Robust Data Protection and High Efficiency for IoT Streams in the Cloud

Alsharif Abuadbba

BSc (Computer science)
MSc (Computer science)

A thesis submitted in fulfilment of the degree of Doctor of Philosophy

Discipline of Computer science, information technology and software engineering
School of Science
College of Science, Engineering and Health
RMIT University
Melbourne, Australia

November 1, 2017

Declaration

I certify that except where due acknowledgement has been made, the work is that of the author alone; the work has not been submitted previously, in whole or in part, to qualify for any other academic award; the content of the thesis/project is the result of work which has been carried out since the official commencement date of the approved research program; any editorial work, paid or unpaid, carried out by a third party is acknowledged; and, ethics procedures and guidelines have been followed. I acknowledge the support I have received for my research through the provision of an Australian Government Research Training Program Scholarship.

Alsharif Abuadbba
November 1, 2017

Acknowledgements

If I have seen further than others, it is by standing upon the shoulders of giants.
—Isaac Newton

First and foremost, I would like to express my special appreciation and substantial thanks to my two supervisors, Associate Professor Ibrahim Khalil and Professor Timos Sellis. I am grateful for their supervision, insightful suggestions, precise feedback and commitment. I am very honoured to have been supervised by two truly experienced and extensively acknowledgeable and yet genuinely humble and open-minded individuals. Without Ibrahim and Timos' guidance, this achievement would not be possible.

I would also like to express my deep thanks to my fellow PhD candidates and close friends who made the journey very enjoyable and memorable: Ayman Ibaida, Dhiah Alshammary and Shaahin Madani. I deeply cherish the valuable time we have spent together, the inspiring discussions, the knowledge and experience we mutually gain from each other.

Special thanks goes to my family. Words can not express the extreme gratitude I have for my parents, Mohammed and Masouda, who forever raised, encouraged and supported me on everything. I also thank my brothers and sisters who always inspired and motivated me. Finally, I express great thanks and love for my wife, Zahra, for her patience, for being my dearest friend and companion. My family, without you, I could not have accomplished what has been achieved today.

Lastly, I must acknowledge that I indeed enjoyed the journey. I strongly believe in the motto, 'it is all about the journey and not about the destination.'

Credits

Portions of the material in this thesis have previously appeared in the following publications:

Peer Reviewed Papers

- Alsharif Abuadbba and I. Khalil, "Walsh-Hadamard Based 3D Steganography for Protecting Sensitive Information in Point-of-Care." IEEE Transactions on Biomedical Engineering (2016), vol.PP, no.99, pp.1-1, DOI: 10.1109/TBME.2016.2631885 **(SCImago Q1) (ERA A*)**.

- Alsharif Abuadbba and I. Khalil. "Wavelet based steganographic technique to protect household confidential information and seal the transmitted smart grid readings." Information Systems 53 (2015), Vol.53, p.224-236, DOI: 10.1016/j.is.2014.09.004 **(SCImago Q1) (ERA A*)**.

- Alsharif Abuadbba, I. Khalil, and M. Atiquzzaman. "Robust privacy preservation and authenticity of the collected data in cognitive radio networkWalshHadamard based steganographic approach." Pervasive and Mobile Computing 22 (2015), Vol.22, p.58-70, DOI: 10.1016/j.pmcj.2015.02.003 **(SCImago Q1)**.

- Alsharif Abuadbba, I. Khalil and Xinghuo Yu Gaussian Approximation Based Lossless Compression of Smart Meter Readings. IEEE Transactions on Smart Grids (2017), Vol.PP, Issue:99, DOI: 10.1109/TSG.2017.2679111 **(SCImago Q1)**

- Alsharif Abuadbba, et al. "Resilient to shared spectrum noise scheme for protecting cognitive radio smart grid readings BCH based steganographic approach." Ad Hoc Networks 41 (2016), Vol.41, p.30-46, DOI: 10.1016/j.adhoc.2015.11.002 **(SCImago Q1)**.

- U. Premarathne, A. Abuadbba, A. Alabdulatif, I. Khalil, Z. Tari, A. Zomaya, and R. Buyya. Hybrid Cryptographic Access Control for Cloud-Based EHR Systems. IEEE Cloud Computing (2016), Vol.3, Issue:4, p.58-64, DOI: 10.1109/MCC.2016.76.

- Alsharif Abuadbba, I. Khalil and Timos Sellis Can the Multi-Incoming Smart Meter Compressed Streams be Re-Compressed? IEEE Transactions on Smart Grids, (2017) (Submitted) **(SCImago Q1)**.
- Abuadbba Alsharif, I. Khalil and Timos Sellis N-Split Based Lossless Compression of Smart Grid Meters. IEEE Transactions on Smart Grids, (2017) (Submitted) **(SCImago Q1)**.

The thesis was typeset using the LaTeX 2_ε document preparation system.

Contents

List of Figures

List of Tables

List of Acronyms

AC	Arithmetic Coding
AES	Advanced Encryption Standard
ASCII	American Standard Code for Information Interchange
BCH	Bose-Chaudhuri-Hocquenghem
BER	Bit Error Rate
BWT	Burrow-Wheeler Transform
CP	Cloud Provider
CR	Cognitive Radio
CRi	Compression Ratio
CRN	CR Network
D-RLE	Dynamic-Run Length
DoB	Date of Birth
DWT	Discrete Wavelet Transform
FWHT	Fast Walsh-Hadamrd Transform
IoT	Internet of Thing
MTF	Move-To-Front
PLC	Power-Line Communication
PMU	Phasor Measurement Unit
PRD	Percent of Root-mean-square Difference
PU	Primary User
RLE	Run Length
RMS	Root Mean Square

stego	steganography
SU	Secondary User
WSN	Wireless Sensor Network
XG	neXt Generation
XOR	Exclusive OR

Abstract

Remotely generated streaming of the Internet of Things (IoTs) data has become a vital category upon which many applications rely. Smart meters collect readings for household activities such as power and gas consumption every second - the readings are transmitted wirelessly through various channels and public hops to the operation centres. Due to the unusually large streams sizes, the operation centres are using cloud servers where various entities process the data on a real-time basis for billing and power management. It is possible that smart pipe projects (where oil pipes are continuously monitored using sensors) and collected streams are sent to the public cloud for real-time flawed detection. There are many other similar applications that can render the world a convenient place which result in climate change mitigation and transportation improvement to name a few. Despite the obvious advantages of these applications, some unique challenges arise posing some questions regarding a suitable balance between guaranteeing the streams security, such as privacy, authenticity and integrity, while not hindering the direct operations on those streams, while also handling data management issues, such as the volume of protected streams during transmission and storage. These challenges become more complicated when the streams reside on third-party cloud servers. In this thesis, a few novel techniques are introduced to address these problems.

We begin by protecting the privacy and authenticity of transmitted readings without disrupting the direct operations. We propose two steganography techniques that rely on different mathematical security models. The results look promising - security: only the approved party who has the required security tokens can retrieve the hidden secret, and distortion effect with the difference between the original and protected readings that are almost at zero. This means the streams can be used in their protected form at intermediate hops or third party servers.

We then improved the integrity of the transmitted protected streams which are prone to intentional or unintentional noise - we proposed a secure error detection and correction based stenographic technique. This allows legitimate recipients to (1) detect and recover any noise loss from the hidden sensitive information without privacy disclosure, and (2) remedy the received protected readings by using the corrected version of the secret hidden data. It is evident from the experiments that our technique has robust recovery capabilities (i.e. Root Mean Square (RMS) $<$

0.01%, Bit Error Rate (BER) = 0 and PRD < 1%).

To solve the issue of huge transmitted protected streams, two compression algorithms for lossless IoTs readings are introduced to ensure the volume of protected readings at intermediate hops is reduced without revealing the hidden secrets. The first uses Gaussian approximation function to represent IoTs streams in a few parameters regardless of the roughness in the signal. The second reduces the randomness of the IoTs streams into a smaller finite field by splitting to enhance repetition and avoiding the floating operations round errors issues. Under the same conditions, our both techniques were superior to existing models mathematically (i.e. the entropy was halved) and empirically (i.e. achieved ratio was 3.8:1 to 4.5:1).

We were driven by the question 'Can the size of multi-incoming compressed protected streams be re-reduced on the cloud without decompression?' to overcome the issue of vast quantities of compressed and protected IoTs streams on the cloud. A novel lossless size reduction algorithm was introduced to prove the possibility of reducing the size of already compressed IoTs protected readings. This is successfully achieved by employing similarity measurements to classify the compressed streams into subsets in order to reduce the effect of uncorrelated compressed streams. The values of every subset was treated independently for further reduction. Both mathematical and empirical experiments proved the possibility of enhancing the entropy (i.e. almost reduced by 50%) and the resultant size reduction (i.e. up to 2:1).

Chapter 1

Introduction

Recently, there has been an enormous interest in the remote gathering of data to effectively monitor various activities on real-time bases such as smart homes, smart grids, climate change, border invasion, smart manufacturing, nuclear facilities or smart transportations [11]. The data is collected wirelessly using small devices (e.g. sensors) known as Internet of Things (IoTs) and forwarded to their final destination such as operation centres through the Internet [11]. The continuous streams usually contain two types of data: (1) normal readings (e.g. meter power readings) and (2) highly-sensitive information (e.g. household identity, nuclear facility tags, border screen geometric location, facility IDs or small pictures of the locations coupled with date and time).

According to Cisco Visual Networking Index (VNI) Complete Forecast for 2015 to 2020 [12], the growth in the number of these devices will increase three-fold from 4.9 billion to 12.2 billion between 2015 and 2020. The IP generated data traffic is expected to explode from 72.5 exabytes (1 exabyte = 2^{60} bytes) to 194.4 exabytes per month between 2015 and 2020. The report also highlights that 46 percent of that traffic will be sensor streams, in other words, nearly half produced by IoTs. The extremely large generated streams (i.e. big data) will pose unique challenges regarding security and data management.

Dealing with these concerns in the high-stream sensors data context will be more complicated for two main reasons. Firstly, the generators of these streams (e.g.

remote sensors or smart meters) pose unique challenges (e.g. their presence in an uncontrollable environment, resource constraints such as memory and power, and topological constraints where the data should go through multiple public hops to the final destination) which prevent a direct transplant of existing privacy preservation and authenticity techniques [13]. Secondly, the significant size and the speed of these streams force the operation centres to (1) use a concept called 'cognitive radio' where the data can be sent wirelessly through various shared licensed spectra (i.e. channels) based on their availability and (2) conduct offshore operations by using cloud public servers, for example.

Additionally, in 2013 Intel surveyed more than 200 information technology managers from various technology companies in the United States about their main concerns regarding IoTs generated big data streams [14]. The top two obstacles are data security (i.e. privacy, authenticity, and integrity) and data management (i.e. increased network bottleneck, storage and real-time analysis). Surprisingly, although these concerns have been well studied independently for static data, tackling them together received limited attention and are still immature for streaming, dynamic high-speed IoTs sensors data into the cloud environment [15, 16]. This is due to (1) the fast, sharp increase of IoTs and their compelled adoption, and (2) the emergence and the urgent need for the power and capabilities of the cloud infrastructure. To achieve the incredible benefits from the generated IoTs streams using real-time analysis while guaranteeing the outcomes, the issues of privacy (i.e. transmitted sensitive information), authenticity (i.e. the origin of transmitted streams), integrity (i.e. manipulation detection, remedy, and recovery) and size (i.e. during transmission and storage) have to be treated together.

The aim of this research is to address these four major problems together concurrently with the IoTs streaming data to cloud environment and bridge the gap between them. Therefore, this research proposes new algorithms to be integrated into the IoTs end-point devices, such as meters and sensors. Further compatible algorithms are also introduced at intermediate hops or gateways. In contrast, additional

harmonious techniques will be presented as cloud-enabled algorithms appropriate for deployment to the cloud.

1.1 Research Scope and Challenges

IoTs streaming data has become a core category upon which many applications rely. For instance, smart meters gather household power and gas consumptions periodically (e.g. every second) and transmit them wirelessly through various channels and public hops to the operation centres [17]. Due to the unprecedented volume of streams and for real-time analysis, the operation centres are using cloud servers where various entities process the data on a real-time basis for billings and power management. For instance, imagine a smart oil pipe project where the pipes are continuously monitored, and the collected streams are sent to the public cloud for real-time detection of any possible flaws. There are many projects that help create a more convenient world, such as solar panels and transportations, and cloud technology is crucial one. Despite the clear merits of these applications, unique challenges may arise [18]; how can we have a suitable balance between (1) ensuring the streams security (i.e. privacy, authenticity and integrity) while not hindering the direct operations on those streams, and (2) handling the data management issues such as its size during the transmission and storage. These challenges become more complex in cases where the streams reside on third-party cloud servers. In the following, these challenges are discussed in more depth.

- **Privacy and Authenticity**: The following points shape the unique privacy and authenticity challenge:
  - The presence of IoTs end-point devices in hostile areas.
  - The transmission of entities' (customers) highly-sensitive data through public networks
  - Their residence and processing on third-party cloud servers.

Due to these challenges, countries such as Australia and the United States are forming strict regulations on companies mandating that an IoTs client's sensitive information **must** be kept secure from unauthorised access even when the company performs offshore operations, such as using and storing on cloud servers [19, 20]. In fact, there are two concerns. (a) From an IoTs customer point of view, a main concern regards their sensitivity to the privacy and confidentiality of their personal information, such as their identification documents, addresses and geometric locations; while contrarily they want to ensure the authenticity of collected power consumption readings (for example) and resultant billings are completely accurate (i.e. Is this bill calculated correctly and are the readings from the customer's premise?). (b) The main worry from an operational centres' viewpoint is about ensuring the efficient (applying direct operation on collected streams) and secure technique that helps them protect their customers' confidential information.

- **Integrity**: Due to the sharp growth in the number of IoTs devices that simultaneously transmit continuous critical streams, the chances of interference between applications that use identical or overlapping bands are highly increased (e.g. Bluetooth and ZigBee at 2.4 GHz) [21]. To overcome this, a new wireless communication technique called Cognitive Radio (CR) has emerged where applications can use any other available idle bands. The shared spectrum characteristics create a huge integrity challenge where the transmitted data is highly prone to intentional attacks (i.e. interference) and unintentional attacks (i.e. noise) rendering these applications impractical [22]. This is simply because any slight change in the transmitted readings will result in loss of crucial information and, more significantly, loss of faith in the received readings. In some cases, it may be too late to ask the source to resend especially in critical cases; other times the source is often configured to forget what it sends directly (i.e. due to resource constraints); lastly, the destination has no return channel to the source - these contributing factors create many complications [23]. In an

extreme case, this may happen to millions of premises in the same day!

- **Data management**: The unforeseen magnification in the speed and size of collected IoTs sensor streams renders many operation centres who own these streams incapable of handling them. This imposes utilizing a third-party elastic cloud environment to perform direct operations on the collected streams. The two drawbacks of that are: (1) a huge-unexpected pressure on the communication infrastructure during the transmission, and (2) exponential increase in the storage space cost and burdening the available data management. The unique challenge is that there is a lack of proper data management [14] models, such as size reduction and direct operations, that is harmoniously compatible with privacy, authenticity and integrity. This is because many data management models, such as real-time data mining and size reductions, work on the original form of the collected readings, but neglect to observe security aspects or assume that they are beyond the scope of focus. This was acceptable in the traditional model where the data travelled between point A and point B, and both belong to the owners of the data. However, this assumption is not applicable in the cloud environment where the owners of the data want to employ the power of the third-party cloud infrastructure for better management while not disclosing any security aspects such as privacy.

Therefore, deeply considering the above core challenges, the following is the summary of the core issues derived and targeted (See Figure 1.1):

- Protecting privacy requires a way to preserve the sensitive information while not hindering the direct operation on the collected readings.
- Ensuring the authenticity of the collected readings requires a unique imperishable seal at the origin that is not disclosed on the cloud level.
- Guaranteeing the integrity requires a mechanism to easily detect any alteration to the transmitted readings without disrupting direct operations on the transmitted readings.

- Improving the communication infrastructure requires size reduction techniques at the IoTs end-points without losing any bit or disclosing the privacy and authenticity.

- Improving data management by reducing the burden and the cost on the cloud level, which requires special size re-reduction models by finding any potentially similar correlation between the myriad of multiple-incoming streams.

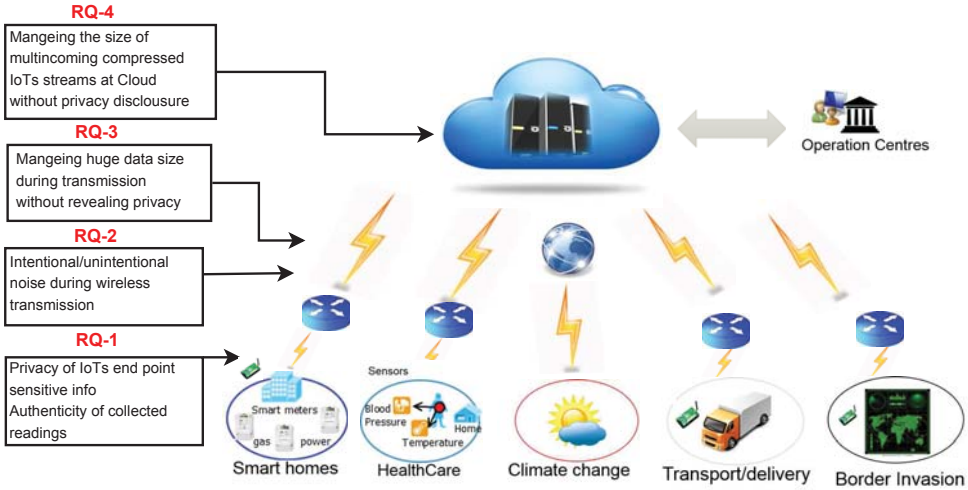

Figure 1.1: High IoTs streams in the cloud environment, and the correlated security and data management problems.

1.2 Research Questions

The core objective of this thesis is to introduce harmonious algorithms that guarantee and secure the privacy, authenticity and integrity of dynamically high IoTs streams into the cloud environment while providing better data management by the way of size reduction. On the other hand, these streams allow direct operations on third-party servers without security disclosure.

To overcome the above unique challenges and to achieve the stated objectives, the following research questions were formed.

- **Research Question (RQ)-1. How can the privacy of the sensitive information and the authenticity of the transmitted IoTs sensor**

streams be protected without hindering direct operations at intermediate hops or cloud?

One of the major issues in IoTs field is with the privacy of customer sensitive information as well as the authenticity of the transmitted collected streams. These streams reside in remote third-party servers. Researchers have introduced various models to protect these streams. They mostly rely on implementing traditional heavy cryptographic mathematical mechanisms, which means they inherit issues such as large overheads, expensive computational complexity and changing the form of the data. In doing so, sensitive information would have to be disclosed to process the streams. Therefore, to address this question, we understand that:

(i) this must ensure the privacy of transmitted sensitive information

(ii) this must guarantee the authenticity of the normal transmitted readings

(iii) this needs to avoid an increase or change from the original structure of normal readings

(iv) allows direct operations on the received IoTs streams at cloud public servers without security disclosure.

- **RQ-2.** In the first research question, our novel technique to guarantee both privacy and authenticity without changing the form of the readings is by encrypting the sensitive information and to hide them by using steganography, randomly, bit-by-bit inside the normal transmitted readings. However, if the transmitted streams are altered either intentionally or unintentionally during their travel from the end-point IoTs device to the cloud, we may not be able to detect or recover the hidden information, and we may lose confidence in the received readings. Re-requesting and resending these streams will be very expensive and a burden on the infrastructure. Therefore, our second research question is: **How can any alteration to the transmitted information be detected and recovered without hindering direct operations at**

intermediate hops or cloud?

The identified targets from this question are that we need to:

(i) detect any alteration to the transmitted streams using hidden sensitive information

(ii) recover any lost data from the transmitted sensitive information

(iii) remedy the received streams using the recovered sensitive information

(iv) understand that this does not increase or change the original form of the transmitted readings to allow continuous direct operations on the collected streams.

- **RQ-3.** In the first research question, our solution to ensure both the privacy and the authenticity at the same time is by encrypting only the sensitive information and hide them randomly on the bit level inside the transmitted normal readings. Therefore, this leads us to the following question: **How can the size of the protected transmitted IoTs streams that contain the encrypted hidden information be blindly reduced without any security disclosure?**

The objectives designed from this question are:

(i) reduce the huge size of transmitted sealed readings for improving the performance and so the efficiency

(ii) achieve better results than general compressors by exploiting the new form of protected readings characteristics

(iii) work seamlessly and blindly with our solution to questions 1 and 2, thereby not losing any data and ensuring all encrypted and hidden information can be recovered.

- **RQ-4.** On cloud level, multiple compressed streams of protected readings are received from various sources in a very short period that will quickly reserve a huge space. Therefore, focus of this question is: **Can the multi-incoming compressed protected IoTs sensor streams be re-reduced without privacy or authenticity disclosure?**

  The objectives of this question are:

  (i) to re-reduce the massive volume of received compressed streams on cloud level by exploiting the similarities among a wide range of multi-incoming streams

  (ii) to work seamlessly and blindly with our solution to questions 1, 2 and 3, thereby not losing any data and so all encrypted and hidden information can be recovered.

1.3 Limitations of Existing Solutions

A majority of the early solutions that ensure privacy and authenticity relied heavily on transplanting cryptography techniques such as symmetric, asymmetric and digital signature [24–29]. These models focused on how to protect the data from point A to point B (i.e. sender to receiver) during transmission. However, following the emergence of the cloud (and its compelled adoption for its immense benefits), another dimension became crucial to any proposed security model in the IoTs field - the efficiency aspect because of the huge size and speed of the collected streams (i.e. big data) that needs to be analysed directly on the cloud without tolerating the security aspects. Therefore, these solutions became outdated and is currently unfit to todays handling infrastructure. Firstly, they change the form of all original data into a ciphertext which hinders the ability to work directly on the transmitted streams at intermediate hops (i.e. where the transmitted data may be collected and compressed)

or on cloud servers. Secondly, on cloud servers, the privacy of the sensitive information should be disclosed (i.e. decrypted) to perform operations on the collected streams.

On the other hand, most of the data management models, such as size reduction and data mining, neglect the security aspects and assume they have access to the readings directly. This was acceptable when both ends were controlled by the data owners. However, this is not the case in the cloud infrastructure model where the readings may end up in public servers in unknown jurisdictions. For instance, most of the current size reduction (i.e. compression) models are lossy [30] - this means the models rely on losing some information while trying to maintain the main features of the waveforms signal. This means they have access to the actual readings, and that the streams are not required for crucial applications. This kind of compression was acceptable in the traditional sensor streams, and so most research has been conducted in this path which can be classified [31–33] into transformation techniques, parametric coding and mixed (see Chapter Four). This can easily achieve higher compression ratios while losing small amount of data. Lossy compression was recently discouraged after the emergence of smart IoTs applications such as smart meters, and after the potential use of the remotely collected readings for billings and other purposes.

Conversely, only few research has been done under lossless reduction category such as in [8, 9]. This is because in 'lossless compression', it is an obligation to reconstruct the exact waveform signal of the readings with original zero loss. These models are also assuming direct access to the original data.

Therefore, due to the limitations gap between the security and data management models of IoTs streams in the cloud environment, this thesis concentrates on developing an end-to-end framework that comprises some unique methods to ensure security (i.e. privacy, authenticity and integrity). It also provides a privacy preserving size reduction without hindering direct operations on the protected transmitted readings.

1.4 Contributions

To address the research questions explained in Section 1.2, and to bridge the gap between security techniques and data management models, this thesis introduces a novel framework that consists of several new techniques to the area of high IoTs data streams, such as sensors and smart meters. The contributions can be classified into two main domains: security for privacy, authenticity and integrity, and data management for size reduction and data recovery. Figure 1.2 identifies the core contributions and recognises their links to the core issues.

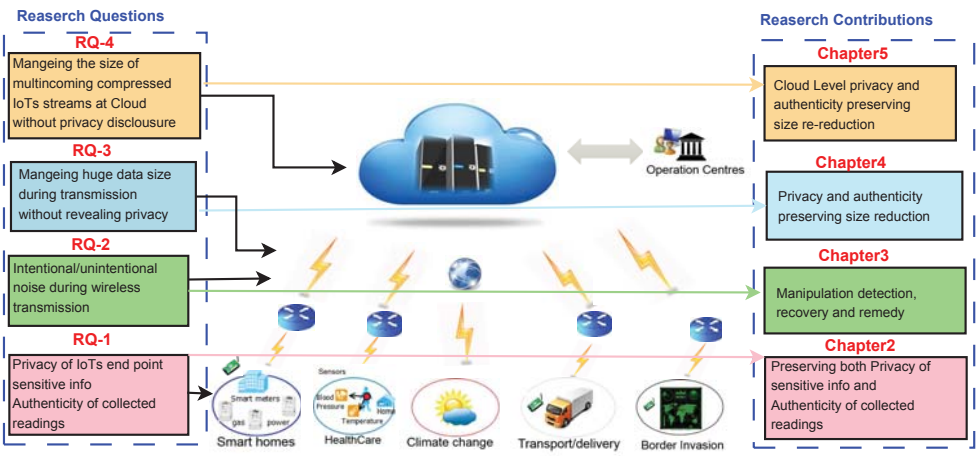

Figure 1.2: A summary of research questions, contributions and thesis structure.

- **Privacy preserving of private information and authenticity of collected streams while not hindering direct operations on the readings.**

  The focus of Chapter 2 is to (1) guarantee the privacy of the customer sensitive information and (2) ensure the source of origin of the collected readings. Steganography is employed as an underline mechanism to conceal the secret information. Two new models are proposed in this chapter which are Walsh-Hadamard based stenography and Wavelet based steganography. They vary in their security, hiding capacity and simplicity. Both techniques are neither increasing nor changing the form of the transmitted readings. In other words, only data owners can retrieve the seal whereas others are just monitoring and can use the protected form of the readings.

- **Manipulation detection, remedy, and recovery while not hindering direct operations on the data nor changing the form of the readings.**
  Chapter 3 concentrates on the integrity of the transmitted readings by (a) detecting and recovering any loss from the hidden confidential information without privacy disclosure, and (b) also curing the received normal readings by using the corrected version of the secret hidden data. A new model is introduced in this chapter that uses both steganography, and error detection and correction technique called Bose-Chaudhuri-Hocquenghem (BCH) syndrome codes. The model is extensively measured using various well-known benchmarks such as Bit Error Rate (BER), Percent of Root-mean-square Difference (PRD) and Root Mean Square (RMS). It is obvious from the experiments that our technique has robust recovery capabilities (i.e. BER =0, PRD $< 1\%$ and RMS $< 0.01\%$).

- **Hidden information preserving size reduction without privacy and authenticity disclosure**
  Chapter 4 focuses on reducing the size of protected readings at intermediate hops without revealing the hidden secrets. Two novel lossless IoTs (smart meter) readings compression algorithms are proposed. The first Gaussian-based model target is representing IoTs sensor streams in a few parameters regardless of the irregularity in the signal. The second N-Split model target is reducing the randomness in the IoTs streams into a smaller finite field to boost duplications and to avoid the floating operations round errors issues. After a thorough evaluation under the same conditions, both our techniques were mathematically superior to existing models where we successfully halved the entropy, and empirically we achieved a ratio of 3.8:1 to 4.5:1.

- **Cloud-based hidden features preserving size re-reduction**
  The work in Chapter 5 addresses whether the protected readings of multi-incoming compressed IoTs can be re-compressed? The answer is an affirmative - by pre-processing the compressed streams in such a way that it improves the theoretical entropy and exploits it. This is successfully achieved by

employing a supervised learning classifier as a similarity measurement to classify the compressed streams into subsets to reduce the noise impact of dissimilar compressed streams. The streams in every resultant subgroup have been treated separately to reduce the size. Both mathematical and empirical experiments proved the significant improvement of the compressed streams entropy (i.e. almost reduced by half) and the resultant compression ratio (i.e. more than 50%). To the best of our knowledge, there has been no other study that tackles this issue within the IoTs sensor streams field.

1.5 Thesis Structure

The rest of the thesis is organised as follows.

1. Chapter 2 explains the solution to Research Question 1. It presents two novel privacy preserving and authenticity protection models that rely on steganography. It also summarises relevant work and presents our algorithms in main stages. Next, evaluation of different characteristics of the proposed steganography-based techniques are introduced followed by a discussion about the experiments we performed and the attained results. Finally, we summarised this chapter.

2. Chapter 3 describes the solution to Research Question 2. It introduces a resilience to the shared spectrum noise scheme using BCH syndrome codes combined with steganography to protect the integrity of transmitted protected IoTs streams. A preliminary introduction about BCH syndrome codes is provided. Next, we present the mathematical model of our algorithm. A thorough security and noise impact evaluation are then performed and provided. The conclusion is finally drawn for this chapter.

3. Chapter 4 explains the solution to Research Question 3. It introduces two novel techniques that ensures hidden information preserving size reduction. We begin by summarising the relevant work and introducing the mathematical models of

our size reduction algorithms. This is followed up with a detailed discussion on our experiments and we make comparisons with existing models. Finally, we summarised this chapter.

4. Chapter 5 introduces the solution to Research Question 4. It presents our cloud-based size re-reduction model. A preliminary analysis of the received compressed streams is introduced. Then, our algorithm along with the evaluation and results are presented. This chapter is lastly concluded.

5. Chapter 6 concludes the thesis by summarising the main contributions. It also draws attention to some of limitations along with possible progressions.

Note, four core chapters (Chapter 2-5) are introduced in a stand-alone and self-content way. Therefore, each of which has its context including related work, architecture, algorithms, discussions, evaluations matrices and experiments.

Chapter 2

Steganography to Protect IoTs Streams

This chapter answers the first research question discussed in Section 1.2. The main concerns about the privacy of the confidential Internet of Things (IoTs) end-point nodes, such as household identity, and the authenticity of transmitted streams are examined along with the limitations raised due to transplanting existing security algorithms to solve this issue. This chapter then highlights why steganography (stego) can be a candidate technique to take some of the challenges and what limitations should be taken into account. Section 2.1.1 introduces the main contributions of this chapter and how they have been accomplished. Section 2.2 highlights briefly the key related works and the efforts of other researchers to solve the issue. Sections 2.3 and 2.4 explain in detail our two novel models including the design, sensitive information protection, randomisation mathematical models, and the hiding and retrieval strategies. The evaluation of various characteristics of both models along with an overview comparison between them is introduced in Section 2.5. In Section 2.6, we present detailed experimental examinations of both models, their resultant effect on the actual IoTs high streams and a comparison with available techniques. Section 2.7 summarises this chapter.

2.1 Introduction

Recently, there has been great-interest in and an increase in the amount of remotely collected data [11]. The purposes of such collections is to monitor the following: environmental phenomenon, battlefield scenarios, surveillance, manufacturing automation, traffic screening and remote healthcare. The data is mainly collected using Wireless Sensor Networks (WSNs); these comprise large numbers of small IoTs devices that have limited computational capabilities and low battery power [34]. Commonly, the collected data includes: 1) normal readings (environmental or activities data), and 2) highly sensitive information (IDs, battlefield geometric location or secret nuclear facility features). This information is periodically sent via a predetermined spectrum (e.g. 2.4 GHz) [21]. However, the extraordinary amount of transmitted data through continuous surveillance and the massive demand on the spectrum reservation results in wireless communications issues, such as 'spectrum scarcity' [35].

To overcome these issues, a new wireless communication technology called Cognitive Radio (CR) has emerged [36]. It deploys a simple idea where the licensed spectrum can be shared by a Secondary User (SU) whenever the Primary User (PU) is idle, also known as white space. CR allows SU to sense the licensed bands and whenever the space is white, SUs can utilise these bands to improve the IoTs communication performance, throughput and reduce the interference between the applications that use identical or overlapping bands, such as Bluetooth and ZigBee at 2.4 GHz [21]. Therefore, tremendous effort is currently being spent on developing various standardisations to exploit this opportunity. For example, a Defence Advanced Research Projects Agency (DARPA) project, titled neXt Generation (XG), is focused on how unused spectrum technologies, such as television (TV) bands, can be utilised for US military applications [37, 38]. Also, CR technology has recently been implemented in various sets of IoTs applications such as smart homes, medicine and traffic screening [39–41]. Despite the obvious advantages, CR Networks (CRNs) cause many security issues for the transmitted IoTs streams in addition to the traditional

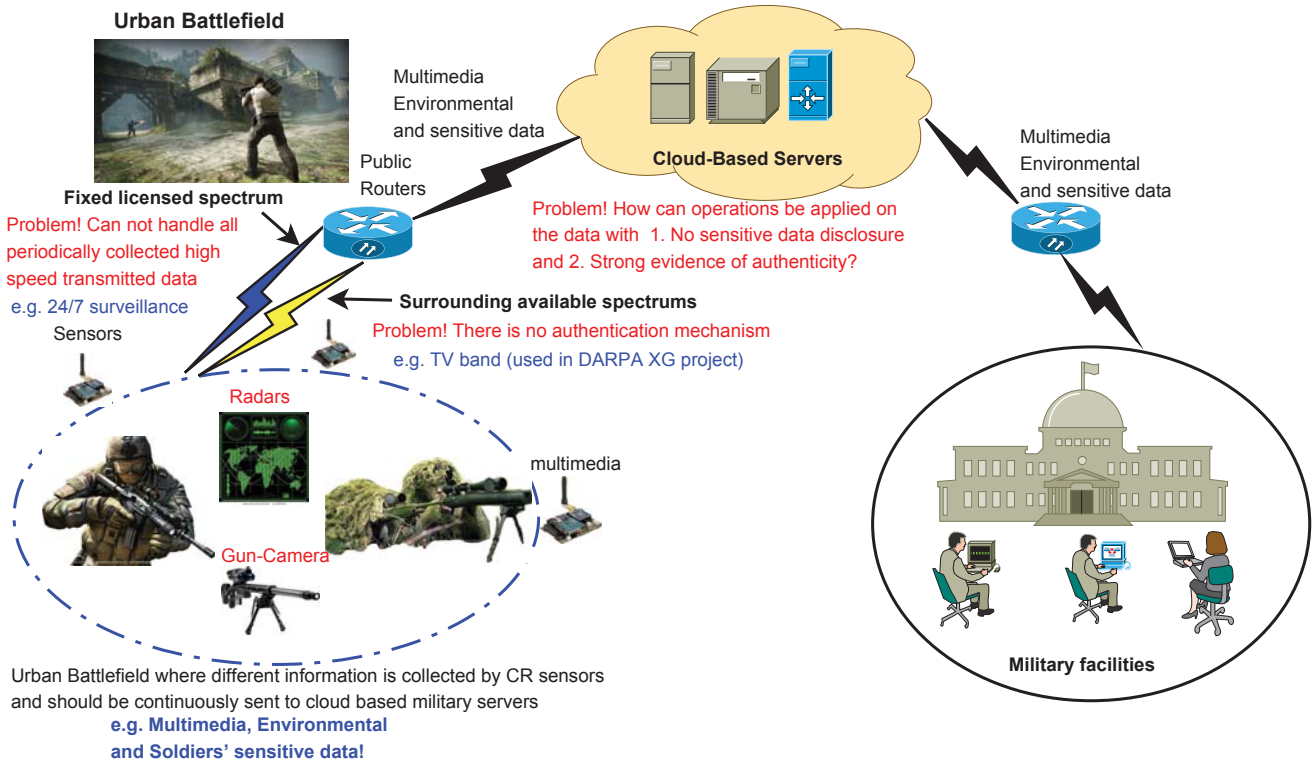

Figure 2.1: Main issues faced when different information (e.g. environmental, multimedia and soldiers' sensitive data) are periodically collected by IoTs devices in a battlefield area and should be sent to cloud-based servers for authorised management.

WSNs troubles which is illustrated in Figure 2.1 and can be categorised in the following way:

1. The confidentiality and the privacy of the transmitted sensitive content (e.g. soldiers' sensitive and geometric locations).

2. The authenticity of the collected normal IoTs readings because of the IoTs device presence in hostile areas and the possible-natural or malicious interference.

Although these problems have been discussed in traditional WSNs, we are compelled to target these issues in the context of IoTs today because: 1) they are rarely targeted in today's management model where the data is stored and processed by third parties' machines (i.e. Cloud Providers (CPs)), and 2) in using CRNs technology, the data may be sent in a spectrum that has no authentication mechanism, such as TV

broadcasting [42], as in the XG project (see Fig. 2.1). Therefore, this chapter proposes a novel solution for these issues based on the following questions:

- How can the IoTs transmitted sensitive information be protected without disrupting any possible operations at data aggregators and CPs?
- How can the authenticity of the transmitted normal readings be checked, especially if the data are sent in a spectrum that has no authentication mechanism, such as a TV band?
- Can both requirements be met without revealing the sensitive information to CPs?

To solve these issues, most of the early solutions relied heavily on traditional cryptography techniques, such as symmetric, asymmetric and digital signature [24–29, 43]. However, they suffer from two main limitations:

- The huge delay and overhead of the approaches that result from thousands or millions of mathematical operations in order to achieve high security, which usually cannot be handled by existing IoTs end-point devices capabilities (i.e. memory and power).
- Changing the form of all original data into a ciphertext makes applying operations on the data more difficult at aggregators (i.e. where the transmitted data may be collected and compressed) and at CPs.

To solve some of the traditional cryptography issues, a recent non-traditional cryptography technique called homomorphism has been used [44–46]. The advantage of this technique is that the encrypted data can be worked on at data aggregators and CPs points without revealing its meaning and thus provides strong end-to-end security. However, homomorphic techniques are still not feasible in practical applications, because their computational operations are very complex [22].

Steganography is another means of protecting sensitive information where a portion of a secret message, like a watermark, is hidden inside host data and can only

be retrieved by authorised users. The advantage of steganography over cryptography is that it requires much lower power and processing capabilities. Therefore, many solutions have been proposed using one of the steganography aspects called digital watermarking (i.e. protecting integrity) where a small fixed-length message (e.g. MAC) is hidden inside the host data [22, 47–51]. There are limitations in these digital watermarking techniques.

- They mainly provide strong integrity and so the receiver can extract the watermark and check the validity of the data, but they do not protect the privacy of the transferred sensitive information, thus raising a confidentiality issue.
- The watermark is embedded directly inside sensors' readings, which is called, 'hiding in the spatial domain'. This restricts the size of the hidden secret message and becomes a capacity issue. Also, the amount of distortion (the difference between the watermarked and the original forms) on the sensors' readings is very high, so the watermark should be removed whenever normal readings are used.

2.1.1 Contributions

This chapter proposes two novel steganographic algorithms: 1) to protect sensitive node information by hiding them randomly bit-by-bit inside normal IoTs streams using a generated key; and 2) to provide strong evidence of authenticity by sealing the normal transmitted readings. These algorithms will address the aforementioned concerns derived from the first research question in Section 1.2. To overcome the hiding capacity issue, two signal processing techniques called 'Walsh-Hadamard' and 'Wavelet' are used to transform the normal readings from their spatial domain to their frequency domain. This results in a set of decomposed values called 'coefficients' [52]. Our two models vary in terms of speed, security and the maximum size of confidential information that can be handled. This will be summarised in the evaluation section

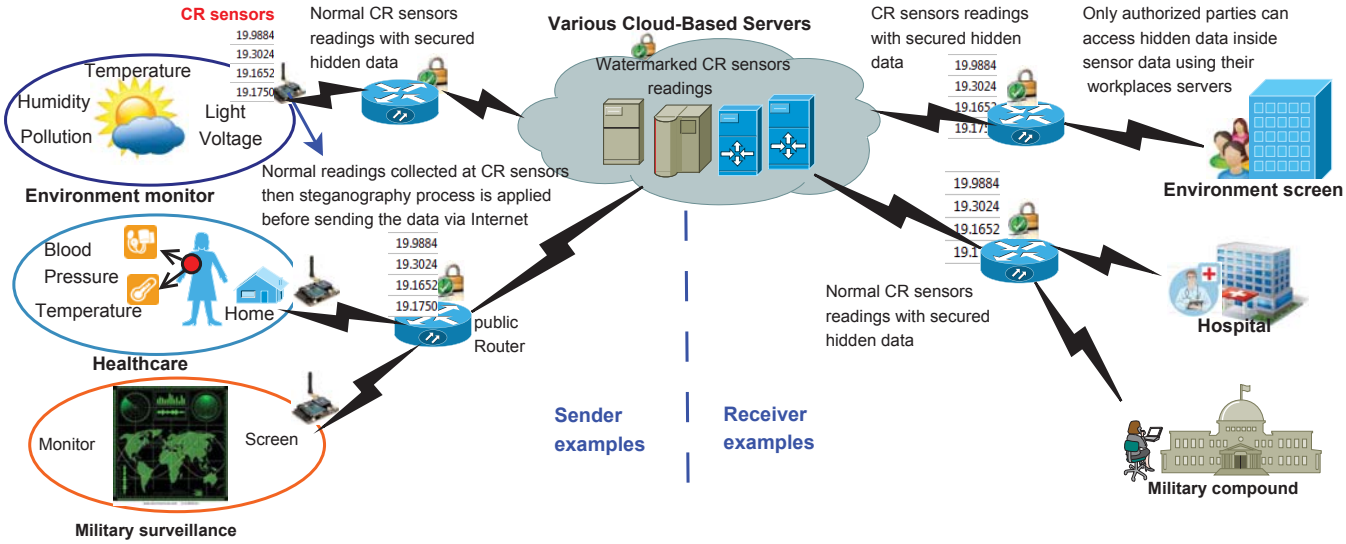

Figure 2.2: The main scenario of our proposed model where sensors' sensitive information is hidden inside normal readings and only authorised users can retrieve this confidential data.

of this chapter.

2.2 Related Work

Any solution proposed to protect sensitive transmitted IoTs end-point information should carefully consider the security, efficiency and capacity because of the nature of IoTs devices capabilities and their surrounding environment. However, most existing proposed solutions lack a suitable balance between these three features.

A majority of the solutions focused on the first aspect of security and ignored the other aspects, such as efficiency. For example, models in [53, 54, 57, 58] provide strong security by using classical cryptography techniques, known as asymmetric encryption. However, their efficiency is poor, because all the data should be decrypted whenever it is used.

The other stream of solutions has targeted this issue by using homomorphism, which is a new cryptography technique [46, 56]. Although they tried to have a reasonable balance between security and efficiency by using homomorphic encryption,

Table 2.1: Related Work Summary.

Protection Technique	Features	Comments
Traditional Cryptography [29, 53, 54]	• Use symmetric/asymmetric keys for encryption at sensors side • Aggregators and CPs should decrypt all data before operations • Receiver should decrypt all data before using them	• Low Confidentiality • Poor Efficiency • Key' management issue • Unlimited Capacity
Homomorphic Encryption [45, 46, 55, 56]	• Use homomorphic encryption at sensors side • Aggregators and CPs can work on the encrypted form • Receiver should decrypt all data before using them	• Strong Confidentiality • Is not feasible in practical applications • Low Efficiency • Unlimited Capacity
Digital Watermark [22, 49–51]	• Embed secret message (e.g. MAC) at sensors side • Using the spatial domain for hiding • Aggregators and CPs can work on the normal readings • Receiver should remove the watermark before using the data	• Strong Integrity • No Confidentiality • High Efficiency • Low Capacity of embedded messages with High Distortion

this non-traditional cryptography is still not feasible in practical applications for its complexity [22].

The third stream of solutions applied a well-known technique used in multimedia, known as 'digital watermarking' to guarantee the authenticity of the transmitted data as in models [22, 49–51]. However, these solutions mainly focus on the integrity of the transmitted data, but they neglect the end-to-end privacy preservation of the sensitive information, meaning its a security issue. Secondly, only a few bits can be embedded into the transmitted readings, thus creating capacity issue. Thirdly, the classical watermarking technique results in a certain amount of overhead in the transmitted data as can be seen in [51].

Table 2.1 summarises the work that can be categorised into three classes based on the techniques used: traditional cryptography, non-traditional cryptography, and digital watermarking.

2.3 Model 1(MD1): Walsh-Hadamard based Steganography

In this section, a new fast steganography algorithm is introduced which takes into consideration a reasonable balance between security and efficiency in such a way that: 1) there is little effect on the normal IoTs streams, so the readings can be used without removing the stego (i.e. hidden sensitive information) and 2) it is impossible for illegitimate parties to extract the hidden information without using an appropriate key. The algorithm relies on a simple and fast signal processing technique called Walsh-Hadamard.

The operations at the remotely distributed IoTs end-point device can be categorised into four stages:

2.3.1 Obfuscation

The goal of this stage is that the encryption and hiding processes of the sensitive information in the normal IoTs sensors' readings must be completely random and different among IoTs devices, and so disallow unauthorised parties from retrieving them properly. Therefore, a security key is generated for every distributed IoTs node and will be known only to the end receiver of the data. This key has two main tasks:

1. Encrypt the sensitive information (e.g. IDs and geometric location data) before the hiding process using Exclusive OR (XOR) operation, which is very fast and suits the IoTs small devices technical capabilities. This can be shown in Eq 2.1

$$\widetilde{S} = S \bigoplus KEY \tag{2.1}$$

   where $\bigoplus$ is a XOR operation, S is the original sensitive information and $\widetilde{S}$ is the encrypted form.

2. Generate the sequence of selected coefficients that will be used to hide the confidential IoTs devices' information. Then, it will be shifted by one character and will generate the second layer of selected coefficients and so on. This is shown in Eq 2.2 and will be fully comprehensible after reading Section 2.3.3.

$$\widetilde{N} = f_x(KEY) \tag{2.2}$$

   where $\widetilde{N}$ is the generated sequence of coefficients.

