Reliable Information Exchange in IIoT

Investigation into the Role of Data and Data-Driven Modelling

Mehrzad Lavassani

Department of Information Systems and Technology
Mid Sweden University
Licentiate Thesis No. 147
Sundsvall, Sweden
2018
Akademisk avhandling som med tillstånd av Mittuniversitetet framlägges till offentlig granskning för avläggande av teknologie licentiatexamen den 12 Dec 2018 klockan 10:15 i sal L111, Mittuniversitetet Holmgatan 10, Sundsvall. Seminariet kommer att hållas på engelska.

© Mehrzad Lavassani, September 2018

Tryck: Tryckeriet Mittuniversitetet
When wireless is perfectly applied the whole earth will be converted into a huge brain, which in fact it is, all things being particles of a real and rhythmic whole...

- Nikola Tesla

The Cosmic AC said, “There is as yet insufficient data for a meaningful answer.”

- Isaac Asimov, The Last Question
Abstract

The concept of Industrial Internet of Things (IIoT) is the tangible building block for the realisation of the fourth industrial revolution. It should improve productivity, efficiency and reliability of industrial automation systems, leading to revenue growth in industrial scenarios. IIoT needs to encompass various disciplines and technologies to constitute an operable and harmonious system. One essential requirement for a system to exhibit such behaviour is reliable exchange of information. In industrial automation, the information life-cycle starts at the field level, with data collected by sensors, and ends at the enterprise level, where that data is processed into knowledge for business decision making. In IIoT, the process of knowledge discovery is expected to start in the lower layers of the automation hierarchy, and to cover the data exchange between the connected smart objects to perform collaborative tasks.

This thesis aims to assist the comprehension of the processes for information exchange in IIoT-enabled industrial automation- in particular, how reliable exchange of information can be performed by communication systems at field level given an underlying wireless sensor technology, and how data analytics can complement the processes of various levels of the automation hierarchy. Furthermore, this work explores how an IIoT monitoring system can be designed and developed.

The communication reliability is addressed by proposing a redundancy-based medium access control protocol for mission critical applications, and analysing its performance regarding real-time and deterministic delivery. The importance of the data and the benefits of data analytics for various levels of the automation hierarchy are examined by suggesting data-driven methods for visualisation, centralised system modelling and distributed data streams modelling. The design and development of an IIoT monitoring system are addressed by proposing a novel three-layer framework that incorporates wireless sensor, fog, and cloud technologies. Moreover, an IIoT testbed system is developed to realise the proposed framework.

The outcome of this study suggests that redundancy-based mechanisms improve communication reliability. However, they can also introduce drawbacks, such as poor link utilisation and limited scalability, in the context of IIoT. Data-driven methods result in enhanced readability of visualisation, and reduced necessity of the ground truth in system modelling. The results illustrate that distributed modelling can lower the negative effect of the redundancy-based mechanisms on link utilisation, by reducing the up-link traffic. Mathematical analysis reveals that introducing
fog layer in the IIoT framework removes the single point of failure and enhances scalability, while meeting the latency requirements of the monitoring application. Finally, the experiment results shows that the IIoT testbed works adequately and can serve for the future development and deployment of IIoT applications.
Acknowledgements

Firstly, I would like to thank Tingting Zhang for administrating the research project that gave me the opportunity to look at the research challenges from a new perspective. I would also like to thank Mikael Gidlund and Ulf Jennehag for their invaluable guidance and advice throughout the research process. Thank you for your friendship, support and encouragement even during times that I was finding it hard to continue.

Thanks to Leif Olsson for reviewing this work, his constructive comments, and many books he lent me over the years. Thanks to Aamir Mahmood and Stefan Forström for their helpful comments on this thesis, and all the interesting conversations about research, work and life.

Thanks to all the colleagues and fellow PhD students at Information System and Technology department for creating a friendly and enjoyable work environment. Thanks to Annika Berggren, Karl Pettersson, Lena Höijer, Lenart Franked, Magnus Eriksson and Patrik Österberg for their help, advice and interesting Fika conversations.

Thanks to my parents Zohreh and Abbas, my family, and my friends Beignran, Bobby, Dae, Elijs, Jiayi, Jörgen, Lino, Luca, Ran and Ulla not only for their love, care and support, but also for their interest in my research that shaped some of the sentences of this thesis.

And Alireza, you have been my love and my source of inspiration. Your seemingly endless love, wisdom and support brightens my life everyday. You are the one, and without you nothing is ever possible. Thank you for everything!
# Contents

Abstract v  
Acknowledgements vii  
Terminology xiii  

1 Introduction 1  
1.1 Internet of Things and Industrial IoT 1  
   1.1.1 A Paradigm Shift 3  
1.2 IIoT Towards the Next Industrial Revolution 4  
   1.2.1 IIoT and Communication 4  
   1.2.2 IIoT and Data Analytics 5  
1.3 Purpose Statement 6  
1.4 Scope 7  
1.5 Research Goals and Questions 8  
1.6 Research Methodology 9  
1.7 Thesis Organisation and Contributions 11  

2 Communication and control in IIoT 15  
2.1 Communication in Industrial Automation 15  
2.2 Industrial Wireless Sensor Networks 16  
   2.2.1 IWSN Challenges in Industrial Automation 17  
   2.2.2 IWSN Standards 18  
2.3 Communication Reliability in IWSN 18  
   2.3.1 Redundancy Mechanisms and Reliability 19
## CONTENTS

4 An IIoT Monitoring System Framework and Testbed 55
  4.1 Frameworks and Architecture for IIoT Systems ................. 55
  4.2 An IIoT Monitoring Framework .................................... 57
    4.2.1 Sensor Layer .................................................. 57
    4.2.2 Fog Layer ...................................................... 58
    4.2.3 Cloud Layer .................................................... 59
  4.3 The Testbed System Implementation ............................ 59
    4.3.1 Wireless Sensor Layer ......................................... 60
    4.3.2 Fog Computing Layer .......................................... 60
    4.3.3 Cloud Computing Layer ...................................... 61
  4.4 Evaluation and Results ............................................ 61
  4.5 Discussion .......................................................... 63
  4.6 Chapter Summary .................................................. 64

5 Conclusion and Outlook 65
  5.1 Overview and Outcome ............................................. 65
  5.2 Impacts, Social and Ethical Considerations .................... 67
    5.2.1 Impacts .......................................................... 67
    5.2.2 Ethical Considerations ........................................ 67
  5.3 Future Work ....................................................... 68

Bibliography 75
Terminology

Abbreviations

AGNES  AGglomerate NESting
AIC    Akaike Information Criterion
BIC    Baysian Information Criterion
CFEP   Contention Free Emergency Period
CSMA-CA Carrier Sense Multiple Access-Collision Avoidance
EDR    Error Delivery Rate
ESS    Emergency Sub-Slot
GTS    Guaranteed Time Slot
HMM    Hidden Markov Model
IIoT   Industrial Internet of Things
IoT    Internet of Things
IRT    Improved Real-Time
IWSN   Industrial Wireless sensor Network
MAC    Medium Access Control
ML     Machine Learning
MSE    Mean Square Error
PCA    Principal Component Analysis
PDD    Probability Distribution of Delay
PDR    Packet Delivery Rate
QoS    Quality of Service
RFID   Radio Frequency Identification
RMSE   Root Mean Square Error
SHTS   Shared Time Slot
SVM    Support Vector Machine
TDMA   Time Division Multiple Access
WCD    Worst Case Delay
WSN    Wireless Sensor Network
Chapter 1

Introduction

The late 18th century marked the beginning of the first industrial revolution. The power of water and steam was introduced as the driving force for mechanical equipment, and a step towards mechanisation. In the 1870s, by utilising electrical energy, mass production through assembly lines and the second industrial revolution became reality. Advances of information technology and electronics led to the first programmable logic controller (PLC) in the late 60s, and started the third wave of industrial revolution by pursuing automation in industrial production lines. Industrial automation, or automatic control, referred to the technology where factory procedures were carried out without human assistance. In the early days, automation in industry achieved by parallel wiring and point-to-point connection between field devices. The result of this approach was slow and uncomplicated networks with no special components requirements [Car09].

By the mid 70s, and the first generation of microprocessors, digital automation and computation became possible, so to steadily replace the analogue control systems. The development of fieldbus systems was the next step to solve the limitations of point-to-point connections and to replace parallel wiring with a single dedicated network [Tho05, GH13, Ram03, Sau10], also known as field-level networks [Sau10]. Field-level network filled the gap between industrial field devices and the already existing networks of the upper functional levels of process and control applications. This step finalised the development of the well-known automation pyramid.

1.1 Internet of Things and Industrial IoT

In the twilight of the last millennium, Kevin Ashton’s vision that “computers needed to gather their own information by sensing the world for themselves” [Ash15] introduced a new concept to the information and communication technology, and coined the term Internet of Things (IoT). While IoT began with the idea of deploying radio frequency identification (RFID) technology to connect physical objects together, over
the years its focus has expanded and now it is covering all means of communication systems and technologies. There are various definitions of IoT, but in common form it is defined as a network of physical objects [PPP16]. In essence, the idea is a world where all physical objects are connected together, and to the information networks, and can actively participate in various processes. IoT looks at physical objects as intelligent entities, equipped with communication means, which can facilitate data flow and transparency in business models.

In the years that followed, the ambitious vision of IoT not only led to many studies in scientific and academic communities, but also attracted many industrial domains. Considering the overall methodological perspective of IoT, integrating the IoT concept and industrial networks is an attractive choice for industrial processes, where it may optimise operational efficiency, automation, maintenance, and rationalisation [BPL+18]. IoT ensures large-scale interconnection between machines, computers, and people, enabling intelligent industrial operations [BPL+18] by integrating industrial data into all layers of the automation pyramid, and further representing it to the end-users. In other words, IoT can contain various aspects of industrial automation, from field devices and data collection to control and data processing, and even intelligent decision-making in business model. Consequently, Industrial Internet of Things (IIoT) has been introduced as a new term, see Figure 1.1.

IIoT is a new industrial ecosystem that combines intelligent and autonomous machines, advanced predictive analytics, and machine-human collaboration to improve productivity, efficiency and reliability [SWCM17]. Hence, IIoT concerns a wide range of technologies, from communication and networking to big data and data analytics, and cloud computing and smart control [LYD’17, WCZ15] (see Figure 1.2) as the key enabling technologies for adaptation of the IoT vision in the industrial environment and process automation.
1.1 Internet of Things and Industrial IoT

1.1.1 A Paradigm Shift

Since the 80’s technological advances have changed many aspects of industrial automation, from architecture of devices to communication networks. However, none of these transformations has changed the overall perspective or purpose of industrial automation systems. With the rise of the IoT and Cyber Physical Systems (CPS), and the further emergence of IIoT, industrial automation is going through a paradigm change. The objectives are changing from mass production to servisisation and Quality of Service (QoS)-tailored products. The idea of infrastructure connectivity is shifting from local areas to transparency and connection through the Internet. Automation is no longer enough, and technologies are applied to build smart field devices to better utilise industrial data for intelligent business decision-making. The new methodological approaches differ so vastly from conventional purposes of industrial automation that the new wave is being considered the beginning of a new industrial revolution: Industry 4.0 (Figure 1.3).
1.2 IIoT Towards the Next Industrial Revolution

In the down of the fourth industrial revolution, the German government has coined the term "Industry 4.0", reminiscent of software versioning [LFK+14], to describe a high-technology strategy [SZ17]. Industry 4.0 concerns with different aspects of future industries, through digital transformation [Rio17], where IIoT is essentially addressed as one of the enabling blocks for this digital transformation. In other words, Industry 4.0 sets new business models and use-cases, and opens novel opportunities for industries for the years to come. That is where in IIoT focus lies on enhancing the productivity and reliability of communication and control in mission critical applications [Son18] by developing and accelerating the technological advances that fulfil the requirements of such industries.

For Industry 4.0 to become a reality, incorporation of a wide range of disciplines and concepts is needed, as are rapid advances in many technologies. It has been discussed that the vision of Industry 4.0 can only be achieved by extensively applying the existing enabling technologies while actively coping with the technical challenges [WWLZ16, LFK+14]. In particular, Industry 4.0 cannot be achieved without understanding, enhancing, and advancing IIoT’s key enabling technologies.

1.2.1 IIoT and Communication

The overarching goal of industrial automation has been to make the processes more efficient [VPPS16, WSJ17, SSKD11]. The process efficiency to increase revenue has been defined in terms of saving energy and materials, lowering costs, and reducing,
or even eliminating human intervention. The essential requirement for a distributed automation system to work continuously and autonomously, without human intervention, is reliable exchange of information [WSJ17]. From an abstract perspective, the source of information in industrial automation networks is field devices, namely sensors, actuators, and controllers. A reliable flow of data from sensors to controllers, and control commands from controllers to actuators, can potentially ensure continuous and autonomous operation of an automation system. In other words, the successful operation of any automation process depends on well-designed and reliable communication system [WI16], which can facilitate information exchange between the field-level network and the upper level networks.

While industrial automation has undergone many technical changes over the last two decades by adopting new technologies that contributed to its efficiency, the requirements derived from its characteristics have never changed. Indeed, reliability remains the essential requirement of industrial automation that needs to be addressed in communication systems. Reliability is defined in terms of real-time and deterministic data processing and transmission, within pre-set hard deadlines: requirements that need to be met, despite the challenges that communication technologies face in various levels of the automation network, such as unreliable communication links in wireless technologies. Handling these challenges and meeting the requirements become even more critical when communication technologies must handle transmission of the aperiodic events, as well as the periodic sampled data.

Many studies have been carried out to address the reliability requirements of industrial automation networks, and to propose possible solutions to overcome the previously mentioned challenges [LLW+17, KGS15]. The physical and datalink layers have been two of the focal points. In the datalink layer, many protocols have been proposed to move the best effort solutions towards deterministic medium access control (MAC) mechanism [HXS+13, KZ17]. Other solutions suggested handling deterministic delivery of data with redundant routing mechanisms [HDG+15, AKJ15, NRM+17, TNSA17]. More recent approaches [ZWG18, ZK17] suggest the use of software-defined networking for MAC and routing protocols implementation in industrial communication networks to provide lower latency and deterministic delivery.

### 1.2.2 IIoT and Data Analytics

In industrial automation, information flow starts at the field-level network, with the raw data collected from embedded sensors in field devices [WSJ17]. Traditionally, the data life-cycle used to follow a straightforward lane: collect raw data at resource constraint sensors, process for automated control at control level, and store in data warehouses to analyse with powerful centralised computers at the supervisory level for further knowledge discovery and process and product optimisation.

Along with adoption of IoT in industrial automation and advances in sensor technologies, the conventional work flow of collecting, processing and evaluating the data is undergoing tremendous changes. The vision of connecting the physical
and digital worlds is becoming possible by embedding low-cost sensors in industrial equipment and the environment. As a result, a large volume of data become available from various heterogeneous sources in industrial plants, also known as big data. Collected data from field devices are considered valuable source of information, since they provide records about infrastructure and process conditions. This value does not hinge solely on the sheer volume of data under consideration, but rather on the information and knowledge that lies hidden in it [TQLK18]. For a long time, data analytics methods have been applied on raw data to improve efficiency of the industrial systems with enhanced and informed decision-making. However, the traditional analytics methods fall short when facing the overwhelming amount of volatile, unstructured, and heterogeneous raw data collected from sensors embedded in everything and everywhere, in the new transforming industries.

Furthermore, in the context of IIoT, automation systems contain smart things that are expected to collaborate and to perform self-optimisation and self-diagnosis tasks [JBM+17] to facilitate the processes. As a consequence, the well-structured and task-oriented pattern of the automation pyramid is gradually being replaced [JBM+17] by a more distributed model that enables local and dynamic data processing. The new model allows close to real-time response and adaptation of the system to changes of the dynamic environment, and consequently increasing reliability and efficiency of the automation system. This methodological change can potentially expand the role of the embedded sensors: from data collection only, to the first stage of processing, or even to basic evaluation of the data. Previously, limited processing and energy source of embedded sensors were preventing distributed data processing at the sensor level, but advances in sensor technologies, the launch of smart sensors, and the miniaturisation of computing technologies [uRAY+18] are making this task possible.

