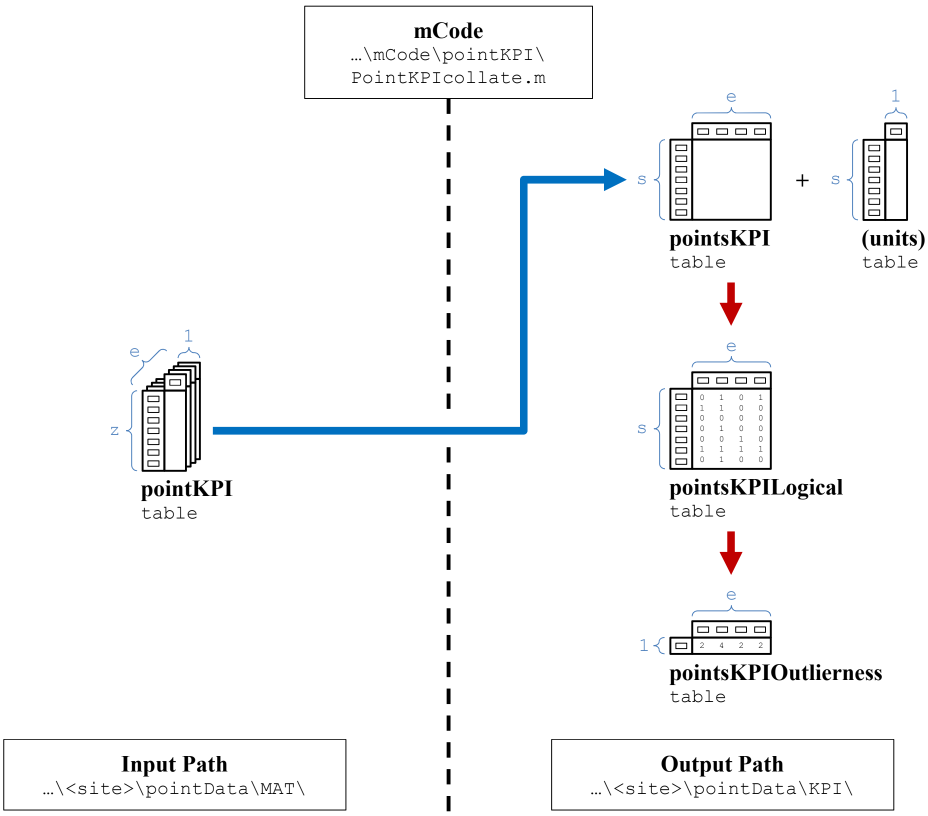


Figure 3.7: **Collation of Individual Point KPI:** The individual **pointKPI** tables of a site are collated to a single **pointsKPI** table. Definitions for determining outliers in the set of KPI are used to produce a **pointsKPILogical** table, which holds logical information in the cells instead of the KPI values. To provide an overview of the "outlierness" of a point, a summarising **pointsKPIOutlierness** table is generated.

Validity range definitions and conditions per KPI are used to identify values that are considered outliers. For more efficient subsequent processing, a table of the same dimensions of **pointsKPI** is created, called **pointsKPILogical**. Its cells hold logical values for each KPI value, indicating whether an outlier has been detected or not. An additional summary table, **pointsKPIOutlierness**, is generated, which provides the total count per column to identify points with a higher number of outliers.

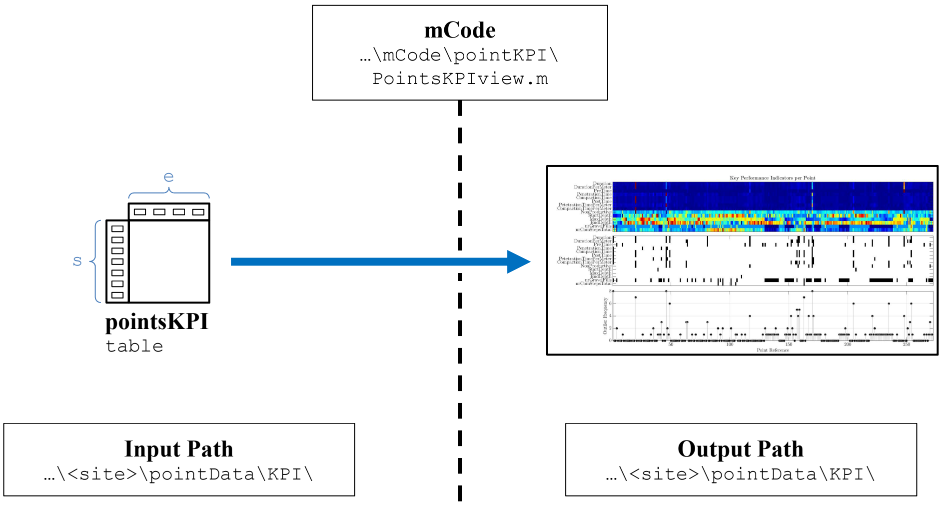


Figure 3.8: **View of Collated KPI per Set:** The previously collated KPI information is plotted to provide a visual representation for further processing by domain experts.

In a next step, the collated data sets are used to produce a visual representation of the previously calculated KPI information – see Figure 3.8.

A close-up of the produced overview plot can be seen in Figure 3.9. The plot is split into three main sections; the horizontal dimension corresponds to a chronological order of individual points. The topmost subplot illustrates the key performance indicators of the table **plotsKPI**; each row stands for a specific KPI, while each column stands for an individual point. The values are normalised over their value range of all points, the colours indicate the measure of how much the individual values are off standard. The middle section represents the **pointsKPILogical** table, showcasing KPI outliers of a point. The sum of all outliers per point are visualised in the bottom subplot: The higher the number per point, the higher the level of "outlierness". The existing reference to a point of interest allows the investigation of a detailed point plot as exemplarily shown in Figure 3.5.

In terms of overlap computations, corresponding key performance indicators are being processed similarly to the general KPI. As illustrated in Figure 3.10, the individual per point **overlapKPI** tables are read and collated by the `overlapKPIcollate` method. The main result of the collation is a merged **overlapsKPI** table. The horizontal dimension of the new table corresponds to the number of files and, hence, to the number of points. The units belonging to the table are kept in a separate table. Derived logical information is saved in a separate table, **overlapsKPILogical**: measures for what qualifies a value to represent an outlier are implemented to produce such a table, containing only logical information. As a last step, a summary table **overlapKPIOutlierness** is produced and saved to provide an overview of the number of outliers per point, according to overlap KPI.

To visualise the previously generated tabular data, a view of the KPI data is produced by the data flow illustrated in Figure 3.11. This enables domain experts to identify points of interest for further investigation or processing.

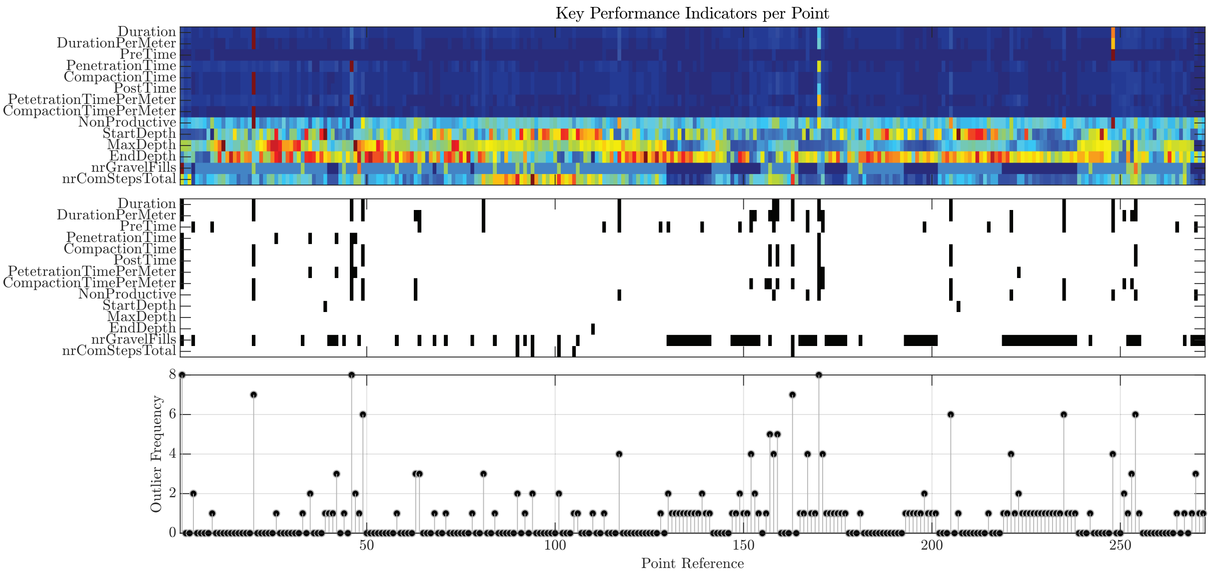


Figure 3.9: **Example View of Collated KPI:** The plot is divided into three main sections. The horizontal dimension corresponds to the individual points, one column per point. The topmost subplot showcases a row-wise list of KPI, the colours correspond to how far a value is off a normalised (range) standard. The middle subplot shows the KPI outliers per point, the bottom plot summarises all outliers per point.

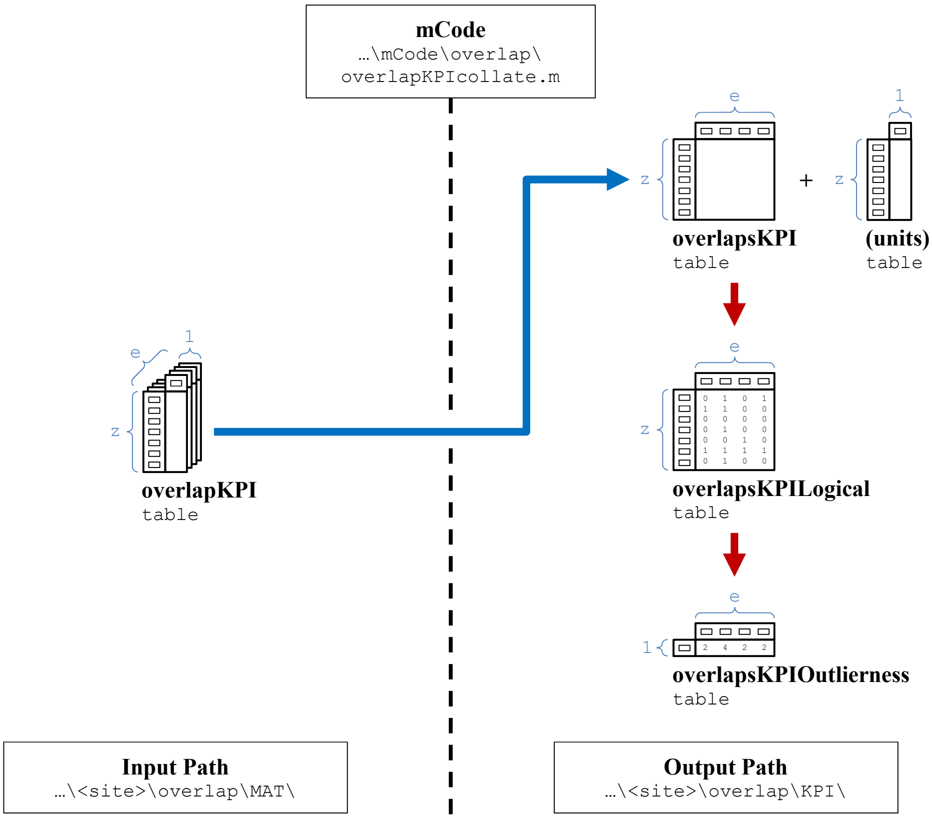


Figure 3.10: **Collation of "Overlap" KPI of Individual Points:** The values of all **overlapKPI** of a specific site are collated to form a single **overlapsKPI** table, containing all point-wise overlap KPI of a site. A derived **overlapsKPILogical** table contains logical values indicating, whether a particular KPI is considered an outlier. A summary of all outliers per point is given in another separate table, **overlapsKPIOutlierness**.

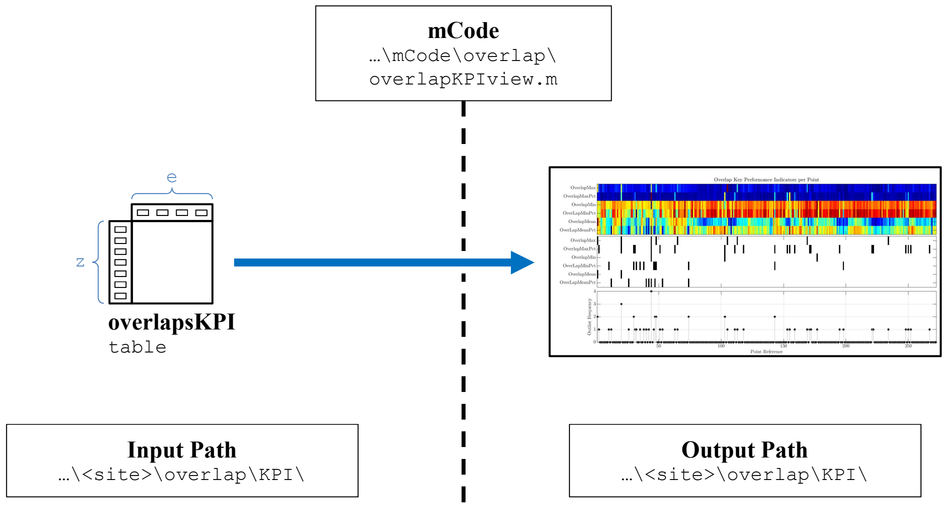


Figure 3.11: **View of Collated "Overlap" KPI per Set:** Tabular data of the previous step is plotted to provide a visualisation for subject matter experts.

A detailed view of the generated plot of the overlap KPI is given in Figure 3.12. The number of overlap KPI is rather small in comparison to the list of general KPI of the previous steps in level 2 processing. The reason for this is the specific focus on the particular issue of overlap characterisation per point. The mechanisms are the same as previously, the plot is divided into three main sections. The subplot at the top illustrates the KPI per point (**overlapsKPI** table), the values are again normalised over the full value range of the respective KPI of all points. The middle portion represents the logical table **overlapsKPILogical** of outliers, whereas the bottom subplot summarises the existing outliers per point.

If a point of interest has been identified during the evaluation by subject matter experts, the reference to the individual point allows a more detailed investigation based on its visual representation in the form of a plot. Such an element view is plotted in Figure 3.13. The channels relevant to the respective evaluation are selectable and show up in the plot as needed. Since the overlap evaluation focuses in particular on the compaction phase of the production process, only this phase is visualised. The parts of the signal plots highlighted in red showcase the temporal dimension of each partial penetration step – the part of an overlap.

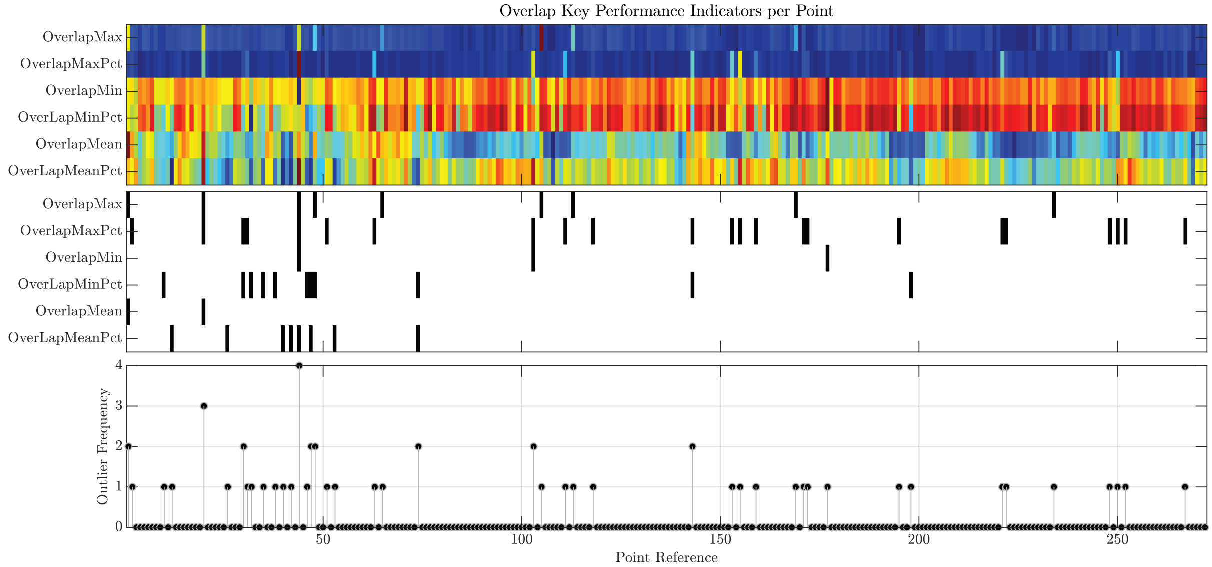


Figure 3.12: **Example View of Collated "Overlap" KPI:** The representation of the collated KPI per point is organised in three parts. The top section provides an overview of the overlap KPI values per point, whereas the rows correspond to the list of overlap KPI and the horizontal axis represents the chronological order of points. The subplot in the middle of the figure gives an overview of which KPI values are characterised as an outlier. The bottom part showcases the point-wise sum of outliers.

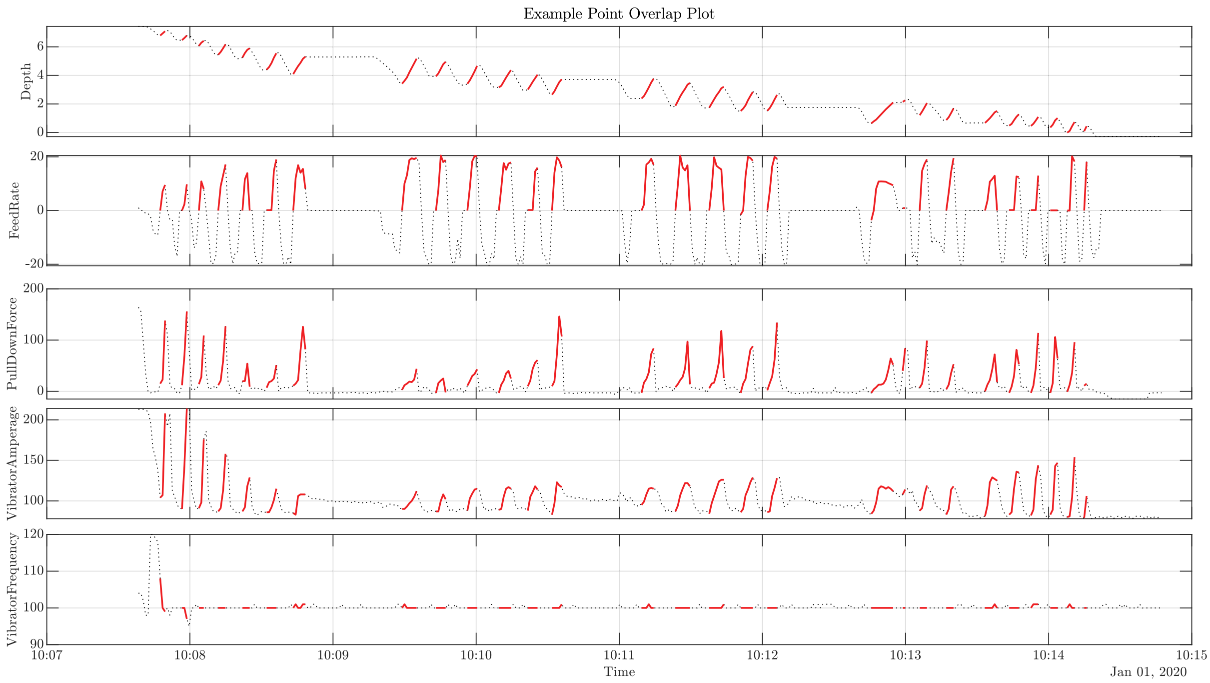


Figure 3.13: **Single Point Plot of "Overlaps":** The channels of choice are plotted against the compaction time of the production process. The highlighted portions indicate, in a cross-channel manner, the durations of the individual partial penetration portions of the overlap steps.

3.4 Spatial Referencing

The previous means of collating intermediate results of multiple elements are based on a chronological order of the points: the leftmost point is also the first one produced on the particular construction site, the rightmost corresponds to the last one. The installation order of the points is designed to follow the constraints of the principal construction sequence or local restrictions of the site; a point produced right after another does not necessarily need to be located next to the preceding point. To visualise the spatial distribution of the installed points in addition to the sole chronological overview, the points are plotted with coordinate metadata. Such a plot can be seen in Figure 3.14. Design coordinates relative to a local grid are used for spatial referencing of the points. Such a viewing tool does not only enable printing the locations of the individual points, it can be further used to attribute KPI information to the respective elements. The illustrated two-dimensional view makes it possible to associate two values to a point; one is presented as the colour of the point, the other uses the scale of the diameter to convey information. In the given example, the colour refers to the normalised maximum depth of the point, while the diameter or radius gives an indication of the non-productive time per element. Such a plot provides a spatial overview of KPI that might indicate locally clustered areas. This is of particular interest for identifying areas of different soil layers or changing environmental conditions. The additional information of the spatial distribution of key value indicators can be of significant support in the evaluation of a site or of the response behaviour of a system or machine to specific environmental and soil layer conditions.

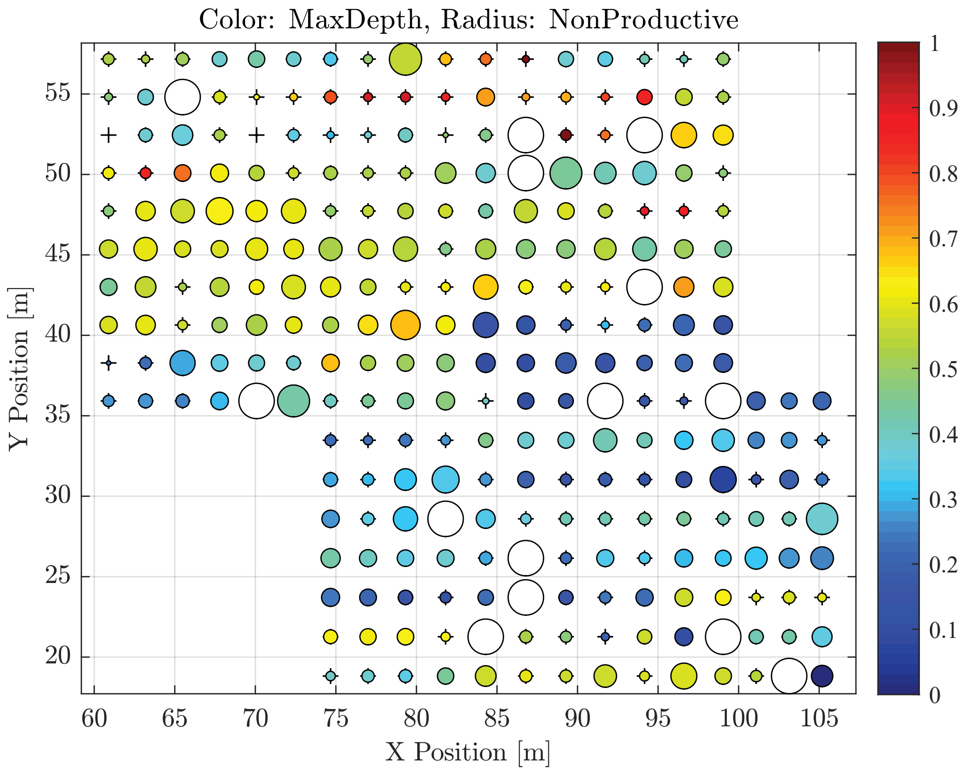


Figure 3.14: **Example Plot Illustrating Spatial Distribution of Produced Points,** which is capable of conveying three different means of information: the location of a specific point element; a first KPI in form of the plotted element's colour; as well as a second KPI, visualised as the radius of the element. In this particular example, an element's colour refers to the normalised maximum depth, while the radius provides information about the non-productive time during installation of the particular point.

3.5 Chapter Conclusions

The presented set of processes and tools supports the interface between methods of data analytics and domain experts, who are often not proficient enough to handle time series in an efficient manner. The mechanisms let the experts interact with the data to discover insights and to identify areas of interest, which require further investigation.

During the initial exploration sessions, it has been found that it is of value to divide the key performance indicators into subgroups, e.g., into KPI relevant to production quality or to the production process itself. Additionally, those value groups can be divided into individual sets for the penetration and the compaction phase.

Ideas for additional KPI have been found as well; once their derivation is known, the implementation can be performed with a reasonable amount of effort. The generation of a new result set can be established in a simple manner by following the steps described in this chapter, since the time series in raw format are still available to the analysis system, even if a different set of KPI has been produced before.

In addition to ordering the point data chronologically as shown in the KPI collation examples, they can also be ordered based on depth or other measures – combining coordinate metadata with the calculated KPI provides a whole new layer of information. The spatial distribution helps identifying, whether an outlier in a site context is still an outlier based on the surrounding elements – there may be close elements that are identified as outliers as well, however, as a cluster they would identify an evident change in soil conditions for instance. In future approaches, it can be of additional support to compare the design coordinate metadata with as-built spatial information – also to find areas with higher deviation measures than others.

The methods discussed in this chapter do not only provide the identification of outliers, they can also be used to filter the data of a site or a certain aggregation – temporal or spatial – based on an initial recognised set of outliers or non-standard behaviour: in a subsequent step, the outliers can be classified into those which need further attention and those which do not. By all means, the framework provides the possibility for the domain experts to explore time series of notable size in an efficient manner and without the permanent requirement of the support of highly specialised data scientists.

Chapter 4

The Metaphorical Concept of Language

4.1 Introduction

As the procedures in the previous chapter show, the identification of elements or patterns of interest is dependent on statistical properties. The methodology of analysing data in this manner does not take the special nature of the data into account: the machines are fundamentally cyber physical systems (CPS), which follow physical principles. Additionally, the analysts add meaning to outliers identified in the data, not the analysis process itself. This requires a high level of user interaction throughout the data evaluation process. In the following chapter, the classical approaches described in the previous part are contrasted with an attempt to analyse time series emanating from cyber physical systems in a novel manner: by mimicking the mechanisms of natural language to support an unsupervised detection of hierarchical structure and pattern significance.

This part of the work follows an unconventional approach: partial theoretical background is given first, before an example is presented, based on the initial context. After the exemplary results have been discussed, a more comprehensive study on the phenomenological background is provided. Such a presentation strategy allows a forbearance[1]-driven discussion of a number of related fields influential to novel data-science approaches, such as the matter of the emergence of natural language. This methodology is meant to phase out this topic in a manner that further discussions are initiated and future work can tie in simply.

A starting point could be the discussion of what knowledge is and how it can be derived from time series. This serves as a precursor to the metaphorical concept of natural language. Then symbolic analysis is visited, since the serialisation of time series is prerequisite to introducing elements of language. Subsequently, an example is given on data of

[1] In this context, *forbearance* is used in the meaning of the associated Proto-Indo-European root **bher-**, which means "to carry, to bear a children, endure" [50]. This association allows to extend the contemporary meaning of the word by a nuance of having to go a long distance to arrive at a certain goal, by undertaking all steps in the continually present intention of working towards the goal.

a mobile machine used in the mining industry. The emergence of language and associated issues are presented in a metaphorical context to put the results into a broader, theoretical framework. The last part of this chapter focuses particularly on the phenomenological aspects of the emergence of language.

Previous work also covers the application of learning methods on multivariate time series of machines and plant used in the mining industry [51], similar to the machines described in chapter 2. However, the work discussed in the following section focuses on analytical, conceptional methods rather than solely on perceptional models, such as representation learning (feature learning, where the features can be identified using deep learning approaches). This does not imply that such methods are not reasonable to use; however, they are not of primary concern for what is to be shown with the methods presented here. The analytical exploration of language-affine properties in the presented time series is elaborated and put into a theoretical context, which is primarily based on analytical approaches rather than solely on perceptional methods. However, the latter can still be used to close gaps where analytical methods reach their (current) limits. Still, it needs to be kept in mind that perception and conception are not equivalent. A fascinating example is provided by the attempt to train pigeons to distinguish benign from malignant samples of cancerous human tissue [52]. The birds have been trained with a special apparatus that lets them choose between two differently coloured buttons: one indicating a right choice and releasing a food pellet through a chamber and another one that did not do anything, thus indicating a wrong choice. The birds' perceptive capacity was sufficiently good to reach impressive accuracy values when shown novel images not contained in the training sets. However, interestingly, they showed clear deficiencies in detecting the right choice for another particular task of image detection. This behaviour can be seen as acting similarly as perceptive learning methods: they are well-suited for particular tasks and can be configured to deliver impressive performance indicators. However, attempts to generalise such methods or trying to make analytical approaches obsolete will involve multiple issues that need to be considered. In general, the premise behind representation learning is that if a signal can be recreated from a representation, then a viable model for the data is given. However, representations are non-unique, there are multiple representations which encapsulate the features of the data. This begs the question: what makes for a better representation?

A hybrid approach can bear significant potential to utilise the advantages of both concepts. In this chapter, the emergence of language is presented as a contribution to artificial intelligence, which is not (solely) based on learning methods, such as representation learning.

4.2 From Belief to Knowledge

The analysis in the scope of this thesis is performed on time series of mobile machines of the mining and the geotechnical engineering industry. Such machines or plant can be seen as cyber physical systems. Such CPS can be defined by the following statement [9, 53, 54]:

> *Cyber physical systems are systems with a tight coupling of the cyber aspects of computing and communications with the physical aspects of dynamics and engineering* ***that must abide by the laws of physics****.*

Hence, as per definition, cyber physical systems must strictly follow physical laws. The nature of sensor data is needed to be taken into account when data analysis is performed on time series emanating from physical machines. The physics of a system can be modelled as mathematical equations, e.g., to consider system dynamics. The forward problem would be given by providing parameters to the physical model (the mathematical equations), yielding a result. An inverse problem instead, generally defined as $\hat{y} = f^{-1}(x)$, is used to derive causal factors from the *observed effect* – see Figure 4.1 for a reference. The observed effect is represented by sensor data, the *model* is given by the mathematical formulation of the physical problem. To derive a *cause* from the observation, *a priori knowledge* (metadata) is required, since, in general, the solutions to inverse problems are non-unique. [9, 55, 56]

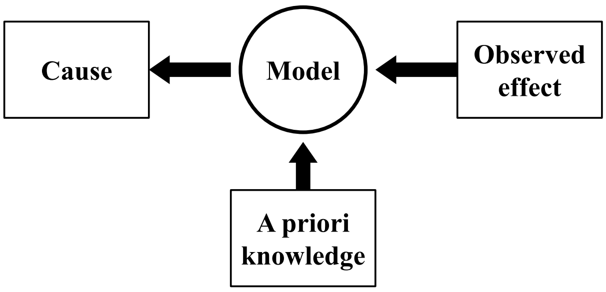


Figure 4.1: **Scheme of an Inverse Problem:** A mathematical *model* is used to describe a physical law. Sensor data corresponds to the *observed effect*. The *cause* is able to be determined by solving the inverse problem. In general, solutions to inverse problems are non-unique. *A priori knowledge* is required to yield desired results. [9]

The consideration of inverse problems is rarely addressed in literature concerned with the mining of sensor data [57–60]. It seems that some approaches to mining sensor data take correlation as a reliable measure for significance; hence, neglecting the fact that this kind of data follows physical models. Inverse problems are an instrument to establish semantic links between sensor observations and the respective causes. Missing this semantic relationship means that no knowledge discovery can be performed on the basis of physical models. [9, 28]

Models for a structured approach to data mining exist; one example is given by the ***Cr****oss-****I****ndustry* ***S****tandard* ***P****rocess for* ***D****ata* ***M****ining*, abbreviated as *CRISP-DM*. It can be used as a guideline for investigating and documenting processes in the data mining domain. Although the approaches in CRISP-DM are of a generalised form, which does

not specifically consider the nature of data from physical systems, it can be of support in business-driven data mining processes. [61]
In addition to CRISP-DM, IBM published the ***A**nalytics **S**olutions **U**nified **M**ethod*, *ASUM* in short: it is based on the concepts of CRISP-DM, which it modifies and extends. ASUM follows a hybrid approach of agile and traditional implementation principles to adopt contemporary requirements of agility and flexibility in data mining developments. [62]

When looking at the particular issue of data mining in the context of cyber physical systems operated by human beings, we are interested in how to derive insights from the sensors' raw data. Prior to visiting how knowledge can be discovered, a number of terms need to be defined for the context of this work. During the analysis of time series, terms such as *data*, *information*, *knowledge*, or *understanding* are used – as it is the case with the work presented so far. An attempt to combine those terms, and to give them a hierarchical structure, is provided by the *wisdom pyramid*: this concept has been developed by Embrechts et al. [63] after the fundamental work of Ackoff [64].

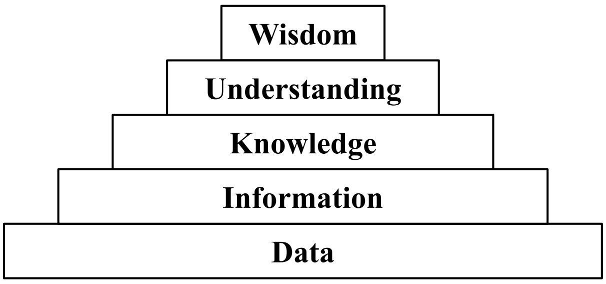


Figure 4.2: **Pyramid of Wisdom for Data Mining:** This illustration is an attempt to put the terms used in data mining into a hierarchical structure to further put emphasis on the relationship and dependencies between the individual components. [9]

The three bottommost terms, *data*, *information*, and *knowledge*, are solely based on past actions and, hence, are concerned with what already has happened. In contrast, the topmost term, *wisdom*, tends to point towards future occurrences, since it facilitates all underlying levels for a situative and contextual judgement of an experience arising in the present (or in the future in relation to the past experiences). *Understanding*, however, can be seen as residing in between both positions – it exhibits a *conceptional* meaning, which is mainly based on past experiences, but is still mandatory for subsequent steps for present (future) references for reliable prediction of behaviour.

Each step of the pyramid is built upon another, indicating dependencies. In the context of mining data emanating from cyber physical systems, the terms could potentially be defined in the following manner: *Data* is acquired from cyber physical systems. Machines or whole plants can represent such systems. A hypothesis must be given for collecting the signals of relevance in a certain quality for the time range(s) of interest. *Information*[2]

[2] *Information theory* as it is known today was originally described by Claude Shannon in his paper *A Mathematical Theory of Communication* [65] (information content, entropy); previous work by Harry Nyquist already provided initial ideas [66] (Nyquist criterion in sampling a signal). Ronald Fisher contributed to the concept of maximum-likelihood estimation, enabling the calculation of covariance matrices [67].

can be gained by adding metadata to the previously acquired data. Causal links and inverse models makes *knowledge* derivable from information. The next step to get to *understanding* involves advanced methods, such as symbolic analysis – as will be presented later in this chapter. In the underlying context it can be proposed that the term has the ability of formulating models of explanatory, or – in some cases – also of predictive potential. The term *wisdom* cannot be defined so clearly: as stated in previous work, a potential attempt for a definition can be given by seeing associated actions based on wisdom as contextual and situative. However, the component might be better named with terms such as "good, sufficient, or educated decision-making".

What the pyramid notably illustrates is that all our analyses and further work fundamentally depends on the data used. This justifies the importance of establishing reliable, robust and secure data acquisition frameworks to assure the input data for our evaluations and methods abides by the quality measures of legitimate and consistent research. All our analyses are based on the data we use as an input; however, can we trust its source? Do we believe its trustworthiness or that of its source? Is this belief justified? If yes, what justifies it?

The exemplary correlation between American civil engineering doctorates and Mozzarella cheese consumption in the introduction showcased a high level of correlation – 95.6 %. Other examples for such spurious correlations are [10]:

- Computer science doctorates versus comic book sales – 99.5 %;
- Money spent on pets versus alcohol sold in liquor stores – 99.4 %;
- LEGO® revenue versus worldwide revenue from commercial space launches – 99.4 %.

Although the degree of correlation is considerably high in each case, human intuition negates the validity of these examples in terms of causal relationships. Their nature and context do not justify correlation to be a valid measure for causality. But how are relationships examined, which do not exhibit such clear indicators? The examples refer to another issue: do we believe the sources the cited book is using? Although the sources of the underlying data are given, they are often generalised pointing to an entire data repository of an organisation and are not simple to find – many of the used data sets are older than ten years. Still, such sources are usually not followed up by readers, the belief is justified for most. It calls up the question, how we can justify beliefs in a general manner. How can we derive knowledge or even understanding from something that is based on justified belief? What is justified belief? Does it equal knowledge?

