Low Power Wireless Sensor Network

by

Siamak Aram

MS, Sharif University, 2009
BS, Zanjan Azad University, 2006

Dissertation Submitted in Partial Fulfillment

of the Requirements for the Degree of

Doctor of Philosophy

School of Control and Computer Engineering

Professor Eros Pasero

Polytechnic of Turin University

February 2015

Italy - Turin

Abstract

Wireless sensor networks (WSN) take on an invaluable technology in many applications. Their prevalence, however, is threatened by a number of technical difficulties. In particular, the shortage of energy in sensors is a serious problem to which many solutions have been proposed in recent years. This thesis takes this area of research one step further and proposes solutions to better conserve energy in sensors. The research conducted can be divided into two parts. The first part is on the design and development of low-power sensors and communication devices capable of monitoring the environment. In this part of research, we first show how smartphones can be employed as a device to acquire data from low-power sensors. Then, by using the idea of duty cycling, we achieve a significant reduction in power consumption in environmental sensing.

The second part of this research is on the use of data-driven approaches where scholars suggest reducing the amount of required communication so that more energy can be saved in sensors. The main idea is that the components of a sensor, including its radio, can be turned off most of the time without noticeable influence on the judgments made using the sensed data. In fact, the data not sensed when the sensor is powered down can be predicted using the computational intelligence methods. To do so, we employ a multi-layer perceptron to predict missing environmental data on the basis of what is sensed. We also show that the

effectiveness of this technique highly relies on the correlation between the points making the time series of sensed data.

Our experimental results evidence the usefulness of the technique we propose in the second part of this research. Indeed, we train a nonlinear autoregressive network against various datasets of sensed humidity and temperature in different environments. It is then observed that sensors can be powered on intermittently without any significant influence on the desired behavior of the sensor network. By testing on actual data, it is shown that the predictions by the device greatly obviates the need for sensed data during sensors' idle periods and saves over 65 percent of energy. It is also established that, among the solutions already proposed, the data-driven approach is best suited to Wireless Sensor Networks especially environmental sensing.

Dedication

This humble effort is dedicated to my beloved Parents, Nadereh and Fraidoun, whose love, company, efforts and understanding can never be forgotten.

Also I have dedication to my **future child/children** whom I may probably have already had if I did not choose to pursue my PhD studies. Since I may have you later, I hope you will have learnt that in order to have a better life you will need a wise plan, hard work and hope that you can become whoever you want to become in future.

Acknowledgments

Completing a PhD is truly a marathon event, and I would not have been able to complete this journey without the aid and support of countless people over the past recent years.

I am very much indebted to my adviser **Eros Pasero** for his fundamental role in my doctoral work. It would not be possible for me to complete this thesis without his industriousness, great assistance in identifying important scientific aspects, valuable suggestions and exceptional professional and human qualities. At all stages of my research Eros has been not only contributing his feedback as an academic adviser, but also giving me friendly and warm encouragement. Close rapport which me and Eros have established in these years has had a significant positive impact on my life.

I would like to thank the Vice-Chairman for Education at the Tehran Polytechnic University and my Master advisor, **Mehran S. Fallah**, for the substantial influence that his course and impressive expertise have had on my research during my Master education, and for the time and generous guidance which he has been humbly giving me since the day I met him.

I gratefully acknowledge the members of my **Ph.D. committee in DAUIN** for their time and valuable feedback on a preliminary version of this defence and my yearly reports to make my work and thesis more valuable. Also I would like to extend a special thanks to **deans/coordinators of DET and DAUIN** for the help provided at various stages of my education.

I am particularly thankful to **Flavio Canavero**, the Doctoral School Director, for his dedicated attitude and kind attention to the needs of the PhD students.

I would particularly like to acknowledge **Amedeo Troiano**, my colleague who has become my friend in the course of time, for his encouragement and willingness to share his scientific knowledge and experiments. I would also like to mention **Ikramullah Khosa**, who have been a good colleague to me.

Among many other things, I am thankful to **Kostas Karatzas** who granted me a wonderful and valuable opportunity to join the research mission under his auspices at Aristotle University in Thessaloniki, Greece, and for his careful advising throughout the period of my work there.

I am very thankful to **Hesam Tehrani**, my English instructor at the Sharif University in Iran in 2011 and a good friend ever since, for all his help and encouragement.

I owe a great debt of gratitude to my best friend **Kateryna Ianishevska** for her company in this last year.

Moreover, I would like to thank my **friends, family and relatives** from Iran, Italy, and Greece to Australia and United States, all of whom cannot be listed here, for the great times we have shared and their encouragements, love and kindness.

I am deeply thankful to the President of the Politecnico of Torino, **Marco Gilli**, who successfully administers the university in cooperation with his solid and

hardworking team (**from secretory office, vice-rectors to international departments**), and who takes particular care to maintain plausible educational environment for international students. I appreciate his personal kindness and substantial assistance throughout all these years of my education at POLITO.

My last word is for **Turin**, the city that I really loved. Good people and special atmosphere of joy dwelling in Turin made me indeed happy, helping me go through the times of hardship and become more mature, when I was missing my family/friends, Tehran and my country.

Washington DC
February 2015

Financial Support

The first three years of this work that is presented in this thesis was supported by the Reply Company.

Table of Contents

Chapter 1

Introduction

Wireless sensor networks (WSN) have received a great attention in recent years. They are based on multilayered structure of interactive sensor nodes and have a wide variety of applications such as event detection, target tracking, environment sensing, elder people monitoring, and security [1-8].

A WSN is usually made up of a large number of sensors that communicate their sensed information to other nodes. Sensors are often supplied with scarce energy resources. Therefore what is necessary here to WSNs operation is conserving the energy. Moreover, developing methods in order to yield efficient power consumption is of importance in researching this arena. According to previous studies, the power consumption of communication component of a sensor is higher than that of a computational unit and that the sleep state of radio communication identifies the minimum consumption of power [9]. Scholars are currently struggling to increase short lifetime of wireless sensor networks [10, 11, and 12]. The problem, in fact, is rooted in limited energy resources available to sensors. Thus, energy consumption should be made efficient at all layers of sensors' operation. To this end, there are proposals to reduce power consumption at the level of system routines, network protocols, data processing, and even hardware modules. Many approaches were proposed to reduce the power consumption of a sensor network, but the following three main techniques are the most important among them [8,9]:

- Duty cycling,
- Data-driven approaches, and
- Mobility

We have worked on two first approaches in [4, 5, 13, 14, 15, 16 and 17] which will be discussed and demonstrated in this thesis.

Since duty cycling patterns are unaware of data which are gathered from sensor nodes, data-driven approaches are more appropriate to reduce the energy consumption of the WSNs. The microcontroller can switch on the sensors only during the measurement, reducing the power consumption [4, 5]. Yet, sporadic occurrence of unwanted communication as a result of unnecessary data transference is quite possible. Reducing extra communications is a way to save energy which can be followed by data-driven techniques. While energy-efficient data acquisition' schemes are mainly concerned with decreasing power consumption relevant to the sensing subsystem, 'data reduction' schemes focus on unneeded samples.

The current thesis has employed duty cycling along with data prediction, yet profound reduced data was spotted on prediction of data. An innovative method is proposed and tested on simulated and experimental data. A neural algorithm is considered to forecast sensor measurements and their uncertainties to allow the system to reduce communications and transmitted data. Particularly, first a multilayer perceptron (MLP) network [18] then NAR algorithm [19] as Time Series approaches with dramatically high reduction on both data and error were applied.

1. aBluSen , a new data Acquisition System

In 1980s, the first actual headphone did not depict a light or small gadget, and power consumption as one of current challenges even today, caused many hardships for working in Wireless Sensor Networks (WSN). The technological evolution from the first generation of mobile phones has shown swift growth in different aspects. This development started from the very low-end mobile phone category [20] up to smartphones today. Multitasking operating systems, running myriad of applications and having several features (such as Bluetooth) [21] are the attributes that make these types of phones important in daily life and consequently make them one of the most-used electronic devices. Now, they do more than just connecting people to each other, and have various applications, such as health and medical [22], military, environmental, home and office [3], and commercial.

Thanks to the likelihood of computing, analyzing, communicating and monitoring found in the current generations of phones, the quality of life is enhanced by gathering and scrutinizing environmental information. Traditional environmental monitoring systems have been changed by the new features that WSNs offer.

In some applications, wireless sensor networks use sensors placed in pre-planned locations or equipped on carriers or vehicles, with fixed routing strategies; on the other hand, currently, the capabilities of mobile devices, especially smartphones, as data collectors make it possible to distribute sensors, and helps to share information with communities, that is crucial for making decisions for different communities such as governments, groups of people, and researchers [23].

In [4], a smartphone was applied to get environmental information from small spaces, using a Bluetooth transceiver. As shown in Figure 1, this approach was to acquire temperature and humidity values (considering low cost and low power components) using a Bluetooth communication system for the transmission of the acquired data to an android-based smartphone. The system was tested in a climate chamber that changes the environmental conditions. Getting access to environmental information on the move become more smoothly possible upon applying a Bluetooth communicational system. Likewise, Bluetooth system facilities having control and giving warnings in various fields such as medical, social services and agriculture. Thus, Bluetooth is rightfully considered as one of the most important features in all smartphones that enables both sensor and phone to communicate and transfer data at short distances using less power than Wi-Fi [24]. As shown in Figure 2, a new scenario was introduced by applying several sensors with different structure.

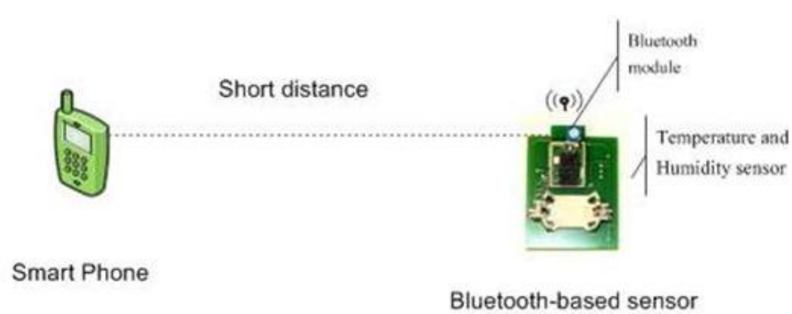

Figure 1. Acquisition of the environmental data using a smartphone by the approach that is described in chapter 3.2.a, direct connection between the smartphone and a sensor.

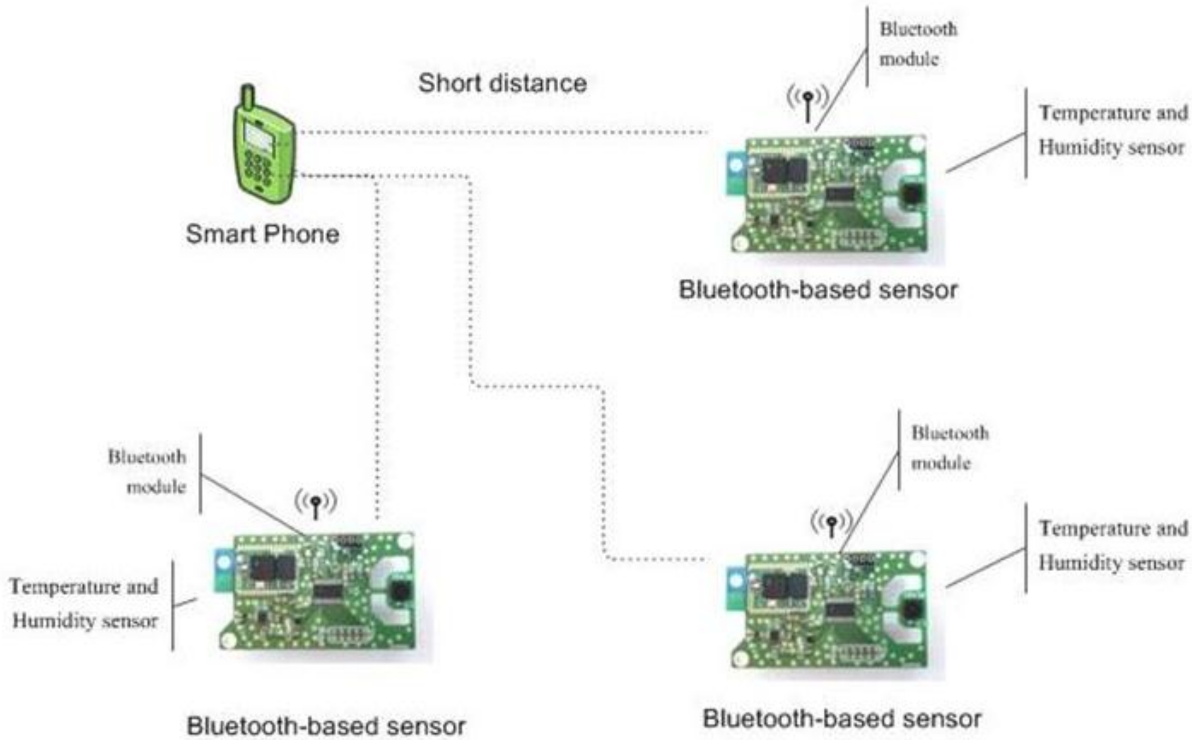

Figure 2. Acquisition of the environmental data using a smartphone by the second approach of this work that is described in chapter 3.2.b, connection between the smartphone and several sensors.

In fact, increasing the number of distributed sensors maximizes the lifetime of the network, since more failures can be tolerated; this tackles the power consumption challenge. Another advantage of applying more sensors at the same time is an increased reliability of the network. The smartphone application manages the sampling frequency and the method of connection between sensors using an independent communication with each sensor, without the need for multi-hop routing [25] to gather environmental information. The data is gathered and analyzed directly by the smartphone application.

In furtherance of our research objectives, the low-power sleeping mode for radio transceiver was switched on at the times when the transceiver had not been engaged into any communicative operation. In theory, it is recommendable to turn down the transceiver upon depletion of data scheduled for transmissions. The work of the transceiver is therefore resumed upon receiving another packet of data. The aforementioned interaction of nodes is scientifically defined as duty cycling. Under "duty cycle" we understand activity of certain fraction of nodes throughout the period of network's performance. Similarly, cooperation between the nodes is achieved by synchronizing their sleeping phases and waking phases. Thus, each duty cycling scheme requires corresponding algorithm or method for planning nodes' passive and active periods. The method of such kind facilitates nodes' changeovers from waking to sleeping and vice versa.

2. Data Driven approach

It is worth noting that planning duty cycles is not contingent on results of previous samplings. With regard to that, energy efficiency could be manifestly improved upon adopting data driven approaches. Energy consumption may be influenced by the following factors [8]:

- Unneeded samples; when data loses its practical validity, its further communication to the sink should be interrupted. Instead, notable correlation of previous samples should be taken into account
- Sensing subsystem's power consumption; when the sensor has low power supply, communicational reductions will be of no effect.

It is as well important to duly handle the sensing processes: unreasonable data sampling results in initiation of unnecessary communications which in the end play no role in the sensing. Such omissions inevitably lead to excessive energy consumption. Another problem tends to emerge in the case when the power expenditure arising out of the operation of sensory system is considerably high and, therefore, cannot be ignored. In this instance, the issue may be resolved with aid of data prediction method which seeks to bring the accuracy of sensing's results into compliance with applicable standards. This objective can be met upon cutting down volumes of data due for sampling.

3. Time Series approaches

In cases when dynamics of the system is supposed to be nonlinear it is advisable to extend the scope of time series prediction to neural networks. So far, time series prediction method has proved to be thoroughly compliant with neural networks. For the first time, application of multi-layer perception neural networks for performing nonlinear time series predictions was described by the scientists in [26]. It is supposed that neural networks somewhat replicate the brain of a human, in particular with regard to their flexibility and extraordinary efficiency. In contrast to multi-layer perception neural networks, ordinary neural networks have limited signal processing capacity, i.e. low quality of the signals may manifestly impede processing. In the following chapters, especially chapter 3, MLP and practical aspects of its implementation will be expressly explained. In this work, we deem it necessary to provide elaborate description of time series approach under research due to the importance and wide applicability of these methods.

Today, time series analysis is applied to the wide range of sciences including biology, physics, economy and technology. Technically, time series is represented as an ordered set of vectors which are determined as per the formula given below [27]:

$$y\ (t);\ t = 0\ ,1\ ,2 \tag{1}$$

Practical aspects of time series forecasting can be found in a variety of related scientific articles and publications. Any time series forecast requires preliminary choice of the prediction strategy. This choice should be made with due regard to the end objectives of the time series prediction i.e. facilitating production and activities. So far, time series method has proven useful for many areas. Specifically, time series forecasts are often required to solve problems in medical, econometric and engineering field. Accuracy of predictions are occasionally impeded by the chaotic behavior of time series. Hence the need to determine the exact state of the analyzed domain in the beginning of experiment. Time series forecasts are mainly conducted relying on AR (AutoRegressive), ARMA (AutoRegressive–Moving-Average) and MA (Moving-Average) models. Nonetheless, neither of these linear models are suitable for application to non-linear signals. Assuming that, the scientists found an alternative to linear module predictions. It was established that ANNs (Artificial Neural Networks) could assist predicting time series with better accuracy. As a computational structure, ANN incorporates models based on biological patterns. Another option is NN which exploits its non-linear constituent elements in order to select the most accurate data hypothesis. These integral elements are joint by links with weights containing all data formulated in the course of forecasting. NN's exceptional capacity for aggravation and approximation operations gives it an advantage over other networks in terms of predictions. Consequently, manufacturers have already devised a number of neural network-based tools for generating statistics and modeling. Owing to the properties of NN referred to above, neural network approaches have recently become highly applicable for time series forecasting. Accordingly, there are two major points which should be taken into consideration while performing neural network based data readings [27]:

a. Intervals of data sampling
b. Sequence of points where sample data will be collected.