Recent years have seen a significant interest in the scientific community for utilising data and examining the deployment of computer and data science methods, such as machine learning, data mining, and statistics, for possible performance improvement of industrial automation systems [Qin12, YDXL14, Qin14, GCD15, Ge17, GF17, TQLK18]. Ongoing studies are mainly carried out within two methodologies. One approach investigates deployment of new technologies, such as cloud and fog computing, in industrial systems to enhance data processing and evaluation processes [Gil16] in plant networks, also known as advanced data analytics. The other approach examines the introduction of new distributed algorithms and methods to facilitate information exchange, communication, and collaboration between smart sensors at the field-level network, also known as basic data analytics [YDXL14]. Hence, data analytics is recognised as one of the enabling technologies for IIoT that can substantially increase efficiency and reliability in industrial automation.

1.3 Purpose Statement

The fourth industrial revolution is approaching, and correspondingly changing the ecosystem of industrial automation systems; see Figure 1.4. Solid lines that were separating different levels of the industrial automation pyramid are blurring.
new paradigm demands data transparency and enhanced information exchange for collaboration among smart objects. The value of raw data is being replaced by the importance of information, and the lowest level of the automation pyramid is obtaining complementary data processing functionality. The transformation from conventional industrial automation towards the new automation, and industries architecture in the next industrial revolution are arising exciting opportunities and unique challenges that were difficult to imagine even a decade ago.

The future industries cannot be realised without exercising existing technologies to identify the challenges in the new context, and ensuing innovative approaches to solve the current challenges, and those which are yet to come. IIoT concerns with developing and accelerating the technological advances that respond to challenges of the future industries. The studies presented in this thesis are devoted to investigation of industrial automation system in the context of IIoT. Overall, the purpose is to contribute to a better understanding of data exchange processes in IIoT systems by exploring how a communication system can perform reliable exchange of information, and how data analytics can enhance processes at various levels of the automation hierarchy, in an IIoT framework.

1.4 Scope

Industrial communication systems commonly adopt a layered structure to reduce their complexity. Within this structure, it is primarily the lower layers - the physical layer and the datalink layer - that are important for industrial communications in order to guarantee the performance requirements of specific applications [WI16]. In this work, communication is addressed in the scope of the data Link layer, which corresponds to the link layer in the IIoT connectivity stack model [JDJC17], of an un-
underlying wireless sensor network in industry (IWSN). Reliability as a performance consideration can be examined from a wide range of perspectives. This study defines the communication reliability in terms of real-time and deterministic transmission of data and events. Other communication requirements, such as connectivity, scalability, availability, resilience and security, and alternative technologies for successful implementation of industrial communication systems in IIoT are beyond the scope of this thesis.

Data analytics can enhance the performance of an IIoT system in term of efficiency by integrating data analysis into different functional domains, including various layers of the automation hierarchy. Data analysis methods are utilised for many purposes, roughly categorised into system modelling, anomaly and fault detection, and predictive modelling and prognostics. Supervised and unsupervised techniques have been deployed in the learning pipeline to enhance the process of knowledge discovery and system modelling in the data-driven approaches. This work is not an exhaustive literature study of all data-driven methods, rather an investigation of the role of data in increasing system efficiency, and how to reduce the role of a prior knowledge and expert feedback to autonomously model the industrial system. For this reason, semi-supervised learning approaches on unlabelled data are adopted for data visualisation, distributed data stream modelling and centralised system modelling.

For an IIoT system to work successfully and continuously, harmonious collaboration of various systems and subsystems is required. Moreover, in the design of an IIoT system many application-dependent technological choices and technical concerns need to be addressed. The Industrial Internet Consortium (IIC) maintains foundation frameworks for IIoT architecture, analytics, and connectivity, and provides guidance for development, documentation, communication, and deployment of such systems. The framework presented in this thesis is inspired by mapping the IIoT analytics framework \cite{ADF17} to the IIoT reference architecture \cite{LMD17}, from a functional viewpoint, for a monitoring system. Within the framework, focus lies on an upward data flow, partially covering the control and information domains. The considered processes include data collection and modelling at field network, transmission to the edge and further representation at an application, with the aim of keeping the balance between the number of up-link transmissions and an acceptable level of accuracy in the regenerated data streams by the model parameters utilising fog computing at the edge. Thus, functionalities such as high-level operational design and system modelling, knowledge visualisation, decision-making, and process control, which are relevant to business, operation and application domains, are outside of the scope of this thesis.

1.5 Research Goals and Questions

To realise the main purpose of this study, within its scope, three primary research goals are defined. Correspondingly, sets of questions are formulated to address each of the research goals. These goals and questions are the following.
• **Goal 1**: To investigate mechanisms for reliable exchange of information in the link layer, and to identify challenges, shortcomings, and drawbacks given IWSN as the underlying technology
  
  – **Research question 1.1**: How can a medium access control (MAC) in IWSN guarantee reliability by utilising transmission and link redundancy?
  
  – **Research question 1.2**: What are the drawbacks of the redundancy-based methods, specifically when mapped to IIoT applications?

• **Goal 2**: To investigate the impact of exploiting raw data, and integrating data analytics to the automation hierarchy, in the industrial automation performance.
  
  – **Research question 2.1**: How can a data-driven approach enhance readability of the visualised data collected from a complex system with dynamic behaviour?
  
  – **Research question 2.2**: How can the behaviour of a complex and multimode system be modelled with a centralised data-driven approach without prior knowledge about the nature of the data and the system parameters?
  
  – **Research question 2.3**: How can the behaviour of a data stream be modelled with basic data analytics at sensor level?
  
  – **Research question 2.4**: In which criterion, and to what extent can a system be beneficial by utilising data stream modelling at the sensor level?

• **Goal 3**: To propose an IIoT framework for an industrial monitoring system to study the performance of the proposed data stream modelling in an IIoT system, and to develop a testbed accordingly for future research.
  
  – **Research question 3.1**: Can the proposed data stream modelling at the sensor level (RQ 2.2) be successfully implemented on IEEE 802.15.4 compliant hardware?
  
  – **Research question 3.2**: Can this solution be practically implemented and utilised in a fog computing architecture?

### 1.6 Research Methodology

The research followed a pragmatic approach. It started with a review of the literature, and the identification of research directions and open issues in the research areas surrounding IIoT technology, from a post-positivist perspective. Hence, the efforts put into this research aims to provide a probabilistic but incomplete knowledge about the reality of the complex phenomena of study. The experiments were designed and conducted either on the data collected from field devices, or on data generated from simulations. Quantitative evaluations, such as mathematical analysis, were used to examine the results of simulations or implementation of the proposed solutions, to draw relevant conclusion with respect to the research goals, and
to answer the research questions presented in section 1.5. Figure 1.5 illustrates the order in which this research was conducted.

The initial literature study identified reliable exchange of information by communication systems as one of the main prerequisites for realisation of IIoT in industrial automation systems. Communication systems reliability can be studied from various perspectives, and in different layers of the automation hierarchy. The link layer was chosen as the main focus of this stage of the research. This choice was motivated by the layered architecture of the communication systems and the IIoT communication stack. The link layer is the connection between field devices and upper functional layers, providing data flow from sensors to controllers, and commands to actuators. Thus, it is the first block to provide a reliable exchange of information in industrial automation. Therefore, the first goal and corresponding research questions were formulated with respect to the aforementioned aspect and the scope of this study. This phase of the research was conducted by analysing the data collected from the simulation of a MAC protocol, which utilises both transmission and link redundancy techniques to provide reliability.

The result of this first phase, and an additional review of the literature on data analytics in industrial applications initiated the idea that exploiting raw data and utilising distributed data processing could reduce some of the drawbacks of the redundancy-based methods and improve the performance of communication systems. Hence, it could positively affect reliable and efficient exchange of information in IIoT systems. The second goal was addressed through examining data-driven approaches for centralised data visualisation, system modelling, and distributed data streams modelling. The research question 2.3 was the result of the second literature
study, but the corresponding experiments were conducted partially in parallel with the third goal and research questions. This semi-simultaneous approach was the direct effect of the overall purpose of this research: from the beginning this study was devoted to study of an IIoT system. Therefore distributed modelling was not enough to investigate how the performance of a system could be beneficial from data and data analytics. Thus, an IIoT framework for a monitoring system was designed to place the proposed approaches in perspective. Since the source of information is the sampled data at sensors, a data-driven approach was adopted to develop a modelling method for the data stream at sensor level. With respect to the evaluation criteria, the performance of the method within the proposed IIoT framework was analysed, and compared to a base model through simulation in MATLAB, using the data collected from sensors embedded in an industrial plant.

For further investigation on the performance of the proposed model, under more realistic conditions, it was decided to develop an IIoT testbed system. The testbed is a reflection of the proposed framework with three layers - sensor network, fog computing and cloud computing - for an IIoT monitoring system. It was realised by implementation of the stream modelling method on IEEE 802.15.4 compliant devices in the sensor network layer, reconstruction of the data stream using the model parameters on Raspberry Pi in the fog computing layer, and visualisation of the collected data from fog layer, using Thingboard, in the cloud computing layer. The performance of the model implemented in the testbed was evaluated by mathematical analysis, on the data sampled from experiments run on the testbed.

1.7 Thesis Organisation and Contributions

This thesis studies reliable exchange of information in IIoT, utilising IWSNs and data analytics as two of the IIoT-enabling technologies. The scientific contributions of the studies constitute the chapters of this monograph. Some of the presented results in this monograph have been already published as journal article or conference papers, i.e. Handling Event-Triggered Traffic of Safety and Closed-Loop Control Systems in WSANs [LP14], Pixvid: Capturing Temporal Correlated Changes in Time Series [LLLZ17] and Combining Fog Computing with Sensor Mote Machine Learning for Industrial IoT [LFJZ18], while some are ongoing research for manuscripts under preparation. The research map and corresponding chapters are illustrated in Figure 1.6. The following provides a brief overview to each chapter, and gives a map on how the thesis is structured, and what can be expected from the presented material in each chapter.

Chapter 2
This chapter investigates the reliable exchange of information from communication perspective, and addresses the first research goal and corresponding research questions. It briefly reviews the challenges that IWSN technologies encounter and commonly applied methods to provide reliability. Due to the focus of this study on the link layer, the performance of a MAC protocol that utilises transmission and link redundancy is studied, and the drawbacks of such redundancy-based methods are
discussed. The MAC protocol under study in this chapter has been proposed, and its performance regarding deterministic packet delivery within deadline has been studied in [LBGZ16]. The drawbacks of the proposed method are discussed in this thesis for the first time.

Chapter 3
This chapter studies the role of data and data analytics in the context of IIoT, and addresses the second research goal and corresponding research questions. The potential benefits of exploiting data in IIoT systems are discussed and data-driven approach with complementary methods to enhance performance of various functionalities in automation systems are briefly reviewed. Several centralised methods for transforming the raw data to system insight are suggested, such as visualisation of the data and data-driven system modelling. To address the identified drawbacks of the redundancy-based methods in Chapter 2, a novel distributed data stream modelling is proposed. The method aims to enhance reliable and efficient exchange of information by mitigating some of the identified drawbacks. The contents presented in this chapter are presented in [LFJZ18] and [LLLZ17], and some are the materials for a manuscript under preparation.

Chapter 4
This chapter explores how introducing IIoT can be beneficial to the industrial monitoring systems, and addresses the third research goal and corresponding research questions. A three-layer framework is designed for a monitoring system. The distributed data stream modelling, presented in Chapter 3, is expanded and deployed in the wireless network layer and the middle fog layer of the proposed framework,
and its performance is examined. Furthermore, the chapter reports the development of an IIoT testbed system for realisation of the framework. The methods and results presented in this chapter are based on the materials represented in [LF]Z18.

Chapter 5
This chapter concludes the thesis by summarising the presented research. The outcomes of the study are reviewed and their links to the research goals are illustrated. The chapter also discusses the potential impacts and ethical considerations of this research, along with the ethical issues in the research field. Finally, it draws the path for the future work.
Chapter 2

Communication and control in IIoT

IIoT builds upon the premise that a globally accessible communication infrastructure is available to a plethora of devices involved in industrial processes [RSS+17]. IIoT realisation depends on the collaboration of different communication components, from devices’ local communication within the factory halls, to the applications in cloud services.

This chapter addresses the first goal of this thesis by investigating local communication in IIoT: the field-level network in industrial automation. The role of communication systems in industrial automation is briefly reviewed and the requirements and challenges are identified. After a short introduction on wireless communication technologies, the chapter summarises some of the solutions to achieve the goals and to overcome the challenges in industrial automation domain. Furthermore, it examines the performance of a MAC protocol with reliability guarantees in terms of real-time and deterministic data transmission. Finally, the chapter concludes by identifying the drawbacks of the applied redundancy-based methods. This serves as the foundation and motivation for the research presented in the following chapters.

2.1 Communication in Industrial Automation

The overarching goal of industrial automation has been to make processes more efficient [VPPS16, WSJ17, SSKD11]. The process efficiency to increase revenue has been defined in terms of saving energy and materials, lowering costs, and reducing, or even eliminating human intervention. Reliable exchange of information is the essential requirement for a distributed automation system [WSJ17] to work continuously and autonomously, without human intervention. In other words, the successful operation of any automation system depends on a well-designed and reliable com-
munication system [WI16], which can facilitate information exchange between the field-level network and the upper level networks.

In 90s, advances in information and communication technology introduced new solutions to enhance efficiency in industrial communication. One of the important problems in field-level networks was the fact that the different levels in the automation pyramid were controlled by mutually largely incompatible networking concepts [Sau10]: fieldbus and Ethernet. Widely accepted, cost-effective, and high-performance Ethernet networks were proposed to be incorporated in industrial networks. Ethernet was not only compatible with industrial standards, but also easy to understand, deploy, manage, and maintain. While this solution attracted the interest of the scientific community and researchers, and was widely deployed in commercial networks, its application in industry was slow, limited, and met with scepticism. The main reason for this slow adaption was that Ethernet could not provide essential reliability guarantees, i.e. real-time and deterministic data delivery, that were provided by the fieldbus networks and their communication protocols. The reason for the different levels of reliability guarantees can be found in the fundamentally different requirements of the targeted application domains. Ethernet had matured in commercial networks with various sets of Quality of Service (QoS) considerations that differed from those of industrial networks [GH+13].

Industrial networks’ reliability requirements emphasise real-time and determinism, while they need to handle both periodically sampled data and aperiodic events for safety and alarm conditions in industrial plants. In conventional Ethernet, these requirements are relaxed as the main application domains, i.e. home and office networks, have higher tolerance for failure and delay.

Efforts to make Ethernet more suited for industrial networks resulted in Ethernet-based fieldbus covering all functional levels of industrial networks, except the field-level network. Real-time Ethernet technology improved the real-time quality by utilising the increased data rate of Ethernet and full-duplex Ethernet lines, allowing simultaneous transmission and reception [GH+13]. Later on, new forwarding message techniques also reduced switching delays, which contributed to better real-time response in congested networks. Nevertheless, despite all the benefits of Ethernet, i.e. its function as a unified network for the industrial network, and all the efforts to reduce communication delay, Ethernet failed to be the best solution for the field-level network [Sau10]. This was partly due to the high expenses of Ethernet fieldbusses compared to serial fieldusses with the extra cost of the required hardware, and more importantly the distance limitation of copper Ethernet cables.

2.2 Industrial Wireless Sensor Networks

Another technological advancement that affected communication systems was the evolution of Wireless Sensor Networks (WSN), from military-limited technology [KDM05] to an open technology applicable for commercial use, such as health-care monitoring and building automation. WSNs potentially relieve field devices from
cable constraints and can enhance industrial infrastructure connectivity; without cables, hazardous environments and hard-to-reach areas of industrial infrastructure become accessible. Moreover, as the result of electronic technology efforts, sensor devices have become cheaper and more advanced in terms of functional capabilities. On the basis of the above explanations, adaptation of WSN for industrial automation has become an appealing idea.