In general, the philosophical underpinnings of research can be formulated as a paradigm consisting of questions of *ontological*, *epistemological*, and *methodological* nature [68]. The outcome can be formulated as *methods*. The questions each field is concerned with can be posed as follows:

Ontology:	What is out there to know? What can be known?
Epistemology:	How do we know what we know, how to gain knowledge? What are the limitations of the knowledge gained in a specific manner?
Methodology:	How can knowledge be acquired?
Methods:	Which procedures can be used to acquire knowledge?

Ontology, in a generalised manner, is the philosophy concerned with the discourse about *what there is to know* and with the fundamental nature of reality [69]. The term is a combination of the Greek words *ontos* and *logy*, with *ontos*[3] meaning "being, existence" when translated from Greek and with the Proto-Indo-European (PIE) root **es-**, meaning "to be" [50]. The second term, *logy*[4], translates to "discourse, theory" and has the PIE root **leg-**, meaning "to collect, to speak" [50]. The field ontology puts emphasis on *what there is*, independent from how knowledge can be derived from entities' existence.

Epistemology is concerned with the theory of knowledge, *how we know what we know* as well as with the justification and rationality of belief. It is derived from the Greek words *episteme*, which translates to "knowledge, understanding, acquaintance", and *logos*, meaning "account, argument, reason, study of" [70, 71]. The two Proto-Indo-European roots contained in the first part of the term are **epi-**, meaning "near, at, against", as well as **-sta**, meaning "to stand" or "place or thing that is standing" [50].

In the traditional, Platonist analysis of knowledge, there are three components necessary and sufficient for gaining knowledge, also called *tripartite* or JTB analysis (Justified True Belief) [72–74]:

***S* knows that *p* if and only if**
i. p is **true**;
ii. S **believes** that p is true;
iii. S is **justified** in believing that p is true.

The tripartite involves conditions for *truth* (i), *belief* (ii), and *justification* (iii). If all are met, the claim is a JTB, which is furthermore considered to be equivalent to knowledge. In Plato's dialogue *Theaetetus*, such a justification condition is postulated, since the formerly prevalent *true opinion* is in general insufficient for knowledge acquisition [72]. Unfortunately, it has been shown that complete justification cannot be reached: the so-called *Gettier cases* illustrate this fact. The second of two examples (*Case II*) in the short work of Edmund Gettier is given as follows [75]:

The main protagonist of the case, named Smith, has justified belief that his friend Jones owns a Ford, since he remembers Jones always driving a Ford. Smith is not aware of the location of another friend, Brown. So Smith constructs the following three propositions with randomly picked places:

i. Either Jones owns a Ford, or Brown is in Boston;
ii. Either Jones owns a Ford, or Brown is in Barcelona;
iii. Either Jones owns a Ford, or Brown is in Brest-Litovsk.

In fact, Jones does *not own* a Ford, he is currently driving a rental car. Additionally, and by sheer coincidence, Brown *is* really in Barcelona – without Smith having a chance of being aware of this. With these two conditions in place, it can be concluded that: Smith does *not* know that (ii) is true, Smith *does* believe that (ii) is true and Smith *is* justified in believing that (ii) is true. This case, together with the first one given in Gettier's work, shows that the JTB definition does not give a sufficient condition for knowing a proposition. [75]

[3]Genitive of *on*; retrieved from `https://www.etymonline.com/word/onto-` on 2020-04-11.
[4]Retrieved from `https://www.etymonline.com/word/-logy` on 2020-04-11.

Attempts have been made to overcome the issues mentioned by Edmund Gettier by amending or extending the tripartite with additional conditions, however, they were found to be insufficient – the Gettier problem persists and so does the gap between JTB and knowledge. [74]

As a consequence of the justification discussion, it cannot be denied that there is the issue of subjectiveness in justification of knowledge. Antoine de Saint-Exupéry gave a relatable example in his famous book *Le Petit Prince – The Little Prince* [76], when the narrator complained about adults not correctly interpreting his drawing of a snake eating an elephant, which he had drawn while being a child: they simply perceived the child's drawing to be depicting a hat. A more technical problem that describes this circumstance in an illustrative manner is how we perceive two-dimensional maps of the world. Those maps are often based on the Mercator projection, which is well established in contemporary life. An example map of the world in Mercator projection is given in Figure 4.3. Such a conformal projection, also known as orthomorphic projection, ensures all angles of all points are preserved throughout the projection [77]. This results in a significant size distortion of objects, which are far away from the equator; a distorted perception of the relative size distribution leads to erroneous beliefs about the actual sizes of countries or regions. An overview of surface areas of selected countries and regions is given in Table 4.1. The numbers reveal that the United States of America only hold one third of the area of Africa. Greenland can size-wise nearly fit fourteen times into Africa. The Russian Federation, a dominant area on the Northern hemisphere, is indeed only approximately twice as big as Australia and New Zealand.

Country or Region	**Surface Area**
Africa (Continent)	30,311,000 km^2
Russian Federation	17,098,246 km^2
United States of America	9,833,517 km^2
Australia and New Zealand	8,012,000 km^2
Greenland	2,166,086 km^2

Table 4.1: **Actual Sizes of Selected Countries and Regions:** This list stands in contrast to the impression one might have from looking at a Mercator-projected world map. The items are ordered by size, numbers accord to the official statistics of the United Nations. [79]

Misconceptions, such as the distorted perception of the Mercator-based world map, are tough to avoid. So, how can something be known? How can it be made sure that a belief becomes knowledge? What is the nature of truth? An attempt to provide an action set for such issues is the neo-classical theory of truth. The contemporary debate is based on three fundamental theories to investigate whether a knowledge claim is indeed true: *correspondence*, *coherence*, and *pragmatism* [80].

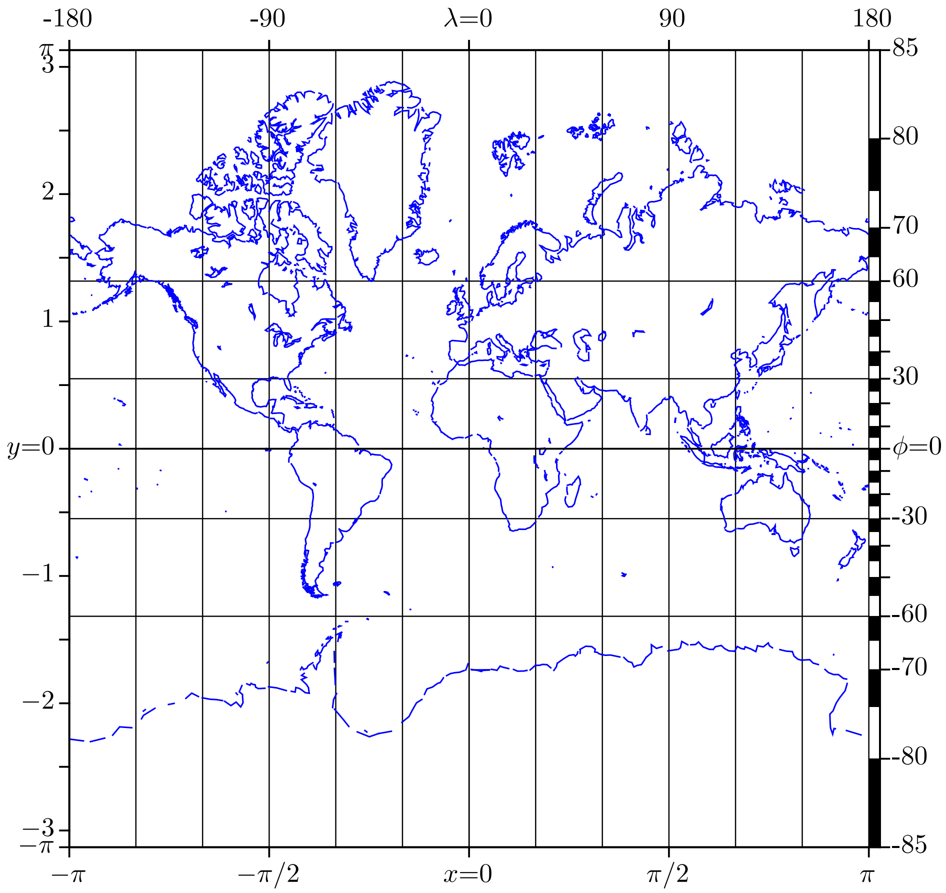


Figure 4.3: **World Map Based on the Mercator Projection:** The conventional world map uses the Mercator projection, which is an angle-preserving *conformal projection*. The projection leads to a size distortion of objects, which are far away from the equator. For instance, this map lets perceive that Greenland is roughly of the same size of Africa, which it actually could fit in nearly fourteen times. See Table 4.1 for size references. [78]

The *correspondence theory* poses the question if a knowledge claim corresponds to a fact. Examples are that the grass is green and the sky is blue. But: Is the perception of a single person necessarily the same as those of others? What is with people suffering from colour-blindness? The *coherence theory* instead investigates, if the claim does fit into (our) current set of beliefs. For instance, Alfred Wegener proposed the theory of continental drift in 1912 [81]. His work was highly disputed, since his theory did not fit into contemporary beliefs. He reworked and extended his theory several times to gain more acceptance amongst peers, his ideas were published in 1929 already as the fourth edition [82]. Today, the theory is established and well-accepted and fits into our current set of beliefs. *Pragmatism theory* asks for the answer to whether a knowledge claim works in practice. A prominent example is given by Newton's law of universal gravitation [83]: although there are other theories, such as Einstein's general theory of relativity [84] that proves Newton's theory is not entirely correct, it was and is still accurate enough for applications in modern science and engineering – the concept is *fit for purpose*.

From the examples given, it becomes obvious that a gap between justified true belief and knowledge persists. Karl Popper contributed his sets of arguments to the discourse and created the fundamentals for critical rationalism or *falsificationism*. According to his theory, entire justification of knowledge cannot be reached and, hence, cannot be the goal of science. Instead he suggested that all theories need to constantly undergo verification and are true (or at least not false) as long as their falseness has not been proven. [85, 86]

Apart from the question *how we can know* there is the question of *what can be known* emerging. Immanuel Kant used two specific terms in his work *Critique of Pure Reason* (originally appeared in German as *Critik der reinen Vernunft*) [87, 88]: *a priori* and *a posteriori*. Those terms are used to distinguish between two forms of knowledge [89,90]:

A priori: Knowledge that is entirely independent from experience, e.g., arithmetic operations, from deductive reasoning, or from tautologies; no scientific research needs to be performed to know the claim in question is true;

A posteriori: Knowledge which is gained by experience or empirical evidence (through experiments).

In his contributions to the theory of knowledge, Kant moreover stated multiple times in the same book that judgements substantially correspond to propositional cognitions; judgments are complex conscious cognitions referring to objects [87,89]. This is of interest to note, since Kant already remarked the significance of the role the cognition of humans plays in the judgement of conclusions.

With the development and emergence of the *linguistic turn* during the early 20$^{\text{th}}$ century, the emphasis in epistemology was shifted towards the significance of language and its use. Besides others, Bertrand Russell and especially Ludwig Wittgenstein were notable contributors to the emerging field, which was concerned with language as a vehicle of thought. Parts of their work was about structure as well as meaning of sentences and its components – including the hierarchical concept in the construction of sentences. [91, 92]

When looking at Wittgenstein's *Tractatus Logico-Philosophicus*, two propositions are of direct interest for this contextual elaboration; 5.6 and 5.632 (according to the original numbering in the cited book) [92]. The first one is:

> *5.6 The limits of* ***my language*** *mean the limits of* ***my world.***

This proposition is of interest, because it depicts the relation of the language user to his or her interaction with the world. Only because one's language "ends", it does not mean his "world" ends – there might be suitable alternatives in other languages, for example: in Japanese exists the word *komorebi*, 木漏れ日. It does not have a direct translation in English, however, a close attempt to define it would be "sunlight filtering through trees"[5]. Still, for the Japanese it is only a name for a frequent sequence of words: three kanji (木, 漏, 日) and a hiragana particle (れ). Another example is given by the English verb *to intuit*[6], since it does not have a direct equivalent in German. However, there

[5] As retrieved from `https://jisho.org/search/komorebi` on 2020-04-07.

[6] Defined as "to know, sense, or understand by intuition"; retrieved from `https://www.merriam-webster.com/dictionary/intuit` on 2020-04-12.

are objects existing in the world that cannot be described by the sole use of language. If somebody tries to describe to someone else how a banana tastes[7] without this person having ever eaten a banana before, it will potentially lie beyond the bounds of possibility of delivering an appropriate description.

The second proposition of direct interest is [92]:

> *5.632 The subject does not belong to the world but it is a limit of the world.*

Proposition 5.632 is able to intensify what proposition 5.6 already hinted: pointing out the existence and significance of subjectivism. *Our* experience is based on *our* cognition and *our* consciousness. Hence, there is no absolute and objective view on the world and its phenomena.

In addition to Wittgenstein's *Tractatus*, Russell contributed to the formation of logical atomism with his work *The Philosophy of Logical Atomism.* The premise Russell claims is that there is a fundamental language, which is put together from not further reducible, atomic facts. Since Russell denies human capacity to be capable of reaching the absolute simple in terms of his logical atomism, he emphasises his theory of relative simplicity in further work: knowledge is still advanced, if components of anything complex are discovered; even if the components themselves remain of complex nature. [93]

A quote from Russell's *The Philosophy of Logical Atomism* summarises the fundamental premise well [91]:

> *In a logically perfect language, there will be one word and no more for every simple object, and everything that is not simple will be expressed by a combination of words, by a combination derived, of course, from the words for the simple things that enter in, one word for each simple component. A language of that sort will be completely analytic, and will show at a glance the logical structure of the facts asserted or denied.*

Certainly, there has been further philosophical work on the ideas of Russell and Wittgenstein. However, the fundamental concepts supports the intention to work towards such a logic language, which can add dimension(s) of meaning to the data acquired from cyber physical systems.

[7]This example was originally provided by Paul O'Leary, which he has used in many discourses on how to experience phenomena and on how to describe such experiences. He is professor at the Chair of Automation at the University of Leoben.

4.3 The Metaphorical Concept of Language

In the context of this work, *language* is portrayed as a metaphorical concept. The significance of metaphors is elaborated on in the book *Metaphors We Live By* by Lakoff and Johnson [94]: although metaphor theory is a subdiscipline of cognitive linguistics, and often only language is directly associated with metaphors, they do exhibit significance beyond pure linguistics. Human thinking, action, and speaking is suggested to follow metaphorical concepts, not only in daily life but also within scientific environments. The concepts also support the evaluation of time series originating from physical systems by using language as a vehicle of information, knowledge, and understanding between time series and domain expertise.

For instance, the concept of *more of form is more of content* addresses the issue of iterative reduplication [94]: the difference between "He ran." and "He ran and ran and ran and ran." lies in the fact that the latter indicates *more* running than the first sentence does. Repetitions in time series are of similar nature – just take the compaction phase of a vibro ground improvement process as an example: the more often a compaction step is repeated, the more work has been performed, which – by considering additional conditions such as treated depth – can lead to the conclusion that a certain ground improvement quality has been achieved. Another example shows, which effect the syntax of a statement can have on its meaning. Consider the following three sentences:

1. Brutus killed Caesar.
2. Brutus caused Caesar to die.
3. Brutus brought it about that Caesar died.

The first one directly links the cause to the effect. The second sentence indicates a different, more remote or indirect causation, since the syntax increases the "distance" between the cause and the effect. The last sentence increases this indirect causation even further. The *closer* the causation is to the effect, the *stronger* is the causal link [94]. In time series analysis, this corresponds to the perception of causality, which can indicate stronger links between causes and effects when the both are closely together – either in a spatial or temporal manner. Effects on the life span of particular machine components may not be linked so easily to particular causes, since they could be distributed over a longer period of operation time. However, in previous work it has been found that there is the strong indication of a direct relationship between identified discrete events of overutilisation and the effect of a decreasing life span of the related component [9].

Metaphorical concepts are also found in the order of terms used in a language. The orientation of concepts within a conceptual system usually has an origin, e.g., for human language this can be a person: the English words *here*, *now*, *up* or *active* indicate an orientation *towards* the canonical person, whereas *there*, *then*, *down* or *passive* are oriented away from the canonical person. Such a subtle metaphorical concept is responsible for letting certain orders appear more normal than others, e.g., the expression *now and then* seems to be more normal than *then and now*. A related issue is found in the concept of *unkown is up* – "Let's *bring it up* for discussion" – and *known is down* – "That *settles* the question". This concept is also integral to how intonation is used in spoken languages: Often, there is a rising intonation at the end of sentences lacking confirmation or questions.

Statements, in contrast, often end with a falling intonation. [94, 95]
The introductory reference to the landscaping device of Mr. Budding also lives on the premise that order is significant in the use of languages: A lawnmower seems *more normal* to the reader than a “mower-lawn”. In time series analysis, it can be beneficial to define the normal or standard operation patterns in the symbolised, language-affine form to determine what is normal for a machine’s operation and what is not. For instance, a drill rig is usually in production when the effector, i.e., a drill bit or vibrator, is active (rotating or vibrating) and is below ground level. An active effector outside ground level is usually considered to be non-productive.

The use of metaphors also refers back to the question of what a valid source of truth is. It needs to be pointed out that the use of metaphors also implies past experience or contextual properties, which leads to highlighting or hiding specific parts or properties. For instance, stating that light consists of waves puts emphasis on wave-like characteristics, however, it neglects and hides the particle-like features. [94]
It always needs to be considered that the evaluation of a language emerging from time series of a physical system must take context into account. This fact emphasises once more the significance of appropriate measures to incorporate domain-specific expertise.

Lakoff and Johnson additionally describe how new metaphorical ideas, i.e., new ways of organising and understanding experience, are aggregated by the combination of simpler conceptual metaphors, which results in the formation of more complex metaphors. A non-technical but demonstrative example is given with *I have fallen in love, but we seem to be going in different directions*. It consists of the following, simpler metaphorical concepts [94]:

- *fall*: Lack of control is *down*;
- *in love*: States are locations (*in*);
- *falling in love*: Changes to new states are motions (*falling*);
- *going in different directions*: Love is a journey (*going*, *directions*).

In a technical context, hierarchical metaphors are of particular interest to wrap complex machine operations into simpler statements to focus on particular portions of the machine’s production process. For instance, the accountable for a machine in question is interested in statements such as *the machine is running OK*. But which underlying metaphorical assumptions do justify, if a machine’s operational state is okay? Considering a machine that exhibits two operation modes: Is it fine when it runs in operation mode 1? May operation mode 2 be better for the current production process in terms of certain components’ life span? The distinction may be bound to situative properties, such as spatial or temporal circumstances. Is a defective sensor for the on-side ambient temperature crucial to the statement of the machine’s state? Probably not, however, a process expert instead might be interested in the value readings, since the ambient temperature might be significant to the quality of the production process. Hence, it can be concluded that the focus of time series evaluation and the metaphors, which are used to formulate the statements, are heavily depending on the enquirer’s contextual setting.

From the examples given, it can be claimed that metaphors possess self-contained value, instead of them just providing simple comparisons [94]. Metaphors are not considered being mappings, they add elements and attribute meaning to a domain; however, the meaning might vary based on context and situation. Still, metaphors in general are an inevitable method to express nuances in the language; this is also true for time series analysis, as the exemplary analogies suggest.

4.4 Mimicking the Mechanisms of Language

One of the core issues of time series analysis of cyber physical systems is how to attribute domain expertise to operational data. The metaphor of language exhibits potential to be of significance in the evaluation of such data. For this reason, an example is given on a mobile mining machine, a bucket-wheel excavator, which is being operated by humans. The sampling rate of the time series acquired from this machine corresponds to 1 Hz. Prior to the introduction of basic mechanisms of language, symbolic analysis has to be introduced as a mandatory initial step to symbolise the time series acquired by systems as described in detail in chapter 2.

4.4.1 Symbolic Analysis of Time Series From Cyber Physical Systems

Symbolic representations of time series are used for two main reasons in the context of this thesis: firstly, to reduce the complexity of the time series to simplify the identification of patterns, and, secondly, to attribute *meaning* to the symbols. The first issue is taken into account by algorithms such as the symbolic aggregate approximation (SAX) [96]. However, such algorithms do not consider the specific nature of the system being observed. When developing symbolic representations for time series from physical systems, the metaphoric nature of the symbols should be considered and human-readable text should be associated with the symbols.

In the present example, we need to address the issue of discontinuity, since the continuous operation of the physical system in question is only observed at discrete times. The result of these observations are discrete time series, which are – by nature – discontinuous at every sampling point. To take this into consideration, *linear differential operators* (LDO) are used to model the dynamics of the observed physical system [97–99]. Consequently, the LDO provide natural links to linguistic elements describing motions or actions, i.e., *verbs*.

Given an ordinary differential equation (ODE) of the form

$$a_n \left(\frac{d}{dt}\right)^{(n)} y(t) + a_{n-1} \left(\frac{d}{dt}\right)^{(n-1)} y(t) + ... + a_1 \frac{d}{dt} y(t) + a_0 y(t) = g(t), \tag{4.4.1}$$

where y is a function of t, y^n denotes its n^{th} derivative with respect to t and $g(t)$ is considered the excitation function, which corresponds to the sensor observations, it can be rewritten with the linear differential operator L,

$$L \triangleq a_n \left(\frac{d}{dt}\right)^{(n)} + a_{n-1} \left(\frac{d}{dt}\right)^{(n-1)} + ... + a_1 \frac{d}{dt} + a_0 = \sum_{i=0}^{n} a_i \left(\frac{d}{dt}\right)^{(i)}, \tag{4.4.2}$$

to yield the equation in matrix-vector form,

$$\mathbf{L}\mathbf{y} = \mathbf{g}. \tag{4.4.3}$$

An LDO is selected in a sense to model the ODE approximating the system dynamics in a generic manner when computing regularised derivatives. This permits estimates for state vectors, which is a step towards assigning symbols.

To symbolise multivariate time series, every signal is statistically evaluated to determine the respective modes; they can be derived from their histograms or entropic distribution (see [65]). The definitions for the modes of a particular signal are stored together with a designated symbol per mode. These symbols can be called *words*, thus all the metadata is saved to a *dictionary*, containing all the identified words. Words for states correspond to *nouns*, while actions or movements are considered *verbs*. In addition to the value range definitions of the modes, as well as their associated words, short identifiers as well as elaborate descriptions can be kept available at the same time. Maintaining such dictionaries benefits exploratory work, where simple and short symbols contribute to an efficient way of working, as well as result presentation for interactions with subject matter experts or specific interest groups, who profit more from eloquent descriptions.

During the analysis phase, LDO are applied to transform time series to symbolic time series, which correspond to sequences of words. If the same symbol consecutively occurs several times, the individual occurrences can be *contracted* to a single word with the number of substituted instances as *predicate*.

An example for the result of such a symbolisation process is given in Figure 4.4: the underlying dictionary contains definitions for a mode **u**, for *moving up*, with a green overlay colour, as well as for another mode **d** for *moving down*, which is associated with a blue colour. The same signal is visualised in both plots on the left side of the figure, but different time intervals are shown. The words with their predicated lengths are listed on the right for both signals: based on the specific sequences of words in these lists it can be concluded that both signals, top and bottom, exhibit the same sequence, but with different run-lengths of the words. This implicitly solves the issue of dynamic time warping; the compression of the symbols is furthermore lossless, since the original sequence can be reconstructed from the predicates.

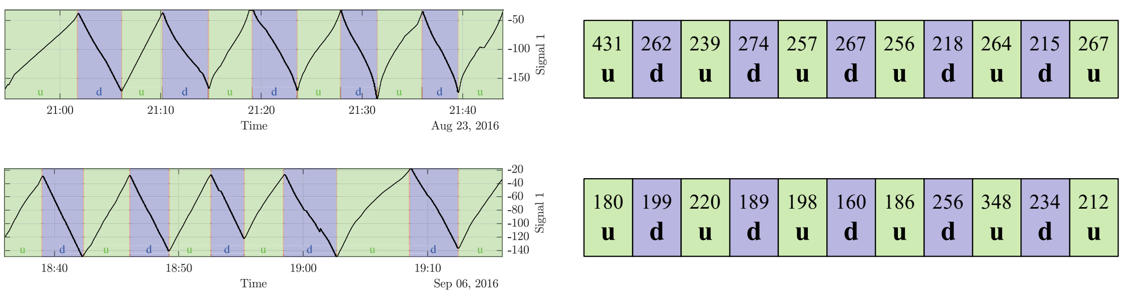


Figure 4.4: **Contraction of Symbolised Time Series:** On the left there are two plots of the same signal, but of different time ranges. The top plot spans about 50 minutes, while the bottom plot covers about 40 minutes of operation. The original signals are overlaid with coloured patches, representing the individual symbols: **u** in green, **d** in blue. On the right, a list of each plot's contracted words is given, together with the corresponding predicates. The predicates correspond to the length of the contracted symbol sequences. This example showcases that, when the predicates are not considered, both symbol sequences are similar to each other. [100]

4.4.2 Monosyllables and Polysyllables

As already shown, the symbolic representations of time series contain significant potential for applying linguistic elements to the symbols. They are advantageous when mechanisms of language are introduced. In the previous step, words in form of nouns, verbs and predicates were defined. By taking a closer look at the predicates, it can be stated that they correspond to modifiers of the nouns and verbs, so they are essentially *adjectives* or *adverbs*.

The Yogācāra[8] school of thought proposes that repetitions in our sensory excitation, which are of significance to our situation, are assigned a language representation – words. The use of a word is considered to be a metaphor, which points to a specific sensory experience, rather than directly describing a experience. *Monosyllabic* words are considered to be a representation of simpler sensory experience. More abstract experiences, such as complex multi-sensory experiences, tend to be described with *polysyllabic* words. In a linguistic context, the nouns and verbs are fundamentally *monosyllables*. Contracting sequences of consecutive same symbols do again yield monosyllables, but with predicates (adjectives). However, if different symbols are *compounded*, they form a new symbol corresponding to a *polysyllable*. Polysyllabic words may consist of combinations of symbols from one and the same signal, of symbols from different signals (cross-channel), or – on a higher hierarchical level – of both. The polysyllables obtained in this manner are new elements in the dictionary based on monosyllables. Such a polysyllable dictionary can be created prior to the analyses in case they are known, or, via iteration through the required steps of compounding (using a specified set of rules) until the desired result is obtained. An example of compounding is given in Figure 4.5: On the left, plots of two individual signals are presented above each other with their symbolised sequences of monosyllables overlaid. Both signals are of the same machine and of the same time range. Under each signal plot, the contracted words can be found (the predicates are not shown to maintain simplicity). The initial contraction step has already been performed. To compound the symbols of both signals, they are combined at all occurrence times to form novel polysyllables, which establish another hierarchy level (since they can be simplified to monosyllables); this yields the plot on the top right. In a next step, combinations of polysyllables, which have a higher Markov probability than others, can be compounded to gain the next level of hierarchy, as shown in the bottom plot on the right.

The sequences yielded from a single or multi-iteration compounding process provide a summarising overview of all existing word combinations. The description of the symbols or words can already provide a powerful tool to domain experts: by ordering the detected sequences by their occurrence probability during operation, a *frequency dictionary* for the machine's operation is generated. This is of particular interest in the context of commissioning support, as discussed in chapter 2. A characteristic profile for the response behaviour of a physical system can be derived in this manner; for operation process evaluations, or for a reference profile for machines of similar type.

[8]The concepts and models of the Yogācāra school of thought [101] are discussed in more detail in section 4.5, where it is also put into a phenomenological context for this work.

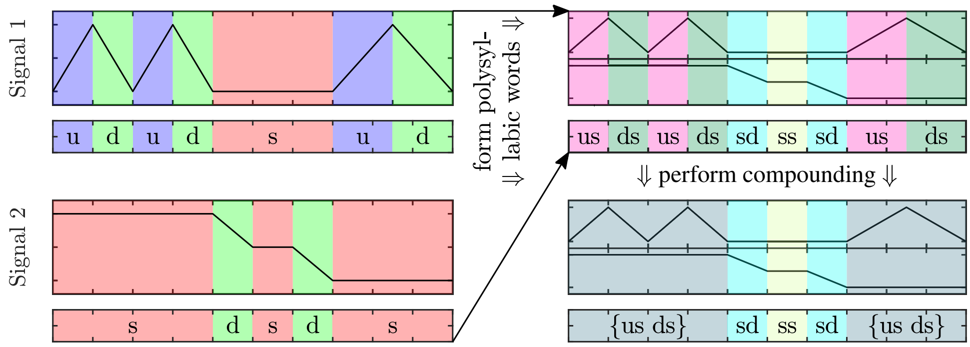
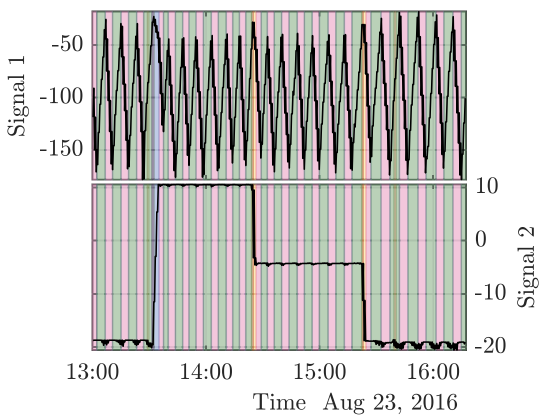


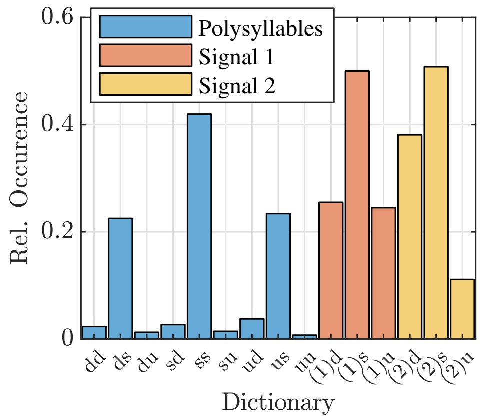
Figure 4.5: **Hierarchical Compounding:** Plots of two individual signals of a physical system are given on the left, both plots cover the same time range. The contracted versions of the monosyllables are visualised as overlays on the original time series. An initial compounding step, which combines the monosyllables of both signals vertically, yields the intermediate result plot on the top right. In a next step, common sequences are compounded horizontally – the result can be seen in the bottom plot on the right side. [102, cf.]

There are two plots on the left in Figure 4.6, where each corresponds to a different type of operation (top plot: *terrace-cut* operation; bottom plot: *drop-cut* operation; [36]). The signals of both plots are again the same, but showing different time ranges. Both signals have been symbolised and to each three monosyllabic words were attributed: **u** for up, **d** for down, and **s** for stationary (no movement). The words have been contracted and once compounded to yield multi-channel words – no further hierarchy has been introduced. The overlays visualise the cross-channel polysyllables. The respective histogram is given on the right side of each row: polysyllables are plotted in blue, the monosyllables in orange and yellow (the predicated lengths of the monosyllables have been considered in the histogram). A visual comparison of the statistics of the monosyllabic words do not reveal major differences between both histograms. However, the distribution of the polysyllabic words indicates that both time series correspond to different operations modes. This emphasises the value of having compounding mechanisms at hand when evaluating such kind of time series.

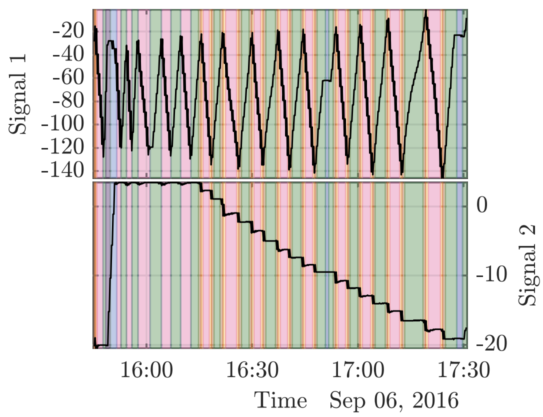
As presented, the mechanisms of language introduced follow a clear and structured set of rules: the *grammar* of the machine language. The language-metaphorical capacity of the symbolised time series can be even further increased by introducing rules for punctuation. Intended pauses, shift changes, downtime due to production or machine issues can be assigned a punctuation representation. This would contribute significantly to the understanding of the nature of longer time series, on which subject matter experts may need to segment based on operational conditions, rather than on rigid temporal conditions.



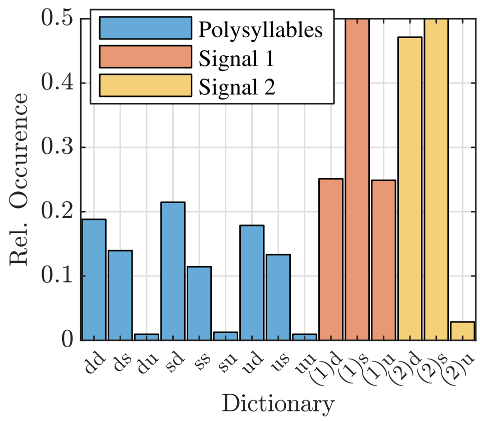
(a) Time series of two signals which characterise a *terrace-cut* operation; the polysyllables from an initial vertical (cross-channel) compounding are illustrated with colour patches



(b) Histogram showing the normalised distribution of monosyllables (Signal 1 and Signal 2) and compounded polysyllabes of the *terrace-cut* operation



(c) Time series of two signals which characterise a *drop-cut* operation; the polysyllables from an initial vertical (cross-channel) compounding are illustrated with colour patches

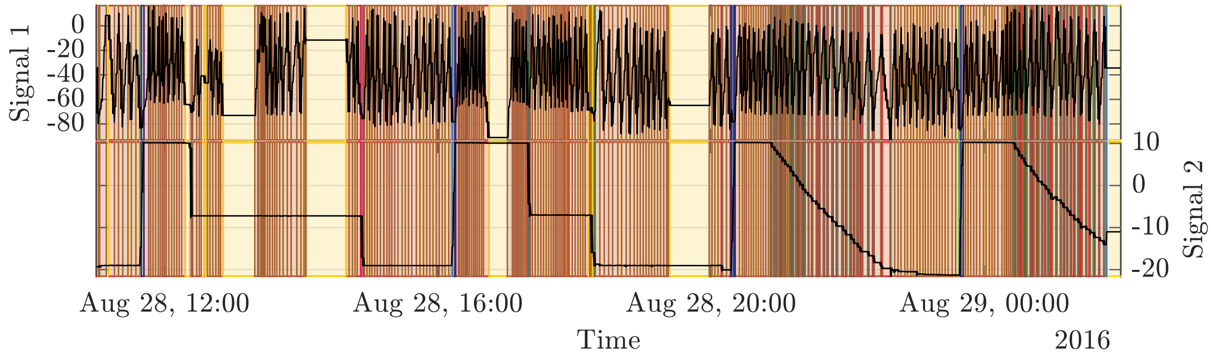


(d) Histogram showing the normalised distribution of monosyllables (Signal 1 and Signal 2) and compounded polysyllabes of the *drop-cut* operation

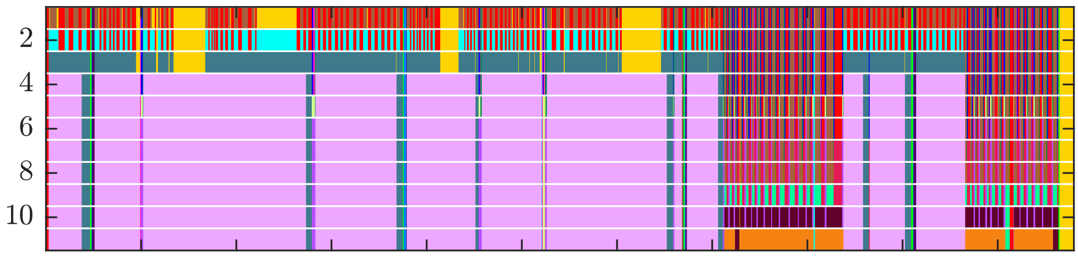
Figure 4.6: **Distinction of Operation Modes:** Each row shows a different operation mode of a bucket-wheel excavator: terrace-cut (operation mode 1) at the top, drop-cut (operation mode 2) at the bottom. The time series plots of two signals, which represent the respective operation mode, are given on the left of each row. The coloured patches, which overlay the time series, indicate the polysyllables compounded from the monosyllables of Signal 1 and Signal 2. The histograms of the normalised, relative distribution of the mono- and polysyllables per operation mode are given in the right column. [100]

4.4.3 Detecting Implicit Hierarchical Structure

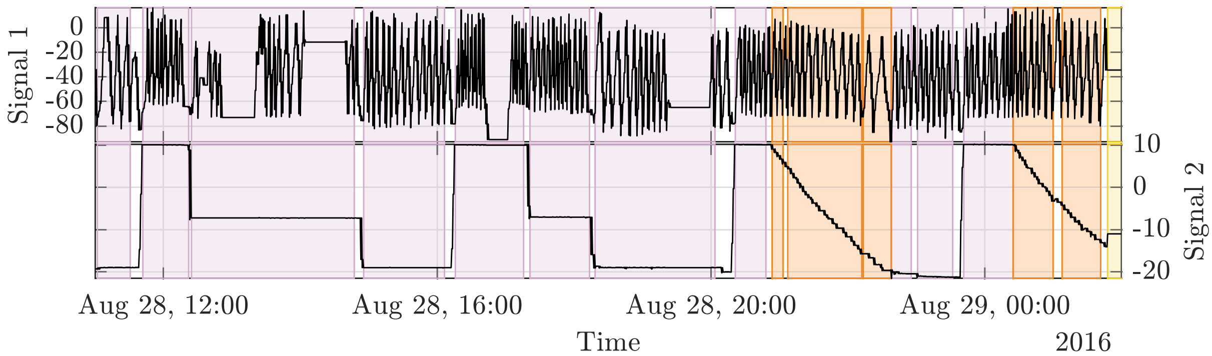
The last example of this section is intended to investigate implicit hierarchical structures, which may exist in the operational data. The signals are the same as in the previous example, the definitions of the monosyllables remain unchanged. The monosyllables are contracted and then compounded between both signals to generate the first level of polysyllables: this yields a total of 9 definitions in the dictionary for the polysyllables, which correspond to the definitions of the previous example. A plot of the investigated time series signals is given in Figure 4.7a. This plot is considered the initial level 1; the represented time range contains 712 polysyllabic words. In an iteration step, the most frequent words are compounded to form new polysyllabic words. This iteration is repeated until a stop criterion is reached – this is the case after 11 iterations, since no further significant reduction of word count was observable. All 11 iterations are illustrated in Figure 4.7b, the colours indicate individual words. Each row corresponds to an iteration step. The last step, level 11, is plotted in Figure 4.7c: the count of words has been reduced to 67 words. This plot illustrates – in a quite apparent manner – that there is implicit structure existing in the time series of the bucket-wheel excavator. Both operation modes, which are contained in the time series sequence, were identified automatically, solely by compounding the most frequent words of an iteration level. In the resulting plot the operation mode *terrace-cut* has been overlaid with purple colour, while *drop-cut* operation has been overlaid with orange colour patches.