It is supposed these matters demand empirical solution. For the aims of this paper, we facilitated selecting appropriate intervals and data sampled points by creating a new algorithm. The aforementioned algorithm was developed upon comparison and analysis of data collected from a number of sources and in variable circumstances. The optimized algorithm proposed by us was devised with regard to the principle of lowering power consumption. This goal was achieved on account of reducing the volumes of data involved in the sampling process. In this paper, we concentrate on data prediction to conserve energy in wireless sensor networks. To do so, sensed data is thought of as making a time series [28] where there may be a correlation between the points. This fact serves as the rationale behind our method and implies that sensors could be powered down in judiciously chosen time intervals. The correlation among the points would then allow prediction of sensed data during sensors' idle periods. To bring functionality to this idea, we make use of neural networks from NAR (Non-linear AutoRegressive) model [29]. In accordance with the proposed method, prediction of data was performed via non-linear autoregressive network. Further on, this method was affirmed by conducting a series of empirical experiments conducted within the frame of the research at hand. Sets of data collected specially for the experimental part of the research included humidity and temperature parameters from real sources. Performance of neural networks in differentiated circumstances was observed and assessed by us. In order to meet the objectives of the study the neural network was fed by two delayed targets. As a result, the size of the network's hidden layer was gradually altered. The best prediction outcome was achieved upon feeding the network 20 hidden layers' neurons. In the parallel series of experiments the quantity of data inputs was varied. Notwithstanding the increase in the input, the error rate had no sufficient decrease. Therefore, it was proven experimentally that the method tested during this study allows decreasing energy costs in WSNs. The results show that the proposed method is very effective and saves over 65 percent of the energy while preserving the qualitative characteristics of the test data. Our method is then evaluated empirically through the experiments we conduct as part of this research. As said earlier, datasets employed within the frame of the experiments contain actual sensed humidity and temperature in different environments.

In our experiments, we do not employ a round robin scheduler to schedule sample acquisition in sensors. Instead, a central control unit schedules sample acquisition in a nondeterministic manner. Thus, it is not required to perform any supportive transmissions within the interval between any two sample acquisitions in a sensor. It is also shown that there is no need for extensive knowledge of the deployment domain so that optimizing the neural network's parameters can merely be accomplished on the basis of residual power volumes and sleep state periods.

4. Outline of the thesis

In this thesis, firstly, we focus on two major phases of the work. Accordingly, each phase has two parts. In the first phase dedicated to duty cycling, a new device for the acquisition of environmental parameters with consideration of low power consumption was developed. Then, the acquisition system was improved by creating possibility of acquiring data from a number of sensors at the same time and thus achieving higher amount of saving in power consumption. Respectively, this approach is elaborated in each section.

Chapter 2 provides the reader with an overview of the notions, approaches and methods that will be valuable to make clear the work that is accomplished in the thesis. Each phase and step of this research is covered by related works that have been followed by other scholars. The chapter is divided into four main parts. The first three parts deal with the domain of wireless sensor networks, environmental sensing and mobiles and duty cycling followed by an overview of the main networking issues. Time series approaches related works are covered by the last part in this chapter. Chapter 2 mainly reviews the different modeling approaches which have been investigated in the literature to address the issue of power consumption in Wireless sensor networks.

The research methods are explained in Section 3 in four parts. The main part of this section is elaborated in Subsection 3. a and b. It acquaints the reader with the domain of Multilayer perceptron and nonlinear autoregressive, and covers the methods regarding reducing the amount of communication employed by each of these models.

Experimental results are explained in Section 4 based on the main phases of the research. Section 5 discusses the method and the results obtained from the experiments. Section 6 concludes the thesis.

Chapter 2

Literature Review

1. Environment and Sensing

The number of people who tend to share and record data through the Internet has widely increased in recent years. This similar tendency has required researchers to analyze these distributed data. Many scientists and researchers encounter with the explosion of information through blogs and social networks, therefore they persuaded to investigate on this phenomenon. . Due to the advent of sharing the data, new styles of life have been developed that influence different aspects of life such as business, government and politics, commerce, public discourse, health and medical, etc. Besides this phenomenon generates considerable changes on related technologies.

In recent decades, scientists consider the environmental information as an important aspect since it affects people's real life. Due to the importance of environmental influences, changes and interactions of information on our life, these elements considered operationally valuable to define the environmental this studying and indicates the limitations that exist regarding the knowledge and uncertainties [30].

In this section, a general and practical definition of mobile environmental sensing is described to facilitate collecting and analyzing data to obtain environmental information. However, wireless sensor networks make the following environmental information available regarding temperature, humidity, pressure, solar radiation. This chapter deals with the effectiveness of the smartphones, which are similar to the sensors, to gather the environmental data from distance. Furthermore, this technology has the potential development that many groups of people are participating to gather data process. This feature is identified as participatory sensing that allows to perceive the parameters of environment by mobile phones. Some mobile phones are equipped with built –sensors and help to sense and interpret the condition of environment.

In general, the term sensing is defined as an ability such as sight, hearing, smell, taste, or touch which is used from the outside of body [30]. Generally, “Sensing” and “Remote sensing” technology refer to detection or perception of a phenomenon via awareness of perceiving the human’s external objects. It is recognized as a simulated ability to detect and perceive the phenomena. According to Hussein [31] definition:

“Gathering of information regarding an object without having physical contact is named as remote sensing technique which is used to collect information. It is considered to analyze the digital information quantitatively. ”.

Likewise, Bureau from Africa’s Office of Sustainable Development [32] specified that remote sensing is defined as:

"The science of attainment and interpretation of information remotely, utilizing sensors without coming into physical contact with the observed object."

Data gathered from distance, supplies the information that is not simply accessible. In addition, another significant parameter is the time that facilitates remote sensing systems to analyze the data at the required interval. Various researches and projects applied different systems to sense from distance. Inaccessible information can lead to more achievable data (but not completely) therefore, these systems are favored since they make sensing the environments possible. There are some differences between the method or the way of sensing and the accuracy of information. Whereas, more sensing leads to more information but not necessarily leads to accurate data. Consequently the process of sensing must be improved to lead to the accurate data.

It is necessary to define the Environmental Remote Sensing, which means to predict, analyze, and make decision regarding the environmental condition. Subsequently the concept of environmental information and the process of data gathering are concerned with the ambiguity and complexity, the information can be accessed remotely is applied with different techniques to overcome subsequent problems. The techniques of environmental remote sensing can be extended, to apply various approaches, methods, and technologies to acquire related information and furthermore achieve the accuracy of obtained data. Mobile phones are used as a remote sensing method to access environmental information that enables the systems to access data remotely by means of the most popular devices (especially for small distances). For example, in [4], we presented a method by using smartphone to obtain environmental information for small distances. It is revealed that environmental information can be obtained as moving is easier through controlling and having warning facilities.

2. Mobile phone as a sensor

Power consumption is still considered a challenge from inventing the first mobile phones that were light, small, and smart, in 80s. Mobile phones enable people to connect with each other via text and voice. The manufacturers of mobile phones included more services which is called PDA (Personal Digital Assistant) namely checking email, playing game and sending and receiving files. When characteristics of primitive mobile phones are combined with the PDAs' abilities with more applications are upgraded devices called Smartphones [33]. The process of cell phones evolution is illustrated in Table 1. Three various generations of the cell phones are generally compared in terms of size, shape, equipment and power consumption. The mobile phones technology evolution in various fields beginning with the first generation of mobile phones is revealed in this table. This growth of mobile phones was very fast starting from category of simple low-end mobile phones [20] and out to the smartphones that includes wide range of operations. Several characteristics make these types of phones a part of daily life and identify them as one of most commonly used electronic devices [21]:

- Multitasking operating systems

- Running myriad of applications, and
- Having several features (such as Bluetooth)

Now they are involved in various applications:

- Health and medical [34]
- Military
- Environmental
- Home and office [3]
- Commercial

In reality, these potential attributes of new generation mobile technology such as computing, communicating, analyzing, and monitoring considerably influenced the quality of life.

Currently, the subsequent compound is applied in different fields for example environmental monitoring, business, healthcare, social networks, safety and transport. These equipped devices facilitated various application with respect to different domains along with their operating systems [35]. These new mobile generations, particularly smartphones and tablets, have been using to monitor the parameters of quality of life. The following sensors in these smartphones:

- Accelerometers
- Compasses
- GPSs
- Gyroscope
- Microphones, and

- Cameras

These kinds of sensors are still set in the smartphone [36], however the air quality or the pollutants of the environment are not considered. These sensors are embedded as part of the mobile phones that can enable the internal APIs of them. The microphone, the camera, the GPS modules are sensors which are easily recognized easily by everybody. Some external sensors are utilized besides the embedded sensors; two kinds of external sensors are also used as a prototype in this chapter. Commonly, wireless sensor networks apply sensors as pre-planned sensors which are equipped on carriers or vehicles, with fixed routing plans; otherwise, at the present time, mobile devices used as data collectors, particularly smartphones, play major role as distributed movable sensors. Moreover, they can share information with groups or communities as a key parameter to assist governments, groups of people, and researchers in various communities for making important decisions [23]. As it is mentioned the abilities of smartphones such as computation, communication, and sensing, create the conditions of participatory or opportunistic operations [37], There are two types of sensors in case of mobility; the first type is identified as the wearable sensors, which is worn by people [38]; the second is named as Phone to web [39], that is also known as phone sensors. Some mobile devices such as PDAs and cell phones can be connected to devices to transfer data via Bluetooth. In other usages they acted as memory cards to store data rather than transfer it online. In contrast, the gathered data can periodically or continuously be sent to other devices or center station. The second type of mobile application is to record the information of the environment. The mobile phones are able to record necessary data and afterwards send it quickly to a particular station or upload the data through internet to the web portal [6]. The environmental information can be gathered by users in different circumstances for instance walking, biking, driving, and running; also they can include the location and tagged (i.e. custom) data. Otherwise, it is not possible to apply camera, send text messages, or tag more information in these moving circumstances. Therefore, mobile phones and Web are used to communicate and promote the quality of lifestyle, and to assist elders, their families, and doctors in medical cases [6].

Monitoring the environment can be conducted in different situations. In these situations, personals, groups, and communities have been using wireless sensor networks [36] to obtain environmental information. Nowadays, simplest way to obtain environmental information is to employ the mobile devices. Environmental sensors, such as temperature, humidity, solar radiation, and pressure provide possibility to monitor outdoor and indoor sites simultaneously. However, one of these types of networks' features is that they usually enable transfering the short distance data transferring. In design of any related networks' system, two parameters of low power and low cost are considered essential. Generally, four parameters significantly influence the structures of sensors and subsequently their platforms:

- Deployment (based on activities)
- Location (indoor or outdoor)

- The application
- Data

Two useful parameters of sensing environmental information such as temperature and humidity are employed to control and monitor diverse fields such as medical, social services, and agriculture. Mobile robot temperature sensing is identified as a sample solution to monitor temperature of the airport and hospital [40]. This robot has the potential and capacity to be employed in other application for instance detecting heat temperature for firefighting. Likewise, wireless sensor networks' solutions are used with fixed distributed sensors [41, 42]. Small environment can be monitored via mobile phones rather than using pre-planned or robot sensors systems [43].

This section deals with the probability of the smartphones application for collecting data from other phones or sensors. Currently, two important parameters for monitoring climate condition are identified as temperature and humidity to sense the environmental changes in order to prepare more useful information about living or working places. To this end, it is a novel solution to use distributed devices in various environments equipped with high-resolution sensors and wireless transmission devices to transmit data to smartphones. The Bluetooth which is embedded in all smartphones can be utilized as a tool to transmit data in the place where Wi-Fi connection is not available. Smartphones facilitate communication with other gadgets equipped with the programmable tools with the possibility of having diverse types of applications. They are capable to gather, analyze, and confirm data. In [5], we proposed and improved a prototype to sense Temperature and Humidity by using a number of Bluetooth-based sensors to monitor environmental conditions in the android-based smartphone.

Table 1. The Growth of Mobile Phones

Type	Analog	Digital	Smart Phone
Weight	About 2 lb	About 7 oz.	Less than 5 oz.
Processor	Simple tasks	Preliminary tasks	Advanced tasks such as multimedia, internet communication, playback
Memory	Only for storing some setting and phone numbers	For storing more data in the range of megabytes	More than 32MB with possibility to have extra memory
Bluetooth	-	A few of them	Yes
Battery	For talking short time and for standby time	For talking long time and for standby time	For talking in longer time and for standby time

3. Duty Cycling

As of today, it is common practice for developers to refine on the structure of wireless sensor networks so as the latter would incorporate duty cycling strategies. Duty cycling method assists in mitigating power consumption by regulating the work of the motes. In particular, the essence of the aforesaid method lies in switching on the radio of the mote strictly at the time of its active participation in data processing. Having been introduced into the field quite recently, duty cycling as a method has gained popularity soon. To date, duty researchers work on accomplishment of 0,1% duty cycles, which is rather high-reaching objective [44]. Updates on the duty cyclings usually envisage different modifications on nodes' interaction and its further synchronization as per the hardware and acceptable complexity parameters. Also, varying duty cycling methods are distinguishable by their inherent features, such as, for example, topology, thresholds for network density, fluctuations in increase of delays etc. In [8] duty cycling is categorized through two different and complementary approaches shown in Figure 3.

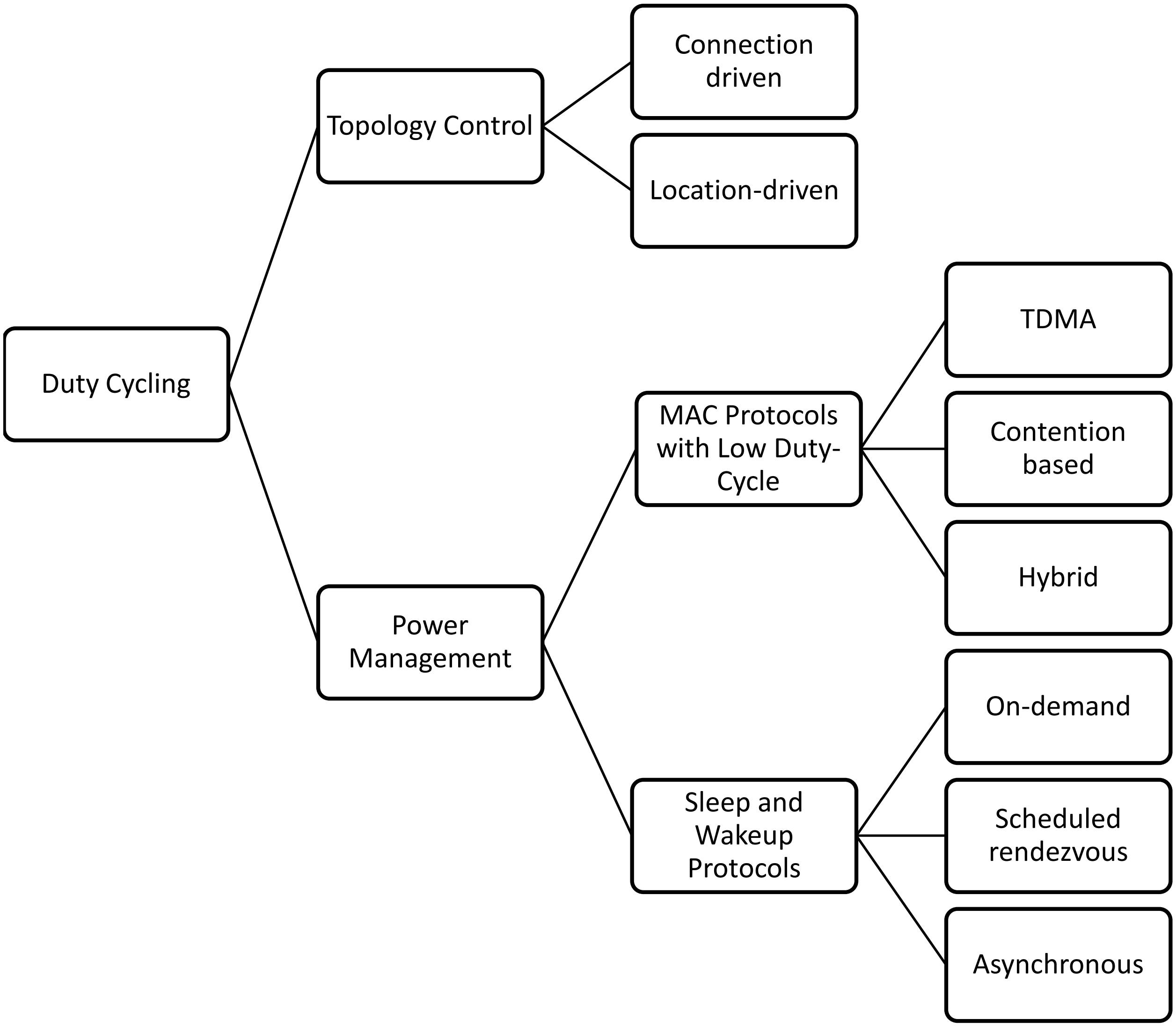

Figure 3. A detailed taxonomy of Duty Cycling schemes considering Energy conservation in wireless sensor networks based on the general taxonomy that is presented in [8].