However, similar to Ethernet, WSN technology has been developed in commercial environments without considering real-time and determinism guarantees, which are the main QoS requirements in the industrial automation domain [WJ16].

### 2.2.1 IWSN Challenges in Industrial Automation

WSNs posed novel challenges compared to the wired solutions, for adaptation in industrial automation systems. The main design concern in WSNs has been energy efficiency in battery-powered sensors, to extend network lifetime, which contradicts the main concerns of real-time and deterministic data transmission in automation processes. The performance of wireless communication can be highly affected by obstructions and noisy environments, since the attainable capacity of the links depends on the interference level perceived at the receiver [WI16]. Hence, the adverse properties of radio channels, when wireless links are included, also contribute to the difficulty of meeting the reliability requirements in industrial automation.

The reliability requirements of industrial automation, and the challenges posed in communication systems due to the diverse characteristics of wireless links, have prevented deployment of the wireless technology in industrial networks immediately, and to its full potential. Nonetheless, its advantages for improving efficiency have kept Industrial Wireless Sensor Networks (IWSN) an active and interesting research area.
2.2.2 IWSN Standards

The IEEE 802.15.4 standard [IEE06] is the result of the efforts towards IWSN standardisation. It provides an implementation framework for future technological developments. Like all IEEE 802 standards, IEEE 802.15.4 covers up to portion of the data link layer, and higher layers' protocols are open to be utilised for individual applications [CGH+02]. Consequently, IWSN solutions such as WirelessHART [SHM+08], ISA100.11a [ISA], and WIA-PA [ZZPH10] built their comprehensive communication architectures based on the IEEE 802.15.4 physical layer (PHY) specification. In addition, these solutions include mechanisms to reduce latency and increase reliability to meet the requirements of the industrial automation domain. The IEEE 802.15.4e MAC enhanced standard [IEE12] was approved in 2012 to make the IEEE 802.15.4 standard more suitable for the mission critical applications in industrial automation. The amendment is solely dedicated to the MAC enhancement and borrows some of the mechanisms from IWSN solutions, such as time slotted channel hoping from WirelessHART.

2.3 Communication Reliability in IWSN

Mechanisms that increase reliability and reduce latency are primarily implemented on the physical and MAC layer [RSS+17]. As the lowest layer in the communication stack, the PHY is directly affected by the quality of the underlying communication medium, i.e. radio channel. The MAC layer, on the other hand, can provide reliability and low latency by controlling the access to the medium, scheduling and resource management. The functionalities of all the upper layers depend on the services provided by the protocols deployed in the MAC layer, which makes it a primary factor for the overall performance of the network [YIE11].

Depending on the application, designs of the MAC protocols are required to consider demands such as energy efficiency, throughput, transmission reliability, and latency. While energy efficiency and throughput are the concerns in monitoring systems, for mission critical applications the emphasis is on transmission reliability and latency; in other words, real-time and deterministic transmission.

Methods for designing MAC protocols can generally be divided into three main classes: schedule-based protocols, contention-based protocols, and hybrid protocols. The schedule-based protocols, such as Time Division Multiple Access (TDMA), guarantee access to the link and provide deterministic transmission by a pre-set scheduling. As the result of this pre-scheduling, there is no guarantee of real-time communication. In contention-based protocols, such as ALOHA and Carrier Sense Multiple Access with Collision Avoidance (CSMA-CA), access to the link is possible as soon as there is a packet in the buffer for transmission. In this case, real-time transmission cannot be guaranteed since multiple transmission from different sources might be initiated simultaneously, which results in holding back the transmission until the next clear channel assessment. Finally, the hybrid protocols are another attempt to design MAC protocols that overcome the limitations of the other two classes by util-
ising these methods’ benefits.

On the foundation of the above explanations, it is conceivable that none of the conventional design approaches to MAC protocols can fully meet the reliability requirements of industrial automation. For this reason, many complementary mechanisms have been proposed and examined to enhance the communication reliability of IWSNs.

2.3.1 Redundancy Mechanisms and Reliability

Redundancy is one of the commonly practiced mechanisms in IWSNs and WSNs to provide reliability guarantees [CVP+09, RSS+17, Kal18, VS18]. Redundancy initially can be defined as the provision of additional or duplicate resources, which can produce similar results [CVP+09]. Redundancy methods are applied in a wide spectrum of applications, such as energy efficiency [JKM16] and data reliability [VS18], with sometimes contradictory purposes, such as eliminating redundancy [ANM18] or utilising redundancy [VS18].

Redundancy can be generally categorised into two groups: spatial and temporal. Spatial redundancy is concerned with the possibility of obtaining information for a specific location from different sources [CVP+09], either to gain more information, or exclude the unnecessary redundant information. Temporal redundancy can be defined as performing a specific action more than once, skewed in time, to increase reliability [CVP+09]. It is concerned with either sensing or transmission, or both. In sensing redundancy, the focus lies on improving reliability by redundant measurements or reads at sensor level. Transmission redundancy, on the other hand, achieves reliability by retransmission opportunities for the same packet [CV08, SRS09, SBR10, SZBG14], transmission through alternative paths, or link redundancy, such as the solutions reported in [ZGÅ16, PB16, RLMA+17]. Based on the provided explanations, it is easy to infer that MAC protocols in IWSNs can be beneficial from deploying transmission redundancy methods to enhance communication reliability, in terms of deterministic delivery. Although, the same redundancy approach may have its burden on real-time performance.

To investigate how reliability can be improved by applying redundancy, the next section proposes a MAC protocol compliant with IWSN. This protocol utilises transmission redundancy to meet the reliability requirements of a mission critical application in industrial automation.

2.4 Overview of a Deterministic MAC for Aperiodic Events in IWSN

DeMAC [LBGZ16] uses various methods to expand the deployment of the IWSNs to the mission critical applications in industrial automation. The goal is to provide a higher level of reliability, in terms of real-time and deterministic transmission, for
Figure 2.2: The superframe structure of (a) IEEE 802.15.4, (b) an alternative WirelessHART shared sub-slots allocation, and (c) DeMAC contention free structure.

apriodic events. To achieve this goal, DeMAC utilises TDMA-based sub-slots in contention free emergency access periods, group acknowledgement, an alternative packet structure for events, and a relay node as an alternative route for data transmission.

2.4.1 DeMAC Algorithm Overview

DeMAC is a cross-layer protocol. It is based on TDMA so that all nodes in the network have guaranteed access to the medium, and the probability of collision, due to simultaneous transmission is reduced. It introduces a new superframe structure. A superframe might contain several work cycles depending on the application requirements, and each work cycle contains two Contention Free Emergency Periods (CFEP). In each work cycle, a node has one Guaranteed Time Slot (GTS), and one

1. Communication with the Sink:
   
   `if ESS ∈ Framec then`
   
   2. `flag_e = True;`
   3. `Payload ← Event Code;`
   4. `TimerESS = ESS_remaining;`
   5. `TimerACK = DownLink_duration + Downlink_remaining;`
   6. `if TimerESS = 0 then`
      - `Transmit Pet in ESS;`

9. Communication with the Relay Node:

   `if TimerACK = 0 and !ACK then`

10. `while !ACK do`
   11. `Send Pblink to NR;`
   12. `Send Pet with ACK request to NR;`

Nodes generate different packets for periodically sampled data and aperiodic events. Transmission of the periodically sampled data takes place in the GTS. Upon detection of an event, an emergency packet (Pet) is transmitted, either in GTS or ESS. To transmit an emergency packet, a node compares the remaining time to the next GTS and ESS. Pet transmission takes place in the closest transmission opportunity. If ESS is the choice, the node firstly sets a timer (TimerESS) for the remaining time to ESS, and another timer (TimerACK) for the next expected downlink time-slot. When TimerESS is fired, the node transmits Pet in ESS. Not receiving an acknowledgement from sink within the expected time interval, is considered a failed transmission. In this situation node initiates transmission through a relay node. The procedure for emergency packet transmission is summarised in Algorithm 1. The relay node collects all the packets received during one work cycle, aggregates payloads, and sends one packet in its own dedicated timeslot, which is the last GTS of the current work cycle.

2.4.2 Redundancy and Reliability in DeMAC

Based on the explanation in the previous section, it is conceivable that DeMAC makes use of redundancy from the temporal perspective. Transmission redundancy is utilised in the algorithm by allocating several retransmission opportunities for the events data, skewed in time. The proposed superframe structure is tailored to ac-
Table 2.1: Simulation parameters and settings.

<table>
<thead>
<tr>
<th>Standard default</th>
<th>Settings</th>
</tr>
</thead>
<tbody>
<tr>
<td>Frequency band</td>
<td>2.4 GHz</td>
</tr>
<tr>
<td>Transmit rate</td>
<td>250kbps</td>
</tr>
<tr>
<td>RF power</td>
<td>-24 to 0 dBm</td>
</tr>
<tr>
<td>Receive sensitivity</td>
<td>-90dBm</td>
</tr>
<tr>
<td>Time-slot</td>
<td>10ms</td>
</tr>
<tr>
<td>Sub-slot</td>
<td>4ms</td>
</tr>
<tr>
<td>$P_{et}$ data length</td>
<td>1 bytes</td>
</tr>
<tr>
<td>$P_{et}$ size</td>
<td>10 bytes</td>
</tr>
</tbody>
</table>

commodate emergency access periods to the link for the higher priority data. The introduced TDMA-based sub-slots (ESS) provide the nodes with an extra direct data transmission possibility to the sink, in each work cycle. The introduced relay node in the algorithm implies link redundancy in addition to the transmission redundancy. It provides an alternative path for data to be delivered to the sink. Furthermore, the relay node provides the third opportunity for data transmission in each work cycle.

A work cycle initially indicates the validity of the data for a specific time interval; if the data is delivered too late, it is of limited use [ÅGL+13]. Reducing the latency can potentially improve the real-time performance. The altering characteristics of the wireless link in time and space [GH+09] add an uncertainty factor to the success of any transmission. DeMAC utilises redundancy to increase the probability of successful transmission. This approach tries to enhance the reliability performance of IWSN in the case of existing aperiodic events. In other words, DeMAC uses transmission redundancy to improve real-time performance by delivering the data before its value is reduced, and uses link redundancy to improve the deterministic delivery of the data.

Next, the performance of the DeMAC is evaluated. The results assist to investigate the effects of redundancy approaches to enhance communication reliability.

### 2.4.3 Methodology and Evaluation

The DeMAC is implemented in the TOSSIM [LLWC03] emulator, and the sensors’ functionalities are implemented by wiring the component in nesC programming language [GLVB+14], in TinyOS [LMP+05]. TOSSIM emulator has been chosen to create a more realistic experimental setup by adding real-world collected noises from the industrial environment. High-level system events are implemented with no stack modification, so the algorithm stays within the standard, and can be implemented as a supplement to the current wireless technologies in industrial plants. Slot length is set to 10ms, which is the slot duration in WirelessHART. The emergency event generation is random, memory-less, and continuous until the end of the simulation time, i.e. 4500 events. Network behaviour, and the topology are set with a Python program. Radio noise, and interference (-40dBm to -99dBm) are added to the system according to the Closest Pattern Matching (CPM) algorithm [LCL07]. Simulation settings are summarised in Table 2.1.

A large network in industrial automation contains 25-50 nodes [ÅGL+13] with
Table 2.2: The performance comparison of the two algorithms regarding the defined metrics.

<table>
<thead>
<tr>
<th>MAC protocol</th>
<th>PDR (%)</th>
<th>EDR (%)</th>
<th>IRT (%)</th>
<th>WCD (ms)</th>
<th>Average Delay (ms)</th>
</tr>
</thead>
<tbody>
<tr>
<td>DeMAC</td>
<td>79.99</td>
<td>99.52</td>
<td>74.7</td>
<td>380</td>
<td>142.79</td>
</tr>
<tr>
<td>Slotted-ALOHA</td>
<td>22.95</td>
<td>92.87</td>
<td>51.4</td>
<td>448</td>
<td>214.38</td>
</tr>
</tbody>
</table>

preferably direct communication to the sink. Therefore, DeMAC is studied for a star topology network with 29 nodes, where transmission is limited to one channel. This section investigates the performance of the protocol in the following terms:

- **The Worst-Case Delay (WCD)** is defined as the largest value for the end-to-end delay \(d_{e2e}\). It contains processing, queuing, propagation, and transmission delay. Delay due to retransmissions is also considered. Thus, the end-to-end delay for each packet, \(d_{e2e,ij}\), is measured as:

  \[
  d_{e2e,ij} = t_{ij,r} - t_{ij,s},
  \]

  where \(t_{ij,r}\) is the time in which packet \(i\) from node \(j\) is received at the sink, and \(t_{ij,s}\) is the time when the event is sensed at the node. Then WCD can be formulated as:

  \[
  D_{e2e} = \max(P_{et}(d_{e2e,ij})|0 < i \leq E \land 0 < j \leq N_e),
  \]

  where \(D_{e2e}\) indicates the maximum delay, \(N_e\) is the maximum number of nodes, and \(E\) is the total number of events occurring in the studied time interval.

- **Event Delivery Rate (EDR)** is the proportion of the received events, and not the received number of \(P_{et}\), at the sink from all the events detected in the network for one test interval;

  \[
  EDR = \sum e/E \times 100, (0 \leq e \leq E).
  \]

- **Probability Distribution of Delay (PDD)** shows how the end-to-end delay of the successfully transmitted \(P_{et}\) is distributed over one work cycle.

- **Improved real-time performance (IRT)** is the percentage of received \(P_{ei}\s\) that the algorithm delivered within half of the maximum delay boundary.

- **Average Delay** is measured as the mean of the accumulated end-to-end delay of all \(P_{ei}\s\).

To draw a more informative conclusion, the performance of DeMAC is compared to slotted-ALOHA, the contention-based MAC protocol in WirelessHART with the superframe structure shown in Figure 2.2-(b).
2.4.4 Results and Discussion

The results are summarised in Table 2.2. The table compares the performance of DeMAC and slotted-ALOHA for a network with 29 nodes.

The performance in terms of reliable event transmission can be improved by nearly 7%, when redundant link and transmissions are deployed in DeMAC. The temporal redundancy, combined with the dedicated ESS in the TDMA-based contention free emergency periods, can reduce the WCD by more than one forth, namely 26%.

![Figure 2.3: Comparison of the MAC protocols regarding timely delivery of the events.](image)

(a) Probability distribution of delay (PDD) for all received $P_\text{ets}$ over the work cycle.

(b) Relative performance comparison of the algorithms regarding delay; the boxplot shows the median, and whiskers show the best and the worst case delay.
The allocation of the relay node’s dedicated time-slot at the end of each work cycle contributes to a higher event delivery rate, as well as a lower WCD.

Utilising redundant transmission in each work cycle, DeMAC can cut the latency in half for more than two-thirds of the events; that is nearly 25% performance improvement in terms of real-time transmission. This can also be seen in the results presented in Figure 2.3. As a result of the CFEPs allocation with close to equal intervals between each transmission attempt, the delay boundaries can be predicted. Figure 2.3(b) illustrates the impact of temporal redundancy on latency. The average delay can be reduced by 33% when nodes are not required to compete over the shared link or have an alternative path for data transmission.

In terms of reliable packet delivery, the simulation results show that DeMAC can out-perform slotted-ALOHA by delivering three times more packets. Nonetheless, both of the algorithms fall short on meeting the required reliability regarding deterministic transmission and delivery of the periodic packets. It is noteworthy that the poor performance of the two algorithms, aside from the noise model fed to the simulation, is due to different reasons. In slotted-ALOHA, nodes compete over accessing the shared link for event transmission; thus there always exists a high probability of collision because of the simultaneous transmission attempts. On the other hand, DeMAC is a fully TDMA-based algorithm, so collision is the least of its concerns. However, event transmission has a higher priority compared to periodically sampled data, and event data transmission can take place in GTSs. In the presence of events, this algorithm neglects the delivery of the periodically sampled data and the number of transmitted packets will be reduced consequently. In general, DeMAC generates a larger number of packets, regardless of the type, than slotted-ALOHA does. This larger number is the result of the ESS allocation in each work cycle. The number of packets, and consequently transmissions, is proportional to the number of nodes in a network. DeMAC produces almost twice the number of packets that slotted-ALOHA does when events are sensed in the network.