(a) Iteration Level 1: The initial level contains 712 polysyllabic words.



(b) All intermediate iteration steps are illustrated from level 1 to level 11.



(c) Iteration Level 11: The last level contains 67 polysyllables.

Figure 4.7: **Detection of Implicit Hierarchical Structure in Time Series:** The iteration process starts at level 1 with an initial set of 9 definitions for polysyllables. In each iteration step, the most frequent words are compounded to form new words. The iterations are stopped when a certain stop criterion is met. In this example, the iteration is repeated until level 11 is reached, since no significant further reduction of word count was observable. The result on the bottom shows that two operation modes became visible from the process of revealing implicit, hierarchical structure. [100, cf.]

4.5 The Phenomenological Aspects of the Emergence of Language

This section touches "understanding" in exploring parallels between detecting implicit hierarchical structure in data from cyber physical systems and the emergence of natural language. When observing a physical machine's response behaviour to external loads and operational environments, we are actually looking at a particular *sequence of phenomena* – observable events. The investigation of issues, which arise from the observation and exploration of such sequences, demands a structured approach. To explore this propositional thought on the sequential occurrence of phenomena more deeply, *phenomenology* has to be considered: phenomenology is the study of phenomena, it studies the structures of experiences and consciousness [103]. As a branch of physics, it is defined as: *a body of knowledge that relates empirical observations of phenomena to each other, in a way that is consistent with fundamental theory, but is not directly derived from theory* [54]. The word *phenomenon*, plural *phenomena*, is derived from the Greek word φαινόμενον, *phainómenon*, meaning "that which appears or is seen"[9]. The Proto-Indo-European root is **bhā-**: "to shine" [50]. A phenomenon is opposed to a *noumenon*. The latter derives from Greek νοούμενον, *nooúmenon*[10], which is a form of νοεῖν, noeîn, which means "to perceive by the mind, to apprehend" [104]. Immanuel Kant defines the contrasting terms as follows [88]:

> *Appearances, to the extent that as objects they are thought in accordance with the unity of the categories, are called phaenomena. If, however, I suppose there to be things that are merely objects of the understanding and that, nevertheless, can be given to an intuition, although not to sensible intuition (as coram intuiti intellectuali [by means of intellectual intuition]), then such things would be called noumena (intelligibilia).*

From all these definitions, it can be concluded that a phenomenon is everything that can be experienced by the (human) senses, while a noumenon is an object of intellectual intuition, of cognition.

Phenomenology as a term itself has already been used by Immanuel Kant et al. since the 18$^{\text{th}}$ century [103, 105]. Edmund Husserl[11] is considered the founder of Western phenomenology, as it is conceived in a contemporary sense [100, 103, 106]. Husserl states that the world is not an abstract philosophical problem, it is made of complex structures of meaning that are grasped by human experience [107]. Husserl concerns part of his work with fundamental contemplations of Immanuel Kant and René Descartes [106]. The latter was reputable for his fundamental scepticism of how knowledge can be acquired and that all objects and all forms of certainty are mediated by the mind [108]. However, controversy originated from Descartes' statements, especially on his view on subjectivism and his scholastic perspectives. Still, Russell emphasised that Descartes insisted thoroughly

[9]Retrieved from `https://www.etymonline.com/word/phenomenon` on 2020-04-11.

[10]Retrieved from `https://www.etymonline.com/word/noumenon` on 2020-04-11.

[11]Notably, Edmund Husserl graduated from the University of Vienna with a doctoral thesis on the theory of variations in theoretical mathematics. [106]

not to accept foundations laid by predecessors [108]. Although neither Husserl nor his disciples adopted the entirety of Descartes' work, they appreciated his disruptive way of thinking and his investigation of the cognitive capacities of the "knower" [109]. Martin Heidegger[12] was Husserl's assistant at the University of Freiburg and later his successor [106, 110]. One of Heidegger's main works was *Sein und Zeit* [111], which has been translated to *Being and Time* [112]. Heidegger criticised that philosophers since Plato appeared to utilize some form of higher ground for validating experiences [113]: Plato had the ideal world, then god was declared, and Descartes brought in the subject's experience of being. Although Husserl put emphasis on a holistic understanding of experiences, he needed to refer to transcendental essences as mandatory aspects of explaining how the world can truly be known [113]. Heidegger, although expressing his friendship and admiration to Husserl in the dedication of *Sein and Zeit*, represented an entirely different standpoint: Heidegger argued that the meaning of a phenomenon is a result of how a person comes to make sense of a particular aspect of the world by using historical and holistic aspects of it; Heidegger's view on phenomenology is based on a truly holistic understanding of the world, where any aspect can only be understood by knowing the greater context of it [113]. An explanatory example is given by the situation of coming home and being welcomed by the smell of freshly baked cookies [114]: the pleasant associations we have with the smell are neither created by us exclusively nor created at the moment of walking through the door. They are memories of our own experiences we had, maybe from our own baking efforts or from that of our parents'. Social memories, such as the welcoming by a caregiver, are present, as well as deeper memories of being fed and protected by caregivers. These memories are not filed somewhere in our mind and wait to be used to interpret this fragrance; the smell has merely directed our attention to those memories. It is part of a web of relations between our experiences.
One of Heidegger's basic claims is that practise comes prior to theory. He introduces the principles of *ready-at-hand*, from German *Zuhandensein*, which comes first, and *present-at-hand*, *Vorhandensein*, which comes second. The first term describes the practical relationship to things that are useful. The second term denotes the theoretical understanding of the world of objects. [111, 112]
With the position Heidegger adopted, he proposed that knowledge can only be acquired by studying our constant and everyday understanding of the world, which we gain from direct interaction with the objects in it.

Maurice Merleau-Ponty, a French phenomenologist, focused on the problems of perception and *embodiment*. His work was concerned with how we perceive as a result of experiencing phenomena. From his work it can be concluded – in a rather simplified manner – that *we perceive phenomena first, and, afterwards, reflect on them.* [100, 115, 116]

It was Heidegger, who is believed to be the first reputable Western philosopher that received attention in Eastern Asia and, at the same time, is also the first one who was in interaction and discussion with Asian thinkers [117]. In his work *A Dialogue on Language between a Japanese and an Inquirer*, he pointed out hindrances between the understanding of Eastern and Western philosophies. He identified the issues as lying in the barrier of Asian languages, or in the particular case, Japanese language [118]. However, later

[12] Even though Heidegger was a controversial person for his activities during the period around the second world war, his professional work is seen as relevant to this chapter.

reception of Heidegger's work has found that his writings exhibit numerous parallels and adopted thoughts from Asian philosophy [117, 119].

The Eastern view on phenomenological aspects date back longer than the Western school of phenomenological thought. The Eastern philosophical literature about these aspects are heavily depending on Buddhist tradition and texts. In the early phase of Buddhism, multiple communities scattered over the Indian subcontinent developed separate schools of thoughts and teacher lineages. To organise, interpret and revitalise the distinct theoretical interests, a consolidated system of thought has been established, called *Abhidharma*. A potential interpretation of this Sanskrit term is constructed from the words *abhi*, meaning "concerning", and *dharma*, for "teaching(s)". Its theory states that every constituent of the experiential world can be known and named, and, additionally, that words and concepts used in the course of discernment of these constituents uniquely define their corresponding referents. In an epistemological view, the analysis of sentient experience of the Abhidharma suggests that what is perceived as an uninterrupted flow of phenomena is, in fact, a sequence of causally connected moments of consciousness. [120]

The Yogācāra school of philosophy is concerned with the relationships between body, mind and language. The half-brothers Asaṅga and Vasubandhu, both living around the 4th century, are considered the founders of Yogācāra in the area of the Indian subcontinent. It was later brought to more Eastern areas, where it was also adopted. [101, 121, 122]

Yogācāra school of thought was established to address issues from *Abhidharma* thought to elaborate on consciousness. It does not declare per se that the world is made of consciousness, however, it is suggested nothing can be known that is not mediated by consciousness [123]. The ālāyavijñāna as a central focus and fundamental concept is considered as a stream of consciousness [124]. Although Yogācāra is much concerned with language, it does not consider terms such as subject and object, however, it rather distinguishes between *grasping* and *what is being grasped* [101].

One of Yogācāra's constitutional models is the one of *The Five Skandhas*, a linear concatenation of five aggregates [101]. This concept is based on five distinct steps, which are used to structure the path from sensory excitation to discursive thought. Yogācāra asserts that there is no direct contact with objects in the world, but with a model for the world. The five steps of this model are [101, 125, cf.]:

rūpa: *Form*, or *matter*, *materiality that is capable of being sensed*;

vedanā: *Feeling* (not to be confused with emotion) – the three modalities through which anything is cognised: *pleasant* (positive, pleasurable), *unpleasant* (negative, painful), neutral;

saṃjñā: *Perception*, *knowledge which puts together*, *associational knowledge* – classifies and labels (sensory and mental) objects, e.g., the object is *blue*, a *lawnmower*, or fear;

saṃskāra: *Impulse*, or *association*, *embodied conditioning*, *constructing activities*;

vijñāna: *Consciousness*, *that which cognises* (*-jñāna*) *through bifurcation* (*vi-*) – includes awareness of a sensory or mental object and discrimination of its aspects.

The original Sanskrit transcript versions of all individual steps are listed in Table 4.2, along with their English translation most relevant to the focus of this thesis and an attempt to define the terms in a technical context.

Sanskrit	English	Technical Context
rūpa	Form	Context-dependent sensor information
vedanā	Feeling (Sensation)	Low level model-based control
saṃjñā	Perception	Combination of low level data to identify a situation
saṃskāra	Impulse (Association)	Learned situative semi-autonomous behaviour
vijñāna	Consciousness (Discursive Thinking)	Artificial reasoning, e.g., rule systems

Table 4.2: **The Five Skandhas in a Technical Context:** The five distinct steps of the model of how humans interact with the world are listed with their Sanskrit transcript form and their English translations. A potential interpretation of the steps in a technical context is given as well. These definitions can benefit the development of intelligent systems – not only for sensing, but also for initiating actions. [100]

Another important doctrine of the Yogācāra school is the model of the *The Eight Vijñānas*, the concept of *The Eight Consciousnesses*. It relates to the dynamic nature of our experience on how a form of understanding is established from the process of emergent consciousness.

The first six consciousnesses, also referred to as *surface consciousnesses*, are well accepted throughout the still existing Buddhist schools [126]. The first five constituents are sensory-based consciousnesses [101, 127]:

***cakṣur*-vijñāna:** *seeing*-consciousness;
***śrota*-vijñāna:** *hearing*-consciousness;
***ghrāṇa*-vijñāna:** *smelling*-consciousness;
***jihvā*-vijñāna:** *tasting*-consciousness;
***kāya*-vijñāna:** *tactile- or kinetic*–consciousness.

The sextet is completed by *mano*-vijñāna (*conceptual-, mental-, or empiric*-consciousness for ideas) [101].

Each of the surface consciousnesses is discrete and occurs separately from the others. The sixth consciousness, mano-vijñāna as a psycho-cognitive sense for modelling sensory experiences, is of particular interest, since it interprets the other five to a certain extent and is directly related to *manas*[13] [128], which in turn is one of two consciousnesses that are added by the Yogācāra philosophy – the other one being *ālāyavijñāna* [101, 126, 129]. The inner faculty manas (*deluded awareness*) has willing woven into its texture, which implicates that manas discriminates and, hence, leads to dualistic interpretations; it does so by attaching strongly to the result of thinking – it is self-referential [128]. The ālāyavijñāna is

[13]The seventh consciousness, manas, is also referred to as *kliṣṭamanas* or defiled mind [127].

the foundational consciousness, the *all-encompassing foundation consciousness* [101,128]. It is also referred to as storehouse or warehouse consciousness, however, using such simple term neglects contextually significant nuances; also see [130]. This consciousness accumulates everything with which it is being concerned with, though it does not possess any discriminative capability and is considered entirely neutral [128]. It is asserted to be a non-determining cognition of what appears to the objects of the underlying cognitions, rather than being concerned with the objects themselves [126]. The defiled mind, manas, adds an enduring aspect of selfness to the momentary flow of the ālāyavijñāna, resulting in a distinction of an inner *perceiving* subject and outer *perceived* objects as separate entities [127]. The ālāyavijñāna is not deluded with such dualistic tendencies, it is further asserted to be a revaluated version of saṃskāra (the fourth Skandha) [101]. All vijñānas are listed in Table 4.3, where they are associated with potential English translations and corresponding physical forms (*what is observed*), type of cognitions (*what is cognised*) and used "sensor" (*with which is observed*).

Vijñānas		**Associated Non-Static Phenomena**		
Sanskrit	**English**	**Physical Form**	**Type of Cognition**	**Cognitive Sensor**
cakṣur-vijñāna	Seeing-consciousness	Sight	Seeing	Eyes
śrota-vijñāna	Hearing-consciousness	Sound	Hearing	Ears
ghrāṇa-vijñāna	Smelling-consciousness	Smell	Smelling	Nose
jihvā-vijñāna	Tasting-consciousness	Taste	Tasting	Tongue
kāya-vijñāna	Tactile-/kinetic-consciousness	Haptic feeling	Touch	Body
mano-vijñāna	Conceptual-/mental-/empiric-consciousness	Thought	Ideation	Mind
manas	Focusing; deluded awareness	Self-grasping	Disturbing emotion or attitude	Mind
ālāyavijñāna	All-encompassing foundation consciousness	Memory	Reflexive awareness	Mind

Table 4.3: **The Eight Vijñānas:** This list contains the Sanskrit transcript versions of The Eight Consciousnesses, along with an interpretation of their probable English translations. Each consciousness has its associated non-static phenomena attributed, to address the questions *what is observed*, *what is cognised*, and *with which is observed*. [101, 124, 127, 128, 130, 131, cf.]

The five aggregates, together with the eight consciousnesses, support the view of life as a continuous flow of sensory experience, to which meaning and significance is added. An attempt to combine both models, based on the numerous sources employed for the interpretations of this section, is given in Figure 4.8. The attempt considers various interpretations of phenomenological and related theories from the *Abhidharma* thought, the *Yogācāra* school of philosophy, as well as from *Mahāyāna*-related concepts, such as the Laṇkāvatārasūtra and, hence, *Chan* as well as *Zen* thoughts.

In the illustration, the vijñānas are organised in a linear manner to allow a categorical association with the distinct steps of the skandhas. The skandhas are given as a staircase of the distinct steps the model is consisting of. Such a representation supports in providing an attempt to correlate both models. The path of how we come from *Form* to *Consciousness*, *perception*, is illustrated with red arrows. They indicate the incremental transitions from one aggregate to the other when something is *perceived*. In contrast, when something is *concepted* in the mind, the blue arrows indicate the increments from bottom up and towards left. An example following the perceptive path can be stated as follows: We *hear* a *sound* with *our ears*; we have a *neutral feeling* about it; we *perceive* it to be a lawnmower; we *associate* this nearby lawnmower to be used by our neighbour; we start to *discursively think* about when we should mow our lawn next. Also different kinds of responses emanating from mental consciousnesses can be described, when a situation requires them. Examples from everyday life: an infant stumbles besides us and we reach out to prevent falling – we did not even perceive what happened or who stumbled when we reached out to help (*semi-automatic response*, originating from *feeling*, *mano-vijñāna*); we approach a junction and the car coming from the left does not decelerate, we are getting slower and increase our attention (*conditioned response*, originating from *impulse*, *manas*); you hurt yourself and you need immediate medical attention, so you make an emergency call (*premediated response*, originating from *consciousness*, *ālāyavijñāna*). In a system which is dynamically adapting to varying operation scenarios, it can occur that some responses may be shifted to the right, towards "more thinking". Just imagine driving a car in a country with a different driving direction, opposed to what you are used to: surely, the first days you will be more actively aware of certain aspects of driving, until "your system" adopts to the new environment and shifts your response behaviour back towards more subtle regions of your mind.

As previously discussed, language is used as a metaphorical reference to the mental content of an experience. Language is asserted to emerge within the mental consciousnesses as illustrated in the figure. This implies that all aggregates are part of such an emergence process. With this illustration, the concept of language is interpreted as a metaphor for how experiences are understood. Additionally, as language emerges throughout the proposed, combined model, it can establish or exhibit variable stages or differential strengths and significances of metaphorical references.

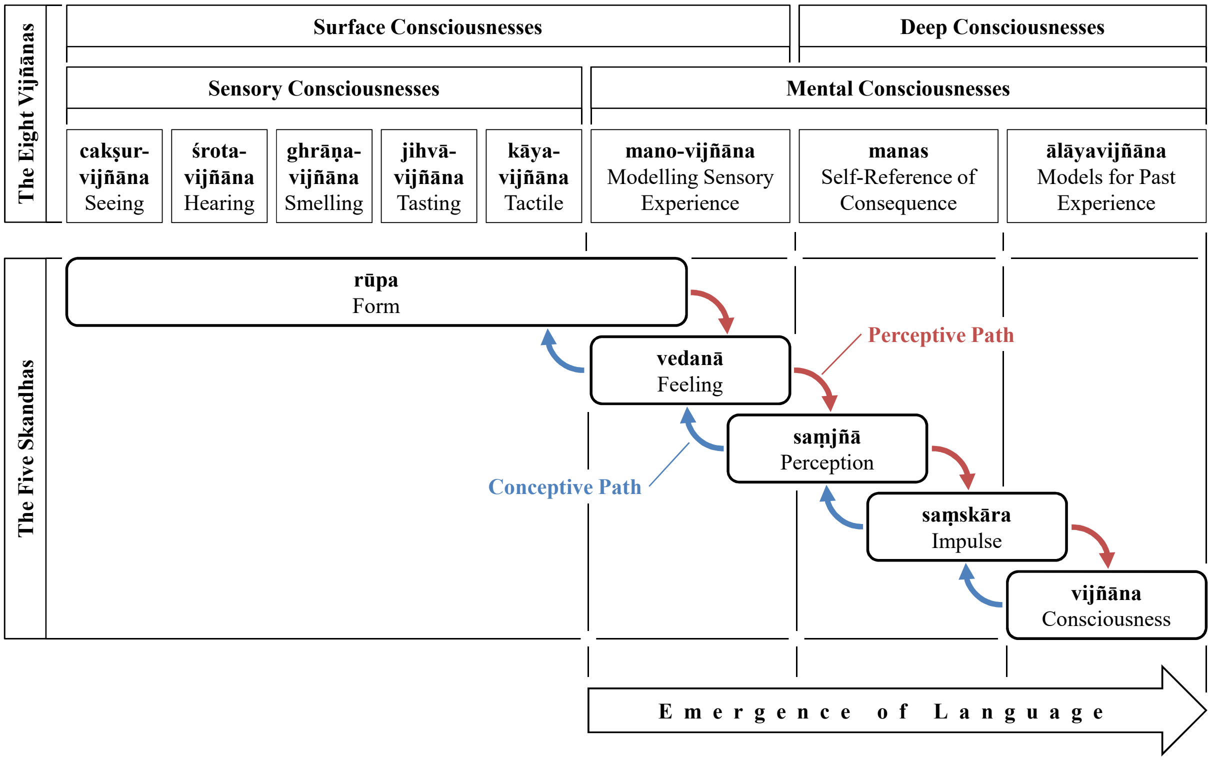


Figure 4.8: **Concept of the Emergence of Language:** This attempt to combine the doctrine of The Eight Vijñānas with the model of The Five Skandhas illustrates potential relationships and dependencies between the constituents of both concepts. The path from *Form* to *Consciousness* is highlighted with red arrows and corresponds to the process of *perceiving* phenomena. The other direction would coincide with a *conceptive path*, which is indicated with blue arrows. Language is considered to emerge within all five distinct aggregates and in the domain of the mental consciousnesses *mano-vijñāna*, *manas*, as well as *ālāyavijñāna*. [100, cf.]

4.6 Chapter Conclusions

Although there are many approaches on how to conquer time series, they often do not take the specific nature of their source into account. Cyber physical systems are bound to physical laws and, hence, the time series representing their operation do not exhibit the same characteristics as time series from other domains; such as the financial or medical sector, which have other properties that are not directly dependent on physics. When the response behaviour to environmental load scenarios of such systems is to be evaluated, the systems' dynamics are mandatory to be considered; this is also true for the influences from human operation. The common sampling rate of the described mobile machines usually corresponds to 1 Hz, which is sufficient for assessing process- or machine-related aspects in the presence of human interaction.

Knowing *how* something works does not automatically equal knowing *why* something works. Purely perceptional approaches, such as various learning methods, can be configured to produce impressive results. Still, the mechanisms behind those methods remain of perceptional nature and, hence, are not able to conceptualise. For instance, learning methods can – if set up and trained accordingly – discriminate whether a sheep or a lawnmower is pictured in a sample. However, a system driven by learning methods potentially fails to properly draw a sheep by itself, since it lacks the capability of intuitive conception. It does not know what a sheep is. This is also the reason, why a sheer perceptional system is able to distinct a sheep from a lawnmower, while it is not able to possibly conclude in a conceptional manner that both entities might be capable of mowing lawn. Furthermore, it would potentially classify "having a roast leg of lawnmower" as pure nonsense.

It has been shown that knowledge is bound to many conditions and dependencies. This must always be considered when approaching any data analysis task. It is furthermore noted that knowledge tends to be subjective and requires contextual, domain-specific interpretation. The task of establishing a sufficient connection from subject matter experts to the time series – and, hence, to the physical systems – can be significantly improved and supported by the metaphor of the emergence of language. The metaphorical concept of language and its elements can be beneficial in adding meaning to certain aspects of the time series.

When modes of knowledge acquisition are discussed, also the question is raised, whether there can be belief without language [73]: it can be clearly negated in the context of this work. As shown in this chapter, there can not be any knowledge without some form of belief. Language is considered to be the vehicle of thought between time series of a physical system and respective subject matter experts. An approach to extend the basic mechanisms of language with metaphorical meaning is found in the combination of Eastern doctrines, which emanate from a phenomenological context. Time series representing the operation of a physical system are treated as a continuous flow of experiences, which follows the concepts suggested in the Yogācāra school of thought. The insight that implicit hierarchical structure is detectable in an unsupervised manner, shows that the adoption of mechanisms of language can be of significant benefit when analysing time series of a machine operated by humans.

Chapter 5

Conclusions and Outlook

A major part of this work is concerned with the development and establishment of an infrastructure to acquire time series data in real-time machine environments. The focal points necessary to be taken care of when creating the concepts for such frameworks are numerous, however, they are of similar significance for the process of collecting, handling, and provisioning data. Although the systems for both industries, mining and materials handling, as well as geotechnical engineering, share many similarities in structure and processing, they exhibit many individual aspects, such as the means of data acquisition (continuous versus element-based) or data handling and provision (dedicated cluster versus cloud environment). However, the more it is discussed about the validity of a data source, the more it is important to be aware of the circumstances of the data collection process and its constituents. As pointed out in the last chapter, the trustworthiness of data – and that of its source – is of direct relevance for gaining knowledge. As a consequence, data consumers put more trust into data, which has been collected using their own methods and systems, than into data acquired via other means they are less familiar with: data may have been filtered, altered, or portions have been left out – unintentionally or on purpose. Especially the fact of selecting a specific temporally or spatially limited amount of data from a bigger set already equals a data manipulation. To be aware of such influencing factors may in some cases not have a direct impact on the time series evaluation processes, however, they are crucial for the interpretation of derived results. The availability of data sets via a flexible and robust framework as developed is integral to approaching the evaluation of time series data in novel manners. Without such a platform for providing data of heterogeneous machine fleets to the evaluation end points, no advanced methods of data analysis can be applied.

In this work, the approach based on features of natural language is put into a metaphorical context. It is asserted that the whole data analysis process possesses implicit metaphorical potential, also beyond linguistics. This becomes visible when selecting time series out of a longer or bigger set, or even by selecting specific channels out of an array of many: whenever focus is put on something particular, it is automatically implied that other aspects are neglected or that at least the strength of their influence is diminished. An indicator hypothesis preceding an analysis already casts the potential outcome and, hence, has implications on the results. Metaphors bear potential to add meaning, however, it

always must be kept in mind that a hypothesis, which a specific analysis is based on, is already an initial form of manipulation of the outcome or result, since emphasis is put on a particular issue while implicitly neglecting other aspects. The presented conventional or "classical" approaches on time series analysis exhibit such behaviour quite clearly: the mode of how a time series cluster is marked as outlier has direct influence on the overall results of the analysis.
Furthermore, it is concluded that the involvement of subject matter experts in the data evaluation process is not only a benefit; it is a necessity to derive insights of significance.

The novel approach of mining data of a cyber physical system utilises the metaphorical concept of the emergence of language, which is brought into the context of time series analysis. The symbolised form of time series is of significant support in implementing language elements in a manner additional meaning can be added to the individual symbols. With only a basic set of language elements defined, it was possible to detect implicit hierarchical structure within a data set in an unsupervised manner. This contribution already points towards artificial intelligence, since the resulting sequence of elements bears the entire metaphorical capacity of all its constituents, as well as its very own. This work puts emphasis on the framework mandatory to have operational data available in time series format, which is – in a subsequent step – the input for approaches based on the mechanisms of language. Much potential lies in adding additional meaning to pauses, interruptions or missing data; it might be of benefit to have a closer look on punctuation features to address this in an appropriate manner. The metaphorical concept is also not bound to sole linguistic elements; graphical icons can be seen as a novel form of metaphors. To lower the barrier between the language emerging from a machine and the language of a result consumer, e.g., an operator or analyst, illustrative icons can be of benefit. This also became obvious during the development of the HMI portion of *System II*, where icons are used wherever possible to bundle the metaphorical potential in a graphical symbol in order to overcome the issue of misunderstanding words in a language foreign to the consumer; in this case the machine operator.
The results of detecting implicit structure in the data in an unsupervised manner allow the conclusion that the mechanisms of language bear significant potential in intensifying the interaction between subject matter experts and the data availability framework. The proposed model for the emergence of language provides a structured approach for conquering time series data. It establishes an awareness for the distinct steps involved in evaluating data in a novel manner.

The model of emergence of language can be of additional support for the design of intelligent sensor systems in future work: clearly distinct steps are at hand, which can be utilised in a technical context. This can be of significance for coming from an indicator hypothesis to a metaphorical language-similar output, which is human-readable. This would benefit the development of intelligent sensor systems in terms of structure and demarcation of authority and logic of the distinct steps. Also the response behaviour of a system can be organised in a more structured manner, if the different kinds of response classes are implemented according to the proposed model. For instance, we all are certainly not aware of each tiny movement of our legs and entire body during the process of walking. Still, when we barefooted step on a LEGO® brick, we instantly are aware of the contact surface between our feet and the floor (or, parts of it directly with the brick) and take appropriate corrective action by redistributing the pressure our body exerts on the particular foot. The model also implies that our mind concludes that we have two feet while walking. However, this justified true belief lacks confirmation until there is actually pressure on the soles of our feet. As long as there is no contact, no form, there is no confirmation of the body part's existence. Simply, because it is not mandatory for the process controller (our mind) to confirm the part's existence mid-air. The underlying concept can be of benefit in novel control paradigms, where a system's attention can be pointed at the subprocesses of temporal relevance, while other, low-level subprocesses can run at an increased level of autonomy. One of the many benefits is the awareness that all interactions between a machine or a sensor and its respective working environment follows the sequence of the distinct steps and allows to adopt subsystems or subsequent processes in a structured and organised manner.

Bibliography

[1] E. G. Passmore, “Improvement in lawn-mowers,” Patent USRE8560E.

[2] EDWIN BUDDING AWARD. The Golf Course Superintendents Association of America (GCSAA). (Retrieved on 2020-03-16). [Online]. Available: https://www.gcsaa.org/about-gcsaa/awards/edwin-budding-award

[3] D. Roberts. A new solar heat technology could help solve one of the trickiest climate problems – Making high-temperature industrial heat from sunlight. (Retrieved on 2019-11-20). [Online]. Available: https://www.vox.com/science-and-health/2019/11/19/20970252/climate-change-solar-heat-heliogen-csp

[4] D. Lee. LaLiga’s app listened in on fans to catch bars illegally streaming soccer. (Retrieved on 2019-07-15). [Online]. Available: https://www.theverge.com/2019/6/12/18662968/la-liga-app-illegal-soccer-streaming-fine

[5] Official U.S. government information about the Global Positioning System (GPS) and related topics. GPS Accuracy. (Retrieved on 2020-02-12). [Online]. Available: https://www.gps.gov/systems/gps/performance/accuracy/

[6] S. Weckert. Google Maps Hacks – Performance & Installation, 2020. (Retrieved on 2020-02-05). [Online]. Available: http://www.simonweckert.com/googlemapshacks.html

[7] A. Holmes. An artist wheeled 99 smartphones around in a wagon to create fake traffic jams on Google Maps. Business Insider. (Retrieved on 2020-02-05). [Online]. Available: https://www.businessinsider.de/international/google-maps-traffic-jam-99-smartphones-wagon-2020-2/

[8] R. Agarwal, S. Chandrasekaran, and M. Sridhar, “McKinsey Global Institute – Imagining Construction’s Digital Future,” 2016.

[9] C. J. Rothschedl, “Condition Monitoring of Large-Scale Slew Bearings in Bucket-Wheel Boom-Type Reclaimers,” Diploma Thesis, Chair of Automation, University of Leoben, 2016.

[10] T. Vigen, *Spurious Correlations – Correlation does not equal Causation.* Hachette Books, 2015.

[11] R. Ritt, P. O'Leary, C. J. Rothschedl, and M. Harker, "Advanced Symbolic Time Series Analysis in Cyber Physical Systems," *Proceedings of the International Work-Conference on Time Series Analysis, ITISE 2017*, vol. 1, pp. 155–160, September 2017.

[12] Verizon, "2014 Data Breach Investigations Report – DBIR," 2014.

[13] ——, "2019 Data Breach Investigations Report – DBIR," 2019.

[14] Siemens, "SIMATIC – Process Control System PCS 7 – Security Concept PCS 7 & WinCC (Basic) – Function Manual," 2016.

[15] G. I. Caesar, *Commentarii de bello Gallico. The Gallic War*, H. J. Edwards, Ed. Harvard University Press, 1917, (Translated by H. J. Edwards).

[16] B. Schneier. (2016) Integrity and Availability Threats. (Retrieved on 2020-01-06). [Online]. Available: https://www.schneier.com/blog/archives/2016/01/integrity_and_a.html

[17] H. Ning and H. Liu, "Cyber-Physical-Social Based Security Architecture for Future Internet of Things," *Advances in Internet of Things, 2012, 2, 1-7*, January 2012.

[18] Bundesamt für Sicherheit in der Informationstechnik, "Sicherheitsspezifische Empfehlungen für Maschinenbauer und Integratoren – BSI-CS 106, Version 2.0," 2018.

[19] Verein Deutscher Ingenieure/Verband der Elektrotechnik, Elektronik, Informationstechnik, "VDI/VDE 2182 – IT-Security for Industrial Automation," 2020.

[20] C. Richardson and J. R. Rymer, "Vendor Landscape: The Fractured, Fertile Terrain Of Low-Code Application Platforms – The Landscape Reflects A Market In Its Formative Years," 2016.

[21] D. L. Mills, "Network Time Protocol (NTP)," RFC Editor, RFC 958, September 1985, (Retrieved on 2020-01-10). [Online]. Available: https://www.rfc-editor.org/rfc/rfc958.txt

[22] Gartner. 5 Trends Appear on the Gartner Hype Cycle for Emerging Technologies, 2019. (Retrieved on 2019-08-29). [Online]. Available: https://www.gartner.com/smarterwithgartner/5-trends-appear-on-the-gartner-hype-cycle-for-emerging-technologies-2019/

[23] C. J. Rothschedl, R. Ritt, P. O'Leary, M. Harker, M. Habacher, and M. Brandner, "Extended Abstract: Real-Time Data Analytics in Raw Materials Handling," *Real-Time Mining Conference*, October 2017.

[24] W. Feller, *An Introduction to Probability Theory and its Applications, 2^{nd} Edition*, ser. Wiley Publication in Mathematical Statistics. Wiley India Pvt. Limited, 2008, no. 2.

[25] What is OPC? OPC Foundation. (Retrieved on 2020-01-14). [Online]. Available: https://opcfoundation.org/about/what-is-opc/

[26] International Organization for Standardization, "Information Technology – Open Systems Interconnection – The Directory – Part 8: Public-key and attribute certificate frameworks," International Organization for Standardization, Geneva, Switzerland, Standard ISO/IEC 9594-8:2017(en), 2017.

[27] D. Petersen. (2014, May) Cryptographic Algorithms. Institut für Internet-Sicherheit – if(is). (Retrieved on 2016-08-01). [Online]. Available: https://www.internet-sicherheit.de/crypto-poster/

[28] P. O'Leary, M. Harker, R. Ritt, M. Habacher, K. Landl, and M. Brandner, "Mining Sensor Data in Larger Physical Systems," *17th IFAC Symposium on Control, Optimization and Automation in Mining, Mineral and Metal Processing, MMM 2016*, vol. 49, no. 20, pp. 37–42, 2016.

[29] M. Gschwandl, "Data Analytics and Data Mining Methods for Heavy Plant and Machinery," Diploma Thesis, Chair of Automation, University of Leoben, 2016.

[30] K. Landl, "Machine Sensor Data Mining in Bulk Handling Systems," Bachelor Thesis, Chair Of Automation, University of Leoben, 2015.

[31] S. F. Nussdorfer, R. Ritt, C. J. Rothschedl, and P. O'Leary, "Condition Monitoring of Hydraulics in Heavy Plant and Machinery," *Automining 2018*, June 2018.

[32] Unified Architecture. OPC Foundation. (Retrieved on 2020-02-06). [Online]. Available: https://opcfoundation.org/about/opc-technologies/opc-ua/

[33] C. J. Rothschedl and D. Zoglauer, "Technisch/Wirtschaftliche Betrachtung des Schütt- und Stückgutumschlags," Project Thesis, Chair of Conveying Technology, University of Leoben, 2011.

[34] Port Highlights. Mid West Ports. (Retrieved on 2020-01-13). [Online]. Available: https://www.midwestports.com.au/image-gallery/port-highlights/5

[35] L. Machniak and W. Kozioł, "Method of Assessment of Hard Rock Workability using Bucket Wheel Excavators," *Archives of Mining Sciences*, vol. 62, March 2017.

[36] W. Durst and W. Vogt, *Bucket Wheel Excavator*. 3392 Clausthal-Zellerfeld, Germany: Trans Tech Publications Ltd, 1988.

[37] Mobile Crushing Station. (Retrieved on 2020-01-05). [Online]. Available: https://www.flsmidth.com/en-gb/products/material-handling/mobile-crushing-station

[38] K. Ingmarsson and A. Oberrauner, "Combining Flexibility with Conveyor Based Mining," *World Mining Congress*, pp. 37–47, October 2016.