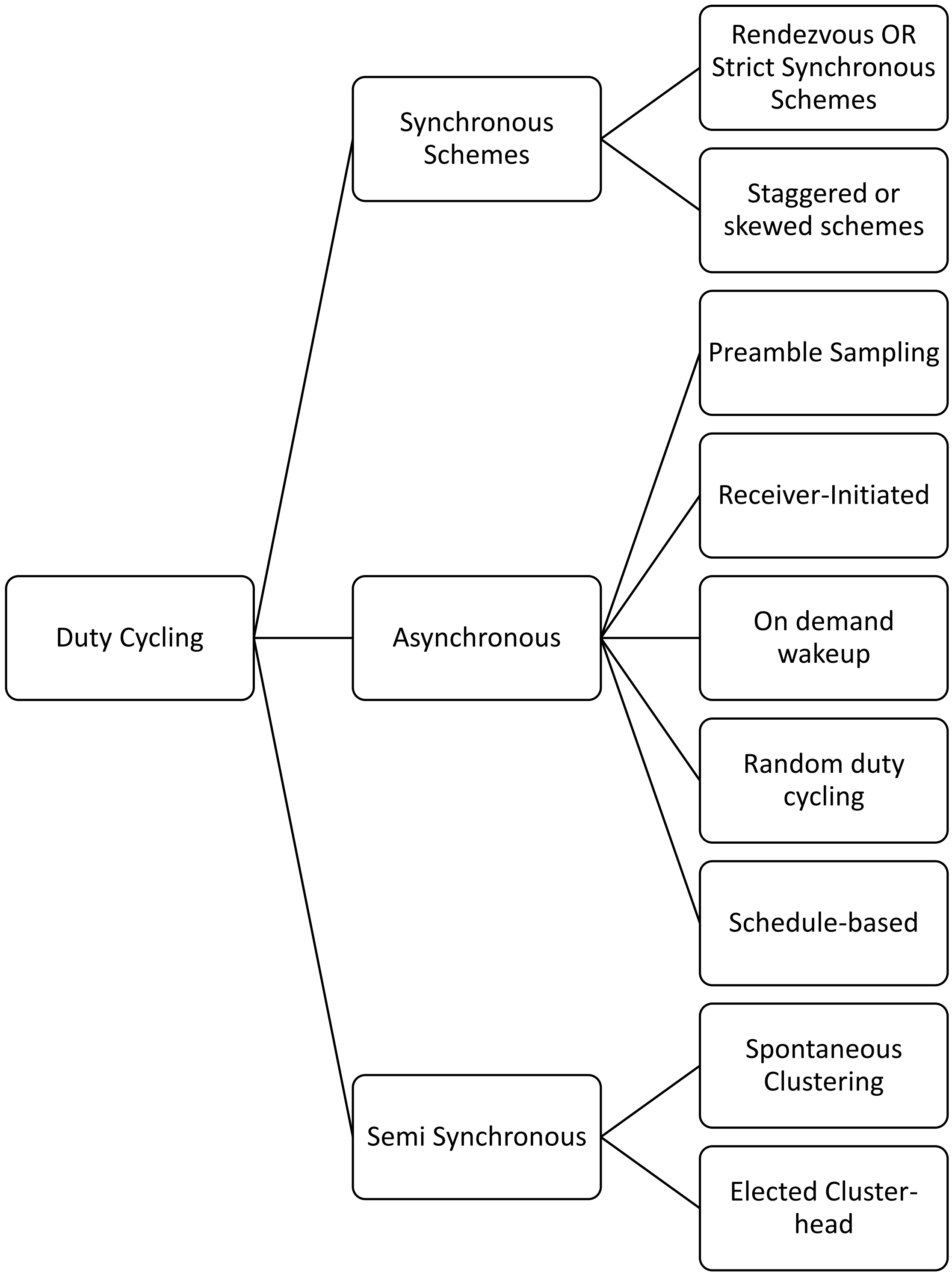

Figure 4. A taxonomy of Duty Cycling techniques in wireless sensor networks based on the general taxonomy that is presented in [44].

Since in [44], as shown in Figure 4 the authors presented other taxonomy.

It is important to know that there is a potential for taking advantage of the redundancy of the nodes. The latter commonly occurs in WSNs: the connectivity is therefore maintained merely by reliance on a small fragment of active nodes, chosen adaptively. Respectively, the sleeping mode can be enabled for the nodes which do not take part in maintenance of connectivity. As opposed to that, other nodes which contribute to stabilizing connectivity are defined as topology control. Topology control is established with the aims of extending the lifetime of the network by two or three points in comparison to the networks which encompass only active nodes [45, 46].

However, the studies show no strong need for keeping the radio of the active nodes switched on at all material times. Instead, in the absence of any activity of the network the nodes, singled out by the topology protocol may continuously make changeovers between the periods of waking and sleeping. Similar the approach was adopted by us in the second step of our research’s first phase. The first survey presented above focuses on the duty cycling processes which involve only active nodes as "management of power'. Power managing and control of topology both depend on duty cycling processes. The granularity of these processes may considerably vary. The type of the network's architecture determines which of the two existent categories the techniques for power management may be fitted in. This category is explained in more details in the second mentioned survey which also includes expanded taxonomy. To be more specific, there are two categories of cycling strategies, i.e. synchronous, asynchronous schemes. As a matter of fact, synchronous and asynchronous methods sufficiently differ as regards the extent of synchronization itself. For instance, asynchronous strategies do not envisage maintaining uniformity of the clock on account of the nodes, whereas synchronous strategies require strict adherence to time limits. Also, distinguishable principles of these two methods allow for creation of a combined method, known as semi-synchronous or semi-asynchronous scheme. This hybrid method is based on the relative simplicity to accomplish pair-wise synchronization rather than attempting more general synchronizing. Figure 4 overviews the taxonomy described above.

Figure 3 demonstrates two possible options for the integration of power management protocols, i.e. first option suggests that such protocols be fully implemented into the MAC protocol, while the second option presupposes integrating the sleeping and waking components of power management protocols at the top layers of MAC protocols. Figure 4 provides for more extensive description of sleep/wakeup protocols.

Total integration of power management protocols into the MAC layers facilitates regulating power expenditures by assisting access to the medium within the duration of active and passive periods, in contrast to that, separate protocols responsible for sleeping and waking phases are more easily adaptable to requirements of particular applications and circumstances. In other words, these protocols have greater flexibility in comparison to top layer protocols, and thus may be used at any level of the MAC protocol.

As it was referred to above, main purpose of developing data cycling method was optimizing the nodes' energy costs and prolonging lifetime for the network. This method seeks to eliminate idle periods of the network's components, for instance, radio continuing its work outside related data transmissions. Though, solving the problem of the idleness of the radio have never been straightforward: it still remains difficult to predict at which moment the mote will receive a new package of data.

Besides, apart from the idle periods attributable to the radio there exist other risks threatening stability of the network. For example, energy supply can be exhausted when the node overhears those frames which do not have any impact on the system's power. The occurrences of wasting power require thorough analysis due to the fact that duty cycling may unwillingly trigger collisions between the nodes, which are likely to boost power expenditures contrary to the goals pursued by the method. Even though duty cycling has gained reputation of as a deeply-rooted method, this area demands profound studies. For instance, one of the developers' nearest objectives is to adopt duty cycles which can retain efficiency by abandoning secondary goals of network performance.

One of the latest developers' innovations is incorporating duty cycling into WSN's architecture, even though it may require further research and upgrading. Ideally, duty cycle introduced to the system should not exceed 1% of the radio's performance [44]. Yet, it is almost inevitable that other targets of the network's activity might be eventually overlooked.

All in all, there exists a wide range of wireless sensor networks, which differ by their architecture, weight, practical applications etc. Following the objectives of this research, it important to explain distinctions between the wireless sensor networks regarding to duty cycling. WSNs, as said earlier, may have a lot of discernible variations based on the categories to which they belong. Within the frame of the research, [44] suggests that the differences between WSNs be analyzed according to the networks' hardware, deployment and application characteristics that all showed in Figures 5, 6 and 7 respectively. It should be noted that the categories listed above happen to have mutual dependence. With regard to the fact that WSNs are usually tailored for predefined application, deployment and mote features of the network should be compatible with requirements dictated by that application.

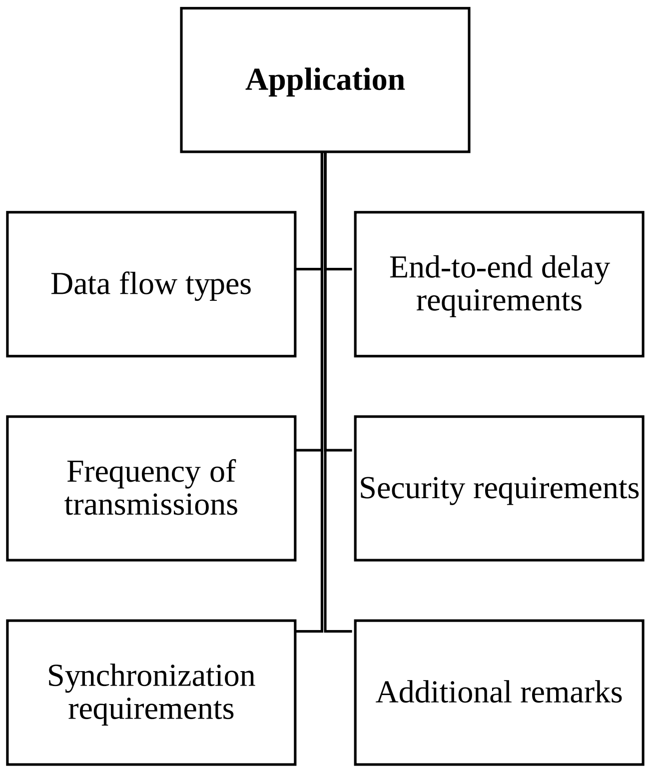

Figure 5. Duty cycling performed for actually existing wireless sensor networks requires considering extensive applicational taxonomy at the stage of the networks' designing. Therefore, apart from establishing the requirements pertinent to the communication of entire network the application will also engage in selecting mote hardware for the sensor.

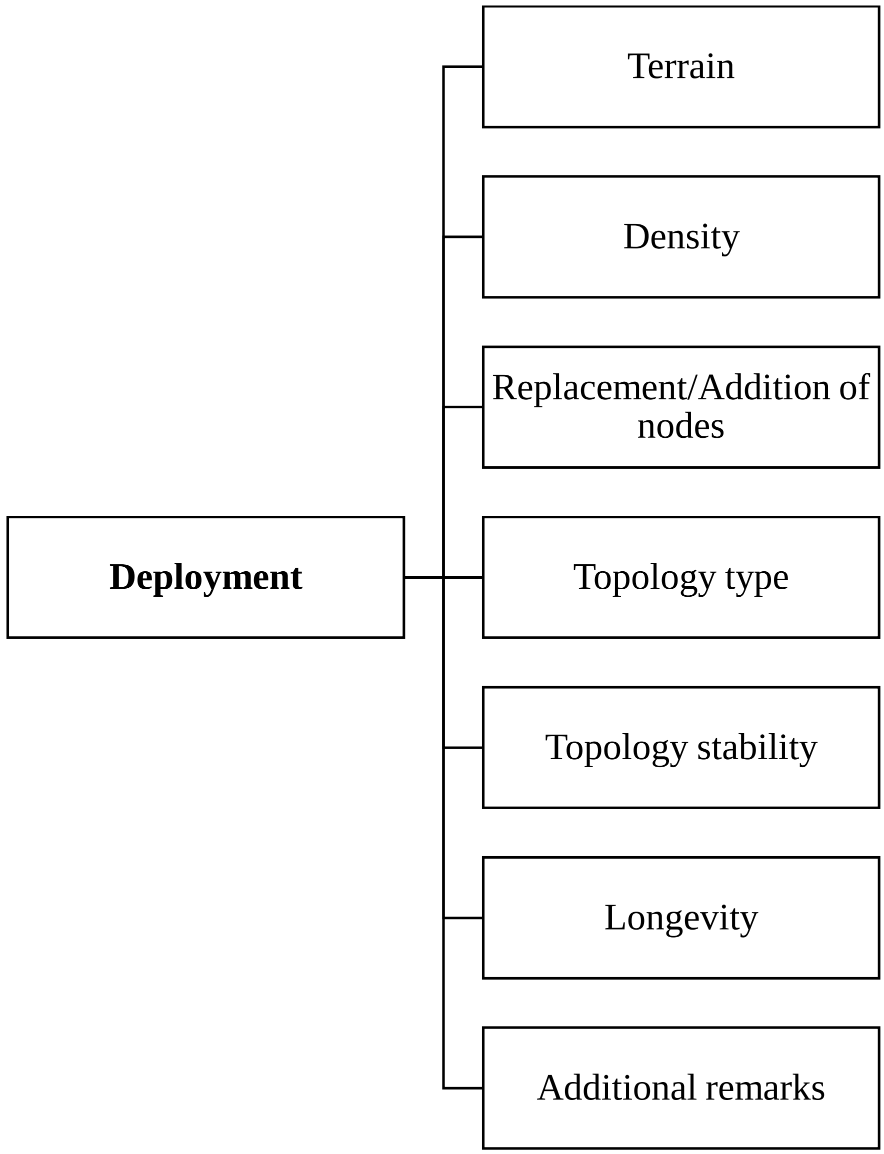

Figure 6. Usually, it is not feasible to consider extended taxonomy of customized hardware, rarely available in the market. Instead, it is advisable for the developer to select more readily-available and convenient technical solutions. Complications arising out of the hardware's incompatibility with the main system threaten to impede the process of duty cycling. Duty cycling technique will not be able to exploit its inherent features such as synchronization, memory, processing, communication abilities in case of hardware-related constraints.

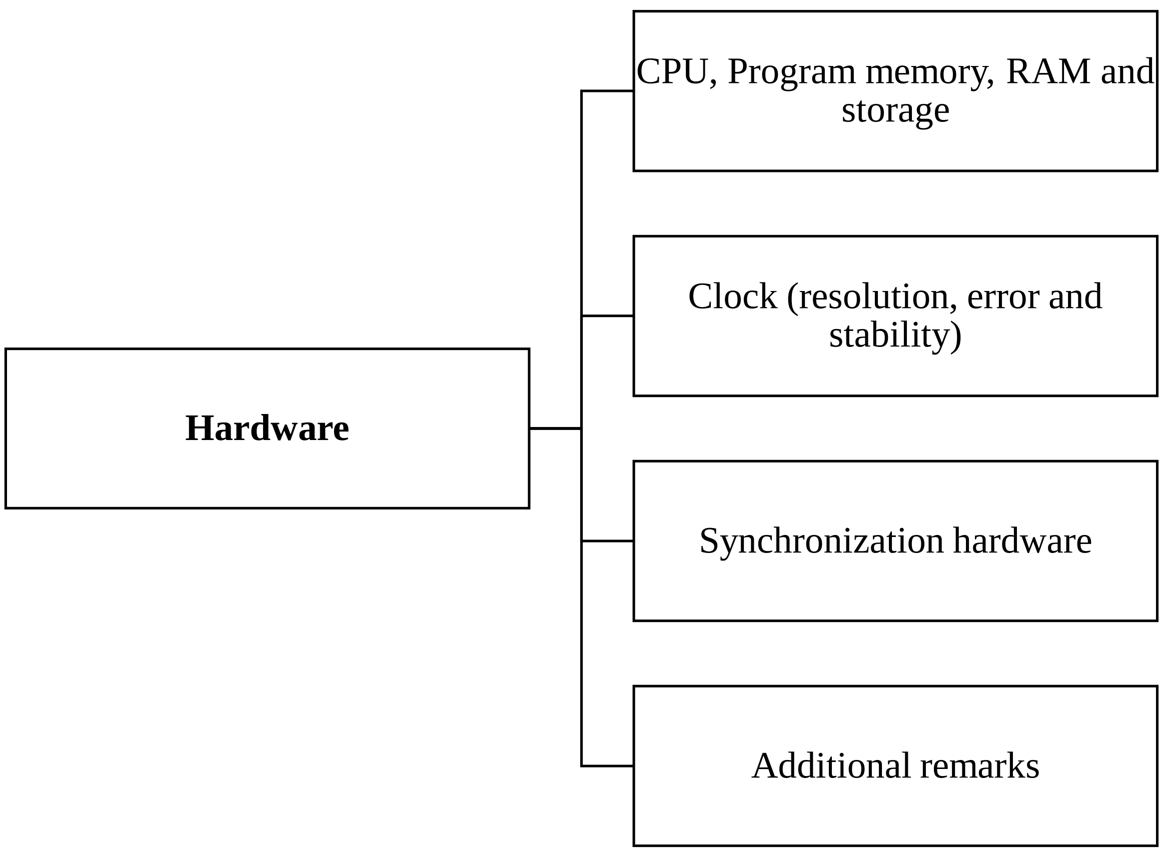

Figure 7. The individual needs of the application may not only significantly influence the taxonomy elaborated for the characteristics of deployment, but also make an impact on the designing and selection of the scheme for duty cycling.

4. Data Prediction approaches

Data readings in WSNs are often impeded by insufficient energy supply of the network. However, the actual data which could be acquired from the nodes may be substituted by the predicted data, which is the aggregate of the readings from one or several sensors.

In [8] and [47] the authors analyzed three basic approaches to data prediction, namely algorithmic, stochastic and time series forecasting methods. For the purpose of our study, these three methods are summarized in Figure 8.