All in all, the redundancy-based approach of DeMAC for reliable transmission of aperiodic events shows promising preliminary results. The communication reliability is improved in terms of real-time and deterministic transmission with low latency, but the method also shows some limitation and drawbacks. The following section reviews some of these limitations and drawbacks and discusses how they are mapped to an IIoT system.

### 2.5 Open Issues and Challenges in IIoT

DeMAC uses TDMA to exclude collision in accessing the shared link, and temporal redundancy to handle real-time transmission. While these methods are effective for deterministic event transmission, they also have some limitations and drawbacks, including link utilisation, peer-to-peer communication, and scalability.

First, although DeMAC assign CFEPs that achieve reliable event transmission, this strategy does not take link utilisation into account. The wireless link is a shared
Communication and control in IIoT

resource, and it needs to be used efficiently, ideally, using 100% of the link capacity. Considering the occasional occurrence of events in the network, it is reasonable to assume that events are sensed and sent in bursts. Therefore, DeMAC shows a poor link utilisation, except when events exist in the network.

The second concern is peer-to-peer communication. While this is not a primary issue in a local automation networks, where a limited number of nodes are in direct communication with a sink in a control network, IIoT builds on the premise of smart things that can exchange information to enhance processes. In factory halls, the machines equipped with embedded sensors are expected to imitate smart behaviour, and they are required to communicate with each other. The concern lies in the limitation of peer-to-peer communication in the TDMA mechanism. The hierarchical organisation of TDMA restricts the nodes to communicate only with the associated sink, and peer-to-peer communication cannot be directly supported [YH04] with a pure TDMA mechanism.

The third concern in DeMAC is scalability. The traffic load of the machine-to-machine communication in IIoT is also going to be added to the shared link. In the IIoT the number of sensors in one system is going to increase exponentially. In an isolated control network of a factory hall, scalability is not an issue, while in an IIoT system, scalability is going to be a prerequisite; this is in addition to the reliability, interoperability, fairness and security requirements. Hence, novel solutions need to be examined to simultaneously provide the required reliability guarantees of the control networks, and fit the characteristics of the automation systems in the IIoT.

The IIoT is concerned with the future industries. To realise this vision, a spectrum of technologies are required to collaborate harmoniously. This opens up the opportunity of looking at the challenges that each discipline faces from a different perspective, examining innovative approaches and deploying multi-disciplinary solutions to overcome the challenges.

2.6 Chapter Summary

This chapter addressed reliable information exchange from the communication perspective. Some of the mechanisms that are commonly deployed for the link layer reliability, were investigated. Moreover, the drawbacks of the mechanisms and the open issues regarding their adoption in IIoT systems were discussed. By doing so, this chapter accomplished the first research goal defined in Section 1.5, and answered the corresponding research questions, RQ 1.1 and RQ 1.2.

The next chapter focuses on the data that is transmitted in communication systems, and how deploying data analytic methods to exploit the data might provide insight into the industrial processes, and enhance information exchange in IIoT systems.
Chapter 3

Big Data and Data Analytics in IIoT

The IIoT pursues the transformation of industrial automation towards the vision of Industry 4.0, facilitating the communication, collection, aggregation, and analysis of industrial plants’ data.

Previous chapter studied reliable local communication in IWSNs, and the importance of the enhanced use of shared resources. In the present chapter, the attention is given to the data, which is the main source of knowledge discovery, and the subject of information exchange. The second research goal and corresponding research questions are addressed through review of the literature study and providing some examples. The characteristics and potential role of the industrial data, specifically in the IIoT, are discussed. Furthermore, the chapter summarises the purposes and solutions for deploying data analytics in industrial automation, and the challenges they encounter. In addition, different application areas of the data-driven approaches are reviewed, and elucidates by studying some proposed solutions. Finally, the chapter concludes by identifying some of the aspects in which the studied data-driven methods can be beneficial for the performance improvement of an IIoT system.

3.1 Data in Industry

The idea of knowledge discovery from industrial data dates back to the early 60s [KK18]. The first attempts focused on adding value to business models and increasing revenues through service and product demand prediction using historical data sets. Henceforth, knowledge discovered from data have been used for cost reduction, quality control, and inventory management in industrial plants [PSBK +96, TQLK18]. Advances in information technologies led to exponential growth in deploying these technologies in industrial systems, starting from information system
for product management, to computer systems for product and process optimisation, and lastly in process automation [TQLK18].

Advent of IoT and the new vision of connecting physical and digital worlds in conjunction with advances in sensor technology, which makes this connection possible, introduced the new concept of Big Data. Initially, Big Data refers to the large volume of data whose size, complexity, and high velocity are beyond the ability of conventional storage and management tools. The characteristics of Big Data were originally summarised as the three Vs: Volume (size of data sets and storage), Variety (data types), and Velocity (speed of incoming data). This list was later expanded to include Veracity (integrity of data) and Validity (correctness and transparency of data) [OJB13, ZLZ+16, Ake14].

3.1.1 Industrial Big Data, Characteristics and Challenges

In industrial automation, Big Data, or Industrial Big Data, refers to the huge amount of heterogeneous data collected from the equipment and environment in field-level network, as well as management, and process and control data, accumulated in higher level networks and data warehouses.

The data collected from the field-level network is considered an important resource in industrial automation. The importance lies in the valuable information that can be extracted from the data after processing and analysing, which can support intelligent decision-making and system flexibility [ZLZ+16]. The data generated from machines and/or collected by sensors in industrial plants have unique characteristics and features, which differ from those mentioned previously. The field-level network data comes from continuous measurements of sensors that, if not interrupted, ideally remains in the same state. Industrial data might project high correlation and it is sensitive to time order [ADF+17]. This data is required to be processed in real-time.

The data is regularly collected with a high sampling rate, and not necessarily with the same rate for all sensors. Missing values and data outliers are two of the well-known characteristics of data sets acquired from sensors. Furthermore, changes in the processes modes and degradation of sensor devices can cause the problem known as data drift [JONK14]. The challenges onset from the deployment of the sensors in industrial plants and in real scenarios. Large volume of data are continuously generated with high frequency by sensors embedded in devices. The data need to be collected, stored, queried, visualised and analysed to provide insight to the system. In general, data sets acquired from industrial equipment and environments demonstrate high complexity, and they project the dynamic behaviour of the underlying system, with abrupt or gradual changes. In other words, the obtained industrial data sets are rarely stationary or independent time series. The characteristics and challenges of industrial data and data sets pose new requirements on the process flow for knowledge discovery and intelligent decision-making, which conventional data analytics cannot fulfil.
3.2 Industrial Data Analytics

Big Data analytics, by and large, refers to the process of acquiring, processing, and analysing raw data, and the tools and techniques that are deployed in this chain of processes [ZLZ+16, WW16]. In general, industrial data analytics frameworks are designed based on the purpose of a system, and the questions that need to be answered. The questions are commonly about the system behaviour: what is happening, why it happened and what will happen next. Accordingly, data analytics frameworks are developed with health assessment, diagnostics and prognostics [ADF+17] purposes. Figure 3.1 illustrates some of the considerations in developing data analytics frameworks for IIoT applications.

Health assessment data analytics, also known as basic data analytics or baseline analytics, refers to the real-time data processing, as well as change and anomaly detection. It is usually carried out locally, on the data acquired from sensors in a dis-
Diagnostics make use of the acquired data and previous knowledge of the system, about normal and faulty states to find the reasons for the abnormal behaviour of the system. Prognostics data analytics utilises various sources of data, from recently sampled data to historical and management data, to make predictions about the system behaviour in the future. The latter two cases are also known as advanced data analytics, and are commonly designed for centralised systems.

3.2.1 Data Analytics Tools

The tools for Big Data are developed to store, manage and analyse large volumes of data in near real-time [DHB+17, KYH+14]. Since Big Data cannot be stored in one machine, the new technologies propose distributed solutions to fulfil the requirements of data analytics systems. The Big Data tools are usually classified based on their approach towards analysing the data: batch analysis, stream analysis, and interactive analysis [DHB+17, RMRESC+16]. In batch analysis, the process is done on the stored data. On-line and near real-time analysis are carried out in tools for stream analysis. Interactive analysis provides users with the opportunity to add extra information, and study the data under those circumstances. Each of the classes is best suited to different applications based on the type of analysis and the requirements of a specific application, such as stream analysis for IWSNs and interactive analysis for management and planning.

Hadoop from Apache Software Foundation, MapReduce from Google, and MOA from Weka project [KWG13] are some of the well-known and well-designed tools for Big Data analytics.

3.2.2 Data Analytics Techniques

Big Data techniques are targeted towards solving system-level problems that cannot be solved with the conventional methods and technologies [ZLZ+16]. With regard to the purpose of the system and data analytics framework, various techniques have been recommended. These techniques either introduce new methods, or adapt traditional statistical data analysis, to provide the abilities required in data analytics to handle the Big Data in industrial automation. Big Data techniques are used to provide insight to the system and contribute to a more efficient knowledge discovery process. Data Mining, Statistics, Machine Learning (ML), Signal Processing and Visualisation [DHB+17] are some of the commonly used techniques.

Data mining is the process of discovering insightful, interesting, and novel patterns, as well as descriptive, understandable, and predictive models from large-scale data [ZMJM14]. Data mining can be defined in three concepts: Big Data, statistics, and machine learning [Rat17, Rat17, DHB+17]. As an interdisciplinary subject, it utilises many statistics methods and machine learning techniques to carry out the tasks such as exploratory data analysis, frequent pattern mining, clustering, and classification [HPK11, ZMJM14]. Data mining tasks are usually done in batch mode,
where some data points are already available before a specific method is applied. Hence, more often than not, data mining is used in centralised systems for knowledge discovery, and not in real-time processing of the live data.

For long, statistical analysis have been facilitating the process of knowledge discovery in industries by evaluating and judging the relationship between the system variables [DHB+17], identifying their correlations, and causality of events occurrences. Many statistical techniques have been adapted to machine learning to improve the performance of classification, and to increase computational efficiency [WFHP16].

The term machine learning was given to the field of study that assigns computers the ability to learn without being explicitly programmed [Rat17]. It is the study of data-driven methods, and statistical techniques to understand, imitate, and enhance human processing. Machine learning uses data to learn, drive, and adapt the system model [Bar12] for assessing, diagnosing, and predicting the system behaviours.

### 3.3 Data Analytics Approaches in Industrial Automation

There are two main approaches for understanding and predicting the system behaviour in industrial automation: Model-driven and Data-driven. The model-driven approach starts with deep knowledge about the complex system, followed by hypothesis formation regarding possible points of failure and the reasons behind them as well as experiments to test the correlation between the parameters, to examine the causation of the events, and to validate the designed model. The system modelling process in the model-driven approach results in a well-established model that can be applied in the system [YDXL14]. However, it is a tiresome task that is done by experts with complete knowledge about the complex system.

#### 3.3.1 Data-Driven Approach

Data-driven approaches have been enabled by the emergence of Big Data, and the machine learning algorithms and techniques. These approaches can extract necessary process information directly from the recorded data, and potentially remove the limitations of model-based approaches in the sense that they do not require a prior knowledge about the system [YDXL14]. Data-driven approaches are able to extract effective relationships between the system variables by processing the data, and analysing the information to monitor, estimate, or predict the system behaviour on-line [Ge17]. A typical data-driven approach follows the steps of data collection, feature extraction, feature design, model training, and model testing, also known as the learning pipeline. However, since data is the key element in the data-driven approach, it inherits most of the previously mentioned challenges that Big Data analytics encounter when dealing with industrial big data. In the literature, it has
also been discussed that an integration of model-driven and data-driven approaches could overcome the limitations of each approach, and improve the efficiency of the system [YDXL14].

In industrial automation, data-driven approaches have been successfully used, as alternative solutions, for applications such as industrial process monitoring [YDXL14, Ge17, WYCS16, YDASH13] and fault detection systems [YWK14, HCT17, MY16, SL14, ZZYW15, JPP+18]. The solutions were not necessarily introduced to replace the model-based systems, but in many cases to enhance the efficiency of a specific functionality of the automation system.

### 3.3.2 Learning Methods

As previously mentioned, machine learning methods and algorithms are used as a technique in data-driven approaches, for system modelling by following the steps of the learning pipeline. Machine learning methods can be classified based on the training data set from which they learn the model parameters. The training data sets either contain both the input parameters and the output values, also known as labelled data, or only the input parameters, called unlabelled data. The methods that train on the former case are known as **supervised learning**, while the latter case is referred to as **unsupervised learning**. The most notable algorithms of supervised learning are Support Vector Machines (SVM), Stochastic Gradient Decent, Nearest Neighbours, Decision Tree and supervised Neural Networks. In unsupervised learning, clustering methods such as K-means, MeanShift [CM99, CM02], and BIRCH [ZRL97], and Outlier Detection, Gaussian Mixture, and unsupervised Neural Networks are the commonly practiced methods.

In general, learning algorithms are categorised as off-line learning and on-line learning. In off-line learning the common learning pipeline is followed; it trains the model over the training data sets, and deploys this model in the system to make predictions about future events. In off-line learning, if the performance of the model is degraded, meaning that the accuracy of the predictions decreases, the model is retrained with new data, or a new set of parameters. If a complete data set is given, the off-line learning can make accurate predictions, though at the cost of losing real-time performance.

In contrast, the on-line learning process starts with each batch of available data, and in each step the model parameters are updated. In other words, in on-line learning the process of retraining the model is replaced by adapting the model to the dynamic changes in the system. The on-line learning predictions are real-time, but vulnerable to anomalies in the data that can make their predictions less accurate by incorrectly updating the model parameters.

Many proposals and complementary solutions have been presented in the literature for each of these classes and categories. The aforementioned algorithms are practiced, solely or in an integration manner, to address the challenges of Big Data in industrial automation, such as visualisation of high dimensional data streams.
[HS04, BAP+05, LKL+04], capturing changes [SH11, PFVR+17, AC17], and system modelling [LWQM17, LIB17, RXL+16]. In data-driven system modelling, the focus is commonly on scalability, computational and accuracy enhancements, or on tailoring the solutions to meet the requirements of a specific application. Recent studies also show an increasing trend towards capturing systematic changes of multivariate processes [Yua15, MBMO16, SPN17, CFF+16] and data-driven modelling of multi-mode systems [ZZGS17, HRBA+18].

The following sections demonstrate how data analytics can be applied on industrial data to facilitate some of the processes and functionalities in industrial automation. In particular, data sets acquired from embedded sensors are investigated to provide insight into the complex processes. A clustering method is proposed to identify correlation between variables, and their simultaneous changes; these correlations are then affiliated with a visualisation system to aid detection of changes in high dimensional time series. A centralised data-driven approach is also presented that employs probabilistic modelling to identify the various functional modes of a system, without having a prior knowledge about the system parameters. Finally, a distributed data-driven approach is employed to adaptively model the behaviour of the data streams.

3.4 Visualisation for Exploratory Data Analysis

Visualisation is an important exploratory method in data analysis, which exposes the characteristics and dynamics of the data set [LCWL14]. As stated previously, one of the challenges for data analytics in IoT-enabled industry is how to handle the temporally ordered and high dimensional Big Data. The main concern in visualisation is communicating the condition of the underlying system without much information loss. Sophisticated methods such as parallel coordinates and pixel-oriented techniques visualise all the dimensions to the user, at the expense of readability; the higher the number of parameters to visualise, the less clarity in the visualised information. In this regard, to improve readability and to reduce the required dimensions to present, methods such as Principal Component Analysis (PCA), and Landmark Multidimensional Scaling are suggested. However, these methods may result in information loss. Many techniques have been proposed to enhance readability of the visualised data without information loss, such as TimeSearcher [HS04], TimeSearcher2 [BAP+05], and VizTree [LKL+04]. These visualisation techniques commonly provide a snapshot of the dimensions in a static figure. This raises the question of whether a static figure is the best way of visualising the dynamics of the underlying system, specifically for the continuous measurements from sensors embedded in an IoT-enabled system.
3.4.1 Visualisation of Temporal Correlated Changes

[LLLZ17] proposes PixVid, a video-based technique to visualise the dynamics of the underlying system. The goal is to enhance readability without information loss when visualising temporal changes in the system. This study improves the pixel-oriented technique [Kei00] so that it can handle large-scale and high-dimensional data. For this purpose, the study proposes a novel clustering algorithm, and constructs a hierarchical cluster tree by ordering the dimensions based on the correlation and distance similarity measures. The work follows two main hypotheses. The first is that video-based visualisation of the continuous and high dimensional data can enhance exploratory analysis by displaying the correlated changes of parameters in the underlying phenomena under study. The second hypothesis is that machine learning methods, in this case a clustering technique, can be utilised to improve the readability of the visualised data.