2.3.2 FWHT Transform

Walsh-Hadamrd Transform (WHT) is a well-known process that is used to decompose a signal into a set of coefficients representing its frequency components [59, 60]. The significance of that is the resultant coefficients can be classified into: 1) low-sequence coefficients which represent most of the signal energy, and 2) high-sequence coefficients representing the less important parts of the signal. The advantage of this technique

is that the original signal can almost be reconstructed from only the low sequence components.

For better understanding, Fig 2.3 shows an example of IoTs devices' readings. a) Plot for more than 500 temperature samples. b) Plot for the resultant coefficients after applying WHT, which clearly shows the most energy is in the low sequence coefficients from 0 to <50, whereas others are less important. Accordingly we erased all coefficients from >50 to 512 to show their effect on the reconstructed samples. c) Plot the reconstructed original temperature samples from only <50 coefficients. This figure demonstrates the flexibility and the capacity that will be derived from these coefficients. This inspired us to use signal transformation techniques to hide more sensitive information related to IoTs end-point data without increasing the actual IoTs devices' readings.

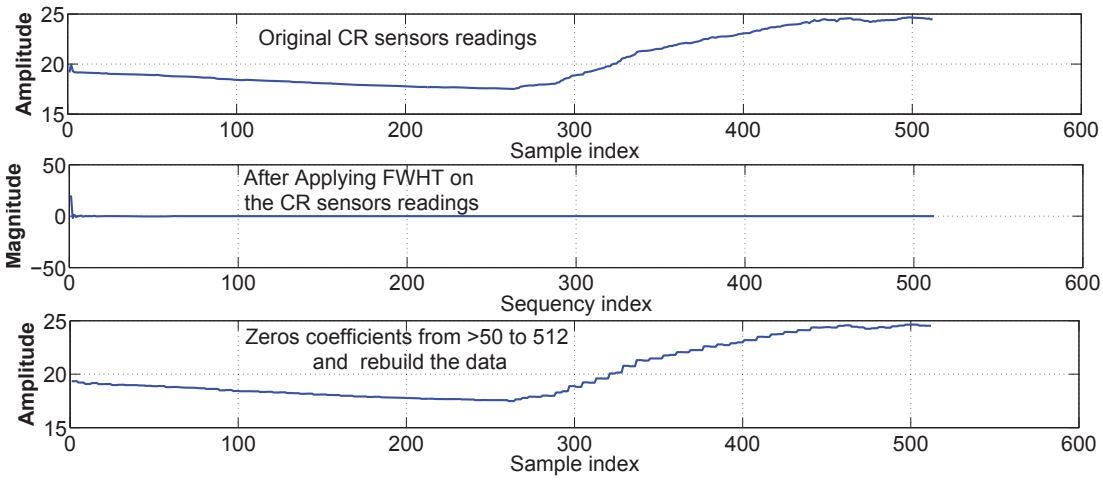

Figure 2.3: IoTs device' readings: (a) direct plot, (b) after applying Fast Walsh-Hadamrd Transform (FWHT), and (c) rebuilt form after zeros more than 90 % of FWHT coefficients.

Additionally, there is a fast version of WHT algorithm that will provides the computational complexity $NLogN$ [61], whereas the complexity of the commonly used WHT is $O(N^2)$ [62]. Therefore, FWHT is used in our algorithm and is shown in Eq 2.3.

$$y_n = \frac{1}{N}\sum_{i=0}^{N-1} x_i FWHT(n,i), n = 1, 2, ..., N-1 \tag{2.3}$$

where y_n is the resultant coefficient, x_i is the original sample value and $FWHT(n,i)$ is the applied transformation [61, 62].

FWHT simply works by applying a Walsh generated matrix that is correlated to the number of samples. The matrix values are +1 and -1. The order of rows in this matrix can be 'Sequence', which is used in signal processing, Hadamard - is used in controls applications or Dyadic is used in mathematics. A simple FWHT matrix for only four samples is shown in Eq 2.4 [63].

$$\begin{pmatrix} s_1 \\ s_2 \\ s_3 \\ s_4 \end{pmatrix} \cdot \begin{pmatrix} +w_{11} & +w_{12} & +w_{13} & +w_{14} \\ +w_{21} & +w_{22} & -w_{23} & -w_{24} \\ +w_{31} & -w_{32} & -w_{33} & +w_{34} \\ +w_{41} & -w_{42} & +w_{43} & -w_{44} \end{pmatrix} = \begin{pmatrix} c_1 \\ c_2 \\ c_3 \\ c_4 \end{pmatrix} \tag{2.4}$$

where S_s is the original IoTs device stream, w_s is the FWHT matrix value and c_s is the resultant coefficient values.

In our research, FWHT is chosen for two main reasons: 1) the original IoTs sensors' readings can be accurately reconstructed from only a few coefficients, allowing others to be used freely to hide a reasonable amount of sensitive information, and 2) FWHT uses less storage space, is faster to calculate and consumes fewer resources than many other transformation techniques, such as Fast Fourier, Chirp Z and Frequency Response of Digital Filter, because it uses only real additions and subtractions [52].

Therefore, in this work, FWHT is applied to different real-time collected IoTs devices' readings (e.g. temperature, humidity, light and voltage) and the resultant coefficients will be reshaped to a two-dimensional (2D) matrix. The first few low-sequence coefficients will not be manipulated because they represent the most important part of the IoTs devices' readings. On the other hand, several bits will be changed in the remainder of FWHT coefficients - the steganography level. Also, to guarantee the minimum acceptable amount of distortion to the actual IoTs devices' readings, we performed many experiments to select an appropriate steganography

level to demonstrate how many bits can be hidden in the less important coefficients. This is shown in Fig 2.4. From the results of those experiments, about 5 bits will be hidden in the randomly-selected high-sequence coefficients.

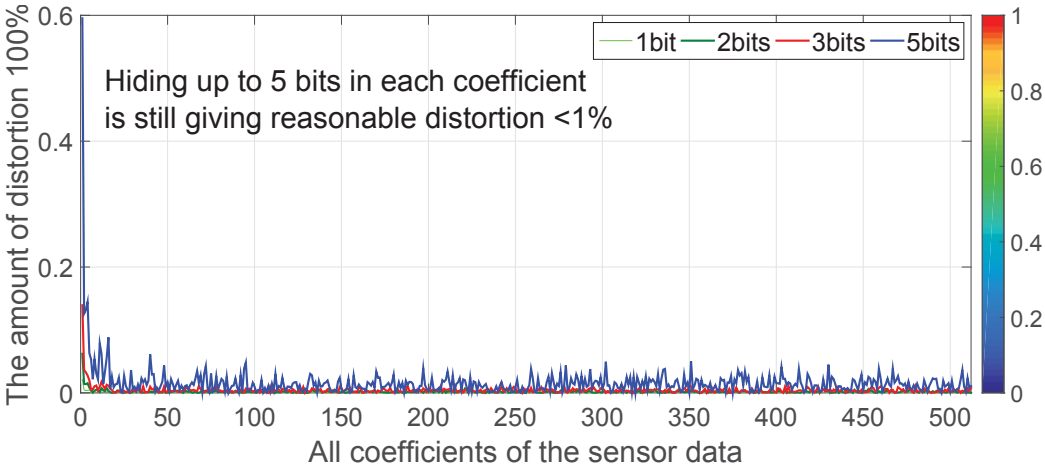

Figure 2.4: Effect of applying steganography using different levels 1bit/2bits/3bits/5bits on the resultant distortion for each coefficient.

2.3.3 Hiding in Walsh-Hadamard Matrix

At this stage, the sensitive information will be hidden inside the resultant coefficients, after applying FWHT to the IoTs devices' readings. However, to guarantee a high level of security and to prevent unauthorised parties from accessing this information, three layers of security are implemented. These are:

1. the sensitive information is encrypted using a security shared key
2. the key is used to scramble and reshape the resultant coefficients from a vector to a matrix of M-by-N
3. the key is used to generate the selected coefficients' order in a vector space.

This will guarantee that only an authorised receiver who has the security key can extract and decrypt the sensitive information properly.

The detailed process of hiding is shown in Fig 2.5. After applying FWHT to the normal IoTs readings, the resultant coefficients are scrambled and reshaped to

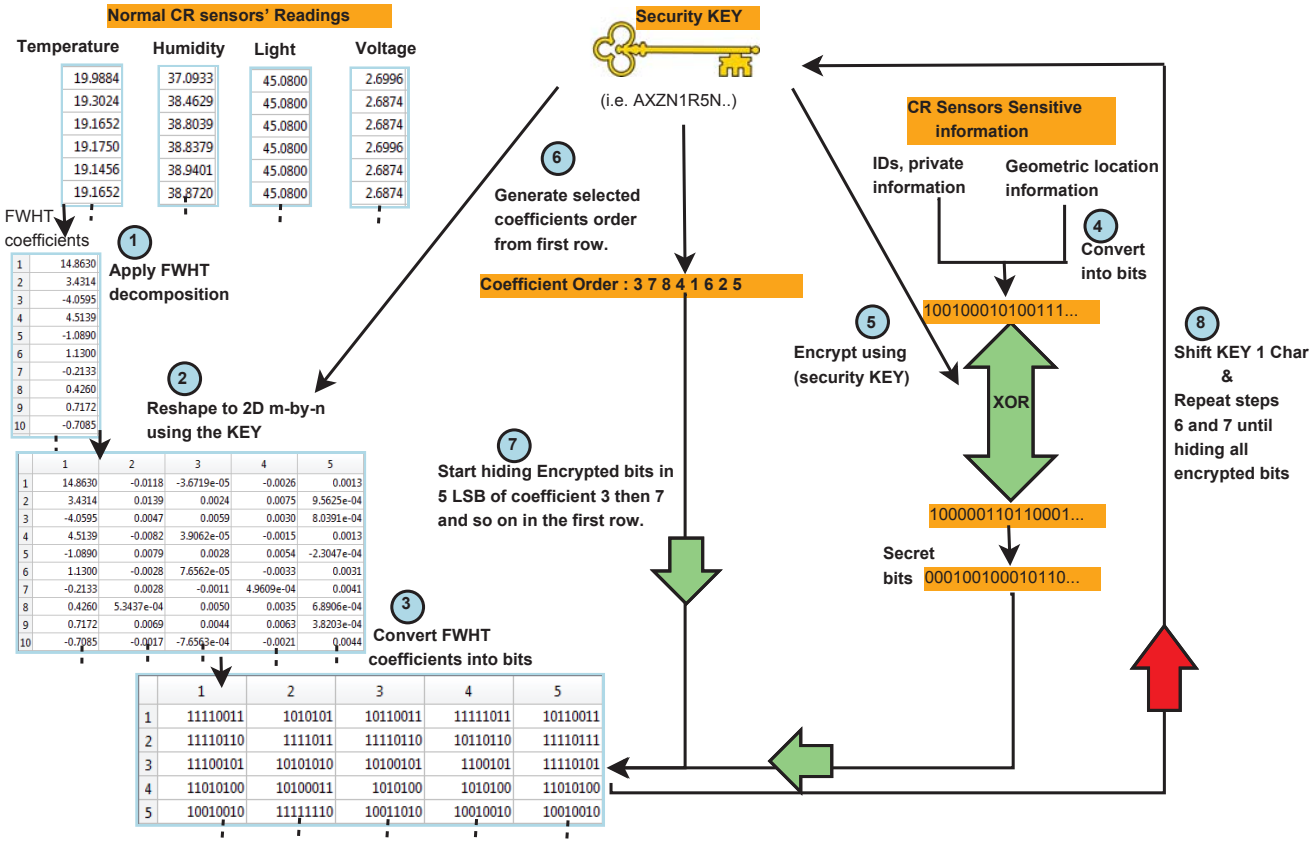

Figure 2.5: Block diagram shows the steps of hiding sensitive IoTs device information inside their normal readings.

M-by-N matrix. The key is then used to encrypt the sensitive information after converting both to bits. Next, the key will be used to generate the selected order of first row coefficients, which is based on the American Standard Code for Information Interchange (ASCII) positions of the key character set. After that, the secret bits will be embedded corresponding to this order. After finishing all the selected coefficients, the key will be shifted by one character to generate the second row of selected coefficients; this will be used to hiding again, and so on.

Fig 2.6 shows a simple example of how the shared key can be used to generate the order of selected coefficients for a single row. It starts by converting the key into ASCII and a default position value will be given. Then, order the key ASCII in ascending order and another position value will be assigned (i.e. ascending position order). Finally, return the key ASCII to its original order using the default position order

(to avoid having two different keys resulting in a similar sequence of numbers).The resultant ascending position order is regarded as the 'selected coefficients order'. In our algorithm, the key length will be more than 32 symbols in length.

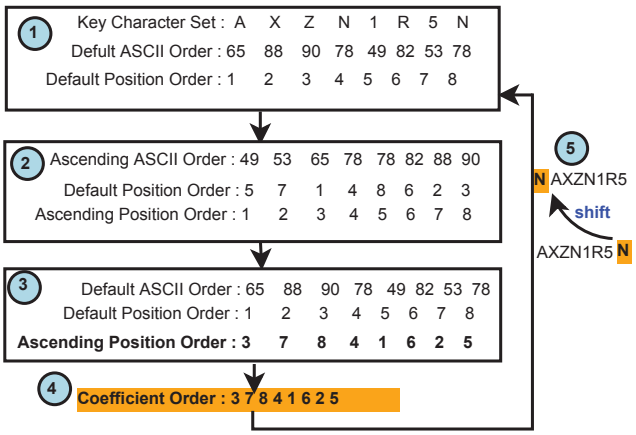

Figure 2.6: Block diagram shows how a selected coefficients order is generated from the key in five steps. The complexity is a vector space.

The detailed hiding algorithm is demonstrated in Algorithm 1. It begins by initialising the required variables. FWHT is then applied and the resultant coefficients will be shifted and processed into an integer format. Next, by using the key, the algorithm will shuffle the coefficients into 2D. The secret bits are then hidden in different coefficients. Finally, the 2D coefficients will be reshuffled and rescaled to their original format, then the inverse FWHT is implemented to produce the watermarked readings which is then transmitted.

2.3.4 Inverse FWHT Re-composition

Following the hiding process, the resultant coefficients are called stego (or sealed) coefficients. At this stage, the stego coefficients will be re-reshaped and the inverse FWHT applied to convert IoTs readings from their frequency domain to their original time domain. The result is a reconstructed form called stego IoTs streams, meaning it contains hidden confidential information, which is quite similar to the original IoTs sensor readings. The beauty of that is even the stego IoTs sensor streams can be used as the original form; however, only authorised receivers with a security key can

Algorithm 1 The hiding algorithm

```
i, j, k : counters
m, n : 2D matrix size
coef : FWHT coefficients matrix
S : secret bits
Apply FWHT on host data
Rescale coef → int
shuffle coef → 2D
/*Start hiding*/
while counter < EoF coef do
  for i = 1 : n do
    for j = 1 : m do
      Hide 5 bits of S(k) → coef(i, j)
      if EoF S then
        Start again
      end if
    end for
  end for
  counter = counter + 1
end while
Re − shuffle coef → original form
Re − rescale coef → original form
Apply inverse FWHT to host data
return watermarked readings
```

extract the hidden information, such as IDs and geometric location information, and verify them. The inverse FWHT can be defined by Eq 2.5

$$x_i = \frac{1}{N}\sum_{i=0}^{N-1} y_n IFWHT(n,i), i = 1, 2, ..., N-1 \tag{2.5}$$

where x_i is the original sample value, y_n is the resultant coefficient from the decomposition process and $IFWHT(n,i)$ is the inverse transformation [61, 62].

2.3.5 Retrieval from Walsh-Hadamard Matrix

To properly retrieve and decrypt the hidden sensitive information, the receiver must have the security key. The process is almost identical to the hiding process except that it extracts the bits instead of hiding them. Fig 2.7 shows the detailed process. First, we apply FWHT onto the IoTs sensor readings. The key is then used to reshape the FWHT coefficients into a 2D matrix. Next, the key is used to generate the

selected order of the first row and start extracting the sensitive bits from each selected coefficient. The key will then be shifted and will generate the second layer of selected coefficients, and so on. Finally, by using the key, we can decrypt the secret bits and verify the resultant information.

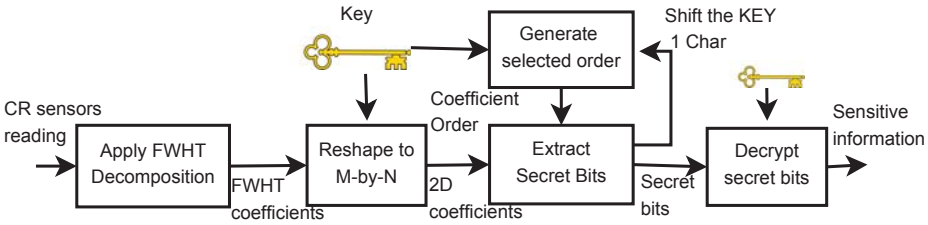

Figure 2.7: The main steps to retrieve the sensitive information.

2.4 Model 2 (MD2): Wavelet-based Steganography

In this section, another novel steganography algorithm is proposed. It focuses on strengthening the security by using a two-dimensional signal processing technique called 'Wavelet'. Our proposed hiding mathematical model takes the advantage of the obtained wavelet sub-bands tree that complicates the time required for retrieving the hidden bits illegally.

The operations on the IoTs device's side can be classified into the following stages.

2.4.1 Strong Encryption

The goal of this stage is to ensure that the encryption and hiding processes of the confidential information in the normal IoTs device's readings must be strong, completely random and different among IoTs devices, and so disallows illegitimate parties from retrieving them accurately. Therefore, a security key will be generated for every IoTs device and will be known only to the end receiver of the data (e.g. operation centres). This key has two main tasks:

1. Encrypts the confidential information, such as IoTs device IDs and geometric location data, household name, Date of Birth (DoB), address and total watts

consumption, before the hiding process using Advanced Encryption Standard (AES) encryption - this is very strong and suits the IoTs device's technical capabilities. This can be defined in Eq 2.6

$$f_x(O, KEY) \Rightarrow \widetilde{E} \tag{2.6}$$

where f_x is AES operations, O is the original confidential information and $\widetilde{E}$ is the encrypted form.

2. Generates two random sequences of numbers to build a random matrix that will be used to hide the sensitive information of IoTs devices. This will become fully comprehensible after reading Section 2.4.3.

2.4.2 Wavelet Decomposition

Wavelet Transform (WT) is a well-known technique used in signal and image processing where the content, such as biomedical signal, is transformed from its spatial domain into its frequency domain to identify the most and the least significant parts of the signal [64]. More precisely, it is a linear transformation that is performed on the given signal which leads to decomposing it into differing values that represent its frequency components at a given time called coefficients. WT is shown in Eq 2.7

$$C(i, j) = \int_{-\infty}^{\infty} f(t)\psi(i, j)dt \tag{2.7}$$

where i and j are positive integers that represent transform parameters. C represents resultant coefficients. ψ is wavelet function [65]. The WT can be used in two ways which are either Discrete and Continuous. The Discrete Wavelet Transform (DWT) is preferred in most applications, because the produced and analysed real-life information comes in discrete numbers rather than continuous functions [66].

To conduct DWT decomposition, high-pass and low-pass band filters are applied on the original signal. Consequently, two sub-signals called 'bands' are obtained. The first sub-band is the low frequency components that represents the approximation of the original signal. The second is the high-frequency components that represents the

detailed coefficients. When this process is repeated multiple times, it is known as multi-level wavelet decomposition [65, 67]. DWT is defined in Eq 2.8.

$$D(a,b) = \sum_{a} \sum_{b} X(a)\Phi_{ab}(n) \tag{2.8}$$

where $D(a,b)$ represents DWT coefficients, a and b represent the shift and scale transform parameters and $\Phi_{ab}(n)$ represents the base-time wavelet function that is shown in Eq 2.9.

$$\Phi_{ab}(n) = 2^{-a/2}\Phi(2^{-a}n - b) \tag{2.9}$$

In this work, DWT is chosen for two main reasons: 1) it produces a tree of sub-bands which helps with strengthening the security of embedding while not increasing the original size of the data; where the total number of coefficients from all decomposed sub-bands are equal to the original signal samples, and 2) the signal can almost be reconstructed from only the most significant coefficients - the approximation sub-band coefficients - allowing free use of the detailed sub-band coefficients in order to hide confidential information with minimum amount of distortion on the original transmitted readings (as in electricity consumption). This inspired us to use wavelet to hide more confidential information related to IoTs device securely without effecting the actual readings of the IoTs devices.

Therefore, in this work, the 5-level wavelet packet decomposition will be applied on different real-time readings from IoTs devices readings (e.g. watts, heat-index, wind chill, humidity and temperature), which results in 32 sub-bands as shown in Fig. 2.8. A wavelet family called 'Daubechies' with the order 2 ($db2$) is chosen in the decomposition process, because its performance in analysing discontinuous-disturbance-dynamic signals has already been proven to be perfect [68]. To achieve minimum amount of distortion, the low frequency sub-bands coefficients (from 1 to 16) will not be changed because they represent the most significant parts of the IoTs readings. On the other hand, a number of bits will be manipulated into the rest of the detailed sub-bands coefficients - this is called the steganography level, because the confidentiality of the hidden sensitive information and the imperceptibility

to human senses are our top priorities[69]. To guarantee the minimum acceptable distortion on the actual readings, many experiments have been conducted to select an appropriate steganography level (i.e. how many bits can be embedded in the least important sub-bands). Consequently, about five bits will be hidden in the randomly-selected high-frequency sub-bands coefficients.

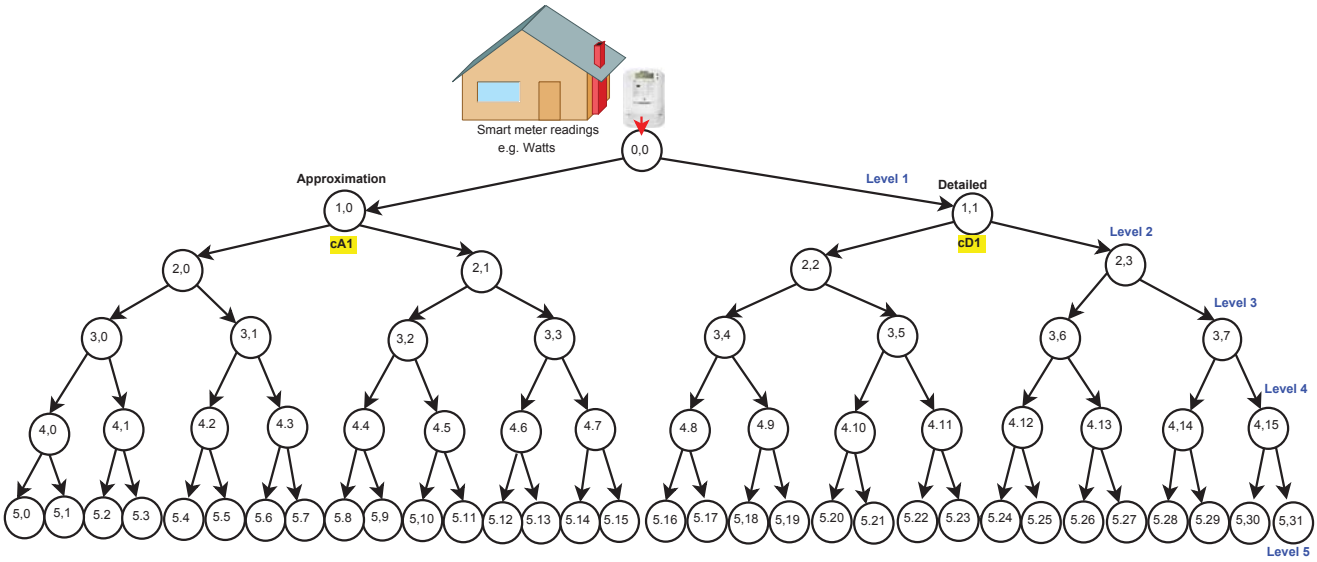

Figure 2.8: Decomposing IoTs streams (e.g. smart meter) into 32 sub-bands.

2.4.3 Embedding in Wavelet Tree

At this stage, the confidential information from households will be hidden inside the resultant sub-bands coefficients tree after applying DWT to the IoTs device, namely from the smart meter. However, to guarantee a high level of security and to prevent unauthorised parties from retrieving this information, two layers of security are implemented. These are explained below:

1. the confidential information is encrypted using the security key (i.e. shared key)

2. the key is used to generate the selected coefficients order from different rows in the form of a random 2D space $M \times N$

This will guarantee that only an authorised receiver who has the security key can extract and decrypt the confidential information properly.

Generating Hiding Matrix: To achieve the goal of hiding the secret bits in an entirely random way that is different among IoTs devices, the security key is used to generate two random sets of numbers as defined in Eq 2.10

$$Z \Leftarrow \begin{cases} \widetilde{c} = f_x 1(K) \\ \widetilde{r} = f_x 2(K) \end{cases} \tag{2.10}$$

where $\widetilde{c}$ and $\widetilde{r}$ are the generated sequences of numbers. The combinations of $\widetilde{c}$ and $\widetilde{r}$ is used to build a 2D $M\ X\ N$ matrix Z.

$$Z\{m,n\} = \begin{bmatrix} r_1, c_1 & r_1, c_2 & \cdots & r_1, c_n \\ r_2, c_1 & r_2, c_2 & \cdots & r_2, c_n \\ \vdots & \vdots & \ddots & \vdots \\ r_m, c_1 & a_{m,2} & \cdots & r_m, c_n \end{bmatrix} \tag{2.11}$$

where Z is a 2D generated matrix (see 2.11). m represents the number of rows, and n represents the number of columns which is < 32. (r_m, c_n) which represents a position of a sub-band coefficient in the wavelet tree.

There are two crucial 2D matrices in our algorithm that can be seen in Fig 2.12. The first is matrix C which corresponds to the readings' coefficients. The second is matrix Z which corresponds to the generated random order that should be followed to hide the secret bits. Here, the size of the 2D matrices is $m \times n$. Therefore, our algorithm relies on the total number of readings, the decomposition level and the total number of sub-bands used in hiding process. This is shown in Eq 2.12.

$$(m,n) = \left\{ \frac{T_R \times 2}{(2^L)_2} \right\}, \left\{ \frac{T_R}{(2^L)} \right\} \tag{2.12}$$

where T_R is the total number of readings. L represents the decomposition level used.

For example, assume that the total number of IoTs readings (in watts) contains 1024 samples and the used decomposition level is five (see Eq 2.13). Consider:

$$\left\{ m = \frac{1024 \times 2}{(2^5)_2} \Rightarrow 16 \right\}, \left\{ n = \frac{1024}{(2^5)} \Rightarrow 32 \right\} \tag{2.13}$$

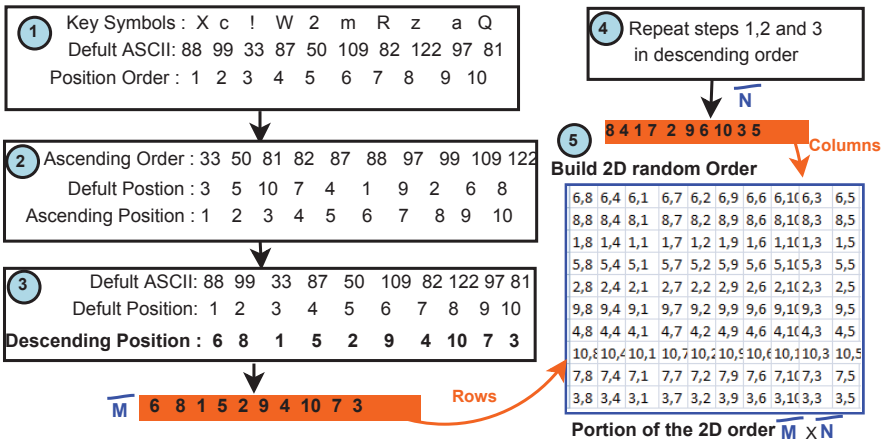

Figure 2.9: This example demonstrates how the 2D hiding matrix order is generated from the key in five steps.

For a better understanding, Fig 2.9 shows a simple example of how the key can be used to generate the 2D hiding matrix order (Z). First, the key is converted into ASCII and a default position value will be allocated. The key ASCII is then ordered in descending mode and another position value will be given (i.e. descending position order). Next, the key ASCII is returned to its original order using the default position order to avoid having two different keys resulting in an identical sequence of numbers and the resultant descending position order is regarded as $\widetilde{r}$. A similar process is repeated to generate $\widetilde{c}$, except that ascending order is used instead of descending. Finally, Z is built from the combination of r and c. However, in our algorithm, the key length contain more than 32 symbols.

Embedding Process: The detailed process of hiding is shown in Fig 2.10, 2.11 and 2.12. DWT is applied to the normal IoTs device's readings, which result into a 32 sub-bands coefficients tree. Only non-significant sub-bands coefficients (from 16 to 31) will be split, rescaled and converted into bits be used in the hiding process (see

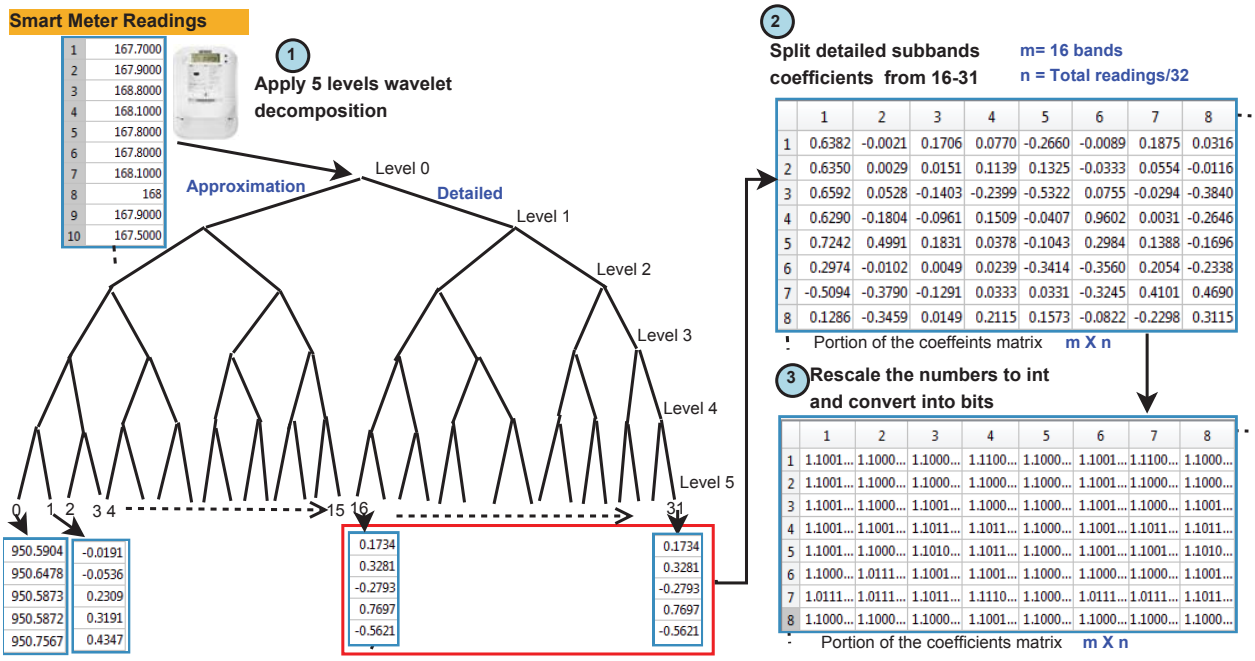

Figure 2.10: Block diagram shows how an example of IoTs readings are decomposed, split, rescaled and converted into bits.

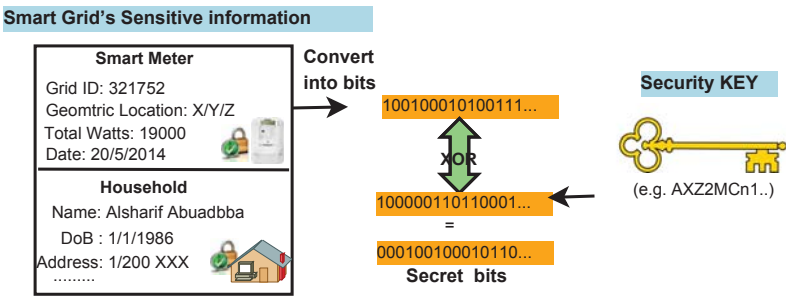

Figure 2.11: An example of how confidential information is encrypted before hiding.

Fig 2.10). The key is then used to encrypt the confidential information (see Fig 2.11). Finally, the secret bits will be hidden in the sub-bands coefficients tree corresponding to the early generated 2D matrix order Z (see Fig 2.12). If all selected coefficients are used and if there are more secret bits, the key then will be shifted 1-character to generate another two random sequences of numbers that will be used to hide, and the process repeats.

Algorithm 2 clearly demonstrates the steps followed in the hiding process in a pseudo-code. However, to make sure that all high-frequency coefficients are ready for

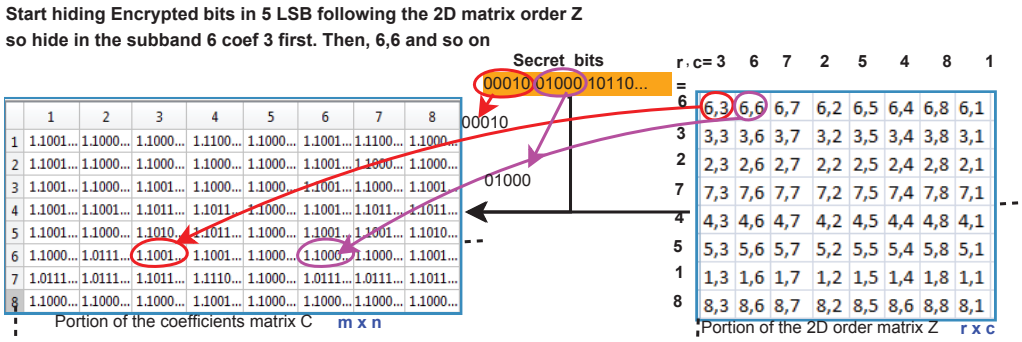

Figure 2.12: Block diagram shows how the secret bits are hidden corresponding to the 2D matrix order Z.

the steganography, they should be rescaled to a positive integer format before the process.

Algorithm 2 The hiding algorithm

1: *Apply* ⇒ 5 level wavelet decomposition
2: *Split* ⇒ high frequency sub-bands from 16 to 31
3: *Add* ⇒ (lowest value + -1) to all coefficients
4: *Multiply* ⇒ all coefficients by 10000
5: *Convert* ⇒ coefficients from *double* to *integer*
6: **while** *counter* < *Total coefficients* **do**
7: *Start* hide secret bits into 2D matrix
8: **end while**
9: *Convert* ⇒ coefficients from *integer* to *double*
10: *Divide* ⇒ all coefficients by 10000
11: *Subtract* ⇒ (lowest value + -1) from all coefficients
12: *Recombine* ⇒ high and low frequency sub-bands
13: *Apply* ⇒ Inverse 5 level wavelet re-composition
14: *return* ⇒ stego readings

For clarification, Fig 2.10 and 2.12 show a simple example of a split sub-bands coefficients of size 8×8. However, the size of the 2D matrix has no fixed size and can be changed based on the total number of decomposed readings and the level of decomposition. For example, if the total number of readings is 512, and 5-level wavelet decomposition is applied, a 2D matrix will be generated. The columns represent the 32 sub-bands and each has its own row that is a list of coefficients (i.e. each sub-band has a list of 16 coefficients). Only the high-frequency sub-bands that range from 16 to 31 will be used. Therefore, 16 sub-bands are split and, according to our example,

each has its own 16 coefficients. Due to space limitations in the illustrated figures, only size 8×8 portions are on view.

2.4.4 Wavelet Reconstruction

Following the hiding process, the resultant sub-bands coefficients are called stego coefficients. At this stage, the stego coefficients will be re-embedded into the 32 sub-bands coefficients tree and the inverse DWT is applied to convert IoTs readings from their frequency domain to their original time domain. The result of that is a reconstructed form called 'stego IoTs readings' (i.e. they contain hidden confidential information) which is similar to the original IoTs stream. The beauty of that is even the stego IoTs readings can be used as the original form; however, only authorised receivers with a security key can extract the hidden information and verify them (e.g. grid IDs, household name, an individuals DoB, address and the total watts used). The inverse DWT is defined by Eq 2.14

$$X = \sum_{a} \sum_{b} D(a, b) \Phi_{ab}(n) \tag{2.14}$$

where X is the original form of data.

It should be noted that the focus of this chapter is to protect the IoTs confidential information and to seal the accumulated readings. However, to improve the efficiency of communication and reduce the payload, compatible size reduction techniques may be applied before transmitting the stego readings [68, 70]. The only condition is that the stego readings must be fully recovered at the receiver side, because losing a single bit may result in losing a portion of the hidden secret bits. This will be discussed further in detail in Chapter 4.

2.4.5 Extracting from Wavelet Tree

To accurately retrieve and decrypt the hidden confidential information, the receiver must have the security key. The process is roughly identical to the hiding process except that it retrieves the bits instead of hiding them. First, DWT is applied to

the IoTs readings which results in 32 sub-bands tree. Only the detailed sub-bands coefficients will be split and used in the retrieving process. Next, the key is used to generate two random sequences of numbers which are used to build the $M \times N$ matrix (Z). The secret bits will then be retrieved corresponding to this order. After that, the key will be shifted into 1-character to generate another two random sequences of numbers that will be used to extract information, and so on. Finally, by using the key the secret bits will be decrypted and the resultant information can be verified.

To evaluate the accuracy of the extracted information, the secret bits have been carefully checked after every retrieval process using Bit Error Ratio (BER) which can detect any data loss even down to one bit[71]. From our simulations, all BERs were 0 which means that all secret bits are recoverable.

$$BER = \frac{B_r}{B_T} \times 100\% \tag{2.15}$$

B_r is the number of erroneous bits and B_T is the number of bits.

2.5 Evaluation

In this section, the proposed algorithms will be evaluated in terms of the key strength, security of the hidden data, the size of the embedded sensitive data and the used distortion measurements.

2.5.1 Key Strength Analysis

The security of the algorithms proposed in this work requires that the recipient know in advance the full IoTs devices readings and the security key, otherwise the confidential hidden data cannot be extracted and decrypted properly.

The most significant parameter is the security key. It is used to provide two layers of security: 1) by encrypting confidential information and 2) by generating a random order of coefficients in the form of a vector and 2D $M \times N$ matrix to hide the encrypted bits randomly (see Fig 2.6 and 2.11). Therefore, this key should be kept secret and known only to two parties: 1) the sender (e.g. smart meter) where the key should be

Table 2.2: Examples of key lengths and possible combinations

Key length	Character set	Probabilities
64	US-ASCII	7.2e+134
64	UTF-8	1.3e+154
64	UTF-16	∞
128	US-ASCII	5.2e+269
128	UTF-8	∞
128	UTF-16	∞

burned and used whenever the collected IoTs device's readings are sent, and 2) the receiver (e.g. operation centres) who can properly retrieve and check the validity of the hidden sensitive information, whereas only the stego IoTs device's readings can be seen by other parties (e.g. public cloud providers). In our research, the key is generated and is kept secret at the IoTs end-point and at the recipient side. Our algorithms' key strength can be quantified as the number of entropy bits H (see Eq 2.16) where 2^H is the number of possibilities that should be exhausted by an intruder during a brute-force attack.

$$H = log_2 N^L \tag{2.16}$$

L is the symbol's length and N symbolises the probabilities of these symbols. Table **??** displays examples of different key lengths, character sets and the total number of their probabilities. Therefore, the longer the key and character set are, the stronger the algorithm becomes.

2.5.2 MD1: Walsh-Hadamard based Steganography

Unauthorised retrieval: The key is used to generate a hiding order in one-dimensional vector format. Therefore, to protect the hidden and secret information from being extracted without the key, the total number of combinations after applying FWHTs on the host data (i.e. normal IoTs device readings) should have a suitable size (see Eq 2.17).

$$T = \sum_{i=1}^{m} R! \times \sum_{j=1}^{n} C! \times N^L \tag{2.17}$$

where T is the total number of combinations, R and C are the number of rows and columns from the reshaped coefficients matrix using the generated vectors.

Hidden Data Size: The total amount of hidden data mainly relies on the total number of IoTs device's readings (e.g. watts) and the Walsh-Hadamard resultant one-dimensional coefficients (see Eq 2.18).

$$b = \sum_{i=1}^{t} v \times l \tag{2.18}$$

where b is the total number of hidden bits, t is the total coefficients, v is the selected values, and l is the number of hidden bits in each Walsh-Hadamard resultant values.

2.5.3 MD2: Wavelet-based Steganography

Unauthorised Retrieval Analysis: The key is employed to create a hiding order in 2D format to match the resultant wavelet tree. Therefore, to prevent retrieving the hidden, confidential information without the key, after the DWT decomposition of the IoTs devices readings host data, the 32 sub-bands coefficients tree should have a suitable size (e.g. > Key length)(see Eq 2.19).

$$T = \sum_{i=t}^{r} R! \times \sum_{j=t}^{c} C! \times N^{L} \tag{2.19}$$

where T is the total combinations' number, R and C are the rows and columns numbers from the 32 sub-bands coefficients tree and t is the minimum number of sub-bands coefficients that can be used from each row.

Assume smart meter's readings (in watts) comprise 1024 samples in length, and the 32 sub-bands coefficients matrix are a 32×32 size after applying DWT. The assumed threshold (t) is 32×16, the key length is 128 and its character set is UTF-8 (see Eq 2.20).

$$T = \sum_{i=1}^{32} 32! \times \sum_{j=16}^{32} 32! \times 256^{128} \Rightarrow T = \infty \tag{2.20}$$

Consequently, this proves that the ability to properly retrieve and decrypt the intended confidential information in a reasonable time is highly improbable.

The Size of Hidden Data: The total amount of hidden data relies on the total number of IoTs device's readings (e.g. watts) and the steganography level (see Eq 2.21).

$$b = \sum_{i=1}^{t} \frac{n}{2} \times B \tag{2.21}$$

where b is the total number of hidden bits, t is the total coefficients' number in each sub-band after decomposing the original samples, n is the total sub-bands number, B is the steganography level, that is the number of hidden bits in each sub-band's coefficient.

This means that after decomposing the normal readings with 5-level DWT , the resultant coefficients will be distributed under 32 sub-bands. Only high-frequency sub-bands from 16 to 31 are split resulting in a 2D matrix. The total number of columns represent the maximum sub-bands' number of 16, and the total number of rows represent the maximum number of resultant coefficients under each sub-band. In every coefficient, several bits will be hidden at the steganography level; therefore, the longer the total number of readings, the larger the size of the hidden data. We conducted intensive experiments to achieve a reasonable balance between the size of hidden bits in each coefficient and the resultant distortion as explained in Section 2.5.3. Based on that, about 5 bits will be hidden in each coefficient. Therefore, the total size of hidden data will be an accumulation of the total number of bits hidden in each high-frequency sub-bands coefficient.

For example, assume that 5-level DWT decomposition is applied to normal IoTs readings, each resultant sub-band coefficients number is 512 (i.e. value of n). Therefore, the 2D split matrix will be the size of 16×512 (i.e. 16 is the value of t). Also, assume about 5 bits (i.e. value of B) are hidden in each coefficient. Therefore, around 5120 bytes (5 KB) of confidential data can be hidden inside these coefficients.

2.5.4 Comparison of our MD1 vs MD2

Two novel models are introduced that rely on the Walsh-Hadamard and Wavelet signal processing techniques. Our first algorithm, MD1, relies on a much lighter and faster

transformation technique, such as Fast Walsh-Hadamard Transform in terms of time (i.e. linearithmic complexity $n\ log\ n$) and operations (i.e. based on additions and subtractions) [61]. However, the output is a one-dimensional vector of coefficients and so the generated random order is also a vector level which results in mild security. On the other hand, the second model, MD2, relies on wavelet transform and the production of multi-dimensional sub-bands wavelets tree. This is a relatively expensive process requiring both time, based on quadratic complexity, and operations, based on multiplications[72]. Nevertheless, due to the obtained two-dimensional wavelet tree, the random order also generated in 2D dimensional space.

In terms of the possible of hidden data size, MD1 can hide more data by using more than 80% of the obtained coefficients, whereas MD2 can embed less data by using only low sub-bands that hold 50% of the coefficients. Conversely, both models demonstrated $< 0.5\%$ resultant distortion in all cases, as discussed in the Experiments and Results section. In brief, this means the Walsh-Hadamard based model (MD1) has a faster and higher hiding capacity with mild security, whereas the wavelet-based model (MD2) is slower with a lower hiding capacity, but maintains stronger security.

2.5.5 Steganography Efficiency Measurement

To accurately evaluate the effect of our algorithms on the difference between the original and the stego IoTs device's readings (i.e. resultant distortion), a well-known measurement called a Percent of Root-mean-square Difference (PRD) is calculated, after applying the signal processing transforms. The PRD can accurately measure any reconstruction error between the original and the reconstructed form as defined in Eq 2.22 [1].

$$PRD_j = \sqrt{\frac{\sum_{i=1}^{m}(c_i - \widetilde{c_i})}{\sum_{i=1}^{m}(c_i^2)}} \times 100 \tag{2.22}$$

where c_i is an original coefficient, and $\widetilde{c_i}$ is the reconstructed stego form.

Identically, the distortion caused by the retrieving process is measured by calculating PRD between the original and the extracted IoTs device's readings after

removing all hidden, sensitive bits. The results are presented in the next section of this paper.

2.6 Experiments and Results

Our experiments are classified into two main parts: 1) Embedding which is used by distributed IoTs devices to hide secret information in their normal readings and the steps explained in Sections 2.3 and 2.4; 2) Extraction occurring on the recipient side (e.g. operation centres) so even if the IoTs devices' normal readings contain the hidden sensitive information are intercepted or tampered with by unauthorised parties, it will not disclose any information and it can be easily checked and verified.