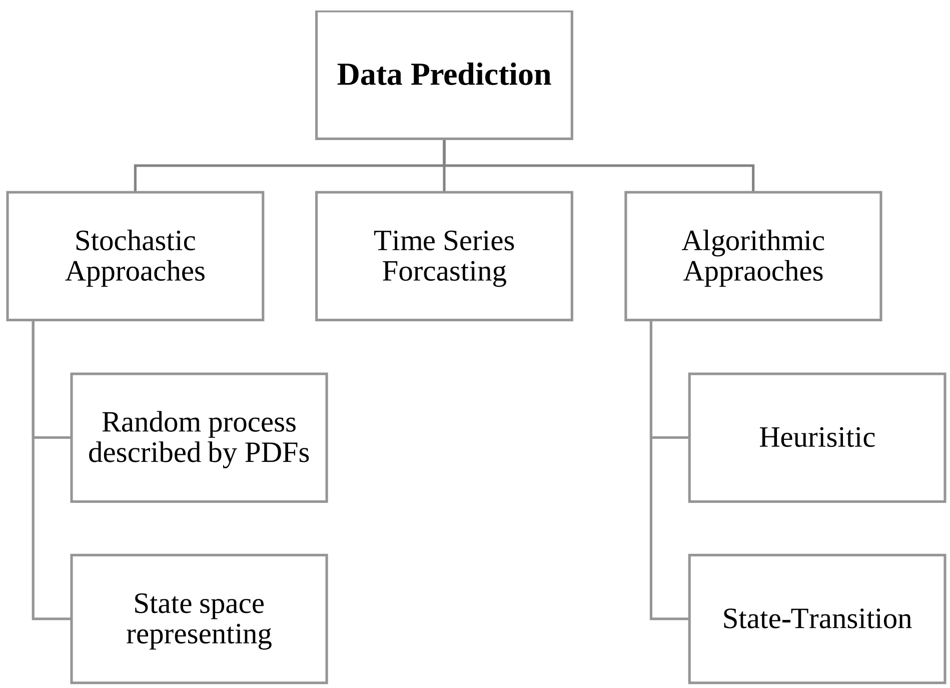

Figure 8. A detailed taxonomy of Data Prediction schemes

The goal of data forecasting method is to substitute any real data with a model which applies predictions to respond to a set of relevant queries. For the purposes of testing the validity of a model, the data is first subjected to regular sampling by sensor nodes and then compared against the forecast. Consequently, validity is considered to be affirmed when the data prediction falls within the extent of set thresholds and/or tolerances. In the contrary case, the sensor node will update the model in question or use actual data sampled before. The structure of the model and its working principles usually define which forecasting method would be compatible with it.

a. Stochastic Approaches

For the first time, the description of method for data forecasting applicable for WSNs was described in [48]. The aforementioned technique involved a so-called "probabilistic model". In greater details, the framework developed by the authors allowed wireless systems to exploit correlation-aware probabilistic models while processing the queries. Upon incorporation of the model, it was no longer needed for the system to make any direct connection to the network itself. Accordingly, the quantity of data transmissions was sufficiently reduced. Though, the aim of the work referred to above was not confined merely to reducing the data transfers but to find a means to cut down the quantity of required data samplings.

Probabilistic model relies on the forms which apply stochastic characterization methods to statistical properties and probabilities of the system. Basic techniques applied in this respect are as follows:

1) State space representation

2) Random processing.

Firstly, in the course of performing state space representation non-predictable components (noises) are eliminated, which, in turn, enables predicting further coming samples. It is possible to randomize the data, thus fitting it into the probability density function (herein - "pdf"). PDFs generated this way may serve as base for data prediction [49]. However, preliminary combining of PDFs with the samples acquired earlier is obligatory in such case. Secondly, labeling noises as unpredictable components and excluding them from the transmission helps obtaining state space representation for the chosen phenomenon. The approach addressed above may be illustrated by the Ken solution cited in [49]. This solution is targeted at reconsideration of the technique used to process basic tasks on data collection, such as, for example, "SELECT" queries for collecting data and detection of anomalies. Following the contemporary practices (such as BBQ [48]), Ken achieves desirable accuracy by relying on probabilistic models. The key asset of Ken's solution is its capability to ensure compliance of the predicted data with the real value determined by sampling. At the same time, Ken helps shrinking the volumes of data subjected to transferring. During the prediction each model involved in the data transfer is duplicated two times: first time when it leaves the source, second time when it enters the sink. Following this scheme, it becomes possible to acquire pdfs which correlates with specified attributes. Should the probabilistic base model lose its validity, it will be automatically upgraded by the corresponding node. As soon as the upgrade of the model is effected, the sink will receive new samples required for further updates. It should be emphasized that models built with regard to temporal and special correlations may be used during the training of Ken's method equally well as the ordinary models. The same refers to the models devised to deal with the peculiarities of particular phenomena. By way of example, it is rather challenging task to develop the analogue to spatial correlations, whereas temporal correlation known to resemble Markov processes, and thus can be generated as such. Since Ken is capable of processing varying kinds of processes, it may be uniformly applied both to spatial and temporal correlations. Though, the potentiality of Ken's solution is limited. Hence, there arises the need of expanding the scope of its functions by reaching to a Dynamic Probabilistic Model (herein - DPM) which provides for a smarter interface and enhances the view of probabilistic database. DPM is elaborated in more details in [50]. DPM opens up new opportunities to perform model-based views and enables more convenient managing of the databases by users. Via DPM users receive insight into databases stored by the sensors.

b. Time Series forecasting

Time series prediction is the second data prediction method which will be considered within the frame of the research at hand. During the time series prediction most credible values of future transactions are generated by analyzing values acquired from previous data samplings. In contrast both to probabilistic and statistical methods, in the interim of time series prediction only the internal data structuring is being processed. The forecasting involves the following successive steps:

1) An error is randomly selected and compared against the established pattern

2) The pattern is defined as regards its inherent features, i.e. fluctuation, periodicity etc.

3) Generating prediction model based on accomplished characterization of the pattern.

Then by the generated model it is possible to predict future values. So far, time series forecasting proves to be most compatible with non-complex basic models, such as, for example, auto-regressive, moving average or combined techniques [8]. In theory, the aforementioned models may be substituted by more contemporary and more complicated solutions, for example, GARCH and ARIMA [51]. However, in the case with WSNs more lightweight technique is preferred, since high intricacy of the models threaten the stability of entire systems.

[52] refers to PAQ which makes predictions of future values dependent on the in-built autoregressive models, attributable to every sensor. During the transmission, any immediate communication of the models with sensors is set aside. Instead, models are processed by a sink node. Accordingly, predicted values are then modeled. The sink is regularly updated as regards any new developments of the models or acquired data on external readings. Upon implementing this method the monitoring of sensors becomes more straightforward on account of dropping mostly unnecessary communications. Also, error-bound rate of forecasted data remains within control of WSN's users.

The prediction itself starts at the learning phase when values acquired previously are used for generating appropriate model. Meanwhile, the sampled data is queued with the aid of the corresponding nodes. As soon as the queue is completed the model can be generated and transferred further to the sink. In order for the model to be regarded as feasible the values obtained via it should not exceed the acceptable rate of errors. In the opposite case, the system may follow the scenarios:

a) Defining outliers among the data sets and excluding them from the reading (marking the samples),

b) Singling out invalid models and forwarding the latter for recalculation (marking model). It should be mentioned that the model is no longer valid when a sufficiently high quantity of readings performed within a series overlap the error threshold. Therefore, as soon as the update is completed, the model is directed back to the sink.

Another workable technique that can facilitate time series forecasting is familiar as "distributed clustering scheme". This technique is feasible for regrouping the nodes based on their similarities. Nodes are considered to be if they relate to one specific model operating within the limits of the chosen threshold. This threshold is defined by the users of the system. Also, PAQ makes it possible to give probabilistic responses to the queries relying on AR models. There exists two basic schemes for the operation of the model, i.e. it may be applied directly to the sink and to individual sensors;

In the first case, it produces ordinary reading, whereas in the other case the model aims to determine discrepancies between real and predicted data or any other failures requiring timely notification to the sink. As it was discussed earlier in this work, the model must be reconsidered if the data predicted by it appears to have a large error rate when compared to recent readings. On account of that, it may be concluded that AR models are fully capable of cutting down the number of communications involved in observation of the nodes. In the course of AR's operation the quality and accuracy of readings is not compromised. In contrast to the techniques that envisage use of probabilistic models, Because of AR, PAQ shows considerably better results, since the modest size of this solution enhances data processing. To be precise, PAQ allows maintaining high precision of monitoring performed on the models without increasing the number of relevant communications. Also, PAQ enables straightforward and prompt scanning of the models for detection of outliers and unpredicted parameter variations.

Further on, in [53] the author overview another type of time series models known as Similarity-based Adaptive Framework (SAF). SAF represents the combination of AR and a time-varying function. SAF encompasses benefits that would be enlisted below:

1) It is efficient in performing value predictions for the sensors which evaluate environmental parameters like humidity, temperature and others,

2) It has low operational cost,

3) It is compatible with contemporary WSNs.

Unlike PAQ, SAF is not devised for performing repeated readings for the purposes of increasing precision of the prediction. Instead, SAF seeks to prevent the involvement of highly noisy data and outliers in the readings. Moreover, implementation of SAF allows forecasting the values disregarding abnormalities of their variations. This is accomplished by way of including the trend component into the volume of data under sampling. This feature of SAF contributes in the accuracy of performed prediction and extends the scope of detections to discrepant data. There is a risk for the data to become inconsistent should any complications impede the sensors' calculation of models. Assuming that data degrade actually happens, the node will be commanded to initiate the scenario for restoring the stability of the model.

c. Algorithmic Approaches

Final chapter on comparing methods for data forecasting is dedicated to heuristic models. Heuristic models are also referred to as "state transition models". The task attributed to heuristic models lies in selecting correct techniques for devising novel models or inputting updates on characterization into currently valid models. There also exist some alternative models which can be tailored to technical requirements of WSNs. [54] addresses one of the alternative solutions mentioned earlier. In particular, the authors refer to Energy Efficient Data Collection (EEDC) mechanisms. It serves as an example of a behavioral model. The role of EEDC may be described as conducting source-initiated updates. In the course of source-initiated update, real value of sensed data is compared against the upper and lower node bounds. The precision of performed readings is confirmed by calculating differences between these two bounds. Later on, the sink distributes upper and lower bounds between the sensors which altogether form the network. Further data acquisition envisages matching bounds to the acquired samples. If the anticipated precision is not met, the sink would be immediately updated. Apart from aforementioned factors, several other issues may require more focus within the frame of this analysis. Particularly, it is essential to provide explanation of the reading process for better understanding of applicable heuristics' principles. Therefore, the process is initiated when users of the system send queries to the sink. These queries embody strict accuracy parameters. Contingent upon determining of a variation between the real values bound and the accuracy of the incoming query the sink may switch to the usage of cached range instead of actual range. Approximation generated this way will be communicated to the node. Though power consumption limits should be minded during the whole process. With regard to the need to keep consumption of power at moderate level the researchers propose to implement a new computational method. The methods should seek to organize representation of data by most suitable ranges.

Besides, reduction of power consumption is also possible to achieve upon compressing the sensed data. In [55] scientists provide overview for PREMON, which aims to observe different kinds of correlations typical for the readings performed by spatially proximate sensors. Respectively, temporal, spatial and spatial-temporal correlations may occur. PREMON method relies on the same principles as adopted in compressing the size of videos. In other words, MPEG technique embrace on wireless sensor Networks' behavior from the moment when sink receives their first readings. Here, the role of the sink is to build a prediction model based on correlative properties. The sensors receive the model as soon as it is devised. Similarly to other techniques discussed in this chapter, MPEG presupposes comparison of the real values and predictions formulated by the model. If the discrepancy of the actual data and predicted values is insignificant, there is no necessity to communicate the real value to the sink. Pursuing the objective to increase the accuracy of predictions, model is occasionally annulled and new models are created based on more recent data samples.

d. Comparison and Summary

Among the aforementioned options, the stochastic approach is the most holistic and integrated solution. Feasibility of the said method is widely acknowledged. Moreover, such technique offers new opportunities for conducting data aggregation and other related high-level tasks. At the same time, the major disadvantage of stochastic approach and/or similar methods is their excessive consumption of energy. Our observations reveal that the stochastic framework is especially feasible when applied to many sensors. In this respect, stochastic methods do not suit the purposes of the present study, which primarily concentrates on low-power sensor networks. Yet, the computational cost of the stochastic framework can be reduced with the aid to a distributed model which retains robustness of network without extra energy losses. Referring to Algorithmic methods it should be highlighted that such methods cannot be analyzed in the cumulate. There is no general concept of an algorithm. On the contrary, each algorithm serves a certain purpose, the peculiarities of which should be considered in development process. Specific features of these algorithms, as said earlier, are mainly revealed during their practical application. With regard to that, it is advisable to analyze each algorithm separately.. Hence, Time series prediction method most effectively meets the goals of the study in question since time series prediction usually runs at a moderate energy cost. In this regard, it has been proven experimentally that time series forecasting may be performed on low power networks without compromising the accuracy of predicted data. In the same vein, up-to-date technical solutions do not involve the whole amount of data in the sensing processes before enabling a compatible model. As a result, sufficient computational reduction is achieved. It should be also apparent that the extent of possible reduction is directly proportional to the volumes of stationary data subject to sensing. Upon comparison of corresponding approaches, it appears that Time Series predictions are most prevalent in the WSNs realm. During past ten years the scientists working in this field have aptly compared the techniques designed for time series predictions demonstrated in Table 2 that is mostly listed in [47].

Table 2. Time series samples of data prediction

Predicting Method	Samples	Description
Dual Prediction Scheme(DPS) or prediction approach based on Kalman Filter	[28, 56 – 60]	Agreement between node and sink with threshold
Least Mean Square (LMS)	[57 – 59]	No Agreement between node and sink – No prior knowledge
Moving Average or Autoregressive based models (AR, ARMA and ARIMA)	[47, 61, 62]	Sink and sensors exchanging data and performing prediction on both sides.
A hybrid model based on Grey-Model-based and Kalman Filter	[63]	-
Proportional–integral–derivative (PID)	[64]	-
“Send on delta”	[65]	Calculates the difference between the current value and the predicted value.
Mean square error (MSE)	[58, 59]	Calculates the difference between the current value and the predicted value.
Root mean square error (RMSE)	[62, 66]	Calculates the difference between the current value and the predicted value. / Ratio reduction/RMSE.

Today's WSNs are capable of processing complex algorithms but the highly variable data should be avoided for the sake of accuracy. In the meantime, WSN data prediction is usually performed via models based on time series forecasting instruments, such as Moving Average (MA), Autoregressive Moving Average (ARMA) [61] and GM(1,1) [67]. In case the probability density model is tried on the data in laboratory conditions, stochastic approaches have proved to be more effective. Meanwhile, it is still impossible to disregard the risk of computational overhead of the applied algorithms approaches. In view of the foregoing, this paper aims to propose the time series based method of data reduction essential for decreasing energy consumption during sensors' communication. This objective is met by putting the network through meticulous examination prior to selecting the appropriate time to commence interrogation of certain sensors.

Chapter 3

Research Method

In the first phase of the project, duty cycling approach is applied. Then, in second phase a data driven approach is used both for attenuating amount of communication and consequently, reducing power consumption. In this thesis, two different scenarios are followed during the first phase. First phase involves our primary version of both sensor and application. In the second step, both sensor and application are improved. There are two different methods that are used for the second phase as applied data prediction approaches.

1. Bluetooth as communication Protocol

We have analyzed a very wide range of basic communication protocols. Even though it is considered to be an application-specific choice, but we have tried to make a general overview on these protocols (see Appendix A).

2. Bluetooth-based, low power temperature and humidity acquisition system

a. First version of the Bluetooth sensor

The goal of the first version of the Bluetooth sensor was to demonstrate the feasibility of the architecture and to provide a device which is easy to interface to; this way showed the ability to have a simple data stream, so it made it possible to develop the “client side” application more easily. The sensor is presented in [4]. It consists of a temperature and humidity sensor, a microcontroller and a Bluetooth module. Figure 9 and 10 shows the first version of the sensor from both sides. The Bluetooth-based temperature and humidity acquisition system consists of a device comprising a sensor and a microcontroller that wirelessly transmits these climatic parameters to a receiver using the Bluetooth communication system.

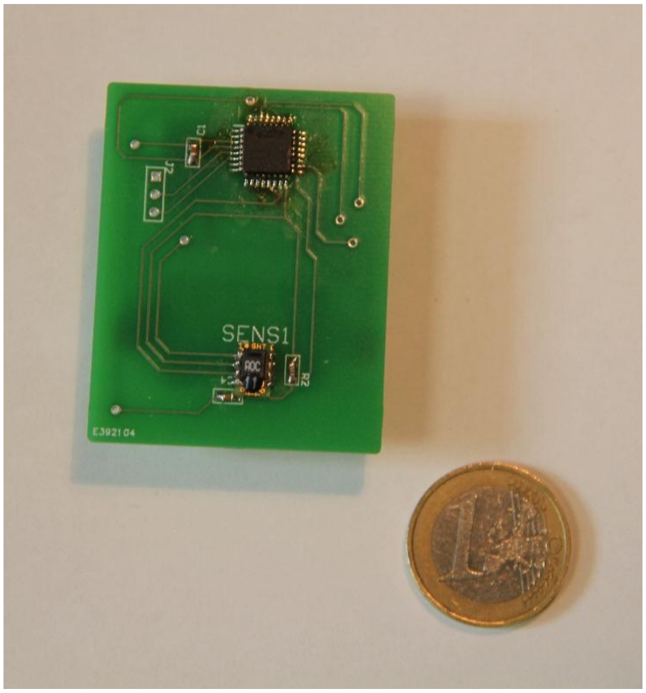

Figure 9. The electronic circuit related to the Bluetooth-based temperature and humidity acquisition system at Bottom view of the device including the temperature and humidity sensor and the microcontroller

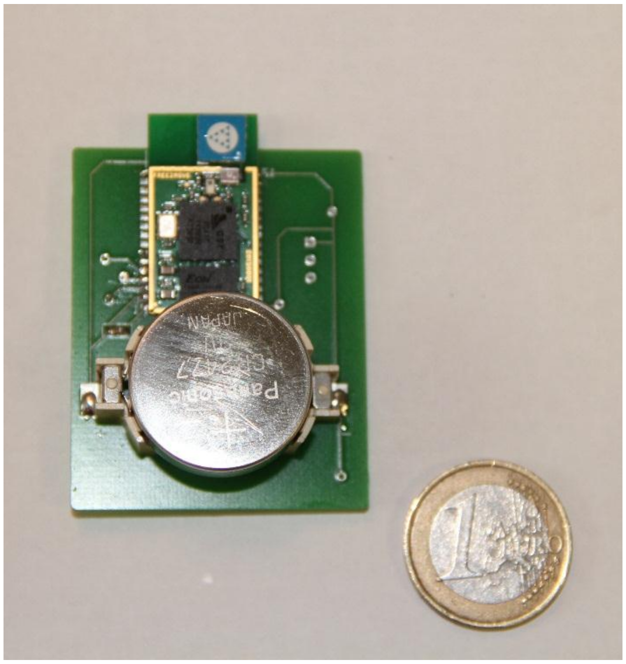

Figure 10. The electronic circuit related to the Bluetooth-based temperature and humidity acquisition system at Top view of the device including the battery holder and the Bluetooth module.