3.4.2 Proposed Clustering Algorithm

The basic idea is that the most correlated and the closest dimensions should be grouped together. Through iteration, dimensions are grouped into sets, $C_i$, based on their Pearson correlation coefficient. Let $Y$ be a data set with $k$ dimensions and $N$ data points. Each dimension can be defined as $y_i, 0 < i < k$. The correlation between two dimensions $C(y_i, y_j)$ is defined as:

$$C(y_i, y_j) = \frac{\sum y_i y_j - \sum y_i \sum y_j}{\sqrt{(\sum y_i^2 - \sum y_i^2/ N)(\sum y_j^2 - \sum y_j^2/ N)}}$$

(3.1)

The closest neighbours of each dimension are also calculated and are placed in another set $D_i$. The pairwise distance between each pair of dimensions can be calculated as:

$$D(y_i, y_j) = \sqrt{\sum (y_i - y_j)^2}$$

(3.2)

The number of $C_i$s is considered as the threshold of the maximum neighbours that one dimension can have. The intersection of $C_i$ and $D_i$ is the nearest neighbours of the dimensions. Each dimension forms a cluster with its neighbours in the nearest neighbour graph. To avoid multiple assignment of clusters, each dimension belongs to a cluster that contains most of its neighbours.

3.4.3 Evaluation and Results

The proposed algorithm and visualization technique were tested on the data provided in [RCR$^+$10]. The data was collected from body-worn sensors, measuring various home activities. The reason for choosing this data set was two fold; firstly, the data set contains continuous measurements of each activity, and secondly, the dimensions are highly correlated. The data set characteristics are summarised in Table 3.1.
Table 3.1: Statistics of the data sets in PixVid experiments.

<table>
<thead>
<tr>
<th>Data Set</th>
<th>Instances</th>
<th>Dimension</th>
<th>Correlation</th>
</tr>
</thead>
<tbody>
<tr>
<td>A1</td>
<td>51116</td>
<td>230</td>
<td>High</td>
</tr>
<tr>
<td>A2</td>
<td>33273</td>
<td>230</td>
<td>High</td>
</tr>
<tr>
<td>A3</td>
<td>32955</td>
<td>230</td>
<td>High</td>
</tr>
</tbody>
</table>

Figure 3.2: Performance comparison between the proposed clustering algorithm, and distance-based and the correlation-based hierarchical clustering algorithms; correlation and distance are balanced better in the proposed approach.
Figure 3.3: Average time of running each of the algorithms for 30 times on data sets with increasing the number of dimensions.

The performance of the algorithm was evaluated based on goodness of the similarity order of dimensions $Q_{CD}$, and compared to distance-based and correlation-based Agglomerate NESting (AGNES) hierarchical clustering algorithms. The order that has the maximum sum of correlation, $Sum_c$, and minimum sum of distance, $Sum_D$, is considered a good order:

$$Q_{CD} = \frac{Sum_c}{Sum_D}. \quad (3.3)$$

The efficiency of the algorithm was also studied as a function of running time, which is the cost of constructing the cluster tree and generating the order.

The proposed approach that takes into account both distance and correlation of the high dimensional data can improve the performance and efficiency of clustering algorithms. In addition, the performance in terms of goodness of clustering is improved by the proposed clustering algorithm, compared to both distance-based and correlation-based AGNES. This is true for various number of dimensions, as well as different data sets; see Figure 3.2. The proposed algorithm generates a better order and makes a better balance between correlation and distance for ordering the dimensions.

Figure 3.3, illustrates the effect of number of dimensions on running time. The running time is calculated regarding the time spent constructing the cluster tree and generating the order of dimensions, which depends on the number of dimensions in a data set. Since all the data sets in this experiment have the same dimensions, time costs for all the algorithms for each of the data sets are also the same. The proposed algorithm fairly maintains the efficiency when the number of dimensions increases. In contrast, the efficiency of the AGNES algorithms gradually decreases with each added dimension, since they adapt to this change by creating more nodes. The proposed approach reduces this negative effect on the efficiency by constructing the cluster tree at the beginning.
3.5 Data-Driven Multi-Mode System Modelling

One of the goals of data-driven approaches is to reduce or remove the role of expert knowledge in identifying and modelling a complex system. This task is even more challenging when considering the stochastic nature of the complex industrial systems. The data gathered from industrial environments and equipment are not stationary data sets, since they project the dynamics of the underlying systems and various functional levels where a system can operate. The multi-mode nature of the industrial settings has garnered great interest in the scientific community to drive models based on fewer governing states for various applications, such as machinery fault detection [LWQM17], state identification [LIB17], and traffic prediction [RXL+16]. Although this approach can simplify the change detection task by reducing the state-space where the changes can happen, it also raises the question of how these governing states can be identified without a prior expert knowledge about the system, including missing knowledge about the true model and missing labels of associated states for each data point, or set of data points. This question is usually identified as model selection with unlabelled data, partially labelled data, or incomplete observation, and has been one of the main points of concerns in deploying machine learning algorithms for real industrial scenarios, where required level of reliability and real-time guarantees are intensified.

3.5.1 Multi-Mode System Formulation

Let $Y_{1:T} = \{y_1, y_2, ..., y_T\}$ be an observed stream of data generated by a stochastic dynamic system at time $t = 1, 2, ... T$, where each $y_t$ is the joint reading of all features in the system, i.e., $y_t \subset \mathbb{R}^n$; and $X_{1:T} = \{x_1, x_2, ..., x_T\}$ is a set of features or characteristics that the data streams project in various instances of time, due to the effect of an outside factor on the system. The first objective is to identify the state space that the system can obtain - that is, to partition the time into $k$ consecutive and non-overlapping segments $\{t_{1:k}, s_{1:k}\}$, where $t_k$ represents $k^{th}$ segment of time with state
Clusters methods have been deployed to identify different states of a system, where data points in each state are assigned to one cluster. The main problem with such an approach is that in clustering the temporal dependencies between data points - the chronological orders - are not considered. This leads to insufficient information about the transition between states/modes, which is a requirement for continuously functional industrial equipment. It has been shown that the Hidden Markov Model (HMM) is an effective method to identify different states of stochastic systems [Gha01], whose behaviours are governed by some outside factor, or a latent variable.

The HMM is a tool to represent the probability distributions over a sequence of observations [Gha01]. In other words, in a complex system where the stochastic processes are not observable, the HMM models the observations as a probabilistic function of the hidden states, as the name implies. In compact notation an HMM can be defined as \( \lambda = (A, B, \pi) \), where \( \pi = \{\pi_i\} \) is the initial state distribution, \( A = \{a_{ij}\} \) is the state transition probability, and \( B = \{b_j\}, 1 \leq i, j \leq m \) is the probability of the observation in the current hidden state.

Depending on which part of the model or data is given, an HMM can evaluate the probability of an observed sequence belonging to a state, identify the best model describing the observed sequence, or optimise the model parameters that best describe the observed sequence. A comprehensive introduction to the HMM is presented in [Gha01].

### 3.5.2 Model Selection with Unlabelled Data

Since in a data-driven approach there is no knowledge about the true model of a system, the first step should be to find the model that gives the best approximation - that is providing the best fit for the available data; this process is the second problem that can be addressed by the HMM. Each state in the HMM can be associated to a mode in which the system can be operational. In other words, the order of the HMM, i.e. number of states, can be translated into the number of modes of the system.

Many methods have been proposed in the literature to compare model accuracy for various data sets with different characteristics [Aka11, KK08, KL51, Sun74, ZDG01]. Although comparing the likelihood of each of the models on the fitted data seems like an obvious choice, these models suffer from the overfitting problem. In likelihood-based model selection, model parameters are not considered. Thus, increasing the number of states leads to a higher likelihood, which adds to system complexity without providing additional information [BA04]. Therefore, methods that consider the number of model parameters are desirable. Among these methods, the Bayesian Information Criterion (BIC) [KK08] and Akaike’s Information Criterion (AIC) [Aka11] are commonly applied methods for model selection.
Akaike Information Criterion

Model selection using AIC is an optimisation problem to minimise AIC. AIC is an estimator of expected relative information based on the maximised log-likelihood function:

\[
AIC = -2\log(\hat{L}) + 2k,
\]

where \( k \) is the number of estimated parameters in the approximated model. For small sample data where \( \frac{n}{k} \leq 40 \), \( AIC_c \) [Sug78] is calculated as:

\[
AIC_c = -2\log(\hat{L}) + 2k + \frac{2k(k+1)}{(n-k-1)}.
\]

Then, the model selection is the process of fitting the HMM with various orders to the data, and choosing the model with the smallest AIC value:

\[
\hat{r} = \arg \min_{0 \leq k \leq K} AIC(K),
\]

or in case of a small sample size:

\[
\hat{r} = \arg \min_{0 \leq k \leq K} AIC_c(K).
\]

3.5.3 Data-Driven Modelling with Unlabelled Data

In this section, the main idea is to model the system with no previous knowledge. The only assumption is that the system works normally when the first batch of data is acquired. The data set comprises the values collected from 21 sensors resided in a control unit of an industrial machinery, and it is time indexed. The dynamics of the underlying system and the volatility of the data streams are easily apparent when plotted; see Figure 3.5.

The first 250 data points are used for model selection. The rest of the data set is divided into two batches of chronologically ordered data instances, a training set and a testing set. Each data set contains 2,000 time indexes, excluding the time index gaps that indicate the equipment was shut down. The training set generates a list of labels that are used to evaluate the performance of labelling and to make predictions on the states of the data instances in the testing set.

In this approach, the first step is to choose the system model. In the absence of knowledge about the true model and the order, the model with the order that minimises \( AIC_c \) is considered the best approximation. \( AIC_c \) does not have a concrete meaning by itself [BA04]; therefore, instead of choosing one model, a set of models is chosen. The selection condition is to choose consecutive models with a minimum distance from each other, and a maximum distance from the other sets. In this context, distance is defined as the difference between \( AIC_c \) values. The order of the
model is approximated either as the mean value of the order of the selected models, or the order that has the greatest distance from the previous order.

After choosing the model, a Gaussian HMM fits the data and makes predictions on the state for each data point. The associated state with each data point is considered the label for that instance. The data set is then divided, based on their labels, into the groups of data instances with the same label. Subsequently, this new data set is fitted by a classification algorithm and can be applied to make predictions for the newly acquired data instances.

### 3.5.4 Evaluation and Results

In essence, the prediction task is to study the quality of a classification task that associates each data point to the best approximated state. Let \( z = (y, s) \) be the set of predicted pairs, and \( \hat{z} \) the set of true labels - that is the labels learned in training step of this experiment. \( z_s \) is denoted as the subset of \( z \) with label \( s \). The performance of the HMM-based approach was measured with regard to the following terms:

- **Precision** is the ratio of correctly predicted values by the classifier - that is the fraction of instances that have been correctly labelled, \( A \), relative to all the labelled data in the testing data set, \( B \). In other words, it is the probability that
3.5 Data-Driven Multi-Mode System Modelling

Figure 3.6: Model selection by comparing various criterion values of HMMs with different orders; Akaike Information Criterion $AIC_c$, Bayesian Information Criterion $BIC$, and Efficient Determination Criterion $EDC$ [ZDG01].

A randomly selected data point is correctly labelled, $P(A, B) := \frac{|A \cap B|}{|A|}$. For a multi-label classifier, it is:

$$Precision = \frac{1}{|S|} \sum_{s \in S} P(z_s, \hat{z}_s).$$  \hfill (3.8)

- **Recall** or sensitivity is the ratio of correctly predicted labels over the number of labels that should have been labelled. That is the probability of correctly labelled data points that are correctly classified, $R(A, B) := \frac{|A \cap B|}{|B|}$. For a multi-label classifier, it is:

$$Recall = \frac{1}{|S|} \sum_{s \in S} R(z_s, \hat{z}_s).$$  \hfill (3.9)

- **F1-measure** is approximately the average of recall and precision: $F1(A, B) := \frac{2 \times P(A, B) \times R(A, B)}{P(A, B) + R(A, B)}$. In the multi-labelled class, it is:

$$F1 = \frac{1}{|S|} \sum_{s \in S} F1(z_s, \hat{z}_s).$$  \hfill (3.10)

The presented approach was also compared to BIRCH and MeanShift clustering, in terms of accuracy, mean error of prediction, and running time.

Figure 3.6 illustrates how the $AIC_c$ value is affected by the order of the model - the number of possible modes or states, ranging from 2 to 10, in the studied data set. All the criteria display almost the same behaviour. The results show a more accurate model when the order increases, with sharp falls and steady behaviour between
some of the orders. Considering the value of $AIC_c$, it is clear that the most information gain happens with the third order HMM for this specific data set. That same is observed for the BIC and ECD. The models with higher order add to the complexity of the model without providing significant information gain.

The training data was fitted to a third order HMM, and the list of labels was generated. In Figure 3.8, data instances are associated with colours to distinguish different modes. The transition probability, mean, and variance of each state were then used to predict the labels on the testing data. Table 3.2 summarises the performance of the model on the unseen data.

Table 3.2: Performance of the algorithm regarding various classification metrics.

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>Precision</th>
<th>Recall</th>
<th>F1-score</th>
</tr>
</thead>
<tbody>
<tr>
<td>HMM</td>
<td>0.77</td>
<td>0.77</td>
<td>0.77</td>
</tr>
<tr>
<td>MeanShift</td>
<td>0.75</td>
<td>0.77</td>
<td>0.76</td>
</tr>
<tr>
<td>BIRCH</td>
<td>0.68</td>
<td>0.69</td>
<td>0.68</td>
</tr>
</tbody>
</table>

The third order HMM can improve overall accuracy, and reduce the mean error of the predictions compared to the MeanShift and BIRCH clustering algorithms; see Figure 3.7. The HMM is also more efficient compared to the other algorithms: it decreases the running time by more than two third; see Figure 3.7-(a).
Figure 3.8: The result of state-mode identification by the algorithms on the data set. Each state is distinguished by a different colour. To enhance readability, only three of the data streams are shown; the column to the left, (a)-(d)-(g), Stream A; the column in the middle (b)-(e)-(h) Stream B; and the right column, (c)-(f)-(i), Stream C.
3.6 Distributed and Adaptive Data-Driven Modelling

Data acquisition has been one of the main tasks of data analytics [LAYB15]. The field-level network in industrial automation is the origin of the data. The raw data collected by sensors needs to be transmitted to upper level networks to be transformed into information for knowledge discovery, such as in monitoring systems. The emergence of IIoT is changing the ecosystem of industrial automation by demanding enhanced information exchange, and advances in sensor technology provide the possibility of placing raw data processing closer to the source of data. The limited functionality of the sensors - only to collect and to transmit the data over the communication link - is expanding to smart sensing and performing basic data analysis. Consequently, recent researches show interest in studying various aspects in which the new functionalities could be beneficial for industrial automation in the context of IIoT.

IWSN is one of the building blocks of the IIoT. Two of the main concerns in IWSNs are resource management, such as energy efficiency of sensor devices, and efficient utilisation of the shared resources, namely the communication link. The main source of energy consumption at sensor devices is radio communication [Som17]. In the literature, this problem has been addressed either through duty cycle management [ODC+16, CMMS17, HLWH18], or energy-aware routing algorithms [AKJ15, HDG+15, NSM+17]. In the former case, MAC protocols allow the nodes to stay in sleep mode most of the time to extend the battery life. The latter case tries to find the closest routing path with lower costs to carry out the transmissions. These approaches positively contribute to the energy saving problem but commonly fail to meet the real-time requirements of the industrial applications.