[39] Maior projeto da história da mineração completa um ano de operação. VALE. (Retrieved on 2020-01-14). [Online]. Available: http://www.vale.com/hotsite/PT/Paginas/maior-projeto-da-historia-da-mineracao-completa-um-ano-de-operacao.aspx

[40] Spreaders. Tenova. (Retrieved on 2020-01-14). [Online]. Available: https://www.tenova.com/product/spreaders/

[41] C. J. Rothschedl and C. Eibisberger, "Modern Jet Grouting," 2008.

[42] C. Guan and Y. Yang, "Field Study on the Waterstop of the Rodin Jet Pile Method in a Water-Rich Sandy Gravel," *Applied Sciences*, 2019.

[43] *Soilcrete® (Jet Grouting) – An Efficient and Versatile Method for Soil Treatment*, Keller Grundbau Ges.m.b.H., 2020.

[44] P. Freitag, A. Zöhrer, and T. G. Reichenauer, "Halocrete® – Sanierung von Chlorkohlenwasserstoff-Altlasten mit dem Düsenstrahlverfahren," *ÖGT – 11. Österreichische Geotechnik Tagung*, pp. 185–197, 2017.

[45] G. Byrne, N. Chang, and V. Raju, Eds., *A Guide to Practical Geotechnical Engineering in Africa*. Franki – A Keller Company, 2019.

[46] *TW700 High Pressure Triplex Pump*, Tecniwell, 2018.

[47] *Vibro Techniques – Ground Improvement with Depth Vibrators*, Keller Grundbau Ges.m.b.H., 2020.

[48] A. Zöhrer and V. Winter, "Bodenverbesserung mit Tiefenrüttlern – Moderne Technik mit langer Tradition," *CVK – 31. Christian Veder Kolloquium*, pp. 93–108, 2016.

[49] P. Nagy, "Deep Vibro Compaction – Dynamic Compaction Control Based on the Vibrator Movement," Ph.D. dissertation, Faculty of Civil Engineering, Vienna University of Technology, 2018.

[50] C. Watkins, *The American Heritage Dictionary of Indo-European Roots*. Houghton Mifflin Harcourt, 2011.

[51] E. J. Hagendorfer, "Evaluation of the Potential of Deep Learning for Manufacturing Process Analytics," Master Thesis, Chair of Automation, University of Leoben, 2018.

[52] R. M. Levenson, E. A. Krupinski, V. M. Navarro, and E. A. Wasserman, "Pigeons (Columba livia) as Trainable Observers of Pathology and Radiology Breast Cancer Images," *PLOS ONE*, vol. 10, no. 11, pp. 1–21, 2015.

[53] R. Baheti and H. Gill, "Cyber-Physical Systems," *The Impact of Control Technology*, pp. 161–166, 2011.

[54] P. O'Leary, M. Harker, and C. Gugg, "A Position Paper on: Sensor-Data Analytics in Cyber Physical Systems, from Husserl to Data Mining," *SensorNets*, 2015.

[55] C. W. Groetsch, *Inverse Problems*. The Mathematical Association of America, 1999.

[56] C. Gugg, "An Algebraic Framework for the Solution of Inverse Problems in Cyber-Physical Systems," Ph.D. Dissertation, University of Leoben, 2015.

[57] P. Esling and C. Agon, "Time-Series Data Mining," *ACM Computing Surveys*, vol. 45, no. 1, December 2012.

[58] E. Fuchs, T. Gruber, H. Pree, and B. Sick, "Temporal Data Mining Using Shape Space Representations of Time Series," *Neurocomputing*, vol. 74, pp. 379–393, 2010.

[59] E. Keogh and S. Kasetty, "On the Need for Time Series Data Mining Benchmarks: A Survey and Empirical Demonstration," *Data Mining Knowledge Discovery*, vol. 7, no. 4, pp. 349–371, October 2003.

[60] M. Last, A. Kandel, and H. Bunke, *Data Mining in Time Series Databases*, ser. Series in Machine Perception and Artificial Intelligence. World Scientific, 2004.

[61] P. Chapman, J. Clinton, R. Kerber, T. Khabaza, T. Reinartz, C. Shearer, and R. Wirth, *CRISP-DM 1.0 Step-by-Step Data Mining Guide*, SPSS, August 2000.

[62] *Analytics Services Datasheet: Analytics Solutions Unified Method*, IBM Corporation, 2016.

[63] M. Embrechts, B. Szymanski, and K. Sternickel, *Computationally Intelligent Hybrid Systems: The Fusion of Soft Computing and Hard Computing.* John Wiley and Sons, NewYork, 2005.

[64] R. L. Ackoff, "From data to wisdom," *Journal of Applied Systems Analysis*, vol. 16, pp. 3–9, 1989.

[65] C. Shannon, "A Mathematical Theory of Communication," *Bell System Technical Journal*, vol. 27, pp. 379–423, 1948.

[66] H. Nyquist, "Certain Factors Affecting Telegraph Speed," *Bell System Technical Journal*, vol. 3, pp. 324–346, 1924.

[67] A. Kagan and Z. Landsman, "Relation between the covariance and Fisher information matrices," *Statistics & Probability Letters*, vol. 42, no. 1, pp. 7–13, 1999.

[68] E. G. Guba, *The Paradigm Dialog.* SAGE Publications, 1990.

[69] T. Hofweber, "Logic and Ontology," in *The Stanford Encyclopedia of Philosophy*, Spring 2020 ed., E. N. Zalta, Ed. Metaphysics Research Lab, Stanford University, 2020.

[70] M. Steup and R. Neta, "Epistemology," in *The Stanford Encyclopedia of Philosophy*, Spring 2020 ed., E. N. Zalta, Ed. Metaphysics Research Lab, Stanford University, 2020.

[71] C. J. Wenning, "Scientific epistemology: How scientists know what they know," *Journal of Physics Teacher Education Online*, vol. 5, no. 2, pp. 3–15, 2009.

[72] Platon, *Platon: Die Werke vollständig in deutscher Sprache.*, R. Haller, Ed. Edition Opera-Platonis, 2005.

[73] E. Schwitzgebel, "Belief," in *The Stanford Encyclopedia of Philosophy*, Fall 2019 ed., E. N. Zalta, Ed. Metaphysics Research Lab, Stanford University, 2019.

[74] J. J. Ichikawa and M. Steup, "The Analysis of Knowledge," in *The Stanford Encyclopedia of Philosophy*, Summer 2018 ed., E. N. Zalta, Ed. Metaphysics Research Lab, Stanford University, 2018.

[75] E. L. Gettier, "Is Justified True Belief Knowledge?" *Analysis*, vol. 23, no. 6, pp. 121–123, 1963.

[76] A. de Saint-Exupéry, *Le Petit Prince – The Little Prince*, K. Woods, Ed. Reynal & Hitchcock, 1943, (Translated by K. Woods).

[77] J. P. Snyder and P. M. Voxland, *An Album of Map Projections – U.S. Geological Survey Professional Paper 1453.* Department of the Interior of the United States of America, 1989.

[78] P. Osborne, *The Mercator Projections.* Zenodo, 2013.

[79] United Nations – Department of Economic and Social Affairs – Statistics Division, *World Statistics Pocketbook 2019*, ser. V. United Nations, 2019, no. 43.

[80] M. Glanzberg, "Truth," in *The Stanford Encyclopedia of Philosophy*, Fall 2018 ed., E. N. Zalta, Ed. Metaphysics Research Lab, Stanford University, 2018.

[81] A. Wegener, "Die Entstehung der Kontinente," *Geologische Rundschau*, vol. 3, pp. 276–292, 1912.

[82] ——, "Die Entstehung der Kontinente und Ozeane – 4. Auflage," *Die Wissenschaft*, 1929.

[83] I. Newton, *The Mathematical Principles of Natural Philosophy*, ser. The Mathematical Principles of Natural Philosophy, 1729, no. 1–3, (Translated by A. Motte).

[84] A. Einstein, "Die Grundlage der allgemeinen Relativitätstheorie," *Annalen der Physik*, vol. 354, no. 7, pp. 769–822, 1916.

[85] H. Andersen and B. Hepburn, "Scientific Method," in *The Stanford Encyclopedia of Philosophy*, Summer 2016 ed., E. N. Zalta, Ed. Metaphysics Research Lab, Stanford University, 2016.

[86] K. Popper, *The Logic of Scientific Discovery*, ser. Routledge Classics. Taylor & Francis, 2005.

[87] I. Kant, *Critik der reinen Vernunft*, 1. Auflage ed. Hartknoch, 1781.

[88] ——, *Critique of Pure Reason*, P. Guyer and A. W. Wood, Eds. Cambridge University Press, 1998, (Translated by P. Guyer and A. Wood).

[89] R. Hanna, "Kant's Theory of Judgment," in *The Stanford Encyclopedia of Philosophy*, Winter 2018 ed., E. N. Zalta, Ed. Metaphysics Research Lab, Stanford University, 2018.

[90] G. Rey, "The Analytic/Synthetic Distinction," in *The Stanford Encyclopedia of Philosophy*, Fall 2018 ed., E. N. Zalta, Ed. Metaphysics Research Lab, Stanford University, 2018.

[91] B. A. W. Russell, *The Philosophy of Logical Atomism*, ser. Routledge Classics. Taylor & Francis, 2009.

[92] L. Wittgenstein, *Tractatus Logico-Philosophicus*, C. K. Ogden, Ed. Harcourt, Brace & Company, Inc., 1922.

[93] K. Klement, "Russell's Logical Atomism," in *The Stanford Encyclopedia of Philosophy*, Spring 2020 ed., E. N. Zalta, Ed. Metaphysics Research Lab, Stanford University, 2020.

[94] G. Lakoff and M. Johnson, *Metaphors We Live By*. The University of Chicago Press, 2003.

[95] W. E. Cooper and J. R. Ross, "World Order," *Papers from the Parasession on Functionalism*, pp. 63–111, 1975.

[96] J. Lin, E. Keogh, S. Lonardi, and B. Chiu, "A Symbolic Representation of Time Series, with Implications for Streaming Algorithms," in *Proceedings of the 8th ACM SIGMOD Workshop on Research Issues in Data Mining and Knowledge Discovery*. New York, NY, USA: ACM, 2003, pp. 2–11.

[97] C. Lanczos, *Linear Differential Operators*. SIAM, 1961.

[98] C. Gugg, M. Harker, P. O'Leary, and G. Rath, "An Algebraic Framework for the Real-Time Solution of Inverse Problems on Embedded Systems," vol. V, no. 212. IEEE, August 2015, pp. 1097–1102.

[99] P. O'Leary, M. Harker, and C. Gugg, "An Inverse Problem Approach to Approximating Sensor Data in Cyber Physical Systems," *Instrumentation and Measurement Technology Conference (I2MTC)*, pp. 1717–1722, 2015.

[100] C. J. Rothschedl, P. O'Leary, and R. Ritt, "Mimicking the Mechanisms of Language for the Unsupervised Detection of Hierarchical Structure in Time Series," *Proceedings of the 6th International Conference on Time Series and Forecasting, ITISE 2019*, vol. 1, pp. 155–166, September 2019.

[101] D. Lusthaus, *Buddhist Phenomenology: A Philosophical Investigation of Yogācāra Buddhism and the Ch'eng Wei-shih Lun*, ser. Curzon Critical Studies in Buddhism. Routledge Curzon, 2002.

[102] R. Ritt and P. O'Leary, "Symbolic Analysis of Machine Behaviour and the Emergence of the Machine Language," in *Theory and Practice of Natural Computing*. Springer International Publishing, 2018, pp. 305 – 316.

[103] D. W. Smith, "Phenomenology," in *The Stanford Encyclopedia of Philosophy*, Summer 2018 ed., E. N. Zalta, Ed. Metaphysics Research Lab, Stanford University, 2018.

[104] Perseus Digital Library. Tufts University. (Retrieved on 2020-04-15). [Online]. Available: http://www.perseus.tufts.edu

[105] M. Rohlf, "Immanuel Kant," in *The Stanford Encyclopedia of Philosophy*, Spring 2020 ed., E. N. Zalta, Ed. Metaphysics Research Lab, Stanford University, 2020.

[106] C. Beyer, "Edmund Husserl," in *The Stanford Encyclopedia of Philosophy*, Summer 2018 ed., E. N. Zalta, Ed. Metaphysics Research Lab, Stanford University, 2018.

[107] G. Farina, "Some Reflections on the Phenomenological Method," *Dialogues in Philosophy, Mental and Neuro Sciences*, vol. 7, pp. 50–62, 2014.

[108] B. A. W. Russell, *A History of Western Philosophy.* Simon & Schuster, 1945.

[109] G. Hatfield, "René Descartes," in *The Stanford Encyclopedia of Philosophy*, Summer 2018 ed., E. N. Zalta, Ed. Metaphysics Research Lab, Stanford University, 2018.

[110] M. Wheeler, "Martin Heidegger," in *The Stanford Encyclopedia of Philosophy*, Winter 2018 ed., E. N. Zalta, Ed. Metaphysics Research Lab, Stanford University, 2018.

[111] M. Heidegger, *Sein und Zeit.* Max Niemeyer Verlag, 1967.

[112] ——, *Being And Time*, J. Macquarrie and E. Robinson, Eds. Blackwell, 1962, (Translated by J. Macquarrie and E. Robinson).

[113] H. Berglund, "Researching Entrepreneurship as Lived Experience," *Handbook of Qualitative Research Methods in Entrepreneurship*, January 2007.

[114] C. J. Steiner, "The Technicity Paradigm and Scientism in Qualitative Research," *The Qualitative Report*, vol. 7, 2002.

[115] M. Merleau-Ponty, *Phenomenology of Perception*, ser. Routledge Classics. Routledge, 2002.

[116] T. Toadvine, "Maurice Merleau-Ponty," in *The Stanford Encyclopedia of Philosophy*, Spring 2019 ed., E. N. Zalta, Ed. Metaphysics Research Lab, Stanford University, 2019.

[117] R. Elberfeld, "Heidegger und das ostasiatische Denken – Annäherungen zwischen fremden Welten," in *Heidegger-Handbuch: Leben – Werk – Wirkung*, 2nd ed., D. Thomä, Ed. J.B. Metzler, 2013, pp. 486–490.

[118] M. Heidegger, *A Dialogue on Language between a Japanese and an Inquirer.* Harper and Row, 1971.

[119] R. May, *Ex oriente lux: Heideggers Werk unter ostasiatischem Einfluss.* F. Steiner Verlag Wiesbaden, 1989.

[120] N. Ronkin, "Abhidharma," in *The Stanford Encyclopedia of Philosophy*, Summer 2018 ed., E. N. Zalta, Ed. Metaphysics Research Lab, Stanford University, 2018.

[121] Asaṅga, *Abhidharmasamuccaya – The Compendium of the Higher Teaching (Philosophy) by Asaṅga*, S. Boin-Webb, Ed. Asian Humanities Press, 2001, (Translated by S. Boin-Webb).

[122] Red Pine, *The Heart Sutra – The Womb of Buddhas.* Shoemaker & Hoard, 2004.

[123] B. Connelly, *Inside Vasubandhu's Yogācāra: A Practitioner's Guide.* Wisdom Publications, 2016.

[124] D. J. Kalupahana, *The Principles of Buddhist Psychology*, ser. Buddhist Studies. State University of New York Press, 1987.

[125] P. Harvey, *An Introduction to Buddhism: Teachings, History and Practices*, ser. Introduction to Religion. Cambridge University Press, 2013.

[126] A. Berzin. Primary Minds and the 51 Mental Factors. (Retrieved on 2020-04-05). [Online]. Available: http://http://studybuddhism.com/en/advanced-studies/science-of-mind/mind-mental-factors/primary-minds-and-the-51-mental-factors

[127] S. M. Emmanuel, *A Companion to Buddhist Philosophy*, ser. Blackwell Companions to Philosophy. Wiley-Blackwell, 2013.

[128] D. T. Suzuki, *The Laṇkāvatāra Sutra – A Mahayana Text*, ser. Eastern Buddhist Library. George Routledge and Sons, Ltd., 1932.

[129] R. Gethin, *The Foundations of Buddhism*, ser. Oxford University Press. Oxford University Press, 1998.

[130] W. S. Waldron, *The Buddhist Unconscious – The ālāya-vijñāna in the context of Indian Buddhist thought.* Routledge Curzon, 2003.

[131] C. Coseru, "Mind in Indian Buddhist Philosophy," in *The Stanford Encyclopedia of Philosophy*, Spring 2017 ed., E. N. Zalta, Ed. Metaphysics Research Lab, Stanford University, 2017.

Appendix A

Author's Publications

A.1 List of Author's Publications

R. Ritt, P. O'Leary, C. J. Rothschedl, and M. Harker, "Advanced Symbolic Time Series Analysis in Cyber Physical Systems," *Proceedings of the International Work-Conference on Time Series Analysis, ITISE 2017*, vol. 1, pp. 155–160, September 2017.

C. J. Rothschedl, R. Ritt, P. O'Leary, M. Harker, M. Habacher, and M. Brandner, "Extended Abstract: Real-Time Data Analytics in Raw Materials Handling," *Real-Time Mining Conference*, October 2017.

S. F. Nussdorfer, R. Ritt, C. J. Rothschedl, and P. O'Leary, "Condition Monitoring of Hydraulics in Heavy Plant and Machinery," *Automining 2018*, June 2018.

R. Ritt, P. O'Leary, C. J. Rothschedl, A. Almasri, and M. Harker, "Hierarchical Decomposition and Approximation of Sensor Data," *First International Conference on Numerical Modelling in Engineering – NME 2018*, pp. 351–370, January 2019.

C. J. Rothschedl, P. O'Leary, and R. Ritt, "Mimicking the Mechanisms of Language for the Unsupervised Detection of Hierarchical Structure in Time Series," *Proceedings of the 6th International Conference on Time Series and Forecasting, ITISE 2019*, vol. 1, pp. 155–166, September 2019.

A.2 Advanced Symbolic Time Series Analysis in Cyber Physical Systems

Originally appeared as:

R. Ritt, P. O'Leary, C. J. Rothschedl, and M. Harker, "Advanced Symbolic Time Series Analysis in Cyber Physical Systems," *Proceedings of the International Work-Conference on Time Series Analysis, ITISE 2017*, vol. 1, pp. 155–160, September 2017.

Advanced Symbolic Time Series Analysis in Cyber Physical Systems

Roland Ritt, Paul O'Leary, Christopher Josef Rothschedl, and Matthew Harker

Institute for Automation, University of Leoben, Leoben, Austria
roland.ritt@unileoben.ac.at

Keywords: symbolic time series analysis, single channel lexical analyser, time series, cyber physical system, linear differential operator

This paper presents advanced symbolic time series analysis (ASTSA) for large data sets emanating from cyber physical systems (CPS). The definition of CPS most pertinent to this paper is: *A CPS is a system with a coupling of the cyber aspects of computing and communications with the physical aspects of dynamics and engineering that must abide by the laws of physics. This includes sensor networks, real-time and hybrid systems* [1]. To ensure that the computation results conform to the laws of physics a linear differential operator (LDO) is embedded in the processing channel for each sensor. In this manner the dynamics of the system can be incorporated prior to performing symbolic analysis. A non-linear quantization is used for the intervals corresponding to the symbols. The intervals are based on observed modes of the system, which can be determined either during an exploratory phase or on–line during operation of the system. A complete processing channel (see Fig. 2) is called a single channel lexical analyser; one is made available for each sensor on the machine being observed.

The implementation of LDO in the system is particularly important since it enables the establishment of a causal link between the observations of the dynamic system and their cause. Without causality there can be no semantics and without semantics no knowledge acquisition based on the physical background of the system being observed. Correlation alone is not a guarantee for causality[1]

This work was originally motivated from the observation of large bulk material handling systems, see Fig. 1 for three examples of such systems. Typically, there are $n = 150 \ldots 250$ sensors per machine, and data is collected in a multi rate manner; whereby general sensors are sampled with $f_s = 1\,Hz$ and vibration data being sampled in the kilo-hertz range.

1 Local Linear Differential Operators (LDO)

Although processing the entire 'large' time series is a common practice in exploratory data analysis, reliable local computations (implemented as streaming

[1] Consider an exothermic system with a high activation energy. We must include the exothermic model if we are to establish causality, correlation alone will lead to erroneous interpretation.

Fig. 1. Examples of machines to which the analysis is applied. Image courtesy: Sandvik

algorithms) are preferred in on–line data processing. Since in this work we deal with time series emanating from cyber physical systems new techniques for local computations including the physics of the system (described by differential equations) have to be developed.

An ordinary differential equation (ODE) of the form

$$a_d\left(t\right) y^{(d)}\left(t\right) + a_{d-1}\left(t\right) y^{(d-1)}\left(t\right) + \ldots + a_0\left(t\right) y^{(0)}\left(t\right) = g\left(t\right) \tag{1}$$

can be described using a linear differential operator (LDO) D [2] such that $\mathsf{D}^{(i)} y\left(t\right) = y^{(i)}\left(t\right)$ where y is a function of t, $y^{(i)}$ is the n-th derivative with respect to t and $g\left(t\right)$ is the exciting function, in our case the noisy sensor data. This yields to the notation [3]

$$a_d\left(t\right) \mathsf{D}^{(d)} y^{(d)}\left(t\right) + a_{d-1}\left(t\right) \mathsf{D}^{(d-1)} y^{(d-1)}\left(t\right) + \ldots + a_0\left(t\right) \mathsf{D}^{(0)} y^{(0)}\left(t\right) = g\left(t\right). \tag{2}$$

Factoring $y\left(t\right)$ leads to the compact formulation of the model

$$\mathsf{L} y\left(t\right) = g\left(t\right), \tag{3}$$

with

$$\mathsf{L} \triangleq a_d\left(t\right) \mathsf{D}^{(d)} + a_{d-1}\left(t\right) \mathsf{D}^{(d-1)} + \ldots + a_0\left(t\right) \mathsf{D}^{(0)}. \tag{4}$$

In the discrete case (3) can be formulated as matrix equation. Solving this equation for y is an inverse problem which can be solved numerically in a discrete sense by

$$\boldsymbol{y} = \mathsf{L}^{+} \boldsymbol{g} + \mathsf{N}_{\mathsf{L}} \boldsymbol{\alpha}, \tag{5}$$

where $\boldsymbol{y}$ is the solution to the inverse problem, L^{+} is the pseudo-inverse of L, N_{L} is an orthonormal basis function set of the null space of L , $\boldsymbol{\alpha}$ is a coefficient vector for the null space (computed by initial- and/or the boundary-values) and $\boldsymbol{g}$ is the noisy time series data vector. Algebraic implementations for the solution of such problems can be found in [4–7].

The LDO, and their inverses, can be implemented as local operators and efficiently computed using a convolutional approach. This is basically a streaming-algorithm and thus suitable for big-data processing.

Furthermore, the covariance of the solution (5) is simply propagated as

$$\mathsf{\Lambda}_y = \mathsf{L}^{+} \mathsf{\Lambda}_g \left(\mathsf{L}^{+}\right)^{T}. \tag{6}$$

Using Λ_y as an estimate for the covariance in conjunction with the student-t and/or F-distribution permits the estimation of a confidence interval over the complete solution and allows the computation of a prediction interval for future values.

That is, the approach presented here to implementing linear differential operators not only permits the solution of embedded system dynamics but also yields a confidence interval for the predicted values of the dynamics.

2 Symbolic Time Series Analysis

The availability of the sensor signals, their regularized derivative and/or the application of a LDO permits the implementation of an advanced symbolic time series analysis (ASTSA) which includes the modelling of the system dynamics. As a result the time series (TS) can be discretized and compressed using unique symbols for different intervals (the so called alphabet). This step is named lexical analysis. A number of methods for the selection of the symbol intervals based on, e.g. , equal probability, variance or entropy can be found in literature [8–12]. Here, in a new approach, we define the intervals to correspond to the modes of the dynamic system in operation, i.e. each symbol corresponds to a mode or portion of a mode which should be identified. Commonly controllers are designed to operate optimally in a number of specific but distinct modes of the dynamic system.

In a next step, connected sequences with the same symbol can be compressed to a single symbol predicated with its length. The combination of applying a LDO, lexical analysis of the derived signal and compression is called single channel lexical analyser (SCLA), see Fig. 2. Combining the output of multiple SCLA

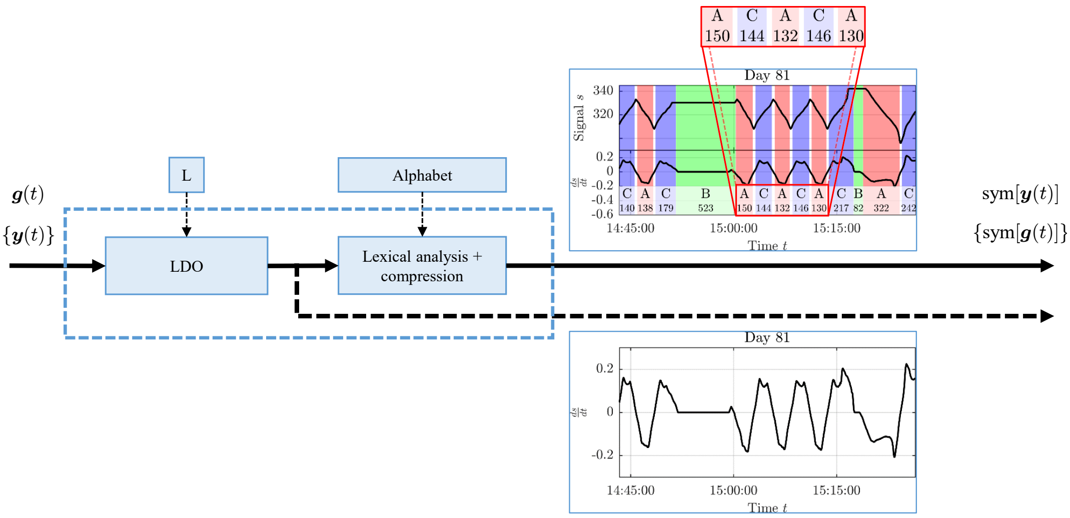


Fig. 2. A single channel lexical analyser (SCLA)

is called multi channel lexical analyser (MCLA). Two examples of symbolic time series analysis using MCLA are demonstrated in Fig. 3 and Fig. 4). For signal 1 and signal 2 the alphabet consists of the three symbols [u, s, d] assigned to the direction of the signal (up, stationary, down). The figures show two operation modes from the same machine. It can be clearly seen, that the operation modes of the machine have a different symbolic representation (visualized as different shaded colours in the plots) and allow a fast intuitive inspection and characterization of the signal. The signal range from the first dashed-blue line to the

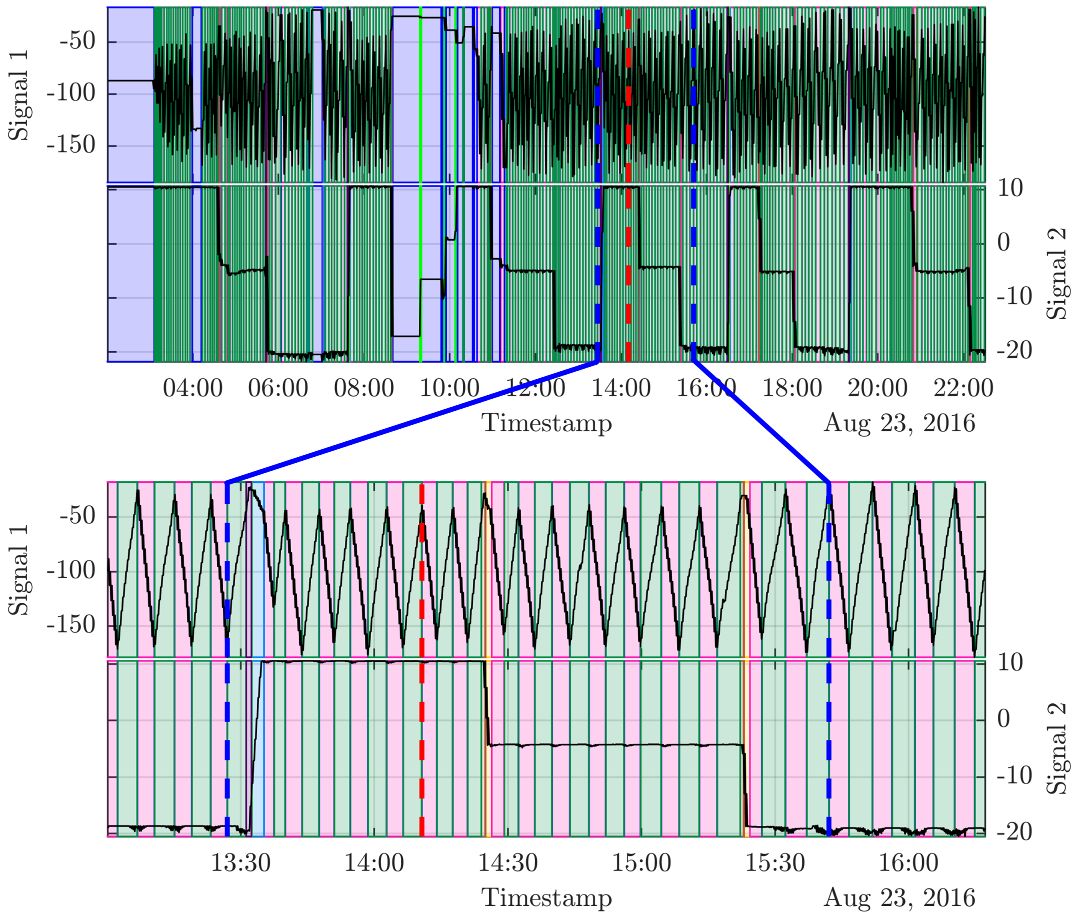


Fig. 3. Operation mode 1; the coloured areas illustrate the output of the MCLA; different colours represent different combinations of symbols from the SCLA of each channel (in this case two channels); the alphabet used for signal 1 and 2 consists of the three symbols [u, s, d]. Top: machine working in operation mode 1 with longer interrupts in-between (light blue area - both signals are stationary); Bottom: snippet of the signal showing the typical repeating pattern of operation mode 1.

dashed-red line (marked in both plots) have the same symbolic representation in both modes, whereas the portion of the signal after the dashed-red line shows a different colour-code for each mode.

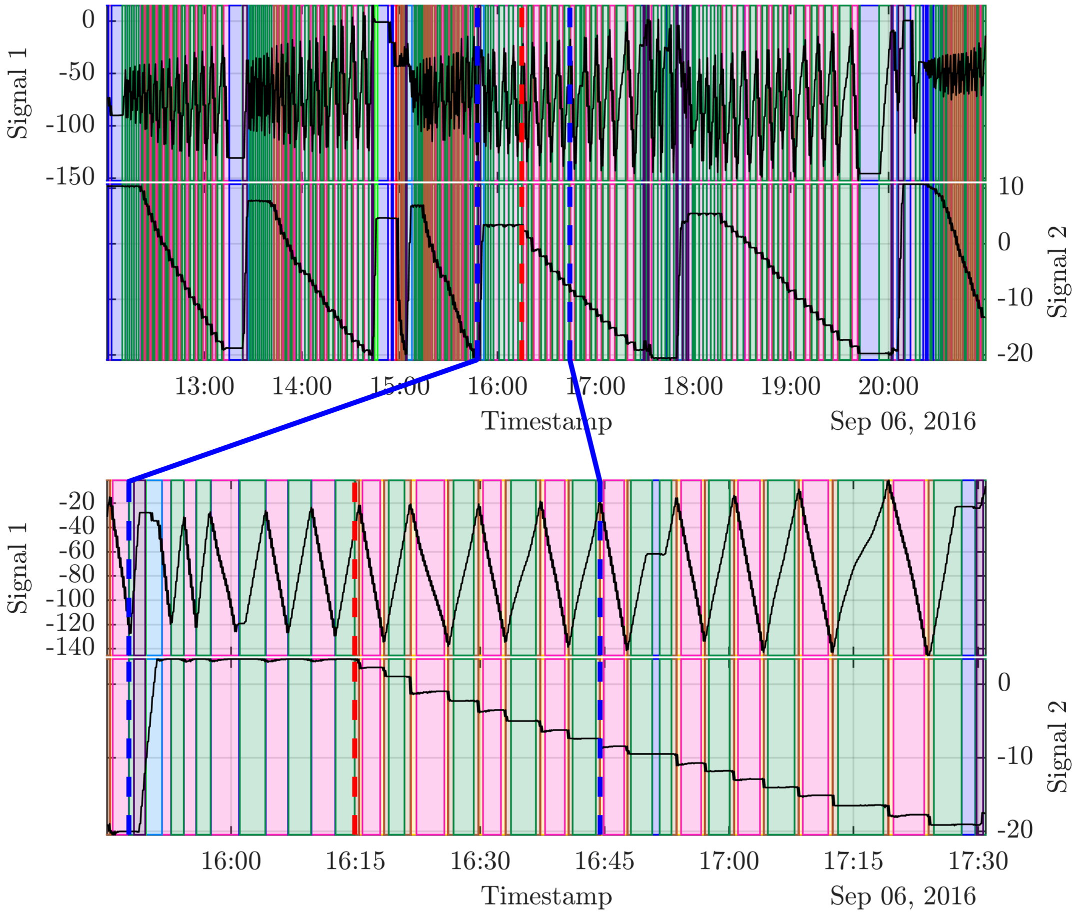
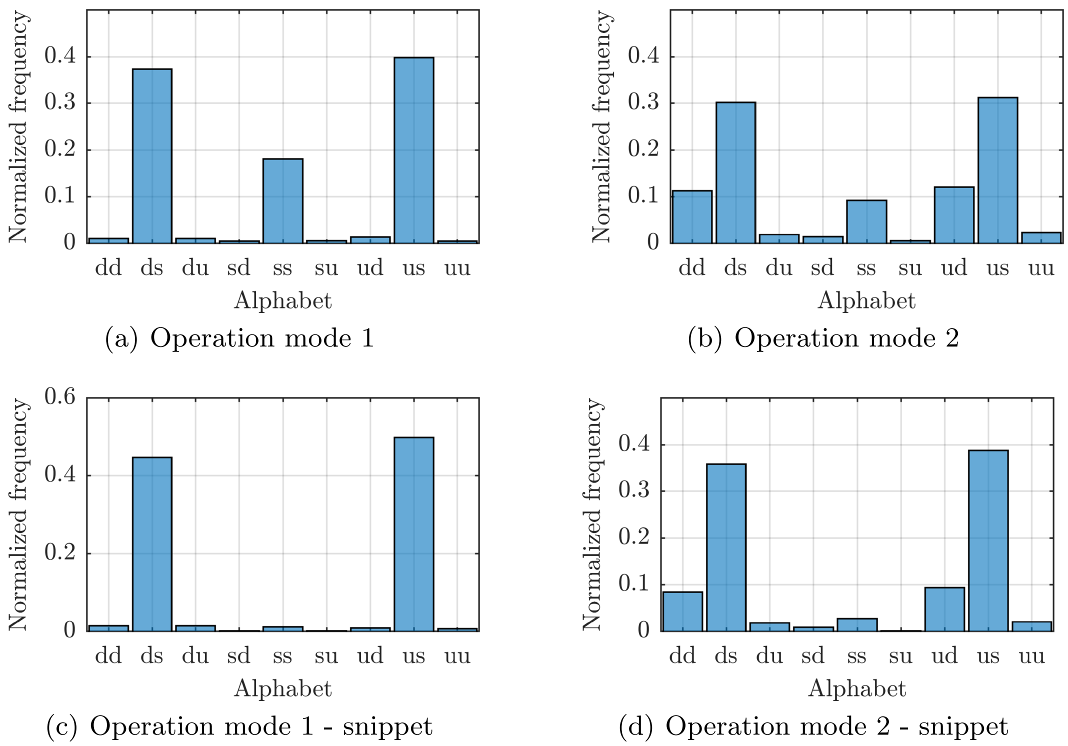


Fig. 4. Operation mode 2; the coloured areas illustrate the output of the MCLA; different colours represent different combinations of symbols from the SCLA of each channel (in this case two channels); the alphabet used for signal 1 and 2 consists of the three symbols [`u`, `s`, `d`]. Top: machine working in operation mode 2 with interrupts in-between (light blue area - both signals are stationary); Bottom: snippet of the signal showing the typical repeating pattern of operation mode 2.

The generated symbolic representation is used for further analyses. Building up histograms for occurring symbol combinations offers an insight in the overall behaviour of the system, see Fig. 5). This allows inter-machine comparison and comparison of different signal portions/ranges as well as classification of the operation mode. On top of Fig. 5 the histograms of the entire signal ranges shown in Fig 3 (top) and Fig 4 (top) are presented. The histograms for the typical repeating snippets, shown in Fig 3 (bottom) and Fig 4 (bottom), are visualized on the bottom. Since the machine is interrupted several times in both operating modes, the bins for the stationary state (`ss`) are more visible for the entire signal sequences (top). Excluding these bins, the statistics (histograms) of the shown snippets can act as representatives (motifs) for the operating modes. It can be seen that the histograms differ whether the machine is operating in mode 1 (left) or mode 2 (right). Especially the occurrences of `dd` and `ud` reveal the differences. In future investigations the definition of a similarity measure for such histograms is planned to compare them qualitatively and may use this for automatic operation recognition and finding motifs. Note: sorting the histograms in decreasing order of occurrences will yield a classical frequency dictionary.