2.6.1 MD1: Walsh-Hadamard based Steganography

In our experiments, various IoTs devices' readings were randomly selected from two different datasets. The first dataset is collected and published by Intel Berkeley Research Lab 2004 [73], and the second is collected and published by research group from University of North Carolina at Greensboro 2010 [74]. For brevity in this chapter, most of the results that will be shown are data, such as temperature, humidity, light and voltage, are collected by environmental monitoring IoTs sensors. Experiments were performed for each result to hide and retrieve the sensitive information according to our algorithm described in Sections 2.3.3 and 2.3.5. The confidential data was a set of information that must be kept confidential such as IDs, geometric location data and other private information. These are converted into bits to be hidden inside IoTs device's readings.

To obtain unbiased results, we experimented with our proposed algorithm with different keys as well as various IoTs streams lengths, such as 512, 1024, 2048 and 4096 samples. Also, to get the highest distortion effect, all high sequence coefficients have been used. For brevity, we present six cases of our results. 1) Fig. 2.13 shows an example of four original IoTs streams plots (temperature, humidity, light and voltage) used to hide sensitive information, and the stego form before and after the stego

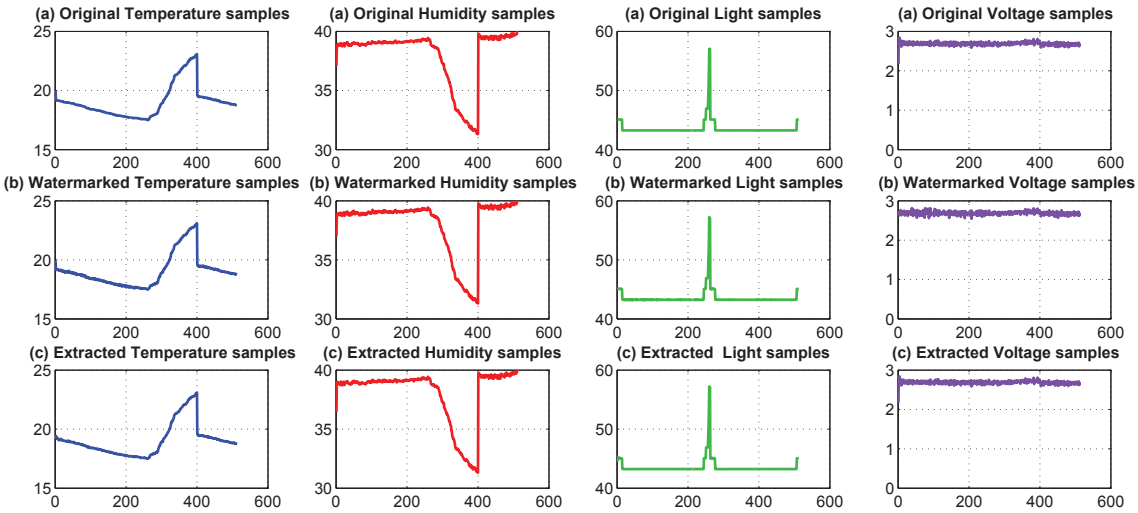

Figure 2.13: Four examples of IoTs sensors' readings: (a) direct plot for original form (b) stego form that contains the hidden sensitive information (i.e. IDs and geometric location data) and (c)extracted form (i.e. after removing the sensitive information).

Table 2.3: PRD results for temperature and humidity IoTs sensors readings from Dataset 1

	Temperature		Humidity	
Segment No	PRD % stego	PRD % Extracted	PRD % stego	PRD % Extracted
1	0.1482	0.1705	0.0820	0.1016
2	0.1413	0.1702	0.0912	0.1048
3	0.1781	0.2128	0.0764	0.0940
4	0.1791	0.2055	0.0717	0.0892
5	0.1205	0.1440	0.0795	0.0943
6	0.1792	0.2115	0.0700	0.0842
7	0.1575	0.1930	0.0678	0.0800
8	0.1339	0.1633	0.0772	0.0933
9	0.1251	0.1535	0.0845	0.1037
10	0.1760	0.1979	0.0769	0.0920
11	0.1683	0.1969	0.0756	0.0892
12	0.1226	0.1495	0.0977	0.1203

extraction process. 2) Table 2.3 shows the exact PRD results from the temperature and humidity IoTs streams between the original and stego form as well as between the original and the extracted forms. 3) Table 2.4 shows the PRD results from light and voltage IoTs readings. 4) Fig. 2.14 proves that whenever a fixed amount of sensitive

data is hidden in a bigger host data size, the amount of distortion is slightly decreased. 5) Fig.2.15 shows that, despite using all host data samples in the hiding process with different sizes (e.g. 512, 1024 and 2048), our algorithm behaviour and the resultant distortion is still stable whenever the same number of host data samples is used. 6) Fig. 2.16 accurately measures the time and the space taken by the proposed algorithm to accomplish both phases of the hiding and the extraction which clearly is very low - < 0.03 seconds - in all cases.

Table 2.4: PRD results for light and voltage IoTs sensors readings

	Light		Voltage	
Segment No	PRD % stego	PRD % Extracted	PRD % stego	PRD % Extracted
1	0.0354	0.0423	0.1712	0.5733
2	0.0129	0.0150	0.5072	0.2203
3	0.0672	0.0670	0.3266	0.2187
4	0.0658	0.0789	0.7298	0.2287
5	0.0617	0.0767	0.0721	0.3220
6	0.0672	0.0715	0.5449	0.2485
7	0.0677	0.0793	0.6236	0.2388
8	0.0410	0.0484	0.6989	0.1923
9	0.0296	0.0357	0.6114	0.2093
10	0.0841	0.1014	0.2169	0.2277
11	0.0691	0.0886	0.3319	0.2509
12	0.0672	0.0670	0.2204	0.2104

In all cases, despite the different sizes of IoTs sensors' readings and different ranges of values, all PRDs are ≤ 0.6 %. This means that the effect will only be to the third or fourth decimal values which are ignored in the temperature and humidity cases. This proves that our proposed algorithm will be stable and provide little distortion on the original IoTs sensors' readings. On the other hand, it offers a great advantage by securing the sensitive information in such a way that: 1) does not increase the actual IoTs streams size, 2) preserves the bandwidth, storage space and power consumption at the distributed IoTs end-point environment, and 3) only an authorised receiver can retrieve the hidden secured information: others (e.g. CPs) can only see the watermarked form which is almost similar to the original IoTs readings.

Table 2.5: PRD results for temperature and humidity IoTs sensors readings from Dataset 2

	Temperature		Humidity	
Segment No	PRD % Watermarked	PRD % Extracted	PRD % Watermarked	PRD % Extracted
1	0.0174	0.0761	0.0327	0.1335
2	0.0177	0.0776	0.0289	0.1386
3	0.0168	0.0766	0.0280	0.1458
4	0.0172	0.0773	0.0326	0.1338
5	0.0145	0.0695	0.0315	0.1333
6	0.0197	0.0765	0.0300	0.1350
7	0.0208	0.0778	0.0265	0.1391
8	0.0148	0.0807	0.0354	0.1447
9	0.0135	0.0789	0.0278	0.1449
10	0.0173	0.0753	0.0277	0.1329
11	0.0175	0.0761	0.0332	0.1475
12	0.0188	0.0784	0.0299	0.1419

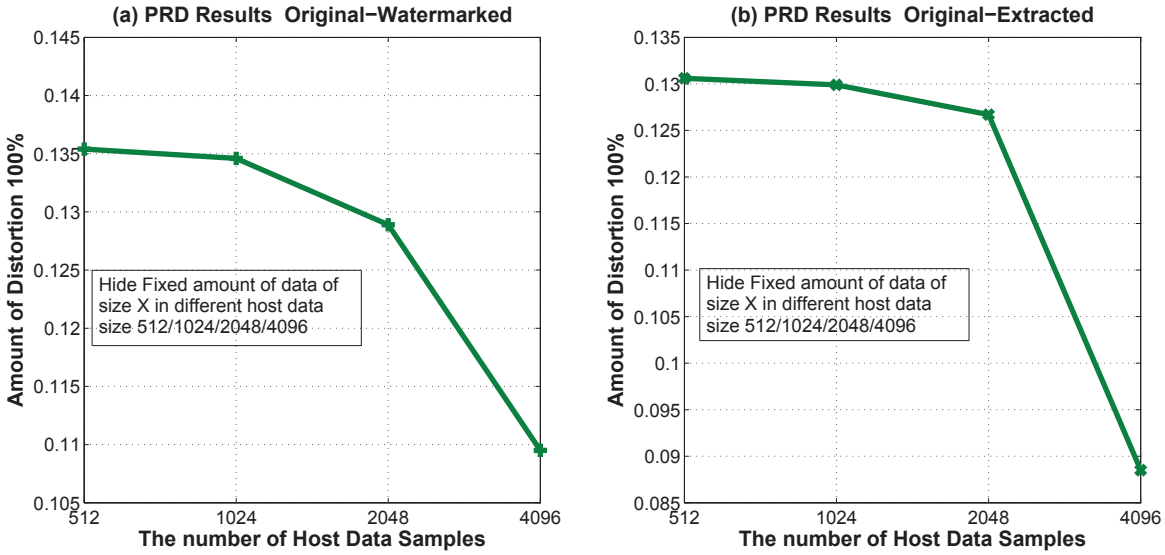

Figure 2.14: Fixed amount of sensitive data of size X is hidden in different host data size 512/1024/2048/4096. Distortion decreased with more host data size. (a) PRDs between the original and stego form, (b) PRDs between the original and the extracted form.

Comparison with Existing Models To the best of our knowledge, there is no research that used steganography with Walsh-Hadamard Transform signal processing to protect the privacy and the authenticity of the collected and transmitted data in the IoTs sensor streams context. However, recent work proposed by Huang and Fang

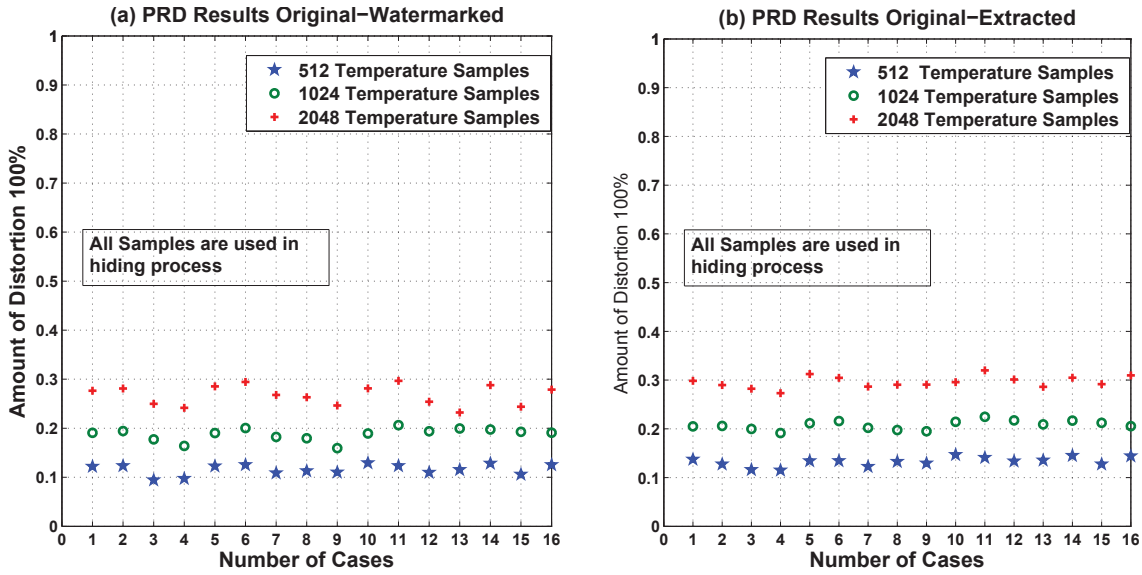

Figure 2.15: Despite using all host data samples, there is stability in the resultant distortion whenever we used the same number of samples (e.g. 16 cases use 512, 16 cases use 1024 and 16 cases use 2048) temperature samples. (a) PRDs between the original and stego form, (b) PRDs results between the original and the extracted form.

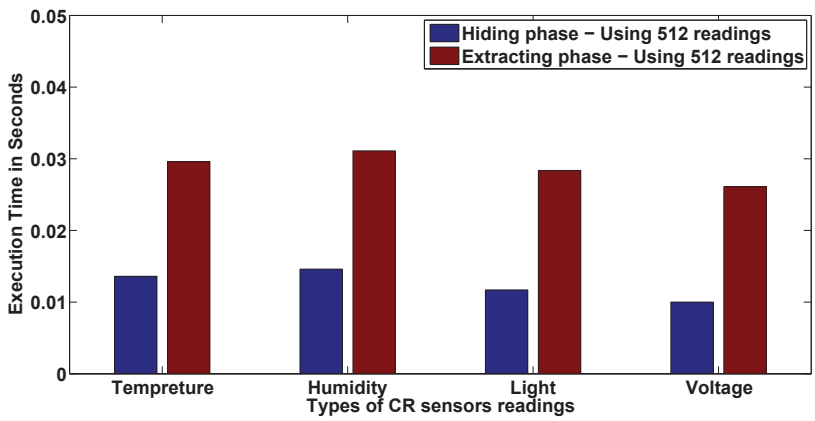

Figure 2.16: The required time and space to hide and extract sensitive information in the collected IoTs sensors readings.

[51] has a similar approach with different context. Therefore, our work is compared with this recent technique in [51] where the authors used a histogram and quad-tree decomposition to hide user identification in the transmitted image.

There are three main improvements in our technique: 1) The capacity of the hidden information is much higher in our algorithm than in model [51] where up

Table 2.6: Summary of improvements.

Feature	Model in [51]	Our algorithm
Size of hidden information per coefficient	1 bit	5 bits
Extra Overheads	$>= x\%$	0%
Hiding Mechanism	Static	Dynamic-Random
Security Key	-	$\surd$

to 5 bits can be hidden in each coefficient because of using FWHT, whereas only 1 bit can be embedded in their algorithm. 2) There is no overhead in our algorithm which means the size of the watermarked coefficients is equal to the original size, whereas the algorithm in [51] has a certain size of overheads because of using quad-tree decomposition. 3) Most importantly, our algorithm is more secure than the model in [51] because they are hiding the sensitive bits directly into a static fixed positions without a security key. In our algorithm three layers of enforced security scramble the resultant coefficients, encrypt the sensitive data and generate a random order to hide the bits dynamically in different and random coefficients. Table 2.6 highlights most of the improvements over existing techniques.

2.6.2 MD2: Wavelet-based Steganography

In our experiments, a wide range of IoTs devices' readings were randomly selected from datasets that are collected and published as part of a project named 'Smart*' by the Laboratory for Advanced System Software [75, 76]. The datasets contain minute-by-minute periodic readings for three months from three homes. The readings are related to power (e.g. watts consumption and heat index) and the environment (e.g. inside/outside temperature and humidity, and wind chill). They also contain electricity power consumption from 400 anonymous homes every minute for $(24 \times 30 \times 3)$ hours. As per the spatial and temporal aggregations definition in [77], these readings are regarded as temporal, because they are collected separately from every single house by equipping it with a smart meter that releases its readings on a regular basis. In this study, all types of readings mentioned were used to thoroughly consider the feasibility of implementing the proposed algorithm to various IoTs streams. Experiments were

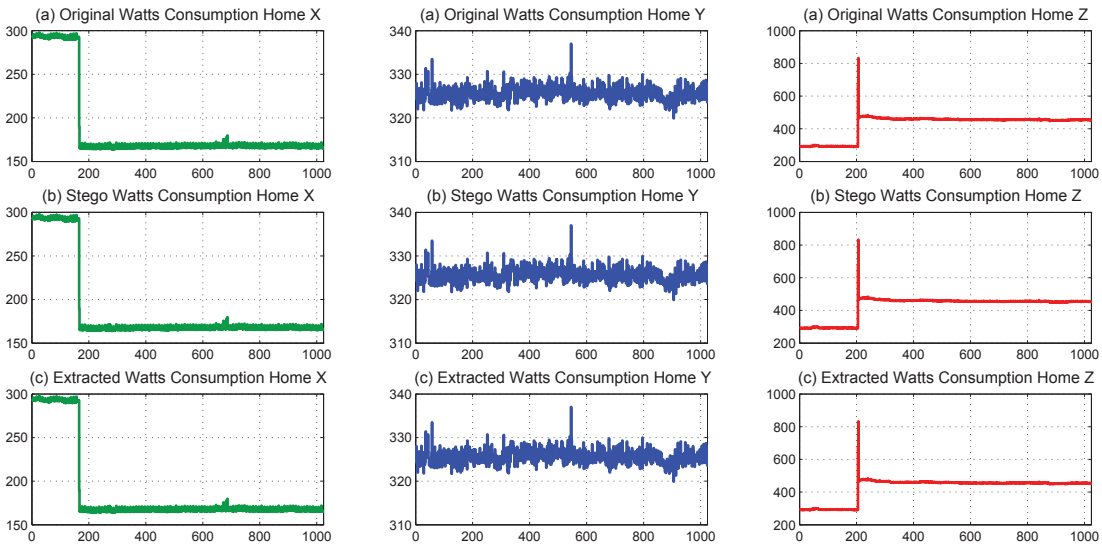

Figure 2.17: Three examples of watts consumption readings collected from different homes: (a) direct plot for original form (b) stego form that contains the hidden sensitive information (i.e. meter ID, household name, DoB, address and total watts) and (c) extracted form (i.e. after removing the confidential information).

performed to hide and extract the confidential information according to our proposed algorithm described in Sections 2.4.3 and 2.4.5. The confidential data was a set of information that is secret, such as smart meter IDs, geometric location data, household names, DoBs, addresses and the total watts consumption all of which were converted into bits to be hidden inside IoTs device readings.

To avoid biased results, we experimented our proposed algorithm with different keys as well as various IoTs sensor streams with lengths of 512, 800, 1024, 2048 and 4096 as samples. Also, to get the worst distortion effect, all detailed sub-band coefficients have been used. For brevity, we present three cases of our results in this section: 1) Fig. 2.17 shows an example of a plot with three original smart meter readings showing watts consumption used to hide sensitive information, and the stego form before and after the stego extraction process; 2) Fig. 2.18 shows an example of another plot of six original smart meters' readings of inside/outside temperature and humidity, wind chill and heat indices, and the stego form before and after the retrieval process; 3) Tables 2.7, 2.8 and 2.9 show the exact PRD results from the watts, heat

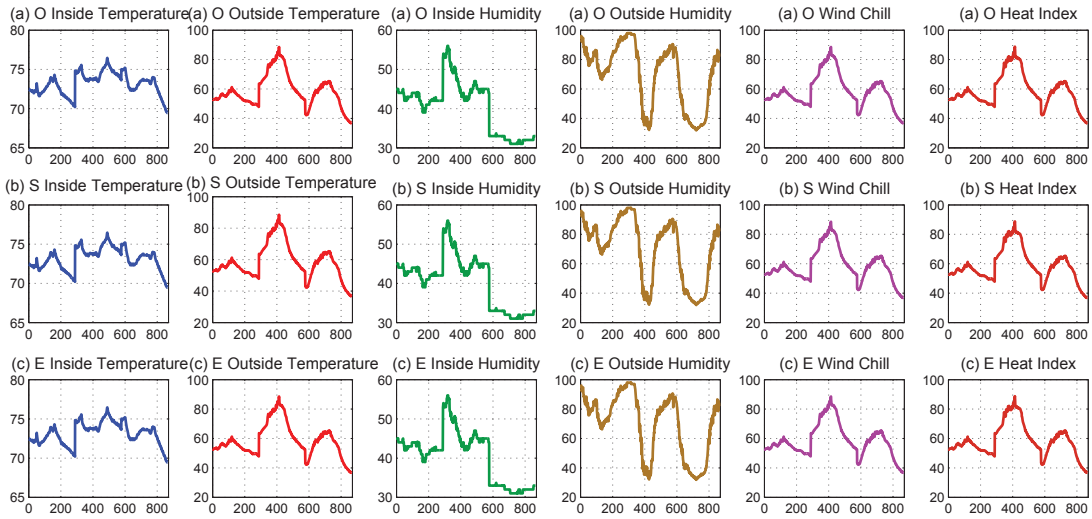

Figure 2.18: Six additional examples of possible smart meter readings: (a) direct plot for original form (b) stego form that contains the hidden confidential information and (c) extracted form (i.e. after removing the confidential information).

index, temperature and humidity readings between the original and stego form, as well as between the original and the extracted forms.

From all cases, despite the different sample lengths of IoTs streams and various ranges of values, all PRDs are less than ≤ 0.6 %. This means that the impact will only be to the third or fourth decimal digits which are usually ignored in many cases; for example, watts, temperature and humidity. To avoid inaccurate calculation of financial bills, the total watts consumption is also hidden. Consequently, this proves that our proposed algorithm will have stable and little distortion on the original IoTs streams. On the other hand, it offers a great solution with a paradigm shift for securing the confidential information that should be transmitted, as well as the authenticity of smart meter readings. The advantages of this solution are: 1) strong end-to-end confidentiality and authenticity where the hidden secured information (including the total watts consumption) can only be retrieved and verified by an authorised receiver, whereas others can only see the stego form which is almost similar to the original IoTs readings; 2) no increase in the actual size of smart meter readings, and 3) no change to the original form of readings which helps the authorised recipients quickly use

Table 2.7: PRD results for Watts and Heat Index readings

	Watts		Heat-Index	
No	PRD % Stego	PRD % Extracted	PRD % Stego	PRD % Extracted
1	0.0006	0.0007	0.0016	0.0021
2	0.0005	0.0006	0.0016	0.0020
3	0.0004	0.0005	0.0015	0.0020
4	0.0006	0.0007	0.0016	0.0019
5	0.0006	0.0008	0.0015	0.0019
6	0.0003	0.0004	0.0016	0.0019
7	0.0004	0.0005	0.0018	0.0021
8	0.8797	0.5797	0.0016	0.0020
9	0.4689	0.4689	0.0018	0.0023
10	0.0003	0.0003	0.0017	0.0020
11	0.0003	0.0004	0.0016	0.0020
12	0.0007	0.0008	0.0017	0.0021
13	0.0004	0.0005	0.3638	0.3638
14	0.0004	0.0006	0.1700	0.1700
15	0.0006	0.0007	0.0019	0.0023

operational administrations, such as cloud providers, without disclosing any sensitive information.

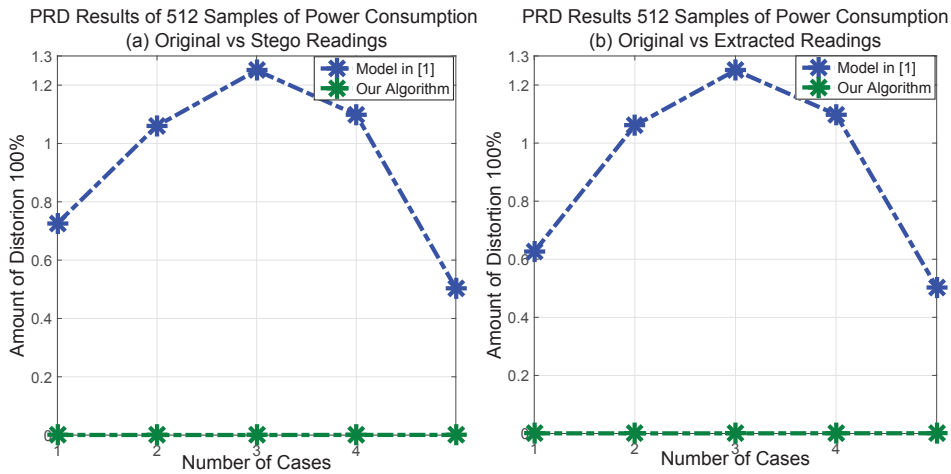

Figure 2.19: Distortion Comparison for different PRD results of 512 samples of power consumptions between our algorithm and the model in [1]. (a) PRDs between the original and Stego form, (b) PRDs results between the original and the extracted form.

Table 2.8: PRD results for Inside and Outside temperature readings

	Inside Temperature		Outside Temperature	
No	PRD % Stego	PRD % Extracted	PRD % Stego	PRD % Extracted
1	0.0015	0.0019	0.0016	0.0021
2	0.0014	0.0019	0.0016	0.0021
3	0.0015	0.0018	0.0016	0.0021
4	0.0015	0.0019	0.0016	0.0020
5	0.0015	0.0020	0.0016	0.0020
6	0.0015	0.0018	0.0016	0.0020
7	0.0014	0.0018	0.0017	0.0021
8	0.0014	0.0018	0.0016	0.0020
9	0.0015	0.0019	0.0018	0.0023
10	0.0015	0.0018	0.0016	0.0021
11	0.0014	0.0018	0.0015	0.0019
12	0.0015	0.0019	0.0016	0.0021
13	0.0015	0.0018	0.3341	0.3341
14	0.0015	0.0019	0.1599	0.1599
15	0.0015	0.0019	0.0018	0.0024

Table 2.9: PRD results for Inside and Outside humidity readings

	Inside Humidity		Outside Humidity	
No	PRD % Stego	PRD % Extracted	PRD % Stego	PRD % Extracted
1	0.0023	0.0029	0.4201	0.4215
2	0.0022	0.0027	0.4580	0.4589
3	0.0022	0.0029	0.1471	0.0613
4	0.0023	0.0029	0.0612	0.0020
5	0.0023	0.0030	0.0087	0.0088
6	0.0026	0.0032	0.3224	0.3224
7	0.0024	0.0031	0.0014	0.0018
8	0.0025	0.0031	0.5102	0.6102
9	0.0026	0.0033	0.0015	0.0019
10	0.0024	0.0031	0.2643	0.2643
11	0.0024	0.0030	0.1432	0.1433
12	0.0024	0.0030	0.1011	0.1011
13	0.0020	0.0025	0.4278	0.5278
14	0.0029	0.0034	0.3898	0.3898
15	0.0027	0.0034	0.0015	0.0019

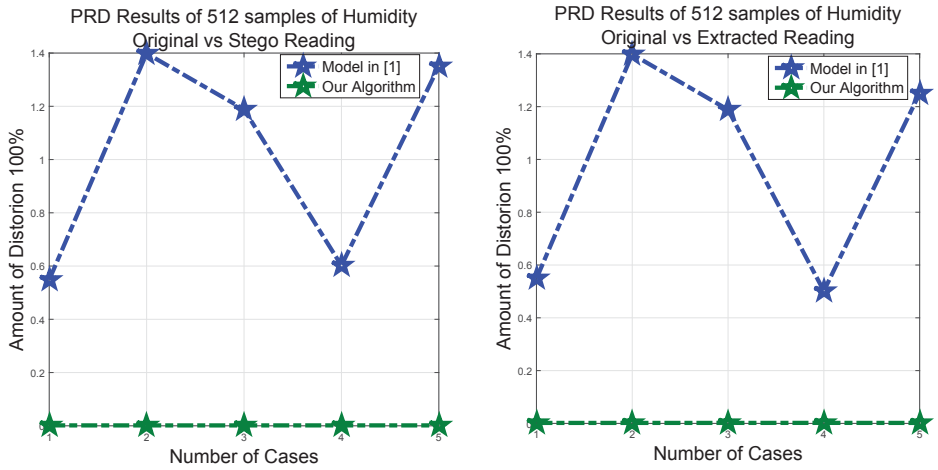

Figure 2.20: Distortion comparison for different PRD results of 512 samples of Humidity readings between our algorithm and the model in [1]. (a) PRDs between the original and Stego form, (b) PRDs results between the original and the extracted form.

Comparison with Existing Models To the best of our knowledge, there is no research work that uses steganography with wavelet transformation signal processing to hide IoTs confidential information in the resultant wavelet tree. However, recently there is a proposed study [1] uses a similar approach in a different context. Therefore, our proposed technique is compared with this recent model in [1] where the authors proposed a steganographic technique to hide patients' confidential information in their collected electrocardiogram signals using signal processing. There are two main improvements in our algorithm: 1) After experimenting both models on different sets of IoTs streams (see Fig 2.19 and Fig 2.20), it should be noted that our algorithm has much less distortion and is more stable than the model in [1]. This is because only high-frequency sub-bands are used in the hiding process, whereas all sub-bands coefficients are used in their model. 2) Both algorithms use a security key to encrypt the confidential information; however, our algorithm is more efficient in terms of generating and managing the random hiding order matrix. This is because in our model, this matrix is dynamically generated on the fly using the security key, whereas in the model [1] the random order is statically stored in a form of 2D matrix of size 128×32 which obviously consumes the storage space in addition to the security risk

of storing and managing this matrix.

2.7 Chapter Summary

In this chapter, two novel steganography algorithms that rely on different mathematical security models (the vector space and the two-dimensional tree) have been designed and implemented by using the key to encrypt the sensitive information, reshape the coefficients into a random hierarchy, and randomly generate an order used in the hiding process. To broaden the hiding capacity, insertion of sensitive information and to maximise the randomisation process, the Fast Walsh-Hadamard and DWT processing techniques were exploited - we transformed the IoTs high streams from their spatial domain to their frequency domain in order to recognise and collect most of the readings' features in some coefficients, thus allowing others to be used freely to hide more data. The two new proposed models in this chapter vary in their simplicity by using the MD1: Walsh Hadamard-based steganography versus security with MD2: Wavelet-based steganography. The key contribution of both techniques is that they protect the privacy and the authenticity by neither increasing nor changing the form of the transmitted IoTs readings. This means only data owners can retrieve the seal, whereas others are just monitoring the protected form of the readings which is almost similar to the original readings. On the other hand, the protected streams can be directly used without revealing privacy nor the authenticity at the third-party cloud servers.

Chapter 3

Protected IoTs Manipulation Detection and Recovery

This chapter answers the second research question discussed in Section 1.2. The main concerns about the possibility of intentional or unintentional noise are examined along with the issues of losing the hidden secret information and damaging the transmitted readings. This chapter highlights how a combination of steganography, error detection and correction is a possible technique to overcome some of the challenges. Section 3.1.1 introduces the main contributions of this chapter and how they were accomplished. Section 3.2 briefly highlights the key related works and the efforts of other researchers to solve the issue. Section 3.3 explains in detail our novel model including the design, secret information encoding, readings decomposition and manipulation, readings re-composition, manipulation detection and recovery. The evaluation of various characteristics of our model including complexity analysis and correction capabilities are introduced in Section 3.4. Experimental examinations of the model are presented in Section 3.5 detailing the test bed scenario and the 'wide range of noise mimicking the real world' scenario, recovery of lost hidden data, a possible remedy of the actual IoTs high streams and a comparison of existing techniques. Section 3.6 summarises this chapter.

3.1 Introduction

The classic power grid of the past 100 years is now regarded as unsuitable to 21st Century requirements for reasons such as outage management deficiencies, a lack of automated and real-time analysis [78, 79]. Consequently, a new Internet of Things (IoTs) infrastructure called the 'smart grid' has presently emerged and can be used to automatically gather periodic smart meter readings at every second or minute for power consumption and environmental characteristics of the premise, and transmit them to operational centres using various techniques [80, 81]. Significant benefits include improved efficiency for automated outage management, accuracy for continuous-dynamic electricity distribution and sustainability regarding climate change mitigation. However, the unusual amount of the continuous transmitted data from millions of premises and the enormous demand on the spectrum reservation results in wireless communications issues, such as 'spectrum scarcity' [82].

To solve these issues, a new wireless communication technology emerged called Cognitive Radio (CR) [83, 84]. The basic idea is that the licensed spectrum for various parties (e.g. premises) can be shared by Secondary User (SU) whenever the Primary User (PU) is idle (i.e. white space). The main purposes are: (1) improving the communication performance and throughput, and (2) reducing the interference between the applications that use the identical or overlapping bands, such as Bluetooth and ZigBee at 2.4 GHz [85–87]. Therefore, tremendous efforts are made to exploit this opportunity in the IoTs context [23, 88–91]. Despite the obvious advantages, CR smart grids cause many security and robustness issues due to sharing the transmission spectrum [92, 93].

Recently, a widely-known technique in multimedia domain called steganography was exploited to ensure privacy without changing the form of the transmitted readings, such as in models [2–4] in Chapter 2. Steganography can protect the sensitive information where a piece of a secret message is hidden inside the host data and can only be retrieved by authorised users. However, due to shared spectrum characteristics in CR mechanism, where PUs and SUs are sharing the same band simultaneously,

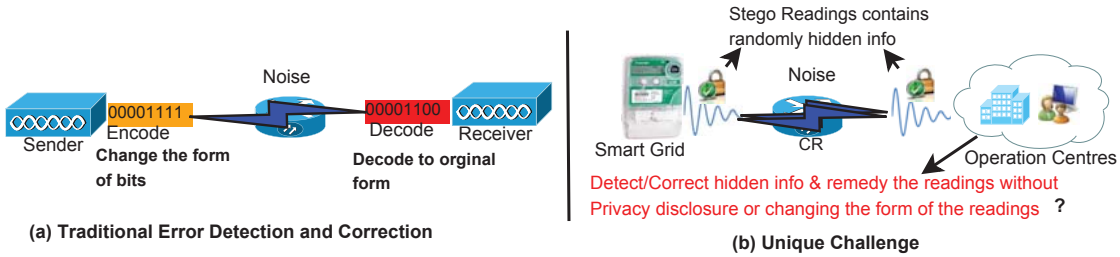

Figure 3.1: The main unique challenges that highlight the contribution of this chapter.

the transmitted data are highly prone to an intentional (i.e. interference) and unintentional (i.e. noise) attacks rendering the existing solutions impractical [23]. This is simply because any slight change in the transmitted stego form of readings will result in a loss of hidden information, such as household sensitive data, and more significantly, a loss of faith in the received readings. There are three possible outcomes: it may be too late to ask the source to resend (especially in critical cases), the source is often configured to forget what it sends directly due to resource constraints, and the destination has no return channel to the source. In an extreme case, this may happen to a million premises on the same day!

To overcome the deficiencies of the aforementioned models, we are compelled to address the following questions (see Fig. 3.1).

1. How can any change in the hidden secret information be detected and recovered?
2. Can the recovered secret information be used to remedy the received collected CR IoTs readings?
3. Can above two be met without revealing the sensitive information to cloud providers or without changing the form of the transmitted readings?

3.1.1 Contributions

- To the best of our knowledge, it is the first novel hybrid model that combines advanced steganographic algorithms with error detection and correction techniques (Bose-Chaudhuri-Hocquenghem (BCH) syndrome codes) in the

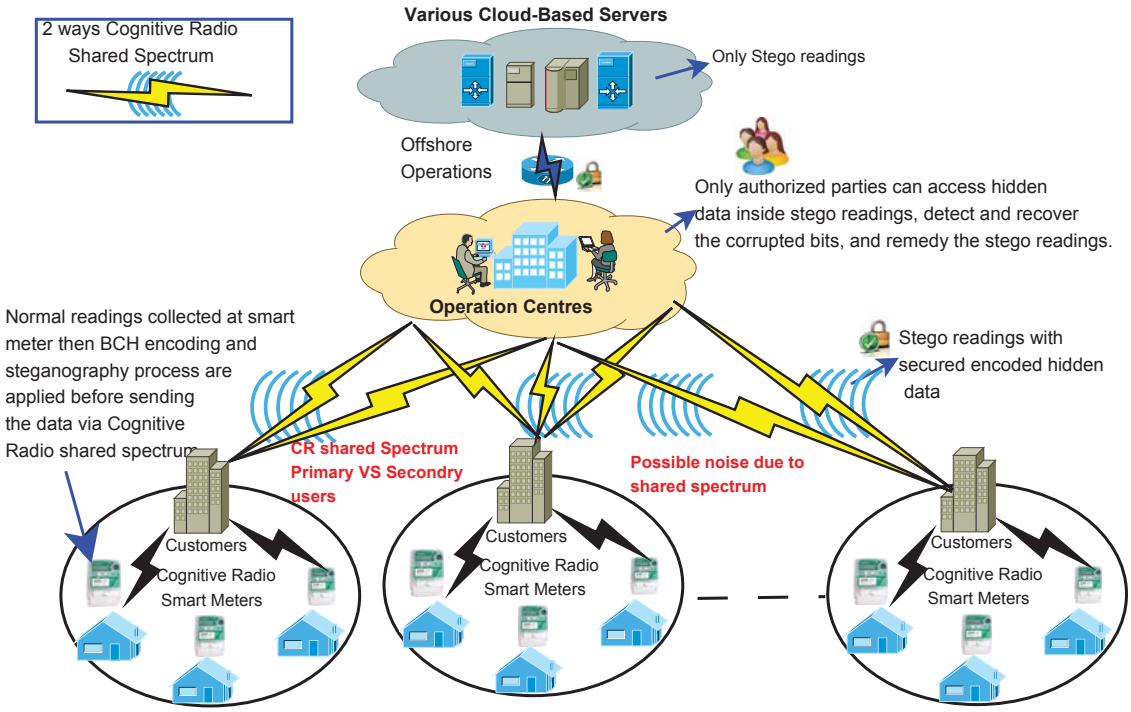

Figure 3.2: The main scenario of our proposed model where customers confidential information is encoded and hidden inside IoTs normal readings and only authorized users can retrieve, detect, recover this confidential data and remedy the received readings.

context of IoTs streams. This will allow us to detect and recover any loss from CR shared spectrum noise drawn from hidden confidential information without privacy disclosure, and it will also allow us to remedy the received normal readings by using the corrected version of the secret hidden data. Both cases have been examined carefully and are explained in Section 3.5.

- To the best of our knowledge, it is the first model that strengthens the security of hiding and increases the randomisation into a 3D level using a fast signal processing technique called 3D Discrete Wavelet Transform (DWT).

- The integration of BCH with advanced steganographic algorithms highlighted a new finding this paper will explore in Section 3.5 by simply integrating error detection and correction techniques with most of the previously proposed steganography algorithms. These algorithms use the widely-known hiding

positions - Least Significant Bits (LSB) - will fail to recover the corrupted hidden bits. Therefore, the hiding positions and coefficients are chosen carefully to achieve the best Bit Error Rate (BER), that is the recovery accuracy of the hidden secret information, and the Percent of Root-mean-square Difference (PRD)/Root Mean Square (RMS) - meaning the remedy precision of the normal readings. This has been examined thoroughly and is presented in Section 3.5 after monitoring the PRD, RMS and BER results with various ranges of hiding positions.

In our model (see Fig 3.2), IoTs smart meters are used to collect normal readings from different customer premises for wattage consumption, heating index, inside and outside temperature, and humidity. Customer secure information (e.g. grid ID, geometric location, name, Date of Birth (DoB), address and total power consumption) will be then encoded using BCH syndrome codes and randomly hidden inside the normal readings. Finally, the stego normal readings are transmitted to the remote operational centres via CR shared spectrum. Consequently, the real-transmitted data size is only of the normal readings with no additional overhead, because the encoded confidential information is embedded inside them. The stego readings that contain the hidden information will be stored at operational centres. However, only authorised users can retrieve the secretly encoded information from the stego normal readings by using an appropriate key, to detect and recover any alteration to them due to possible CR interference, as well as to remedy the stego form of readings. Alternatively, offshore cloud-based servers (and others) can only see the stego form. The second advantage is that based on our experimental results, even the stego CR smart meter readings can be cured so there is no need to resend the stego whenever the data is corrupted.

The rest of this chapter is organised as follows. Section 3.2 summarises the relevant work. Section 3.3 presents our algorithm and its preliminaries. The evaluation of different characteristics of our proposed technique is introduced in Section 3.4. Section 3.5 discusses the experiments we performed and the results we obtained. We draw our conclusions in Section 3.6.

3.2 Related Work

Any solution proposed to protect the transmitted CR IoTs information should carefully consider the nature of CR smart meters shared spectrum and their surrounding environment with: (1) security (the privacy of confidential information and the authenticity of transmitted readings), (2) robustness (detection and recovery of loss), and (3) efficiency (direct usage of the received readings without privacy disclosure). However, most current proposed solutions lack a suitable balance between these three features.

To ensure robustness, most of the existing solutions use traditional error detection and correction techniques (e.g. BCH or LDPC) in the low level data-link layer where the packets are independently encoded at the sender location and decoded at the receiver end [94–97]. The outcomes of these processes will be in the form of a 'codeword' that is completely different to the original packet in both size and shape. Therefore, despite their technical functionalities, their main concern is to ensure the robustness of the transmitted packets and to totally ignore the other two aspects of security and efficiency.

On the other hand, a majority of the solutions that focused on the security (the first aspect) neglect robustness and efficiency. For example, models shown in [98, 99] provide strong security by using traditional cryptography techniques, such as asymmetric encryption. However, their efficiency is poor, because all the data should be decrypted whenever it is used.

An other category of solutions targeted this issue by using a new cryptography technique, homomorphism, as models in [100–102]. Although they attempted to achieve a reasonable balance between security and efficiency by using homomorphic encryption, this non-traditional cryptography is still not feasible in practical applications for its complexity [2].

The third group of solutions applied steganography - a well-known technique used in multimedia to ensure privacy and authenticity without neglecting the efficiency aspect. For example, models [2–4] show applied steganography used to hide a tenant's

confidential information in the collected premises readings using signal processing in [2], whereas the model in [3] was to protect the privacy of the remote CR sensors' nodes using steganography. On the other hand, the model in [4] proposes a least significant qubit (LSQb) information hiding algorithm for quantum images. However, the existing steganography models have limitations - changing a single bit in the transmitted readings means that results with hidden information cannot be retrieved and the received readings cannot be trusted.

With all these streams of solutions, none have carefully considered the three aspects together. This is mainly because they are transplanting widely-known techniques to a completely new territory, IoTs, without considering characteristics such as using shared spectrum for robustness issues, applying direct operations at operation centres for efficiency issues and performing offshore operations using cloud infrastructure for privacy issues.

To the best of our knowledge, this chapter is the first to research work that tackles this unique challenge (see Fig. 3.1).

3.3 Methodology

This paper is introducing the first hybrid BCH-based steganography model that can recover lost hidden secret information and remedy the received corrupted readings. Due to CR shared spectrum interference in CR IoTs networks, we first recall the steganography theoretical process as preliminaries which is formulated based on our previously published works [2, 3].

3.3.1 Preliminaries

The main steps secure the sensitive information, randomise the normal collected readings, hide the sensitive bits inside the resultant coefficients and finally reconstruct the readings.

Securing the hiding At this stage of this paper, we target the hiding process of the sensitive information in the normal IoTs smart meter readings that must be entirely random and different among other CR smart meters. This prevents unlawful parties from retrieving them correctly. Therefore, a security key will be generated for every CR smart meter and will only be known to the end recipient of the data (i.e. operation centres). This key has two main tasks:

1. Encrypt the sensitive information that is related to CR smart meters (e.g. IDs, geometric location), and the household (e.g. name, DoB, address and the total watts consumption) before the hiding process using symmetric encryption (i.e. AES), which is very fast and suits the CR smart grid's technical capabilities (see Fig. 3.3). This can be defined in Eq 3.1.

$$f_E(O, K) \Rightarrow \widetilde{E} \tag{3.1}$$

   where f_E is an AES algorithm, O is the original smart meter IoTs and household sensitive information, K is the key, and $\widetilde{E}$ is the encrypted form.

2. Generate random sequences of coefficients to be used to hide the confidential smart meter IoTs information. This will be completely clear after reading Section 3.3.2.

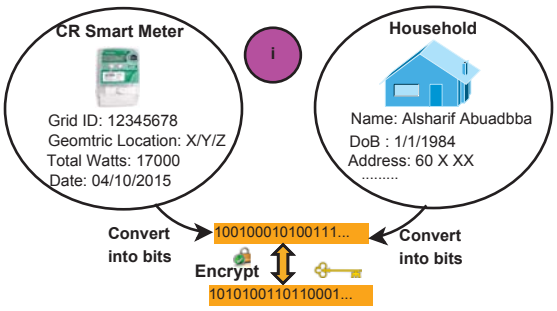

Figure 3.3: An example of how the sensitive information is encrypted before encoding.

Embedding After encrypting the sensitive information, the randomly generated order will be followed to hide the sensitive bits, such as power consumption, in the

normal collected readings. However, to increase the level of randomisations before the hiding process, different signal processing techniques have been applied (e.g. 1D and 2D wavelet decomposition). Finally, an inverse process of these signal processing techniques should be performed to reconstruct the stego readings as they contain the hidden bits.

Retrieving The legitimate recipient of the stego readings must have the security key. The process is almost identical to the hiding process except that it extracts the bits instead of hiding them.

3.3.2 The Proposed Scheme

This section sheds the light on the crucial steps in our algorithm such as encoding, 3D wavelet transformation, 3D random order generation, and finally decoding, as well as correction and recovery.

Secret Information Encoding In information theory, BCH syndrome codes are cyclic linear block error detection and correction codes that are constructed using finite fields or more precisely, the Galois field [103]. The BCH abbreviation stands for the discoverers, Bose, Chaudhuri, and independently Hocquenghem. BCH simply works by breaking the information into chunks of size k, calculating parity check bits of size p after the division process by a generated arithmetic polynomial g_x belongs to the same chosen Galois field. Each $k+p$ block is called a 'codeword' and its size is n. These resultant codewords will be packed together and used in the detection and recovery of the corrupted bits. Theoretically, every BCH (n,k) codes can correct up to t errors [95].

In this paper, BCH is chosen for two main reasons. (1) The encoding process, which will be at the IoTs smart meter, is very simple and fast (see Eq. 3.2) compared with other error detection and correction codes (e.g. LDPC) [96], and so it fits the remote IoTs devices technical capabilities. (2) It exhibits a stable behaviour in terms of its

correcting capabilities up to its theoretical entropy t, which explains its wide usage with satellite communications, disk storage and Automatic Teller Machines (ATMs).

$$\begin{cases} k \mod g_x & \Rightarrow p \\ p + k & \Rightarrow n \end{cases} \tag{3.2}$$

Therefore, various BCH syndrome codes are used to encode the encrypted sensitive information before hiding process by: (1) detecting and correcting the confidential information in their encrypted format to avoid privacy disclosure, and (2) unlike all the aforementioned applications, our algorithm uses BCH in such a way that does not increase or change the actual transmitted IoTs smart meter readings, meaning it has an efficiency advantage.