The communication link is a shared resource that needs to be efficiently utilised to meet the scalability requirements of IWSNs. A large number of devices use the communication link to transmit their packets, and to communicate with the sink or each other. The load on the link increases with the re-transmission traffic of failed transmissions and packet loss due to coexistence with other wireless technologies on the communication link.

Recent studies attempt to tackle the aforementioned problems by utilising machine learning and data mining methods. Innovative approaches try to tailor the traditional methods to suit the requirements of industrial automation, or include the prerequisite of communication systems and the IIoT in the design phase. Nevertheless, the main body of proposed solutions contains limiting assumptions, such as the existence of a prior knowledge about the system, or availability of clean data to learn the system behaviour.

3.6.1 Data-Driven and Event-Based Communication

[LFJZ18] proposed a data streams learning and monitoring model for distributed data-driven modelling. A data-driven approach aims to reduce the up-link traffic without a prior knowledge about the system for two purposes: saving energy and
efficient link utilisation.

The reduction of up-link traffic can be justified through information redundancy, which is a characteristic of WSNs. In industrial scenarios the up-link traffic is generated by packet transmissions from sensors with a high sampling rate. The real-time and deterministic requirements of mission critical applications in industrial automation, specifically for unpredicted events, justify the high sampling rate, but this does not translate to a higher value of the collected data. More often than not, sensors transmit repetitive measurements, with almost the same value, each time they access the shared link. Moreover, numerous sensors are deployed on factory floors, with overlapping coverage area: this means that the packets transmitted by some of the sensors might not add any value to the exchanged information. The aforementioned redundancies are two of the causes of unnecessary energy consumption, and inefficient link utilisation by transmitting unnecessary packets and adding extra load to the up-link traffic without any information gain.

In the proposed approach, raw value transmissions are replaced by transmission of the model parameters that describe the data streams. A sensor learns the initial model of the data stream, and in each transmission turn, it only transmits the updated parameters if any change has been detected: that is, if the prediction error has passed a pre-set threshold. If no change has been detected, the sensor releases the link and skips the current transmission turn.

### 3.6.2 Distributed Learning and Modelling

Let $X = \{x_0, x_1, x_2, \ldots, x_T\}$ be a data stream sampled at uniform and ordered time points $t = 0, \ldots, T$. Then any time point $k$, $0 < k < t$, of a polynomial temporally correlated data stream with respect to a starting point $t_0$, can be represented as:

$$f(t) = \sum_{i=0}^{k} a_i(t_i - t_0) + d(t_0) + \epsilon;$$  \hspace{1cm} (3.11)

where $a_i$ and $d$ are constant coefficients describing the characteristics of the stream, slope and intercept respectively, and $\epsilon$ is a small random value with standard normal distribution.

The sensor first learns the initial model of the data stream, and then updates the model parameters with each sampling. Hence, the sensor operation is divided into an initialisation and a monitoring phase.

#### Initialisation Phase

The initialisation phase starts with a sensor filling a segment with data points, which are the measured values from the monitored phenomena. The maximum length of each segment, $m$, is defined as a maximum time interval acceptable for data transmission by the sensor based on the requirement of a specific application. For each
segment, the sensor extracts a set of statistical information, \( \zeta = (A_{ref}, \text{sum}X, \text{surprise}X, \text{step}X) \), from the normalised collected values to set the parameters of the initial regression model.

**Algorithm 2: Initialisation Phase.**

1. **Initialisation Phase**
   - **Data:** Sensor Values.
   - **Result:** Initial Model Parameters.

2. **Set:**
   - Starting time \( t_0 \); counter \( n = 0 \); and segment length = \( m \)
3. **while** \( n < m \) **do**
   - Read sensor value;
   - Save value in list;
   - \( n += 1 \);
4. **Find** min and max value in the list;
5. **Normalise** sensor values in the list, \( x_0, ... , x_{m-1} \);
6. **Set:**
   - trend, \( A_{ref} = (x_{m-1} - x_0)/m \); segment’s first value, \( x_{ref} = x_0 \);
   - \( n = m \) and \( \text{SendingTime} = m - 1 \);
7. **sum}X = \sum_{i=0}^{m-1} x_i \); \( \text{surprise}X = \sum_{i=0}^{m-1} x_i^2 \);
8. **step}X = \sum_{i=0}^{m-1} ix_i \);
9. **Send to sink:**
   - \( t_0 \), \( x_{ref} \), and \( A_{ref} \).

The reference values are defined as follows.

The reference slope:

\[
A_{ref} = \frac{(x_{m-1} - x_0)}{m};
\]

the sum of the values:

\[
\text{sum}X = \sum_{i=0}^{m-1} x_i;
\]

the second moment (surprise number [LRU14]):

\[
\text{surprise}X = \sum_{i=0}^{m-1} x_i^2;
\]

and the step is calculated as:

\[
\text{step}X = \sum_{i=0}^{m-1} ix_i;
\]
where \( x_i, 0 < i < m \) is the data point in the \( i^{th} \) place of the segment. At the end of the initialisation phase the sensors send one packet with the essential parameters of the initial model to the sink. These parameters are the reference slope \( A_{ref} \), the first value of the segment \( x_0 \), and the beginning time point of the segment \( t_0 \). Algorithm 2 summarises the initialisation phase.

**Monitoring Phase**

The monitoring phase is an on-line learning procedure conducted by the sensor; it is summarised in Algorithm 3.

**Algorithm 3: Monitoring Phase.**

1. **Monitoring Phase**

   **Data:** Sensor Value.

   **Result:** Updated Model Parameters.

2. Read Sensor Values;

3. Using min and max values of initialisation phase, normalise sensor value, \( x_n \);

4. **Update:**

   \[
   \begin{align*}
   \text{sum}X &= \text{sum}X + x_n; \\
   \text{surprise}X &= \text{surprise}X + x_n^2; \\
   \text{step}X &= \text{step}X + nx_n; \\
   \end{align*}
   \]

5. **Set:**

   \[
   A = A_{ref} - \frac{1}{n} \left( \frac{x_n - x_{n-m+1}}{m} \right);
   \]

6. if \( n - \text{SendingTime} > S \) then

7. if Prediction based on \( A_{ref} > \theta \) then

   A trend may have occurred;

   **Set:**

   SendingTime = n;

8. if Prediction based on \( A < \theta \) then

   **Update:**

   \[
   A_{ref} = A;
   \]

   **Send to sink in next packet:**

   \( A_{ref}, n - 1, \text{sum}X, \text{surprise}X, \text{step}X \);

9. else

   End the current segment;

   **Update:**

   \[
   x_{ref} = \frac{\text{sum}X}{2} - \frac{n+1}{2} A;
   \]

   **Send to sink node:**

   \( x_{ref}, A_{ref}, n - 1, \text{sum}X, \text{surprise}X, \text{step}X \);

10. Go to Initialisation Phase.
At each time point, the sensor reads the new value, updates the statistical information, and predicts the value and the trend of the next time point, \( n + 1 \). Since sensors have limited resources in terms of memory and processing, instead of iterating over all the values in a segment, a step-wise process is used to update the statistical information. The values are updated as follows:

\[
\begin{align*}
\text{sum}X &= \text{sum}X + x_n, \\
\text{surprise}X &= \text{surprise}X + x^2_n, \\
\text{step}X &= \text{step}X + nx_n.
\end{align*}
\]

Considering equation 3.11, the next value of the \( l \)th segment can be approximated as:

\[
f_l(t_{n+1}) \approx f_{l-1}(t_{n+1}) + f_{l-1}(t_n).
\]

The trend of the \( l \)th segment, \( A_l \), is calculated as:

\[
A_l = E[f_l(t_n)],
\]

where \( E[\cdot] \) is the average function, and \( A_l \) is the segment’s trend.

The accuracy of the prediction is evaluated by calculating the Root Square Error (RSE). A comparison between a pre-set error threshold, \( \theta \), and the calculated RSE for the recent data point indicates whether the data stream is stable or there is a trend. In the stable condition, the sensor updates the model parameters, and at the end of the segment sends a packet with statistical information, \( \zeta(A_l, x_0, n_l, \text{sum}X_l, \text{surprise}X_l, \text{step}X_l) \), to the sink.

The trend in the system is distinguishable from switches between modes, by comparing the prediction error of the current segment trend to the prediction error with respect to the reference slope. In both condition of detecting trend and mode switch, the sensor ends the current segment. When a trend is detected, the sensor sends the packet to the sink with the updated model parameters. The mode switch results in transmission of statistical information about the current segment to the sink, and the start of the initialisation phase to identify the new parameters of the model that represent the current mode.

### 3.6.3 Model Aggregation Process

The statistical information about each segment sent by sensors needs to be aggregated to form an overall view of the system modes, and to make the differentiation between trends and mode switches possible.
Given the summary information received by the sensors for each segment, the sink estimates the trend of the $i^{th}$ segment,

$$A_i = \frac{2 \cdot stepX - i(n_i + 1)}{2\left(\frac{n_i^2}{3} + \frac{n_i}{2} + \frac{1}{6}\right)} ,$$

and the Mean Square Error (MSE) of the prediction:

$$\sqrt{\frac{surprise_X}{n_i} - 2x_0 \cdot \frac{sum_X}{n_i} - \frac{2A_i \cdot stepX + x_0^2}{n_i} + (n_i + 1)A_i + A_i^2 \left(\frac{n_i^2}{3} + \frac{n_i}{2} + \frac{1}{6}\right)} .$$

The sink merges the segments by comparing the prediction error of the new trend $A_i$ with a pre-set and acceptable error threshold, $\sigma$. In other words, the segments $i_{start}$ to $i_{end}$ can be merged if the following condition is true:

$$\sqrt{\frac{surprise_X}{n_i} - 2x_0 \cdot \frac{sum_X}{n_i} + x_0^2 - \frac{(2 \cdot stepX - (n_i + 1))^2}{4\left(\frac{n_i^2}{3} + \frac{n_i}{2} + \frac{1}{6}\right)}} < \delta .$$

(3.20)
Figure 3.9: The model MSE measure comparison with respect to the segment length and the trend threshold.

Table 3.3: The mean square error of different settings for segment length ($m$) and trend threshold ($\theta$).

<table>
<thead>
<tr>
<th>$\theta / m$</th>
<th>Stream A</th>
<th>Stream B</th>
<th>Stream C</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.005</td>
<td>0.03</td>
<td>0.03</td>
<td>0.03</td>
</tr>
<tr>
<td>0.01</td>
<td>0.04</td>
<td>0.04</td>
<td>0.05</td>
</tr>
<tr>
<td>0.02</td>
<td>0.03</td>
<td>0.10</td>
<td>0.04</td>
</tr>
<tr>
<td>0.03</td>
<td>0.09</td>
<td>0.03</td>
<td>0.03</td>
</tr>
<tr>
<td>0.05</td>
<td>0.05</td>
<td>0.05</td>
<td>0.05</td>
</tr>
</tbody>
</table>

Table 3.4: The mean square error of sending the average value with different segment length.

<table>
<thead>
<tr>
<th>Stream / $m$</th>
<th>16</th>
<th>17</th>
<th>18</th>
<th>19</th>
<th>20</th>
<th>21</th>
<th>22</th>
<th>23</th>
<th>24</th>
<th>25</th>
<th>26</th>
<th>27</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>0.038</td>
<td>0.039</td>
<td>0.04</td>
<td>0.04</td>
<td>0.043</td>
<td>0.043</td>
<td>0.045</td>
<td>0.046</td>
<td>0.046</td>
<td>0.049</td>
<td>0.05</td>
<td>0.05</td>
</tr>
<tr>
<td>B</td>
<td>0.024</td>
<td>0.024</td>
<td>0.024</td>
<td>0.024</td>
<td>0.025</td>
<td>0.025</td>
<td>0.025</td>
<td>0.025</td>
<td>0.025</td>
<td>0.025</td>
<td>0.026</td>
<td>0.026</td>
</tr>
<tr>
<td>C</td>
<td>0.017</td>
<td>0.018</td>
<td>0.018</td>
<td>0.019</td>
<td>0.019</td>
<td>0.02</td>
<td>0.020</td>
<td>0.020</td>
<td>0.021</td>
<td>0.022</td>
<td>0.022</td>
<td>0.023</td>
</tr>
</tbody>
</table>
3.6.4 Evaluation and Results

The evaluation of the presented model was done through mathematical analysis after implementing it in MATLAB. The performance of the model was studied on 40,000 data points from three data streams with different characteristics, chosen from the data set introduced in Section 3.5.3. The effects of the segment length and threshold on the accuracy of the predictions were examined to choose appropriate values for the experiment. The performance of the model was evaluated in terms of the reduced number of packet transmissions, and the accuracy of the reconstructed data streams using the received model parameters. To describe average model-performance, the difference between the simulated data stream at the sink and the original data stream was measured by $MSE$; this was when the sink was only provided with the statistical information received from the sensor. The model was also compared to the moving average model, as the base model that transmits the average value of each segment. The comparison was based on their relative errors in simulating the data streams with similar parameter settings, and the reduced number of transmissions.

The segment length $m$ and threshold $\theta$ are the most important parameters of the proposed model, since they indicate the trade-off between accuracy and energy efficiency. Table 3.3 summarises the results of the different threshold values and the segment length on the $MSE$. Figure 3.9 shows that the model has the best performance with the tuple $\theta = 0.005$, $m = 10$, by minimising the $MSE$. The performance does not show much improvement, or any significant decline, when the segment length increases from 10 to 30. On the other hand, although the $\theta$ value of 0.005, shows the lowest $MSE$, $\theta = 0.01$ produces the most stable $MSE$, when the segment length is increased. Accordingly, the tuple $\theta = 0.01$, $m = 20$ was set for the experiment.

The tuple was set in the model to study the accuracy of the regenerated model at the sink; see Figure 3.10. The performance of the model was set by the acceptable $MSE$, and it can be seen that the regenerated data streams are fairly comparable to the original data stream. The model also regenerated the data streams with a transmission reduction ratio of 49 to 1, 45 to 1, and 43 to 1 for data streams A, B, and C, respectively. In total, approximately 2.2% of the original number of packets was sent - that is 40,000 data points for each of the data streams.

Table 3.4 summarises the performance of the moving average model in terms of $MSE$, with different segment lengths. Comparing the accuracy of the regenerated data stream between the two models with the tuple $\theta = 0.005$, $m = 20$, the proposed model regenerated A, B, and C with an $MSE$ of 0.036, 0.022, and 0.017, respectively, where the values were A = 0.43, B = 0.025 and C = 0.019 for the base model.

In case of a reduced number of transmitted packets, the proposed model regenerated A, B and C with an $MSE$ of 0.038, 0.024, and 0.02, respectively, by transmitting one packet every 20 data points. For the base model to achieve the same level of accuracy, the maximum segment length cannot be more than 17 for stream A, 19 for stream B, and 23 for stream C. This means that approximately 15.5% of all the packets need to be sent, which makes the communication cost of the base model 7.5 times
higher than that of the proposed model.

The performance evaluation of the distributed modelling clearly shows reduced communication, which indicates energy saving and reduced traffic load on the link. Providing sensors with transmission opportunities when detecting a change point instead of limiting them to a schedule-based packet transmission can contribute to a more efficient link utilisation by eliminating the traffic load of the periodic packets with low information gain.

Based on the presented results and discussions, it is easy to infer that data analytics can be considered an effective technique to facilitate the information exchange of an industrial automation system, in the context of IIoT. The accuracy of the regenerated data streams at the sink shows how basic data analysis can be deployed to start the knowledge discovery process from the field-level network, and replace raw data transmission by information exchange. Moreover, as a result of reducing the traffic of the periodically transmitted packets, the shared communication channel can be utilised in a more efficient way to accommodate the traffic of peer-to-peer communication between devices.