For the realization of this system, a very precise temperature and humidity sensor (SHT11 from Sensirion, temperature range from -40 to +125°C and accuracy of 0.4°C, humidity range from 0 to 100% and accuracy of 3%) was inserted to the board. The sensor was chosen due to its very low power consumption (about 80μW) and high accuracy. By using the I2C protocol it communicates the environmental parameters to a microcontroller (C8051F314 from Silicon Labs Inc), which was selected for its low power consumption as well (about 1mW for 1MHz operation) and its internal characteristics that fits the requirements of the device. The microcontroller acquired temperature and humidity values from the sensor each 10 second, and it is connected to a Bluetooth module through its embedded UART (Universal Asynchronous Receiver-Transmitter).

Bluetooth was chosen as wireless communication system because of its relative low power consumption (compared with other high data rate wireless communication systems, such as Wi-Fi), simplicity, wide use in the world, and the capability to work in lack of particular condition (e.g. the absence of the Wi-Fi connection). Moreover, since the main idea is to acquire and store data using a smartphone, Bluetooth was preferred since it is embedded in nearly all smartphones. In order to enable Bluetooth in the device, an embedded module (F2M03GLA from Free2Move) was chosen for its simplicity and relative low power consumption (about 0.6mW in sleep mode and 200mW during the transmission). The Bluetooth module receives data from the UART of the microcontroller using the SPP (Serial Port Profile) service, and directly transmits it to a receiver using the Bluetooth

Protocol. The device is powered by a lithium button battery (CR2247 from Motorola) that allows to it working for a long time according to its high capability of 1000mAh. The voltage supply of the device is 3V.

Generally, the designers overlooked the necessity to implement power saving techniques at the early stages of devising the sensor. As advantage of the Bluetooth module is its extensive connectivity. Accordingly, persons using this module may establish connection with it at any moment. As soon as the user of the module connects to the serial port, data on humidity and temperature measurements will be immediately generated, encoded by ASCII and transferred to the user's device. Yet, in the absence of any energy cost reduction mechanism, operations targeted at performing measurements of environmental parameters are impeded by power supply insufficiency. In this regard, it is suggested that the following points be enhanced:

1. Measures need to be taken in order to prevent excessive energy consumption in the times when the Bluetooth module is activated.
2. Power consumption of the microcontroller should be cut down.
3. The protocol itself requires full upgrading in view of the low energy consumption requirements.

The second version of the sensor created within the frame of this research was devised based on the aforesaid recommendations.

In this context it is necessary to highlight that most recently created Bluetooth modules (i.e. version 4) incorporate certain low power consuming techniques. In fact, most of the power available to our nodes was consumed by the Bluetooth module. Yet, the research at hand envisaged application of version 2. Accordingly, Bluetooth version 4 was not analyzed at this work.

b. Second version of the Bluetooth sensor

As already stated, the first version has a major problem: it consumes too much. That device is battery powered and it comprises a temperature and humidity sensor, and a microcontroller that transmits these climatic parameters to a receiver using a Bluetooth module. However, the presented has excessive consumption of energy since low power techniques were not adopted during the design; consequently, the battery lifetime is limited.

The second version was designed to keep the power as low as possible. This is done in three ways: by changing the components, changing the way they are handled and by changing the protocol.

In order to reduce power consumption, a novel Bluetooth based temperature and humidity acquisition system was designed and as for the components, a new microcontroller was chosen. A low power temperature and humidity sensor (SHT21 from Sensirion, temperature range from -40 to +125°C and accuracy of 0.3°C, humidity range from 0 to 100% and accuracy of 2%) was used to sense environmental parameters. The sensor communicates the environmental parameters to a microcontroller (MSP430 from Texas Instrument) using the I2C protocol. This microcontroller was selected because of its low power consumption and its integrated peripherals. In fact it consumes on average 24μW when transmitting data each minute, while the one used in the previous device consumes about 3mW. In order to avoid power consumption, the microcontroller switches on the temperature and humidity sensor only during the acquisition of the environmental parameters, then it switches it off. Moreover, the microcontroller puts itself in a low power mode between two consecutive measurements in order to save power. The time distance between two consecutive temperature and humidity measures can be set directly by the user; since the power consumption is proportional to the measuring frequency, modifying this parameter the user can select the best tradeoff between high sampling rate and long battery life. Bluetooth was chosen for its relatively low power consumption (compared with other high data rate wireless communication systems, such as Wi-Fi [24]), simplicity, wide use in the world, the capability to work in lack of particular conditions (e.g. the absence of the Wi-Fi connection), and mainly because it is embedded in nearly all smartphones. The Bluetooth module F2M03GLA from Free2Move was chosen for the designed device. Again, the main criterion used in this choice was power consumption; in fact it consumes about 26mW when it is waiting for connection and 90mW during the transmission. The Bluetooth module receives data from the UART interface of the microcontroller, and forwards it to a receiver using the SPP (Serial Port Profile) service. The microcontroller switches the Bluetooth module on only during the transmission of the environmental parameters, and then turns it off. In order to reduce the power consumption at minimum, switching off the Bluetooth module was preferred to putting it in sleep mode. The device is powered by a 3V lithium battery (CR2247 from Motorola) that allows the device to work for a long time thanks to its high capacity (1000mAh). Both sides of the electronic circuit of the Bluetooth based temperature and humidity acquisition system are shown in Figure 11 and 12.

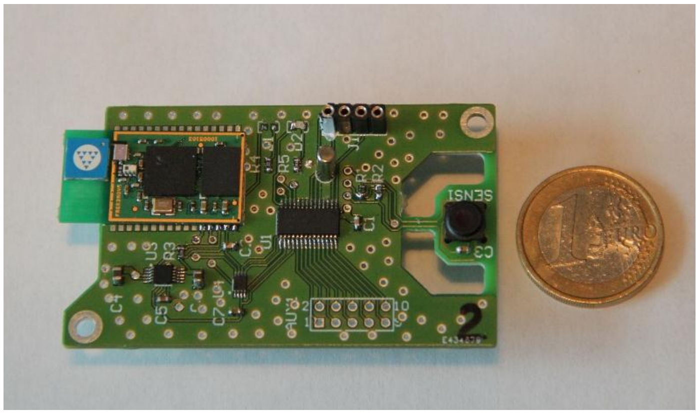

Figure 11. The electronic circuit of the new Bluetooth-based temperature and humidity acquisition system at top view of the device including the temperature and humidity sensor, the microcontroller, and the Bluetooth module with the clearly change in size.

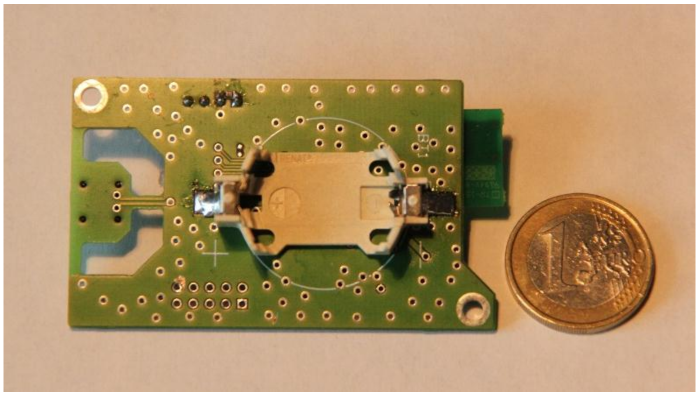

Figure 12. The electronic circuit of the new Bluetooth-based temperature and humidity acquisition system at bottom view of the device including the battery holder.

c. Smartphone application for sensing a number of sensors

As a prototype, an application for performing sensing operations via one sensor was developed for communicating directly with a certain sensor in [4]. The application read and stored data that aided to analyze them and the system performance and accuracy. As it shown in Figure 13, the obtained data of one sensor is displayed.

Figure 13. The "Display" part of the *aBluSen* previous version shows the temperature and humidity values obtained from the Bluetooth-based acquisition system.

The new version of the application reads data for several available sensors that are distributed in short distance around the mobile phone. The user can discover the sensors and select them manually, or this task can be accomplished automatically. It means that the user can select the required sensors by himself/herself or it is possible to organize system to communicate with all available sensors in the certain area automatically. The software continuously checks for the availability of the sensors and the connections. Each minute, the application can repeat the process of connection, disconnection, and reading for each sensor. As it shown in Figure 14, it is possible to select several available sensors and run system to work with selected ones.

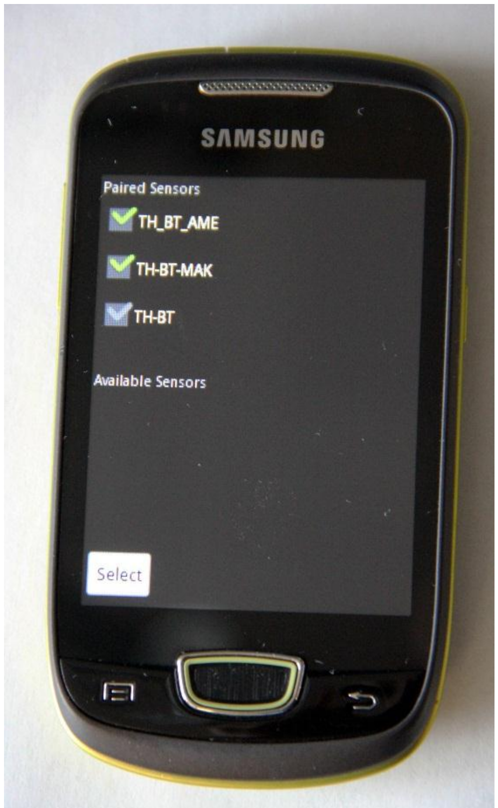

Figure 14. "Select Sensor" part of the new android application (*aBluSen*) for selecting several available sensors.

Among each of these processes, the availability of the selected sensors is checked. Also, there is an alarm if no sensor is detected in the network.

Unlike the first experiment's mobile phone application, the new version enables to show current and previous data of the current connected sensor and the last one (see Figure 15). The system shows the temperature and humidity values of the previous connected sensor at the time the system tries to connect to the next sensor or reading the sensor. It means that, at the same time, there will be the values of last two read sensors in the screen.

Figure 15. "Display" part of the new *aBluSen* application for the android's mobile phone that shows the current and last data of the temperature and humidity values of the two sensors at the same time

The application was developed, as the previous version, for Android[1] mobile phone with the updated version of the Android software development kit (SDK) (with the

---

[1] The Window-based application called "BluSen" is developed as well to meet the needs of the system. It is programmed by C# .net. Readers who want to develop the application or continue this research may pay attention to Appendix B which contains some parts of the source codes.

ADT Plug-in for Eclipse). The application has a menus for different features; *aBluSen* has four main parts:

- Configuration
- Discovering
- Selecting sensors
- Display

Discovering and Selecting parts both have a menu; the former is for scanning the available sensors, while the latter makes it possible to select desired Bluetooth-based sensors from which gather the environmental information. *aBluSen* distinguishes between non-Sensor Bluetooth-based devices and Sensors. Checking the name of the devices is a method to detect differences between Sensors and non-Sensors. In Configuration part, user can set the number of sensors that can be simultaneously connected, sampling frequency, and also the way of system working (e.g. manually or automatically). As mentioned before and showed in Figure 15, Display part shows the temperature and humidity values as environmental information related to the current and previous sensor, number of available and selected sensors, sensor information. After first communication, sensors are set as Paired-Sensors. After selecting sensor/s, the process of connecting starts. In some cases, twice attempts are needed to connect to the sensor. There are two methods of connections:

- Standard method
- Reflection method

If the standard method of connecting failed, the reflection method is then launched. The second method avoids failure of connection during the standard attempts for connecting. After connecting, the *aBluSen* tries immediately to read data from the open port. The mobile phone keeps listening to the port to read Input stream while connected. The size of buffer can set from the application that handles the amount of data for each time of reading. The default size is 64.

The obtained data stream needs to follow the process of *tokenization* to break desired values of temperature and humidity from several lines of data that are read from the sensor. For example, the following line is the sample line of read data for the first attempt to a sensor: *T=+ ., +25.6,46*. Two required values break from the lines. Here *+25.6* is the temperature and 46 is the humidity. At the same time, all sensor data is being stored in a text file in android based mobile phone. The application has a setting to select a number of available sensors to follow mentioned processes automatically. There is a menu with multi-selecting feature to choose sensors then *aBluSen* connects to each of selected sensors as described for the single sensor. The process of reading from connected sensor will be finished after a certain time. This period for each sensor is 1 minute as default. The frequency sample can be changed for communicating and reading to/from the sensors. After this certain interval, the reading process will be stopped then the application tries t communicate between mobile phone and the other selected sensor. The mobile phone communicates with all sensors and reads and stores information according to the configured interval. As it is demonstrated in Figure 13, the display shows the environmental information and the name of the connected sensor while reading data. The application can work continuously.

3. Algorithm for efficient sampling

a. Using a Multilayer perceptron

To reduce the number of acquired data, we predicted them and estimated the uncertainty of the prediction. An additional measurement was required from a sensor when the associated uncertainty went beyond the threshold.

Each available measurement was considered together with its uncertainty, assumed to be equal to the accuracy of the sensor. A MLP was used to perform periodically a forward prediction on 100 realizations of stochastic inputs extracted from a uniform probability distribution with mean and range given by the available data and their uncertainty, respectively (see Appendix 1 for details on the MLP). The prediction was computed as the mean of the obtained 100 estimations. The uncertainty of the prediction U was defined in terms of two contributions. The first was the dispersion of the predictions, indicated in the following as U_1 and defined as the range of the estimations provided by the MLP from the 100 random trials. The second contribution U_2 was the estimated rate of prediction error

$$U_2 = \frac{1}{2}\left(\frac{|p_j - m_j|}{\tau_j - \tau_{j-1}} + \frac{|p_{j-1} - m_{j-1}|}{\tau_{j-1} - \tau_{j-2}}\right) \quad (2)$$

Where p_j and m_j indicate the jth predicted and measured value, respectively (so that $|p_j - m_j|$ is the prediction error), τ_j is the time sample in which the jth measurement is taken (so that $\tau_j - \tau_{j-1}$ is the time delay between the jth and the previous measurement). Thus, U_2 is the mean of the last two estimated ratios between the prediction error and the time delay from the last measurement (so that the estimated rate of increase of prediction error has a memory term). A convex combination of the two contributions was considered as the definition of the uncertainty

$$U = \alpha U_1 + (1 - \alpha)U_2 \tag{3}$$

where the parameter α (with $0 < \alpha < 1$) weights the importance of the two mentioned contributions. In the following, the same algorithm is tested on different datasets. For such general applications, there is no reason to give more importance to one of the two contributions in Equation 2, so that α is considered 0.5 in the following. However, for specific applications, a different weight could be optimal.

A new acquisition was required from a sensor when the uncertainty of the predicted measurement was larger than a threshold (which was chosen as sensor specific). Thus, the MLP was used to estimate when and from which sensor to acquire a measurement. This allows to reduce the number of measurements and, consequently, also the power consumption (as there is a decline on communication, the energy is saved by decreasing the number of transmissions). After acquiring a measurement from a sensor, its present and past data were updated by interpolating the acquired measurements and their uncertainties were updated according to the accuracy of the sensor.

(i) Data test bed

Both simulated and experimental data were used to test the algorithm.

1) Simulated data

Simulated data were deterministic and noise free. Two different simulations were considered. The first one involved the following two signals

$$\begin{aligned} x_1(t) &= \sin(2\pi f(t)t) \\ x_2(t) &= a(t)x_1(t) \end{aligned} \tag{4}$$

where t is the time (in the range of 0 to 200 s, sampled at 20 Hz), f(t) is a square wave varying between 0.5 and 1.0 Hz with period 20 s and a(t) = 4 + sin(0.15πt). The signals were quantized in order to have resolution 0.05 (also considered as the accuracy of the measurement). The two signals x1 and x2 were first used separately, then together.

As the second set of simulations, two uncorrelated signals were considered and sampled every 6 s for 60 min. The first signal is a sinusoid with frequency 0.1 Hz, and the second is defined as the first component y1 of the solution [y_1 y_2 y_3] of a Lorenz system in chaotic regime [68]:

$$\begin{cases} \frac{dy_1}{dt} = -10(y_1 - y_2) \\ \frac{dy_2}{dt} = 28y_1 - y_2 - y_1 y_3 \\ \frac{dy_3}{dt} = y_1 y_2 - \frac{8}{3} y_3 \end{cases} \tag{5}$$

The signals were quantized in order to have resolution 0.1.

2) Experimental data

Two different experimental data were considered. The first dataset was constituted of meteorological data acquired every 15 min from four sensors, measuring temperature, pressure, wind velocity, and humidity, located at the Turin-Caselle airport, for 100 days from June to August 2010 (refer to [69] for details).