Readings Decomposition/Randomisation 3D Wavelet Transform (WT) is a widely-known linear technique used in signal and image processing that is performed on the given signal that decomposes into different values called coefficients. These represent the frequency components at a given time [64]. 3D WT is shown in Eq 3.3.

$$C(a, b, c) = \int_{-\infty}^{\infty} f(t)\psi(a, b, c)dt \tag{3.3}$$

where C represents resultant coefficients, a, b and c are positive integers that represent transform parameters, and ψ is wavelet function [65].

There are two ways of using wavelet which are discrete and continuous. However, the preferred format is discrete, because it simulates the reality where most of the produced and analysed real-life information comes in discrete numbers rather than in continuous functions [66].

There are two main targets when choosing 3D DWT for this paper: (1) Unpredictable randomisation, robust detection and recovery of the IoTs smart meter readings by decomposing the waves into 3D frequency domain sub-bands coefficients; all while avoiding to increase the size in their time domain when the stego form of readings are transmitted to the operation centres. (2) The readings can almost be reconstructed from only the low-frequency components, as the approximation

sub-band coefficients, allowing other detailed sub-band coefficients to be freely used to embed the encoded sensitive information with a minimum amount of distortion on the originally transmitted IoTs readings (e.g. watts consumptions). This inspired us to further use this technique to ensure high randomisation with enough space to embed more encoded secret information related to IoTs without affecting the actual smart meter readings.

To conduct the 3D DWT decomposition, two levels of filters (high-band and low-band) are applied to the original readings. Consequently, two sub-signals called sub-bands are obtained. The first relates to low-frequency components that represent the approximation of the original readings. The second sub-band relates to high-frequency components that represent the detailed coefficients. To avoid the complexity of multi-level 3D decomposition, a type called 'single 3D transform' was used [65] which is defined in Eq 3.4.

$$D(i,j,k) = \sum_i \sum_j \sum_k X(i)\Phi_{ijk}(n) \tag{3.4}$$

where $D(i,j,k)$ represents 3D DWT coefficients, i, j and k represent the shift and scale transform parameters and $\Phi_{ijk}(n)$ represents the base time wavelet function that is shown in Eq 3.5.

$$\Phi_{ijk}(n) = 2^{-i/2}\Phi(2^{-i}(n-j).(n-k)) \tag{3.5}$$

Therefore, in this chapter, different real-time collected IoTs smart meter readings of watts, heat index, wind chill, temperature and humidity will be decomposed into 3D wavelet sub-bands coefficients as shown in Fig. 3.4. The chosen wavelet family in the decomposition is called *Daubechies* with the order 2 ($db2$) because its performance in analysing discontinuous-disturbance-dynamic consecutive readings has already been proven to be perfect [68]. To guarantee the lowest amount of distortion, the most important sub-bands coefficients will not be utilized because they represent the most significant parts of the IoTs readings. On the other hand, a number of bits will be inserted in the rest of the detailed sub-bands coefficients, which is called the steganography level, meaning how many bits can be embedded into each sub-bands

coefficient. To ensure the minimum acceptable distortion on the actual readings, many experiments have been performed in [2] and [3] to select an appropriate steganography level. Consequently, about five bits can be hidden in the randomly-selected high frequency sub-bands coefficients.

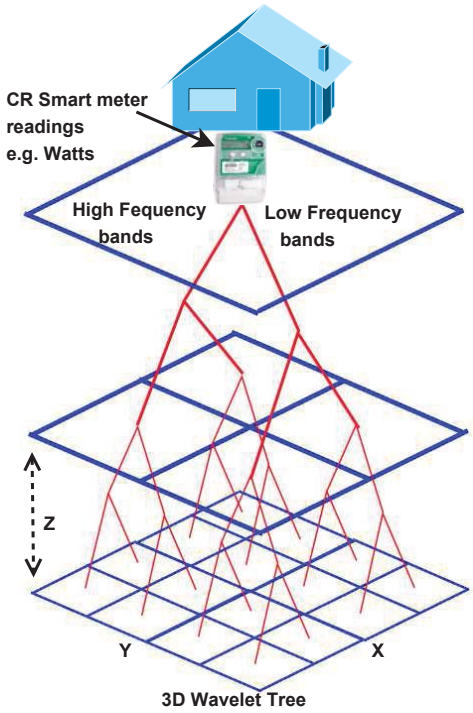

Figure 3.4: Decomposing CR smart meter readings in a 3D sub-bands tree.

Random Coefficients Order The key is utilised to generate three random coefficients' sequences in a 3D matrix format that will be used to hide the encoded secret information. This is defined in Eq 3.6.

$$f_x(K) \Rightarrow \widetilde{X} \times \widetilde{Y} \times \widetilde{Z} \tag{3.6}$$

where $\widetilde{X} \times \widetilde{Y} \times \widetilde{Z}$ is the generated $3D$ sequence of coefficients.

Fig. 3.5 shows an example of how the key can be utilised to generate the selected 3D DWT sub-bands coefficients' order. It initialises by converting the key into ASCII and a default position value will be assigned. Then, the key ASCII will be arranged

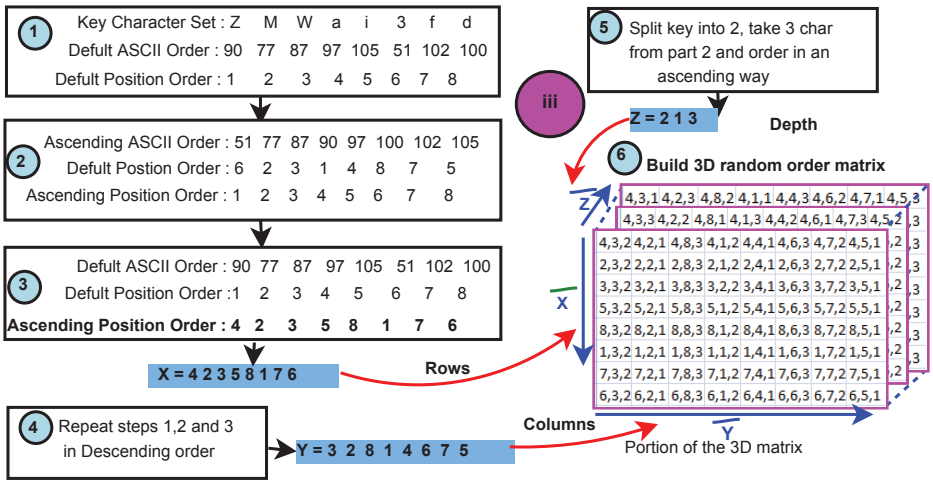

Figure 3.5: An example of how the $3D$ hiding matrix order $\widetilde{X} \times \widetilde{Y} \times \widetilde{Z}$ is generated from the key.

in an ascending manner and another position order will be given, again in ascending order. Next, return the key ASCII to its original format using the default position order (to avoid producing two similar sequence of coefficients from different keys) which represents $\widetilde{X}$. Almost the identical steps are repeated using a descending order to generate another sequence of values representing $\widetilde{Y}$. After that, $\widetilde{Z}$ is generated by splitting the key's characters into two parts, taking three characters from the second part and calculating the ascending order. Finally, $\widetilde{X}$, $\widetilde{Y}$ and $\widetilde{Z}$ are used to build the random sequence 3D matrix. However, in our algorithm, the key length will be ≥ 128 bits in length.

Coefficients Rescaling/Manipulation After applying 3D DWT to the normal CR smart meter readings, the resultant low frequency coefficients are split and rescaled to 3D $X \times Y \times Z$ matrix form (see Fig 3.6). The key is then used to encrypt household confidential information. The secret information will be encoded using various BCH syndrome codes. Next, the key will be utilised to generate the 3D random coefficients order. The encoded bits will finally be packed and embedded corresponding to the generated order. The detailed process of hiding is shown in Figs. 3.3, 3.5, 3.6 and

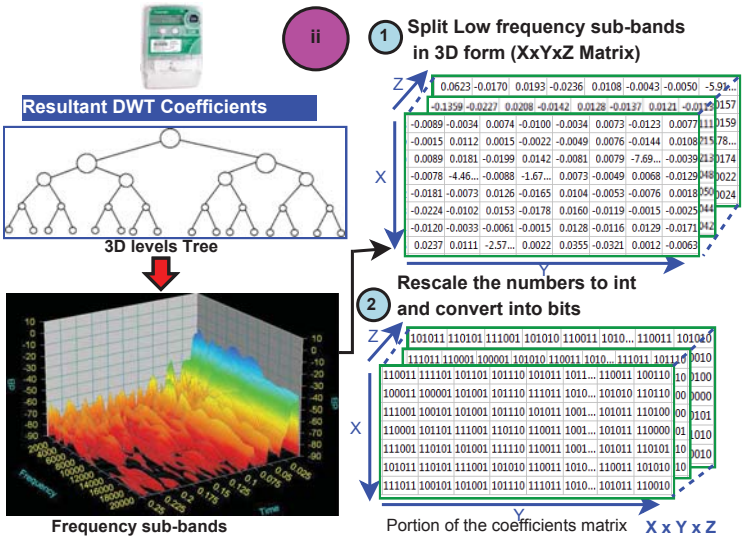

Figure 3.6: Block diagram presents how the 3D DWT sub-bands coefficients are split, reshaped, rescaled and converted into bits.

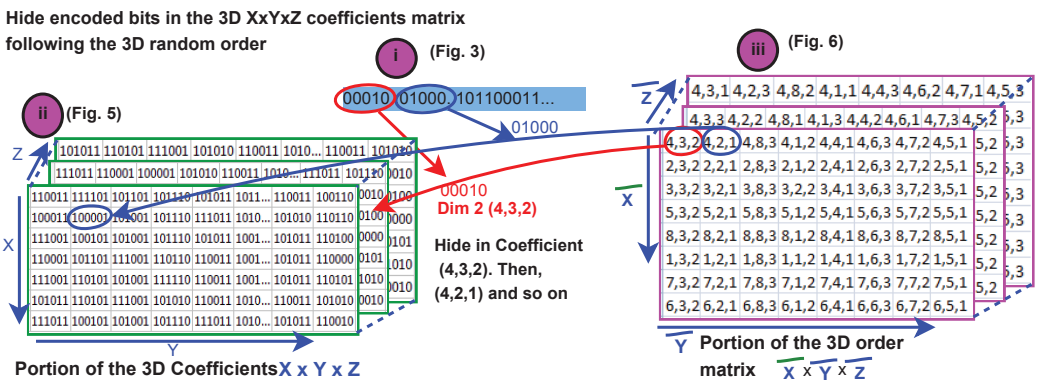

Figure 3.7: Block diagram summarises the hiding process and uses the information explained in Figures 3.3, 3.5 and 3.6.

summarised in Fig 3.7

Readings Re-composition The resultant sub-bands values after the embedding process are called 'stego coefficients'. At this stage, these coefficients will be re-combined into a 3D sub-bands coefficients matrix and the inverse 3D DWT will be applied to convert smart meter readings from their frequency domain to their original time domain. The result is a reconstructed form called stego IoTs readings (i.e. contains hidden encoded information) which is quite similar to the IoTs normal

readings. The significance of that is even the stego IoTs readings can be used as the original form. However, only authorised recipients with a security key can retrieve the hidden secret information (e.g. meter IDs, household name, DoB, address and total watts),and can detect and correct any possible manipulation. The inverse 3D DWT is defined by Eq 3.7.

$$X = \sum_i \sum_j \sum_k D(i,j,k)\Phi_{ijk}(n) \tag{3.7}$$

where X is the original form of IoTs streams.

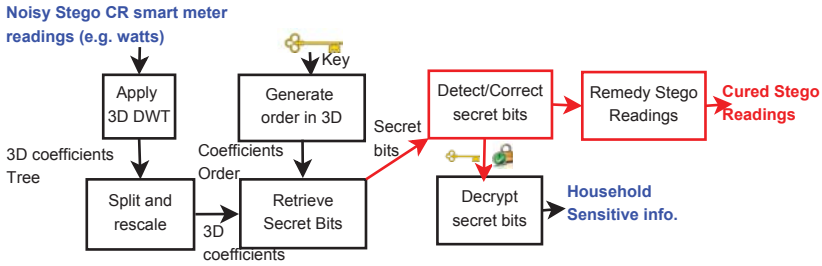

Figure 3.8: An overview of the extraction, correction and remedy processes.

3.3.3 Error Detection and Recovery

To guarantee the robustness of the received IoTs smart meter streams without privacy disclosure, the BCH decoding process will be performed on the encrypted form of retrieved secret bits. The main algebraic BCH decoding steps are syndrome computations, error locator's polynomial determination, error roots findings, and finally inversing and correcting [103] which can be summarised as follows:

- Let $r(x) = r_0 + r_1x + r_2x^2 + ... + r_{n-1}x^{n-1}$ be the received codewords, c is secret bits and $e(x)$ is the error pattern. Then, $r(x) = c(x) + e(x)$.

- The required syndromes are $2t$ as $s = s_1, s_2, ..., s_{2t}$ and is calculated by $r \times H^t$, where H is a parity-check matrix related to BCH (n,k,t) and its members α are

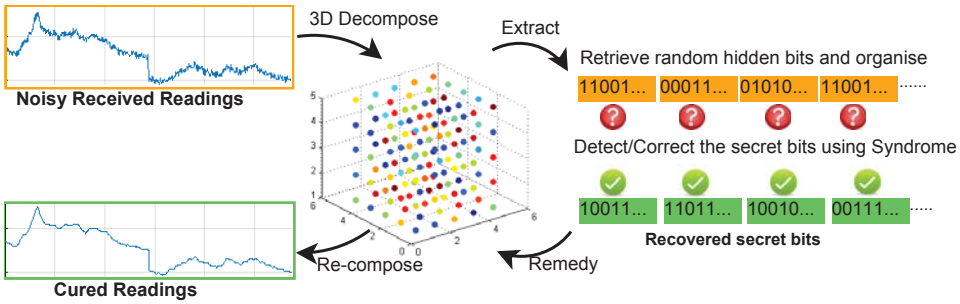

Figure 3.9: Detecting and correcting the random hidden secret bits (in their encrypted form) and remedy the received noisy readings at the operation centres.

the primitive elements in the chosen Galois field. H is shown in Eq 3.8.

$$H = \begin{bmatrix} 1 & \alpha & \alpha^2 & \cdots & \alpha^{n-1} \\ 1 & \alpha^3 & (\alpha^3)^2 & \cdots & (\alpha^3)^{n-1} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & \alpha^{2t-1} & (\alpha^{2t-1})^2 & \cdots & (\alpha^{2t-1})^{n-1} \end{bmatrix} \tag{3.8}$$

- Every s_i is a result of dividing $r(x)$ by a consecutive minimal polynomial $\varnothing_i(x)$ of α_i in the chosen Galois field.

- If any $s(\alpha^i) \neq 0$, this means that the transmitted CR smart meter readings have been tampered with and so these syndromes will be used to locate the errors as shown in Eqs 3.9 and 3.10.

$$\sigma^{\mu+1}(x) = \sigma^{\mu}(x) + d_{\mu} \times d_{\rho}^{-1} \times x^{2(\mu-\rho)} \times \sigma^{\rho}(x) \tag{3.9}$$

$$d_{\mu+1} = s_{2\mu+3} + \sigma_1^{\mu+1} s_{2\mu+2} + \sigma_2^{\mu+1} s_{2\mu+1} + ... + \sigma_l^{\mu+1} s_{2\mu+3-l} \tag{3.10}$$

where σ_i is the coefficients in the i-th term in $\sigma^{\mu}(x)$, μ is $\frac{1}{2} \leqslant \mu \leqslant t$, ρ are the preceding values and l is the degree of current $\sigma^{\mu+1}(x)$.

- The roots of the located errors $\Lambda(x)$ can be found in the form of Eq 3.11.

$$\Lambda(x) = (\alpha^{i1}x - 1)(\alpha^{i2}x - 1)....(\alpha^{iv}x - 1) \tag{3.11}$$

- The calculated roots $\Lambda(x)$ will be inversed within Galois field and used to recover the corrupted secret bits.

Finally, the recovered version of the secret bits is used to remedy the received stego form of IoTs smart meter readings as is shown in Fig 3.9.

However, only the legitimate receiver can decrypt the secret bits by using the key and to reveal the sensitive information. Fig. 3.8 shows an overview of the extraction, correction and remedy processes.

3.4 Evaluation

This section focuses on the proposed algorithm evaluation in terms of the key strength, the security of hidden secret information, the maximum size of household's sensitive data that can be hidden, the distortion measurements, and detection and correction capabilities.

3.4.1 Key Strength

The security of the proposed BCH-based steganographic technique relies on the fact that, unless the collected IoTs readings and the security key are known in full and in advance by the recipient, the hidden secret information cannot be retrieved, decoded and decrypted correctly.

However, the security key is the most critical parameter, because it is used to enforce two layers of security as seen in Figs. 3.3, 3.5 and 3.6: (1) encrypt the sensitive information, and (2) generate a random coefficients' order in the form of the 3D $\widetilde{X} \times \widetilde{Y} \times \widetilde{Z}$ matrix to hide the encoded sensitive bits. Therefore, this key should be kept secret and known only to two parties: (a) the sender (remote IoTs smart meter) where the key should be burned and used at any time the collected IoTs readings are sent, and (b) the legitimate receiver (operation centres) which can properly extract, decode and verify the validity of the hidden sensitive information. Other parties, including the cloud, can only see the transmitted stego IoTs readings.

In this chapter, the key is generated and will be kept secret at the IoTs end-point device and at the recipient's ends. The key strength of our algorithm can be quantified as the number of entropy bits Δ (see Eq. 3.12) where 2^{Δ} is the total

possible combinations that would have to be exhausted by illegitimate parties during a brute-force attack.

$$\Delta = \log_2 S_p^L \tag{3.12}$$

where L is the symbol's length and S_p is the symbols' probabilities. Table 3.1 shows examples of various key lengths, key symbols sets and the total number of possible combinations. Therefore, the longer the key and the symbol combinations are, the stronger is the algorithm.

3.4.2 Unlawful Retrieval

To prevent certain parties from unlawfully extracting the hidden secret information correctly without the key, which is a brute-force attack, the rescaled $3D$ $X \times Y \times Z$ coefficients matrix after 3D DWT decomposition of the host data, meaning IoTs smart device readings, should have an appropriate size (e.g. $>$ Key length) (see Eq 3.13).

$$\mathbb{P} = \sum_{i=\varphi}^{x} X! \times \sum_{j=\varphi}^{y} Y! \times \sum_{k=\varphi}^{z} Z! \times \Delta^L \times \prod GF(p^m)! \tag{3.13}$$

where $\mathbb{P}$ is the total possible combinations, X, Y and Z are the rescaled 3D coefficients matrix, φ is a threshold that represents the minimum selected coefficients number from each row, L is the key length, GF is the chosen Galois field, p and m are its base and space respectively.

For example, assume IoTs smart meter wattage readings are one hour in length, and rescaled to 3D coefficients matrix of size $32 \times 16 \times 16$ after applying 5 levels 3D DWT. The assumed threshold is $16 \times 8 \times 8$, the key symbol collection is UTF-16, its length is 128, and the chosen Galois field of base 2 and space 9 (see Eq 3.14).

$$\mathbb{P} = \sum_{i=16}^{32} 32! \times \sum_{j=8}^{16} 16! \times \sum_{k=8}^{16} 16! \times 65536^{128} \times \prod GF(2^9)! \Rightarrow \mathbb{P} = \infty \tag{3.14}$$

This proves that accurately retrieving and decrypting the intended secret sensitive information is highly impossible.

Table 3.1: Example of various used keys strength

Key length	Symbol Set	Possibilities
64	US-ASCII	7.3e+134
64	UTF-8	1.4e+154
64	UTF-16	∞
128	US-ASCII	5.3e+269
128	UTF-8	∞
128	UTF-16	∞
256	US-ASCII	∞
256	UTF-8	∞
256	UTF-16	∞

3.4.3 Hiding Capacity

The maximum amount of hidden data fundamentally relies on two parameters: (1) the total number of transmitted periodically collected readings (e.g. smart meter samples) and (2) the steganography level (or number of hidden secret bits in each sub-band coefficient) (see Eq 3.15).

$$M = \sum_{i=1}^{n}((X \times Y \times Z) - H_c) \times S_l \tag{3.15}$$

where M is the maximum number of hidden secret bits, n is the total IoTs smart device readings, X, Y and Z are the rows, columns and the depth of the rescaled 3D coefficients matrix after applying 3D DWT decomposition to the original readings, H_c is the high frequency coefficients and S_l is the steganography level in each sub-band coefficient.

For better understanding, assume that 5-level 3D DWT decomposition is applied to normal IoTs smart meter readings, each resultant sub-band coefficients number is 512 (i.e. value of n). The size of the rescaled $3D$ matrix is $32 \times 16 \times 16$ (i.e. values of X, Y and Z). Also, assume about 5 bits (i.e. value of S) are hidden in each coefficient. Therefore, around 5110 bytes (5 KB) of sensitive secret data can be hidden inside these coefficients.

3.4.4 Distortion Measurements

To carefully evaluate the effect of our algorithm on the transmitted IoTs streams and to certify their usability in their stego format, the difference between the original and stego forms of the readings with resultant distortion has been completely monitored using a widely-known measurement called a 'percent of root-mean-square difference' (PRD). The PRD can accurately measure any reconstruction error between the original and the reconstructed forms of the signal as shown in Eq. 3.16 [104].

$$PRD_j = \sqrt{\frac{\sum_{i=1}^{n}(R_i - \widetilde{R_i})}{\sum_{i=1}^{n}(R_i^2)}} \times 100\% \tag{3.16}$$

where R_i and $\widetilde{R}_i$ are the original and the reconstructed form of the sub-bands coefficients, and n is the length of the transmitted IoTs smart meter readings.

Similarly, the PRD is also used to accurately measure the resultant distortion after the retrieval and correction processes between the original and the recovered form of signals after correcting the hidden secret bits. All results are presented in Section 3.5.

In addition, to deeply highlight the detailed impact of the proposed algorithm on the frequency domain level, we used another well-known benchmark called 'Root Mean Square' (RMS) [105]. The RMS uses a mathematical model called 'Parseval's theorem' to precisely monitor the differences between two signals on frequency domain coefficient level as shown in Eq 3.17.

$$RMS = \frac{1}{N}\sum_{m=1}^{N} \mid X(m) - Y(m) \mid^2 \tag{3.17}$$

where X and Y are the 3D wavelet form of coefficients before and after applying the algorithm. N is the total number of manipulated coefficients.

The same process has been repeated to measure the feasibility of our remedy operation on the distorted form of received readings. The obtained results have been summarised in Section 3.5.

3.4.5 Correction Capabilities

To avoid biased results while testing the feasibility of our algorithm, the ability of detecting and correcting the hidden secret information in their encrypted form, as

well as using the recovered bits to remedy the received readings, have been carefully examined. To achieve that goal, three main steps have been followed. (1) The transmitted stego form has been attacked using different levels of random noise, such as Gaussian noise, which mimics the impact of many random processes that occur in nature that is the CR shared spectrum characteristics [106]. (2) The retrieved and corrected form of secret hidden information was examined using a well-known measurement called 'Bit Error Rate' (BER) which is the number of bit errors per unit time (see Eq. 3.18). Therefore, the lower the BER is, the better the correction is. (3) The recovered form of stego readings, after their remedy using the corrected secret bits, has been compared with their original format using PRD. (See Section 3.5 for results.)

$$BER = \frac{B_{err}}{B_{total}} \times 100\% \tag{3.18}$$

where B_{err} is the total corrupted bits, and B_{total} is the total original hidden secret bits.

It should be mentioned that every BCH syndrome code (n,k) has a maximum theoretical entropy t (or maximum correction capability) based on the chosen Galois field space and the number of secret bits k in each codeword n [103]. However, the beauty of BCH syndrome codes is that (1) the entropy of detection capabilities are almost double the correction and so it can easily detect the occurrence of the manipulation, and (2) the t can be dynamically determined and varied even in the same Galois field. In other words, it can be scaled based on the criteria of the CR shared spectrum - the noisier the channel, the higher t should be and vice versa.

3.4.6 Complexity Analysis

Because of the smart meters power and memory constrains, the algorithm's functionalities have been designed carefully to avoid the worst computational complexity such as exponential and factorial. Therefore, the worst computational complexity of our main functions has been thoroughly measured. (1) Theoretically using big $\mathcal{O}$ notations (see Table 3.2). From Table 3.2, it should be noticed that

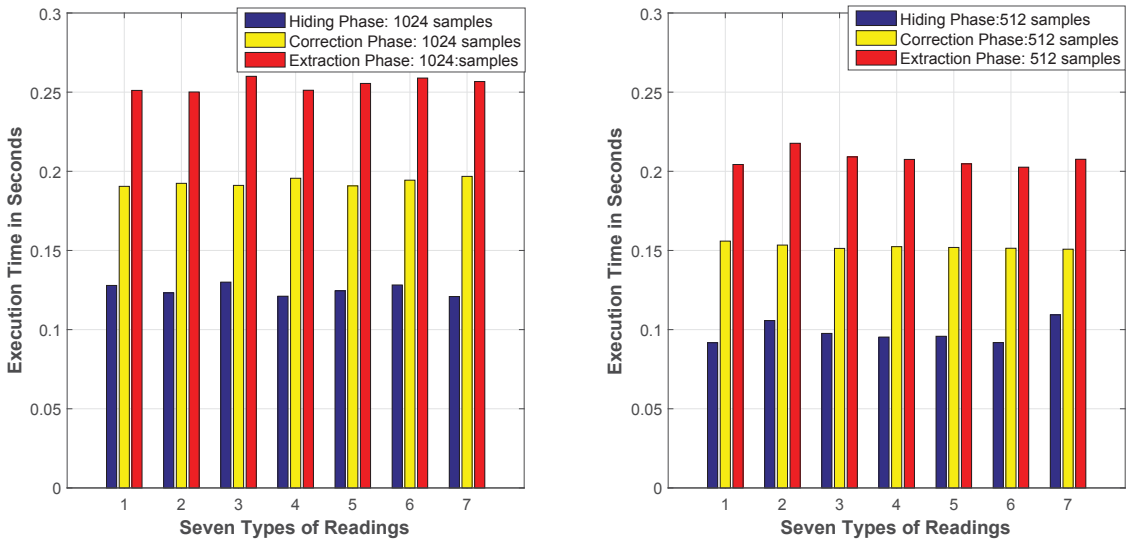

Figure 3.10: The required time and space needed by our algorithm to accomplish both hiding and retrieval process: (a) 7 types of readings of size 512 samples and (b) 7 examples of readings of size 1024 samples.

most of our functions are linear and have stable time complexity including random order generation where it has been improved from quadratic to linear by using radix sorting. Also, to avoid the complexity of the decomposition and reconstruction of 3D wavelet, a single-level 3D-wavelet was chosen instead of the real multi-level 3D. (2) Experimentally by involving all resultant coefficients from all used datasets in the process and proved that it is very low - < 0.3 seconds - in all the cases as shown in Fig 3.10.

Table 3.2: Algorithm Functionalities Computational Complexity

Complexity	Time			Space
	Best	Average	Worst	Worst
3D-DWT/Inverse	$\mathcal{O}(n^2)$	$\mathcal{O}(n^c)$	$\mathcal{O}(n^c)$	$\mathcal{O}(n^c)$
Random Order	$\mathcal{O}(nk)$	$\mathcal{O}(nk)$	$\mathcal{O}(nk)$	$\mathcal{O}(n+k)$
Scramble Secret	$\mathcal{O}(m)$	$\mathcal{O}(m)$	$\mathcal{O}(m)$	$\mathcal{O}(m)$
BCH Encode	$\mathcal{O}(w)$	$\mathcal{O}(w)$	$\mathcal{O}(w)$	$\mathcal{O}(w)$
Embedding	$\mathcal{O}(n^c b)$	$\mathcal{O}(n^c b)$	$\mathcal{O}(n^c b)$	$\mathcal{O}(n^c + m)$
BCH Decode	$\mathcal{O}(wt)$	$\mathcal{O}(wt)$	$\mathcal{O}(wt)$	$\mathcal{O}(wt)$

The main functionalities of our algorithm (see Table 3.2) especially that should run on the remote smart meters are 3D-DWT and its inverse, random order generation,

encoded information scrambling, embedding and BCH encoding. In addition, the correction algorithm (BCH decoding) runs on more powerful machines, such as operating centres. Firstly, let's assume $f(n)$ is $\mathcal{O}(g(n))$ if f grows at most as fast as g. Therefore, $f(n) = \mathcal{O}(g(n))$ only if there exists c, $n_0 \epsilon \mathbb{R}^+$ such that for all $n \geq n_0$, $f(n) \leq c.g(n)$. Thereby, for each 1D vector of collected reading of size n (i.e. varies from 512 to 4096 in this example), the worst complexity of these functionalities is as follows. (i) $\mathcal{O}(n^l)$ for both the time and space required for 3D-DWT and its inverse [65], where l is the dimensional level. This is caused by using single-level 3D instead of the much more complicated multi-level 3D. (ii) $\mathcal{O}(nk)$ is for generating random orders using a constant key length $k \epsilon \mathbb{Z}$ - 128 in the implementations - where it has been improved from quadratic to linear by using radix sorting. The maximum required space will be an accumulated size of $n \quad and \quad k$. (iii) The worst time and space complexity for scrambling the encoded information of size m where at least $< n/2$ is $\mathcal{O}(m)$. (iv) $\mathcal{O}(w)$ -linear- for both the time and space required for BCH encode, where $w \epsilon \mathbb{Z}$ is the number of codewords. (v) $\mathcal{O}(n^l b)$ for embedding $b \epsilon \mathbb{Z}$ (varies from 1 to 5) the number of bits per coefficient of total size n, whereas the required space will be a total size of both $n^l \quad and \quad m$. (vi) $\mathcal{O}(wt)$ is for the time required for BCH decode [107], where $w \epsilon \mathbb{Z}$ is the number of codewords. However, the maximum required space will be an accumulated size of $w \quad and \quad t$.

Finally, it is clear from the time complexity analysis that the best, average and worst complexities are stable and almost the same. This is because all the collected readings of size n are used in each functionality of the hiding and correction algorithms. In addition, to avoid the cost of the 3D model, most of the functionalities have been designed to have complexity less than polynomial especially those run on the smart meters. Therefore, the smaller the transmitted periodical readings are, the less complex is the algorithm.

3.5 Implementations

3.5.1 Datasets

The datasets of various IoTs smart end-point devices readings have been used in our experiments, and collected and published by Laboratory for Advanced System Software as part of a project named 'Smart*'[75, 76]. The datasets comprise continuous readings, every minute, from three homes for three months. The readings are categorised into: (1) power usage, such as watts consumption and heat index, and (2) environmental features, such as inside and outside temperature, inside and outside humidity and wind chill. The datasets also provides periodic electricity power consumption, every minute, from around 400 anonymous homes for $(3 \times 30 \times 24)$ hours. Based on the definition of the spatial and temporal aggregations [77], these readings are temporal, because they are collected separately from every single house after equipping them with a smart meter that gathers its readings periodically.

3.5.2 Cognitive Radio Characteristics

In our experiments, a well-known CR sensing mechanism which is standardised as IEEE 802.22 [10] has been simulated. This standard encompasses cognitive capabilities such as sensing interface, spectrum management, Geo-location and database access. Table 3.3 depicts the key parameters.

Table 3.3: Cognitive Radio Key Parameters [10]

Feature	Value
Operating Frequency	$54 \sim 862$ MHZ
Users	≤ 255 Channels
Sensing Time	$\leq 2ms$ Per channel
Burst Allocation	*Linear*
Self-Coexisting	Dynamic spectrum sharing
Superscription ratio	$40:1$
Capacity	$1.5mbps$ Upstream $384kbps$ Downstream
Area	Typically $33km$ Radius

3.5.3 Experiments Setup

Our main performed experiments are four steps which can be categorised into two main parts and each has two steps. The first part which is performed by distributed CR smart grids as follows. (1) Encoding of CR smart meter and household secret information using BCH. (2) The random hiding by embedding the encoded secret bits in the household collected normal readings and the steps explained in Section 3.3.2. The second part will be at the recipient's end (e.g. operation centres) as follows. (3) Retrieving the hidden bits as described in 3.3.2, and (4) decoding the secret bits using BCH. Therefore, even if the IoTs smart meter normal readings that contain the hidden confidential information are intercepted or tampered with intentionally (i.e. unauthorised parties) or unintentionally (i.e. channel noise), (1) it will not disclose any information, (2) it can be easily detected and recovered.

In this chapter, all aforementioned types of readings were used to fully observe the feasibility of implementing the proposed algorithm on various smart meters. For each, experiments were performed to embed and retrieve the encoded sensitive information according to our proposed algorithm described in Sections 3.3.2. The sensitive data was a set of information that have to be secret which are related to smart meter (e.g. ID, geometric location), and household (e.g. name, DoB, address and the total watts consumption). This information is all converted into bits to be encoded and randomly embedded inside IoTs smart meter readings.

To avoid biased results, our proposed algorithm was experimented with different key lengths as well as various IoTs streams with lengths such as 512, 1024, 2048, 4096 and more. Also, to obtain the most possible worst distortion result, all low-frequency 3D sub-bands coefficients have been used.

3.5.4 Base Testing Scenario

To validate the effectiveness of the error detection and correction capabilities of our algorithm, these steps have been followed. (1) We used various best codeword combinations in every Galois field space; for example, (7,4,1), (15,5,3),(31,6,7) and

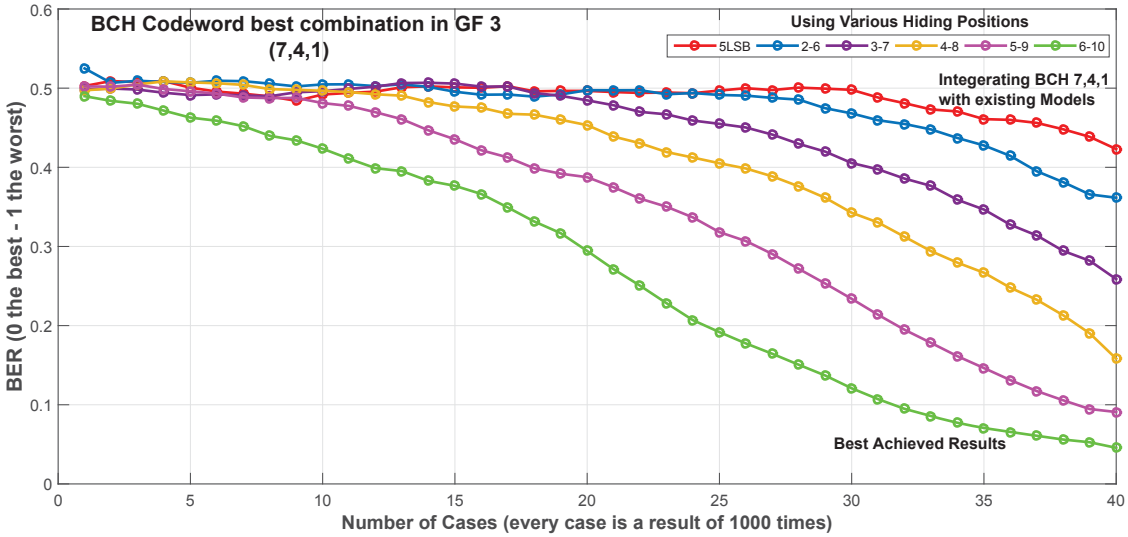

Figure 3.11: Comparison of 40 cases (i.e. after applying 40 random noise levels into the transmitted stego form of readings) of BER of the recovered secret bits (i.e. after extraction, decoding and correction) using BCH codes (7,4,1) and various cases of hiding positions.

(65,7,15). The size of codeword n in Galois field will be 7, 15, 31, 63, 127, 255 and 511 for an easier understanding. Each has maximum theoretical recovery capabilities t. For example, for $k = 7 \Rightarrow t = 1$, for $k = 15 \Rightarrow t = 3$, for $k = 31 \Rightarrow t = 7$, for $k = 63 \Rightarrow t = 15$ and so on. So in our experiments, and to avoid unbiased results, the same levels of random noise have been applied more than 1000 times on every best (n, k, t) possible combinations. (2) During the transmission of the stego readings that contain the random hidden encoded bits, various random Gaussian noise levels from the worst (-9) to medium (30) were imposed to simulate the possible CR channels interference. To get stable and precise recovery capabilities of our algorithm, these experiments were repeated more than 1000 times with every single codeword combination possible. (3) To examine the possibility of improving detection and recovery capabilities of our algorithm, different hiding positions from 1-to-5, 2-6, 3-7, 4-8, 5-9 and 6-10 were used with every codeword. (4) Similarly, these steps have been repeated to compare the best outcomes from the proposed algorithm with previous models as shown in Section 3.5.7.

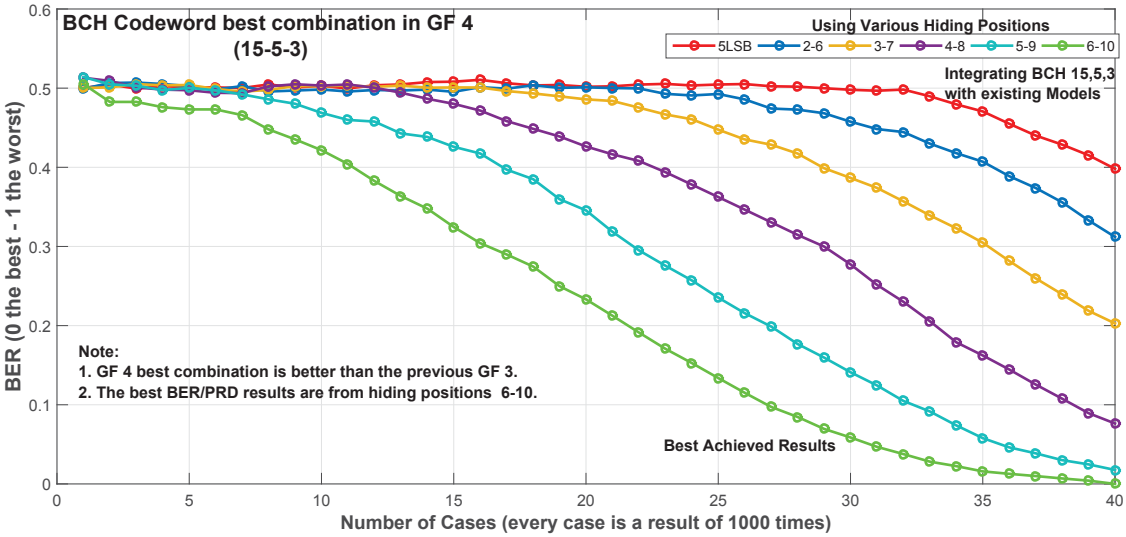

Figure 3.12: Comparison of 40 cases (i.e. using 40 random noise levels) of BER of the recovered secret bits using BCH codes (15,5,3) and various cases of hiding positions.

In all the cases, three benchmarks were carefully measured after the correction and remedy processes of the received stego form of readings. These are (1) the possible recovery capabilities using BER, (2) the remedy effectiveness on the time domain using PRD, and (3) the remedy effectiveness on the frequency domain at 3D level using RMS.

3.5.5 Results

Our main concerns discussed in the results are that after applying random levels of noise to the transmitted stego readings, can that manipulation be detected? Can the hidden secret bits be recovered? And, can the recovered secret bits be used to remedy the received signal? For brevity in this paper, only a few cases of our results are presented. (1) Fig. 3.11 shows 40 cases after applying 40 different noise levels into the transmitted stego form of readings of BER of the recovered secret bits (i.e. after extraction, decoding and correction) using BCH codeword (7,4,1) and various cases of hiding positions. (2) Fig. 3.12, 3.13 and 3.14 also show 40 cases of BER of the recovered secret bits using BCH codewords (15,5,3),(31,6,7) and

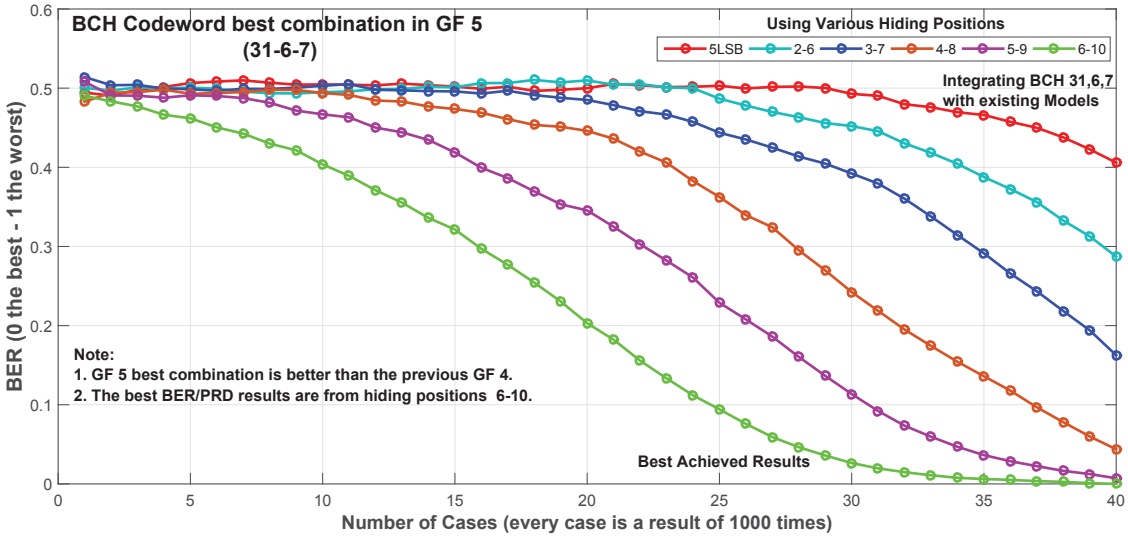

Figure 3.13: Comparison of 40 cases (i.e. using 40 random noise levels) of BER of the recovered secret bits using BCH codes (31,6,7) and various cases of hiding positions.

(65,7,15) respectively with the same hiding positions cases. (3) Fig 3.15 shows a comparison between different BCH codewords combinations using the best observed resultant hiding positions (i.e. 6-10) from previous experiments. (4) Figs. 3.18 and 3.19 show the exact PRD and RMS results - from all hiding positions combinations used - between the original and stego forms before applying the noise, as well as between the original and the cured forms (i.e. after recovering the secret bits and remedying the signal). (5) Fig. 3.16 shows an example of a plot of three original IoTs smart meter readings (i.e. power consumption) used to hide sensitive information, and the stego form before transmission and after recovery process. (6) Finally, Fig. 3.17 shows an example of another plot of six original IoTs smart meter readings (i.e. inside/outside temperature and humidity, wind chill and heat index), and the stego and the recovered forms.

3.5.6 Discussion

In all cases, despite the different sample lengths of the IoTs readings, the various characteristics and value ranges, all PRDs are $< 1\%$, RMS < 0.01 and up to BER=0.

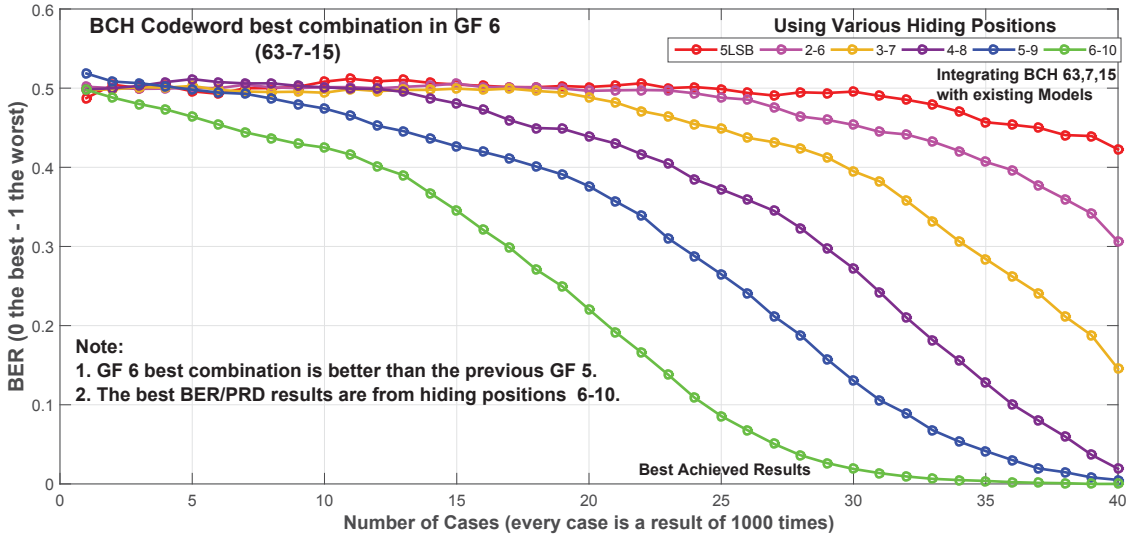

Figure 3.14: Comparison of 40 cases (i.e. using 40 random noise levels) of BER of the recovered secret bits using BCH codes (63,7,15) and various cases of hiding positions.