3.7 Chapter Summary

This chapter investigated the role of data and data analytics in enhancing the performance of industrial automation systems, and thereby achieved the second goal of this thesis. Various data sets were analysed to provide additional insight into different layers of the automation hierarchy. Research question RQ 2.1 was answered by proposing a clustering method that can enhance visualisation readability for volatile behaviour of a system. The second research question RQ 2.2 was answered by suggesting a data-driven approach that can reduce the requirement of expert intervention, and the necessity of the ground truth for centralised modelling of a multi-mode system. Finally, a novel distributed data streams modelling was proposed to answer the last two research questions, RQ 2.3 - RQ 2.4, related to the second goal. The model successfully captured and predicted the volatile behaviour of the data streams with an acceptable accuracy. It also revealed that utilising data processing in the lower layer of the automation hierarchy can eliminate unnecessary up-link transmissions of the less-informative packets, and consequently contribute to better link utilisation and communication efficiency.

The following chapter addresses the IIoT from a system perspective. It presents an IIoT framework for an industrial monitoring system. The proposed distributed data streams modelling constitutes the framework’s lower layer, and it is implemented on hardware in the developed testbed.
Figure 3.10: Comparisons between normalised sensor data (blue) and the simulated sensor stream (orange). The top row (a)-(b)-(c) shows the performance of the proposed method for the duration of the experiment (40,000 samples) on different data streams. To illustrate the results in more detail, the bottom row (d)-(e)-(f) shows a zoomed in view of a shorter interval (1,200 samples on the same data streams).
Chapter 4

An IIoT Monitoring System Framework and Testbed

To realise a coherent industrial system, the IIoT needs to orchestrate a wide range of technologies, such as communication and networking, data analytics, and cloud computing and smart control [LYD+17, WCZ15].

The previous chapters considered local communication and data analytics as two of the enabling blocks of the IIoT. This chapter addresses the third research goal and corresponding research questions by investigating the possible benefits that the adoption of an IIoT framework can introduce to a monitoring system. Design considerations for developing an IIoT framework are briefly reviewed. An IIoT framework is designed, and its performance and applicability are studied through implementation in the developed testbed. The chapter concludes with a discussion of how the findings of the experiment answer the research questions, and how the research goal is achieved.

4.1 Frameworks and Architecture for IIoT Systems

An important factor in developing any complex system is to identify the components and to know how these components interact. An IIoT system consists of various enabling blocks from a wide range of technologies. It is a complex system that demands interoperable processes, and harmonious collaboration between various technologies and elements; see Figure 1.1. The Industrial Internet Consortium (IIC) created and maintains the Industrial Internet Reference Architecture (IIRA) [LMD+17] as the foundational framework to guide and assist the deployment of IIoT systems. The IIRA is complemented by technical reports that address the IIoT analytics framework [ADF+17] and the IIoT connectivity framework [JDJC17]. By analysing various IIoT use-cases, and identifying the relevant purposes that can be pursued, each
Figure 4.1: Analytics mapping to the IIoT Reference Architecture [ADF+ 17].

of these frameworks is studied from four viewpoints: business, usage, functional and implementation. The business viewpoint concerns the business vision, values, and objectives of deploying the IIoT system and expected capabilities. How these capabilities can be achieved through some functionalities are addressed in the usage viewpoint. The functional viewpoint focuses on functional components in IIoT system and their interactions within system components and external elements. The technologies that are required to implement the functional components, their communication schemes and their life-cycle procedure are addressed in the implementation viewpoint [LMD+17, ADF+17, JDJC17].

There are several concerns in terms of real-time performance, energy efficiency, scalability, security and interoperability when it comes to facilitating the IIoT for the new generation of automation systems, with an emphasis on real-time and system safety [Del17, MAASA18]. Advances in cloud computing introduced the new concept of cloud manufacturing to bring the benefits of cloud-based services - such as flexibility, convenient and on-demand network access to a shared pool of configurable resources - to the industrial and manufacturing halls [Xu12]. Consequently, the main body of IoT solutions suggested for industrial scenarios [WWS+16, LZN17, TCDX+14] follow the traditional client-server architecture with a cloud back-end. The data collected from devices in factory halls are sent directly to the cloud, and then accessed by user applications attached to the cloud for visualisation, data analysis, and decision-making. It is comprehensible that in this approach the cloud servers become the single point of failure. Moreover, the cloud back-end introduces excessive delay, when the cloud server is located far away on the Internet. One of the solutions that has recently gained the interest of the scientific community is to utilise
fog technology to overcome the aforementioned challenges [BMZA12, AZH18]. The fog servers can be located closer to the edge of the network and act as a local cloud system. The local fog layers can potentially contribute to the IIoT system efficiency by locally handling some of the computational and analytics tasks, thus improving the real-time performance.

To investigate the premises of the fog layer in IIoT to reduce the end-to-end delay, and to examine the performance of the distributed data modelling to decrease up-link transmission load on the shared link, the following sections present a framework for an industrial monitoring system, and the realisation of the framework in a developed testbed.

4.2 An IIoT Monitoring Framework

[LFJZ18] proposes an IIoT framework for an industrial monitoring system. The framework is the outcome of a crosscutting design between functional viewpoints of IIoT reference architecture and IIoT data analytics framework; see Figure 4.1. Hence, data analytics techniques are integrated into the framework as complementary functionalities. The aim is to reduce unnecessary up-link transmission, and consequently energy consumption, and to study the effect of utilising fog computing to meet the real-time requirements of an industrial scenario. To achieve these objectives, the data-driven modelling approach is deployed in a three-layer framework consisting of a lower sensor layer, a middle fog layer, and a cloud back-end in the upper layer; see Figure 4.2. The three layer architecture makes the division based on the characteristics and deployed technologies in each domain, so that their interoperability leads to a coherent IIoT monitoring system.

4.2.1 Sensor Layer

This layer consists of the resource constraint sensor devices that collect the field-level data, and the gateway that connects this layer to the upper fog layer. The behaviour of the data streams are learned at the sensors, and the periodic transmission is complemented by an event-based transmission scheme. The distributed data-driven modelling presented in Section 3.6 facilitates the functionality of this layer to eliminate the unnecessary up-link transmission load, and to reduce energy
An IIoT Monitoring System Framework and Testbed

Figure 4.3: The learning process in sensor devices and the simulation process in the fog node.

4.2.2 Fog Layer

The fog layer connects the distributed view of the sensor layer to the centralised system view of the cloud layer. Each fog node is associated with a cluster of sensors. It simulates the data streams using the updated parameters received from the sensors. Furthermore, a fog node creates a synthesis directed probability graph by collecting updated parameters from all the associated sensors. The local directed graph can be defined as $G = (V, E)$, where $V = s_i$ is a set of vertices or nodes with state $s_i$. Each state $s_i$, represents multiple data streams in which the temporal correlation of each of the streams has not changed in a specific time interval. A state can be defined...
as:

\[ s_i = \{ \tilde{D}_{i,j}(T) \mid \forall j \in \{i_1, \ldots, i_k\} : \tilde{D}_{i,k}(t_s, t_e) \in FD_{i,j}; T \in [t_{\text{start}}, t_{\text{end}}] \}, \quad (4.1) \]

where \( \tilde{D}_{i,j}(T) \) is a finite subset of the data stream \( \tilde{D}_{i,j} \) for a time interval \( T \) \( (t_{\text{start}} < t_{\text{end}}) \), and \( FD_{i,j} \) is a linear regression model that describes one sub-data stream. An edge of the graph, \( E = \{ < s_i, s_j, p_{ij} > \} \), is a set of links that represent a switch from one node to another with some probability \( p_{ij} \):

\[ p_{ij} = P(\tilde{D}[t_3, t_4] \in s_j \mid \tilde{D}[t_1, t_2] \in s_i \land \forall t_3 : t_3 - t_2 > \delta \land \tilde{D}[t_1, t_3] \notin s_j). \quad (4.2) \]

Each state \( s_i \) provides summary information about the nature of a subset of multiple data streams. The graph constructed by the fog node is the result of merging several short consecutive segments. This graph provides the fog node with a local view of the operational modes of the system, and states of the monitoring area. It also makes the detection of local anomalies, trends and sensor failure possible. When a change happens, a new node is added to the graph based on the new summary information received from the sensors. To distinguish between a trend and a state switch, the fog node follows a procedure similar to the sink’s, presented in Section 3.6. The procedures carried out in both sensor and fog layers are illustrated in Figure 4.3.

### 4.2.3 Cloud Layer

The cloud layer collects all the local graphs from the fog nodes, and constructs the global directed graph, which represents the system’s overall model. The application in the cloud back-end monitors the parameter changes in the global graph to detect anomalies, trends, and systematic faults. It is conceivable that the process of constructing the meta global graph is fairly similar to the process of the fog node.

### 4.3 The Testbed System Implementation

A testbed has been developed to investigate the IIoT monitoring framework in a real world implementation. In parallel with the framework design, the testbed system also includes the three layers of sensor, fog computing, and cloud computing; see Figure 4.4.

The sensor layer is a wireless sensor network consisting of sensors and a gateway. The sensors are programmed to imitate the distributed modelling presented in Section 3.6. The gateway connects the underlying sensor layer to the fog computing layer. The latter is implemented on a small resource-constrained computer, and sends the regenerated sensor values to the cloud computing layer at regular intervals. The cloud layer is a cloud server with a persistent storage and computational power, which stores and visualises the data in the end-user application.
4.3.1 Wireless Sensor Layer

The sensor layer is a wireless sensor network, implemented using TelosB motes [Tec] with IEEE 802.15.4 compliant transceivers, CC2420 [Ins], light and temperature sensors, and CSMA-CA medium access control protocol, running the Contiki [DGV04] operating system. The three required functional state of the sensors, i.e. initialisation, monitoring and transmission, are implemented utilising the multi-threading module of Contiki.

On system start-up, the sensor collects one sample per second, for a pre-set sampling duration $w_{\text{init}}$, and sends the first unicast message to the gateway at the end of this duration. The payload of this message contains the minimum, $\text{min}_{\text{init}}$, and maximum, $\text{max}_{\text{init}}$, values observed in sampling duration. After this process, the sensor maintains a fixed length list with the periodically collected values in the internal flash memory. The sensor-level functionalities are programmed according to the distributed modelling in Section 3.6. The sensor sends unicast messages to the gateway either if the hard-coded prediction error threshold $\delta$, based on the recent observation is exceeded, or if a sudden change within the accepted prediction error is significant enough to indicate a state or mode change.

4.3.2 Fog Computing Layer

The fog computing layer is implemented using Raspberry Pi model B+ hardware, running the Raspbian operating system version 9. The fog node needs to interpret
4.4 Evaluation and Results

the model based on the values received from the sensors, and sends them to the cloud. The functionalities required at the fog node are implemented using four concurrent threads in a Java 7 program: the sensor reader thread, the model interpreter thread, the cloud publisher thread and the REST interface thread. The sensor reader thread reads the values collected at the sensors and received from the gateway. The gateway is connected to the Raspberry Pi via USB, and creates a virtual serial port at 115200 baudrate, to communicate with the fog node. The model interpreter uses the received values from the gateway to regenerate sensor values. The publisher thread publishes sensor values to the cloud via the MQTT protocol on fixed intervals, every two seconds. The fog node connects to a local gigabit Ethernet network created by a Linksys WRT1200AC network router for communication with the Cloud layer. The REST interface thread listens for incoming HTTP GET connections on port 9999 to return the latest sensor value as a JSON object from the model in an HTTP response.

4.3.3 Cloud Computing Layer

The cloud computing layer is a persistent storage for the sensor values to be accessed and visualised in graphs and tables by the end-user applications. This layer is implemented using ThingsBoard 1.3.1 IoT cloud platform, running on a desktop computer as a server system and connected to the Linksys WRT1200AC router. The ThingsBoard IoT cloud system provides a built-in functionality of the MQTT broker and MQTT client to listen to, and to handle MQTT messages received from the fog nodes.

4.4 Evaluation and Results

A series of measurements and evaluations were conducted to verify the performance of the testbed system. More specifically, the performance was evaluated in terms of end-to-end delay of the proposed framework, the query times of the fog and cloud, the scalability of the fog, and the computational cost imposed on the sensor as a result of implementation of the distributed modelling. All the measurements were made on the local network; in other words, the presented results are an indication of the best case scenario of the system response time, but not the best case scenario of the improved response time by introducing the fog layer, since the cloud is also in the same network.

The total end-to-end delay was measured in the testbed system, by considering the delay from the generated packet at the sensor to an end-user application, running either on the cloud or the fog system. The measuring process for the delay consists of three parts including between the sensor mote and the fog node, between the fog node and the cloud node, and between the cloud and the end-user application. The total end-to-end delay \( d_{total} \) can be formulated as:

\[
    d_{total} = d_{sensor} + d_{serial} + d_{fog} + d_{cloud},
\]
Figure 4.5: The testbed experimental setup: (a) sensor motes and fog node with attached sensor gateway; (b) cloud dashboard with regenerated sensor values by the proposed model.

where $d_{\text{sensor}}$ is the sensor delay measured according to equation 2.1 by considering transmission delay, propagation delay, processing delay, and queuing delay in the WSN. $d_{\text{serial}}$ is the delay of the serial communication from the gateway, $d_{\text{fog}}$ is the fog delay for sending MQTT messages, and $d_{\text{cloud}}$ is the cloud REST interface delay. The fog-to-cloud communication delay was measured with a Java program residing on the fog device, which published the sensor values to the cloud. A Java program for end-user application was developed to access the REST interface of the cloud to evaluate the cloud system performance.

To investigate the imposed delay caused by the introduced computational overhead due to the sensor-level modelling, the end-to-end delay of the proposed model and a unicast process were compared through simulation in Cooja [ODE’06]. The query time of the fog and cloud systems were evaluated in the testbed by Java programs that performed and measured the query-response times of the REST interfaces. The scalability of the fog node was investigated with regard to the number of sensors that the fog node can handle, without performance decline.

The testbed results present the features and measurements of the collaborative performance of the sensor network layer, fog computing layer and cloud computing layer, as one coherent IIoT system. The sensors successfully transmit the model parameters to the gateway. The packets received at the fog system layer make the regeneration of the sensor values possible using solely the model parameters. The fog directly presents the sensor values to the user via a REST interface, and sends the values to the cloud system via the MQTT protocol. The values are stored on the cloud system, and presented to the user from different views: as a card with the exact value, as a digital gauge, and as an animated graph; see Figure 4.5-(b).

The end-to-end delay was studied by running the experiment 1,000 times. The results are summarised in Table 4.1. The average delay, $\mu$, is 180 ms with a standard deviation, $\sigma$, of 37 ms, which can be considered an acceptable performance for an
Table 4.1: Delay measurements split into each step.

<table>
<thead>
<tr>
<th>Delay Measurement</th>
<th>$\mu$ (ms)</th>
<th>$\sigma$ (ms)</th>
</tr>
</thead>
<tbody>
<tr>
<td>$d_{\text{sensor}}$</td>
<td>140</td>
<td>14</td>
</tr>
<tr>
<td>$d_{\text{serial}}$</td>
<td>3.4</td>
<td>1.8</td>
</tr>
<tr>
<td>$d_{\text{fog}}$</td>
<td>32</td>
<td>34</td>
</tr>
<tr>
<td>$d_{\text{cloud}}$</td>
<td>8.9</td>
<td>7.1</td>
</tr>
</tbody>
</table>

Table 4.2: Query time of the REST interfaces.

<table>
<thead>
<tr>
<th>Query Measurement</th>
<th>$\mu$ (ms)</th>
<th>$\sigma$ (ms)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fog</td>
<td>5.3</td>
<td>9.0</td>
</tr>
<tr>
<td>Cloud</td>
<td>8.9</td>
<td>7.1</td>
</tr>
</tbody>
</table>

industrial monitoring system. It should be noted that even in this testbed, where the cloud is resided in the same network, a considerable delay is added to the system. In the worst case, introducing the fog layer can reduce the end-to-end delay by 5%. The query response time of the REST interfaces on the fog and cloud layer were also measured, and are summarised in Table 4.2. The values are in the magnitude of milliseconds, which could be considered an acceptable response time.

The comparison between the hardware implementation and Cooja simulation showed that the computational overhead has a negligible effect on delay, considering the precision in the order of milliseconds in the test bed. It is added to the processing time within the allocated time slot in a magnitude of less than 2 milliseconds. Since the computational overhead does not exceed the timely delivery requirements, which is in the order of seconds, the system performance remains in the acceptable level even with this computational overhead.