The second dataset was gathered from two sensors of a Bluetooth-based acquisition system that measures temperature and humidity. The general structure of the WSN is shown in Figure 2. A smartphone communicates and reads data from sensors separately. The Bluetooth module F2M03GLA from Free2Move was attached to the device. It consumes about 44 mW when it is waiting for connection and 108 mW during the transmission. Data is received from the UART interface of the microcontroller by the Bluetooth module and forwards it to a receiver using the serial port profile (SPP) service. The device is powered by a 3-V lithium battery (CR2247 from Motorola) with 1,000 mAh. The sensors were fixed on a carrier, and their location was changed irregularly and sequentially in three different locations in a laboratory:

- Close to air conditioner (cold source)
- Close to working laboratory’s equipment (warm source), and
- Far away from any other sources (normal).

The sources were sufficient to prompt changes on temperature and humidity up to 5°C and 10%, respectively. Data were recorded every 15 s for about an hour.

b. Using Nonlinear Autoregressive

The efficient time series prediction of the sensor's output is needed to achieve the goal of minimization of power consumption by the sensor. Lesser the communication, the lesser will be the power consumption. As discussed before, there are different kinds of time series prediction methods depending on the applicable parameters and the practical usage. We have implemented a Nonlinear Autoregressive (NAR) model for prediction. This model is used for prediction of an output at time t by using the subsequent outputs as shown in Figure 16. We used an Artificial Neural Network (ANN) by employing NAR model for time series prediction. An ANN is a network composed of large number of inter-connected units called neurons. An ANN architecture may have one or more hidden layers, but typically one hidden layer is sufficient to map any kind of linear as well as nonlinear approximation as shown in Figure 17 [70]. Estimation of optimized number of neurons in the hidden layer is a vital task. Higher number of hidden layer neurons may result overfitting due to over parameterization. On the contrary, a small number of hidden neurons may become insufficient to fit the data.

We used sensor's values against the time and applied to ANN using one hidden layer. In this application, the temperature values vector $[T_1, T_2, \ldots, T_n]$ against n time-steps n are fed as input to the network. We introduced two (feedback) delays in the input layer to store the previous two values: T_{j-1} and T_{j-2} for the prediction of target value T_j at the jth time stamp. Therefore the network uses the temperature values at two delayed time-stamps to predict the current value (see equation 6) [70]. Learning of the neural network plays an important role in achieving optimum results. We used the Levenberg-Marquardt (LM) algorithm [71] as the training algorithm of the classifier. This is a sophisticated form of gradient descent back-propagation algorithm which performs nonlinear least square minimization. The mathematical details of the LM algorithm are provided in [72]. Parameters for network training are summarized in Table 3.

$$y(t) = F\left[y(t-1), y(t-2), .., y(t-d)\right] \tag{6}$$

Table 3. Network's training parameters

Parameter	Value
Minimum gradient threshold	$1\ e^{-10}$
Initial learning rate (μ)	0.01
Increasing ratio of μ	10
Decreasing ratio of μ	0.1
Maximum value for μ	$1e^{8}$

The data is allocated as follows: 70% for network training, 15% for cross validation, and the rest 15% for test purpose. The network is set to be trained using the training data, and simultaneously to be optimized based on cross validation outcome. In every iteration, regularized cost for the training data is calculated as:

$$J(\beta)=\left[-\frac{1}{m}\sum_{i=1}^{m}y^{i}\log(P(x^{i}))+(1-y^{i})\log(1-P(x^{i}))\right]+\frac{\lambda}{2m}\sum_{j=1}^{n}\beta_j^2 \qquad (7)$$

Where xi represents the input feature vector for ith sample, yi represents target value of ith sample, λ is the regularization parameter, set as 0.01, and βj represents the weight parameter for jth sample. P(xi) represents the sigmoidal output for ith sample and is calculated as

$$P(x)=\frac{1}{1+e^{-\left(\beta_0+\sum_{i=1}^{n}\beta_i x_i\right)}} \qquad (8)$$

In order to train the network, the cost equation 2 needs to be reduced after every iteration. Weight parameters are optimized after each iteration according to the LM algorithm. For the hidden layer, sigmoid activation function (see Equation 8) is used whereas linear function is applied at output neuron. Training is set to be stopped if either there are six consecutive increasing in validation error or the gradient becomes less than the selected threshold (see Table 3). Later the network calculates the test data results.

Our proposed solution's artificial neural network architecture is shown in Figure 18.

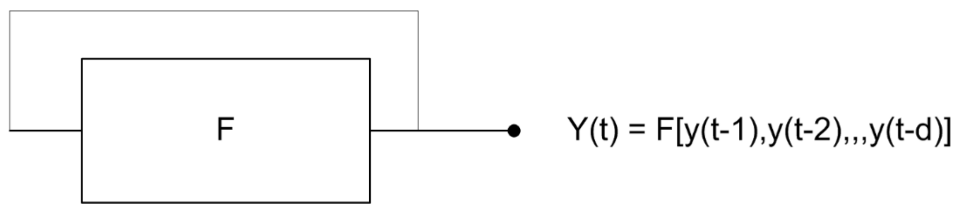

Figure 16. Nonlinear Autoregressive model

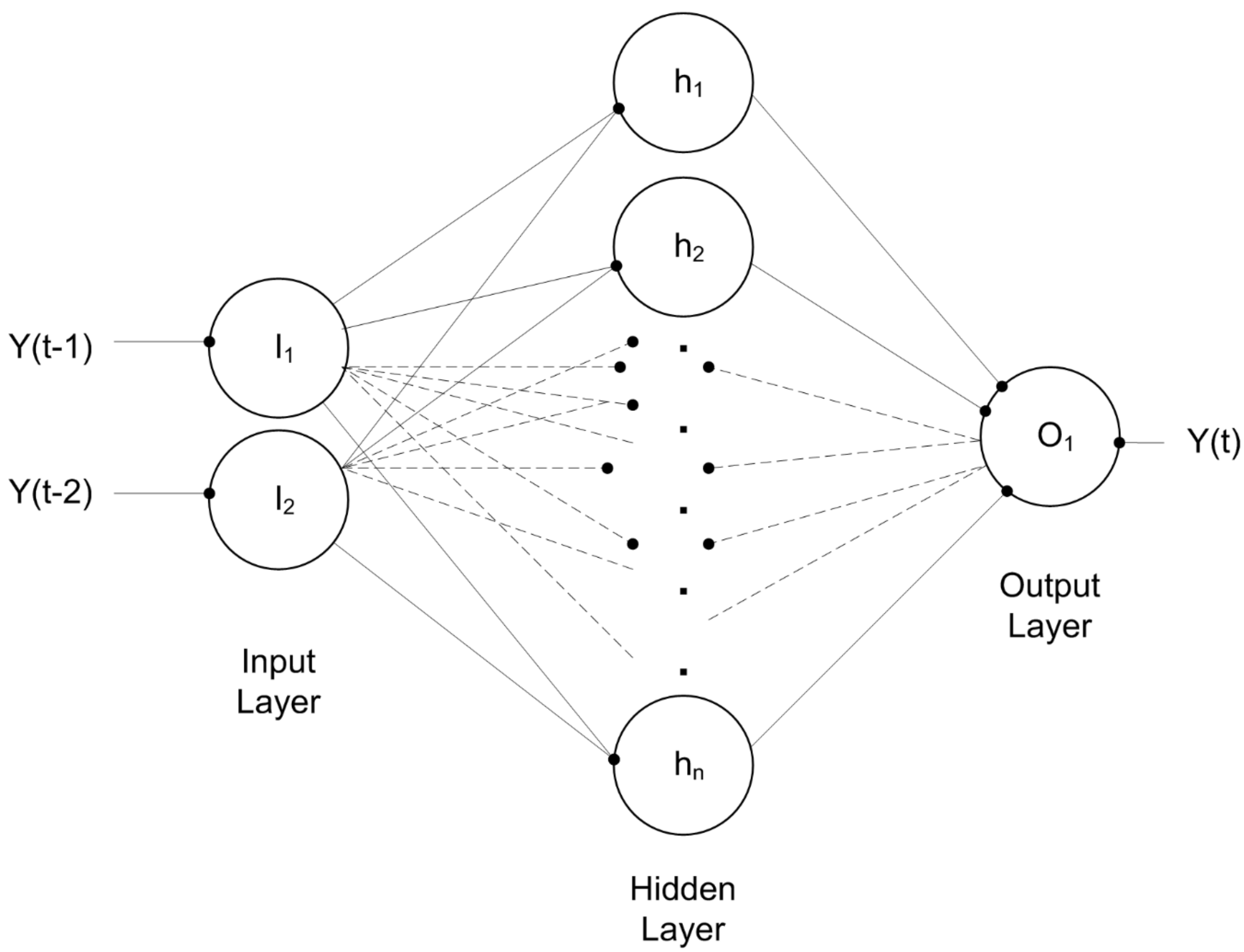

Figure 17. A three layer artificial neural network architecture

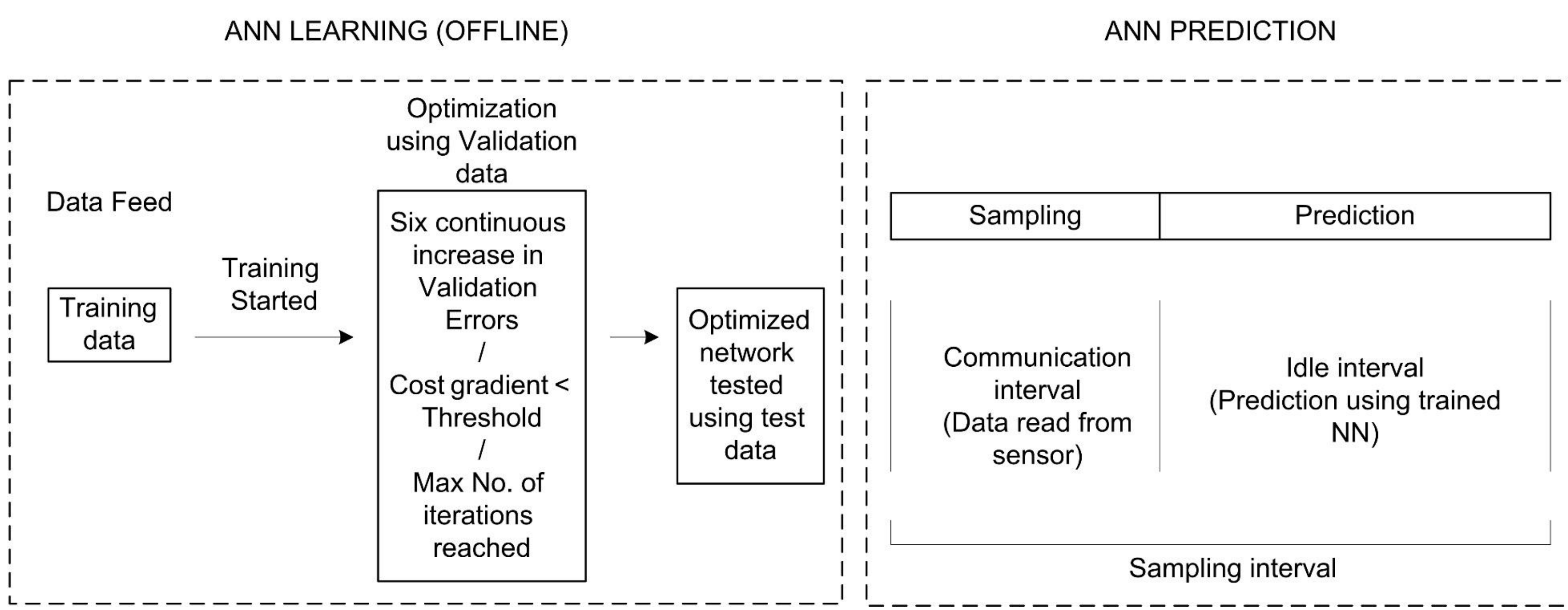

Figure 18. Block Diagram of overall Methodology

Algorithms (1 and 2) for network training for efficient sampling are defined as follows:

Algorithm 1 Neural Network Training

```
Input: training data for n time stamps at input layer
Return sum
  while Number of iterations = Max iterations do
     Calculate cost J(β), as in (3)
     Calculate gradient of J(β),
     Compute the validation data errors,
     Compute the no of iterations c, if validation error continues
     increasing
     if ( thenc == 6) then
        Stop training
     else if ( thengradient ¡ Min threshold defined) then
        Stop training
     end if
  end while
   Test the network by using test data
```

Algorithm 2 Efficient sampling from sensor

```
Define: Max number of time stamps
  while time stamps = Max time stamps do
     Define no. of time stamps for sampling (opted equal to 2)
     Define sampling interval (8 for Temp sensor)
     if ( thensampling interval ends) then
        Read data from sensor
        if ( thensampling sampling time ends) then
           Start prediction
        end if
     end if
  end while
```

(ii) Data test bed

The algorithm was verified based on two types of data. Likewise, the proposed solution was analyzed under different conditions of datasets, i.e. in the chamber and in the natural environment of the Neuronica Laboratory of Politecnico di Torino.

1) First Dataset

In [4], the choice of Texas Instruments MSP430F2132 is determined by its low power consumption. The sensor Sensirion SHT21 was selected for the same reasons [5]. This Bluetooth-based sensor incorporates 3V lithium battery (CR2247). In the course of the experiment the environmental data was obtained via three Bluetooth-based temperature and humidity acquisition systems. The experiment was conducted in the controlled environment, in particular, in a climatic chamber with temperature range for climatic test from -40°C to +180°C and Angelantoni Challenge 250.

Initial environment of the chamber was established as follows (as shown in Figure 19): a relative humidity of 50%; temperature of 25C. These circumstances were maintained for the period of 10 min. Then the temperature was decreased until -20C. Its gradient was set as -0.5°C per minute. The lowered temperature was preserved within the chamber for about 10 minutes and then brought back to 25C. Then gradient was estimated as 0.5°C per minute. At the final stage of the experiment the stable temperature of 25C had been supported within chamber for 10 minutes. The sampling frequency of the wireless sensor network was set to a rate of one sample per minute. The interval between two corresponding measurements was set to 60-second schedule.

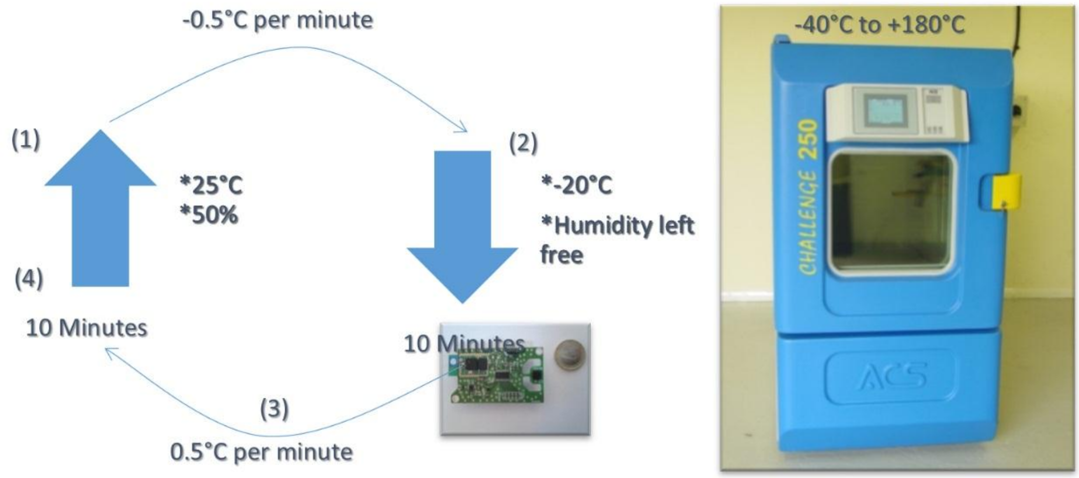

Figure 19. The process of testing, recording data and analyzing acquisition system in a climatic chamber with different temperature ranges.

2) Second Dataset

The second consequent sampling was performed to assess the temperature and humidity parameters via Bluetooth-based tool in real conditions. Sampling of the datasets was performed in the laboratory, where carrier was used to move the sensors between the warm and cold sources. The data reading at every source points lasted for about 2-3 minutes. In accordance with the scientific requirements, the laboratory sources used in the experiment could vary their own temperature by 5C and humidity by %10. Data readings were performed every 15s. The length of entire experiment was one-hour. The stationary conditions of the experiment can be potentially reproduced in any ordinary environment. For the sake of proper comparison between method that is proposed in [16] and the final proposed solution in this thesis, this data set is the same as the one tested in the second part of the 3.a.1.2.

4. Power Consumption; Relation between power saved and reduction ratio

Here, sensor measurement was conducted. The sensor can be either in a reading or a no communication state. The power consumption of the Bluetooth-based temperature and humidity acquisition system was measured. The sensor was supplied with a constant voltage of 3 V, and the current was measured using a digital multimeter (DM3051 Digital Multimeter) with a sampling frequency of 10 Hz. As showed in Figure 20, P_R = 108 mW and P_{NC} = 44 mW, so that their ratio is about k = 2.45

$$P_{\mathrm{R}} = kP_{\mathrm{NC}} \tag{9}$$

The average power P_{AVE} is the sum of the power spent during reading P_R or during no communication P_{NC} weighted by the percentage time spent in the two states (T_{ON} and T_{OFF}, respectively)

$$\begin{aligned} P_{\mathrm{AVE}} &= \frac{T_{\mathrm{ON}}}{T_{\mathrm{ON}}+T_{\mathrm{OFF}}} P_{\mathrm{R}} + \frac{T_{\mathrm{OFF}}}{T_{\mathrm{ON}}+T_{\mathrm{OFF}}} P_{\mathrm{NC}} = \\ &= \frac{T_{\mathrm{ON}}k+T_{OFF}}{T_{\mathrm{ON}}+T_{\mathrm{OFF}}} P_{\mathrm{NC}} \end{aligned} \tag{10}$$

When our algorithm is applied, the signals are under-sampled, so that the average time of the reading state is reduced (of a ratio given by the reduction ratio imposed by the choice of the threshold). The reduction in average power is described by the nonlinear function multiplying the power of the no communication state in the previous equation

$$f(T_{\mathrm{ON}}) = \frac{T_{\mathrm{ON}}k+T_{\mathrm{OFF}}}{T_{\mathrm{ON}}+T_{\mathrm{OFF}}} \tag{11}$$

This function is monotonically increasing and is larger than 1 for positive values of T_{ON}. If the time of no communication is much larger than the reading time, the function is close to be linear with angular coefficient k/T_{OFF}. Considering our experiment, assuming that the reading time is 1 s long and that the reference sampling is at 0.1 Hz, the factor in Equation 11 is varying between 1 and 1.17. For example, we could achieve about 7.5% of power saving with a reduction factor of 50%.