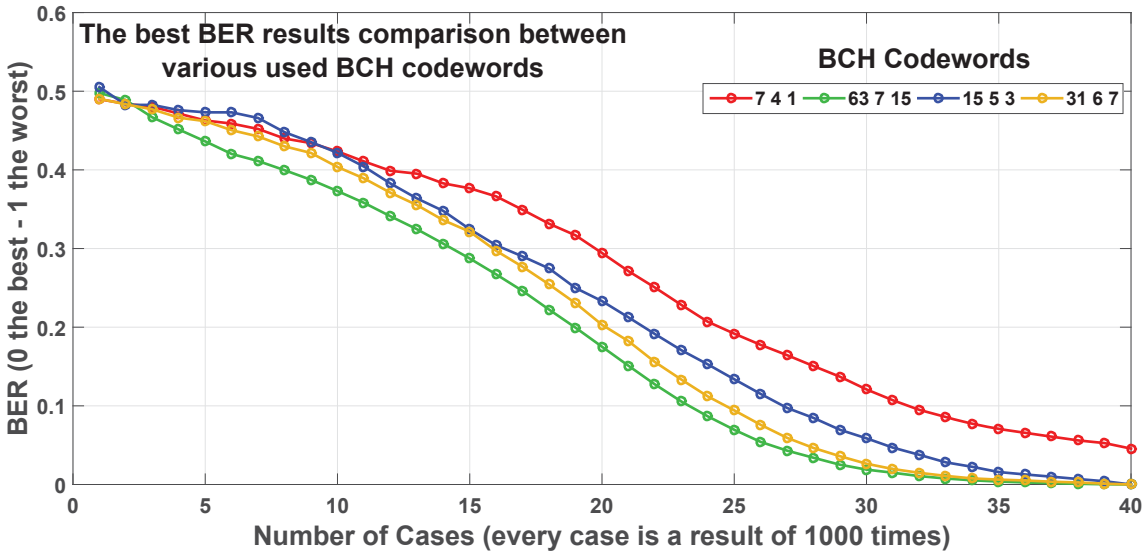

Figure 3.15: Comparison of 40 cases (i.e. using 40 random noise levels) of BER of the recovered secret bits using best hiding positions with various BCH syndrome codes(n,k,t).

This proves that our proposed algorithm is stable with little distortion impact, reliable recovery and remedy mechanisms. However, it should be noticed that from the detailed experiments performed, it was discovered that the same BCH codewords have different

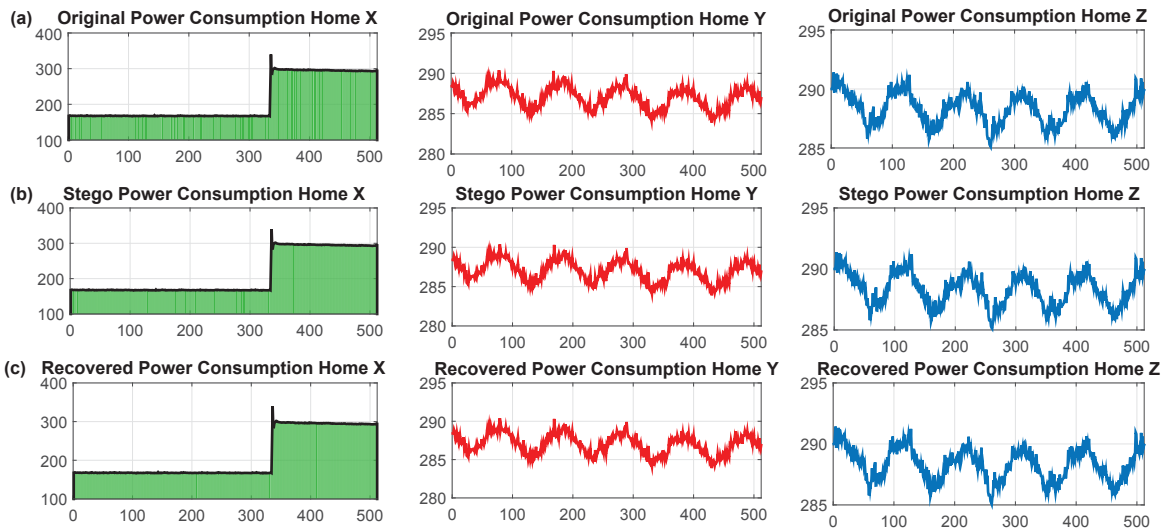

Figure 3.16: Three examples of watts consumptions' readings collected from different homes: (a) direct plot for original form (b) stego form that contains the hidden sensitive information (i.e. grid ID, household name, DoB, address and total watts) and (c) recovered form (i.e. after applying the noise and remedy processes).

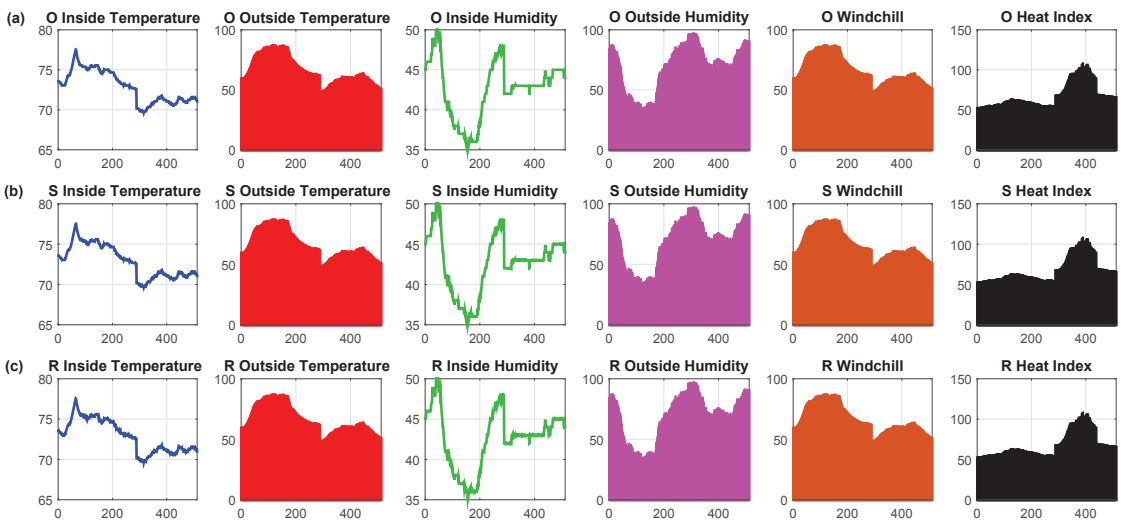

Figure 3.17: Six additional examples of possible IoTs (i.e. smart meter) readings: (a) Direct plot for original form (b) stego form that contains the encoded hidden information and (c) recovered form (i.e. after applying the noise and remedy processes).

behaviours, being the recovery capabilities, based on the hiding positions used. For example, the least significant hiding positions (1 to 5) that have been used in lots of papers such as [2–4], are the worst in terms of detecting and correcting the random hidden bits, whereas the newly configured hiding positions from 6 to 10 are the best in terms of recovery. Additionally, BCH codeword (63,7,15) has the highest ability to detect and recover data among these various codes whenever the transmitted stego readings are faced with high noise interference: it can recover more random error bits with the penalty of more control bits and less payload. On the other hand, BCH codeword (7,4,1) is the most desirable among these codes in terms of detecting and correcting capabilities when the channel is faced with less noise or attack; for example one corrupted bit in every seven bits.

In summary, the hiding positions (6 to 10) and the BCH codeword (63,7,15) are the most powerful combinations whenever the transmission channels are prone to higher noise and attacks, especially in shared environments such as CR spectrum. Their resultant PRDs are all $< 1\%$, RMS $< 0.01\%$ and BER $=0$. The BER results mean that all secret hiding information can be recovered from the distorted readings. The PRD and RMS results mean that the distorted reading can be cured using the corrected hidden information and the effect will only be to the third or fourth decimal digits, which are usually ignored in many cases, such as watts, temperature and humidity. On the other hand, it offers a great solution with a paradigm shift for securing the sensitive household information that should be transmitted as well as the authenticity of IoTs smart device readings. The advantages are as follows. (1) Strong detection and correction mechanisms recover both the hidden bits and the transmitted readings up to the theoretical entropy without any increase in the actual size of smart meter readings. (2) There is no change to the original form of readings that facilitate the usage of operational administration such as cloud providers by the authorised recipients without disclosing any confidential information.

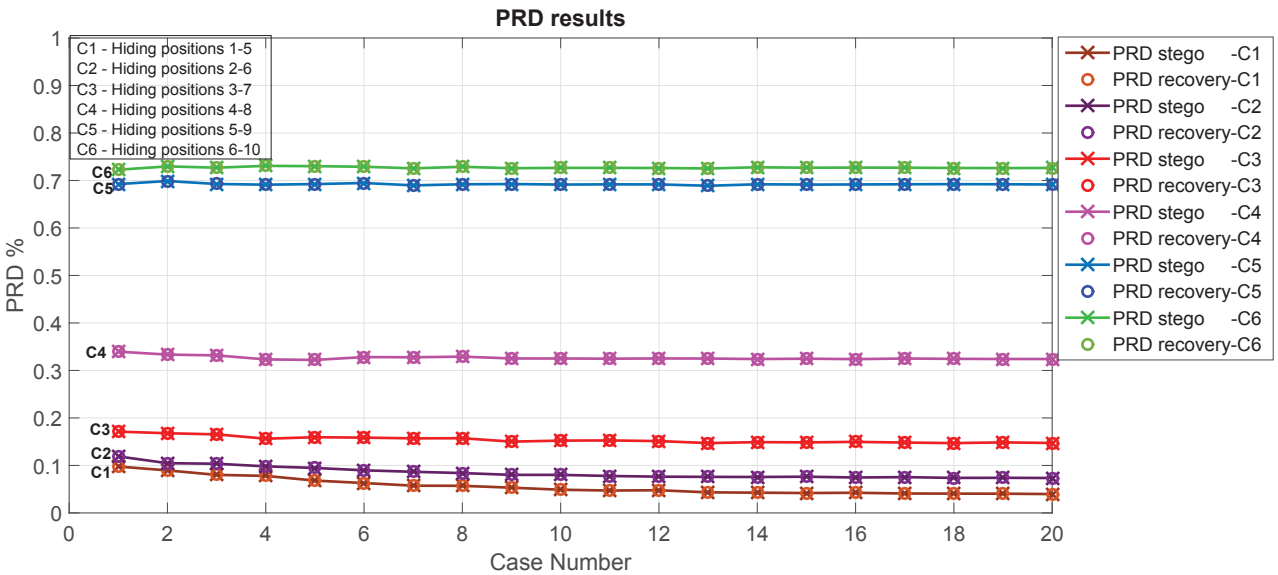

Figure 3.18: (20×6) PRD results obtained from various combinations of used hiding positions both (a) after hiding and (b) after recovery of lost bits and remedy the received signal in relation to BER. This proves that all resultant distortion has been minimised to $< 1\%$ which means the readings still can be used.

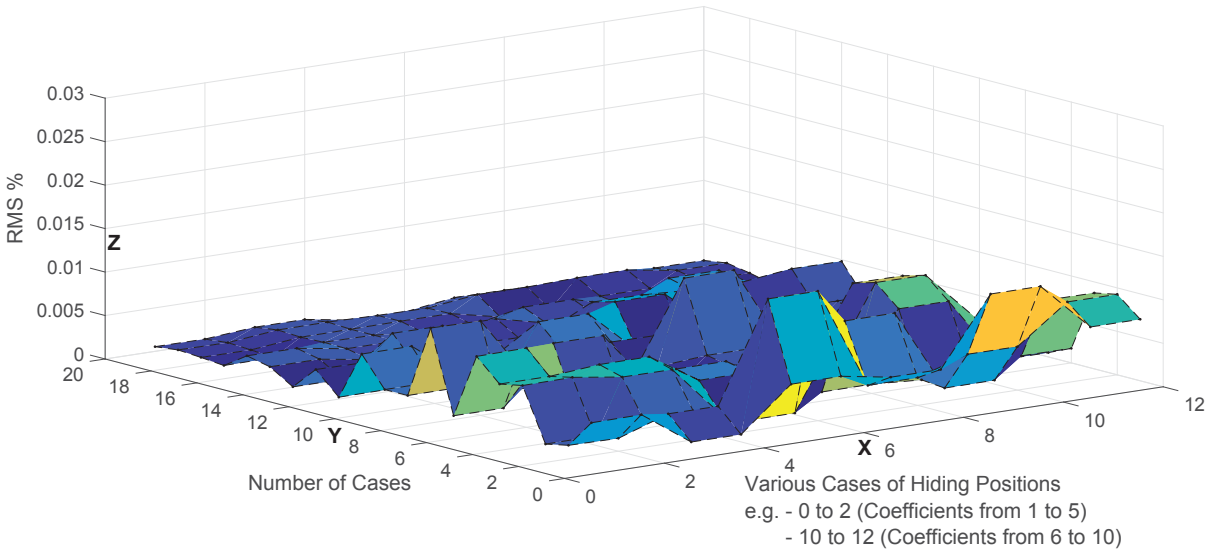

Figure 3.19: (20×6) RMS results obtained from various combinations of used hiding positions (i.e. as in Figures 3.11, 3.12,3.13 and 3.14) on 3D frequency domain coefficients level both (a) after hiding and (b) after recovery of lost bits and remedy the received signal in relation to BER. This also proves that even on the 3D frequency domain coefficient level, the resultant distortion are very low $< 0.01\%$ which means the recovery of the original readings is possible after the interference at CR channels.

3.5.7 Comparison with Existing Models

To the best of our knowledge, there is no research that combines steganography with error detection and correction techniques, such as BCH syndrome codes, to embed household sensitive information in their IoTs smart device readings in such a way that it helps to protect the privacy, and authenticate and provide robustness, as in detecting and recovering the lost information, without hindering the efficiency of direct operations on the transmitted readings. However, there are recently proposed works [2], [3] and [4] that use only steganography in a different context. Therefore, our proposed technique is compared with these recent models in [2] where a steganography used to hide tenants' confidential information in their collected premises readings using signal processing (Wavelet-stego), whereas the model in [3] was to preserve the privacy of the remote IoTs nodes using Walsh-stego. On the other hand, the model in [4] proposes the least significant qubit (LSQb-stego) information hiding algorithm for quantum images.

There are three main improvements in our algorithm. (1) After applying 40 levels of random noise on the stego form of readings of all models (see Fig 3.20), it should be noticed that our algorithm is more robust in terms of detecting and recovering the distorted hidden information as well as curing the received form of readings (i.e. up to the theoretical entropy) than the models in [2], [3] and [4]. This is because privacy was the only concern in those models, so they have zero detection and recovery capabilities, whereas the targets of this algorithm are privacy, authenticity and robustness. (2) The selected hiding positions of the models [2], [3] and [4] (i.e. LSB) are much more prone to be lost whenever the transmitted readings are faced with intentional interference or unintentional noisy channel attacks as shown in our results (see Fig. 3.20). Therefore, the hiding positions of the proposed algorithm were chosen carefully to minimize the loss to the lowest possible level. (3) All algorithms use a security key to cipher the sensitive information; however, the proposed algorithm is much more secure in terms of generating the random hiding order matrix. This is because in the proposed model, this matrix is dynamically generated on the fly using the key in 3D format, whereas

in the model [3] the random order is only in a vector complexity, in [2] is in a 2D complexity and in [4] relies on the randomness of the used frequency transformation technique.

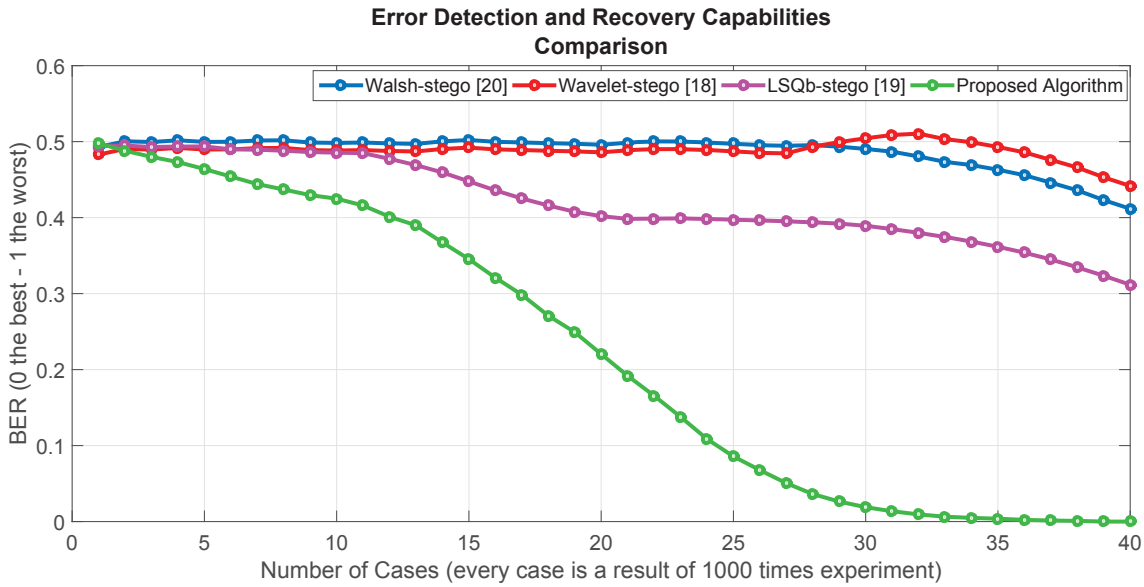

Figure 3.20: Comparison of 40 cases (i.e. using 40 random noise level cases) of detection and recovery capabilities (i.e. BER) between the proposed algorithm and the models in [2], [3] and [4] respectively.

3.6 Chapter Summary

In this chapter, a novel secure BCH based stenographic technique was proposed that allows legitimate recipients to: (1) detect and recover any loss due to CR shared spectrum noise from the hidden sensitive information without privacy disclosure; (2) remedy the received stego readings by using the corrected version of the secret hidden data, and (3) directly work on the stego form of the normal IoTs readings without neglecting the privacy or the authenticity of the received readings. To guarantee the minimum distortion, high randomisation and robust detection and recovery, a 3D WT is used to decompose the normal IoTs readings to their frequency domain. The less significant sub-bands are used to embed the encoded sensitive information. To thoroughly measure the detection and recovery capabilities, random noise levels are

applied to the transmitted readings. Then, the detection, recovery of the sensitive information and the remedy of stego readings are deeply examined using BER, PRD and RMS. It is evident from the experiments that our technique has solid recovery capabilities (i.e. BER = 0, PRD $< 1\%$ and RMS $< 0.01\%$).

Chapter 4

Protected IoTs Size Reduction

This chapter answers the third research question discussed in Section 1.2 by introducing two novel models. This chapter discusses the main concerns about the transmission of the unusually large size of Internet of Things (IoTs) readings, the resultant negative effect on the bit error rate and transmission energy consumption. Further on, we discuss the issues around the current lossy models and their effect on hidden secret information, and on the damage of transmitted readings. This chapter then highlights how a blind lossless compression technique is a likely candidate to overcome the reduction of protected IoTs readings size at intermediate hops without revealing the hidden information. This chapter then highlights how a blind lossless compression technique is a likely candidate to overcome the reduction of protected IoTs readings size at intermediate hops without revealing the hidden information. Section 4.1.5 introduces the main contributions of this chapter and how they have been accomplished. Section 4.2 briefly highlights the key-related works and the efforts that other researchers took to solve the issue. Section 4.3 explains in detail our first novel model including the design, Gaussian approximation, margin calculation, Burrow-Wheeler and Move-To-Front (MTF) transform, and entropy coding. Section 4.4 explains in detail our second novel model including, design, splitting, stable group reduction, noise group reduction, and code chaining. We evaluate various characteristics of our models, including a theoretical entropy, empirical ratio improvement, and performance analyses which are introduced in

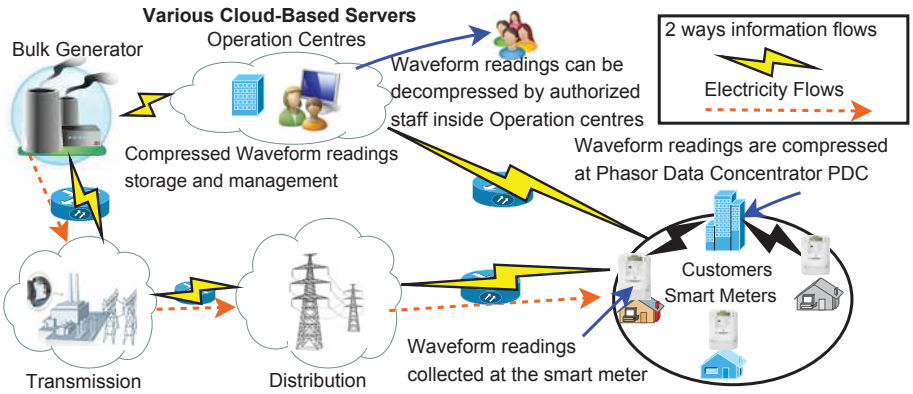

Figure 4.1: Our model where power consumptions readings from houses are collected as waveform readings and compressed before transmission to operation centres.

Section 4.5. Section 4.6 presents detailed experimental examinations of the models, including the test bed scenario and a wide range of protected IoTs readings after the stego process, their possible reduction ratios (i.e. compress), the actual IoTs protected high streams after decompression and comparison with available techniques. Section 4.7 finally summarises this chapter.

4.1 Introduction

The IoTs such as smart meters are currently being investigated and deployed into premises. These can be used to gather periodic waveform readings automatically every second, such as the power consumption of a premise, and can transmit them to operational centres using various techniques [79]. According to the latest survey conducted by the International Council on Large Electric Systems (CIGRE) [108], there are more than twelve key applications using cases that can be achieved from the distributed IoTs smart meters. At the top of the list are automatic metering services, load forecasting and energy feedback. Therefore, there will be an unusual volume of collected readings from small meter devices that should be transmitted simultaneously on real-time through limited bandwidth and low energy environment.

By reducing the huge size of the collection of readings before transmission, we noticed many operational benefits. These are discussed further into the chapter.

4.1.1 Limited Bandwidth Reservation

Many IoTs smart meter projects are connected through Narrowband Power-Line Communication (PLC) Links. In Germany, 13 out of 24 metering projects use PLC [109]. Narrowband PLC works at lower frequencies (3-500 kHz), lower data rates up to 100s of kbps, and has a longer range going into distance of kilometres. The collision probability increases relatively with the rise of data volume. Therefore, the more data every meter must transmit, the more time the bandwidth is required, as a result disallow or disrupt each other. With the compression model, the data rate decreases significantly, and likewise, the probability of the collusion.

4.1.2 Energy Saving

The power required for bits' transfer considerably exceeds the power needed for the computational complexity of operations on the same device. For example, transmitting a single bit from embedded devices, such as Mica2Dot, is equivalent to executing around 2090 clock cycles [110]. Therefore, up to 6270 clock cycles in every 8260 are saved by the proposed compression in this chapter.

4.1.3 Bit Error Rate Reduction

By reducing the volume of collected data, the time interval required by every meter to transmit reduces significantly lowering the bit error rate this assists with avoiding repeated retransmissions of corrupted traffic [109]. This will open space allowing room for the addition of more devices; thereby, enhancing the reliability and extensibility of communication infrastructure.

4.1.4 Low Storage Cost

The cost of storing collected data increase exponentially due to the readings volume. For example, the generated data volume from IoTs smart meters projects of 40 million households in Germany alone exceeding 25TB per day [6]. Management of the data worsens due to many government regulations that enforce the preservation the data

for many years without losing any of its features. Therefore, compression is an ideal remedy for this problem.

Compression methods of IoTs waveform readings can be classified into two groups: 'lossy' and 'lossless' [30]. Lossy compression relies on losing some information while trying to maintain the main features of the waveform signal. Therefore, the decompressed signal is somewhat different from the original signal. This kind of compression was acceptable in the traditional grid model, and so lots of research went into this field (see summary table 4.1). However, lossy compression has been recently discouraged for two reasons: (1) the recent IoTs smart devices usage demonstrates the use of data for crucial purposes such as billings, and (2) to maintain the privacy and authenticity of the transmitted readings, current models are using steganography to hide the secret information randomly inside these readings [3]. Consequently, losing any bit of these readings is no longer tolerated.

In contrast, lossless compression has an obligation to reconstruct the exact waveform signal as the original with zero loss. With these constraints in mind, some research has been done in the following categories [5, 8, 9]. However, as per a recent state-of-the-art study [30], this route is far from being as mature as image, voice and video lossless compressions. Therefore, we were compelled to look for better lossless compression mechanisms that achieve a higher compression ratio, while preserving all features of IoTs transmitted readings. Therefore, the main question that drives this chapter is that, **how can the waveform IoTs protected readings be pre-processed to significantly reduce the entropy?**

4.1.5 Contributions

In this chapter, we introduce the compression algorithms of two novel lossless IoTs smart meter readings that reduces the protected readings size at intermediate hops without revealing the hidden information. The first generic Gaussian-based model target is representing IoTs smart device readings in few parameters regardless of

the roughness in the signal. This is successfully accomplished using the Gaussian approximation [111]. The difference between the approximated and the actual waveform is calculated. Therefore, the compression will be only for margin space rather than the entire stream of waveform readings. The margin space values are finally encoded. After thorough evaluation under the same conditions, our technique was superior to existing models mathematically (the entropy was halved) and empirically (the achieved ratio was 3.8:1). In the second target model (or the N-Split model), we lessened the randomness in the IoTs streams into a smaller finite field to expand the possibility of repetition and avoid the accumulated rounding errors due to floating operations. This has been successfully achieved by splitting the collected readings into different groups before reducing them independently. The achieved ratio was 4.48:1.

4.2 Related Work

Most studies have been conducted on the waveforms collected IoTs smart meter readings focusing on lossy compression. This is because (1) the readings were not directly transmitted and used for crucial purposes, such as billings and real-time analysis in the tradition grid; (2) the effectiveness of a transform technique called 'wavelet' helps to represent waveform signals in few values, this means losing some bits from every reading. The lossy compression can be grouped based on the used techniques into transform, parametric coding and mixed.

Firstly, transformation models such as the work of J. Ning et al. [33], is where discrete wavelet transform has been applied to identify most of the signal energy in low-frequency coefficients (i.e. using dbX), and allowed others to be removed. Additional work has been conducted using different families of wavelet such as Sluntlet [115] and B-Spline [116]. Secondly, parametric coding models such as the work of Tcheou Michel et al. [32] where they used damped sinusoids models to extract signal features before compression. Finally, mixed transformation and parametric models such as the work proposed by Ribeiro Moises et al. [121], is where fundamental

Table 4.1: Related Work Summary.

Set	Category	Main Tech	Rept CRi	Test CRi	Mat-ric	Value	Comments	Ref
Loss-less	Dictionary	Lemp-Ziv	5:1	2.5:1	-	-	Measurement should be bin-bin not text to text	[7]
	Entropy	Huffman & Del-Huf	1.7:1 & 2:1	1.8:1 2.2:1	-	-	Accurate numbers based on bin-to-bin	[5]
		Arithmic Code	2.5:1	2.4:1	-	-	Better than Huffman	[9, 112]
		Invert Golomb	2.8:1	2.8:1	-	-	Accurate results & less overhead	[8]
	Mixed	Prediction & LZMA	2.5:1	2.5:1	-	-	Improve 10% over LZMA	[113]
		Bzip2 & Del-Bzip2	2.8:1 & 3:1	2.7:1 2.9:1	-	-	The best existing lossless model	[9]
		General	2:1 & 4:1	1.7:1 & 1.9:1	-	-	Reported CRi tested with ≤ 2 decimal precision / Measured CRi Tested with 4 decimal	[114]
		Normalise based	2:1 & 5:1	1.5:1 & 2.2:1	-	-	Reported CRi tested with ≤ 2 decimal precision / Measured CRi tested with 4 decimal	[6]
Lossy	Transform	Daubech DWT	5:1	-	RMS	10^{-3}	Losing info	[33]
		Slantlet DWT	10:1	-	MSE	-19dB	Serious loss	[115]
		B-Spline DWT	15:1	-	MSE	-25dB	Very high distortion	[116]
		WPT & AC	6:1	-	NMSE	10^{-5}	Losing info.	[117]
		EZW	10-16:1	-	NMSE	10^{-5}	Losing info.	[118]
		Lifting WT & Huffman	20:1	-	SNR	> 25	Huge loss	[119]
		Singular Value	> 20	-	MAE MPE	4.1% 0.036	Losing many bits	[120]
	Param. Coding	Damped sinusoids modeling	16:1	-	SNR	$>$ 30dB	Very high distortion	[32]
	Mixed Trans & Param	Basic, Hamonic & Trans	16:1	-	MSE	30dB	Higher distortion	[121]

harmonic and transient coding are used together.

On the other hand, a few studies have been conducted on lossless compression due to its restrictions and nature of waveform readings. The lossless compression can be classified based on the technique used into the dictionary, entropy and mixed based models.

Dictionary-based algorithms rely mainly on general compressors (e.g. ZIP, GZIP and LZO) where a dictionary is built, more frequent samples will be represented in fewer bits, whereas more bits are allocated to less frequent samples; for example, Gerek Omer et al. [7] used Lempel-Ziv to compress a stream of waveform readings. The achieved compression ratio was 2.5:1 bin-to-bin. However, dictionary algorithms are fundamentally designed for letters (e.g. English characters) where the number of options is limited. This is ill-suited for waveform signals because of their floating point nature. This means every integer number has thousands of images due to its floating values.

Entropy-based algorithms are statistical models designed by measuring the probability of every symbol within a stream and allocating a fewer number of bits for a higher probability and vice-versa; for example, the work of Kraus Jan et al. [9], Arithmetic coding was used to replace the input symbols with a single floating-point value. The achieved compression ratio was 2.6:1. Zhang Dahai et al. [5] also proposed a model that improves Huffman coding by preprocessing the data by using a higher order delta modulation. The improvement was from 1.7 to 2.3:1. Additionally, Joseph Tate [8] recently introduced a model that uses Golomb-Rice coding after preprocessing the data using several methods, such as frequency compensated difference. The achieved compression ratio was 2.8:1.

Mixed algorithms are more sophisticated techniques using both dictionary and statistical algorithms. This allows the exploitation both the frequency of repetition and its probability within a stream of values. For example, Kraus Jan et al. [113] introduced a model that improves LZMA algorithm by reducing the redundancy in waveform readings. This is accomplished by using prediction models based on interval

selection optimisation and differential encoding. The achieved compression was 2.6:1. Kraus Jan and Tobiska Tomas [9] also proposed a model that improves BZIP2 by using delta modulation after applying an efficient block sorting Burrows-Wheeler algorithm. The achieved compression ratio was 2.9:1.

Additionally, M. Ringwelski et al. [114] utilised various general-purpose mixed algorithms (e.g. Adaptive Huffman Coding, tiny Lempel-Ziv Markov Chain, and Lempel Ziv Markov Chain Huffman Coding) and performed them on public data sets (i.e. REDD and TUD) that have either integer entries with one or a maximum two decimal points. The achieved compression ratio varies between 2-4:1. Andreas Unterweger et al. [6] also achieved a better compression ratio (e.g. $>$ 2.5-5:1) using the same data sets by proposing a multi-steps algorithm. They started by normalising the collected readings before deriving the differential coding. Then, variable length coding is applied followed by resultant code concatenation before entropy coding.

The effect of data granularity and a wider detailed precision (e.g. three decimal) of both works [6, 114] have been examined thoroughly by Andreas Unterweger et al. in [122]. Although they performed well in some appliances-levels, they become less effective at a coarser granularity especially when the decimal precision exceeds two. This is because both are designed and performed well for datasets that have either an integer, and one or a maximum two decimal precisions. In cases relating to recent use and published datasets (e.g. the datasets targeted in this paper [78, 79]), the recommended precision is four decimal. Table 4.1 summarises most of the related work.

4.3 Model 1 (MD1): Gaussian-based Model

The principle behind this proposed technique is to represent IoTs smart meter collected readings using only a few parameters as accurately as possible. Various curve fitting functions (e.g. radial base, polynomial, exponential, Gaussian, Fourier and linear) are investigated to approximate the signal using certain parameters while maintaining the minimum error margin. Gaussian approximation function is chosen

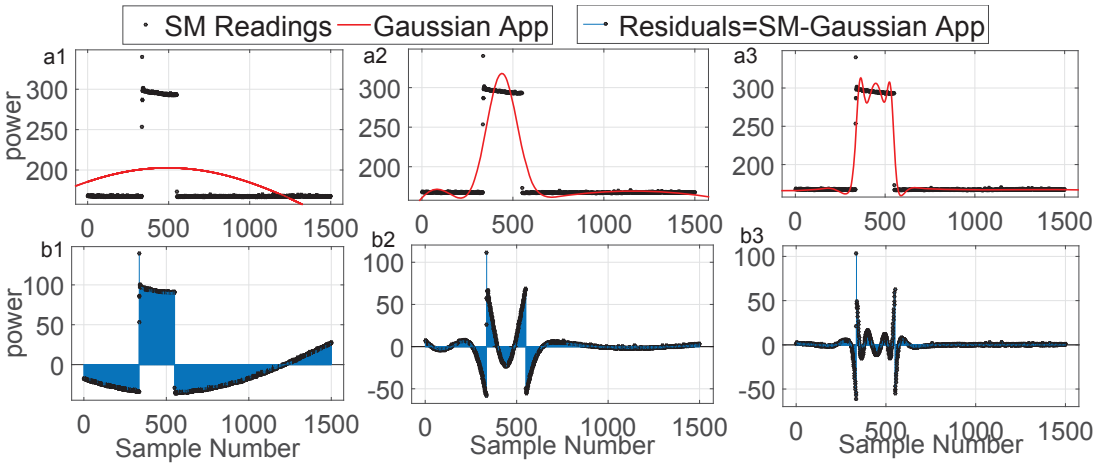

Figure 4.2: Three examples of Gaussian approximation optimisation. (a) Plot of more than 1500 IoTs (e.g. smart meter) power readings and their Gaussian approximations and (b) plot of the resultant residuals after calculating the margin (i.e. highlighted in blue). $b3$ is obviously better due to its very low residuals.

due to its superiority over the others. The margin space between the approximated and the actual readings has been calculated before encoding them using Burrow-Wheeler Transform (BWT). This has been followed by MTF and Run Length (RLE) to eliminate the repetition. Entropy coding is finally applied. The core challenging task of our model is that the precision of Gaussian model and the selection of its appropriate parameters is to produce accurate approximated smart meter readings.

4.3.1 Gaussian Approximation

In mathematics, Gaussian approximation is a well-known continuous probability distribution. It's significance comes due to its ability to represent real-value random fluctuated signals whose distribution are not known [111]. This inspired us to use Gaussian distribution functions in our compression algorithm to approximate the IoTs smart device readings. Simple Gaussian function is depicted in Fig 4.2 and shown in Eq. 4.1

$$f(x) = ae^{-\left(\frac{(x-b)}{2c}\right)^2} \tag{4.1}$$

where x is a discrete variable and the Gaussian function parameters are represented

by $a = \frac{1}{\sigma\sqrt{2\pi}}$, which is the amplitude of the highest peak value, $b = \mu$ is the centroid of the model and $c = \sigma$ is the peak's width.

To depict a multi-peak signal, the identical Gaussian equation can be reformulated as shown in Eq. 4.2.

$$f(x) = \sum_{i=1}^{n} a_i e^{\left[-\left(\frac{(x-b_i)}{2c_i}\right)^2\right]} \tag{4.2}$$

where n reflects the number of Gaussian functions (i.e. the required peaks to fit). $f(x)$ represents the IoTs readings. The crucial part is that hthe values of the parameters are selected? Therefore, a suitable optimisation theory called trust region algorithm is used to calculate these parameters in our model, due to its robust behaviour in bad conditions while maintaining very strong convergence properties. This resulted in minimising the difference between the approximated and the actual values of the IoTs smart meter readings. The variation between the original readings (y) and the Gaussian approximated signal ($\widehat{y}$) is carefully measured by using a non-linear least square equation as in Eq. 4.3

$$f = \sum_{i=1}^{n} (y_i - \widehat{y}_i)^2 \tag{4.3}$$

where f demonstrates the sum of the squared-residuals that should be reduced. Therefore, the Trust region algorithm [111] is used to find the optimum Gaussian parameters x that is utilised to minimise the objective function represented in f. This is done by approximating the original function f utilising a quadratic equation $m_\delta(p)$ to find out the optimum step size p by which the parameter values x should be scaled up or down. Additionally, the step value at iteration δ can be specified by resolving this quadratic equation [123] as shown in Eq. 4.4

$$m_\delta(p) = f_\delta + p^T g_\delta + \frac{1}{2} p^T B_\delta p \tag{4.4}$$

where the value of the objective function at iteration δ using the current parameters values x_δ can be reflected as $f_\delta = f(x_\delta)$, B_δ is the Hessian of f, and g_δ is the gradient of both the parameter values of x_δ at iteration δ and f. The step size value p (i.e. the

solution of this problem) is restricted to a particular region Δ_k called trust region as depicted in Eq. 4.5

$$\|p\| \leq \Delta_\delta \tag{4.5}$$

The trust region can be scaled up or down based on the original objective function and its accuracy in the approximated quadratic function. For this target, a reduction factor r_k is introduced to examine the performance of the quadratic approximation as formulated in Eq. 4.6

$$r_\delta = \frac{f(x_\delta) - f(x_\delta + p_\delta)}{m_\delta(0) - m_\delta(p_\delta)} \tag{4.6}$$

The trust region Δ_δ is changed based on r_δ as follows:

$$\begin{cases} \Delta_\delta \uparrow & \text{if } r_\delta > \frac{3}{4} \\ \Delta_\delta \downarrow & \text{if } r_\delta < \frac{1}{4} \\ \Delta_\delta \rightarrow & \text{else} \end{cases} \tag{4.7}$$

The parameters are finally updated utilising step p_δ and the operation is repeated until the stop condition is reached.

For clarity, Fig 4.2 shows three examples of Gaussian approximation optimisation. (a) Plot for more than 1500 smart meter power readings and their Gaussian approximations. (b) Plot of the resultant residuals after calculating the margin (i.e. highlighted in blue), which clearly shows the more the accuracy of Gaussian approximations is the less left margin values are. In this process alone, more than 50% of the values have been zeroed and others have been significantly reduced. The advantage of that is achieving higher compression ratio while keeping just a few parameters to reproduce Gaussian approximation to recover the readings.

4.3.2 Margin Calculation

Gaussian distribution optimisations have been thoroughly examined to achieve the best possible generic distribution. To avoid unbiased results, the final optimum

collection of parameters has been used in all our experiments. Next, the difference between smart meter readings and their Gaussian approximation is calculated (see Eq. 4.8). The significance of that is the compression will only be on the calculated margin space rather than the entire set of readings. Also, to avoid large differences in any consecutive margin calculated values, the first derivative is applied as shown in Eq. 4.9.

$$\varphi = \int_{i=1}^{n} [y_i - \widehat{y_i}] \tag{4.8}$$

$$D = [\varphi(2) - \varphi(1)\varphi(3) - \varphi(2)...\varphi(\Omega) - \varphi(\Omega - 1)] \tag{4.9}$$

where φ is the calculated margin, Ω is the length of φ and D is the resultant derivative vector.

4.3.3 Burrow-Wheeler Transform

After calculating the first derivative on the margin space, we observed there were less than 10% unique values (see Fig. 4.3). However, they are scattered which minimise their compression effectiveness. Therefore, BWT is used to rearrange the values to be in a consecutive long sequence of identical symbols. BWT is originally introduced by M. Burrows and D. Wheeler [124] to transform text into a different format to increase its compressibility by applying techniques, such as the MTF technique. The significance of this algorithm is that it is reversible with zero additional information; the general idea is to rotate the data (i.e. 1 to n blocks) . Let's Λ be the block of symbols in textual or numerical (i.e. numerical in our algorithm) form to be compressed).

$$\Lambda = \Lambda_1, \Lambda_2, ..., \Lambda_n \tag{4.10}$$

The BWT begins by left rotation of the vector Λ in an iterative manner. This generates a 2D matrix called ω, as in Eq. 4.11

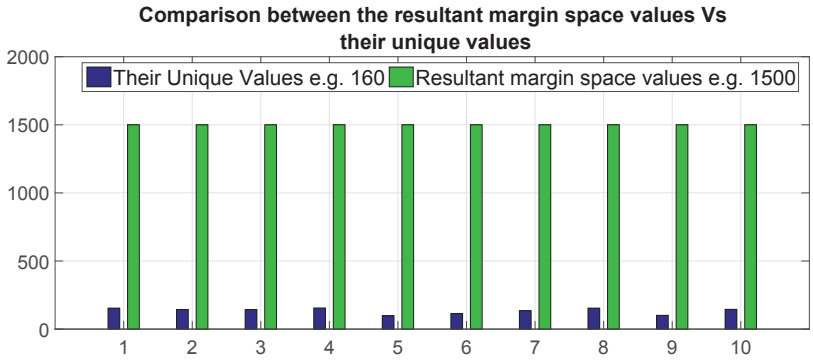

Figure 4.3: Comparison between the calculated margin and their unique values.

$$\omega = \begin{pmatrix} \Lambda_1 & \Lambda_2 & \Lambda_3 & \cdots & \Lambda_n \\ \Lambda_2 & \Lambda_3 & \cdots & \Lambda_n & \Lambda_1 \\ \Lambda_3 & \cdots & \Lambda_n & \Lambda_1 & \Lambda_2 \\ \vdots & \vdots & & \ddots & \vdots \\ \Lambda_n & \Lambda_1 & \Lambda_2 & \cdots & \Lambda_n - 1 \end{pmatrix} \tag{4.11}$$

It is clear from Eq. 4.11 that the rows in ω represent various rotations of Λ. A new version of the initial ω called $\widetilde{\omega}$ is generated by sorting its rows in ascending way. The last column C of $\widetilde{\omega}$ is retrieved and accompanies the index I that points to the original block Λ.

For a better understanding, let's assume that the resultant values after the calculation of the margin and first derivative as shown in Table 4.2. For simplicity, they have been converted to characters (i.e. represent Λ). Λ is rotated n times (i.e. number of elements) to generate ω as shown in Eq. 4.12.

Table 4.2: Conversion from numerical to character values

Λ	0.0999	0.2000	0.1000	0.2000	0.1000	0.2000
Ch	b	a	n	a	n	a

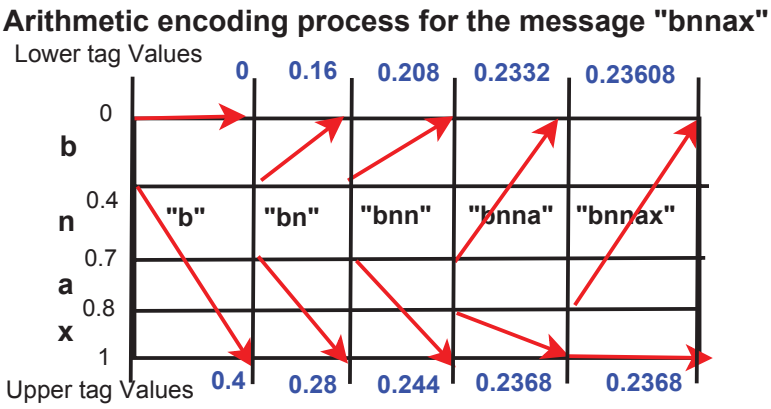

Figure 4.4: Graphical representation for encoding a message "bnnax".

$$\omega = \begin{Bmatrix} \wedge & b & a & n & a & n & a & | \\ | & \wedge & b & a & n & a & n & a \\ a & | & \wedge & b & a & n & a & n \\ n & a & | & \wedge & b & a & n & a \\ a & n & a & | & \wedge & b & a & n \\ n & a & n & a & | & \wedge & b & a \\ a & n & a & n & a & | & \wedge & b \\ b & a & n & a & n & a & | & \wedge \end{Bmatrix} \tag{4.12}$$

$|$ represents the end of data. ω then is sorted to generate $\widetilde{\omega}$.

L (i.e. last column) and I (i.e. the index) will be the final output from this stage.

The decoder will rely on L and I to retrieve the original form. This is done by inserting L as the last column in a temporary 2D matrix of size $n \times n$ (i.e. the number of elements). This column is sorted to figure out the first column. Then the first and last columns (of each row) together give you all the pairs of successive characters. Finally, the resultant matrix is identical to $\widetilde{\omega}$. Therefore, the original form of data is easily accessible using the parameter I.

4.3.4 Move-To-Front

Although the output of BWT perfectly clusters similar symbols in long runs, in the case of IoTs smart meter readings these symbols vary from very small (e.g. 10) to large values (e.g. 5000). Therefore, to expand the compression effectiveness of any entropy encoder such as Arithmetic Coding, MTF transform is applied. MTF is a lightweight algorithm proposed by Ryabko [125], used to increase the probability of small numbers near zero while decreasing the large numbers in a data list. The general idea is that each symbol in the data is replaced by its index in the list of currently used symbols. Therefore, long sequences of similar symbols are replaced by many zeros, while a rear symbol that has not been used for a long time will be replaced by a larger number.

Let Υ be all the distinct symbols in the list L obtained from BWT stage which is shared between the encoder and decoder The MTF algorithm can be summarised in three steps. (1) Υ is initialised using L. (2) Each L_x in the list L is encoded as its preceding number of symbols in Υ, which will then be moved to the front of the distinct list Υ. (3) The final output is constructed in a list ∂ by combining the codes of step 2.

The decoding process is the inverse of these steps.

For better clarification (see Table 4.3), let's assume $L = [b, n, n, a, a, a]$ which is the output of BWT and its distinct symbols $\Upsilon = [a, b, n]$. The first symbol L_0 is b, and its preceding symbol index in Υ is 1. Consequently, the encoder will output 1 in ∂ and move b to the front of $\Upsilon = [b, a, n]$. The second symbol L_1 is n which is preceded by two symbols, and so the encoder output will be 2 and updated $\Upsilon = [n, b, a]$. This will continue until L_{last} and the final output of this stage is $\partial = [1, 2, 0, 2, 0, 0, 0, 0]$. Note that; two zeros have been added at the end of L to emphasize the significance of MTF.