(a) Operation mode 1

(b) Operation mode 2

(c) Operation mode 1 - snippet

(d) Operation mode 2 - snippet

Fig. 5. Histograms of occurring symbol combinations of a machine in two different operation modes. Top: Histograms for the entire time range shown in Fig 3 (top) and Fig 4 (top); Bottom: Histograms for the signal snippets presented in Fig 3 (bottom) and Fig 4 (bottom).

A big advantage of the presented symbolic time series analysis is, that he sequence of symbols - either single or multi channel - can now be addressed with techniques more common to computational linguistics (e.g. *regex*) [13], which is a growing field of research.

3 Conclusion

Successful data analytics in large physical systems must embed the modelling of the individual component and complete system dynamics. This has been addressed by providing for a linear differential operator or its inverse in each and every signal- or derived-data-channel. A multi-variate symbolic time series analysis has been introduced. It permits a symbolic view of the system and its dynamics. The concept of frequency dictionaries has been applied to automatic operation recognition; this functions for operation types which are characterised by a specific distribution of symbols. A major advantage of the proposed method is its intrinsic multi-scale property. This enables the identification of very short events in very large data sets. Currently, we are performing research on the relationships between the sequences of symbols and the metaphor of language. Initial results indicate that this opens the door to take advantage of new methods emerging in computational linguistics.

References

1. Baheti, R., Gill, H.: Cyber-physical systems. The Impact of Control Technology (2011) 161–166
2. Lanczos, C.: Linear differential operators. SIAM (1961)
3. O'Leary, P., Harker, M., Gugg, C.: An inverse problem approach to approximating sensor data in cyber physical systems. In: 2015 IEEE International Instrumentation and Measurement Technology Conference (I2MTC) Proceedings. Volume 2015-July., IEEE (may 2015) 1717–1722
4. Gugg, C., Harker, M., O'Leary, P., Rath, G.: An Algebraic Framework for the Real-Time Solution of Inverse Problems on Embedded Systems. In: 2015 IEEE 17th International Conference on High Performance Computing and Communications, 2015 IEEE 7th International Symposium on Cyberspace Safety and Security, and 2015 IEEE 12th International Conference on Embedded Software and Systems. Volume V., IEEE (aug 2015) 1097–1102
5. Harker, M., O'Leary, P.: Discrete Orthogonal Polynomial Toolbox - Matlab File Exchange
6. Gugg, C.: An Algebraic Framework for the Solution of Inverse Problems in Cyber-Physical Systems. Phd thesis, Montanuniversitaet Leoben (2015)
7. O'Leary, P., Harker, M.: An algebraic framework for discrete basis functions in computer vision. In: Proceedings - 6th Indian Conference on Computer Vision, Graphics and Image Processing, ICVGIP 2008, IEEE (dec 2008) 150–157
8. Lin, J., Keogh, E., Wei, L., Lonardi, S.: Experiencing SAX: a novel symbolic representation of time series. Data Mining and Knowledge Discovery **15**(2) (aug 2007) 107–144
9. Veenman, C., Reinders, M., Backer, E.: A maximum variance cluster algorithm. IEEE Transactions on Pattern Analysis and Machine Intelligence **24**(9) (sep 2002) 1273–1280
10. Chau, T., Wong, A.: Pattern discovery by residual analysis and recursive partitioning. IEEE Transactions on Knowledge and Data Engineering **11**(6) (1999) 833–852
11. Keogh, E., Lonardi, S., Chiu, B.Y.c.: Finding surprising patterns in a time series database in linear time and space. In: Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining - KDD '02, New York, New York, USA, ACM Press (2002) 550
12. Daw, C.S., Finney, C.E.A., Tracy, E.R.: A review of symbolic analysis of experimental data. Review of Scientific Instruments **74**(2) (feb 2003) 915–930
13. Clark, A., Fox, C., Lappin, S.: The Handbook of Computational Linguistics and Natural Language Processing. Volume XXXIII. Wiley-Blackwell (2010)

A.3 Real-Time-Data Analytics in Raw Materials Handling

Originally appeared as:

C. J. Rothschedl, R. Ritt, P. O'Leary, M. Harker, M. Habacher, and M. Brandner, "Extended Abstract: Real-Time Data Analytics in Raw Materials Handling," *Real-Time Mining Conference*, October 2017.

Real-Time-Data Analytics in Raw Materials Handling

Christopher Rothschedl, Roland Ritt, Paul O'Leary, Matthew Harker, Michael Habacher and Michael Brandner

Chair of Automation, University of Leoben, Leoben, Austria.
(e-mail: roland.ritt@unileoben.ac.at).

1 Introduction

This paper proposes a system for the ingestion and analysis of real-time sensor and actor data of bulk materials handling plants and machinery. It references issues that concern mining sensor data in cyber physical systems (CPS) as addressed in O'Leary et al. [2015].

The advance of cyber physical systems has created a significant change in the architecture of sensor and actor data. It affects the complexity of the observed systems in general, the number of signals being processed, the spatial distribution of the signal sources on a machine or plant and the global availability of the data. There are different definitions for what constitutes cyber physical systems Baheti and Gill [2011], Geisberger and Broy [2012], IOSB [2013], Lee [2008], NIST [2012], Park et al. [2012], Spath et al. [2013a,b], Tabuada [2006]: the most succinct and pertinent to the work shown in this paper is the definition given by the IEEE Baheti and Gill [2011] and ACM[1]:

> A CPS is a system with a coupling of the cyber aspects of computing and communications with the physical aspects of dynamics and engineering that *must abide by the laws of physics*. This includes sensor networks, real-time and hybrid systems.

Results computed from sensor and actor data **must** obey the equations used for modelling the physics of the observed system — this fundamentally poses an *inverse problem*. Such problems are not covered sufficiently by literature addressing *mining of sensor data*, see for example Esling and Agon [2012], Fuchs et al. [2010], Keogh and Kasetty [2003], Last et al. [2004]. Even available standard books, such as Aggarwal [2013] on mining sensor data, do not discuss the special nature of sensor data. Typically, present approaches of mining data rely on correlation as being a sole, reliable measure for significance. It is not taken into account that the inverse solutions to the model-describing equations are required to establish a semantic link between a sensor observation and its precedent cause. Without this link — without *causality* — there can be no physics based knowledge discovery.

[1] ACM/IEEE International Conference on Cyber-Physical Systems (ICCPS) (iccps.acm.org)

The underlying data analytics problem can be described generally by the following statements:

1. The momentum of what is called Industry 4.0 promotes an increasing amount and availability of data. A suitable data ingestion system becomes necessary to acquire real-time sensor and actor data on a global scale. The fundamental concept on how to acquire, transport, ingest, and provide data needs to be sufficiently secure and adaptable enough to accommodate data of mining machines that may be located in remote areas.
2. Mathematical tasks are required to apply data analytics to industrial data sets, such as the solution of inverse problems and optimal-control-type problems.
3. Complex systems are modelled mathematically by following principles gained from modelling simple engineering systems, e.g., a vibrating string or a vibrating beam. These can be modelled using differential equations, ordinary and partial. More sophisticated mathematical models will be required to conquer the expanding complexity of modern mechatronic systems.
4. Data analytics will determine the particular causes to specific behaviour witnessed by sensor and actor data. Inverse problems are fundamental to accomplish such tasks. Additional metadata is required to accurately interpret the results of inverse models, as inverse problems do not have unique solutions per definition.
5. Extracting knowledge from data lies beyond simple information extraction. A more profound view on the philosophy of science points towards the necessity of assigning semantic information to data channels to establish such investigations. The metaphorical parallels between machine behaviour and natural language provide a form of knowledge extraction. It can be shown that machines have their own specific polysyllabic language. Once identified, it can be efficiently queried for symbolic patterns of normal or anomalous behaviour.

2 System Premiss

As an extension of Ackoff's work (Ackoff [1989]), Embrechts (Embrechts et al. [2005]) proposes the data mining pyramid consisting of the terms *data*, *information*, *knowledge*, *understanding* and *wisdom*. Embrechts does not provide any definitions for these terms, Ackoff offers intuitive but rather nebulous definitions; both do not provide a scientific basis for mining sensor data. Based on the integral idea we propose the fundamental concept behind the data analytics in Fig. 1.

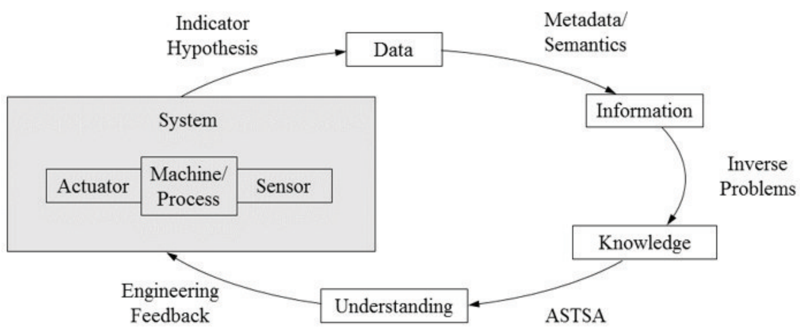


Fig. 1: The process behind the data analysis system.

The presented hierarchy illustrates how the questions of processing large data sets can be approached in a coherent and structured manner. The fundamental relationships of this premiss are:

1. A suitable indicator hypothesis builds the basis for the collection of data. If a specific sensor is chosen, an implicit indicator hypothesis has been selected as well, i.e., a temperature sensor defines that temperature is of relevance for the task.
2. Once acquired, data is only present as a simple stream of numbers; *metadata* adds meaning to the data. Beyond that, *context* is required to establish *significance*: a temperature value can have entirely contrasting significances for measurements of two different sources.
3. System models and the solution of the corresponding inverse problems are required to establish a causal link between measurement data and its possible cause. In general, there are no unique solutions to inverse problems.
4. Hence, a-priori knowledge is necessary to find the desired solution. These results of the inverse problems (the causes) constitute *knowledge*.
5. Effects of *human-machine interaction* must be considered to gain *understanding* of the whole system behaviour. Our approach, *Advanced Symbolic Time Series Analysis (ASTSA)*, is based on the emergence of language as it is modelled by the philosophy of phenomenology. The basic principle consists of symbols that are assigned to actions — *verbs*. The symbols for states are *nouns*. *Adverbs* and *adjectives* are used to predicate the verbs and nouns. *Punctuation* represents different lengths of pauses. Following such a segmentation, the time series is automatically converted into a sequence of symbols, enabling symbolic querying.
6. The whole process serves the understanding of what was originally only a stream of numbers. *Engineering feedback* can be derived from understanding the system response behaviour to certain loads and circumstances. Existing systems can be optimised and future revisions benefit from this as well.

3 Data Ingestion

A versatile data handling system is necessary to conquer large sets of time series data in a structured and efficient manner. Before such a system is able to provide any data, it has to ingest data following a specific workflow. In the course of the ingestion process, data is collected, quality-checked, and merged with corresponding metadata before it is prepared to fit a consistent data model. Sensor values are handled in the same way as derived measurements, i.e., the force of a hydraulic cylinder calculated from its dimensions (metadata) and its pressure values (time series from sensors).

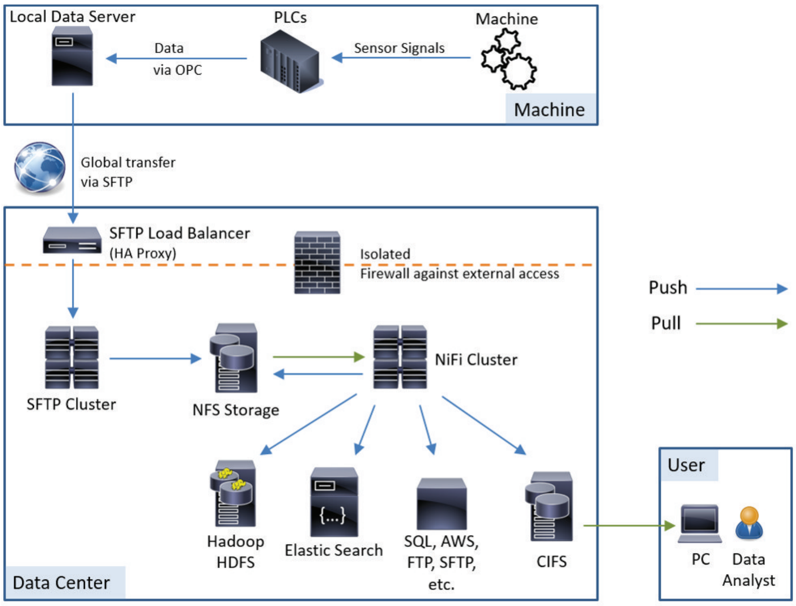


Fig. 2: This illustration shows the main processes of data ingestion. The top section corresponds to the machine or plant on which data is being collected, while the bottom part represents the data center located at a different location. The data is provided in several formats after it has been ingested.

The concept describing the data ingestion process is illustrated in Fig. 2. Data of a machine's sensors is collected from its main programmable logic controller (PLC) and stored on a local data server before it is transported via a secured connection to the data center. After passing quality control, the data is stored permanently according to the data model and specified data manipulation workflows can be triggered on the cluster. Ultimately, the data is made available to consumers (data analysts, report recipients, domain experts, etc.) in different formats: this ensures that all users are independent in their choice of working environment.

The data is stored as a contiguous data stream as a result of the data ingestion process, see Fig. 3. The data input can be split, e.g., as daily exports of a buffering database running on the local data server at the machine's location. The data of all packets are merged to a contiguous, multi-channel stream of time series. When a user requests data from the system, they have the experience of querying the machine directly and in real-time. This opens the door to evaluations spanning time ranges varying from days to months and years. Furthermore, time ranges fitting a machine's operation characteristics can be queried, such as the time for loading a vessel in the case of analysing a ship loader. This permits a complete differentiation between input and output segmentation.

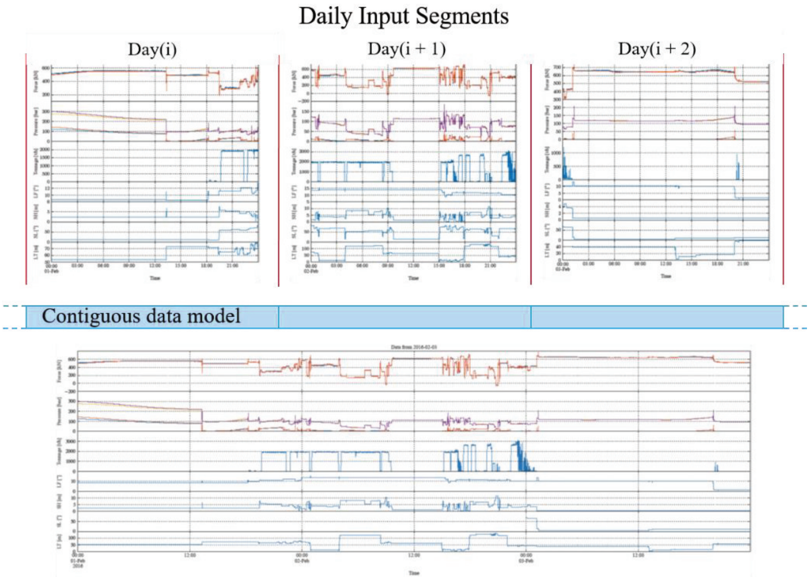


Fig. 3: Three single days of data are assembled to a contiguous data stream. The illustrated contiguous section corresponds to the time portion a ship loader needs to load a vessel: this enables evaluations based on time ranges that are significant to particular fields of interest.

4 Systems Currently Being Monitored

Four mining machines that are currently being monitored using the approach presented in this paper are shown in Fig. 4. Data of these systems is collected constantly with a sampling interval of 1s. Typically, 50 to 850 sensor signals are collected, depending on the complexity of the monitored system.

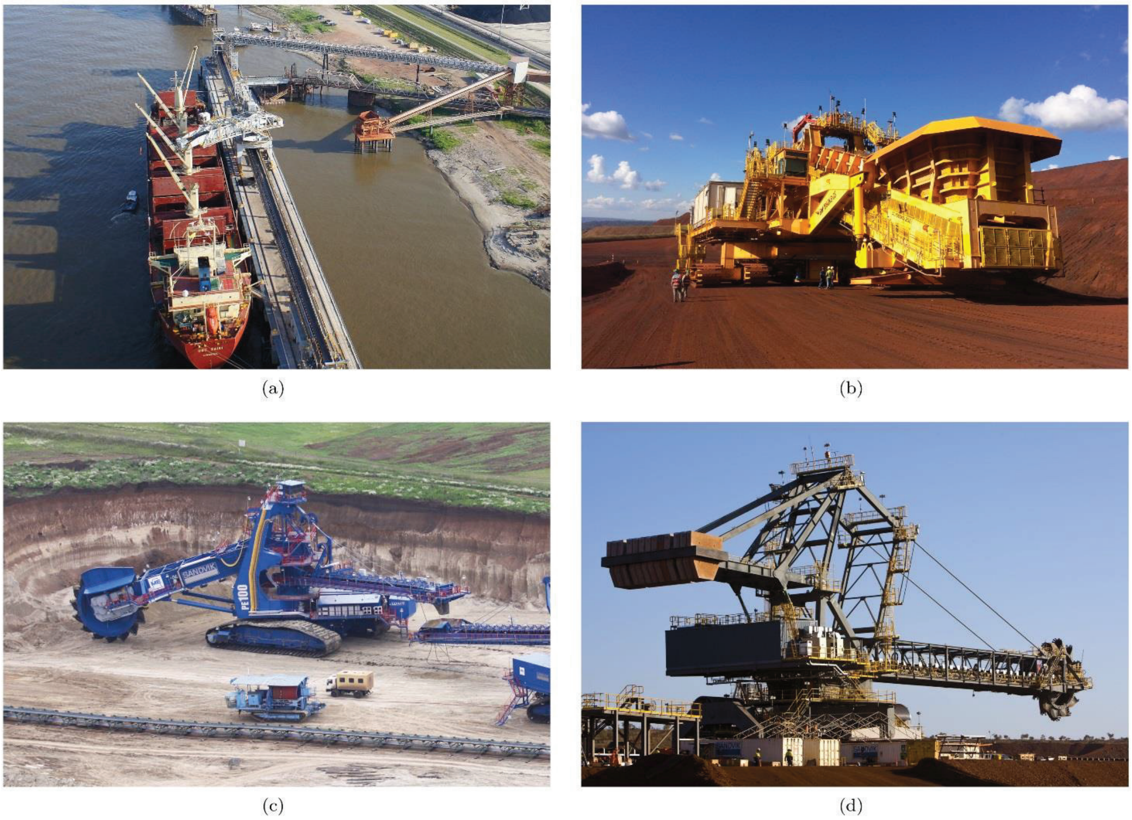


Fig. 4: Examples of four systems that are currently being monitored using the described approach: a) ship loader, b) mobile sizing rig, c) bucket-wheel excavator, d) bucket-wheel reclaimer. The sensor channels of these systems are monitored with a sampling interval of 1s. (Sources: (a) – http://www.flickriver.com/photos/ impalaterminals_images/17557941415/, retrieved on 2016-02-08; (b), (c), (d) – Courtesy of Sandvik.)

5 Exemplary Data Evaluations

The collection and analysis of data can be used for many different aspects of evaluating a machine during its life-cycle:

Condition Monitoring: Undoubtedly, data analytics can be used to address questions regarding condition monitoring or preventative maintenance, see Rothschedl [2016]. However, in this work we focus on issues that have received less attention in literature, e.g., incident analysis.

Commissioning: If data is already collected during the commissioning phase of a machine, analysing it can support shortening the time needed for this phase. Controlled tests can be verified with manageable effort and unexpected response behaviour to specific load scenarios can be detected. On several occasions, it was possible to identify sensors that delivered erroneous values for only a few samples a day. Judging from the nature of such error patterns, it would not be possible for a commissioning engineer to spot these defective sensors without such a system.

Fleet management: Insights gained from analysing one machine can support understanding the behaviour of other machines of similar design. For example, two identical bucket-wheel excavators were monitored which are operated in the same mine, handling the same type of material. The characteristics of both machines matched in many aspects. In contrast, two similar ship loaders exhibited behaviour that was significantly different. This raises the question whether these machines fulfil the conditions required to be ergodic systems.

Automatic Operations Recognition: With ASTSA, several data channels can be combined to define machine states. Sequences of these states refer to corresponding operation modes which can be used to characterise how a machine is being controlled. These sequences support the identification of inappropriate operations that may lead to damages or to missing performance goals.

Incident Analysis: Incidents with equipment in mining environments bear serious financial and legal issues. Unplanned maintenance and repair work in such environments and locations quickly reach immense financial dimensions, also because associated materials handling processes are interrupted, provoking serious follow-up costs. Liability for injury and damages are the main concerns from the legal point of view. The analysis of real-time operational data prior to incidents supports the determination of the possible causes for their occurrences and, hence, can provide more certainty to the financial and legal claims. Although this form of analysis can shed light on the clarification of far-reaching issues, this topic has been rarely mentioned in literature. It is evident that incident analysis plays a major role when working with mining machines.

Logistics Optimisation: The analysis of long-term time series allows evaluations based on aggregated data: the distribution of conveyed material over the full slewing range of a machine over a long period of time can support identifying unevenly distributed component utilisation. Such problems can often be avoided or mitigated if the logistics of a machine are adapted.

Two exemplary evaluations are presented:

5.1 Incident Analysis

The figures below (Fig. 5 and Fig. 6) show the results of performing incident analysis for a bucket-wheel excavator. The analysis shows a large number of events distributed over time and conspicuous times during which no events occurred: this is with most certainty operator-dependent behaviour of the system as a whole.

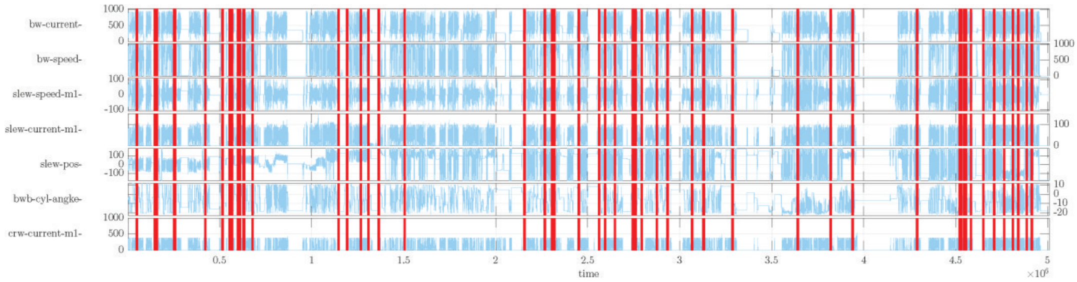


Fig. 5: This example of incident analysis shows data for a time period of two months, acquired with a sampling time of 1s. Each vertical line corresponds to an event; 63 events were found in total by using Advanced Symbolic Time Series Analysis (ASTSA). Every event corresponds to an inappropriate operation of the machine: the data can be zoomed in on automatically for every single event to perform local analysis, i.e., in the seconds and minutes right before the occurrence of the event (see Fig. 6).

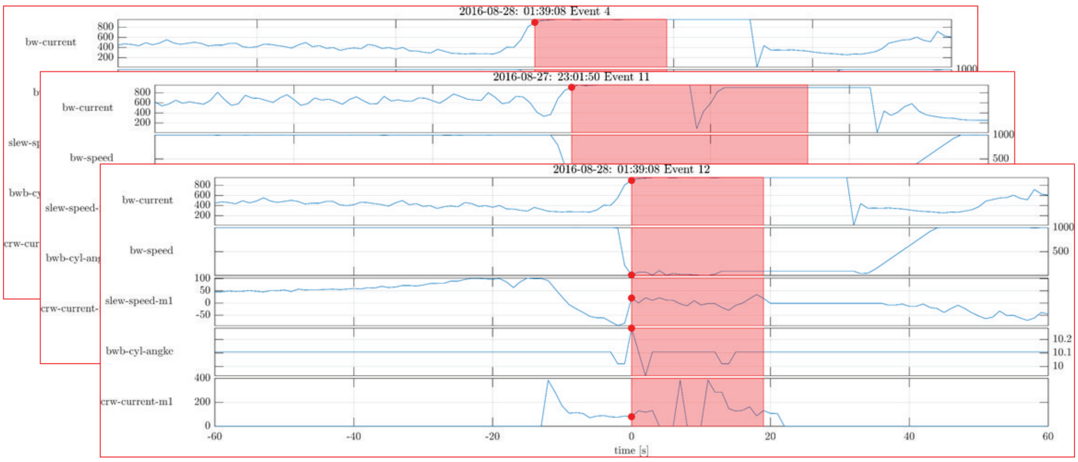


Fig. 6: Plots of the identified events with 1s resolution for three of the 63 events reported in Fig. 5.

5.2 Long-Term Logistics Optimisation

The data shown in Fig. 7 is the polar histogram of loading on the slew bearing of a bucket-wheel reclaimer. The data has been aggregated with t_s=1s over an observation period of one year. Interestingly, the overloading in one quadrant is not visible on a daily basis. The higher loading, evident from aggregated long-term data in the figure, has significant consequences on the life span of the bearing.

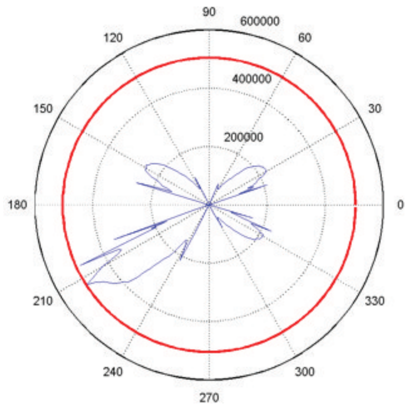


Fig. 7: Polar histogram of loading on the slew bearing of a bucket-wheel reclaimer. The data has been aggregated with a sampling time of 1s over an observation period of one year.

6 Conclusions

The collection of very large real-time data series from plant and machinery is highly relevant in a mining context. A strongly structured approach is required, if the best use is to be made of the data. The results are relevant for both, machine constructors and also their operators. It is significantly more than just preventative maintenance.

REFERENCES

Ackoff, R.L. (1989). From data to wisdom. Journal of Applied Systems Analysis, 16, 3–9.

Aggarwal, C.C. (ed.) (2013). Managing and Mining Sensor Data. Springer.

Baheti, R. and Gill, H. (2011). Cyber-physical systems. The Impact of Control Technology, 161–166.

Embrechts, M., Szymanski, B., and Sternickel, K. (2005). Computationally Intelligent Hybrid Systems: The Fusion of Soft Computing and Hard Computing. John Wiley and Sons, NewYork.

Esling, P. and Agon, C. (2012). Time-series data mining. ACM Comput. Surv., 45(1), 12:1–12:34. doi:10.1145/2379776.2379788. URL http://doi.acm.org/10.1145/2379776.2379788.

Fuchs, E., Gruber, T., Pree, H., and Sick, B. (2010). Temporal data mining using shape space representations of time series. Neurocomputing, 74(13), 379 – 393. doi:http://dx.doi.org/10.1016/j.neucom.2010.03.022. URL http://www.sciencedirect.com/science/article/pii/S0925231210002237. Artificial Brains.

Geisberger, E. and Broy, M. (2012). agendaCPS: Integrierte Forschungsagenda Cyber-Physical Systems, volume 1. Springer.

IOSB, F. (2013). Industry 4.0 information technology is the key element in the factory of the future. Press Information.

Keogh, E. and Kasetty, S. (2003). On the need for time series data mining benchmarks: A survey and empirical demonstration. Data Min. Knowl. Discov., 7(4), 349–371. doi:10.1023/A:1024988512476. URL http://dx.doi. org/10.1023/A:1024988512476.

Last, M., Kandel, A., and Bunke, H. (2004). Data Mining in Time Series Databases. Series in machine perception and artificial intelligence. World Scientific. URL http://books.google.at/books?id=f38wqKjyBm4C.

Lee, E.A. (2008). Cyber physical systems: Design challenges. In Object Oriented Real-Time Distributed Computing (ISORC), 2008 11th IEEE International Symposium on, 363–369. IEEE.

NIST (2012). Cyber-physical systems: Situation analysis of current trends, technologies, and challenges. Technical report, National Institute of Standards and Technology (NIST). URL www.nist.gov.

O'Leary, P., Harker, M., and Gugg, C. (2015). A position paper on: Sensor-data analytics in cyber physical systems, from Husserl to data mining. In SensorNets 2015, Le Cresout, France.

Park, K.J., Zheng, R., and Liu, X. (2012). Cyber-physical systems: Milestones and research challenges. Computer Communications, 36(1), 1–7.

Rothschedl, C.J. (2016). Condition Monitoring of Large-Scale Slew Bearings in Bucket-Wheel Boom-Type Reclaimers. Diploma Thesis, University of Leoben.

Spath, D., Gerlach, S., Hämmerle, M., Schlund, S., and Strölin, T. (2013a). Cyber-physical system for self-organised and flexible labour utilisation. Personnel, 50, 22.

Spath, D., Ganschar, O., Gerlach, S., Hämmerle, M., Krause, T., and Schlund, S. (2013b). Produktionsarbeit der Zukunft-Industrie 4.0. Fraunhofer IAO Stuttgart.

Tabuada, P. (2006). Cyber-physical systems: Position paper. In NSF Workshop on Cyber-Physical Systems.

A.4 Condition Monitoring of Hydraulics in Heavy Plant and Machinery

Originally appeared as:

S. F. Nussdorfer, R. Ritt, C. J. Rothschedl, and P. O'Leary, "Condition Monitoring of Hydraulics in Heavy Plant and Machinery," *Automining 2018*, June 2018.

Condition Monitoring of Hydraulics in Heavy Plant and Machinery

Stefan Franz Nussdorfer[1], Roland Ritt[1*], Christopher Josef Rothschedl[1] and Paul O'Leary[1]
1. *Chair of Automation, Department Product Engineering, University of Leoben, Austria*

ABSTRACT

This paper presents a new approach to remote condition monitoring of hydraulic actuators as used in mining equipment. The condition of the hydraulic system has a major influence on the life span of machine parts and the equipment's performance in general. The aim is to use real-time monitoring to extract a better understanding of the state of the machine, its behaviour, and how it is being operated. Specifically, the goal is to generate added value along the complete life cycle of the equipment.

Within an existing framework, all available sensor and actuator data is collected. The hydraulic analysis module uses the pressures measured on the rod side $p_{r,i}$ and on the piston side $p_{p,i}$ of each hydraulic cylinder, with i indicating the i-th cylinder. Additionally, the values are combined with the metadata of the rod and piston areas $a_{r,i}$ and $a_{p,i}$, yielding the operating force $f_{o,i}$. The exemplary system consists of two hydraulic cylinders working in parallel. By analysing the sum f_s and difference f_d forces, the total lifting force and torsion exerted on the machine's boom are determined. For each of the variables $p_{r,i}$, $p_{p,i}$ and resulting forces, time series data is available for a period of 93 days, with the data being collected at sampling intervals of 1 second. Consequently, there are 8.035.200 samples per signal. The large number of samples ensures a well-defined confidence interval in the statistical evaluation of the data.

The time-varying histograms for 24 hour intervals make both short and long term changes visible. Whereby the large number of samples enables a reliable separation of systematic and stochastic components in the signal.

Operational analysis is presented for a bucket-wheel excavator which demonstrates the monitoring capabilities and specific results. The monitoring system has enabled the automatic detection of various defects in the observed hydraulics, e.g., a defective sensor was identified, as was the systematic occurrence of negative pressures.

INTRODUCTION

The condition of remotely located heavy plant and machinery, especially mining equipment, is vital to the processes they are part of. This paper focuses on the monitoring of the hydraulic systems of such machines, which are needed to carry out the luffing movements of a machine's boom. The condition of these hydraulic systems has a major influence on the life span of the machine and, hence, has an important influence on the materials handling process. An unplanned shutdown can be the consequence of an unexpected failure of a crucial machine component: subsequent processes need to be stopped for the duration of the time-consuming maintenance works, which can quickly accumulate to high costs (Rothschedl, 2016).

The presented tool is part of a data analytics framework that supports the decision-making processes of mining and machine experts (domain experts): using their input, improved estimations of the component condition can be given. To accomplish the tasks necessary for this, a flexible and sufficiently secure framework is required; Figure 1 illustrates the implemented setup of such a framework. (Rothschedl et al., 2017)

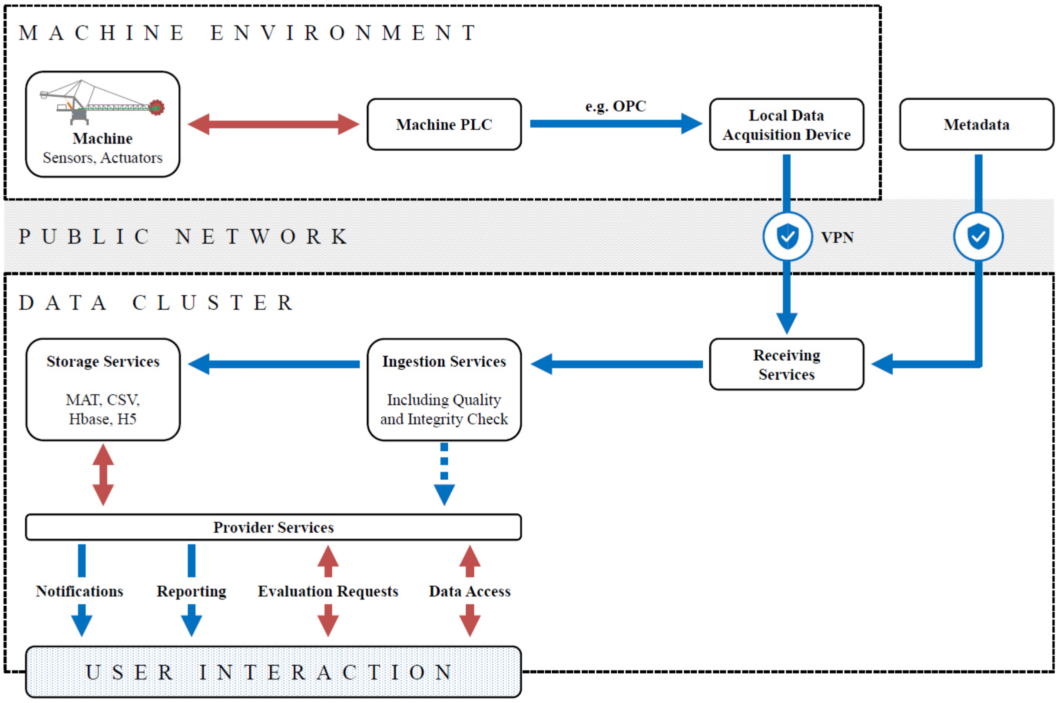


Figure 1 Schematic of the data ingestion system

It consists of a central data cluster that ingests, handles and processes operational data from currently 15 mobile mining machines, which are primarily located in remote locations and are equipped with a local data acquisition device. The ingestion process running on the data cluster ensures a consistent data format, the incoming values are accommodated in a contiguous data model. Such a model provides a distinct separation between input and output segmentation: data can be queried for an arbitrary period of time, even if the data is input on a daily basis only. Different analysing tools are available, which can be triggered manually by a user request or automatically on a regular basis. Metadata is integrated into the system via a secured connection that is entirely separated from the numerical sensor data. This is an additional security layer, as the numerical data is considered worthless without the corresponding metadata – no knowledge discovery is possible (O'Leary et al., 2016).

The available time series of the existing machines span several years, between 150 and 600 actuator and sensor signals are monitored per machine at a sampling rate of 1 Hz. Relevant channels of a bucket-wheel excavator were chosen to demonstrate the methods used in this paper.

METHODOLOGY

The herein presented analysis module gives an overview of tools and methods used for analysing hydraulics data to gain insights into machine behaviour. The aim is to support domain experts in monitoring the hydraulic systems of machines and in figuring out system improvement potential.

The developed analysis module addresses the hydraulics of boom luffing systems of machines used in mining: two cylinders are used to luff the boom up and down. The signal channels relevant for the exemplary evaluations are the rod and piston side pressures ($p_{r,i}$, $p_{p,i}$) of the cylinders in the hydraulic system. In a preliminary step, additional signal channels are used to partition the data into sequences to identify periods when the machine is operating and not operating. This is an important step, since the conclusions of evaluations may differ significantly based on the operation modes, e.g., the sequences where the machine is not operating can be used to characterise the hydraulic system, whereas sequences where the machine is operating can be used to monitor the operational behaviour.

Based on the available data, additionally derived channels can be included in the investigations, e.g., geometric metadata of the cylinders, operational forces produced by the cylinders can be calculated. They, again, can support the derivation of statistical properties, e.g., local entropy (Kollment, O'Leary, Ritt, & Klünsner, 2017).