Figure 20 shows power measurement of one sensor during two states of “reading” and “no communication” states.

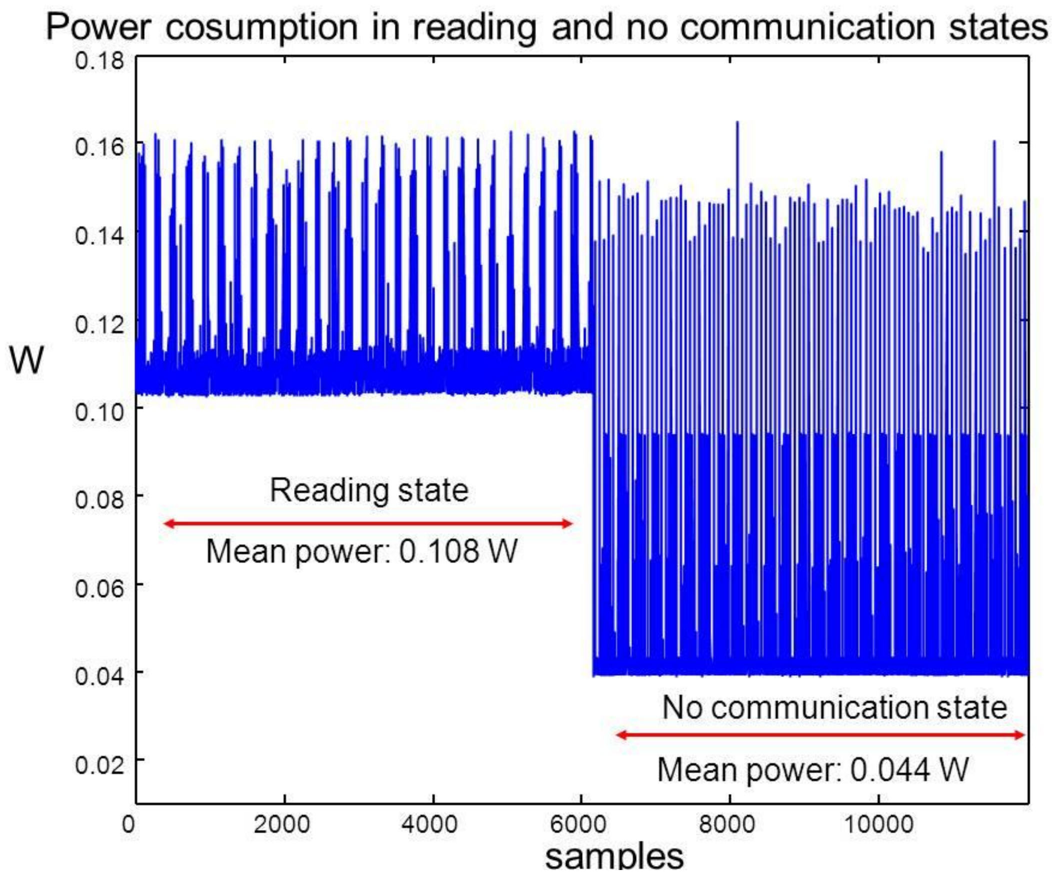

Figure 19. Power measurement of one sensor during two states. The sensor is supplied with a constant voltage of 3V, and the current is measured using a digital multi-meter with a sampling frequency of 10 Hz.

Chapter 4

Results

1. Bluetooth-based acquisition system

During the first experiment, temperature and humidity values, as raw data, were acquired using the smartphone application from three Bluetooth-based acquisition systems. The results are shown in Figure 21. Temperature values obtained by the three devices correctly follow the temperature condition provided by the climatic chamber (see the description in the previous section). The humidity values for temperatures higher than 0°C are nearly at 50% as imposed by the climatic chamber. For temperature lower than 0°C, since the climatic chamber cannot impose the humidity condition for these temperatures, the humidity values gathered by the three devices are not constant to a fixed value.

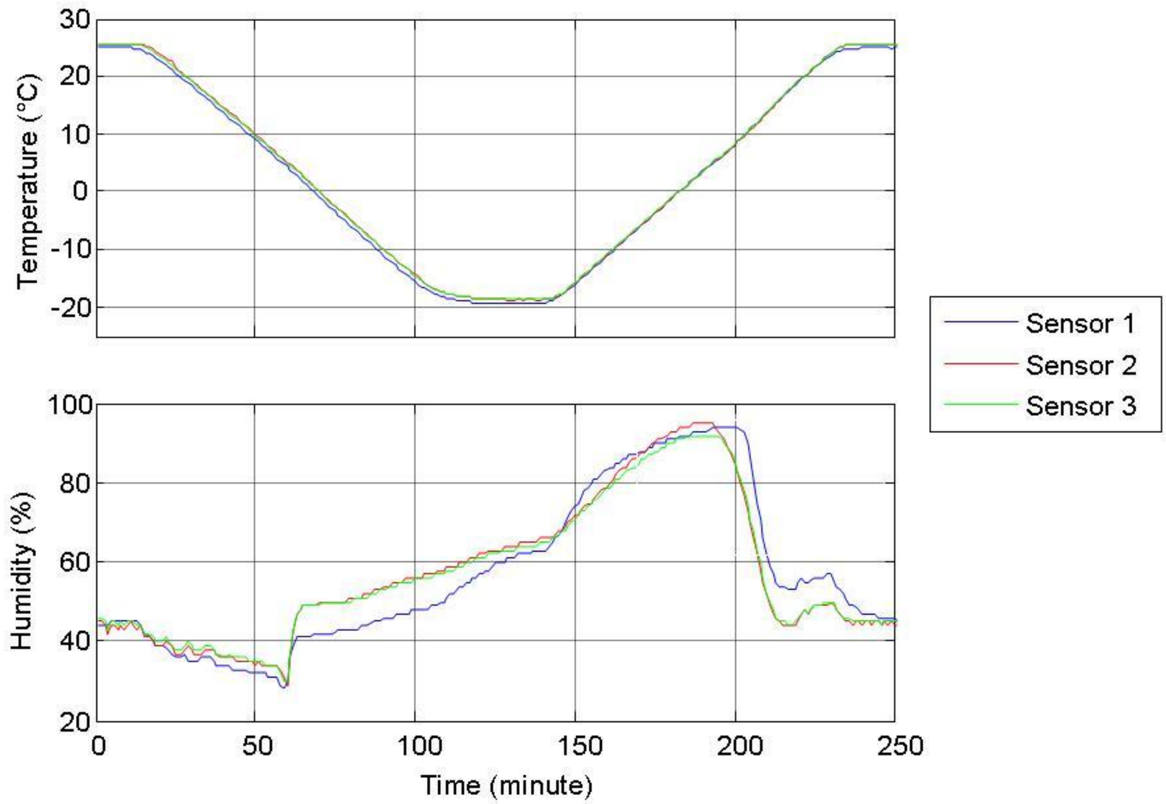

Figure 20. Temperature and humidity values that are obtained by the three Bluetooth-based temperature and humidity acquisition systems during the first experiment.

The values of the current raw data that are gathered by the Bluetooth based temperature and humidity acquisition system described in this work and the acquired values by the digital multimeter during the second experiment. The device was connected in series to a digital multimeter, and it was powered by a power supply. The experiment lasted for an hour, and the current values were averaged (since the device transmits the environmental data each minute). The current value absorbed by the device between two transmissions is in the order of 1µA. The current consumption during the transmission, without any smartphone connected, is in average 8mA (Figure 22), while during the acquisition with a smartphone is in average 27mA (Figure 23).

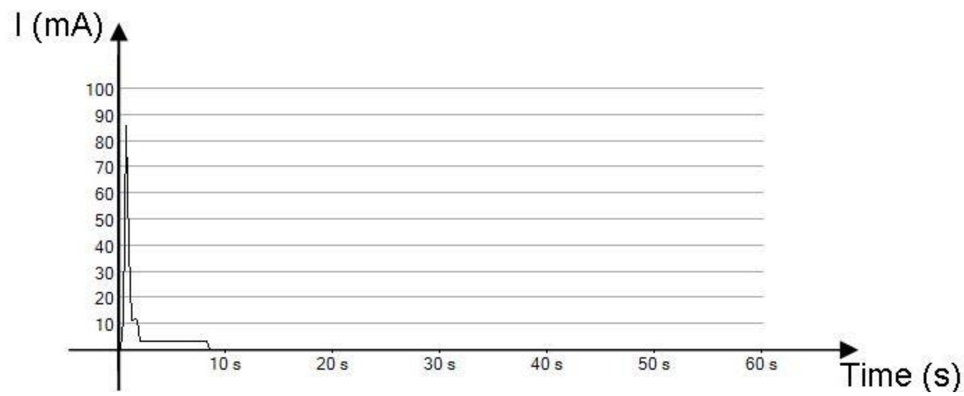

Figure 21. **Current values absorbed by the Bluetooth based temperature and humidity acquisition systems during the second experiment without smartphone connected.**

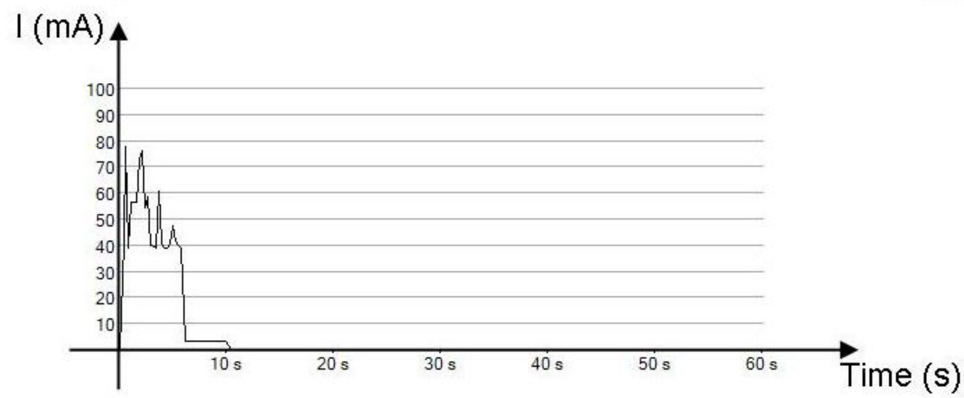

Figure 22. **Current values absorbed by the Bluetooth based temperature and humidity acquisition systems during the second experiment with smartphone connected.**

2. Using a Multilayer Perceptron

Figures 24, 25, 26, 27 and 28 illustrate the application of the method to simulated signals. In Figure 24, 25 and 26, correlated non-stationary signals are considered.

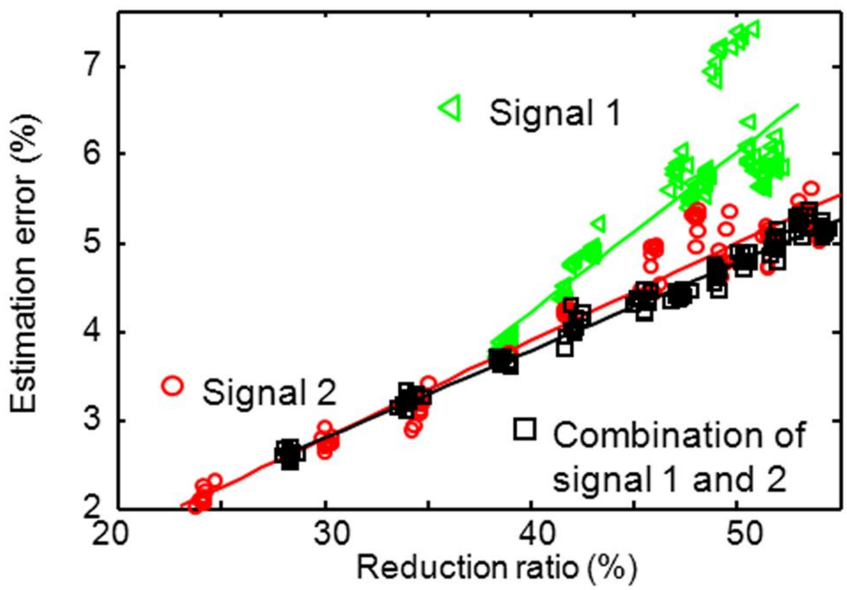

Figure 23. Application of the method to non-stationary, correlated signals. Relation between reduction ratio and error (100 simulations with different thresholds are considered).

As shown in Figure 24, the method using the combination of the two signals has a lower slope of the reduction ratio versus estimation error. This indicates higher performances when information is jointly extracted from the two correlated signals. Moreover, the method selects more samples for the portions of the signals with higher frequency (Figure 25), showing the ability to adapt in time to temporal variations of the signal.

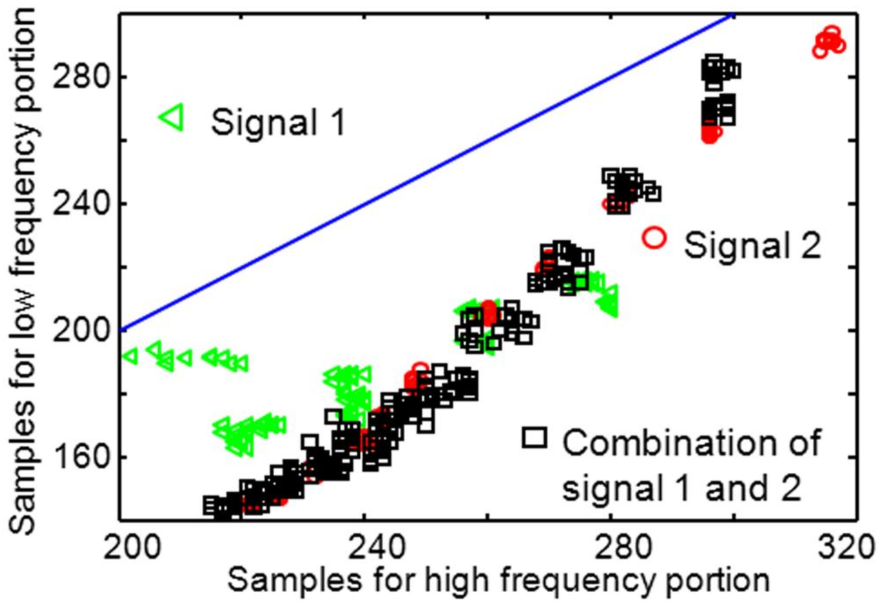

Figure 24. Application of the method to non-stationary, correlated signals Samples for the portions of the signals with higher frequency versus those with lower frequency (same 100 simulations as in Figure 24).

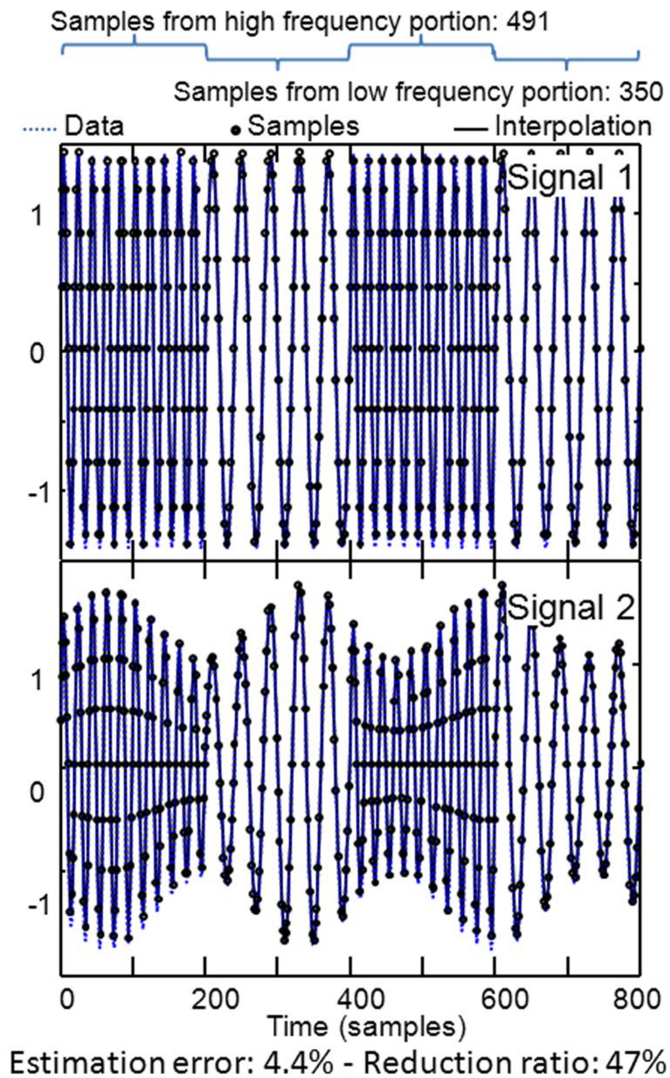

Figure 25. Application of the method to non-stationary, correlated signals. Representative example application for the method.