Table 4.3: MTF of $L = [b, n, n, a, a, a]$ and $\Upsilon = [a, b, n]$

L_x	1	2	0	2	0	0	0	0
Υ	abn	ban	nba	nba	anb	anb	anb	anb
∂	b	n	n	a	a	a	a	a

4.3.5 Run Length

The output from MTF contains many identical consecutive symbols; therefore, a simple technique called RLE [126] is applied before the entropy codes. The general idea behind this approach to data compression is this: let's assume a data item d occurs n consecutive times in the list of values, the n occurrences of this item will be replaced with the single pair nd. The n consecutive occurrences of the item are called a run length of n. For example, the consecutive zeros in $\partial = [1, 2, 0, 2, 0, 0, 0, 0]$ will be $\partial = [1, 2, 0, 2, 0\#4]$.

4.3.6 Arithmetic Coding

An entropy encoding technique called Arithmetic Coding AC is applied in our algorithm to achieve the most possible optimum compression ratio. AC is a statistical variable length coding by which frequently used numbers will be stored with fewer bits and not-so-frequently occurring symbols will be represented with more bits [126]. It is superior in most respects to the better-known entropy coders, such as the Huffman method. This is because that rather than segregating the input into component symbols and replacing each with a code, it encodes the entire message into a single number, a fraction n where $(0.0 \leq n < 1.0)$.

The main idea is initiating from a certain interval, reading the input list by symbols and using the probability of each number to narrow the interval. In other words, AC begins by defining the current interval as (0,1). It then repeats the following two steps for each symbol S_i in the data list: (a) to divide the present interval into subintervals whose sizes are proportional to the symbols probabilities, and (b) choose the subinterval for S_i which will be defined as the new present interval. Finally, when the entire data list is processed in this way, the output should be any number that

Table 4.4: Frequency Distribution in message M

Symbol	Probability	Accumulative Freq
b	0.4	0.4
n	0.3	0.7
a	0.1	0.8
x	0.2	1

uniquely distinguishes the current interval (i.e. any number in the present interval).

The present interval gets smaller for each symbol processed. The final output is a single number, called 'tag value', and does not consist of codes for the individual symbols.

To illustrate AC code construction, let's consider encoding a portion of message $\tilde{M} = (b, n, n, a, x)$. The full frequency distribution of that message is shown in Table 4.4. Default probability limit is between $(0, 1)$. First, if b occurs, then the tag value has to be between $(0, 0.4)$. Next, n is detected, so the current interval $(0, 0.4)$ should be divided into subintervals by using lower and upper limit equations as shown in 4.13 and 4.14 [126].

$$l_n = l_{n-1} + (u_{n-1} - l_{n-1}) \times F_x(x_{n-1}) \tag{4.13}$$

$$u_n = l_{n-1} + (u_{n-1} - l_{n-1}) \times F_x(x) \tag{4.14}$$

where l_n and u_n are the lower and upper limit of the nth symbol, F_x represents its accumulative frequency.

After substituting in Eq 4.13 and 4.14, the tag value of the sequence b, n is $(0.16, 0.28)$. This must be repeated for the entire message accumulatively. All tag values have been graphically summarised in fig. 4.4. The final compressed value is the average of the lower and the upper tag values $\frac{0.2360+0.2368}{2} = 0.2364$ which will be converted into binary.

On the decoder side, this tag value will be received and the probabilities of the message must be known. Then, the steps are identical but an inverse where the letters or numbers should be found by their accumulative probability.

4.3.7 Decompression - Gaussian-based Model

The decompression is identical to the stated compression steps, but in an inverse way. The algorithm begins by Arithmetic decoding. Next, the run length values are segregated followed by MTF decoding. Burrow-Wheeler inverse transform then is applied which will generate the calculated margin space values. After that, the first derivative inverse is conducted to retrieve the actual margin space values. The stored Gaussian parameters are then used to calculate the approximated waveform readings. The summation of the approximated readings and the margin space values is finally calculated to reconstruct the exact lossless smart meter readings.

Fig. 4.7 in Section 4.6 shows a few examples of the original smart meter readings and the decompressed version which obviously proves that the readings are fully recoverable in our approach with zero loss.

4.4 Model 2 (MD2): N-Split Based Model

The principle behind this proposed technique is for minimising the randomness of IoTs smart meter readings into a smaller finite field to boost the possibility of repetition and avoid the accumulated rounding errors due to floating operations. This was successfully achieved by splitting the collected readings into different groups. The more stable group (the integer side) is highly compressed using the first derivative followed by BWT. This was followed by MTF and RLE to eliminate the repetition. Entropy coding is finally applied. The more random group (the floating part) is mapped to a smaller space using a mathematical model before applying BWT, MTF and RLE followed by entropy coding.

Although it appears there is some stability in the collected readings, most of the general dictionary and statistical compressors are ineffective due to the precision problem. Most current readings formats are four decimal precision, which means every single reading has ten thousand images from 12.0000 to 12.9999. Therefore, at this stage every reading is split into N - two groups G_A and G_B. G_A only contains the more stable part (i.e. integer) of the readings and G_B includes the noise (i.e. four

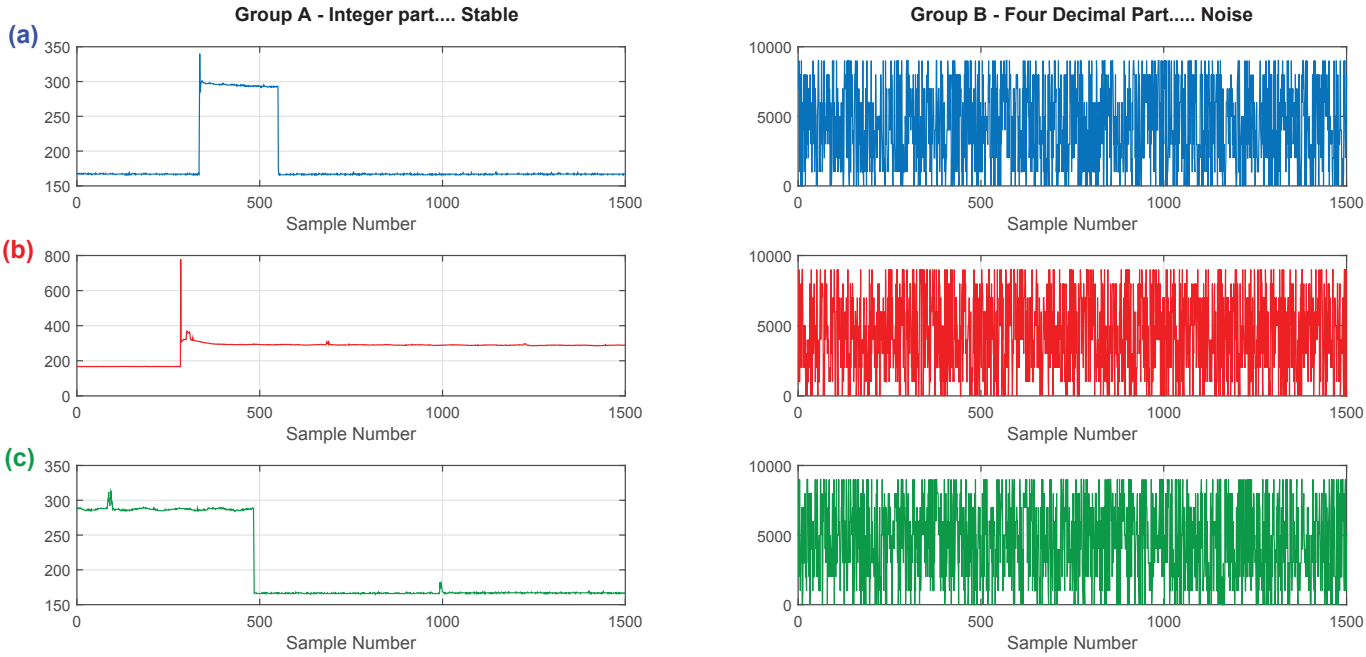

Figure 4.5: Three examples of the split output. Group A represents the more stable part, and group B highlights the noise part.

decimal precision) part. Every group is treated differently due to its characteristics.

4.4.1 Stable Group Reduction

To exploit the stability feature in G_A, the first derivative is calculated as shown in Eq 4.15.

$$D = [\varphi(2) - \varphi(1)\varphi(3) - \varphi(2)...\varphi(\Omega) - \varphi(\Omega - 1)] \tag{4.15}$$

where φ is the ith reading in G_A, Ω is the length of φ and D is the resultant derivative vector.

We observed that there is lots of redundancy, but in scattered format. Therefore, BWT (Read 4.3.3) is used to rearrange the values to be in the consecutive long sequence of identical symbols. BWT simply rotates a suffix array SA values (G_A readings) and its original is T. BWT can be defined as Eq. 4.16.

$$BWT[i] = \begin{cases} T[SA[i]-1] & if \iff SA[i] > 1 \\ \$ & Otherwise \end{cases} \tag{4.16}$$

To increase the effectiveness of any entropy coding, MTF (see 4.3.4) is employed. The basic idea is that each symbol in the resultant series from BWT - these symbols are replaced by its index in the list of currently used symbols. Therefore, long sequences of similar symbols by as many zeros.

The output from MTF contains many identical consecutive symbols; therefore, a simple technique called 'run length' (see 4.3.5) is applied before the entropy codes. For instance, the consecutive zeros in $\partial = [1, 2, 0, 2, 0, 0, 0, 0]$ will be $\partial = [1, 2, 0, 2, 0\#4]$.

AC (see 4.3.6) is finally applied on the stable group G_A to achieve the most possible optimum compression ratio.

4.4.2 Noise Group Reduction

The group G_B contains the decimal parts of the readings and we called it 'noise'. This is because they have no stable pattern. However, the merits of pulling them aside are explained thusly: (i) reducing the space from ten thousand images for every single reading (e.g. 11.0000 - 11.9999) to ten thousand images for all readings together, (ii) that operations of floating points are more expensive than the integer values, because of the deficiency and limitations of handling floating points in the majority of embedded systems. Therefore, the split values in G_B (i.e. four decimal .9999) are converted to an integer counterpart in our algorithm.

To eliminate unnecessary digits while not losing any details, the resultant values are mapped into smaller digits space MG_B. For example, 0001 value into 1, 0002 into 2 and so on.

The resultant mapped space MG_B is then reshuffled employing BWT (see 4.3.3) and MTF (see 4.3.4) to obtain consecutive sequences to be exploited later.

Finally, RLE (see 4.3.5) is used to eliminate redundancy before applying the entropy coding AC (See 4.3.6).

4.4.3 Code Chaining

The output from both groups G_A and G_B is in a bit format. Thus, the two streams are concatenated in this step. One small, two bytes, delimiter is added prior to the concatenated stream to identify the size of the first part S_1, which in tern tells the second part $T - S_1$.

4.4.4 Decompression- N-Split Based Model

The decompression is identical to the stated compression steps but in an inverse way. The algorithm begins by code splitting to identify the bit streams that belong to each group. (i) The more stable group (the integer side) is decompressed by entropy coding, RLE segregation, MTF, BWT decoding and first derivative inverse that are conducted to retrieve the actual G_A values. (ii) The more random group (the floating part) is decompressed by AC, RLE segregation, MTF, BWT decoding and mapping inverse is performed to recover the original G_B set. Finally, N-Split stage is inverted by concatenating the two groups to rebuild the exact original IoTs streams.

4.5 Compression Performance Metrics

This section discusses the theoretical and empirical matrices used to evaluate our proposed algorithm.

4.5.1 Theoretical Entropy

In information theory, the term entropy of a signal represents the minimum bitrate, meaning assuming the best compression, that is required to transmit this signal [127]. Therefore, to prove the effectiveness of preprocessing the data in our algorithm, the theoretical entropy is calculated for every data list of IoTs smart meter readings before starting our algorithm. Then, a quantitative comparison of measurements will be conducted between the theoretical entropy and the achieved compression ratio.

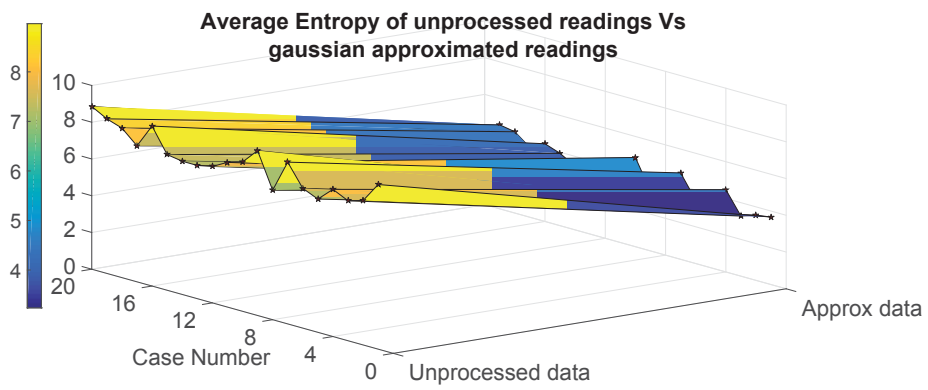

Figure 4.6: Comparison of entropy before/after Gaussian approximation.

Let's assume IoTs readings list consists of the data points $d[1], d[2], ..., d[N]$ with the maximum likelihood that entropy (in bits) is measured as

$$H(d) \triangleq -\sum_{v \in U(d)} \widehat{p}(v) \log_2(\widehat{p}(v)) \tag{4.17}$$

$$\widehat{p}(v) \triangleq \frac{1}{N} \sum_{n=1}^{N} \delta_v(d[n]) \tag{4.18}$$

$$\delta_v(d[n]) \triangleq \begin{cases} 1 & \text{if } d[n] = v \\ 0 & else \end{cases} \tag{4.19}$$

where $U(d)$ is the range of d and $\widehat{p}(v)$ is the empirical probability of $v \in U$.

The highest entropy is the worst-case occurs when each value in U appears at the identical frequency $1/|U|$, where $|U|$ represents the elements in original U (see Eq. 4.20).

$$H_{max} = -\sum_{v \in U} \frac{1}{|U|} \log_2(\frac{1}{|U|}) = \log_2 |U|. \tag{4.20}$$

Contrarily, the lowest entropy (or the best-case) occurs when all d values are similar, which leads to $H_{min} = -1 \log_2(1) = 0$.

Fig. 4.6 emphasises the average improvement in the entropy before and after the major step which is applying the Gaussian approximation and utilising only the margin space. The entropy has been almost halved.

Table 4.5: Compression ratio

Meter	Huf [5]	Del-Huf [5]	Norm+AC [6]	AC [9]	Lemp-Ziv [7]	P-LZMA [113]	Bzip2 [9]	Inv-Golomb [8]	Del+Bzip2 [9]	Our approach
1	1.72	1.97	1.78	2.43	2.47	2.44	2.60	3.07	3.13	3.90
2	1.08	1.71	1.93	2.41	2.21	2.54	2.53	2.81	3.11	3.91
3	1.14	1.63	1.88	2.39	2.13	2.53	2.58	2.73	3.01	3.90
4	1.29	1.64	1.87	2.37	2.14	2.51	2.54	2.74	2.97	3.89
5	1.78	2.01	1.91	2.36	2.01	2.26	2.47	2.61	2.90	3.87
6	1.81	1.78	1.86	2.36	2.28	2.48	2.52	2.88	2.83	3.85
7	1.34	1.70	1.96	2.36	2.20	2.50	2.56	2.80	2.88	3.83
8	0.91	1.62	1.99	2.37	2.12	2.54	2.54	2.72	2.84	3.82
9	1.18	1.58	1.86	2.38	2.08	2.53	2.61	2.68	2.77	3.80
10	1.03	1.80	1.91	2.41	2.30	2.36	2.65	2.90	2.82	3.78
11	1.32	1.67	1.99	2.42	2.17	2.33	2.66	2.77	2.86	3.76
12	1.52	1.74	2.05	2.44	2.24	2.22	2.67	2.84	2.84	3.73
13	1.03	1.84	2.01	2.46	2.34	2.53	2.67	2.94	2.91	3.70
14	1.47	1.92	2.04	2.46	2.42	2.48	2.69	3.02	2.95	3.65
15	0.95	2.09	2.02	2.44	2.09	2.22	2.69	2.69	2.96	3.60
16	0.91	1.74	2.12	2.44	2.24	2.50	2.68	2.84	2.97	3.54
17	1.09	1.80	2.06	2.43	2.30	2.61	2.68	2.90	2.97	3.97
18	0.79	2.06	2.10	2.42	2.06	2.55	2.67	2.66	2.99	3.91
19	0.68	1.74	2.16	2.42	2.24	2.64	2.63	2.84	2.99	3.85
20	0.67	1.81	2.08	2.44	2.31	2.30	2.63	2.91	2.98	3.80
Average	1.19	1.79	1.98	2.41	2.22	2.45	2.61	2.82	2.94	3.80

4.5.2 Empirical Ratio

The Compression Ratio CRi is the main benchmark to measure any proposed compression algorithm performance. Let's denote the original IoTs readings block O (i.e. its unit in byte or bit) and the resultant compressed readings C. Therefore, the empirical CRi in results section is calculated as shown in Eq. 4.21.

$$CRi = \frac{O}{C} \tag{4.21}$$

A well-known and leading power quality storage format for electric power system waveforms that is used in most of the IoTs smart devices called Power Quality Data Interchange Format (PQDIF). This is defined by the IEEE1159 working group [128] and was used to accurately measure the original and compressed size of the readings

in bits. Every reading is represented as 16 bits. The typical block size suggested is around 1500 readings.

4.6 Experiments and Results

4.6.1 Datasets

Various datasets of IoTs smart meter readings have been utilised in our experiments that were collected and published by the Laboratory for Advanced System Software as a part of a project named 'Smart*' [129, 130]. The dataset includes continuous readings (every minute) from three homes for three months. The readings' types can be classified into (1) power usage - watts consumption and heat index, and (2) environmental characteristics - inside/outside temperature, inside/outside humidity and wind chill. The datasets also provide periodical electricity power consumption (every minute) from around 400 anonymous homes for $(3 \times 30 \times 24)$ hours. According to the definition of spatial and temporal aggregations, these readings are temporal, because they are gathered separately from every single house after equipping it with a smart meter that collects its readings periodically.

Our main performed experiments can be classified into two main parts. (1) The compression performed by aggregators that receives an overwhelming amount of collected readings from various entities (e.g. homes). (2) Decompression which is performed at operation centres or on the cloud level.

4.6.2 MD1: Gaussian-based Model

To avoid biased results, all records of meter readings in the aforementioned datasets have been used. The shown results in both Table 4.5 and Fig. 4.8 are from continuous blocks from 15/Apr/2012 to 1/Jul/2012. Similarly, the existing lossless compression models in this domain have been implemented under the same conditions to precisely provide a clear comparison. For brevity, the results have been summarised in Table

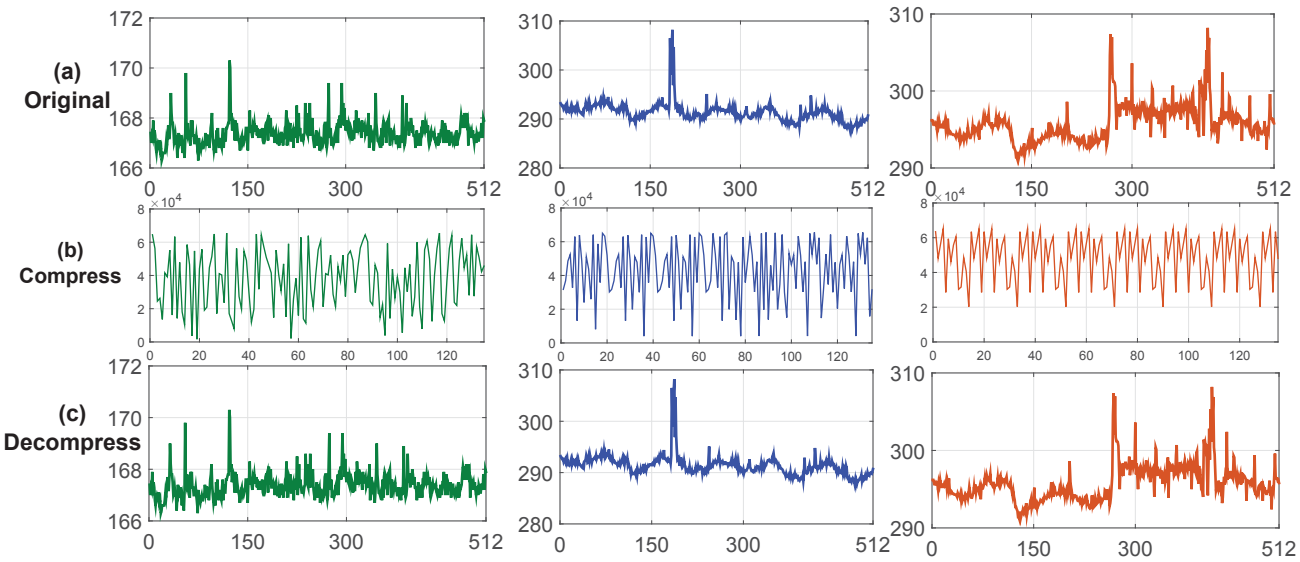

Figure 4.7: Three examples of watts consumptions' readings of three homes: (a) direct plot of original readings, (b) plot of the obtained compressed streams gathered as 16-bit per value, and (C) plot of the readings after decompression.

4.5. Every column represents results from different model as follows. (1) The second and third columns show CRi of using techniques in [5] which mainly relies on Huffman and delta-Huffman. (2) The fourth column presents results of employing the model in [6] based on normalisation, differential coding and other steps before entropy coding. (3) The fifth column depicts the results of utilising models in [9] based on AC. (4) The results of the sixth column are based on the model in [7] that uses Lempel-Ziv. (5) The seventh column presents results obtained by utilising the technique in [113], which is based on linear prediction model and LZMA. (6) The eighth and the ninth columns highlight results using the technique in [9] which relies on bzip2 and delta bzip2 (i.e. BWT,MTF and AC). (7) The ninth column shows the results using the invertible transformation pre-processing followed by Golomb-Rice encoding [8]. (8) The last column shows results using our Gaussian-based algorithm.

Fig 4.8 shows a larger number of the obtained CRi from our algorithm compared to the ones above. Also, Table 4.6 presents the exact improvement factors compared with the best existing model. In addition, Fig. 4.9 shows a comparison of the exact required time between our model and the highest existing compressor working on the

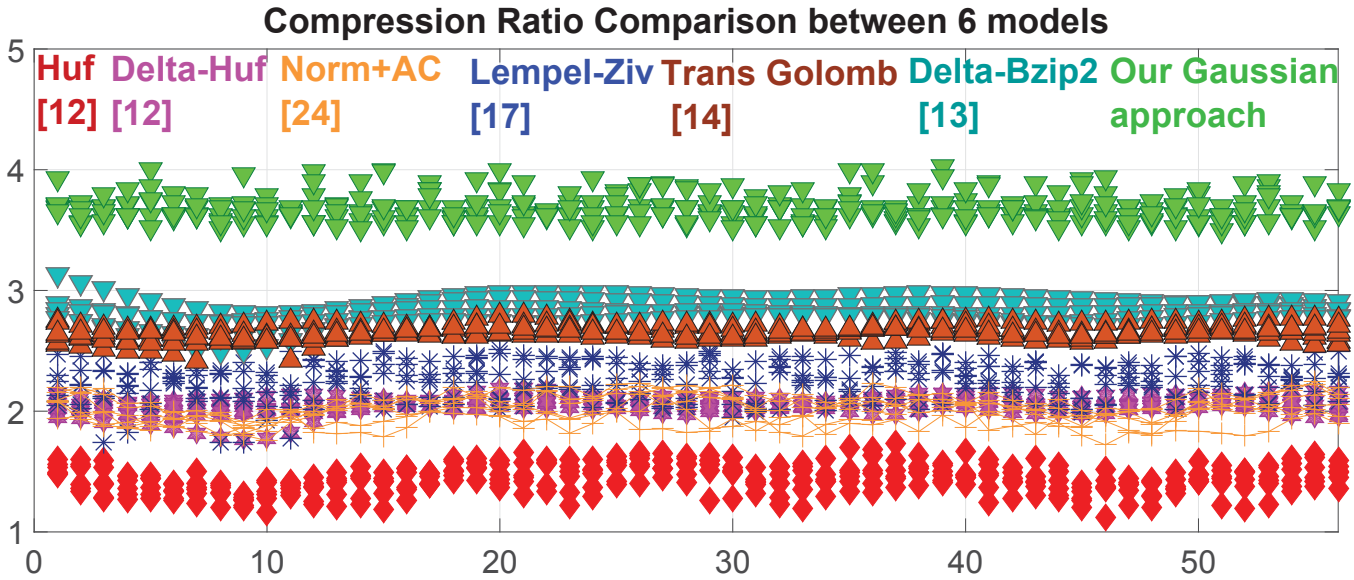

Figure 4.8: Compression ratio of 7 models: (1) Huffman [5] , (2) delta-Huffman [5] , (3) Norm-AC [6], (4) Lempel-Ziv [7], (5) Invert-Trans Golomb [8], (6) delta-Bzip2 [9], and (7) our Gaussian based approach.

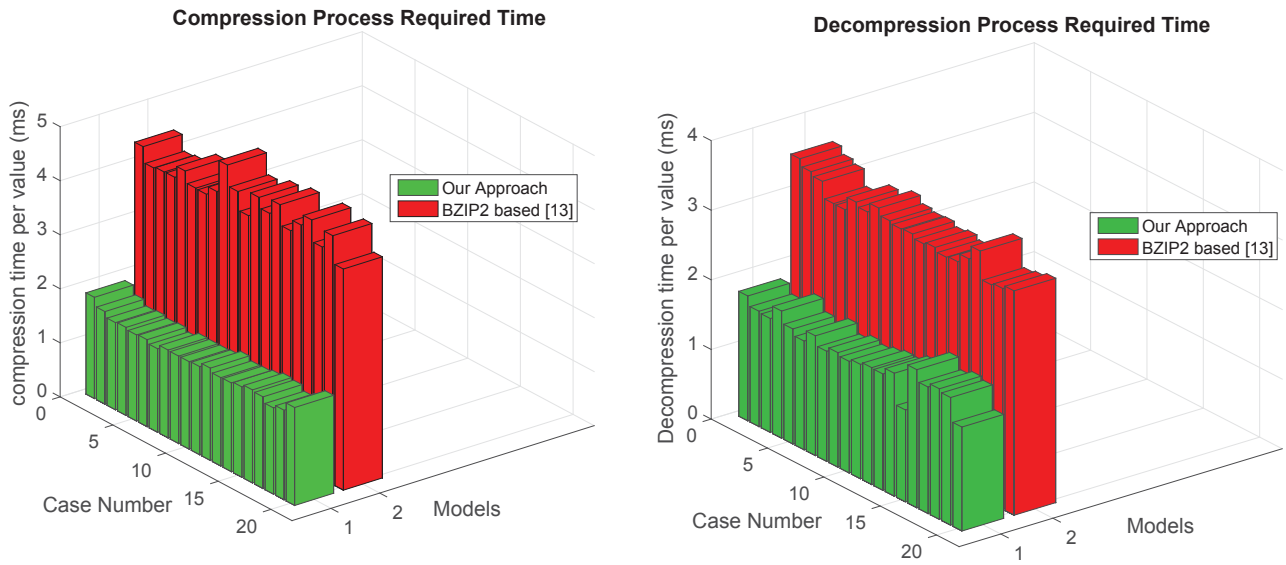

Figure 4.9: Average time required in millisecond to compress/decompress per value of readings from 20 different meter readings using both our approach and BZIP2 based model [13].

same datasets that have four decimal precision. Finally, Fig. 4.7 shows an example of a plot of six original CRi IoTs readings before the compression and after the recovery

Table 4.6: our Gaussian approach against the best existing model

Factor	Delta+Bzip2 [9]	Our approach (MD1: Gaussian-based)
Best ratio	3.13	4.02
Worst ratio	2.49	3.50
Average	2.80	3.73
Standard deviation	0.02	0.11
Improvement percent		33%

(i.e. decompression).

Discussion The experiments obviously emphasise the effectiveness of our Gaussian approximation based lossless compression of IoTs smart meter readings compared to other existing algorithms. This is due to the advantage of excluding many data points and compressing only the margin space. This has been proved by comparing the Shannon entropy before and after our approach (see fig. 4.6) and experimentally as shown in Fig. 4.8.

It is also obvious from Fig. 4.9 that our model requires less time (i.e. ≤ 2 ms vs. 4 ms) to accomplish both compression and decompression than the model that achieved the highest compression among existing techniques. This is because our model uses less number of stages (i.e. only six) and they use more than nine stages between them building the Huffman dictionary which is not that efficient in detailed numerical datasets that have four decimal precision.

4.6.3 MD2: N-Split Model

To avoid biased results, all records of meter readings used in our Gaussian-based model has been reused. The published results in both Table 4.5 and Fig. 4.8 are from continuous blocks (i.e. from 15/Apr/2012 to 1/Jul/2012). For brevity in this chapter, the results have been summarised in Table 4.7. Every column represents results from a different model as follows. (1) The second column portrays the CRi of using our Gaussian-based model, and that has been proven to have the best compression ratio among the existing size reduction algorithms in this field. (2) The last column presents results using our N-Split based algorithm.

Table 4.7: our N-Split approach against the Gaussian model

Mater No	(MD1: Gaussian Based)	(MD2: N-Split Based)
Mater 1	3.90	4.67
Mater 2	3.91	4.61
Mater 3	3.90	4.56
Mater 4	3.89	4.52
Mater 5	3.87	4.48
Mater 6	3.85	4.45
Mater 7	3.83	4.43
Mater 8	3.82	4.42
Mater 9	3.80	4.41
Mater 10	3.78	4.42
Mater 11	3.76	4.43
Mater 12	3.73	4.45
Mater 13	3.70	4.27
Mater 14	3.65	4.39
Mater 15	3.60	4.22
Mater 16	3.54	4.14
Mater 17	3.97	4.56
Mater 18	3.91	4.57
Mater 19	3.85	4.58
Mater 20	3.80	4.58
Average	3.80	4.48

Table 4.8: our N-Split approach against the Gaussian model

Factor	(MD1: Gaussian-based)	(MD2: N-Split-based)
Best ratio	4.02	4.73
Worst ratio	3.50	4.09
Average	3.80	4.48
Standard deviation	0.11	0.05
Improvement percent		17%

Fig 4.10 shows a larger number of the obtained CRi from our N-Split based algorithm compared to the Gaussian-based model. Also, Table 4.6 presents the exact improvement factors compared with the best existing model. Additionally, Fig. 4.9 shows a comparison of the exact required time between our N-Split model and the highest existing compressor (the Gaussian-based) as working on the same datasets that have four decimal precision.

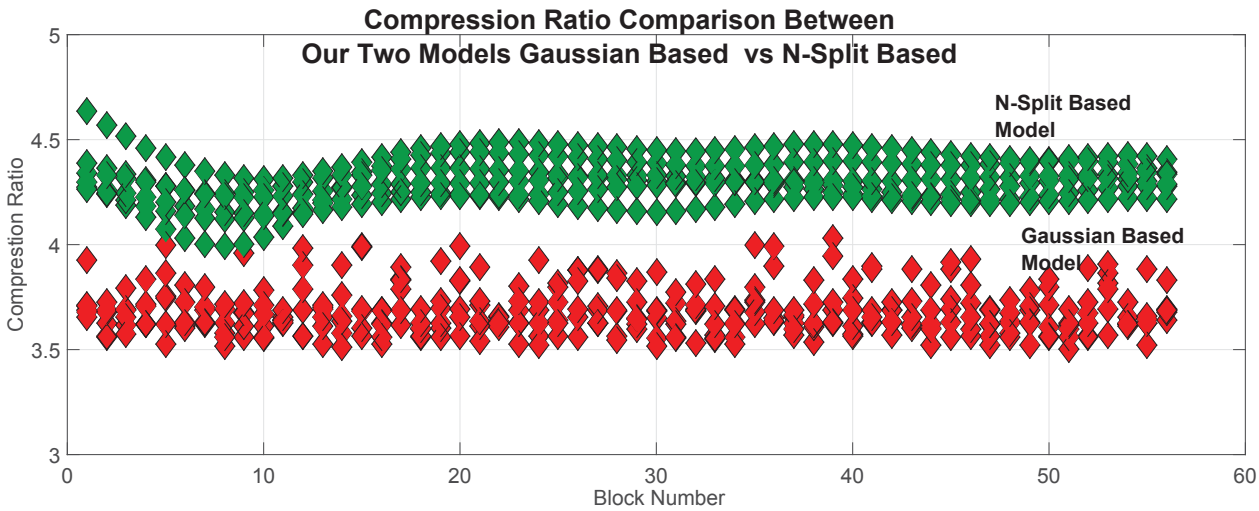

Figure 4.10: Compression ratio of two models our Gaussian based approach (MD1) vs N-Split based model (MD2) .

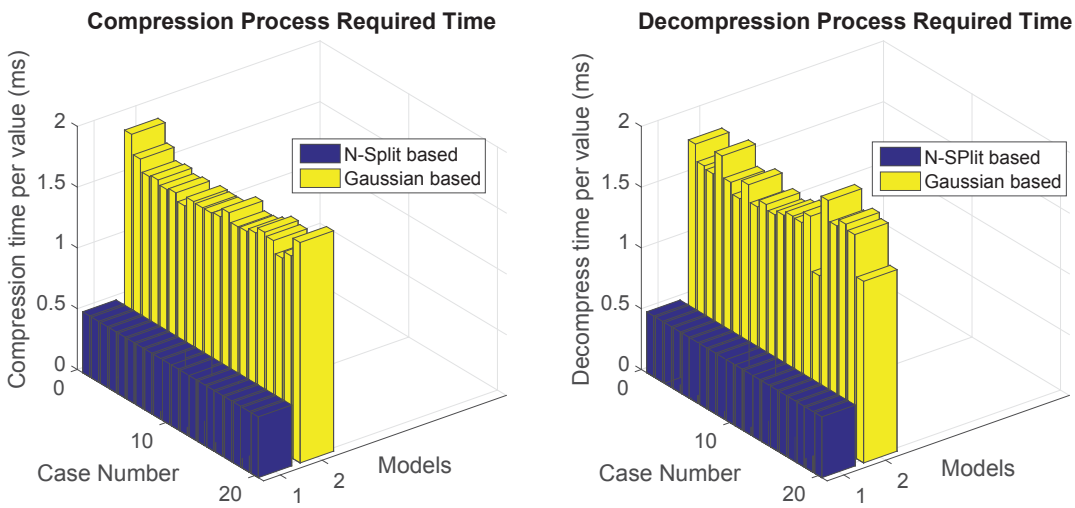

Figure 4.11: Average time required in millisecond to compress/decompress per value of readings from 20 different meter readings using both our Gaussian based and N-Split based models.

4.6.4 Comparison of our MD1 vs. MD2

It is very clear from the results that the Gaussian based model (MD1) achieved better results of 3.80:1 than most of the best existing work in [9] (i.e. 2.80:1). It also requires less time to compress and decompress (i.e. $\leq$ 2 ms vs. 4 ms). In addition, MD1 is a

generic model, because it relies on the Gaussian approximation function that learns from the IoTs signal stream itself. Therefore, regardless of the signal form, MD1 can achieve a stable CRi. On the contrary, our latest approach N-Split based model (MD2) accomplished better CRi (i.e. 4.48) than the Gaussian-based algorithm. The required processing time is also less (i.e. $\leq$ 0.6 ms vs. 2 ms). However, MD2 is dataset driven model. In other words, there is no guarantee that we can achieve the highest CRi with varied datasets. MD2 has been built based on the obtained datasets observations.

4.7 Chapter Summary

We introduced two novel compression algorithms for lossless IoTs smart device readings that reduce the volume of protected readings at intermediate hops without revealing the hidden secrets. The target of the first generic (i.e. Gaussian-based) model represents IoTs streams in several parameters despite the crudity of the signal. This is successfully accomplished using the Gaussian approximation. The difference between the approximated and the actual waveform is calculated in other words, the compression will only be for margin space, rather than the entire stream of waveform readings. The margin space values are finally encoded. The second target uses the N-Split model to reduce the randomness in the IoTs streams into a smaller finite field to enhance repetition and avoid the floating operations round errors issues. After a thorough evaluation under the same conditions, both of our techniques were superior to existing models mathematically, meaning the entropy was halved; and empirically, we achieved a ratio of 3.8:1 to 4.5:1. However, we discovered that the two new proposed models in this chapter varied - the Gaussian-based MD1 is a generic model, whereas the N-Split MD2 is a dataset driven model.

Chapter 5

Cloud-based Protected IoTs Size Re-reduction

This chapter answers the fourth research question discussed in Section 1.2. Although our privacy preserving size reduction models proposed in Chapter 4 reduce the size of the transmitted Internet of Things (IoTs) streams significantly, the unexpected volume of these compressed streams required endless storage and data management space which poses a unique challenge. Therefore, this chapter investigates the question: can the incoming protected compressed IoTs readings be blindly re-compressed without neither revealing their privacy nor decompressing them? Section 5.1.2 introduces the main contributions of this chapter which is pre-processing the compressed streams by finding their similarities. Section 5.2 briefly highlights the key-related works and why they are ineffective by applying them directly due to the high dissimilarities among the streams. Section 5.3 explains in detail our novel model including the design, similarity measurement, group interleaving, differential calculation, rotation, dynamic run length and entropy coding. The evaluation of various characteristics of our model (including a theoretical entropy and empirical ratio improvement, performance and cohesion analysis among the subgroups) are introduced in Section 5.5. Section 5.7 presents detailed experimental examinations of the model, including test bed scenarios and a wide range of protected compressed IoTs readings, its possible reduction (i.e. re-compress) ratio, the actual IoTs protected compressed high streams

after decompression, and a comparison with available techniques. Section 4.7 finally summarises this chapter.

5.1 Introduction

Due to the lack of outage management, automation, poor real-time analysis and deficiencies of the classic power grid of the past century, a new infrastructure called 'smart grid' is recently being investigated around the world. It will automatically collect periodic waveform readings using Phasor Measurement Units (PMUs) every second (e.g. power consumption of a premises) and transmit them to operational centres, such as cloud servers, using various techniques [78]. The considerable advantages are high efficiency, outage management automation, accuracy in continuous-dynamic electricity distribution and billings, and sustainability in climate change mitigation. To that end, nations around the globe, standardisation assemblies, companies and research entities are working around the clock to regulate this field.

However, the unusual volume of the periodically transmitted data from millions of premises is posing unforeseen bandwidth and storage requirement challenges. For instance, in 2009 and 2011, both the United States and China respectively launched the largest electric grid modernisation investment in their history [131]. The collected waveforms data from one of these projects called 'Western Interconnection Synchrophasor Program' (WISP) that has been recently deployed, where 300 PMUs have been distributed on the west coast of the United States for 15 months, was 100 Terabyte - in other words, more than 220 Gigabytes per day [79]. Therefore, the size reduction of IoTs smart meter readings will have a strong impact on minimising the required bandwidth and utilities communication infrastructure.

The proposed compression methods for readings from IoTs smart meter waveforms can be categorised into two groups - lossy and lossless [30]. Lossy compression is dependent on some information loss while preserving the main features of the

waveforms signal. Consequently, the decompressed signal is somewhat dissimilar to the original. This type of compression was acceptable in the classical grid model - much research been done in this field can be grouped into transformation techniques[31, 33, 119], parametric coding [32] and mixed [121]. This is due to its ability of achieving a higher compression ratio while losing some data. However, lossy compression is recently not recommended for two reasons: (1) after the rise of smart grids and the potential use of their remotely collected readings in billings and financial purposes, and (2) to preserve the privacy and authenticity of the transmitted readings, recent models are utilising steganography to conceal the private information randomly inside these readings [2, 3]. Therefore, losing any bit of these readings is unacceptable.

5.1.1 Motivations

Contrary to lossy, lossless compression is obligated to recover the same waveform signal as the original with zero loss. Due to these restrictions, some work was done in this category, such as in [8, 112, 120] . However, according to a recent state-of-the-art study [30], this path is far from being as mature, in relation to image, voice and video lossless compressions. Surprisingly, all this research was undertaken to compress the transmitted streams from the premises to the operation centres, such as public or private cloud servers, but little attention has been paid to the multi-incoming compressed streams after their arrival. Especially, due to some regulations, these streams should be stored for a certain number of years. This means there is an exponential increase in the cost of storage space and an increased burden of available data management.

One of the main limitations for any lossless compression is that less similarity in waveform readings means less likelihood of compression. In information theory, the mathematical benchmark called entropy (i.e. the minimum number of bits required to represent a value after compression) (see 5.5). The entropy level relies on the redundancy among the samples. This becomes worse in the compressed smart waveform meter readings, because the similarity is already exploited. Therefore,

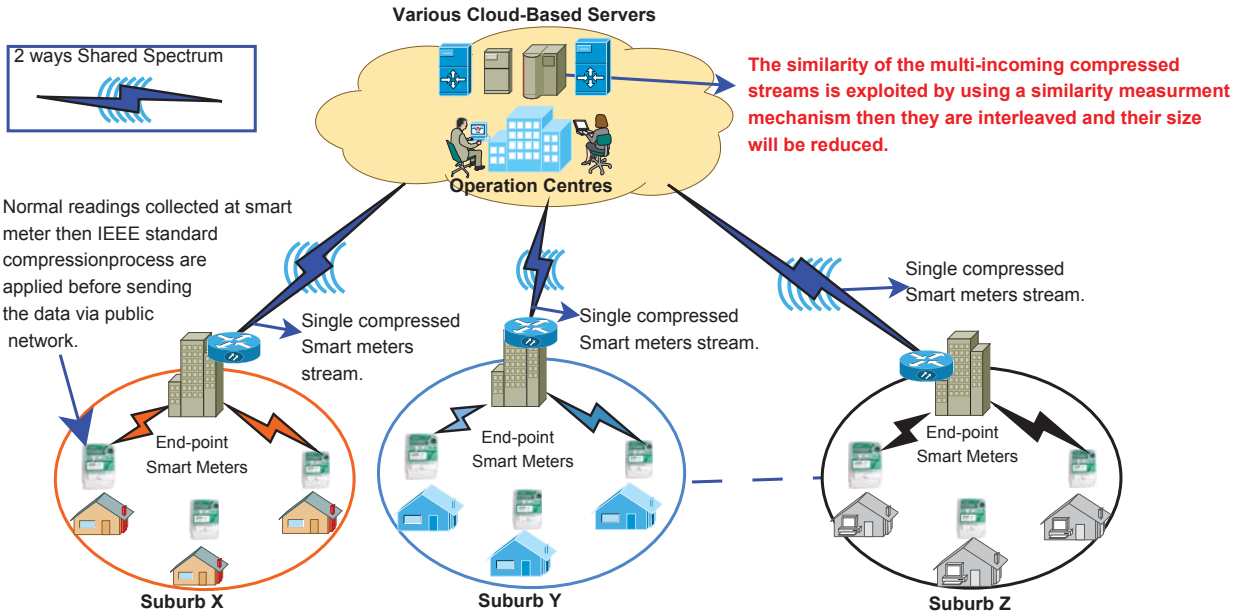

Figure 5.1: The main scenario of our proposed technique where the multi-incoming compressed streams are categorised by their similarity in features followed by interleaving and lossless size reduction.

the main question that drives this chapter is, **can the multi-incoming IoTs smart meter protected compressed streams be re-compressed? In other words, can the size of received compressed streams be reduced without decompression?**

5.1.2 Contributions

In this chapter, a new lossless compression algorithm for waveform IoTs readings of already compressed data is proposed. The main target is pre-processing the protected compressed streams in such a way as to improve the theoretical entropy and invest it. This is effectively achieved using K-means clustering as similarity measurement to classify the compressed streams into subgroups. The streams in every subgroup has been interleaved followed by the first derivative to reduce the values and increase the redundancy. After that, rotation mechanisms have been applied to rearrange the readings in a more consecutive format. Finally, Dynamic-Run Length (D-RLE) and entropy coding are performed. We proved that it is possible to re-compress the

already compressed streams up to 50% of their size without privacy disclosure by re-compressing nor with losing any bits. To the best of our knowledge, there is no other work that tackles this issue in the field of IoTs.

The rest of this chapter is organised as follows. Section 5.2 summarises the relevant work. Section 5.3 introduces our algorithm in different stages. Then, evaluation of the various characteristics of our technique is introduced in Section 5.5. Section 5.7 discusses our performed experiments and the obtained results. Section 5.8 finally draws our conclusions.

5.2 Related Work

Most of the existing research was conducted on gathering waveforms readings targeted on lossy compression. This is because (i) the samples were not being directly transmitted and utilised for crucial purposes, such as real-time diagnoses and billings in the classical grid system, and (2) the efficiency of the transformation technique called 'wavelet transform' that assists in representing waveform signals in a few values (i.e. with losing some bits from every sample). The lossy compression studies can be classified based on the techniques used into transformation, parametric coding and mixed.

As with the work of Santoso et al. [31], transformation models used by employing discrete wavelet decomposition to identify most of the signal energy in low-frequency coefficients by using dbX and allowing others to be neglected. Further work was undertaken using various wavelet families such as B-Spline [116] and Sluntlet [115]. Secondly, parametric coding models, such as the work of Michel et al. [32], utilised damped sinusoids models to elicit the main features of the signal before compression. Finally, mixed transformation and parametric models, such as the work proposed by Moises et al. [121], employed fundamental harmonic and transient coding together.

Contrarily, studies in lossless compression is sparse due to the imposed constraints and the nature of waveforms readings. The lossless compression can be categorised based on the technique utilised into the dictionary, entropy and mixed based models.