Hydraulics Monitoring

In Figure 2, exemplary data of a bucket-wheel excavator is presented, showing the channels most relevant for the evaluations. Selected are: the luffing and slewing positions ($luff$, $slew$) to identify the current operation mode, and the pressure signals from the cylinders holding the bucket-wheel boom. Since there are two cylinders working in parallel, there are four signals: two for the piston side ($p_{p,1}$, $p_{p,2}$) and two for the rod side ($p_{r,1}$, $p_{r,2}$).

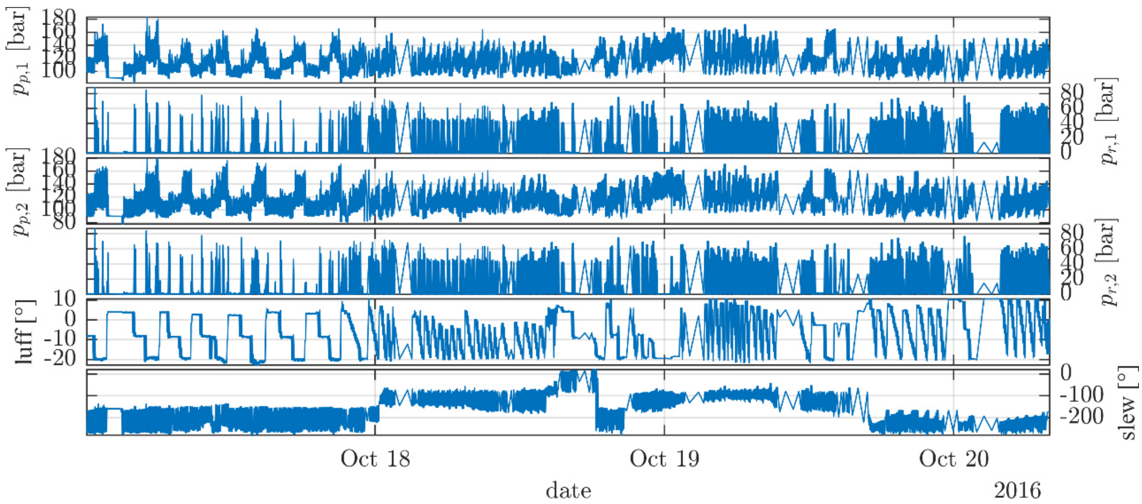


Figure 2 Relevant data channels for a duration of 4 days

In combination with the metadata of the rod and piston areas ($a_{r,i}$, $a_{p,i}$), the operating forces $f_{o,i}$ are calculated as

$$f_{o,i} = p_{p,i}\, a_{p,i} - p_{r,i}\, a_{r,i}. \tag{1}$$

Due to the fact that there are two hydraulic cylinders working in parallel, the sum and difference forces (f_s, f_d) are analysed to include an additional level of information. They are calculated as

$$f_s = f_{o,1} + f_{o,2} \tag{2}$$

$$f_d = f_{o,1} - f_{o,2}. \tag{3}$$

f_s is the total lifting force and f_d is a measure for the torsion exerted on the boom. In an idealistic case $f_d = 0$ and $f_{o,1} = f_{o,2}$. In Figure 3 those signals are shown for an entire day.

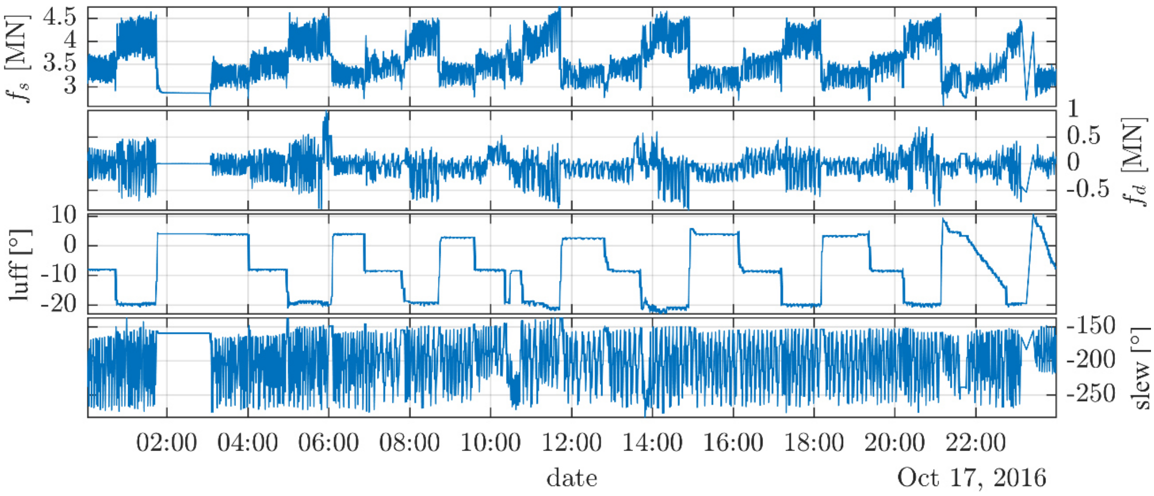


Figure 3 Data for a single day showing the sum and difference forces, the luff angle and the slew angle

As shown in Figure 4, the histogram for f_d reveals a slightly asymmetrical distribution that indicates a higher torsion in one specific direction. The reason is that the bucket-wheel is assembled at a skewed angle on the boom. This behaviour is therefore expected.

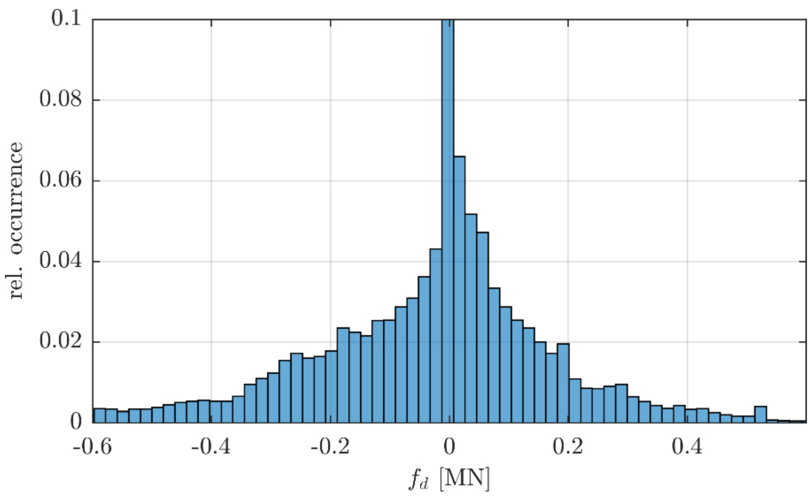


Figure 4 Histogram of f_d in the range of -0.6 to 0.6 MN

Due to the kinematics of the boom and the cylinders, a correlation between the luff angle and the total lifting force is given. This correlation can be quantified by calculating the correlation coefficient of these two signals. To do this for all channels, the correlation coefficient matrix C is derived (Lawson & Hanson, 1987). For the presented data (f_s, f_d, $luff$, $slew$) the correlation coefficient matrix constitutes to

$$C = \begin{bmatrix} 1 & -0.18 & -0.84 & -0.14 \\ -0.18 & 1 & 0.01 & -0.24 \\ -0.84 & 0.01 & 1 & 0.09 \\ -0.14 & -0.24 & 0.09 & 1 \end{bmatrix}. \tag{4}$$

The largest value at C[1,3] is the above mentioned (negative) correlation between the total lifting force and the luff angle, which is to be expected.

Statistics

The signal channels used in this work were sampled at 1 Hz over a period of 93 days. Thus, $n =$ 8.035.200 samples for each channel are available (86.400 samples per day). This large number of samples ensures a well-defined confidence interval in the statistical evaluation of the data. Since there is a contiguous data model provided by the data framework, evaluations can be triggered for different time spans. In this paper, evaluations on a daily basis are presented.

A histogram is computed for each variable for each 24-hour period. The distribution can be characterised by its statistical central moments. The first central moment is the *mean* and the second one is the *variance*. The third moment is the *skewness*: it characterises the asymmetry of the distribution. The fourth central moment is the *kurtosis* and measures the steepness of the distribution (Loether & McTavish, 1980). Those properties can be used to find anomalies and identify time sequences, which demand further investigation.

To quickly get an overview of the entire statistics for a longer time period, the histograms for each day are collected as a column in a matrix, which can be viewed and processed as images. The time-varying histogram is used to visualise the evolution of the system response behaviour to find abrupt changes: these indicate parts that require further investigation. Both, short- and long-term changes in the statistical behaviour of the system become visible quickly (Kollment et al., 2017). This method is a powerful tool for the evaluation of big data sets used in an exploratory phase.

A time-varying histogram for the piston side pressure of cylinder 2 ($p_{p,2}$) is shown in Figure 5. The sudden change indicates a major change in the system behaviour. It was confirmed, that the sensor was flawed and was changed at this exact point in time.

For a more detailed inspection the histograms of single days can be investigated. They are found in the according column of the time-varying histogram matrix.

Figure 5 Time-varying histogram of the pressure of cylinder 2 piston side for a period of 93 days

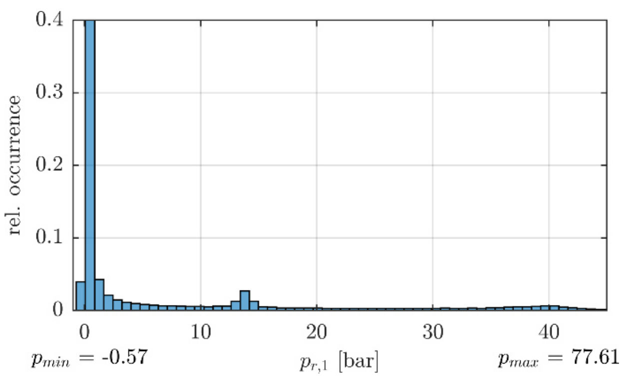


Figure 6 Single-day histogram with a pressure range of -1 to 45 bar of cylinder 1 rod side

In Figure 6, such a histogram is plotted for the rod side pressure of cylinder 1 ($p_{r,1}$). There are 2 modes visible: the mode at $p_{r,1} \approx 30$ bar[1] is the expected pre-tension of the cylinder; the mode at $p_{r,1} \approx 0$ bar indicates a low pre-tension, although the machine is in operation: this may be of interest for domain experts. Rules applied to the value ranges of the pressure signals (p_{max}, p_{min}) can be used to find data sequences where certain limits are exceeded. In Figure 6, the occurrence of negative pressure values is visible. Negative pressures increase the risk of cavitation and can cause dirt and debris to be sucked into the oil cycle through the sealings. Additionally, the stiffness of the luffing system can become more and more unstable: if the load direction of a cylinder changes when the pressure is close to zero in one of its oil chambers, the piston needs to travel a certain distance for the pressure to build up again. Such a scenario would induce or increase superstructure rocking, which can lead to further issues and can have a significant impact on other components as well (Rothschedl, 2016). Hence, close to zero or even negative pressures in the cylinder hydraulics can cause serious problems.

RESULTS AND CONCLUSION

This paper covers the steps of exploratory work on hydraulics data of a mining machine. It was shown that many characteristics of the machine's hydraulic design can be found in the data during and between machine operations. Additionally, a sensor delivering erroneous values was identified. To interpret certain patterns and findings within the data correctly, the involvement of domain experts is required. To integrate this specific domain expertise, data has to be made available to these engineering and process experts in an easy-to-work-with manner. Many of the evaluations provide a technical discussion basis that acts as an interface between the raw data and the engineers.

The presented methods can either be used to analyse historical data sets to provide engineering feedback and to optimise, or can generate reports or notifications in real-time monitoring environments. Notifications triggered by automatic evaluations can facilitate decision-making chains to ensure corrective or preventative actions are taken in a timely manner. Further tools are currently under development with the aim of representing large data sets efficiently in a compact manner, without neglecting important content.

The results presented here have been evaluated by the hydraulics design engineer and the detected defects deemed to be valid results.

[1] The unit *bar* is used for pressure because of common on-site acceptance, although it is not an SI unit.

REFERENCES

Kollment, W., O'Leary, P., Ritt, R., & Klünsner, T. (2017). Force based tool wear detection using Shannon entropy and phase plane. In *2017 IEEE International Instrumentation and Measurement Technology Conference (I2MTC)* (pp. 1–6). IEEE. http://doi.org/10.1109/I2MTC.2017.7969765

Lawson, C. L., & Hanson, R. J. (1987). *Solving Least Squares Problems (Classics in Applied Mathematics)*. *SIAM classics in applied mathematics* (Vol. 15). http://doi.org/10.1137/1.9781611971217

Loether, H. J., & McTavish, D. G. (1980). *Descriptive and Inferential Statistics: An Introduction* (2nd ed.). Allyn and Bacon.

O'Leary, P., Harker, M., Ritt, R., Habacher, M., Landl, K., & Brandner, M. (2016). Mining Sensor Data in Larger Physical Systems. *IFAC-PapersOnLine*, *49*(20), 37–42. http://doi.org/10.1016/j.ifacol.2016.10.093

Rothschedl, C. J. (2016). *Condition Monitoring of Large-Scale Slew Bearings in Bucket-Wheel Boom-Type Reclaimers*. University of Leoben.

Rothschedl, C. J., Ritt, R., O'Leary, P., Harker, M., Habacher, M., & Brandner, M. (2017). Real-Time-Data Analytics in Raw Materials Handling. In T. van Gerwe & D. Hößelbarth (Eds.), *Proceedings of Real-Time Mining, International Raw Materials Extraction Innovation Conference* (pp. 144–153). Amsterdam: Prof. Dr.-Ing. Jörg Benndorf.

A.5 Hierarchical Decomposition and Approximation of Sensor Data

Originally appeared as:

R. Ritt, P. O'Leary, C. J. Rothschedl, A. Almasri, and M. Harker, "Hierarchical Decomposition and Approximation of Sensor Data," *First International Conference on Numerical Modelling in Engineering – NME 2018*, pp. 351–370, January 2019.

Hierarchical Decomposition and Approximation of Sensor Data

Roland Ritt, Paul O'Leary, Christopher Josef Rothschedl, Ahmad Almasri, and Matthew Harker

Chair of Automation, University of Leoben, Austria,
roland.ritt@unileoben.ac.at,
WWW home page: automation.unileoben.ac.at

Abstract. This paper addresses the issue of hierarchical approximation and decomposition of long time series emerging from the observation of physical systems. The first level of the decomposition uses spatial weighted polynomial approximation to obtain local estimates for the state vectors of a system, i.e., values and derivatives. Covariance weighted Hermite approximation is used to approximate the next hierarchy of state vectors by using value and derivative information from the previous hierarchy to improve the approximation. This is repeated until a certain rate of compression and/or smoothing is reached. For further usage, methods for interpolation between the state vectors are presented to reconstruct the signal at arbitrary points. All derivations needed for the presented approach are provided in this paper along with derivations needed for covariance propagation. Additionally, numerical tests reveal the benefits of the single steps. The proposed hierarchical method is successfully tested on synthetic data, proving the validity of the concept.

Keywords: signal decimation, hermite approximation, hermite interpolation, covariance propagation, signal reconstruction, weighted regression

1 Introduction

Currently, much effort is being put into the collection of data, in particular in conjunction with IoT and smart sensors. With the rise of cyber physical systems (CPS) many data is collected from machines which, by nature, must abide by the laws of physics (e.g. dynamic systems). To analyse the behaviour of the system monitored, techniques for the approximation of the observed signal in presence of noise are necessary. A lot of research is done in the area of streaming algorithms,i.e., local regression problems [3, 7, 11, 13, 15, 16, 22, 23], for smoothing and approximation of data. In [24], a method for identifying patterns in dynamical systems using phase space was introduced. The analysis of the phase space is also used in [8] to detect fatigue damage based on ultrasonic data. To transform signals into the phase space, it is necessary to approximate data and derivatives, that is, to compute time series estimates for the state vector. In the past, the authors published work dealing with reconstruction of curves given its derivatives with the requirement to fulfil additional constraints [18, 19].

In this paper a new framework for the approximation of large time series data emanating from physical systems is presented. The main contributions of the paper are:

1. The proposal of a hierarchical approach for approximating large time series data which can be used in signal decimation;
2. The derivations for generating state vectors using spatial weighted polynomial approximation. This improves the quality of approximation by diminishing Runge's phenomenon;
3. A formulation for the approximation of data given value and derivative information (i.e. Hermite approximation). Covariance weighting is used to achieve a consistent metric used in the least squares approximation. This improves the quality of subsequent approximations;
4. The proposal to use a two-point expansion for the reconstruction of the signal and its derivatives based on the state vectors;
5. A consistent formulation of covariance propagation for the proposed derivations.

This paper is structured as follows: In Sect. 2 the framework for the hierarchical approximation is presented. For the calculation of state vectors, spatial weighted polynomial approximation is presented in Sect. 2.1. Different weighting functions are investigated and a matrix approach for the calculation of the state vectors and their according covariance matrix is presented. Section 2.2 presents a novel method for approximating data given collocated value and derivative information (state vectors). Covariance weighting and optional spatial weighting is used to achieve a consistent metric for least squares Hermite approximation. To reconstruct the signal from its decimated version (given its state vectors), a two-point expansion is proposed in Sect. 2.3 which performs better than a single point Taylor expansion. Finally, in Sect. 3 the performance of the proposed framework is tested on synthetic data. Different stages of the hierarchical approximation are presented and discussed.

2 Methodology and Algebraic Framework

A physical process $y(x)$ is observed at discrete points x_i (e.g. time, location). The observations made at these points are denoted as $\hat{y}_i$. Note: The *hat* indicates that the observation is perturbed by noise. This is said to be level 0 of the hierarchy (L_0). The n observations (signal) and locations are collected in the vector $\hat{\boldsymbol{y}} = \left[\hat{y}_1, \ldots, \hat{y}_n\right]^{\mathrm{T}}$ and $\boldsymbol{x} = \left[x_1, \ldots, x_n\right]^{\mathrm{T}} = \left[x_{(0),1}, \ldots, x_{(0),n}\right]^{\mathrm{T}}$, whereby $x_i = x_{(0),i}$ denotes the i-th location (i-th point) in level 0.

In the first level (L_1) of the proposed hierarchical approach the signal and its derivatives up to a certain order d, i.e, the state vectors $\boldsymbol{s}_j$, are approximated at collocated points x_j. These local estimates for the state vectors of the signal to be monitored are defined as

$$\boldsymbol{s}_j = \boldsymbol{s}_{(1),j} \triangleq \left[y_{(1),j}, \dot{y}_{(1),j}, \ddot{y}_{(1),j}, \ldots, y^{(d)}_{(1),j}\right]^{\mathrm{T}} = \left[y(x_j), \dot{y}(x_j), \ddot{y}(x_j), \ldots, y^{(d)}(x_j)\right]^{\mathrm{T}},$$

where $x_j = x_{(1),j} = x_{(0),i=j\times l_1}$ denotes the j-th point[1] in level 1. In this level the first decimation takes place, since only points with a spacing l_1 are approximated. In ot-

[1] To simplify readability, x_i is used for points in L_0, x_j for points in L_1 and x_k for points in L_2 and above if not defined in another way. The subscripts (0), (1) and (2) denote the different levels.

her words, only the points $x_{(0),i=j\times l_1}$ from L_0 are approximated in L_1. Additionally, the matrix Λ_j corresponding to the covariance of $\boldsymbol{s}_j$ can be computed along with the approximation. To improve the quality of the fit, a local weighting function $w_j(x)$ is used to perform weighted regression. Local weighting is used to obtain behaviour similar to splines; that is, input data only influence the result of the approximation in a finite region. The use of weighting functions which limit both, the values and derivatives at the end of the interval also reduce the Runge phenomenon. This is very closely related to Gibbs error and windowing in Fourier analysis [10].

The levels 0 and 1 (L_0 and L_1) of the hierarchical process is shown in Fig. 1.

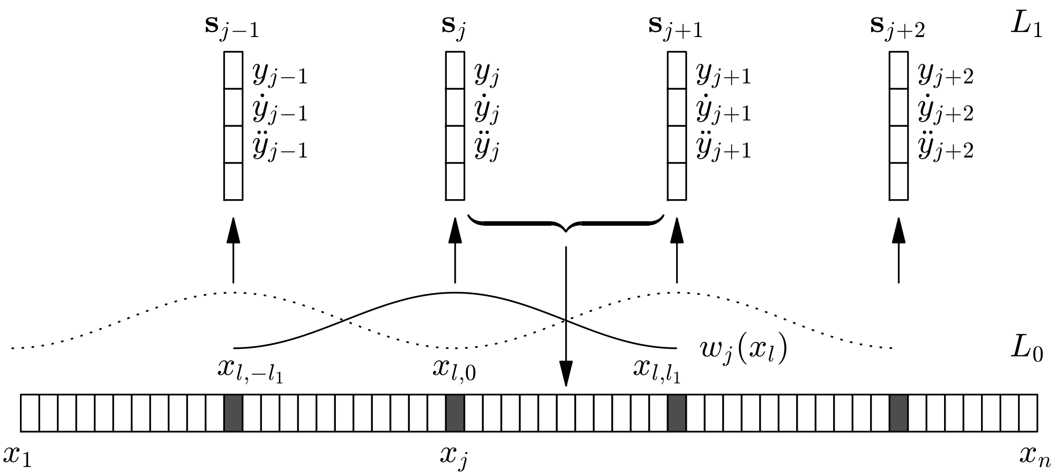


Fig. 1: Level 0 and 1 of the hierarchical approach: decimation is performed using weighted local polynomial approximation for obtaining local estimates for the state vectors $\boldsymbol{s}_j$ (and their covariances Λ_j).

Based on the state vectors $\boldsymbol{s}_j$ and their covariances Λ_j, a new method of covariance weighted Hermite approximation is used to perform the next level of hierarchical approximation. In this step, decimation can be implemented as well, if the approximated function is only evaluated at certain points x_k. As a result, you get the collocated state vectors $\boldsymbol{s}_k = \boldsymbol{s}_{(2),k}$ of level 2 (L_2). Since the derivatives are included in the approximation, a better confidence in the approximation is reached. Additional, spatial weighting can be implemented as well. This hierarchical process (Hermite approximation) can be repeated until the needed level of abstraction and smoothing (and/or decimation) of the signal is reached. This repetitive process is visualized in Fig. 2.

After this, the state vectors (i.e. the decimated signal) can be used to reconstruct (interpolate) the signal at the original positions (or somewhere in-between) using some form of expansion. or interpolation. Since there is derivative information available, an interpolation of higher degree is possible, resulting in better reconstruction. The necessary algebraic formulations are collected in the next sections.

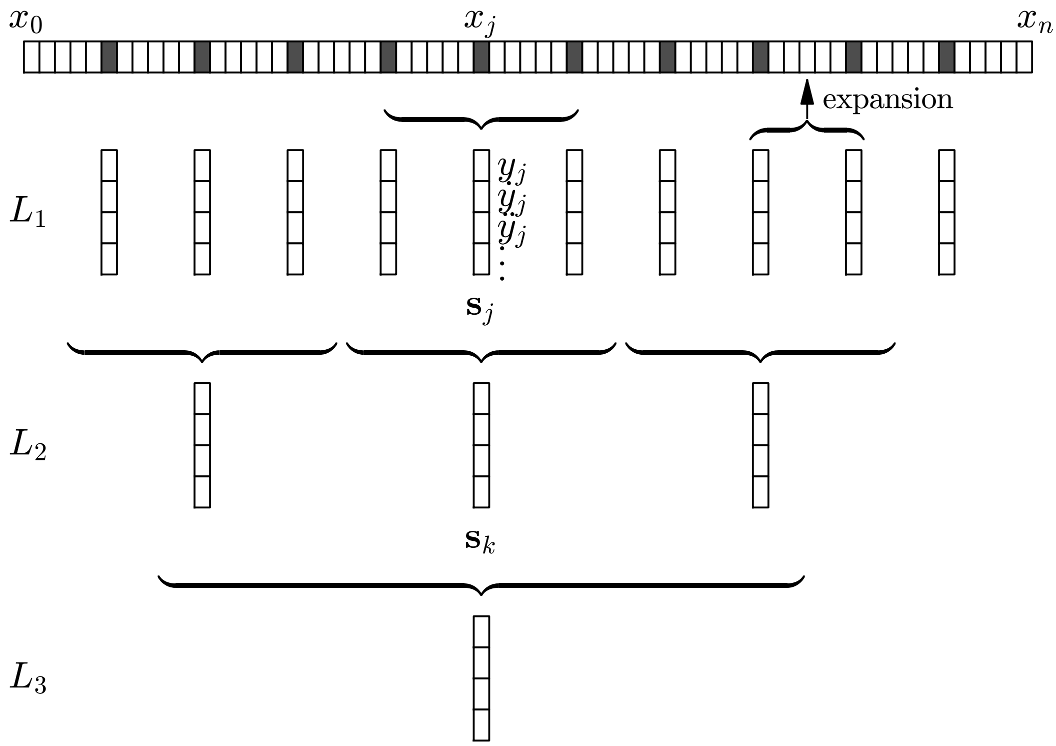


Fig. 2: Schematic of the higher levels of hierarchical approximation by using repetitive covariance weighted Hermite approximation.

2.1 Weighted Local Polynomial Approximation - Hierarchy Level 1

Local weighted regression is well-known in literature [3,5,13,15,17,20,22] with different studies on the weighting function to be used. In this work we investigate different weighting functions suitable for the herein presented hierarchical approach.

Weighting Functions: As [3] proposed, a local weighting function $w_j(u)$ should fulfil the following properties:

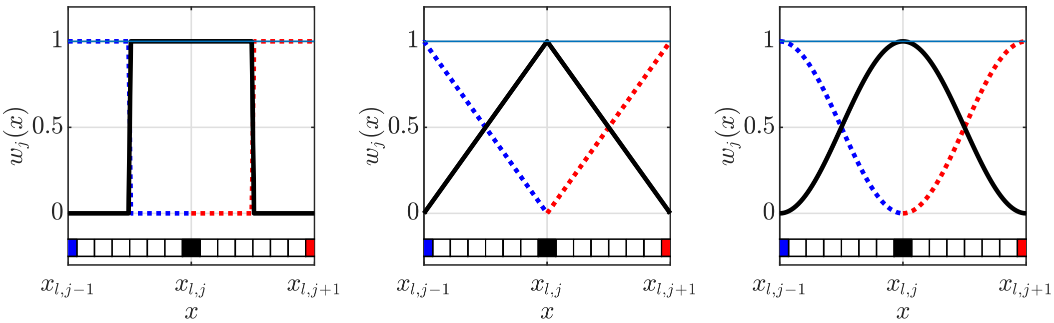
1. $w_j(u) > 0$ for $|u| < 1$: the weighting function influences only points in a certain range $u \in]-1,1[$;
2. $w_j(-u) = w_j(u)$: it is symmetric around $u = 0$;
3. $w_j(u)$ is a non-increasing function for $|u| \geq 0$: the weighting function decreases with increasing distance to the point of interest;
4. $w_j(u) = 0$ for $|u| \geq 1$: everything outside the local window does not influence the approximation.

In our proposed framework, the weighting function w_j is shifted to the point of interest x_j and scaled, so that only a certain number of points $n_{w,1} = 2l_1 - 1$ is within the local window,

$$u = 2\frac{x - x_j}{x_{j+1} - x_{j-1}}. \tag{1}$$

The number of points in the local window used to generate the state vectors in level 1 are denoted by $n_{w,1}$. Note: In the case of evenly spaced points, the weighting function is already 0 at the points x_{j-1} and x_{j+1}, i.e., $w_j(x_{j-1}) = w_j(x_{j+1}) = 0$. In this work two considerations are made with respect to the weighting functions $w_j(u)$:

1. Overlapping weighting functions (from the neighbouring local approximations) should form a partition of unity ($\sum_{j=1}^{n_j} w_j(x) = 1$). This ensures that all points in the input stream contribute with the same total weighting to the result. Some possible weighting functions are shown in Fig. 3. Of special interest is the *raised cosine*



(a) Piecewise constant (b) Triangular (c) Raised cosine

Fig. 3: Weighting functions forming a partition of unity; *black*: local weighting function $w_j(x)$ for approximating $\boldsymbol{s}_j$; *blue*: sum of the local weighting functions; *blue, red dotted*: parts of the weighting functions $w_{j-1}(x)$ and $w_{j+1}(x)$ for generating the neighbouring state vectors $\boldsymbol{s}_{j-1}$ and $\boldsymbol{s}_{j+1}$; *bottom*: schematic visualization of the position of original data points to be weighted.

function

$$w_j(u) = \frac{1+\cos(\pi u)}{2}. \tag{2}$$

This function is known as the *Hanning* window in Fourier analysis which is used to diminish the Gibbs error [10]. This function also provides a first derivative of 0 at the end of the interval and at the centre point, i.e, $\frac{\mathrm{d}w_j}{\mathrm{d}u}\big|_{u=-1} = \frac{\mathrm{d}w_j}{\mathrm{d}u}\big|_{u=1} = \frac{\mathrm{d}w_j}{\mathrm{d}u}\big|_{u=0} = 0$; this is advantageous with respect to the Runge phenomenon.

2. Alternatively, we may wish to define a weighting function $w_j(u)$ such that its values and derivatives up to the k^{th} order tend to zero at the ends of the support; that is,

$$\lim_{|u|\to 1} w^{(0)}(u) \to 0, \qquad \ldots, \qquad \lim_{|u|\to 1} w^{(k)}(u) \to 0. \tag{3}$$

This can be achieved using the polynomials,

$$w(u) = (u-1)^{(k+1)}\,(u+1)^{(k+1)}. \tag{4}$$

The polynomials computed using this weighting function are special cases of the Jacobi polynomials. These functions do not directly form a partition of unity but approximate it. The weighting function for $k = 1$ is shown in Fig. 4.

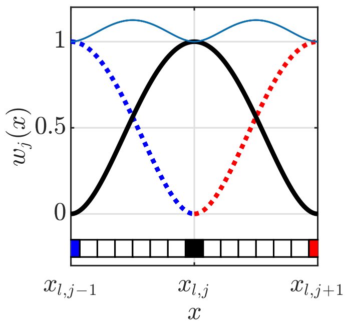


Fig. 4: Jacobi polynomial with $k=1$; *black*: local weighting function $w_j(x)$ for approximating $\boldsymbol{s}_j$; *blue*: sum of the local weighting functions; *blue, red dotted*: parts of the weighting functions $w_{j-1}(x)$ and $w_{j+1}(x)$ for generating the neighbouring state vectors $\boldsymbol{s}_{j-1}$ and $\boldsymbol{s}_{j+1}$; *bottom*: schematic visualization of the position of original data points to be weighted.

Spatial Weighted Local Regression: After choosing a weighting function, weighted regression is performed. The points and the observed values within the segment j are denoted as $\boldsymbol{x}_{l,j}$ and $\hat{\boldsymbol{y}}_{l,j}$. The according weightings are collected in the vector $\boldsymbol{w}_j = w_j(\boldsymbol{x}_{l,j})$. This vector is expanded to form the diagonal weighting matrix $\mathsf{W}_j = \operatorname{diag}\left\{\boldsymbol{w}_j\right\}$. If a linear model of the form

$$\boldsymbol{y}_{l,j} = \mathsf{B}_j \boldsymbol{\alpha}_j \tag{5}$$

is used to model the signal, the cost function to be minimized can be written as

$$\varepsilon = \left(\hat{\boldsymbol{y}}_{l,j}^{\mathrm{T}} - \boldsymbol{\alpha}_j^{\mathrm{T}} \mathsf{B}_j^{\mathrm{T}}\right) \mathsf{W}_j \left(\hat{\boldsymbol{y}}_{l,j} - \mathsf{B}_j \boldsymbol{\alpha}_j\right), \tag{6}$$

which is the sum of the weighted squared errors. B_j is a basis function set and $\boldsymbol{\alpha}_j$ are the according coefficients for modelling the signal. Now the matrix square root of W is defined as

$$\mathsf{U}_j = \mathsf{W}_j^{1/2} \qquad \text{with} \qquad \mathsf{U}_j \mathsf{U}_j = \mathsf{W}_j. \tag{7}$$

Note: if W_j is positive semi-definite and symmetric, the numerically more stable Cholesky factorization ($\mathsf{W}_j = \mathsf{U}_j^{\mathrm{T}} \mathsf{U}_j$) can be used instead of the matrix square root. Minimizing (6) with respect to $\boldsymbol{\alpha}_j$ delivers the least-squares solution, i.e.,

$$\boldsymbol{\alpha}_j = \{\mathsf{U}_j \mathsf{B}_j\}^{+} \mathsf{U}_j \hat{\boldsymbol{y}}_{l,j}, \tag{8}$$

where $\{\mathsf{U}_j \mathsf{B}_j\}^{+}$ denotes the pseudoinverse of $\mathsf{U}_j \mathsf{B}_j$.

To calculate the state vector $\boldsymbol{s}_j$, the model and the derivatives of the model up to order d are evaluated at the point x_j, yielding

$$\boldsymbol{s}_j = \begin{bmatrix} y_j^{(0)} \\ \vdots \\ y_j^{(d)} \end{bmatrix} = \begin{bmatrix} \mathsf{B}_j^{(0)}(x_j) \\ \vdots \\ \mathsf{B}_j^{(d)}(x_j) \end{bmatrix} \boldsymbol{\alpha}_j = \tilde{\mathsf{B}}_j \boldsymbol{\alpha}_j, \tag{9}$$

where $y_j^{(m)}$ is the approximation of the m-th derivative of the signal at the point x_j, and $\mathsf{B}_j^{(m)}(x_j)$ is the m-th derivative of the basis function set B_j evaluated at x_j, so that

$$y_j^{(m)} = \mathsf{B}_j^{(m)}(x_j)\boldsymbol{\alpha}_j. \tag{10}$$

$\tilde{\mathsf{B}}_j$ is the concatenated matrix containing the basis functions and their derivatives evaluated at the point of interest.

Covariance propagation: Additionally, given the covariance $\Lambda_{l,j}$ associated with the original data in the segment j, the covariance for the state vector $\Lambda_{s,j}$ can be propagated using first order covariance propagation, i.e.,

$$\Lambda_{s,j} = \tilde{\mathsf{B}}_j \mathsf{A}_j \Lambda_{l,j} \mathsf{A}_j^{\mathrm{T}} \tilde{\mathsf{B}}_j^{\mathrm{T}} \tag{11}$$

with

$$\mathsf{A}_j = \{\mathsf{U}_j \mathsf{B}_j\}^{+} \mathsf{U}_j. \tag{12}$$

The covariance matrix $\Lambda_{l,j}$ is either known a priori (e.g. knowing the error associated with the sensor used for observation) or can be calculated from the data, e.g., via the norm of the residual vector (especially when using a polynomial basis).

The impact of the weighting function onto the covariance of the approximation is shown in Fig. 5. As a basis function set a Vandermonde basis (polynomial basis) of degree 4 is chosen along with an i.i.d noise on the input. Clearly it can be seen that the piecewise constant weighting function performs badly outside its support due to the Runge phenomenon [6, 25].

The above calculation is repeated for each segment $\boldsymbol{x}_{l,j}$ to approximate the decimated signal at the points x_j. If the signal is equally spaced and the window size does not change, the matrices W_j, B_j and A_j are the same for each interval, i.e, $\mathsf{W}_j = \mathsf{W}$, $\mathsf{B}_j = \mathsf{B}$ and $\mathsf{A}_j = \mathsf{A}$ for $j = 1 \ldots n_j$. This is of major advantage, since these matrices can be calculated a priori and the approximation of the state vectors (see Eq. 8) reduces to the single linear mapping $\boldsymbol{\alpha}_j = \mathsf{A}\,\hat{\boldsymbol{y}}_{l,j}$, which can be easily implemented on embedded devices and smart sensors collecting the data. This makes the presented method suitable for collecting real-time machine data. As shown later, the quality of approximation is improved using state vectors instead of standard approximation techniques using only value information.

The right choice of the basis functions used for approximation depends on the observed system. The most common basis function set, also used in this work, is the Vandermonde basis (polynomial basis). To improve stability, it is referred to discrete orthogonal basis functions [16, 18]. The availability of the state vectors opens the door to analyse signals in the so-called *pseudo phase space* (e.g. investigating Poincaré recurrence times [21]). This is especially interesting for analysing dynamic systems.