The serial communication of Raspberry Pi to handle each sensor packet and model updates is 3.4 milliseconds with a standard deviation of 1.8 ms. In other words, the maximum number of sensors in each fog cluster can scale up to 290 without any decline in the performance of the fog node.

4.5 Discussion

Considering the presented results it is conceivable that the proposed framework can meet the required latency of monitoring systems in industrial scenarios, while keeping the performance above the required accuracy threshold.

The distributed modelling at the sensor layer, Section 3.6.2, implemented on sensor devices, reduced unnecessary communication, and lowered the unnecessary traffic of periodic packet transmission over the shared link by 98%, which can be utilised for high priority and unpredicted event transmissions. The results of the simulation
and experiment show that approximation of the model parameter can regenerate the data streams at the fog node with acceptable accuracy. These results clearly suggest the benefits of a combination of lower frequency periodic sampling and event-based transmissions over the transmission of the periodic and high frequency sampling. Furthermore, it is reasonable to assume that the presented model can reduce some of the negative effects of the wireless sensor communications on the centralised data-driven learning process. The distributed approach can remove the problems such as synchronisation of the received data streams values, which can add additional delay to the learning process, and the high rate of dropped packets that imposes the missing value problem on the centralised learning algorithms.

The crosscutting framework design and the functional testbed system show the benefits of the edge computing, and its possible contribution to enhancing the performance of an IIoT system in terms of real-time and deterministic performance for reliable information exchange. The successful implementation of the distributed data-driven modelling is encouraging in that it shows how multi-disciplinary solutions can be beneficial in the realisation of the IIoT systems.

### 4.6 Chapter Summary

This chapter proposed a three-layer IIoT framework for an industrial monitoring system utilising IWSN, and fog and cloud technologies. Furthermore, an IIoT testbed system was developed and implemented to examine the practicality of the framework. As a result, the third defined research goal of this thesis was achieved. The results illustrated the effectiveness of the introduced fog computing on lowering the end-to-end delay, and fulfilling the latency requirements of an IIoT monitoring system. The successful implementation of the distributed data streams modelling on TelosB motes, and the adequate performance of the testbed system by orchestrating IWSN, fog computing and cloud computing layers, positively answered the research questions RQ 3.1 and RQ 3.2, stated in Section 1.5.
Chapter 5

Conclusion and Outlook

The incoming fourth industrial revolution is changing the industrial ecosystems and demands incorporation of a wide range of disciplines and rapid advances in various technologies. The IIoT is one of the driving forces of the realisation of the future industries by developing and accelerating the technological advances that will fulfil the requirements of such industries. In other words, the future industries are only achievable by understanding, enhancing, and advancing the IIoT’s key enabling technologies, while dealing with the novel challenges by applying innovative and multi-disciplinary approaches.

The previous chapters investigated reliable information exchange in the context of IIoT with respect to communication and data analytics technologies, separately in Chapter 2 and 3, and as a system in Chapter 4. This present chapter provides an overview of this thesis, and illustrates how the material presented in each chapter contributes to the main purpose of this research. In addition, it explains how the experiments and findings of this work answered the research questions and navigated towards achieving the research goals. The potential scientific and social impacts of the presented research, as well as ethical considerations are discussed. Finally, the chapter concludes the thesis by identifying directions for future research.

5.1 Overview and Outcome

The main purpose of this study is to contribute to a better understanding of information exchange processes in IIoT systems by exploring the reliable exchange of data in communication systems, and potential enhancements in the processes when data analytics are integrated into various levels of industrial automation systems.

The research began by identifying reliable exchange of information in industrial automation systems as an open issue that has a direct effect on developing IIoT systems in industrial scenarios. Each part of the study led to the formulation of a set of questions and a goal, presented in Section 1.5, and were carried out as small projects.
Conclusion and Outlook

whose results built the bases of the next steps. This iterative process, provided the opportunity to gain deeper knowledge about the IIoT multi-disciplinary research, and to seek answers to the research questions from different perspectives.

The first goal was to investigate mechanisms for reliable exchange of information in the link layer, and to identify challenges, shortcomings, and drawbacks given IWSN as the underlying technology.

To this end, the study examined the reliable exchange of information from communication perspective by evaluating the performance of the proposed deterministic MAC protocol for aperiodic events in IWSNs - as the field-level network technology for an IIoT system. Redundancy-based techniques, specifically transmission and link redundancy, were examined to provide the reliability guarantees. The study also identified some of the drawbacks of the redundancy-based methods, and raised the question of how these drawbacks could be overcome by applying multi-disciplinary solutions to enhance the performance of an IIoT system.

The second goal of this research was to investigate the impact of exploiting raw data, and integrating data analytics to the automation hierarchy, in the industrial automation performance.

To address this goal, the study examined the role of data and data analytics in the context of IIoT. It explored the potential benefits of exploiting industrial data in IIoT systems using data-driven methods by defining various use-cases for data-driven approaches at different levels of the automation hierarchy. The study illustrated the effects of the methods on enhancing various performance areas of the IIoT systems, namely in improving visualisation readability and centralised system modelling without a prior knowledge about the characteristics of the underlying phenomena. Distributed data-driven modelling was proposed as a complementary functionality at sensors to overcome the drawbacks of the redundancy-based reliability methods, and to enhance the overall information exchange process. The evaluation of the model showed that integrating basic data analysis at the sensor level can reduce the up-link traffic load on the shared link with a negligible effect on accuracy. The consequence of this approach can potentially improve energy efficiency by eliminating unnecessary transmissions. The interesting results of the mathematical analysis of the proposed distributed modelling initiated the idea of studying its performance in an IIoT system, and thus led to the formulation of the third research goal.

The third research goal was to propose an IIoT framework for an industrial monitoring system to study the performance of the proposed data stream modelling in an IIoT system, and to develop a testbed accordingly for future research.

To this end, a three-layer IIoT framework was designed for an industrial monitoring system. An IIoT testbed was developed to examine the performance of the framework in terms of improved link utilisation and reduced latency, and to investigate the interoperability of various technologies for realisation of an IIoT system. In the lower layer, an IWSN was considered and the distributed modelling was successfully implemented on the sensor devices with IEEE 802.15.4 compliant
transceivers. A cloud server composed the upper layer for data storage and hosting the end-user application. A fog layer connecting the sensor layer to the cloud layer was introduced as a technique to improve the real-time performance of the system. The testbed implementation showed an acceptable performance of the IIoT monitoring system when data analytics were integrated into various levels of the system in terms of latency, reduced up-link traffic, and exchange of information instead of raw data transmission.

All in all, the methods, experiments, results and discussions presented in this thesis identified and addressed some of the gaps in the IIoT research area, and contributed to the existing and ongoing research in this field. Therefore, it is comprehensive to conclude that the goals, and consequently the main purpose of this thesis have been achieved.

5.2 Impacts, Social and Ethical Considerations

The discussion on the impacts and ethical considerations of this research can be derived from both a scientific and a social perspective.

5.2.1 Impacts

From a scientific perspective and technical point of view, this work contributes to the active research on IIoT technology. Specifically, the proposed framework and the testbed system can be considered as a reference architecture to study the performance of IIoT systems and their components for various applications. From a higher perspective, IIoT falls under the huge umbrella of the fourth industrial revolution. Researches in the areas related to IIoT will affect the design and development of the systems tailored to the new industrial paradigm. Enhanced automation can free human force from mechanical tasks and open the possibility for creativity and innovative problem solving. On the social level, the fourth industrial revolution is expected to alter the way we think, live, and interact with each other, as well as to improve the quality of life by long-term gains in efficiency and productivity [Sch17]. The hope is that the presented approach and discussions in this thesis can contribute to the research directions and innovative solutions that potentially can accelerate the realisation of the future industries, and achieve the premises of the next industrial revolution.

5.2.2 Ethical Considerations

This thesis investigated the role of data and information exchange in industrial automation. While considerations such as authentication, security and privacy fall outside the scope of this thesis, it is important to point out their gravity in developing ethical IIoT systems. The fusion of data and data analytics in industrial automation
Conclusion and Outlook

systems hints towards two main concerns regarding the source of the data, and the outcomes of the processed data. The data gathered from factory halls are considered assets, since they are valuable source of information that provide insight to the systems. Therefore, the collected data need to be handled safely to prevent unauthorised access to the stakeholders’ assets due to the competitive nature of the industries. Moreover, in the future industries with the blurring lines between the physical and digital worlds, the workforce will constitute of humans and machines. The data collected by machines might contain human data, directly or indirectly, which will pose a challenge on confidentiality, and need to comply with data protection protocols for subject awareness and permission [FW17].

The downside to the love of truth is that it may lead scientists to pursue it regardless of unfortunate consequences. Scientists do bear the heavy responsibility to warn society of those (unfortunate) consequences [Daw99]. Accordingly, the negative effects that the fourth industrial revolution might pose to the society, and the ethical issues which might follow, have justly been speculated in the scientific community [Teg17, FW17, Sch17]. It has been discussed that the fourth industrial revolution could yield great inequality for consumers as well as in labour market [Sch17]; the benefits might be limited to those who can afford the services, and the job market might shrink for the work force with lower skill. Furthermore, while the future industries could be more efficient and productive by integration of the intelligent technologies into the automation systems, the work force need training to adopt the new mind set, and to adapt to the new ecosystem [FW17]. While technological advances are accelerating, strategies to deal with the aforementioned ethical concerns need to be set and carried out by the economical and political policy makers. We have to win this race between the growing power of the technology, and the growing wisdom with which we manage it. We do not want to learn from mistakes [Teg17].

5.3 Future Work

IIoT is in early stage, hence, there are many aspects open for investigation regarding the performance and efficiency of the IIoT systems as a whole, as well as the individual components constituting the systems. In direct connection to the research work presented in this thesis, there are potential researches that could be conducted to improve the proposed methods or to expand the scope of the applied techniques.

The centralised multi-mode system modelling could be extended to an on-line fault detection and prognosis system. A potential approach could be to develop a data-driven collaborative feedback system that in each time instance combines the separate views of individual sensors, and their correlation, with the behaviour of the system in different modes; this might improve the accuracy of the approximated system states, and that of the sensors’ faulty behaviour. The distributed data streams modelling meets the real-time requirements of an industrial monitoring system. The performance of the model can be further investigated for hard real-time requirements of the mission critical applications in industrial automation. This investigation might provide new ideas and research directions for reliable IIoT-based process
control systems.

In the proposed IIoT framework, the construction of the synthesis local graph in the fog nodes and the global graph in the cloud could be examined for multiple data streams. Similarly, an extended model could be implemented in the testbed system for further performance investigation under more realistic conditions. The performance evaluation of the framework and the testbed system were performed with a limited number of nodes, and the scalability of the testbed was evaluated theoretically with mathematical analysis. Considering the importance of scalability in IIoT systems, it is necessary to further study the performance of the testbed with an increased number of sensors in the network. In addition, the testbed system could be further developed to contain various technologies, standards and devices in each layer for interoperability studies regarding the IIoT systems.
List of Figures

1.1 Elements and processes in Industrial IoT. ........................................... 2
1.2 Some of the key enabling technologies of Industrial IoT. ................... 3
1.3 The industrial revolutions timeline [Mjo]. ........................................ 4
1.4 The changing ecosystem of industrial automation. The automation pyramid is gradually changing to adopt the distributed, networked and automated vision of IIoT. (a) The IoT-enabled industrial automation pyramid as of today; higher levels of the pyramid are affected by IoT technology to enhance organisation and business efficiency. (b) The vision of IIoT and Industry 4.0 for industrial automation. ............ 7
1.5 Research work flow. ................................................................. 10
1.6 Research map and chapter guide. .................................................. 12
2.1 An example for IWSN topology. .................................................... 17
2.2 The superframe structure of (a) IEEE 802.15.4, (b) an alternative WirelessHART shared sub-slots allocation, and (c) DeMAC contention free structure. ................................................................. 20
2.3 Comparison of the MAC protocols regarding timely delivery of the events. (a) Probability distribution of delay (PDD) for all received $P_{ets}$ over the work cycle. (b) Relative performance comparison of the algorithms regarding delay; the boxplot shows the median, and whiskers show the best and the worst case delay. ....................... 24
3.1 Data analytics design considerations for IIoT applications. ................. 29
3.2 Performance comparison between the proposed clustering algorithm, and distance-based and the correlation-based hierarchical clustering algorithms; correlation and distance are balanced better in the proposed approach. .................................................. 35
3.3 Average time of running each of the algorithms for 30 times on data sets with increasing the number of dimensions. ......................... 36
3.4 The independence structure of a multi-mode system model. ............ 37
3.5 The data set consist of 21 sensors data streams. The first row is visualises (a) the original data set, and (b) the min-max normalised data set. The second row illustrates three types of diverse behaviour in the data set. The data streams have different change patterns and sampling rates, (c) Stream A 100 ms, (d) Stream B 500 ms, and (e) Stream C 200 ms. .......................................................... 40
3.6 Model selection by comparing various criterion values of HMMs with different orders; Akaike Information Criterion $AIC_c$, Bayesian Information Criterion $BIC$, and Efficient Determination Criterion $EDC$ [ZDG01]. ................................................................. 41
3.7 The performance comparison of the three algorithms: (a) incorrect label predictions, (b) accuracy score of the models, and (c) running time for carrying out the labelling task. .................................. 42
3.8 The result of state-mode identification by the algorithms on the data set. Each state is distinguished by a different colour. To enhance readability, only three of the data streams are shown; the column to the left, (a)-(d)-(g), Stream A; the column in the middle (b)-(e)-(h) Stream B; and the right column, (c)-(f)-(i), Stream C. ......................... 43
3.9 The model MSE measure comparison with respect to the segment length and the trend threshold. ................................. 50
3.10 Comparisons between normalised sensor data (blue) and the simulated sensor stream (orange). The top row ((a)-(b)-(c)) shows the performance of the proposed method for the duration of the experiment (40,000 samples) on different data streams. To illustrate the results in more detail, the bottom row ((d)-(e)-(f)) shows a zoomed in view of a shorter interval (1,200 samples on the same data streams). ............. 53
4.1 Analytics mapping to the IIoT Reference Architecture [ADF+17]. .... 56
4.2 Proposed three layer framework for IIoT monitoring systems. ........ 57
4.3 The learning process in sensor devices and the simulation process in the fog node. .......................................................... 58
4.4 The testbed system consisting of cloud server, Raspberry Pi fog nodes, and TelosB sensor motes. ................................. 60
4.5 The testbed experimental setup: (a) sensor motes and fog node with attached sensor gateway; (b) cloud dashboard with regenerated sensor values by the proposed model. ................. 62
List of Tables

2.1 Simulation parameters and settings. ........................................... 22
2.2 The performance comparison of the two algorithms regarding the defined metrics. ................................................................. 23
3.1 Statistics of the data sets in PixVid experiments. ....................... 35
3.2 Performance of the algorithm regarding various classification metrics. 42
3.3 The mean square error of different settings for segment length \( m \) and trend threshold \( \theta \). ................................................................. 50
3.4 The mean square error of sending the average value with different segment length. ................................................................. 50
4.1 Delay measurements split into each step. ................................. 63
4.2 Query time of the REST interfaces. ........................................... 63


[HPK11] Jiawei Han, Jian Pei, and Micheline Kamber. *Data mining: concepts and techniques*. Elsevier, 2011.


[IEE06] IEEE. IEEE standard for information technology– local and metropolitan area networks– specific requirements– part 15.4: Wireless medium access control (mac) and physical layer (phy) specifications for low rate wireless personal area networks (wpans). *IEEE Std
802.15.4-2006 (Revision of IEEE Std 802.15.4-2003), pages 1–320, Sept 2006.

[IEE12] IEEE. Ieee standard for local and metropolitan area networks–part 15.4: Low-rate wireless personal area networks (lr-wpans) amendment 1: Mac sublayer. IEEE Std 802.15.4e-2012 (Amendment to IEEE Std 802.15.4-2011), pages 1–225, April 2012.