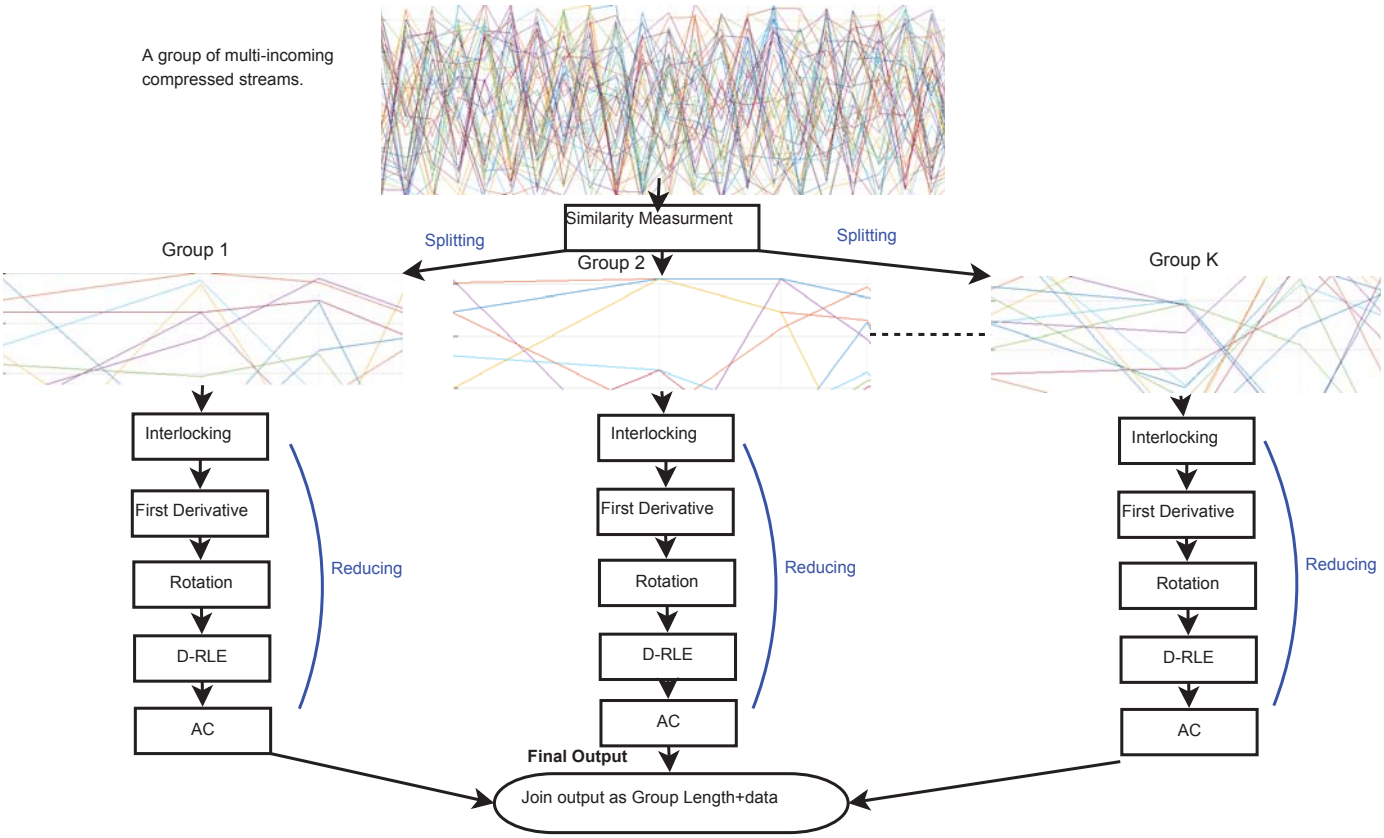

Figure 5.2: An overview for the steps undertaken in our model where the similarity measurement technique is used to split the compressed streams. Then, the grouping and size reduction steps are performed in parallel to exploit the power of the cloud.

Dictionary-based techniques essentially rely on general compressors (e.g. GZIP, ZIP and LZO) where a dictionary is constructed, and more frequent tokens will be represented in fewer bits, whereas more bits are assigned to less frequent samples. For instance, Omer and Dogan [7] employed the Lempel-Ziv to compress a stream of waveforms readings. The accomplished compression ratio was 2.5:1 bin-to-bin. However, dictionary algorithms are mainly designed for letters with English characters where the number of choices is limited. This is unsuitable for waveforms signals due to their floating-point nature. This means every real number has thousands of forms because of its floating values.

Entropy-based techniques are mainly statistical models - these are designed based on screening the probability of every token within a stream and assigning fewer bits

for a higher probability and vice versa. For instance, the work of K. Jan et al. [9] where Arithmetic Coding (AC) was employed to replace the input tokens with a single floating-point value. The accomplished compression ratio was 2.6:1. Z. Dahai et al. [5] also introduced a model that enhances Huffman coding by pre-processing the data utilising higher order delta modulation. The enhancement went from 1.7 to 2.3:1. Moreover, J. Tate [8] recently proposed a model that utilises Golomb-Rice coding after pre-processing the samples with several methods, such as frequency compensated difference. The accomplished compression ratio was 2.8:1.

Mixed techniques are more sophisticated algorithms using both dictionary and statistical mechanisms. This permits exploitation of both the frequency of repetition and its probability within a stream of samples. For instance, K. Jan et al. [113] proposed a model that enhances the LZMA algorithm to minimise the redundancy in waveforms readings. This is achieved by utilising prediction techniques based on differential encoding after optimising the interval selection. The accomplished compression was 2.6:1. K. Jan and T. Tomas [9] also proposed a model that enhances BZIP2 by employing an efficient block-sorting Burrow-Wheeler algorithm and delta modulation. The achieved compression ratio was 2.9:1.

All the above models were conducted on original stream readings and so they exploited the existing high redundancy probabilities among them. However, to the best of our knowledge, there is no existing lossless compression work that targeted already compressed streams.

5.3 Methodology

The main challenge that drives this model is whether multi-incoming compressed streamed be re-compressed? Various of the existing lossless compression algorithms (based on Huffman, Lempel-Ziv or AC) have been directly applied on several compressed streams obtained from Chapter 4 (see Figs. 5.8 and 5.9). The compression ratio was very poor from 0.5 to 1.1 due to the streams that were already compressed. Consequently, the only possible way available is to exploit

the similarities of these compressed streams and design an interlocking friendly compression algorithm. Therefore, K-means clustering has been used as a similarity measurement to classify the compressed streams into subsets. The streams in every subset has been interlocked followed by the first derivative to reduce the space of values and increase the redundancy. After that, two rotation stages have been applied to rearrange the readings in a more consecutive format before employing the developed D-RLE. Finally, entropy coding is performed.

5.3.1 Similarity Measurement - K-means

Based on our preliminary results, mixing all compressed streams will result in a poor re-compression ratio due to the dissimilarity in streams characteristics which increase the noise. Therefore, a well-known unsupervised learning technique called K-means [132] is used as a similarity measurement to classify n observations into K groups. The main idea is that, let's assume $(x_1, x_2, ..., x_n)$ are the n incoming compressed streams where each stream is a d-dimensional vector. K-means will partitions the n streams into $K(\leq n)$ groups $(G_1, G_2, .., G_k)$ by summation of distance functions of each point in the group to K centre. The objective is depicted in Eq 5.1

$$\sum_{i=1}^{k}\sum_{j=1}^{\sigma_i}(\|x_i - \mu_j\|)^2 \tag{5.1}$$

where σ_i is the data points in the ith group, μ_j is the centre of the ith group and $\|x_i - \mu_j\|$ is the Euclidean distance between x_i and μ_j.

The algorithm started by selecting cluster centres μ_j. The distance between each reading point x_i and μ_j is then calculated. Next, the reading point x_i is assigned to μ_j based on the best minimum distance. After that, a new cluster centre μ_j is recalculated as shown in Eq. 5.2

$$\mu_i = \left(\frac{1}{\sigma_i}\right)\sum_{j=1}^{\sigma_i} x_i \tag{5.2}$$

where σ_i represents the reading points in i^{th} cluster. The distance between x_i and μ is then recalculated. The assignment process will be repeated (see Eq 5.3), until no further data points need to be reassigned.

$$G_i = \left\{ x_p : \|x_p - m_i\|^2 \leq \|x_p - m_j\|^2 \, \forall_j, 1 \leq j \leq k \right\} \tag{5.3}$$

The most crucial part is how centroid points are chosen. Therefore, to avoid the exponential time complexity of standard algorithm, the idea proposed by Arthur and Vassilvitskii [133] has been used by utilising a heuristic to find the centroid seeds for the algorithm as follows. Only one random centre μ is uniformly chosen from among the readings. Then, the distance between x_i and the closest centre (i.e. chosen one) is computed. Next, one of readings is chosen to be the new centre μ using a weighted distribution probability (see Eq. 5.4). These steps are repeated until k centres are chosen.

$$\frac{d^2\left(x_i, \mu_p\right)}{\sum_{\{j, x_j \in \Re_p\}} \left(x_j, \mu_p\right)} \tag{5.4}$$

where $\Re_p$ is the group of all observations nearest to centroid. μ_p and x_i are belonging to $\Re_p$.

5.3.2 Parallel Size Reduction

After completing the similarity measurement process, each resultant group is combined and its size will be reduced using the following steps. These steps are not dependent and so designed to run in parallel to exploit the power of cloud.

Readings Interlocking

Let's assume G_i is one of the resultant groups. It has $1, 2, ..., n$ vectors that represents multiple compressed streams as shown in Eq. 5.5.

$$G_{m,n} = \begin{pmatrix} a_{1,1} & a_{1,2} & \cdots & a_{1,n} \\ b_{2,1} & b_{2,2} & \cdots & b_{2,n} \\ \vdots & \vdots & \ddots & \vdots \\ c_{m,1} & c_{m,2} & \cdots & c_{m,n} \end{pmatrix} \tag{5.5}$$

Therefore, these streams will be overlapped to exploit the similar features (see Eq. 5.6).

$$a(1,1), b(2,1), ..., c(m,1) \tag{5.6}$$

To avoid any sharp exponential deviations in the overlapped readings and increase the redundancy, the first derivative is applied as shown Eq 5.7.

$$f_d = [\Upsilon(2) - \Upsilon(1)\Upsilon(3) - \Upsilon()2)...\Upsilon(\Lambda) - \Upsilon(\Lambda - 1)] \tag{5.7}$$

where Υ represents data points in the combined stream an Λ is the latest value.

Rotation

Rotation Stage 1: Based on experimental observations, the resultant values, after applying the first derivative, reflects that there is high redundancy, but in a very scattered format that impedes any size reduction attempts. Consequently, Burrow-Wheeler Transform (BWT) is employed to reshuffle the samples resulting in a long consecutive and identical sequence. Originally, BWT was proposed by Michael Burrows and David Wheeler [124] to rearrange text streams into a format that boosts its compressibility by utilising mechanisms, such as Move-To-Front (MTF) and Run Length (RLE). The beauty of this algorithm is that zero additional overheads were needed to reverse it. Basically, the data (i.e. 1 to n) is rotated lexicographically. Lets assume Ω is the textual or numerical of a symbols group form (i.e. numerical in our algorithm) to be compressed.

$$\Omega = \Omega_1, \Omega_2, ..., \Omega_n \tag{5.8}$$

Iteratively, the vector Ω is rotated to the left which results in a new 2D matrix called β, as shown in Eq. 5.9

$$\beta = \begin{pmatrix} \Omega_1 & \Omega_2 & \Omega_3 & \cdots & \Omega_n \\ \Omega_2 & \Omega_3 & \cdots & \Omega_n & \Omega_1 \\ \Omega_3 & \cdots & \Omega_n & \Omega_1 & \Omega_2 \\ \vdots & \vdots & & \ddots & \vdots \\ \Omega_n & \Omega_1 & \Omega_2 & \cdots & \Omega_n - 1 \end{pmatrix} \tag{5.9}$$

From Eq. 5.9, it is obvious that each rotation of Ω is represented as a row in β. These rows are then sorted in ascending order which will generate a new version of the matrix called $\widetilde{\beta}$. Only the last column C of β and the original block index I are kept to be used for retrieving the original order.

For a clear understanding, a portion of the resultant first derivative values are presented in Table 5.1. These samples have been replaced by characters (i.e. represent Ω) for the sake of simplicity. β is generated by rotating Ω for n (i.e. elements number) times as shown in Eq. 5.10

$$\beta = \begin{Bmatrix} \$ & a & b & a & a & b & a \\ a & \$ & a & b & a & a & b \\ b & a & \$ & a & b & a & a \\ a & b & a & \$ & a & b & a \\ a & a & b & a & \$ & a & b \\ b & a & a & b & a & \$ & a \\ a & b & a & a & b & a & \$ \end{Bmatrix} \tag{5.10}$$

where \$ represents the start of data. The rows of β will then be sorted which results in a new form $\widetilde{\beta}$ as depicted in Eq. 5.11

Table 5.1: Conversion from numeric to characters

ω	309	501	309	309	501	309
Ch	a	b	a	a	b	a

$$\widetilde{\beta} = \begin{Bmatrix} a & \$ & a & b & a & a & \boldsymbol{b} \\ a & a & b & a & \$ & a & \boldsymbol{b} \\ \boldsymbol{a} & \boldsymbol{b} & \boldsymbol{a} & \boldsymbol{a} & \boldsymbol{b} & \boldsymbol{a} & \boldsymbol{\$} \\ a & b & a & \$ & a & b & \boldsymbol{a} \\ b & a & a & b & a & \$ & \boldsymbol{a} \\ b & a & \$ & a & b & a & \boldsymbol{a} \\ \$ & a & b & a & a & b & \boldsymbol{a} \end{Bmatrix} \tag{5.11}$$

The last column C and the index I (e.g. 3 in this example) represent the output.

C and I are the crucial parameters used by the decoder to recover the original form. This is achieved by building a temporary $n \times n$ matrix (where n is the elements number) and C represents its last column. By sorting this column, the first column is retrieved. Next, all successive pairs are recovered using those two columns and the resultant matrix is like $\widetilde{\beta}$. Subsequently, by using the index I, retrieving the original form will be easy.

Rotation Stage 2: Despite the resultant BWT values precisely gathering identical symbols in the long runs, these values still sharply vary from very low (e.g. 20 and 21) to much higher figures (e.g. 4000 to 6000). Consequently, MTF transform is employed to boost the influence of any entropy based encoder (e.g. Arithmetic Coding) to achieve the highest compression rate. MTF is a lightweight mechanism introduced by Ryabko [125] to enhance the low values (e.g. close to zero) probability while minimising the high values in a given list of data. The basic idea is that the data list symbols are substituted by their positions in a unique list. Therefore, the long sequential identical symbols will be substituted by as many zeros, whereas a posterior (i.e. not regularly used) symbols will be exchanged for larger values.

Table 5.2: MTF of $L = [b, b, a, a, a, a, a, a]$ and $u = [a, b]$

L_i	u	ϑ
b	a, b	1
b	b, a	0
a	b, a	1
a	a, b	0
a	a, b	0
a	a, b	0
a	a, b	0
a	a, b	0

Let's assume L is an obtained list from the BWT process and u is the unique symbols of L (i.e. $u \in L$). Then, the summary of MTF process can be depicted in the following steps. (a) L is used to populate u. (b) Every item L_i of vector L is substituted as its the symbol numbers preceding it in u. (c) Latter output is built as a list ϑ by collecting the resultant codes of step 2. The recovery process is the opposite of these steps.

For a clear illustration, assume the BWT resultant list $L = [b, b, a, a, a, a, a, a]$ and its unique list $u = [a, b]$ (see Table 5.2). The initial token L_0 is b, and it is preceded by one symbol in u. Therefore, the digit one is produced in ϑ and the symbol b is moved to the front of $u = [b, a]$. The next token L_1 is b, which is the first in u, and so the produced value is zero with no need to update u. These steps are continued until the last token is reached, so the resultant output will look like: $\vartheta = [1, 0, 1, 0, 0, 0, 0, 0]$.

Dynamic Run Length

Hypothetically, the MTF output includes a series of identical sequential tokens. Consequently, to exploit this fact, a simple mechanism called Run Length (RLE) [126] is employed before the entropy encoding. RLE focuses on substituting the similar consecutive symbols by their count. Allow s to represent a symbol appearing as n in sequential times in a vector V. The n cases are then substituted by ns. The consecutive n appearances of the symbol are called 'run length'. For instance, the sequential zeros in $\vartheta = [1, 0, 1, 0, 0, 0, 0, 0]$ will be $\vartheta = [1, 0, 1, 0\#5]$. However, the observation was that a naive implantation of RLE is not always useful and may

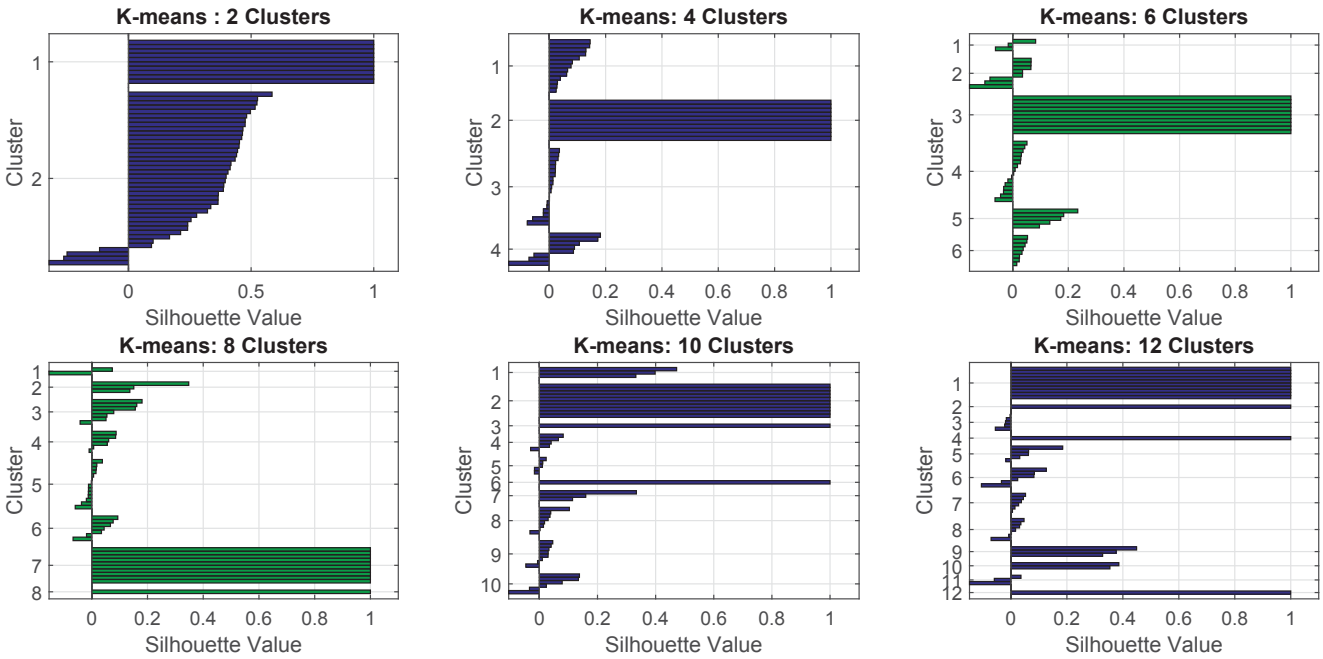

Figure 5.3: Graphical representation of the consistency among various compressed streams combined (or K-means) clusters. Obviously, six and eight clusters are the best in terms of the cohesion.

increase rather than decrease. This is due to its static nature where each consecutive symbol is replaced even in the case if no repeated tokens occurs. RLE is improved to be a dynamic without the extra overhead based on the thresholds $t = t1, t2, .., t_n$ that monitor the consecutive occurrences of the symbols in a given vector. Only one bit is added at the beginning of the resultant encoded stream to indicate whether D-RLE encoding process was employed or not.

Entropy Coding

To achieve the highest possible compression ratio, an entropy coding mechanism called Arithmetic Coding (AC) is ultimately employed in our model. AC is a widely-known variable length statistical coding by which repeatedly occurred values are represented with fewer bits and less frequently appearing tokens are symbolised with higher bits number [126]. AC proved its superiority in most respects to other well-known entropy algorithms such as Huffman coding. This is due to its succinct representation of

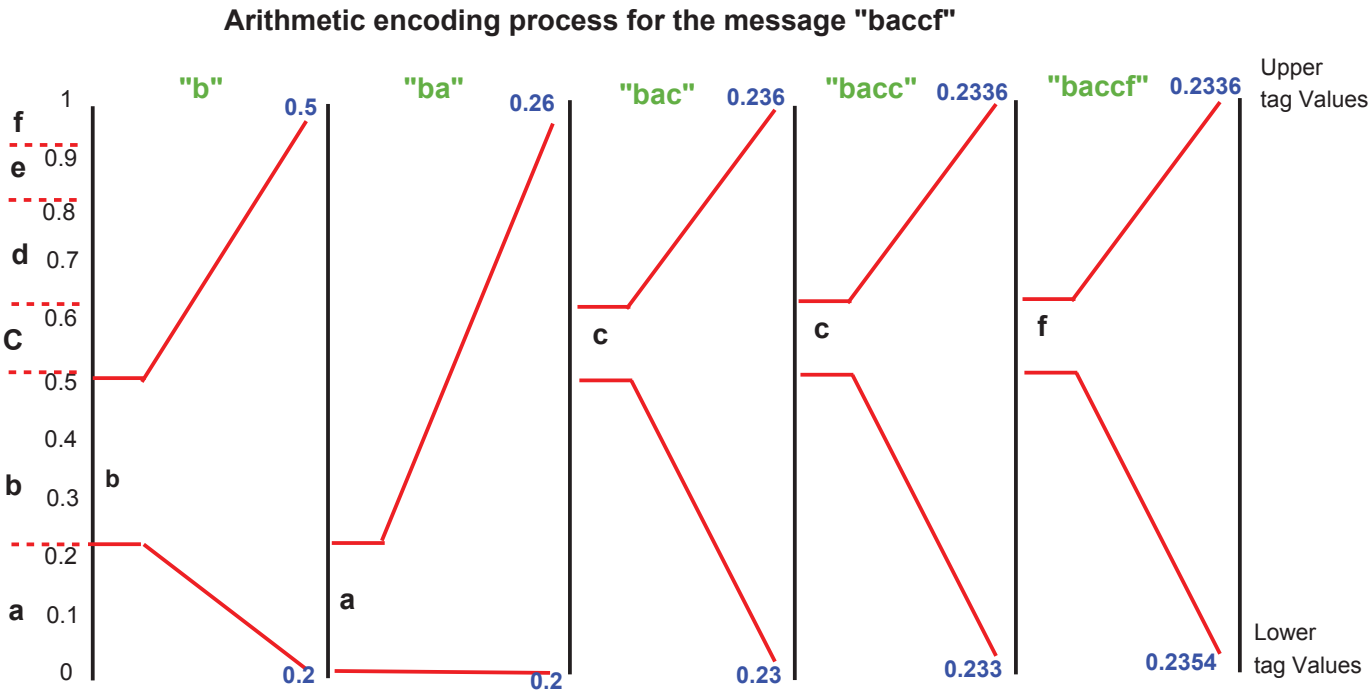

Figure 5.4: Graphical representation for Arithmetic encoding steps for a message b,a,c,c,f

the entire message in a single value as a fraction n where $(0.0 \leq n < 1.0)$, whereas other algorithms working on separating the input into isolated component tokens and substituting each with a unique code.

The general idea is that after choosing a specific interval, the symbols list will be scanned and based on its tokens probabilities the ultimate interval will be narrowed. AC core steps are summarised as follows.

1. The current interval is specified as (0,1).

2. The following two steps are repeated for each token S_i in the data vector.

   a) The current interval is divided into subintervals under the condition that their sizes are proportional to the tokens probabilities.

   b) A subinterval for S_i is chosen which will represent the new current interval.

3. After processing the entire data vector, the result should be any value that distinctly identifies the present interval (i.e. any value in the current interval).

Table 5.3: The message m probability distribution

Symbol	Probability	Accumulative range
a	0.2	(0.0,0.2)
b	0.3	(0.2,0.5)
c	0.1	(0.5,0.6)
d	0.2	(0.6,0.8)
e	0.1	(0.8,0.9)
f	0.1	(0.9,1.0)

Interestingly, the deeper scanned values in the data vectors are, the smaller current interval is obtained. The resultant output is a single measure called 'tag value' that does not include the individual tokens codes.

To demonstrate the AC coding mechanism, let's assume that an entire message M has a probability distribution as given in Table 5.3. For brevity, a fraction of that message $\tilde{m} = (b, a, c, c, f)$ is encoded. The probability boundary is between $(0, 1)$. To begin with, due to the occurrence of symbol b', the tag value should be in the range $(0.2, 0.5)$. After that, the token 'a' is appeared, so the present interval between $(0, 0.2)$ that will be used to calculate the lower and upper appeared in the equations 5.12 and 5.13.

$$w_n = l_{n-1} + (w_{n-1} - \rho_{n-1}) \times F_x(x_{n-1}) \tag{5.12}$$

$$\rho_n = w_{n-1} + (\rho_{n-1} - \rho_{n-1}) \times F_x(x) \tag{5.13}$$

w and ρ represent the lower and upper boundaries of the nth token. F_x is the frequency accumulation. The resultant tag values of symbols sequence 'ba' are $(0.2, 0.26)$. This will accumulatively continue for the full message. The ultimate tag values output has been summed up in Fig. 5.4. The average of both the final upper and lower tags $\frac{w_n+\rho_n}{2} = \frac{0.23354+0.2336}{2} = 0.23357$ represents the compressed value and will be transformed into binary.

The decoder side requires both the average value and the message probabilities. Subsequently, it proceeds through similar steps but in an inverse manner where the probability accumulation is used to find the symbols.

5.4 Decompression and Recovery

The recovery process is almost similar to the steps stated above, but in an opposite manner. It begins by Arithmetic decoding followed by D-RLE if needed based on the conditions mentioned in Section 5.3.2. Then, MTF and BWT are applied respectively. The output represents the derivative values and so their inverse process is employed to reconstruct the actual symbols. These symbols are the compressed streams in an interlocking way. From that, they are disunited to their single compressed streams and so their original format is recovered in a lossless format.

Fig. 5.7 in Section 5.7 presents instances of various compressed streams before and after the size reduction process which clearly proves that the compressed streams are totally retrieved with zero loss.

5.5 Evaluation

Various matrices are used to examine the effectiveness of our lossless size reduction model of the multi-incoming protecting compressed streams on cloud level from both theoretical and experimental angles.

5.5.1 Silhouette Measurement

To validate the coherence of the used similarity measurement clustering techniques, a mathematical model called Silhouette is used. It is a graphical representation technique that was proposed by Peter J. Rousseeuw [134] that proves consistency within data clusters and clearly reflects the correlation of the objects within that group. The Silhouette model produces a value in the range of -1 to 1 in which the higher the value, the more the object is well matched to that group and vice versa.

For clarity, let's assume n vectors are clustered into K groups using any similarity measurement model, for instance K-means. For each group G, $x(i)$ represents the average dissimilarity (i.e. distance) of i within the group the lesser the value, the better the matching. Also, let $y(i)$ be the lowest dissimilarity of i within the group.

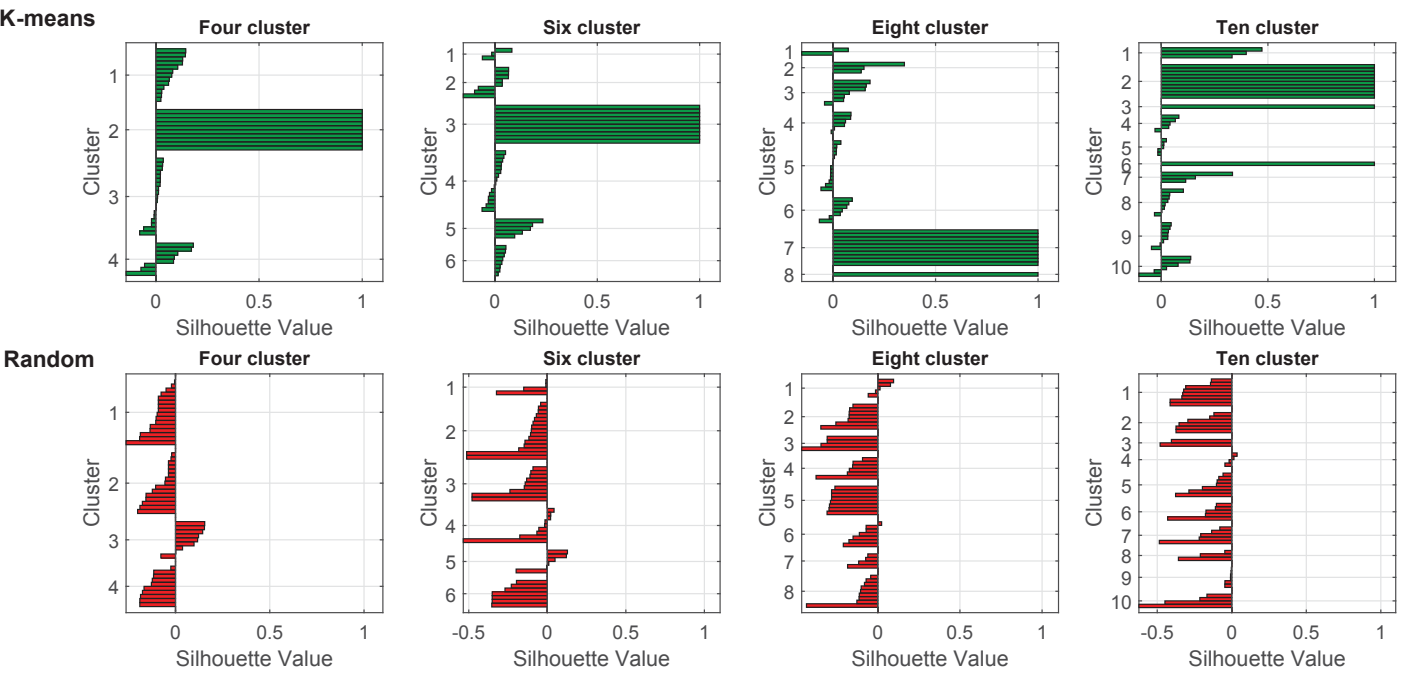

Figure 5.5: Comparison between the consistency among the clusters by changing two parameters. These are the number of clusters and the similarity measurement technique (i.e. K-means vs Rand).

Hereby, silhouette s can be defined as follows.

$$s(i) = \frac{y(i) - x(i)}{\max\{x(i), y(i)\}} \tag{5.14}$$

$$s(i) = \begin{cases} 1 - \frac{x(i)}{y(i)}, & \text{if } x(i) < y(i) \\ 0, & \text{if } x(i) = y(i) \\ \frac{x(i)}{y(i)} - 1 & \text{if } x(i) > y(i) \end{cases} \tag{5.15}$$

From 5.14 and 5.15, it can be derived that $-1 \leq s(i) \leq 1$.

5.5.2 Theoretical Entropy

The so-called entropy of a signal in the information theory field represents the lowest bit-rate (the optimum compression is assumed) needed for transmitting this signal [127]. Consequently, to monitor the influence of pre-processing the compressed streams

in our model, the theoretical entropy is measured for every smart grid compressed stream before and after employing our model. After that, quantitative calculation comparison is performed between the theoretical entropy and the accomplished size reduction ratio.

Let's assume a compressed incoming stream consisting of the symbol points $d[1], d[2], ..., d[N]$. The optimum likelihood entropy in bits is calculated as

$$H(d) \triangleq -\sum_{i \in \mathbb{R}(d)} \widehat{p}(i) \log_2(\widehat{p}(i)) \tag{5.16}$$

$$\widehat{p}(i) \triangleq \frac{1}{N} \sum_{n=1}^{N} \delta_i(d[n]) \tag{5.17}$$

$$\delta_i(d[n]) \triangleq \begin{cases} 1, & \text{if } d[n] = i \\ 0, & else \end{cases} \tag{5.18}$$

where $\widehat{p}(i)$ is the experimental probability of $i \in \mathbb{R}$ and $\mathbb{R}(d)$ represents the range of d.

The smallest entropy (or optimum case) happens when all d symbols are equal, which results in $H_{min} = -1 \log_2(1) = 0$.

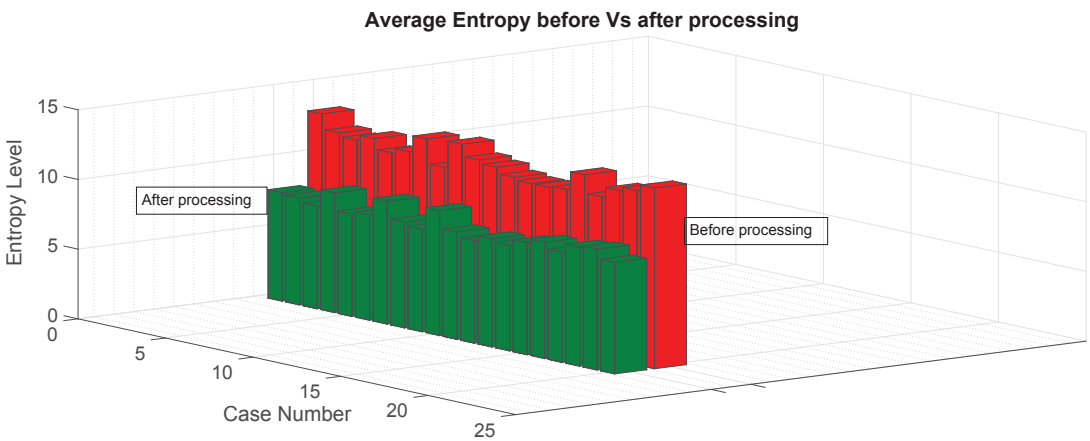

Figure 5.6: Comparison between the average entropy calculated from the compressed streams before and after aggregation and processing.

On the other hand, the worst-case (or maximum entropy) happens when each symbol in $\mathbb{R}$ occurs at the similar frequency $1/\mathbb{R}$, in which $|\mathbb{R}|$ reflects the original elements in $\mathbb{R}$ (see Eq. 5.19).

$$H_{\max} = -\sum_{i \in \mathbb{R}} \frac{1}{|\mathbb{R}|} \log_2(\frac{1}{|\mathbb{R}|}) = \log_2 |\mathbb{R}|. \tag{5.19}$$

5.6 Experimental Ratio

The Compression Ratio (CRi) is the essential benchmark to empirically measure any proposed compression model. Lets symbolise the original compressed streams block O (i.e. its unit in bit or byte) and the resultant re-compressed symbols R. Consequently, the experimental CRi in the results section is measured and defined in Eq. 5.20.

$$CRi = \frac{O}{R} \tag{5.20}$$

To produce the multi-compressed streams dataset, a widely-known leading power quality storage standard for electric waveforms power system utilised in most of the smart grids, is called the Power Quality Data Interchange Format (PQDIF) defined by the IEEE1159 working group [128], has been employed. Every reading represented as 16 bit and the typical suggested block size used is about 1500 readings. Our generic Gaussian-based size reduction explained in Chapter 4 is employed here and has proven to give the best lossless compression ratio.

5.7 Implementations

5.7.1 Datasets

The Laboratory for Advanced System Software collected and published a detailed smart meters datasets as a part of project called 'Smart' [75, 76]. These datasets have been thoroughly used in our experiments. The datasets entries represent a periodical readings per minutes from three houses for more than three months. The entries can be classified into: (i) power consumption as watts and heat index, and (ii) environmental

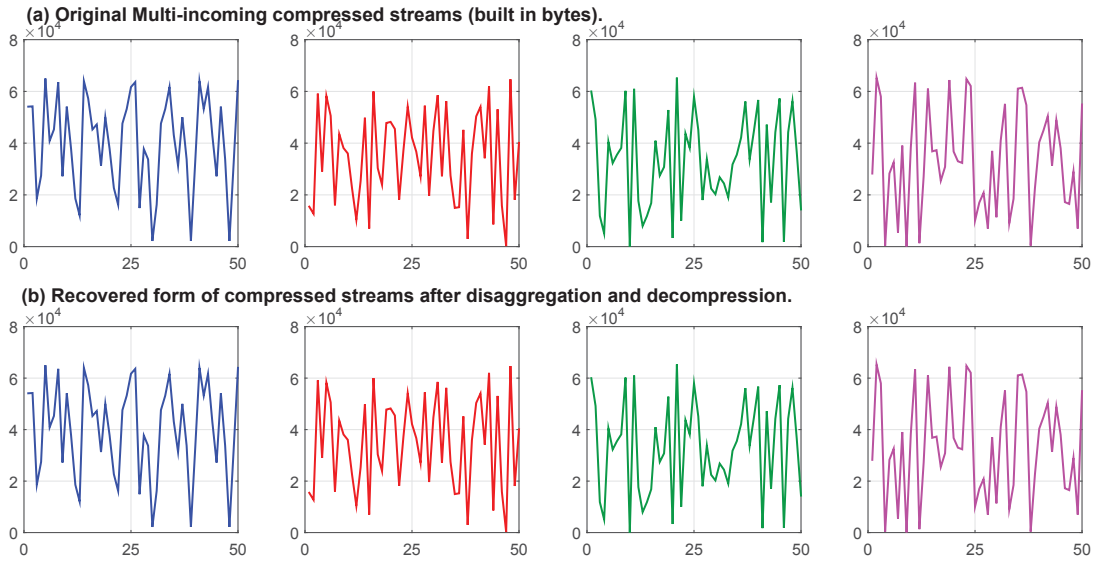

Figure 5.7: Four examples of compressed watts consumptions readings collected from different homes: (a) Direct plot of single compressed streams form, and (b) plot of these streams after disaggregation and recovery.

features as inside and outside temperature, inside and outside humidity, and wind chill. Additionally, a detailed electricity consumption (i.e. per minutes) from about 400 anonymous premises for $(24 \times 30 \times 3)$ hours is provided. According to the spatial and temporal aggregations definition, these symbols are temporal due to their separate periodical collection from every individual premise by equipping it with a smart meter.

Our generic Gaussian based compression (see Chapter 4) is applied on every single stream, to generate the multi-incoming compressed streams as explained in the evaluation in Section 5.5. The compressed streams are laid out as the bed-test in all our experiments.

5.7.2 Experiments and Results

Our prime executed experiments will be done at the operation centres or cloud level after receiving an overwhelming amount of protected IoTs compressed streams from huge number of remote end-points. The experiments can be categorised into (a) similarity measurements, interleaving and size reduction processes, and (b) an original

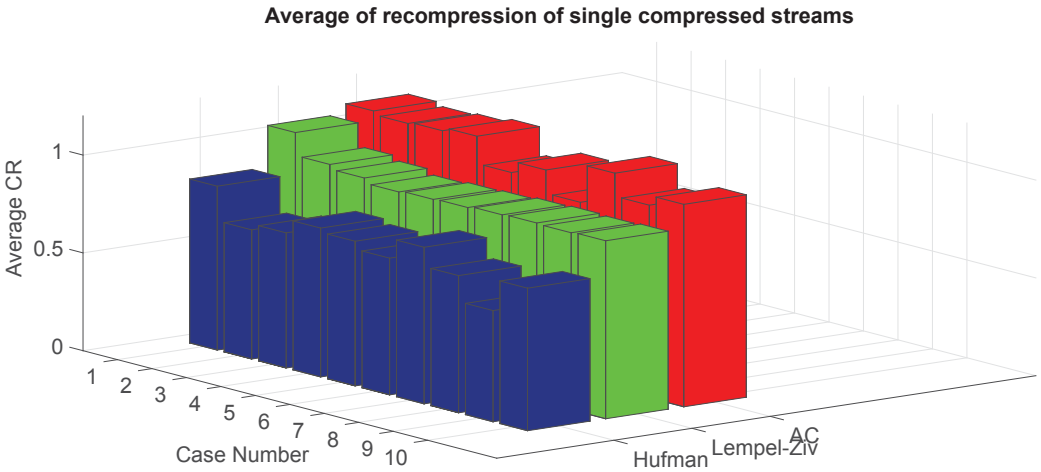

Figure 5.8: Compression ratio after re-compressing single incoming streams directly using various well-known lossless compressors both dictionary-based (Lempel-Ziv [7]) and entropy-based (Huffman [5] and AC[9])

.

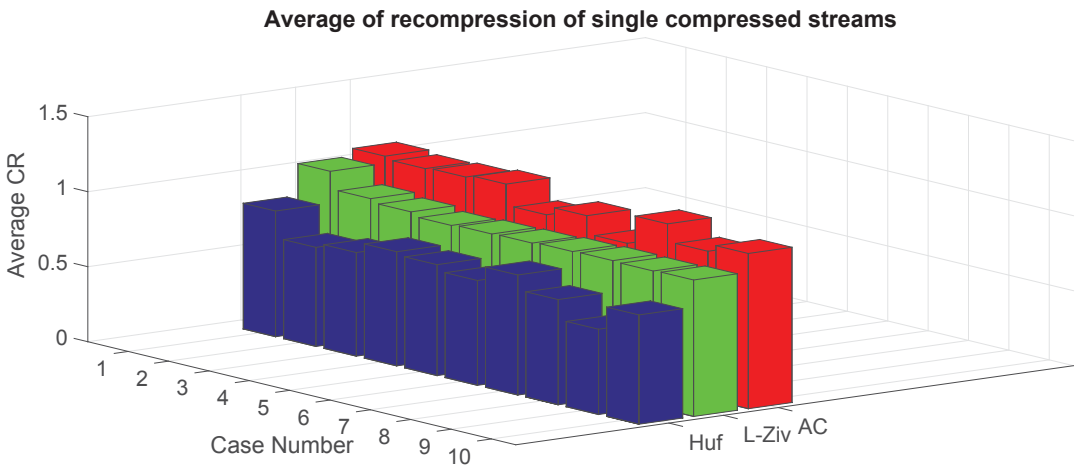

Figure 5.9: Average compression ratio after re-compression the multi-incoming compressed streams all together which was very poor. This is because enforcing all together will result in a high noise.

format recovery and disuniting. Both categories are designed in such a way that can be done in parallel to take advantage of cloud power.

To obtain unbiased outcomes, all compressed streams mentioned above have been employed in our model. Identically, many well-known lossless compression algorithms have been applied directly on the multi-compressed streams as shown in Figs 5.8 and 5.9 to accurately provide a clear comparison and prove that these streams can not be re-compressed using traditional models.

For brevity in this chapter, the results have been summarised as follows. Firstly, Fig 5.8 and 5.9 show the exact CRi (0.5 to 1.3) of single and collective (i.e. all together) multi-incoming compressed streams after applying many-known lossless algorithms from both entropy and dictionary fields such as Huffman [5], Arithmetic Coding [9] and Lempel-Ziv [7]. Secondly, Fig 5.3 highlights the examination process of our similarity measurement technique, K-means, to select the best K number (i.e. 6 and 8 are the best in the case of 56 streams) using the Silhouette benchmark. Also, Fig. 5.5 reflects the superiority as a graphical comparison (i.e. using Silhouette benchmark) between our similarity measurement using K-means against an agnostic random grouping. Thirdly, Fig 5.10 emphasises a possible enhancement on the CRi level by changing various parameters in our algorithm, such as static RLE, D-RLE, agnostic similarity measurement, and K-means similarity measurement in relation to the number of clusters. Each of which these groups contain 56 compressed streams. Fourthly, Fig 5.11 shows the average of CRi ratio obtained from the above four groups (i.e. 8 K-means clusters + D-RLE is the best). Fifth, Table 5.4 presents the exact CRi achieved from the four groups shown in Fig 5.11. Finally, Fig 5.7 shows an example of a plot of various original compressed streams before and after the aggregation and size reduction process.

5.7.3 Discussion

It is obvious, from the experiments, that it is unlikely to re-compress the multi-incoming compressed streams by simply applying known algorithms. This is because the redundancy in these streams is already exploited and re-applying the same algorithms will have little effect if not worse (revisit Fig 5.8 and 5.9). On the contrary, imposing all compressed streams together is also not useful due to the huge dissimilarity in their features. In other words, forcing all streams together will boost the noise and, therefore, decrease the chances of a size reduction.

By exploiting the possible similarities in various compressed streams using similarity measurement techniques, our model achieved up to 2:1 size reduction level.

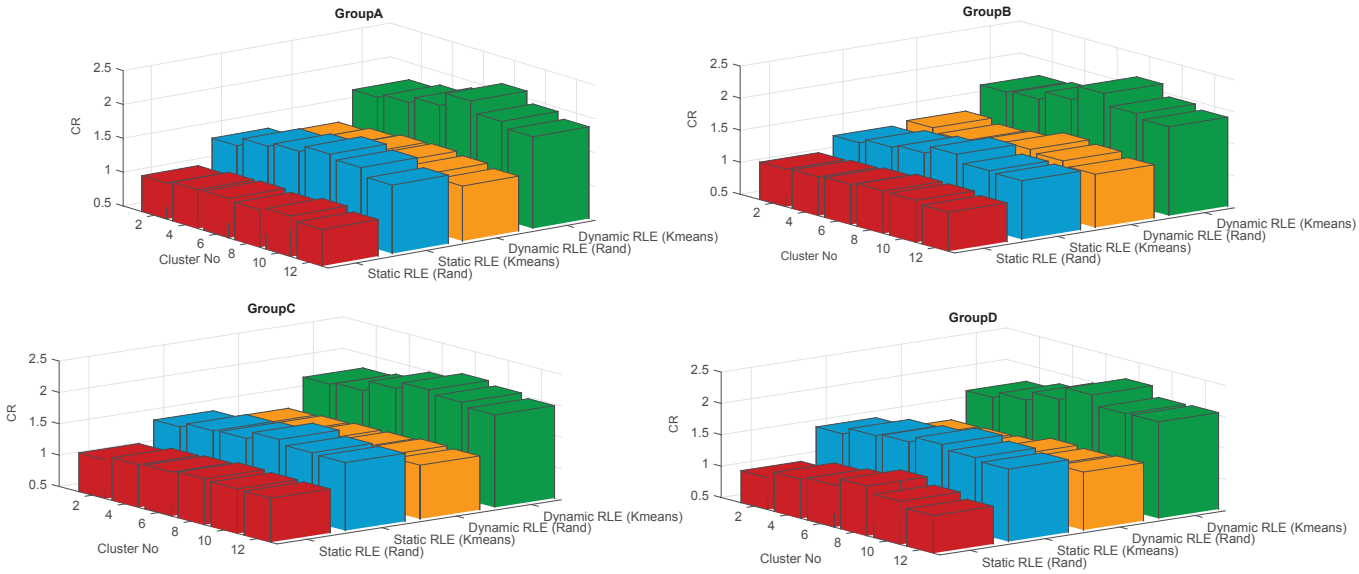

Figure 5.10: 4 groups of achieved re-compression ratio of multi-incoming compressed streams by changing the similarity measurement technique, number of clusters and RLE. Every group contains 56 compressed streams.