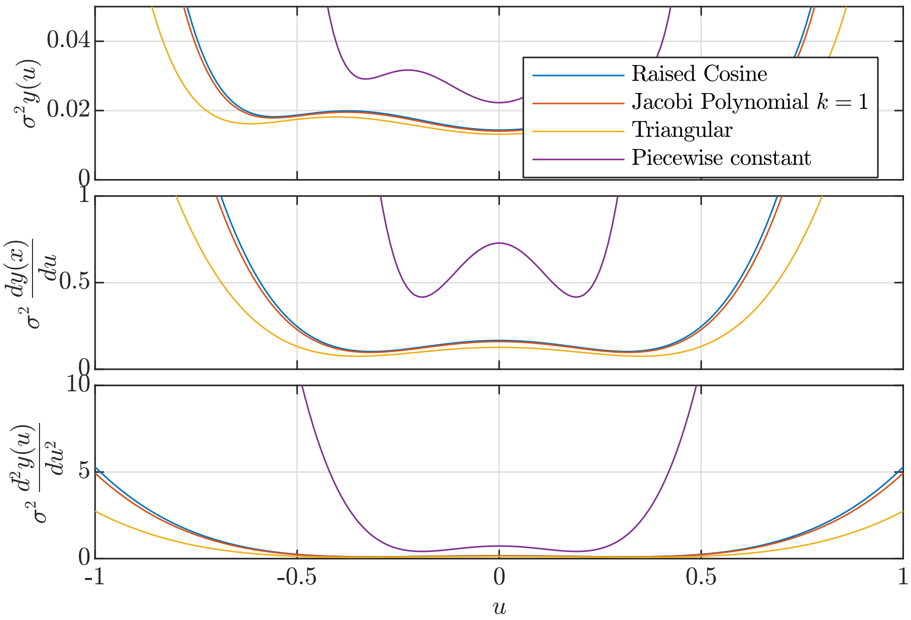


Fig. 5: This figure shows from top to bottom the propagated variances of $y(u)$, $y^{(1)}(u)$ and $y^{(2)}(u)$ respectively, for some weighting functions. Note that in all cases there is a significant improvement in the variance compared to the piecewise constant window.

2.2 Hermite Approximation - Hierarchy Level 2 and above

In the first level of the presented hierarchical approximation the signal and its derivatives (concatenated in the state vector $\hat{\mathbf{s}}_j$)[2] are approximated at points x_j. Additionally, the covariance associated with each state vector is available. The approximation of such signals (given the values and their derivatives) is not common in literature. It is related to Hermite polynomials [2]. In [1] Hermite approximation is used for multidimensional surface approximation. Hermite weighting functions are used in [12] to perform local polynomial approximation. An application using two-point Hermite approximation for solving initial value and boundary value problems can be found in [14]. The idea of reconstructing signals given the first derivatives and additional value information at some points is presented in [19]. Based on this work, we introduce a new method of doing least squares Hermite approximation using covariance weighting which extends the work to higher order derivative information. It is used to approximate higher levels of the hierarchical process presented in this paper.

Algebraic Formulation Given a system model in terms of a linear combination of basis functions, the goal is to find the according coefficients that approximate the given observations (values as well as derivatives) best, i.e., minimizing the residuals in a least squares sense. Therefore, a consistent measure for the residuals in both, the value and

[2]The ˆ notation is used here, since the given state vector is the 'noisy' input for the next level of approximation.

derivative domain, is necessary. Since the covariance information for the value and the derivatives is available at each point, it can be used to weight the residual according to this. That is, if a given information is precise, it has more impact on the solution.

First, the matrix H_j is assembled as

$$\mathsf{H}_j = \begin{bmatrix} \boldsymbol{h}(x_j) \\ \boldsymbol{h}^{(1)}(x_j) \\ \vdots \\ \boldsymbol{h}^{(d)}(x_j) \end{bmatrix}, \tag{13}$$

where $\boldsymbol{h}(x_j)$ denotes the basis function vector (e.g. Vandermonde vector). Each column of this vector contains one of the basis functions evaluated at the point of interest x_j. $\boldsymbol{h}^{(m)}(x_j)$ is the m-th derivative of the basis function evaluated at x_i and d is the number of derivatives given in the state vector $\hat{\boldsymbol{s}}_j$. The local observed state vector $\hat{\boldsymbol{s}}_j$ is now approximated by $\boldsymbol{s}_j = \mathsf{H}_j \boldsymbol{\beta}$, where $\boldsymbol{\beta}$ denotes the coefficient vector. Consequently, the local residual vector $\boldsymbol{r}_j$ is given by,

$$\begin{aligned} \boldsymbol{r}_j &= \hat{\boldsymbol{s}}_j - \boldsymbol{s}_j && (14)\\ &= \hat{\boldsymbol{s}}_j - \mathsf{H}_j \boldsymbol{\beta}. && (15) \end{aligned}$$

Additionally, for each $\hat{\boldsymbol{s}}_j$ we have the corresponding covariance $\Lambda_{s,j}$; consequently we can define an inverse covariance weighted computation of the error ε_j,

$$\begin{aligned} \varepsilon_j &= \boldsymbol{r}_j^{\mathrm{T}} \Lambda_{s,j}^{-1} \boldsymbol{r}_j && (16)\\ &= \boldsymbol{r}_j^{\mathrm{T}} \mathsf{W}_{C,j} \boldsymbol{r}_j && (17)\\ &= (\hat{\boldsymbol{s}}_j - \boldsymbol{s}_j)^{\mathrm{T}} \mathsf{W}_{C,j} (\hat{\boldsymbol{s}}_j - \boldsymbol{s}_j) && (18)\\ &= (\hat{\boldsymbol{s}}_j - \mathsf{H}_j \boldsymbol{\beta})^{\mathrm{T}} \mathsf{W}_{C,j} (\hat{\boldsymbol{s}}_j - \mathsf{H}_j \boldsymbol{\beta}). && (19) \end{aligned}$$

$\mathsf{W}_{C,j}$ denotes the covariance weighting matrix for the point j. All the individual local estimates for the state variables can be vertically concatenated to obtain the normal equations for the minimization problem. Let us start by defining

$$\Lambda \triangleq \begin{bmatrix} \Lambda_{s,1} & 0 & \dots & 0 \\ 0 & \Lambda_{s,2} & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & \Lambda_{s,n} \end{bmatrix}, \qquad \mathsf{H} \triangleq \begin{bmatrix} \mathsf{H}_1 \\ \vdots \\ \mathsf{H}_n \end{bmatrix}, \qquad \hat{\boldsymbol{s}} \triangleq \begin{bmatrix} \hat{\boldsymbol{s}}_1 \\ \vdots \\ \hat{\boldsymbol{s}}_n \end{bmatrix}. \tag{20}$$

Given Λ, the weighting matrix W_C is computed as,

$$\mathsf{W}_C \triangleq \Lambda^{-1} = \begin{bmatrix} \Lambda_{s,1}^{-1} & 0 & \dots & 0 \\ 0 & \Lambda_{s,2}^{-1} & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & \Lambda_{s,n}^{-1} \end{bmatrix} \tag{21}$$

Since all data is approximated with the same underlying function, $\boldsymbol{\beta}$ is the same for each point and the sum of the covariance weighted errors can be written as

$$\varepsilon = \sum_{j=1}^{n} \varepsilon_j = \left(\begin{bmatrix} \hat{\boldsymbol{s}}_1 \\ \vdots \\ \hat{\boldsymbol{s}}_n \end{bmatrix} - \begin{bmatrix} \mathsf{H}_1 \\ \vdots \\ \mathsf{H}_n \end{bmatrix} \boldsymbol{\beta} \right)^{\mathrm{T}} \begin{bmatrix} \Lambda_{s,1}^{-1} & 0 & \dots & 0 \\ 0 & \Lambda_{s,2}^{-1} & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & \Lambda_{s,n}^{-1} \end{bmatrix} \left(\begin{bmatrix} \hat{\boldsymbol{s}}_1 \\ \vdots \\ \hat{\boldsymbol{s}}_n \end{bmatrix} - \begin{bmatrix} \mathsf{H}_1 \\ \vdots \\ \mathsf{H}_n \end{bmatrix} \boldsymbol{\beta} \right). \quad (22)$$

This is again a weighted regression problem of the form

$$\min_{\boldsymbol{\beta}} \varepsilon = \min_{\boldsymbol{\beta}} (\hat{\boldsymbol{s}} - \mathsf{H}\boldsymbol{\beta})^{\mathrm{T}} \mathsf{W}_C (\hat{\boldsymbol{s}} - \mathsf{H}\boldsymbol{\beta}) \quad (23)$$

with the solution,

$$\boldsymbol{\beta} = \left(\mathsf{W}_C^{1/2} \mathsf{H} \right)^{+} \mathsf{W}_C^{1/2} \hat{\boldsymbol{s}}. \quad (24)$$

Now defining,

$$\mathsf{L} \triangleq \left(\mathsf{W}_C^{1/2} \mathsf{H} \right)^{+} \mathsf{W}_C^{1/2}, \quad (25)$$

we also obtain the covariance for $\boldsymbol{\beta}$, i.e.,

$$\Lambda_{\boldsymbol{\beta}} = \mathsf{L} \Lambda \mathsf{L}^{\mathrm{T}}. \quad (26)$$

Local Hermite Approximation The algebraic formulation given above is a generalized version for approximating data from given values and its derivatives. For the proposed hierarchical algorithm this approximation is applied locally to a window k which spans only a certain number of the state vectors approximated in level 1. This is shown in Fig. 2. As a result we get approximations for the coefficients for each window denoted as $\boldsymbol{\beta}_k$. Additionally, spatial weighting can be added as described in Sect. 2.1. When using spacial weighting captured in the weighting matrix $\mathsf{W}_{S,k}$ along with the covariance weighting $\mathsf{W}_{C,k}$, the matrix W_C in (21)-(25) is replaced by the matrix product of the two local weighting matrices $\mathsf{W}_{SC,k} = \mathsf{W}_{S,k} \mathsf{W}_{C,k}$. To implement decimation also in this hierarchical level, the approximation of the state vectors is only done for the centre x_k of the local window, yielding $\boldsymbol{s}_k$, i.e.,

$$\boldsymbol{s}_k = \mathsf{H}_k \mathsf{L}_k \hat{s}_k. \quad (27)$$

H_k denotes the matrix of basis functions (and their derivatives) evaluated at the point x_k as given in (13), L_k denotes the local version of (25) and $\hat{s}_k$ is the concatenation of all state vectors derived in level 1 within the local window. Again, the covariance for this level of state vectors can be propagated as

$$\Lambda_{s,k} = \mathsf{H}_k \mathsf{L}_k \Lambda_k \mathsf{L}_k^{\mathrm{T}} \mathsf{H}_k^{\mathrm{T}}, \quad (28)$$

where Λ_k is the local version of Λ.

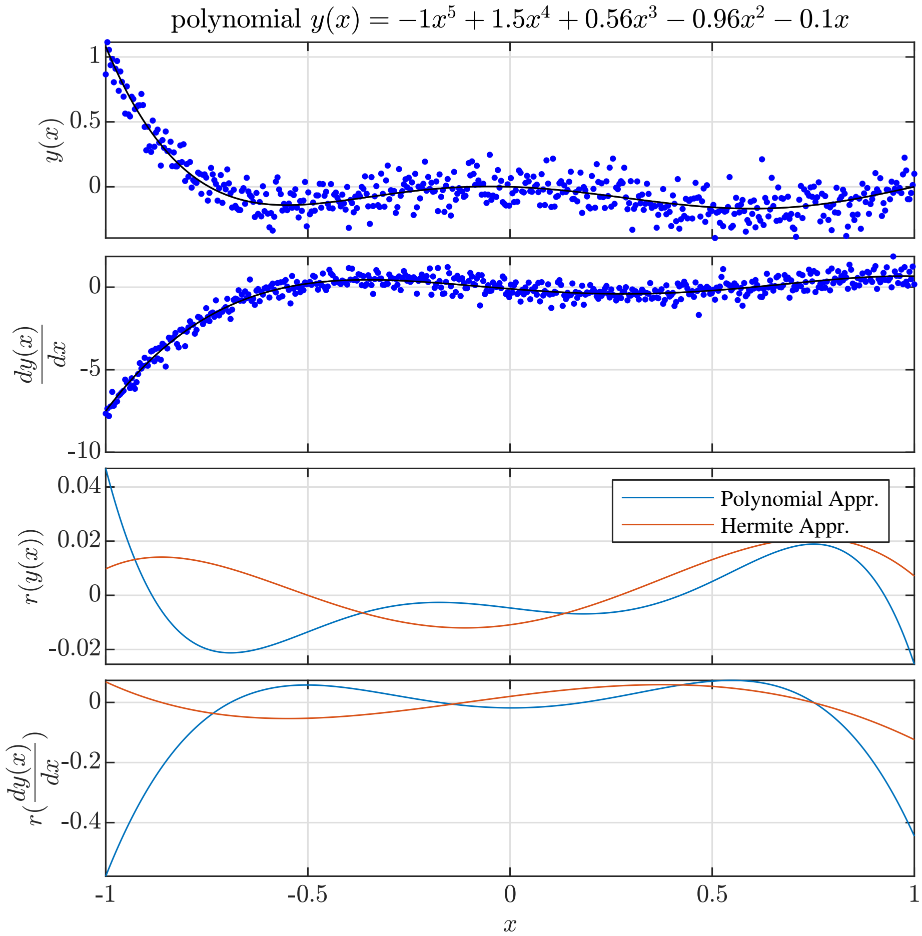


Fig. 6: Synthetic data used for Monte-Carlo simulation. The two plots on the top show the function and its derivative with the addition of Gaussian noise. The two plots on the bottom show residuals for the approximation of the values and the derivatives for both, the Hermite approximation and a standard polynomial fit, with twice the number of samples (same amount of data).

Performance Test A Monte-Carlo experiment with $n = 10000$ iterations revealed that using Hermite approximation is of advantage, if both, the signal and the derivatives, should be approximated well. Therefore, a synthetic polynomial data set (see Fig. 6) was generated with known values and first derivatives. A Gaussian noise with $\sigma_y = 0.1$ was added to the values and noise with $\sigma_{\text{dy}} = 0.4$ was added to the first derivatives. The covariance weighted regression was performed without spatial weighting. The result was compared to standard polynomial fitting with the doubled number of samples but without derivative information (to provide an identical amount information in both methods, to make them comparable). As a measure, the weighted norm of the residual vector $\boldsymbol{r_y} = \boldsymbol{y}_{\text{approx}} - \boldsymbol{y}_{\text{orig}}$ and $\boldsymbol{r}_{\text{d}\boldsymbol{y}} = \dot{\boldsymbol{y}}_{\text{approx}} - \dot{\boldsymbol{y}}_{\text{orig}}$ was taken. The results are presented in Table 1. As it can be seen, Hermite approximation performs around 2 times better than the standard polynomial fit for approximating derivatives. On the other hand, this

Table 1: Result of the Monte-Carlo experiment.

	$\frac{1}{\sigma_y}\|\|\boldsymbol{r_y}\|\|_2$	$\frac{1}{\sigma_{dy}}\|\|\boldsymbol{r}_{d\boldsymbol{y}}\|\|_2$
Hermite approximation	0.0282	0.0262
Standard polynomial fit	0.0245	0.0443

leads to a slightly worse performance in approximating values. As to be expected, the norm of the residual for the values and derivatives are nearly the same for the Hermite approximation, due to the fact of the covariance weighting. This is not the case for standard polynomial fitting. Note: the behaviour at the ends of the interval is also better for Hermite approximation (see Fig. 6).

As a result, it is suggested to provide state vectors sampled at a lower frequency instead of using only value information sampled with full sampling frequency. The amount of data does not change, whereas the quality of approximating derivatives is improved. This fact can be considered in future design of smart sensors and IoT devices collecting data of dynamic systems.

2.3 Data Reconstruction

In this section, methods for the reconstruction of the signal are investigated. During the presented hierarchical approximation, the signal is decimated, and state vectors are only available at certain points x_k. In some applications, it is necessary to describe the signal analytically to provide the possibility to do calculations at arbitrary locations and not only at discrete points. This is known as *Interpolation*. In literature, especially in digital signal processing, a lot of methods are well-established [4]. Basically, there are two main categories:

1. Local interpolation: e.g, piecewise constant, linear or spline interpolation. For each segment, a different interpolating function is used. If one point changes, only neighbouring segments are affected. Note: this is not fully true for splines, since a change in one point (knot) does affect a wider range of segments, based on the degree of continuity to be fulfilled.
2. Global interpolation: this is related to approximation, e.g., polynomial interpolation, trigonometric interpolation. A change in one point does affect the whole range to be interpolated. These types also suffer from problems which arise due to overfitting.

Taylor Expansion: In the presented hierarchical method, state vectors, containing value and derivative information, are available. A straight forward possibility is to use a one-point expansion (i.e Taylor expansion) to interpolate between two given points, i.e.,

$$f_k(x) = y_k + y_k^{(1)}\,(x - x_k) + \cdots + \frac{y_k^{(d)}}{d!}\,(x - x_k)^d\,, \tag{29}$$

where $f_k(x)$ denotes the interpolating function based on the state vector $\boldsymbol{s}_k$ at the position x_k. The order of derivatives available in the state vector is denoted by d. Since the interpolating function is a polynomial, the derivatives are simple to calculate as well. For q discrete points x_i, the interpolated points $\boldsymbol{y}_k$ can be calculated as

$$\begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_q \end{bmatrix} = \begin{bmatrix} (x_1-x_k)^d & \dots & (x_1-x_k) & 1 \\ (x_2-x_k)^d & \dots & (x_2-x_k) & 1 \\ \vdots & \ddots & \vdots & \vdots \\ (x_q-x_k)^d & \dots & (x_q-x_k) & 1 \end{bmatrix} \begin{bmatrix} \frac{1}{d!} & 0 & \dots & 0 \\ 0 & \frac{1}{(d-1)!} & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & 0 \end{bmatrix} \begin{bmatrix} y_k^{(d)} \\ y_k^{(d-1)} \\ \vdots \\ y_k \end{bmatrix}. \tag{30}$$

or in matrix vector equation as

$$\boldsymbol{y}_k = \mathsf{V}\,\mathsf{F}\,\boldsymbol{s}_k, \tag{31}$$

where V denotes the Vandermonde matrix for the expansion around x_k and F holds the scaling values resulting from Taylor approximation. For interpolating derivatives of degree d, one can use

$$\boldsymbol{y}_k^{(d)} = \mathsf{V}^{(d)}\,\mathsf{F}\,\boldsymbol{s}_k, \tag{32}$$

where $\mathsf{V}^{(d)}$ is the d-th derivative of the Vandermonde matrix.

A major problem of using this method is that discontinuities occur where the interpolating functions $f_k(x)$ and $f_{k+1}(x)$ meet (see Fig. 7).

Generalized Hermite Interpolation: To overcome the problem of discontinuities, we take the idea of Hermite Interpolation [9] and extend it to higher degrees. Given n state vectors, each with d states (order of derivatives) at each point x_k, the polynomial fulfilling all the given constraints would be at most of degree $p = n\,d - 1$. The coefficients $\boldsymbol{\rho}$ for this are computed by solving

$$\min_{\boldsymbol{\rho}} \left\| \begin{bmatrix} \boldsymbol{y} \\ \boldsymbol{y}^{(1)} \\ \vdots \\ \boldsymbol{y}^{(d)} \end{bmatrix} - \begin{bmatrix} \mathsf{V}_p \\ \mathsf{V}_p^{(1)} \\ \vdots \\ \mathsf{V}_p^{(d)} \end{bmatrix} \boldsymbol{\rho} \right\|_2^2 = \min_{\boldsymbol{\rho}} \left\| \tilde{\boldsymbol{y}} - \tilde{\mathsf{V}}_p\,\boldsymbol{\rho} \right\|_2^2, \tag{33}$$

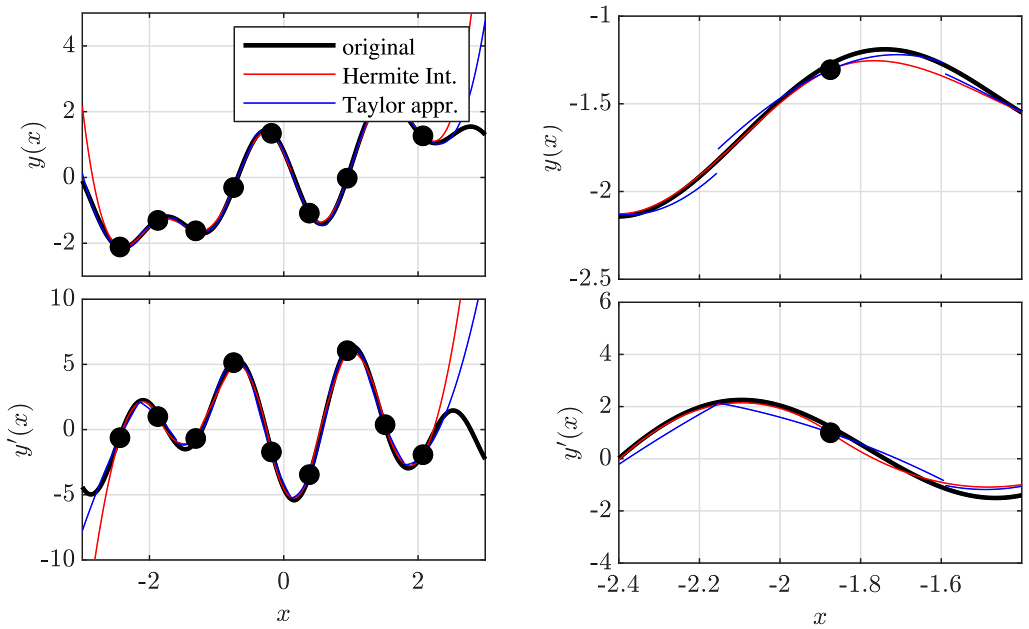
where $\boldsymbol{y}, \boldsymbol{y}^{(1)}, \dots, \boldsymbol{y}^{(d)}$ are the vectors which hold the values and the derivatives for each point x_k used for interpolation. V_p denotes the Vandermonde matrix of degree p and $\mathsf{V}_p^{(i)}$ denotes the i-th derivative of this Vandermonde matrix evaluated at the interpolating points. If $\tilde{\mathsf{V}}_p$ is full rank, the interpolating polynomial is unique. In this case, the solution is given as,

$$\boldsymbol{\rho} = \tilde{\mathsf{V}}_p^{-1}\,\tilde{\boldsymbol{y}}. \tag{34}$$

If the covariances of the state vectors are available, covariance propagation can be calculated as in the above methods.

The presented hierarchical method proposes to use this generalized Hermite interpolation to interpolate between two neighbouring points, which is a two-point expansion. Thus, a change of one state vector does only influence the two neighbouring segments.

With this method, the resulting curve (which is piecewise polynomial) is at least C^d-continuous. This type of interpolation is closely related to splines. In Fig. 7 both, the Taylor expansion and the generalized Hermite interpolation, are demonstrated. Both methods use a cubic polynomial for interpolation. As it can be seen in the plots, the Taylor expansion is discontinuous at points where two functions from neighbouring intervals meet. This is not the case for the Hermite interpolation. At the end of the interval, the Taylor expansion performs better due to the fact of being a single-point expansion with the same degree as the two-point expansion of the Hermite interpolation.



(a) Full range of data

(b) Zoomed region

Fig. 7: Hermite interpolation vs Taylor expansion.

If these methods are used to interpolate from L_1 to L_0, a consistency check can be made simultaneously to identify discontinuities in the sampled data. This can be used to trigger additional state vector samples at these point. However, this is not in the scope of this paper.

3 Numerical Testing

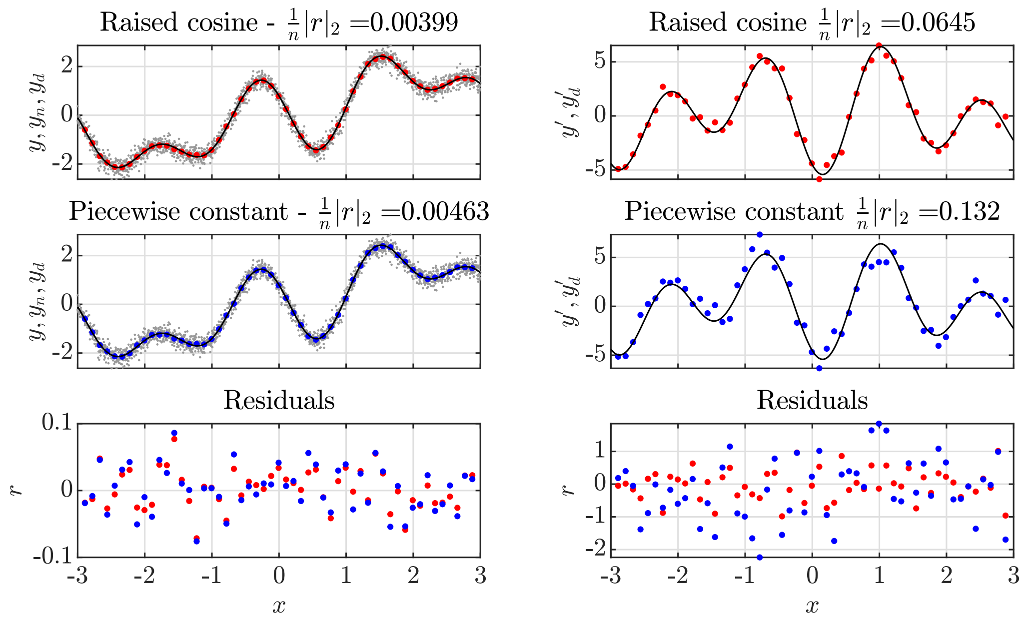
To show the abilities of the herein proposed hierarchical approximation of data, the method is tested on a synthetic dataset. The test data originate from the function

$$y(x) = \sin(x) + \sin(3 + 2.5x) + \sin(15 + 4x) + \frac{x^3}{25} \tag{35}$$

with the analytical first derivative

$$\frac{\mathrm{d}y}{\mathrm{d}x}(x) = \cos(x) + 2.5\cos(3+2.5x) + 4\cos(15+4x) + \frac{3x^2}{25}. \tag{36}$$

The function is sampled at $n = 2001$ equidistant locations. A Gaussian noise with σ_y was added to the function. To generate the first level of the hierarchical approximation (L_1), a local window covering $n_{w,1} = 75$ points from the sampled signal is chosen. This results in a decimated signal with a distance of $l_1 = 37$ data points between the approximated state vectors $\boldsymbol{s}_j$. As a local model, a polynomial of degree 1 is chosen, which can be modelled using a Vandermonde matrix ($\mathsf{B}_j = \mathsf{V}_1$). Returned are state vectors of dimension $d = 1$ containing the approximated value and first derivative. As a weighting function the piecewise constant and a raised cosine (weighted) are used for demonstration. The results are shown in Fig. 8. As it can be seen, the raised cosine

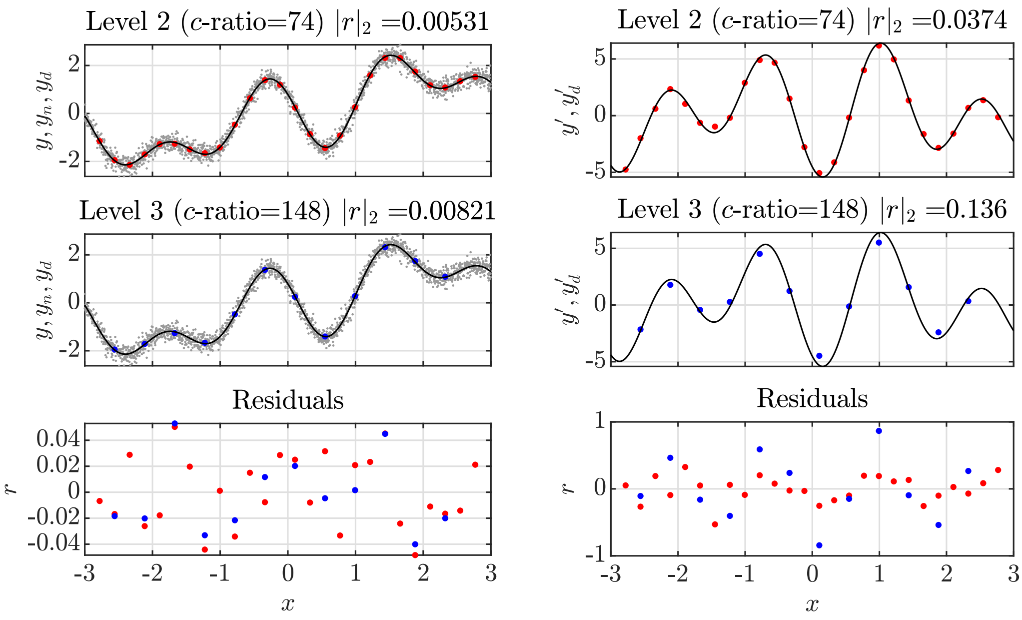


(a) Signal and approximation (b) First derivative and approximation

Fig. 8: L_1 Spatial weighted approximation; *top* and/or *red*: raised cosine weighting; *middle* and/or *blue*: piecewise constant weighting; *bottom*: residuals of approximation; *black line*: original function values; *gray*: noisy data.

weighting performs better than the piecewise constant, which is expected. Especially the approximation of the first derivative is of better quality.

After generating L_1, covariance weighted Hermite approximation is used to generate the subsequent levels (L_2 and L_3). Therefore, a local window, containing $n_{w,2} = n_{w,3} = 5$ state vectors was used, resulting in a decimated signal with a distance of $l_2 = 74$ and $l_3 = 148$ data points in terms of the original signal. This corresponds to a compression c-ratio$= 74$ for level 2 and c-ratio$= 148$ for level 3. For the approximation, a polynomial of degree 2 is used. The results for both, L_2 and L_3, are shown in Fig. 9. Although the



(a) Signal and approximation

(b) First derivative and approximation

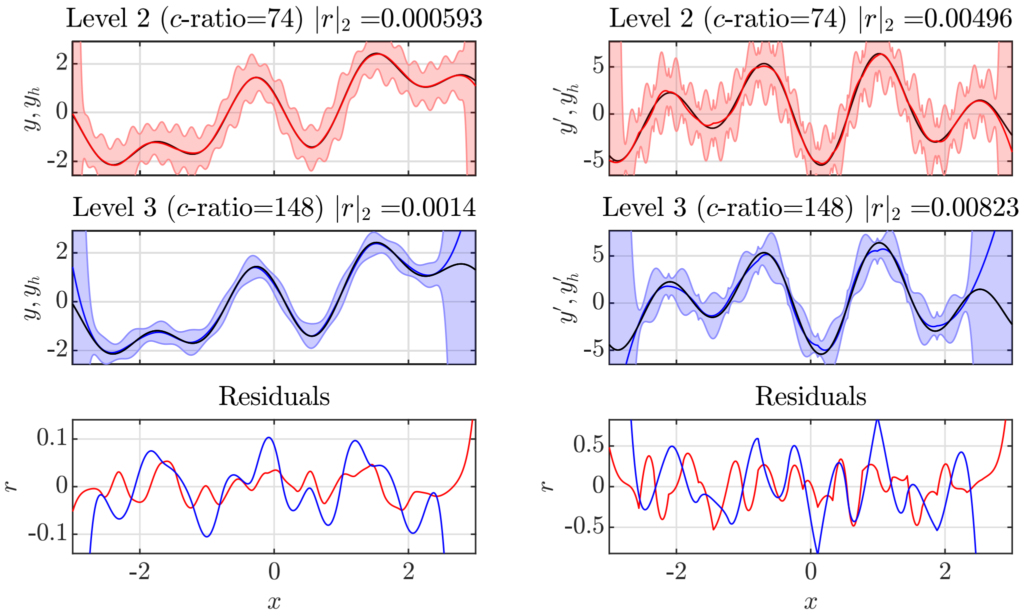
Fig. 9: L_2 and L_3 Hermite approximation (covariance and spatial weighted); *top* and/or *red*: L_2 approximation; *middle* and/or *blue*: L_3 approximation; *bottom*: residuals of approximation; *black line*: original function values; *gray*: noisy data.

compression ratio is high, the hierarchical approximation delivers good results for approximating the values and also the derivatives. This is important, if the approximation of the derivative is used for further calculation.

In Fig. 10 the given state vectors from L_2 and L_3 are used to interpolate the signal at the original locations. The proposed two-point expansion using generalized Hermite interpolation was used. Additionally, covariance propagation was performed through all levels. The resulting variance after interpolation is visualized in the figure as well. Note: different magnification gains are used to make the variance visible in the plots. Again, the proposed method delivers suitable results. As it can be seen, the covariance propagation is getting worse at the end of the interval, which is due to the fact that at the ends the signal is extrapolated, since the supporting points are missing.

4 Conclusion

This paper has presented a new method for hierarchical approximation of sensor data along with all derivations. In the first level spatial weighting was used to approximate state vectors at collocated locations. Different weighting functions have been investigated by analysing their covariance propagation. It has been shown that a piecewise



(a) Signal and interpolation

(b) First derivative and interpolation

Fig. 10: Hermite interpolation based on L_2 and L_3; *top* and/or *red*: L_2 interpolation; *middle* and/or *blue*: L_3 interpolation; *bottom*: residual of interpolation; *black line*: original function values; *shaded areas*: propagated covariances (magnified with $g = 2e3$ on the left, $g = 8e1$ top-right and $g = 4e2$ middle-right).

constant weighting function is not the method of choice. In subsequent levels of the hierarchy, a new method of covariance weighted Hermite approximation is proposed to approximate the signal by the given value and derivative information. This yields a decimated signal which maintains derivative information. This method was compared to a standard fitting method and revealed a large improvement in approximating derivatives by only a minor decrease of quality in approximating values. Based on this, it can be concluded that future sensors should deliver the state vector instead of a higher frequent signal without derivative information. To make use of the approximated state vectors, two interpolation methods are presented which approximate the original signal in a continuous sense. The presented generalized Hermite interpolation proved to be the method of choice, since the signal and its derivatives are continuous within the whole interval, which is beneficial for further derivations based on the signal. A successful test on synthetic data showed the correct functionality of the proposed hierarchical method and delivered good results for the approximation of large data sets, especially for the approximation of derivatives.

Acknowledgments

Partial funding for this work was provided by the Austrian research funding association (FFG) under the scope of the COMET program within the K2 center “IC-MPP” (contract number 859480). This programme is promoted by BMVIT, BMDW and the federal states of Styria, Upper Austria and Tyrol.