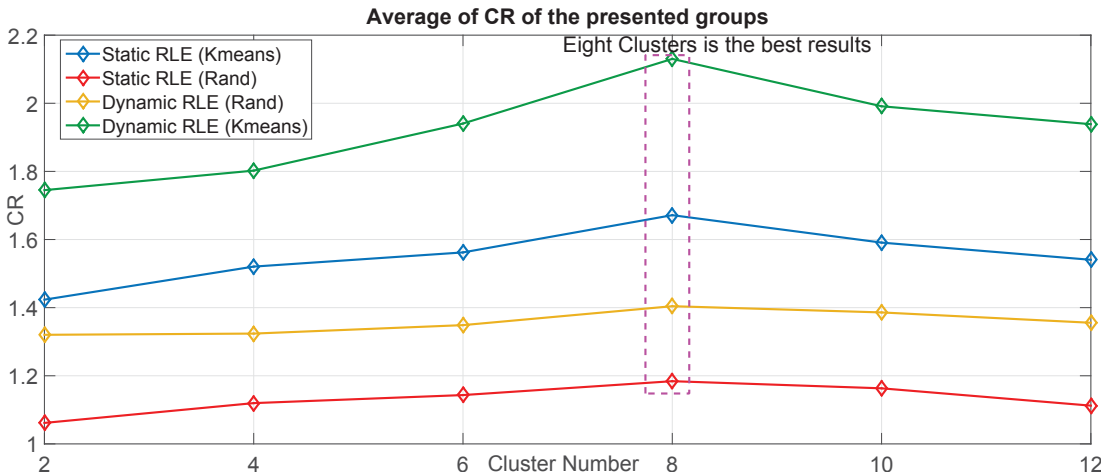

Figure 5.11: The average of re-compression ratio of the four groups examined in Fig.5.10 which shows the best combination of our technique parameters.

This means every 1 Gigabyte byte can be reduced to 500 Megabytes. This has been emphasised theoretically by comparing the entropy before and after our technique (see Fig. 4.6) and experimentally as presented in Fig. 5.10.

Table 5.4: Compression ratio

No	Cluster No	Static RLE (Rand)	Static RLE (Kmeans)	D-RLE (Rand)	D-RLE (Kmeans)
1	2	1.03	0.40	1.30	0.75
2	2	1.08	0.31	1.36	0.73
3	2	1.12	1.46	1.31	1.77
4	2	1.01	1.52	1.30	1.73
5	4	1.07	1.53	.35	1.81
6	4	1.10	1.38	1.30	1.76
7	4	1.18	1.53	1.33	1.81
8	4	1.12	1.63	1.31	1.83
9	6	1.08	1.59	1.39	1.90
10	6	1.12	1.43	1.33	1.90
11	6	1.20	1.55	1.35	1.98
12	6	1.17	1.67	1.33	1.98
13	8	1.05	1.69	1.40	2.11
14	8	1.16	1.55	1.44	2.13
15	8	1.23	1.67	1.35	2.10
16	8	1.29	1.77	1.42	2.19
17	10	1.10	1.63	1.36	1.94
18	10	1.15	1.43	1.40	1.95
19	10	1.21	1.60	1.38	2.04
20	10	1.19	1.70	1.40	2.03
21	12	1.04	1.51	1.31	1.85
22	12	1.11	1.41	1.33	1.89
23	12	1.20	1.58	1.36	1.97
24	12	1.10	1.66	1.42	2.04
Avg		1.13	1.55	1.36	1.92

5.8 Chapter Summary

In this chapter, a novel lossless parallel compression algorithm was introduced to prove the possibility of reducing the size of already compressed waveform IoTs readings. The target was through pre-processing the data to enhance the entropy. This is successfully achieved by employing K-means clustering as similarity measurements to classify the compressed streams into subsets to reduce the noise of dissimilar compressed streams. The tokens of every subset have been interlocked followed by the first derivative to reduce the space of values and boost the redundancy. After that, rotation mechanisms have been applied to rearrange the symbols in a more consecutive format before employing dynamic RLE. Finally, entropy coding is performed. Both mathematical

and empirical experiments proved the possibility of enhancing the entropy, which almost reduced by half, and the resultant size reduction (i.e. up to 50%). To the best of our knowledge, there is no other work that tackles this issue in the field of IoTs streams.

Chapter 6

Conclusion

In this chapter, a holistic overview summary is presented, concentrating on the core thesis challenges and the research questions derived from these challenges. The main contributions achieved by answering the research questions are summarised along with the core findings of this thesis. Section 6.1 highlights the main research goals and restates the research questions. The contributions are reintroduced in Section 6.2. Section 6.3 discusses some of the interesting findings of our research. Finally, Section 6.4 highlights the potential paths that can be followed to further develop using this thesis work.

6.1 Research Aims

The Internet of Things (IoTs) sensor streams have become a vital category upon which many applications rely. For example, smart meters collecting household power and gas consumptions periodically every second and then transmit them wirelessly through various channels and public hops to the operation centres [17]. For the unprecedented volume of large streams and for real-time analysis, the operation centres are employing third-party cloud servers where various entities process the data on a real-time basis for billings and dynamic managements. There are many crucial projects that are working towards making the world a more convenient place, such as with climate change, transportation, healthcare and many more. Despite the clear benefits of these

applications, they pose the unique challenges [18], such as how can we have a suitable balance between: (1) guaranteeing the streams security (i.e. privacy, authenticity and integrity) while not hindering the direct operations on those streams, (2) handling the data management issues, such as the size during the transmission and storage. These challenges become more complicated in cases where the streams should reside on the third-party cloud servers.

Therefore, the main targets of this thesis are to introduce congruous algorithms that ensure the security (i.e. privacy, authenticity and integrity) of dynamic-high IoTs streams in the cloud environment, while providing better data management (as in size reduction). On the other hand, they allow direct operations on third-party servers without security disclosure.

To overcome the above unique challenges and achieve the stated goals, we formed the following research questions:

- **RQ-1. How can the privacy of the sensitive information and the authenticity of the transmitted IoTs sensor streams be ensured without hindering direct operations at intermediate hops or cloud?**
- **RQ-2. How can any alteration to the transmitted information be detected and recovered without hindering direct operations at intermediate hops or cloud?**
- **RQ-3. How can the size of the protected transmitted streams which contains the encrypted hidden information be blindly reduced without any security disclosure?**
- **RQ-4. Can the multi-incoming protected compressed IoTs sensor streams be re-reduced without privacy or authenticity disclosure?**

6.2 Research Contributions

To address the research questions explained in Section 1.2, and to bridge the gap between security techniques and data management models, this thesis introduced a

novel framework that consists of several new techniques to the area of high IoTs (sensors and smart meters) data streams. The detail list of these contributions is as follows.

- **Privacy preserving of confidential information and authenticity of gathered IoTs streams while not hindering direct operations on the data**

  To ensure the privacy of confidential information and the authenticity of the transmitted readings at the same time, steganography is employed to embed the sensitive information randomly inside the transmitted readings. The main merit of steganography is that it adds another layer of security, requires much lower processing capabilities without changing or increasing the form of original data. However, steganography has two main issues: (1) authentication problem where the adversaries can retrieve the embedded information once they know of its existence, and (2) capacity and distortion issue on the source used for hiding. Therefore, to overcome the first limitation, two mathematical security models were designed and implemented for employing the key to (1) encrypt the confidential information, (2) shuffle the coefficients into a random hierarchy, and (3) randomly generate an order used in the embedding process. To broaden hiding capacity, insertion of the private information and to maximise the randomisation, two signal processing techniques (i.e. the Walsh-Hadamard and Discrete Wavelet Transform) were exploited to transform the readings from their spatial to the frequency domain to classify and gather most of the readings features sensitivity in a few coefficients allowing others to be freely utilised to embed more data. Two new models are proposed in this chapter that vary in their simplicity (Walsh-Hadamard based stenography) vs. security (Wavelet based stenography). Both techniques are neither increasing nor changing the form of the transmitted readings. This means only data owners can recover the seal, whereas others are just monitoring the protected form of the IoTs readings.

- **Manipulation detection, remedy, and recovery while not hindering direct operations on the data nor changing the form of the readings**

  Error detection and correction algorithms (BCH) is employed with steganography to guarantee a strong level of reliability and integrity of the transmitted protected data. To conquer the main issue of error detection and correction codes that is increasing or changing the original data form, a novel hybrid model that combines advanced steganographic (stego) algorithms with error detection and correction techniques (BCH syndrome codes) have been designed. This allows us to (a) detect and recover any loss from the hidden confidential information without privacy disclosure, and (b) remedy the received normal readings by employing the corrected version of the secret hidden data. To randomise hiding process, minimise the distortion and boost the detection or recovery, a three-dimensional (3D) wavelet is used to decompose normal IoTs readings into a set of coefficients. To strengthen the security, a key is utilised to generate a randomly selected 3D order used in the hiding process. To accurately measure the detection and recovery capabilities, random noise levels that mimic the real world scenario in a wireless environment are applied to the transmitted readings. The recovered sensitive information and stego readings are extensively measured using BER, PRD and RMS. It was clear from the experiments that our technique has robust recovery capabilities (i.e. BER =0, PRD $< 1\%$ and RMS $< 0.01\%$). This was achieved without increasing nor changing the form of the transmitted IoTs protected readings.

- **Hidden information preserving IoTs streams size reduction without privacy and authenticity disclosure**

  Two novel lossless compression algorithms of IoTs streams were introduced. These algorithms were created to ensure the reduction of the volume of protected readings at intermediate hops without disclosing hidden secrets. The first Gaussian-based model target is representing IoTs sensor readings in a few parameters regardless of the irregularity in the signal. This is successfully

accomplished by employing Gaussian approximation. The margin between the approximated and the actual waveform is measured. In other words, the compression will be for margin space only rather than the entire stream of waveform readings. The margin space values are finally reduced. The second model target (N-Split) is minimising the randomness in the IoTs streams into a smaller, finite field to enhance duplications and avoid the floating operations round errors issues. After a thorough evaluation, under the same conditions, both our techniques were superior to existing models mathematically, meaning the entropy was halved and empirically, meaning the achieved ratio was 3.8:1 to 4.5:1.

- **Cloud-based hidden features preserving IoTs compressed streams size re-reduction**

  This research answers the question: can the compressed, multiple incoming readings of protected IoTs sensors be re-compressed? The answer was yes by pre-processing the compressed streams in such a way that improves the theoretical entropy and exploits it. This is successfully achieved by using similarity measurement, such as using K-means clustering, to classify the compressed streams into subgroups. The streams in every resultant subgroup have been interleaved followed by the first derivative to minimise the values and increase the redundancy. After that, two-steps rotations were applied to rearrange the readings in a consecutive format before applying dynamic run length. Finally, entropy coding is performed. Both mathematical and empirical experiments proved a significant improvement in the entropy of the compressed streams (almost reduced by half) and the resultant compression ratio is more than 50%. To the best of our knowledge, there is no other work that tackles this issue in the IoTs smart devices streams field.

6.3 Key Findings

This thesis discovers several key findings that can be summarised in the following manner:

1. The continuity of the IoTs sensor streams represents a crucial feature that renders them appropriate to be a credible host for other information. Therefore, IoTs continuous smart device (e.g. meter or sensor) streams have been employed as a host for sensitive information. It has also been proven from the findings that the resultant protected IoTs streams can still be directly utilised in their preserved form without revealing the hidden information contained within them.

2. Signal processing techniques, such as the Fast Walsh-Hadamard Transform and Discrete Wavelet, showed very promising results in terms of boosting the hiding randomisation and reducing the distortion impact. This effect can be minimised to almost zero when a few crucial frequency domain coefficients, that contain most of the signal features to be rebuilt accurately, were avoided.

3. The integration of BCH syndrome codes with advanced steganographic algorithms revealed a new discovery (see Section 3.5). By combining error detection and correction techniques with most of the previously proposed steganography algorithms, that used the widely-known hiding positions - Least Significant Bits (LSB), failed to recover the corrupted hidden bits. Therefore, the hiding positions and coefficients are chosen carefully to achieve the best BER (the accuracy in the recovery of the hidden secret information) and PRD/RMS (the remedy precision of the normal readings).

4. Curve-fitting functions are very flexible mathematical models that can be used to represent the IoTs streams in few parameters. Therefore, the Gaussian approximation function is employed in this thesis to approximate the signal. However, to avoid losing any signal features, the margin between the original and the approximated signal should be calculated and further used.

5. This thesis proves that is possible to re-compress multi-incoming IoTs protected compressed streams by identifying their similarity and categorise them into subgroups which then are treated separately.

6.4 Limitations and Future Work

Several problems have been solved in this thesis to bridge the gap between privacy, authenticity, integrity and data management of IoTs sensor streams in cloud environment. However, introduced solutions are just representing a few steps towards the ultimate stable solution in this field. Therefore, to facilitate further developments on top of our solutions, this section concludes with few limitations of the proposed techniques and highlights several suggestions for future research work.

1. In Chapter 2, steganography was used as an underlying layer to ensure privacy and authenticity while not disrupting direct operations on the IoTs streams. The authentication issue (once somebody knows where you hide, they can just retrieve it) in setganography has been solved by two mathematical models that use a secret key to generate random scrambled matrix orders for hiding. Two end-points should share this secret key. However, there are two issues that need be solved in order to render this solution as more robust: (a) in the cases where this key was updated, how can this secret key be shared over an insecure channel in the IoTs field? and (b) in the cases where more than one legitimate party wants to access the hidden information at the operation centres, how can an access control (based on roles) model be developed around the steganographic technique to control who can access what?

2. In Chapter 3, a hybrid model of BCH syndrome codes with advance steganography has been developed to prove that even though the transmitted bits through the channel are equal to its capacity and without changing the form of transmitted streams, the detection and correction is still possible. However, the selection of hiding positions was made based on the obtained Bit Error Rate

(BER) after applying various noise levels onto the transmitted streams. Future researchers may consider to address the question: can our achieved BER be improved by employing other error detection and correction schemes that are not relying on polynomials and Galois fields?

3. In Chapter 4, Gaussian approximation function was employed to represent the IoTs streams by a few parameters. The drawback is that this function is using Trust Region approximation to evaluate its parameters for every chosen IoTs stream block that causes extra process overhead. Therefore, the question we pose is how can Gaussian parameters be predetermined or at least pre-calculated to work on various IoTs streams blocks?

4. Continuing with Chapter 4, the IoTs protected streams were compressed at the streams origin without losing any bits or hidden features. However, the streams should be decompressed without privacy disclosure at the cloud or at operation centres for it to be used for analysis. This leads us to another developmental question: can we learn anything from the compressed protected IoTs streams without decompression? The answer to this question could be revolutionary in the IoTs field. This is because huge amounts of processing power and time will be saved in the case that these streams can be somehow meaningful in their compressed format.

5. In Chapter 5, a re-compression of already compressed IoTs protected streams proved to be possible by identifying block similarities among the IoTs compressed streams and to treat them separately. However, in our model we relied on the compressed streams from its origin using our Gaussian-based algorithm from Chapter 4. In other words, all the streams were compressed using similar compressors at the source. This raises another question for further investigation: can the IoTs streams that were compressed use different mechanisms at the source still be re-compressed?

To conclude, being involved in research that created algorithms to boost privacy,

authenticity and efficiency of high Internet of Things streams in a cloud environment was extremely exciting - we discovered something innovative and revolutionary that it will be interesting to see where further studies can take this work. However, our research represents just a few steps towards a reliable and efficient IoTs applications in cloud environment. This journey is still long because of many other issues including but not limited to the highlighted limitations that should be addressed.

Bibliography

[1] A Ibaida and I Khalil. Wavelet based ecg steganography for protecting patient confidential information in point-of-care systems. *IEEE transactions on bio-medical engineering*, 60:3322–3330, 2013.

[2] Alsharif Abuadbba and Ibrahim Khalil. Wavelet based steganographic technique to protect household confidential information and seal the transmitted smart grid readings. *Information Systems (2014).*, 2014.

[3] Alsharif Abuadbba, Ibrahim Khalil, and Mohammed Atiquzzaman. Robust privacy preservation and authenticity of the collected data in cognitive radio networkwalsh–hadamard based steganographic approach. *Pervasive and Mobile Computing (2015).*, 2015.

[4] Shen Wang, Jianzhi Sang, Xianhua Song, and Xiamu Niu. Least significant qubit (lsqb) information hiding algorithm for quantum image. *Measurement*, 73:352–359, 2015.

[5] Dahai Zhang, Yanqiu Bi, and Jianguo Zhao. A new data compression algorithm for power quality online monitoring. In *Sustainable Power Generation and Supply, 2009. SUPERGEN'09. International Conference on*, pages 1–4. IEEE, 2009.

[6] Andreas Unterweger and Dominik Engel. Resumable load data compression in smart grids. *IEEE Transactions on Smart Grid*, 6(2):919–929, 2015.

[7] Ömer Nezih Gerek and Dogan Gökhan Ece. Compression of power quality event data using 2d representation. *Electric Power Systems Research*, 78(6):1047–1052, 2008.

[8] Joseph Euzebe Tate. Preprocessing and golomb -rice encoding for lossless compression of phasor angle data. *IEEE Transactions on Smart Grid*, 7(2):718–729, March 2016.

[9] Jan Kraus, Tomas Tobiska, and Viktor Bubla. Loooseless encodings and compression algorithms applied on power quality datasets. In *Electricity Distribution-Part 1, 2009. CIRED 2009. 20th International Conference and Exhibition on*, pages 1–4. IET, 2009.

[10] C. Stevenson, G. Chouinard, Zhongding Lei, Wendong Hu, S.J. Shellhammer, and W. Caldwell. Ieee 802.22: The first cognitive radio wireless regional area network standard. *Communications Magazine, IEEE*, 47(1):130–138, January 2009.

[11] Jennifer Yick, Biswanath Mukherjee, and Dipak Ghosal. Wireless sensor network survey. *Computer networks*, 52(12):2292–2330, 2008.

[12] Cisco. Cisco visual networking index complete forecast update, 2015-2020, 2016.

[13] Na Li, Nan Zhang, Sajal K Das, and Bhavani Thuraisingham. Privacy preservation in wireless sensor networks: A state-of-the-art survey. *Ad Hoc Networks*, 7(8):1501–1514, 2009.

[14] Intel. Big data analytics: Intels 2013 it manager survey on how organizations are using big data, 2013.

[15] Ashwin Machanavajjhala, Daniel Kifer, Johannes Gehrke, and Muthuramakrishnan Venkitasubramaniam. l-diversity: Privacy beyond k-anonymity. *ACM Transactions on Knowledge Discovery from Data (TKDD)*, 1(1):3, 2007.

[16] Feifei Li, Jimeng Sun, Spiros Papadimitriou, George A Mihaila, and Ioana Stanoi. Hiding in the crowd: Privacy preservation on evolving streams through correlation tracking. In *2007 IEEE 23rd International Conference on Data Engineering*, pages 686–695. IEEE, 2007.

[17] Khosrow Moslehi and Ranjit Kumar. A reliability perspective of the smart grid. *Smart Grid, IEEE Transactions on*, 1(1):57–64, 2010.

[18] Anthony R Metke and Randy L Ekl. Security technology for smart grid networks. *Smart Grid, IEEE Transactions on*, 1(1):99–107, 2010.

[19] National Institute of Standards and US Technology (NIST). Guidelines for smart grid cybersecurity: Smart grid cybersecurity strategy, architecture, and high-level requirements. http://csrc.nist.gov/publications/, 2013.

[20] Business Department of State Development and Australia Innovation (DSDBI), Victoria. Smart meter privacy and security. www.smartmeters.vic.gov.au/privacy, 2014.

[21] Aqeel Raza Syed and Kok-Lim Alvin Yau. On cognitive radio-based wireless body area networks for medical applications. In *Computational Intelligence in Healthcare and e-health (CICARE), 2013 IEEE Symposium on*, pages 51–57. IEEE, 2013.

[22] Wei Zhang, Yonghe Liu, Sajal K Das, and Pradip De. Secure data aggregation in wireless sensor networks: a watermark based authentication supportive approach. *Pervasive and Mobile Computing*, 4(5):658–680, 2008.

[23] Ruilong Deng, Jiming Chen, Xianghui Cao, Yan Zhang, S. Maharjan, and S. Gjessing. Sensing-performance tradeoff in cognitive radio enabled smart grid. *Smart Grid, IEEE Transactions on*, 4(1):302–310, March 2013.

[24] Thomas A Berson, Bryan Olson, Michael E Fein, Paul D Mannheimer, Charles E Porges, and David Schloemer. Sensor with signature of data relating to sensor, March 16 2004. US Patent 6,708,049.

[25] Ajay Mahimkar and Theodore S Rappaport. Securedav: A secure data aggregation and verification protocol for sensor networks. In *Global Telecommunications Conference, 2004. GLOBECOM'04. IEEE*, volume 4, pages 2175–2179. IEEE, 2004.

[26] H Ozgur Sanli, Suat Ozdemir, and Hasan Cam. Srda: secure reference-based data aggregation protocol for wireless sensor networks. In *Vehicular Technology Conference, 2004. VTC2004-Fall. 2004 IEEE 60th*, volume 7, pages 4650–4654. IEEE, 2004.

[27] Hasan Çam, Suat Özdemir, Prashant Nair, Devasenapathy Muthuavinashiappan, and H Ozgur Sanli. Energy-efficient secure pattern based data aggregation for wireless sensor networks. *Computer Communications*, 29(4):446–455, 2006.

[28] Suat Ozdemir. Secure and reliable data aggregation for wireless sensor networks. In *Ubiquitous Computing Systems*, pages 102–109. Springer, 2007.

[29] Yi Yang, Xinran Wang, Sencun Zhu, and Guohong Cao. Sdap: A secure hop-by-hop data aggregation protocol for sensor networks. *ACM Transactions on Information and System Security (TISSEC)*, 11(4):18, 2008.

[30] Michel P Tcheou, Lisandro Lovisolo, Moises V Ribeiro, Eduardo AB da Silva, Marco AM Rodrigues, João MT Romano, and Paulo SR Diniz. The compression of electric signal waveforms for smart grids: state of the art and future trends. *Smart Grid, IEEE Transactions on*, 5(1):291–302, 2014.

[31] Surya Santoso, Edward J Powers, and WM Grady. Power quality disturbance data compression using wavelet transform methods. *Power Delivery, IEEE Transactions on*, 12(3):1250–1257, 1997.

[32] Michel P Tcheou, Lisandro Lovisolo, Eduardo AB da Silva, Marco AM Rodrigues, and Paulo SR Diniz. Optimum rate-distortion dictionary selection for compression of atomic decompositions of electric disturbance signals. *Signal Processing Letters, IEEE*, 14(2):81–84, 2007.

[33] Jiaxin Ning, Jianhui Wang, Wenzhong Gao, and Cong Liu. A wavelet-based data compression technique for smart grid. *Smart Grid, IEEE Transactions on*, 2(1):212–218, 2011.

[34] Suat Ozdemir and Yang Xiao. Secure data aggregation in wireless sensor networks: A comprehensive overview. *Computer Networks*, 53(12):2022–2037, 2009.

[35] Alexandros G Fragkiadakis, Elias Z Tragos, and Ioannis G Askoxylakis. A survey on security threats and detection techniques in cognitive radio networks. *Communications Surveys & Tutorials, IEEE*, 15(1):428–445, 2013.

[36] Joseph Mitola and Gerald Q Maguire Jr. Cognitive radio: making software radios more personal. *Personal Communications, IEEE*, 6(4):13–18, 1999.

[37] XGWG DARPA. The xg architectural framework v1. 0. Technical report, tech. rep., DARPA, 2003.

[38] DARPA XG Working Group et al. The xg vision. request for comments, bbn technologies, cambridge, ma. Technical report, USA, Tech. Rep. Version 2.0, 2004.

[39] Amjad Soomro and Dave Cavalcanti. Opportunities and challenges in using wpan and wlan technologies in medical environments [accepted from open call]. *Communications Magazine, IEEE*, 45(2):114–122, 2007.

[40] Phond Phunchongharn, Ekram Hossain, Dusit Niyato, and Sergio Camorlinga. A cognitive radio system for e-health applications in a hospital environment. *Wireless Communications, IEEE*, 17(1):20–28, 2010.

[41] Liviu Constantinescu, Jinman Kim, and David Dagan Feng. Sparkmed: A framework for dynamic integration of multimedia medical data into distributed m-health systems. *Information Technology in Biomedicine, IEEE Transactions on*, 16(1):40–52, 2012.

[42] Jack L Burbank. Security in cognitive radio networks: The required evolution in approaches to wireless network security. In *Cognitive Radio Oriented Wireless Networks and Communications, 2008. CrownCom 2008. 3rd International Conference on*, pages 1–7. IEEE, 2008.

[43] Kui Wu, Dennis Dreef, Bo Sun, and Yang Xiao. Secure data aggregation without persistent cryptographic operations in wireless sensor networks. *Ad Hoc Networks*, 5(1):100–111, 2007.

[44] Claude Castelluccia, Einar Mykletun, and Gene Tsudik. Efficient aggregation of encrypted data in wireless sensor networks. In *Mobile and Ubiquitous Systems: Networking and Services, 2005. MobiQuitous 2005. The Second Annual International Conference on*, pages 109–117. IEEE, 2005.

[45] Suat Ozdemir. Concealed data aggregation in heterogeneous sensor networks using privacy homomorphism. In *Pervasive Services, IEEE International Conference on*, pages 165–168. IEEE, 2007.

[46] Soufiene Ben Othman, Abdelbasset Trad, Habib Youssef, and Hani Alzaid. Secure data aggregation with mac authentication in wireless sensor networks. In *Trust, Security and Privacy in Computing and Communications (TrustCom), 2013 12th IEEE International Conference on*, pages 188–195. IEEE, 2013.

[47] Jessica Fang and Miodrag Potkonjak. Real-time watermarking techniques for sensor networks. In *Electronic Imaging 2003*, pages 391–402. International Society for Optics and Photonics, 2003.

[48] Minghua Chen, Yun He, and Reginald L Lagendijk. A fragile watermark error detection scheme for wireless video communications. *Multimedia, IEEE Transactions on*, 7(2):201–211, 2005.

[49] Julia Albath and Sanjay Madria. Practical algorithm for data security (pads) in wireless sensor networks. In *Proceedings of the 6th ACM international workshop on Data engineering for wireless and mobile access*, pages 9–16. ACM, 2007.

[50] Hussam Juma, Ibrahim Kamel, and Lami Kaya. Watermarking sensor data for protecting the integrity. In *Innovations in Information Technology, 2008. IIT 2008. International Conference on*, pages 598–602. IEEE, 2008.

[51] Hsiang-Cheh Huang and Wai-Chi Fang. Authenticity preservation with histogram-based reversible data hiding and quadtree concepts. *Sensors*, 11(10):9717–9731, 2011.

[52] Bernard J. Fino and V. Ralph Algazi. Unified matrix treatment of the fast walsh-hadamard transform. *IEEE Transactions on Computers*, 25(11):1142–1146, 1976.

[53] Wafa Ben Jaballah, Mohamed Mosbah, Habib Youssef, and Akka Zemmari. Lightweight source authentication mechanisms for group communications in wireless sensor networks. In *Advanced Information Networking and Applications (AINA), 2013 IEEE 27th International Conference on*, pages 598–605. IEEE, 2013.

[54] Ghazanfar Ali Safdar, Salah Albermany, Nauman Aslam, Ali Mansour, and Gregory Epiphaniou. Prevention against threats to self co-existence-a novel authentication protocol for cognitive radio networks. In *Wireless and Mobile Networking Conference (WMNC), 2014 7th IFIP*, pages 1–6. IEEE, 2014.

[55] Suat Ozdemir. Secure data aggregation in wireless sensor networks via homomorphic encryption. *Journal of The Faculty of Engineering and Architecture of Gazi University*, 23(2):365–373, 2008.

[56] Zhengrui Qin, Shanhe Yi, Qun Li, and Dmitry Zamkov. Preserving secondary users' privacy in cognitive radio networks. pages 772–780, 2014.

[57] Li Zhu and Huaqing Mao. An efficient authentication mechanism for cognitive radio networks. In *Power and Energy Engineering Conference (APPEEC), 2011 Asia-Pacific*, pages 1–5. IEEE, 2011.

[58] Munam Ali Shah, Sijing Zhang, Carsten Maple, and Omair Shah. A novel symmetric key cryptographic authentication for cooperative communication in cognitive radio networks. In *Automation and Computing (ICAC), 2013 19th International Conference on*, pages 1–5. IEEE, 2013.

[59] Henning F Harmuth. Applications of walsh functions in communications. *Spectrum, IEEE*, 6(11):82–91, 1969.

[60] ZM Yusuf, SA Abbasi, and ARM Alamoud. A novel complete set of walsh and inverse walsh transforms for signal processing. In *Communication Systems and Network Technologies (CSNT), 2011 International Conference on*, pages 504–509. IEEE, 2011.

[61] Kenneth George Beauchamp, Kenneth George Beauchamp, Kenneth George Beauchamp, and Kenneth George Beauchamp. *Applications of Walsh and Related Functions: With an Introduction to Sequency Theory*. Academic press New York, 1984.

[62] Tom Beer. Walsh transforms. *American Journal of Physics*, 49(5):466–472, 1981.

[63] David Salomon. *Data compression: the complete reference*. Springer, 2004.

[64] Stephane G Mallat. A theory for multiresolution signal decomposition: the wavelet representation. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 11(7):674–693, 1989.

[65] Alexander D Poularikas. *Transforms and applications handbook*. CRC Press, 2010.

[66] David Salomon. *Data compression: the complete reference*. Springer, 2004.

[67] Ibrahim Khalil and Fahim Sufi. Real-time ecg data transmission with wavelet packet decomposition over wireless networks. In *Intelligent Sensors, Sensor Networks and Information Processing, 2008. ISSNIP 2008. International Conference on*, pages 267–272. IEEE, 2008.

[68] Jiaxin Ning, Jianhui Wang, Wenzhong Gao, and Cong Liu. A wavelet-based data compression technique for smart grid. *Smart Grid, IEEE Transactions on*, 2(1):212–218, 2011.

[69] Neil F Johnson, Zoran Duric, and Sushil Jajodia. *Information Hiding: Steganography and Watermarking-Attacks and Countermeasures: Steganography and Watermarking: Attacks and Countermeasures*, volume 1. Springer, 2001.

[70] Effrina Yanti Hamid and Z-I Kawasaki. Wavelet-based data compression of power system disturbances using the minimum description length criterion. *Power Delivery, IEEE Transactions on*, 17(2):460–466, 2002.

[71] Bernard Sklar. *Digital communications*, volume 2. Prentice Hall NJ, 2001.

[72] A Akansu and P Haddad. *Multiresolution signal decomposition: transforms, subbands, and wavelets*. Academic Press, 2000.

[73] P. Bodik et al. Intel berkeley research lab. db.csail.mit.edu/labdata/labdata.html, 2004.

[74] S. Suthaharan et al. Labelled data collection for anomaly detection in wireless sensor networks. In *Intelligent sensors, sensor networks and information processing (ISSNIP), 2010 sixth international conference on*, pages 269–274. IEEE, 2010.

[75] Sean Barker, Aditya Mishra, David Irwin, Emmanuel Cecchet, Prashant Shenoy, and Jeannie Albrecht. Smart*: An open data set and tools for enabling research in sustainable homes. *SustKDD, August*, 2012.

[76] S. Barker P. Bodik. Smart project. http://traces.cs.umass.edu/index.php/Smart/Smart, 2012.

[77] Zekeriya Erkin and Gene Tsudik. Private computation of spatial and temporal power consumption with smart meters. In *Applied Cryptography and Network Security*, pages 561–577. Springer, 2012.

[78] Vehbi C Gungor, Bin Lu, and Gerhard P Hancke. Opportunities and challenges of wireless sensor networks in smart grid. *Industrial Electronics, IEEE Transactions on*, 57(10):3557–3564, 2010.

[79] Vehbi C Güngör, Dilan Sahin, Taskin Kocak, Salih Ergüt, Concettina Buccella, Carlo Cecati, and Gerhard P Hancke. Smart grid technologies: communication technologies and standards. *Industrial informatics, IEEE transactions on*, 7(4):529–539, 2011.

[80] Fangxing Li, Wei Qiao, Hongbin Sun, Hui Wan, Jianhui Wang, Yan Xia, Zhao Xu, and Pei Zhang. Smart transmission grid: Vision and framework. *Smart Grid, IEEE Transactions on*, 1(2):168–177, 2010.

[81] Anjan Bose. Smart transmission grid applications and their supporting infrastructure. *Smart Grid, IEEE Transactions on*, 1(1):11–19, 2010.

[82] Alexandros G Fragkiadakis, Elias Z Tragos, and Ioannis G Askoxylakis. A survey on security threats and detection techniques in cognitive radio networks. *Communications Surveys & Tutorials, IEEE*, 15(1):428–445, 2013.

[83] Xin Kang, Ying-Chang Liang, H.K. Garg, and Lan Zhang. Sensing-based spectrum sharing in cognitive radio networks. *Vehicular Technology, IEEE Transactions on*, 58(8):4649–4654, Oct 2009.

[84] G.A. Shah, V.C. Gungor, and O.B. Akan. A cross-layer qos-aware communication framework in cognitive radio sensor networks for smart grid applications. *Industrial Informatics, IEEE Transactions on*, 9(3):1477–1485, Aug 2013.

[85] Aqeel Raza Syed and Kok-Lim Alvin Yau. On cognitive radio-based wireless body area networks for medical applications. In *Computational Intelligence in Healthcare and e-health (CICARE), 2013 IEEE Symposium on*, pages 51–57. IEEE, 2013.

[86] Y. Chen and H.-S. Oh. A survey of measurement-based spectrum occupancy modeling for cognitive radios. *Communications Surveys Tutorials, IEEE*, PP(99):1–1, 2014.

[87] Yasir Saleem and Mubashir Husain Rehmani. Primary radio user activity models for cognitive radio networks: A survey. *Journal of Network and Computer Applications*, 43:1–16, 2014.

[88] Rong Yu, Yan Zhang, S. Gjessing, Chau Yuen, Shengli Xie, and M. Guizani. Cognitive radio based hierarchical communications infrastructure for smart grid. *Network, IEEE*, 25(5):6–14, September 2011.

[89] Yan Zhang, Rong Yu, M. Nekovee, Yi Liu, Shengli Xie, and S. Gjessing. Cognitive machine-to-machine communications: visions and potentials for the smart grid. *Network, IEEE*, 26(3):6–13, May 2012.

[90] Jingfang Huang, Honggang Wang, Yi Qian, and Chonggang Wang. Priority-based traffic scheduling and utility optimization for cognitive radio communication infrastructure-based smart grid. *Smart Grid, IEEE Transactions on*, 4(1):78–86, March 2013.

[91] A.A. Khan, M.H. Rehmani, and M. Reisslein. Cognitive radio for smart grids: Survey of architectures, spectrum sensing mechanisms, and networking protocols. *Communications Surveys Tutorials, IEEE*, PP(99):1–1, 2015.

[92] R.C. Qiu, Zhen Hu, Zhe Chen, Nan Guo, R. Ranganathan, Shujie Hou, and Gang Zheng. Cognitive radio network for the smart grid: Experimental system architecture, control algorithms, security, and microgrid testbed. *Smart Grid, IEEE Transactions on*, 2(4):724–740, Dec 2011.

[93] Jing Liu, Yang Xiao, Shuhui Li, Wei Liang, and C. L. Philip Chen. Cyber security and privacy issues in smart grids. *Communications Surveys Tutorials, IEEE*, 14(4):981–997, Fourth 2012.

[94] W Cary Huffman and Vera Pless. *Fundamentals of error-correcting codes.* Cambridge university press, 2003.

[95] Todd K Moon. Error correction coding. *Mathematical Methods and Algorithms. Jhon Wiley and Son*, 2005.

[96] William Ryan and Shu Lin. *Channel codes: classical and modern.* Cambridge University Press, 2009.

[97] L Jordanova, L Laskov, and D Dobrev. Influence of bch and ldpc code parameters on the ber characteristic of satellite dvb channels. *Engineering, Technology & Applied Science Research*, 4(1):pp–591, 2013.

[98] Young-Jin Kim, M. Thottan, V. Kolesnikov, and Wonsuck Lee. A secure decentralized data-centric information infrastructure for smart grid. *Communications Magazine, IEEE*, 48(11):58–65, November 2010.

[99] Daojing He, Shing-Chow Chan, Yan Zhang, Mohsen Guizani, Chun Chen, and Jiajun Bu. An enhanced public key infrastructure to secure smart grid wireless communication networks. *Network, IEEE*, 28(1):10–16, 2014.

[100] Rongxing Lu, Xiaohui Liang, Xu Li, Xiaodong Lin, and Xuemin Shen. Eppa: An efficient and privacy-preserving aggregation scheme for secure smart grid communications. *Parallel and Distributed Systems, IEEE Transactions on*, 23(9):1621–1631, Sept 2012.

[101] Jinyue Xia and Yongge Wang. Secure key distribution for the smart grid. *Smart Grid, IEEE Transactions on*, 3(3):1437–1443, Sept 2012.

[102] Hong Liu, Huansheng Ning, Yan Zhang, Qingxu Xiong, and L.T. Yang. Role-dependent privacy preservation for secure v2g networks in the smart grid. *Information Forensics and Security, IEEE Transactions on*, 9(2):208–220, Feb 2014.

[103] William Wesley Peterson and Edward J Weldon. *Error-correcting codes*. MIT press, 1972.

[104] A Enis Cetin and Hayrettin Köymen. Compression of digital biomedical signals. *The Biomedical Engineering Handbook: Second Edition. Joseph D. Bonzino, Ed. CRC Press LLC*, 2000.

[105] Ieee standard for transitions, pulses, and related waveforms. *IEEE Std 181-2011 (Revision of IEEE Std 181-2003)*, pages 1–71, Sept 2011.

[106] Goochul Chung, S. Sridharan, S. Vishwanath, and Chan Soo Hwang. On the capacity of overlay cognitive radios with partial cognition. *Information Theory, IEEE Transactions on*, 58(5):2935–2949, May 2012.

[107] Davide Schipani, Michele Elia, and Joachim Rosenthal. On the decoding complexity of cyclic codes up to the bch bound. In *Information Theory Proceedings (ISIT), 2011 IEEE International Symposium on*, pages 835–839. IEEE, 2011.

[108] D2-21 CIGRE Working Group D2.21, WG. Broadband plc applications. CIGRE, 2008.

[109] Bamidele Adebisi, Albert Treytl, Abdelfatteh Haidine, Alexander Portnoy, Rafi Us Shan, David Lund, Hans Pille, and Bahram Honary. Ip-centric high rate narrowband plc for smart grid applications. *IEEE Communications Magazine*, 49(12):46–54, 2011.

[110] Arvinderpal S Wander, Nils Gura, Hans Eberle, Vipul Gupta, and Sheueling Chang Shantz. Energy analysis of public-key cryptography for wireless sensor networks. In *Third IEEE international conference on pervasive computing and communications*, pages 324–328. IEEE, 2005.

[111] Raj Chhikara. *The Inverse Gaussian Distribution: Theory: Methodology, and Applications*, volume 95. CRC Press, 1988.

[112] F Lorio and F Magnago. Analysis of data compression methods for power quality events. In *Power Engineering Society General Meeting, 2004. IEEE*, pages 504–509. IEEE, 2004.

[113] Jan Kraus, Pavel Štěpán, and Leoš Kukačka. Optimal data compression techniques for smart grid and power quality trend data. In *Harmonics and Quality of Power (ICHQP), 2012 IEEE 15th International Conference on*, pages 707–712. IEEE, 2012.

[114] Martin Ringwelski, Christian Renner, Andreas Reinhardt, Andreas Weigel, and Volker Turau. The hitchhiker's guide to choosing the compression algorithm for your smart meter data. In *Energy Conference and Exhibition (ENERGYCON), 2012 IEEE International*, pages 935–940. IEEE, 2012.

[115] G Panda, PK Dash, AK Pradhan, and SK Meher. Data compression of power quality events using the slantlet transform. *Power Delivery, IEEE Transactions on*, 17(2):662–667, 2002.

[116] SK Meher, AK Pradhan, and G Panda. An integrated data compression scheme for power quality events using spline wavelet and neural network. *Electric power systems research*, 69(2):213–220, 2004.

[117] Shyh-Jier Huang and Ming-Jong Jou. Application of arithmetic coding for electric power disturbance data compression with wavelet packet enhancement. *Power Systems, IEEE Transactions on*, 19(3):1334–1341, 2004.

[118] Jesmin Khan, Sharif Bhuiyan, Gregory Murphy, and Morgan Arline. Embedded zerotree wavelet based data compression for smart grid. In *Industry Applications Society Annual Meeting, 2013 IEEE*, pages 1–8. IEEE, 2013.

[119] CF Norman, John YC Chan, Wing-Hong Lau, Jone TY Poon, and LL Lai. Real-time power-quality monitoring with hybrid sinusoidal and lifting wavelet compression algorithm. *Power Delivery, IEEE Transactions on*, 27(4):1718–1726, 2012.

[120] Julio Cesar Stacchini de Souza, Tatiana Mariano Lessa Assis, and Bikash Chandra Pal. Data compression in smart distribution systems via singular value decomposition. *IEEE Transactions on Smart Grid*, (99):1–1, 2015.

[121] Moiss V Ribeiro, Seop Hyeong Park, Joo Marcos T Romano, and Sanjit K Mitra. A novel mdl-based compression method for power quality applications. *Power Delivery, IEEE Transactions on*, 22(1):27–36, 2007.

[122] Andreas Unterweger, Dominik Engel, and Martin Ringwelski. The effect of data granularity on load data compression. In *DA-CH Conference on Energy Informatics*, pages 69–80. Springer, 2015.

[123] Sae-Young Chung, Thomas J Richardson, and Rüdiger L Urbanke. Analysis of sum-product decoding of low-density parity-check codes using a gaussian approximation. *Information Theory, IEEE Transactions on*, 47(2):657–670, 2001.

[124] M. Burrows and D. J. Wheeler. A block-sorting lossless data compression algorithm. *Digital SRC Research Report*, (124), 1994.

[125] B Ryabko. Data compression by means of a book stack. *Problems of Information Transmission*, 16(4):265–269, 1980.

[126] David Salomon. *Data Compression: The Complete Reference*. Springer-Verlag New York, 2004.

[127] C. E. Shannon. A mathematical theory of communication. *The Bell System Technical Journal*, 27(4):623–656, Oct 1948.

[128] Ieee recommended practice for the transfer of power quality data. *IEEE Std 1159.3-2003*, pages 1–119, 2004.

[129] Sean Barker, Aditya Mishra, David Irwin, Emmanuel Cecchet, Prashant Shenoy, and Jeannie Albrecht. Smart*: An open data set and tools for enabling research in sustainable homes. *SustKDD, August*, 2012.

[130] Sean Barker, Aditya Mishra, David Irwin, Emmanuel Cecchet, Prashant Shenoy, and Jeannie Albrecht. Smart project. http://traces.cs.umass.edu/index.php/Smart/Smart, 2012.

[131] Ross Anderson and Shailendra Fuloria. Who controls the off switch? In *Smart Grid Communications (SmartGridComm), 2010 First IEEE International Conference on*, pages 96–101. IEEE, 2010.

[132] Stuart Lloyd. Least squares quantization in pcm. *IEEE transactions on information theory*, 28(2):129–137, 1982.

[133] David Arthur and Sergei Vassilvitskii. k-means++: The advantages of careful seeding. In *Proceedings of the eighteenth annual ACM-SIAM symposium on Discrete algorithms*, pages 1027–1035. Society for Industrial and Applied Mathematics, 2007.

[134] Peter J Rousseeuw. Silhouettes: a graphical aid to the interpretation and validation of cluster analysis. *Journal of computational and applied mathematics*, 20:53–65, 1987.

Glossary

Aggregator refers to an action performed by a machine that is responsible of collecting the data before transmission to the cloud.

brute-force is a cryptanalytic attack by a trial-and-error of all possible combinations to reveal the meaning of encrypted hidden information.

codeword refers to a new block of bits comprising both the message and generated control bits.

intruder refers to an illegitimate party who tries to reveal the meaning of hidden secret information.

spectrum scarcity a situation in which there are a shortage in the available spectra due to large number of requested channels to be allocated.

stream is a set of readings (decimal) that have been measured by an IoTs device over a short period of time (e.g. minute).

third-party refers to a public cloud service provider who is not the owner of transmitted data.

vector space refers to the one-dimensional space in which the obtained coefficients formed.

white space refers to frequencies assigned to broadcasting services but not in use.