References

1. Bajaj, C.L.: Multi-dimensional Hermite Interpolation and Approximation for Modelling and Visualization. In: Proceedings of the IFIP TC5/WG5.2/WG5.10 CSI International Conference on Computer Graphics: Graphics, Design and Visualization, pp. 335–348. North-Holland Publishing Co., Amsterdam, The Netherlands, The Netherlands (1993). URL http://dl.acm.org/citation.cfm?id=645465.653690
2. Burden, R.L., Faires, J.D.: Numerical Analysis, 872 edn. Cengage Learning. Thomson Brooks/Cole (2005). URL http://books.google.at/books?id=wmcL0y2avuUC
3. Cleveland, W.S.: Robust Locally Weighted Regression and Smoothing Scatterplots. Journal of the American Statistical Association **74**(368), 829 (1979). DOI 10.2307/2286407. URL http://www.jstor.org/stable/2286407
4. Crochiere, R.E., Rabiner, L.R.: Multirate digital signal processing. Prentice-Hall (1983). URL https://de.scribd.com/doc/82733332/Multirate-Digital-Signal-Processing-Crochiere-Rabiner
5. Eilers, P.H.C.: A Perfect Smoother. Analytical Chemistry **75**(14), 3631–3636 (2003). DOI 10.1021/ac034173t
6. Epperson, J.F.: On the Runge Example. The American Mathematical Monthly **94**(4), 329 (1987). DOI 10.2307/2323093. URL https://www.jstor.org/stable/2323093?origin=crossref
7. Grabocka, J., Wistuba, M., Schmidt-Thieme, L.: Scalable Classification of Repetitive Time Series Through Frequencies of Local Polynomials. IEEE Transactions on Knowledge and Data Engineering **27**(6), 1683–1695 (2015). DOI 10.1109/TKDE.2014.2377746. URL http://ieeexplore.ieee.org/document/6975152/
8. Gupta, S., Ray, A., Keller, E.: Symbolic time series analysis of ultrasonic data for early detection of fatigue damage. Mechanical Systems and Signal Processing **21**(2), 866–884 (2007). DOI 10.1016/j.ymssp.2005.08.022. URL http://linkinghub.elsevier.com/retrieve/pii/S0888327005001329
9. Hermite, C.: Sur la formule d'interpolation de Lagrange. (Extrait d'une lettre de M. Ch. Hermite à M. Borchardt). Journal für die reine und angewandte Mathematik **84**, 70–79 (1877). URL http://eudml.org/doc/148345
10. Jerri, A.J.: The Gibbs phenomenon in Fourier analysis, splines, and wavelet approximations. Kluwer Academic Publishers (1998)
11. Joldes, G.R., Chowdhury, H.A., Wittek, A., Doyle, B., Miller, K.: Modified moving least squares with polynomial bases for scattered data approximation. Applied Mathematics and Computation **266**, 893–902 (2015). DOI 10.1016/j.amc.2015.05.150. URL http://dx.doi.org/10.1016/j.amc.2015.05.150
12. Komargodski, Z., Levin, D.: Hermite type moving-least-squares approximations. Computers & Mathematics with Applications **51**(8), 1223–1232 (2006). DOI 10.1016/j.camwa.2006.04.005. URL http://linkinghub.elsevier.com/retrieve/pii/S0898122106000757
13. Marron, J.S., Hill, C.: Local Polynomial Smoothing Under Qualitative Constraints. COMPUTING SCIENCE AND STATISTICS pp. 647—-652 (1997)
14. Mennig, J., Auerbach, T., Hälg, W.: Two point hermite approximations for the solution of linear initial value and boundary value problems. Computer Methods in Applied Mechanics and Engineering **39**(2), 199–224 (1983). DOI 10.1016/0045-7825(83)90021-X
15. Moore, A.W., Schneider, J., Deng, K.: Efficient locally weighted polynomial regression predictions. International Conference on Machine Learning pp. 236–244 (1997)

16. O'Leary, P., Harker, M.: Discrete polynomial moments and Savitzky-Golay smoothing. World Academy of Science, Engineering and Technology; International Journal of Computer and Information Engineering **72**, 439–443 (2010). URL https://waset.org/publications/12268/discrete-polynomial-moments-and-savitzky-golay-smoothing
17. O'Leary, P., Harker, M.: Surface Modelling Using Discrete Basis Functions for Real-Time Automatic Inspection. In: 3-D Surface Geometry and Reconstruction, pp. 216–264. IGI Global (2010). DOI 10.4018/978-1-4666-0113-0.ch010. URL http://services.igi-global.com/resolvedoi/resolve.aspx?doi=10.4018/978-1-4666-0113-0.ch010
18. O'Leary, P., Harker, M.: A Framework for the Evaluation of Inclinometer Data in the Measurement of Structures. IEEE Transactions on Instrumentation and Measurement **61**(5), 1237–1251 (2012). DOI 10.1109/TIM.2011.2180969. URL http://ieeexplore.ieee.org/document/6162983/
19. O'Leary, P., Harker, M.: Inverse boundary value problems with uncertain boundary values and their relevance to inclinometer measurements. In: 2014 IEEE International Instrumentation and Measurement Technology Conference (I2MTC) Proceedings, pp. 165–169. IEEE (2014). DOI 10.1109/I2MTC.2014.6860725. URL http://ieeexplore.ieee.org/document/6860725/
20. O'Leary, P., Harker, M., Neumayr, R.: Savitzky-Golay smoothing for multivariate cyclic measurement data. 2010 IEEE International Instrumentation and Measurement Technology Conference, I2MTC 2010 - Proceedings pp. 1585–1590 (2010). DOI 10.1109/IMTC.2010.5488242
21. Poincaré, H.: Sur le probleme des trois corps et les équations de la dynamique. Acta Mathematica **13**(1), 5–7 (1890). DOI 10.1007/BF02392506. URL http://link.springer.com/10.1007/BF02392506
22. Proietti, T., Luati, A.: Low-pass filter design using locally weighted polynomial regression and discrete prolate spheroidal sequences. Journal of Statistical Planning and Inference **141**(2), 831–845 (2011). DOI 10.1016/j.jspi.2010.08.006. URL http://linkinghub.elsevier.com/retrieve/pii/S0378375810003769
23. Racine, J.S.: Local Polynomial Derivative Estimation: Analytic or Taylor? In: Essays in Honor of Aman Ullah (Advances in Econometrics), vol. 36, chap. 18, pp. 617–633. Emerald Group Publishing Limited (2016). DOI 10.1108/S0731-905320160000036027. URL http://www.emeraldinsight.com/doi/10.1108/S0731-905320160000036027
24. Rajagopalan, V., Ray, A., Samsi, R., Mayer, J.: Pattern identification in dynamical systems via symbolic time series analysis. Pattern Recognition **40**(11), 2897–2907 (2007). DOI 10.1016/j.patcog.2007.03.007
25. Runge, C.: Über empirische Funktionen und die Interpolation zwischen äquidistanten Ordinaten. Zeitschrift für Mathematik und Physik **46**, 224–243 (1901)

A.6 Mimicking the Mechanisms of Language for the Unsupervised Detection of Hierarchical Structure in Time Series

Originally appeared as:

C. J. Rothschedl, P. O'Leary, and R. Ritt, "Mimicking the Mechanisms of Language for the Unsupervised Detection of Hierarchical Structure in Time Series," *Proceedings of the 6th International Conference on Time Series and Forecasting, ITISE 2019*, vol. 1, pp. 155–166, September 2019.

Mimicking the Mechanisms of Language for the Unsupervised Detection of Hierarchical Structure in Time Series

Christopher Josef Rothschedl, Paul O'Leary, and Roland Ritt

Chair of Automation, University of Leoben, Leoben, Austria
christopher-josef.rothschedl@stud.unileoben.ac.at
https://automation.unileoben.ac.at

Abstract. The purpose of this paper is to investigate, how mechanisms of natural language can support the analysis of time series data emanating from human-operated physical systems. There may be no physical models to describe human behavior reliably; nevertheless, there is commonly structure in the data acquired from observing this behavior. This paper focuses on investigating a new approach to the unsupervised detection of structure in such data. Symbolic analysis and linear differential operators are adopted, to derive results bearable for resemblance with the metaphorical concept of language and its mechanisms. To support this, phenomenology is introduced and discussed. The general significance of metaphors – and why they may be of specific relevance to analyzing time series data – is discussed and references to exemplary applications are presented to ascertain the validity of the approach.

Keywords: Symbolic analysis, time series analysis, unsupervised detection, metaphor of language, phenomenology

1 Preamble

This paper focuses on time series data emanating from cyber physical systems, i.e., on the example of machines being operated by humans. In an exploratory manner, analytical approaches are adopted to mimic basic concepts of natural language. Especially in the advent of machine learning and related techniques, it needs to be pointed out that learning is fundamentally function without understanding. Hence, such perceptive approaches alone cannot solve all kinds of problems in a generalized manner, which is why analytical methods – such as in the form of symbolic analysis – are required to acquire understanding. A discussion section at the end of this paper poses an attempt to associate results from mimicking the mechanisms of language with phenomenology; in particular with the Asian views on this and how it could be of support when analyzing time series data of human-operated equipment.

Eventually, the aim of this paper is to initiate discussions on alternative approaches to analyzing time series data emanating from phenomena, i.e., the observation of physical systems, as a possible precursor to new implementations of what we might wish to call artificial intelligence. We also present some initial results on the emergence of language in the analysis of multi-channel time series relating to physical systems: the results are not final but their quality certainly justify further research.

2 Introduction

In the following paragraphs, our approach on symbolic analysis of time series data emanating from human-operated equipment is presented. The results of the proposed concept will then be associated with components of language they potentially correspond to. The subsequent discussion section about the relationship of the presented work to phenomenology provides insights on the background and should pave the way for further discourse about its significance to the matter.

3 Symbolic Analysis of Time Series Data Emanating from Human-Operated Machines

Symbolization plays a major role in what we want to achieve with the analysis of time series data of machines, which are operated by humans. Symbolic representations of time series have been used in a number of different manners [1, 2, 10, 13, 16, 15, 17]; most notably among these is the symbolic aggregate approximation (SAX) [9]. The SAX algorithm works directly on the original data stream and uses a set of linear quantization levels to define the alphabet; in this manner, numerosity is reduced with the goal of simplifying the identification of patterns. Furthermore, the approach poses a lower bounding property, giving it a positive semi-definite measure, i.e., the distances between sequences have a strict mathematical meaning. However, SAX does not take the nature of the system being observed into account. They have neither proposed nor offered any additional abstraction relating to the metaphoric nature of language and how it points to implicit structure in the dynamics of the experience of phenomena. For example, in language we have *nouns* and *verbs*, which each refer to a different aspect of an experience and we have implicit structure which is captured by the grammar of a language, in the most general case we may consider grammar as ensuring that the symbols are ordered correctly so that they are a correct and valid reference to the experience of phenomena.

3.1 Adding Physical Dynamics

When analyzing physical systems, dynamics need to be considered in our models and approaches. Such dynamics are usually modeled by differential equations. As humans, we intuitively solve certain classes of differential equations, e.g., when crossing the road: we estimate the position, velocity and acceleration of a car to ascertain if it is safe to cross. Such capabilities are also needed in data analysis. Consequently, we propose adding linear differential operators (LDO; [8], detailed implementation in [3]) as precursors to symbolization of continuous time series data streams. Additionally, the LDO introduce the concept of dynamics, giving a natural link to the linguistic elements called *verbs*[1]. Deriving velocity or acceleration from position signals depicts simple examples for the application of LDO. This enables the generation of additional linguistic elements such as *verbs*, *adjectives*, *adverbs*, etc., which substantiates the emergence of language as a metaphor.

4 Introducing the Mechanisms of Language

We now provide a basic definition of language relating to the analysis of multi-channel time series data streams. The model is somewhat restricted; nevertheless it is sufficient to support initial research into the viability of applying the metaphor of language to real physical systems.
In Yogācāra phenomenology [11, 14], it is proposed that repetitions in our sensory excitation, which are significant to our situation, are assigned a language representation, i.e., words. Consequently, meaning is simultaneously both, experiential and contextual. It is the repetition which is considered to be characteristic and not the thinking about the repetition; this process leads to a naming. At this point we may not understand, however, we do perceive. In this manner the use of a word is considered to be a metaphor, which points at a sensory experience rather than describing the experience directly. Monosyllabic words are considered a representation of simpler sensory experiences, while polysyllabic words tend to describe more abstract experiences, which are commonly the result of complex multi-sensory experience.

In Proto-Indo-European languages, each syllable has a specific meaning and more complex experiences are expressed as *poly*syllabic words. Furthermore, predicates – primarily adjectives and adverbs – emerge to define properties of objects and activities. Other linguistic elements, such as punctuation, can also be of metaphorical support in future research. In this work we investigate the usefulness of the metaphor of language when analyzing real-time time series data of machines – particularly, when these are being operated by humans. One interesting result is that, given the "symbolized" data streams (the words), the mechanisms of language, e.g., contraction, compounding, etc., lead to a natural formulation of a hierarchical decomposition of the data, revealing implicit structure embedded in the data (see example in upcoming section 5.4).

For Lakoff and Johnson [7], metaphors exhibit influential significance beyond linguistics. They suggest that human thinking, action and speaking follow metaphorical concepts, in daily life as well as in science. In their book, metaphors are presented to possess self-contained cognitive value, instead of only being linguistic tools for simple comparisons. Furthermore, it is pointed

[1] Beyond that, LDO can be used to compute local time-dependent estimates for the state space vectors of systems, which, if mapped to the pseudo phase space, allow prediction of system behavior.

out that metaphors – instead of being mappings – could also add elements to a domain. This is exactly what we want to achieve with data analysis: integrate domain knowledge to derive understanding from time series data. Metaphors, and in our case especially the metaphor of language, can be of extensive support with regard to how complex time series data from physical systems can be conquered.

5 Mimicking the Mechanisms of Language

Time series data of bucket-wheel excavators was used to demonstrate initial ideas regarding the metaphorical relevance of emerging language. The sensor and actor data was sampled at a rate of 1 Hz. These machines used in open-pit mining are operated by humans in a non-continuous manner, making the time series data suitable for analysis in the sense of metaphorical language.

5.1 Monosyllables and Polysyllables

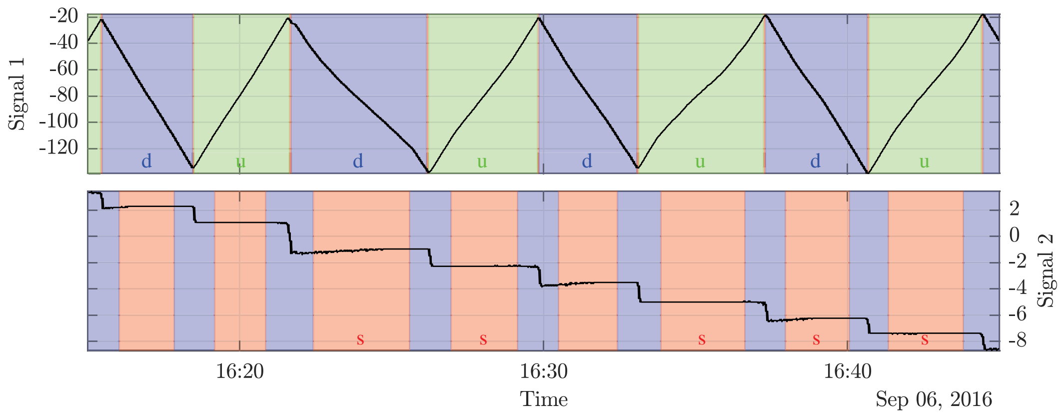
Symbolization of single channels produces words we refer to as *monosyllables*. That are, *nouns* for describing states, and *verbs* for describing activities. The results correspond to sequences of words (see Figure 1a for two individual channels) which are highly suitable for further processing, also for algorithms such as *regular expressions* (*regex*).

Cross-channel combinations of these monosyllables yield *polysyllables*, describing more complex states and/or activities across several channels in a more abstract manner. The example given in Figure 1b illustrates polysyllables gained from the combination of monosyllables of the channel Signal 1 (verbs; **u** (green) and **d** (blue), both describing different directions of boom slewing) and monosyllables of Signal 2 (nouns; **s** (red), identifying non-motion states of boom luffing). The newly generated syllables (**ds** and **us**) now describe machine operations on a more general, abstract level. Such methods support the integration of knowledge from experience of domain experts, as time series data is prepared in a way more practicable for the analysis of (complex) machine processes.

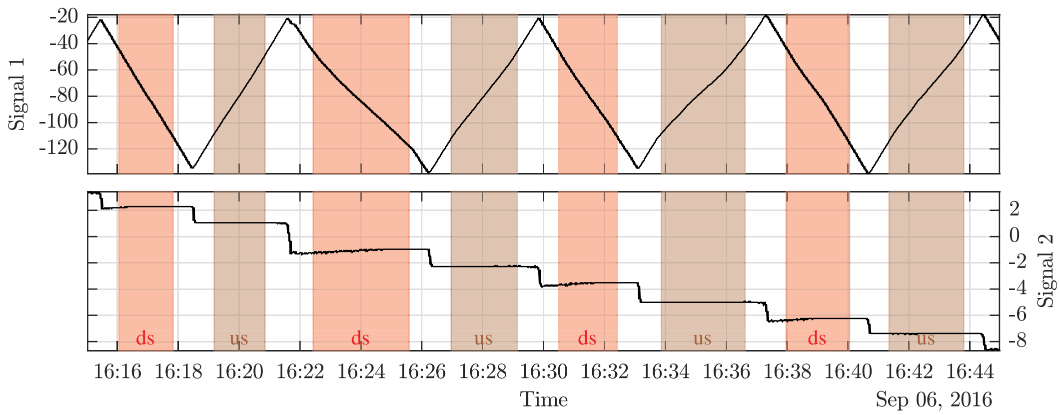
5.2 Predicates: Adjectives, Adverbs

By symbolizing time series data, states (nouns) and activities (verbs) are predicated with their run-lengths[2] (*adjectives*, *adverbs*). The example in Figure 2 shows two separate sequences of a time series with their symbolic mapping. The identified symbol sequences along with the corresponding predicates (referring to the run-length) are shown on the right. This twofold structure (nouns/verbs and associated adjectives/adverbs) can be used when analyzing the underlying structure of such sequences, e.g., both signal snippets in the figure show a different time range, 50 and 40 minutes. However, the sequence structure is the

[2] *Runs* of data are sequences of consecutive same symbols within a time series.



(a) Monosyllables of two individual signal channels are shown at the top, where Signal 1 is symbolized by verbs to represent the specific motion (boom slewing left/right) of the machine and Signal 2 shows the position of the boom luffing unit (up/down) as nouns (**s** in red color identifies the state *luffing is stationary*).



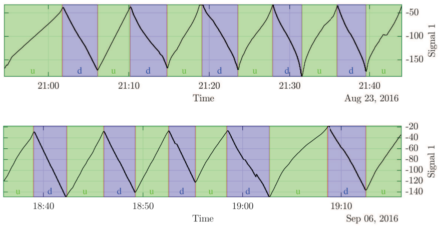
(b) The individual monosyllables from above are combined to form polysyllables that refer to all occurrences where Signal 1 exhibits actions **d** or **u** and Signal 2 exhibits the state **s**.

Fig. 1: Monosyllables and their combinations yielding polysyllables

same, although the lengths of the individual runs differ. This form of run-length encoding enables data compression.

The adjectives and adverbs can deliver additional information, such as motion time or speed, depending on what the corresponding nouns or verbs indicate. Furthermore, irregularities can be detected by identifying words with an uncommon length, speed, etc., or certain analyses can be made possible when omitting words which do not reach a certain length or speed limit.

Symbolized time series



Words with corresponding predicates

431	262	239	274	257	267	256	218	264	215	267
u	**d**	**u**	**d**	**u**	**d**	**u**	**d**	**u**	**d**	**u**

180	199	220	189	198	160	186	256	348	234	212
u	**d**	**u**	**d**	**u**	**d**	**u**	**d**	**u**	**d**	**u**

Fig. 2: Symbolized time series of different time ranges are shown on the left (top: 50 minutes; bottom: 40 minutes), while the corresponding word sequences can be seen on the right side. The words (**u**, **d**) are printed along with the associated predicates (run-lengths). Although the predicates of both signal snippets differ, they exhibit the same underlying structure.

5.3 Frequency Dictionaries

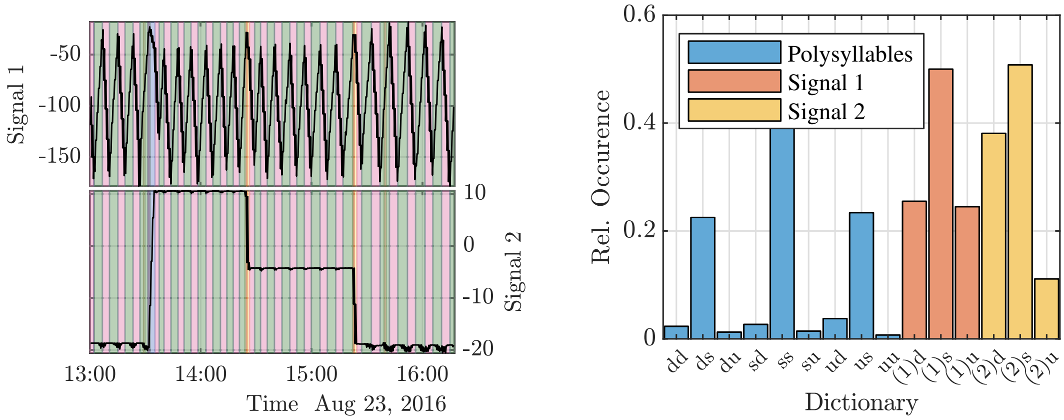
By interpreting time series and the combinations of the individual channels, we not only yield sequences of words, but also the frequency of all word occurrences. Picking up the idea of *frequency dictionaries*, words can be sorted by their frequency to reveal which words are common and which are not. This is especially interesting for machines: if a frequency dictionary is generated during controlled tests in the commissioning phase of a machine, it can build the reference for how the response behavior of the system should look like — a form of *operations recognition*. During operation, deviations can be identified by comparing this reference operation profile with the actual profile. Also outliers, i.e., words that never or rarely occurred during commissioning, or unusual distributions of words, can be detected; further investigations can be performed if necessary.

In Figure 3a, there are two channels characterizing a certain machine operation mode[3] on the left side, while on the right plot the occurrence statistics of the words, mono- and polysyllabic, are given. In contrast, Figure 3b shows another operation, again based on the two channels with the corresponding statistics plot. To distinguish between the two operations, the information regarding the frequency of the polysyllables is essential, as the statistics of the monosyllables of both operations do not differ as distinctively. This indicates that mechanisms of forming polysyllabic words reveal additional information about a system by letting a machine-specific language emerge.

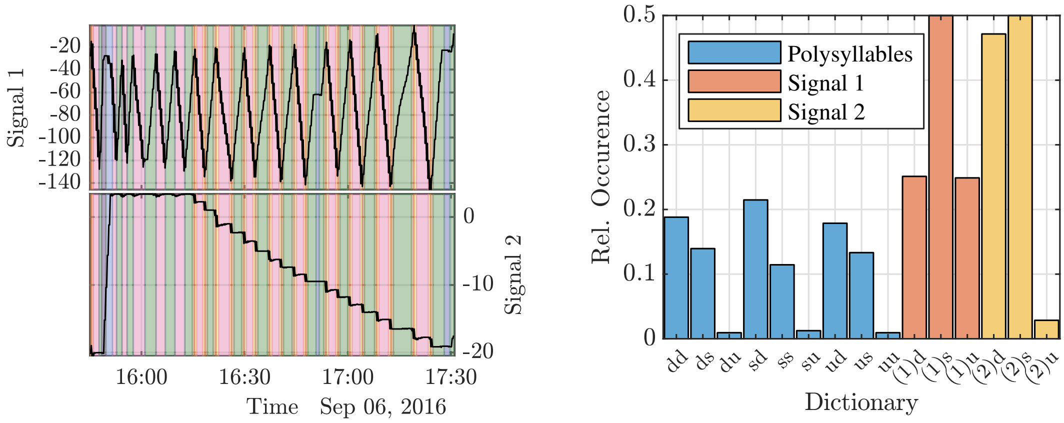
5.4 Compounding

In natural language, patterns that occur repeatedly are often combined to form a simpler and less complex identifier, i.e., a metaphor describing what is meant. For instance, offering a cup of coffee to someone will be understood easily while

[3] For demonstration purposes, the count of signals relevant to the identification of operation modes was limited to two.



(a) Operation mode 1: The bucket-wheel excavator is operated in terrace-cut mode.



(b) Operation mode 2: The bucket-wheel excavator is operated in drop-cut mode.

Fig. 3: Two time series signals with their corresponding mapping of mono- and polysyllables are shown on the left side for each operation mode of a bucket-wheel excavator. The statistics for each mode are plotted on the right.

offering “a ceramic container with blackish fluid, white substance from an animal and sweet-tasting carbohydrates” instead might leave most in confusion. Additionally, this example might also imply another level of abstraction where the initiator seeks a more casual means of communication outside his office by sharing a coffee in the cafeteria.
A similar issue arises when analyzing large time series: the simpler a certain system response behavior or machine operation can be represented for subsequent processes, the more efficient it is to draw conclusions. While the initial time series are only present as raw data signals, the compounding of nouns and verbs (from intermediate symbolization) adds a substantial level of abstraction by providing a succinct identifier (metaphor) for a complex operation or event/incident.
A simple example using two signals of a bucket-wheel excavator is illustrated in Figure 4. The top section (Level 1) shows the original symbolic assignments,

resulting in 712 polysyllabic words for the presented time range; the corresponding dictionary contains 9 definitions. All existing symbols are mapped onto the original signals with different colors. The bottom section (Level 11) shows the result after 11 iteration steps, totaling in 67 words. Each iteration combines the most frequent word combinations. The result exhibits clearly the main operating modes (and pauses between them). The level of complexity is decreased significantly, while the level of abstraction is increased to benefit subsequent analyses. The central part of the figure illustrates all the single iterations (one line per iteration) necessary to achieve the desired level of abstraction; this hierarchical compounding process enables the implicit structure within the data to emerge without using a priori knowledge.

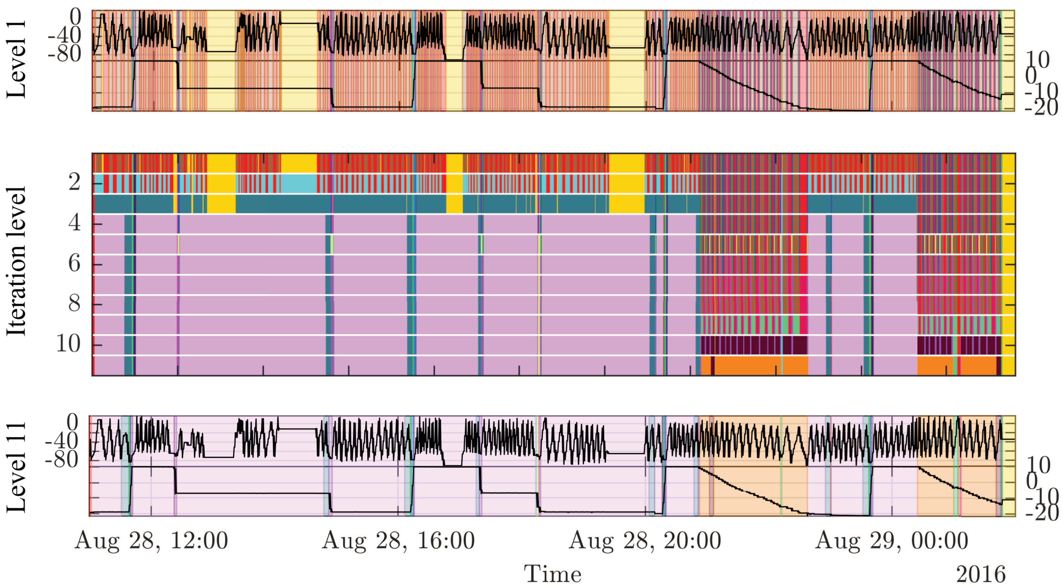


Fig. 4: The top plot represents the initial level (Level 1) with 712 words; the corresponding dictionary contains 9 definitions for the symbolization. The bottom plot shows the result after 11 iterations, with the word count being reduced to 67. The centre part illustrates all steps performed to reach the desired abstraction.

6 Discussion: The Metaphorical Concept of Language and its Relationship to Phenomenology

By introducing the metaphor of language, additional results implicit to the data could be drawn from time series, as the results from observations in the previous section show. This can be interpreted as a pointer to phenomenology.

Although the origins date back to the 18$^{\text{th}}$ century, Edmund Husserl (1859-1938) is considered the founder of the philosophy of Western phenomenology [14].

In simple terms, phenomenological philosophy is about the structures of experiences and consciousness. Husserl describes *experience as the source of all knowledge* [6]. One of his students, Martin Heidegger, also contributed philosophical insights to phenomenology. Heidegger introduced the concept that lived experiences always consist of more than what can be seen [5] — this is considered as a pointer to hidden models, implicit structures within natural experiences. Another student of Husserl, Maurice Merleau-Ponty, extended phenomenology in regard to how we perceive as a result of experiencing phenomena [12]. In quite simple terms it can be concluded that *we experience first and reflect afterwards.*

In contrast to Western philosophy, the Eastern view of phenomenology is provided by the Yogācāra school, which was founded by Asaṅga and Vasubandhu in the 4^{th} century CE. Although this philosophy has had quite some time to mature, it is still considered valid today [11]. Even Heidegger – a representative of Western phenomenology – indicated particular interest in Eastern positions, although pointing out hindrances between Eastern and Western philosophies in his work *A Dialogue on Language between a Japanese and an Inquirer* [4].
Two main concepts of the Yogācāra phenomenology are the *five skandhas* (also known as five aggregates) and the *eight vijñānas* (also known as eight consciousnesses). The concepts support the view of life as a continuous flow of sensory experience to which meaning and significance are added. The skandhas describe how we get from sensory excitation to discursive thinking in five distinct steps, i.e., the five aggregates. This concept is relevant, as it conceptualizes that we are never in direct contact with objects in the world, but we are always in contact with a model of the world. An overview of the skandhas is given in Table 1 along with their English translations and interpretations of meaning in a technical context.

Table 1: The five skandhas and possible technical interpretations [11, 14]

Sanskrit	**English**	**Technical equivalent**
rūpa	Form	Context-dependent sensor information
vedanā	Feeling (Sensation)	Low level model-based control
saṃjñā	Perception	Combination of low level data to identify a situation
saṃskārā	Impulse (Association)	Learned situative semi-autonomous behavior
vijñāna	Consciousness (Discursive thinking)	Artificial reasoning, e.g., rule systems

The second main phenomenological model is the concept of the eight vijñānas. This model relates to the dynamics of our experience on how we establish understanding during the process of emergent consciousness. The first five vijñānas are related to our sensory experiences. The other three are *mano-vijñāna* (modeling sensory experience), *manas-vijñāna* (self-referencing of consequence) and *ālaya-vijñāna*[4] (models for past experience). Language as a metaphorical reference to the content of an experience emerges within these phases as illustrated in Figure 5. Here, language is interpreted as a metaphor for how we understand such an experience. As it emerges throughout the full spectrum of mental models, it can also exhibit variable stages and different strengths of metaphorical references.

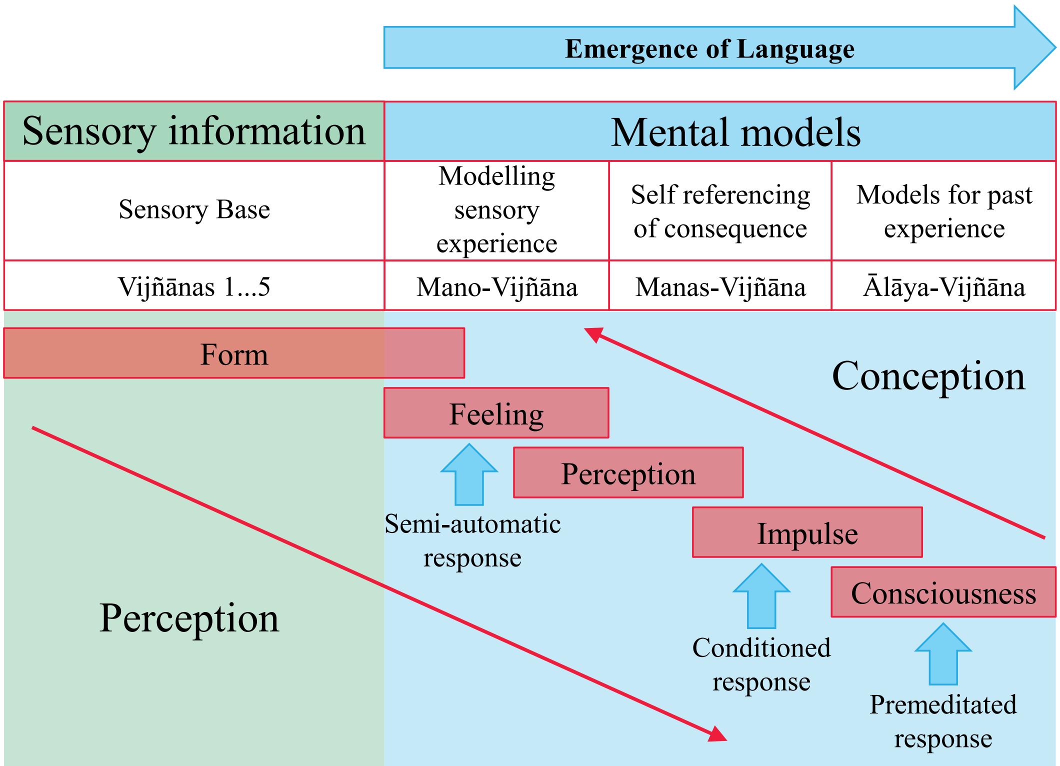


Fig. 5: A schematic overview of a possible interpretation of the eight vijñānas and the five skandhas in regard to the emergence of language. Following this scheme, we can pose a simple example: we hear a sound (*form*); we have a neutral *feeling* about it; we *perceive* it to be an aircraft; it is the plane between Graz and Vienna, just like every day at this very time (*impulse*); we think about whether or not we already checked in for our own flight the next day (*consciousness* or *discursive thinking*).

The different steps in Figure 5 also allow responses along the path from *form* to *consciousness* or vice versa. Examples from everyday life: a child stumbles

[4] In literature, this is often referred to as "store-house consciousness". However, we are careful with translating it into a single English term so as not to neglect any of the semantic nuances as also stated in [18].

besides us and we reach out to prevent falling (*semi-automatic response*); we approach a junction and the car from the left does not decelerate: we are getting slower and more cautious (*conditioned response*); you hurt yourself and you need medical attention, so you make an emergency call (*premeditated response*).

7 Conclusions

The phenomenological model of how humans experience and how language emerge proved valuable for analyzing time series data emanating from physical systems. With the metaphor of language, additional levels of abstraction could be derived from time series data. Implicit structures were found in time series data of physical systems, adding information about the behavior of the system being observed by utilizing the metaphorical nature of those. Modeled system dynamics are significant to the analysis processes for obtaining additional metaphorical, linguistic elements such as verbs, adjectives, and adverbs.

Based on the promising results, it can be concluded that the phenomenological approach and the metaphor of language bear the potential of being of significant relevance to time series analysis of physical systems. At this point in time there may be alternatives to what we have proposed; however, this is not the primary issue in this paper. Our goal is to initiate discussion and further research on completely new approaches to analyzing time series data obtained from the observation of human behavior when operating physical systems. In Asian phenomenology, language is considered to be central as to how humans build mental models for their interaction with the world surrounding them. We feel this – combined with the initial results presented here – justify further discussion and research.

References

1. Beskhyroun, S., Wegner, L.D., Sparling, B.F.: New Methodology for the Application of Vibration-Based Damage Detection Techniques. Structural Control and Health Monitoring (May 2011), n/a–n/a (2011). https://doi.org/10.1002/stc, http://dx.doi.org/10.1002/stc.456
2. Camerra, A., Palpanas, T., Shieh, J., Keogh, E.: iSAX 2.0: Indexing and Mining one Billion Time Series. Proceedings - IEEE International Conference on Data Mining, ICDM pp. 58–67 (2010). https://doi.org/10.1109/ICDM.2010.124
3. Gugg, C., Harker, M., OLeary, P., Rath, G.: An Algebraic Framework for the Real-Time Solution of Inverse Problems on Embedded Systems. In: 2015 IEEE 17th International Conference on High Performance Computing and Communications, 2015 IEEE 7th International Symposium on Cyberspace Safety and Security, and 2015 IEEE 12th International Conference on Embedded Software and Systems. vol. V, pp. 1097–1102. IEEE (aug 2015). https://doi.org/10.1109/HPCC-CSS-ICESS.2015.50, http://ieeexplore.ieee.org/document/7336315/

4. Heidegger, M.: Aus einem Gespräch von der Sprache. Zwischen einem Japaner und einem Fragenden (1959)
5. Heidegger, M.: Sein und Zeit. Max Niemeyer Verlag, 11 edn. (1967)
6. Husserl, E.: Philosophie als strenge Wissenschaft. Logos: Zeitschrift für systematische Philosophie pp. 289–341 (1910)
7. Lakoff, G., Johnson, M.: Metaphors We Live By. The University of Chicago Press (2003)
8. Lanczos, C.: Linear Differential Operators. SIAM (1961), http://epubs.siam.org/doi/pdf/10.1137/1.9781611971187.fm
9. Lin, J., Keogh, E., Lonardi, S., Chiu, B.: A Symbolic Representation of Time Series, with Implications for Streaming Algorithms. In: Proceedings of the 8th ACM SIGMOD Workshop on Research Issues in Data Mining and Knowledge Discovery. pp. 2–11. ACM, New York, NY, USA (2003)
10. Lin, J., Khade, R., Li, Y.: Rotation-Invariant Similarity in Time Series Using Bag-of-Patterns Representation. Journal of Intelligent Information Systems **39**(2), 287–315 (2012). https://doi.org/10.1007/s10844-012-0196-5
11. Lusthaus, D.: Buddhist Phenomenology: A Philosophical Investigation of Yogācāra Buddhism and the Ch'eng Wei-shih Lun. Curzon Critical Studies in Buddhism, Routledge Curzon (2002), http://books.google.at/books?id=IeiwsT-XqwQC
12. Merleau-Ponty, M.: Phenomenology of Perception. Routledge classics, Routledge (2002), http://books.google.at/books?id=oSgaSzvHbaoC
13. Minnen, D., Isbell, C., Essa, I., Starner, T.: Detecting Subdimensional Motifs: An Efficient Algorithm for Generalized Multivariate Pattern Discovery. In: Seventh IEEE International Conference on Data Mining (ICDM 2007). pp. 601–606. IEEE (oct 2007). https://doi.org/10.1109/ICDM.2007.52, http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=4470297 http://ieeexplore.ieee.org/document/4470297/
14. O'Leary, P., Harker, M., Gugg, C.: A Position Paper on: Sensor-Data Analytics in Cyber Physical Systems, from Husserl to Data Mining. In: SensorNets 2015, Le Cresout, France (2015)
15. Senin, P., Lin, J., Wang, X., Oates, T., Gandhi, S., Boedihardjo, A.P., Chen, C., Frankenstein, S.: Time Series Anomaly Discovery with Grammar-Based Compression. In: Edbt. pp. 481–492 (2015). https://doi.org/10.5441/002/edbt.2015.42
16. Senin, P., Malinchik, S.: SAX-VSM: Interpretable Time Series Classification Using SAX and Vector Space Model. In: 2013 IEEE 13th International Conference on Data Mining. pp. 1175–1180. IEEE (dec 2013). https://doi.org/10.1109/ICDM.2013.52, http://ieeexplore.ieee.org/document/6729617/
17. Vespier, U.: Mining Sensor Data from Complex Systems. Ph.D. thesis, Leiden University (2015), http://hdl.handle.net/1887/37027
18. Waldron, W.S.: The Buddhist Unconscious - The Ālaya-Vijñana in the Context of Indian Buddhist Thought. Routledge Curzon