In Figure 27 and 28, the sinusoidal and the chaotic signals are considered. As expected, the prediction of the sinusoid is simpler than that of the chaotic signal; for this reason, more measurements are selected by the algorithm to sample appropriately the second signal (see Figure 27).

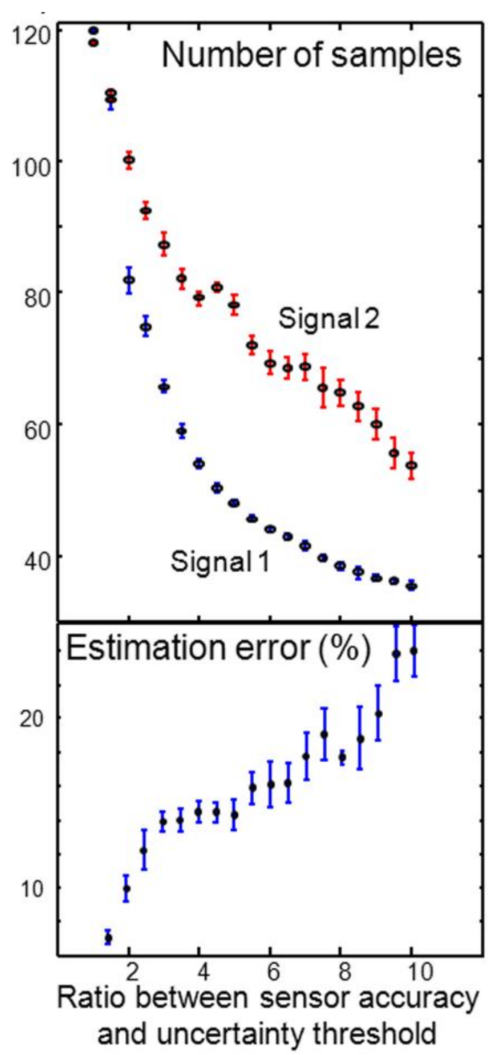

Figure 26. Application of the algorithm to simulated data. Number of samples and mean estimation error (mean and standard deviation over ten repetitions).

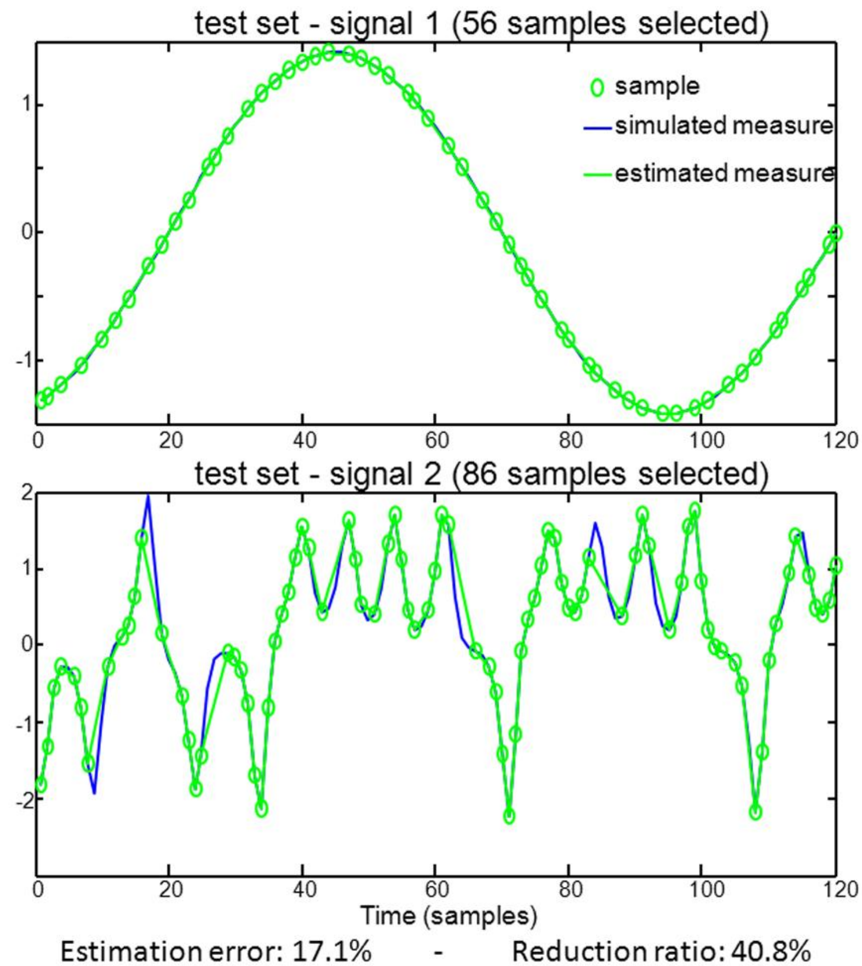

Figure 27. Application of the algorithm to simulated data. Representative example (threshold = 0.03 for both signals).

In Figure 29 and 30, a representative example in the subsequent paragraph, application for our algorithm to our first bunch of experimental data is shown (meteorological data). The MLP was trained and validated on the basis of the first 80 days (see Appendix C). Then, it was applied for the following 20 days considered in Figure 27 and 28 (test set). In Figure 29, we show the results of many applications of the prediction algorithm to the test experimental data, with different thresholds. As expected, by increasings the threshold, the reduction factor

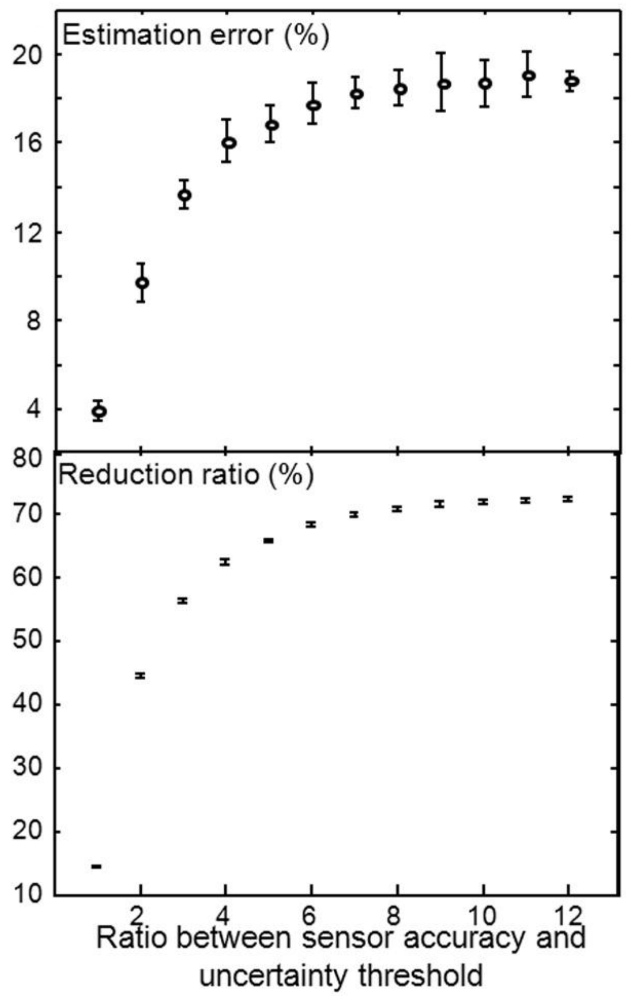

Figure 28. Application of the algorithm to meteorological experiments. Accuracy is assumed to be 0.2°C, 20 hPa, 0.1 km/h, and 1%, for the temperature, pressure, wind velocity, and humidity sensors, respectively. Root mean square estimation error and reduction ratio as functions of the uncertainty threshold (20 repetitions are considered).

increases, at the expense of decreasing the accuracy in estimating the measurements. In Figure 30, a portion of the test data is shown. Note that the number of samples required from the wind velocity sensor is the highest among the four sensors, reflecting the erratic dynamics of the signal. On the other hand, the sampling of humidity, which has smooth variations correlated with temperature, has the lowest rate.

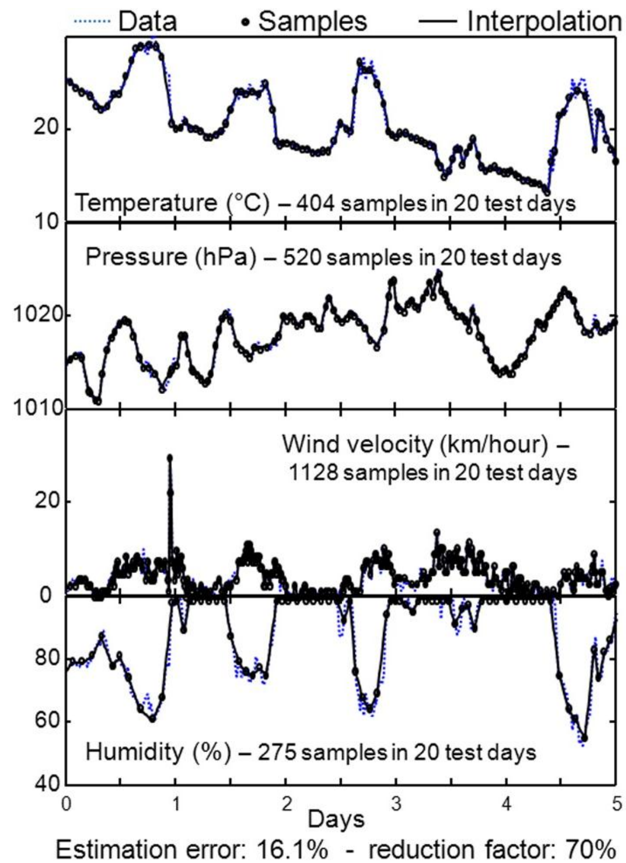

Figure 29. Application of the algorithm to meteorological experiments. Example of application to a portion of the test set.

The second bunch of experimental data used as an example application for our algorithm is shown in Figure 31, 32 and 33 (indoor experiment with a WSN). Temperature and humidity values are clearly correlated when measured from the same sensor. Some lower correlations are also visible from data via different sensors, as they are placed close to each other. Figure 31, 32, 33 show the results

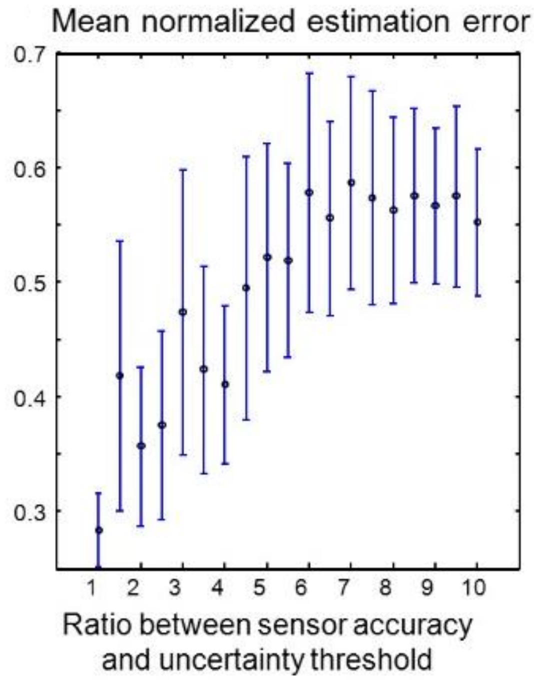

Figure 30. Application of the algorithm to indoor experiments .Root mean square (RMS) estimation error.

of many applications of the prediction algorithm to the test experimental data with different thresholds. Again, by increasing the threshold, the reduction factor

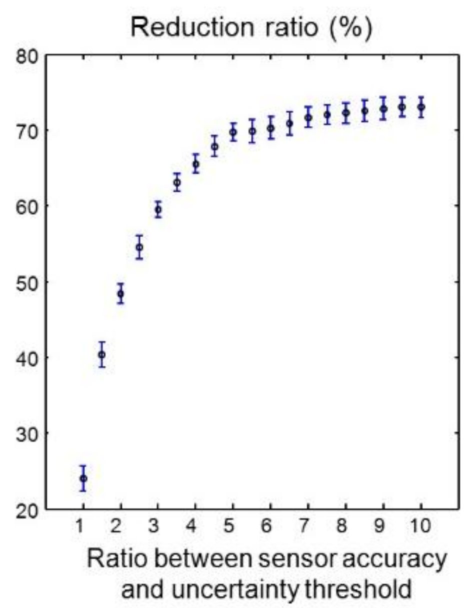

Figure 31. Application of the algorithm to indoor experiments. Reduction ratio as functions of the uncertainty threshold (assumed proportional to sensor accuracy; the method was run 100 times for each choice of the threshold, mean and standard deviations are shown). The accuracy was assumed to be 0.1°C and 0.3%, for the T and H sensors, respectively.

increases and the accuracy in estimating the measurements decreases.

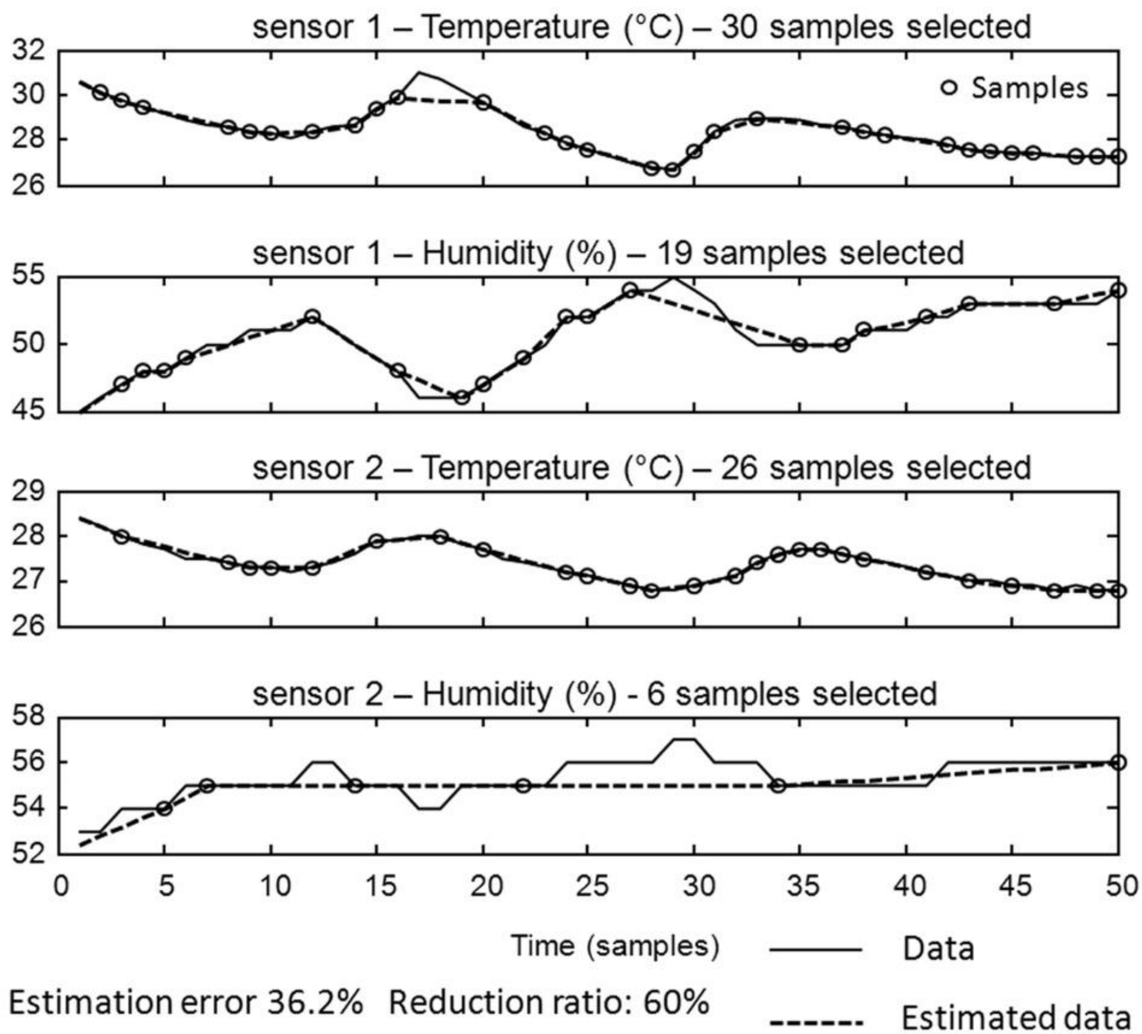

Figure 32. Application of the algorithm to indoor experiments. Representative example application for our method: uncertainty of the measurements was assumed to be two times the accuracy.

3. Applying NAR

We have the dataset obtained from temperature as well as humidity sensors placed in the indoor environment covering a time span more or less of two hours with 96 samples. In addition, we have another dataset composed of temperature values for 260 samples, recorded from the sensor placed in an environmental chamber for 4.5 hours. This data can be assumed as noise free simulated data representing the temperature. We applied the sensor's values as targets against time to the network and analyzed the time series prediction response. As we discussed earlier, a number of hidden layer neurons plays an important role in achieving the optimized network accuracy. To estimate a good choice for hidden layer neurons, we varied this number and analyzed the network performance. For each of the sensor's dataset, we changed the number of hidden neurons as 5, 10, 20 and 30, and observed the performance with respect to mean squared error (MSE). We introduced two feedback delays (two delayed values are fed to the network) to predict the current value. The prediction response for the humidity sensor, dataset 1, showing target and the predicted outcomes for each timestamp is shown in the Figure 34. The error for each quantized timestamp is also plotted at the bottom of each response (recorded with variable number of hidden layer neurons). Similarly, the prediction response for humidity sensor, dataset 2 is also recorded and illustrated in Figure 35.

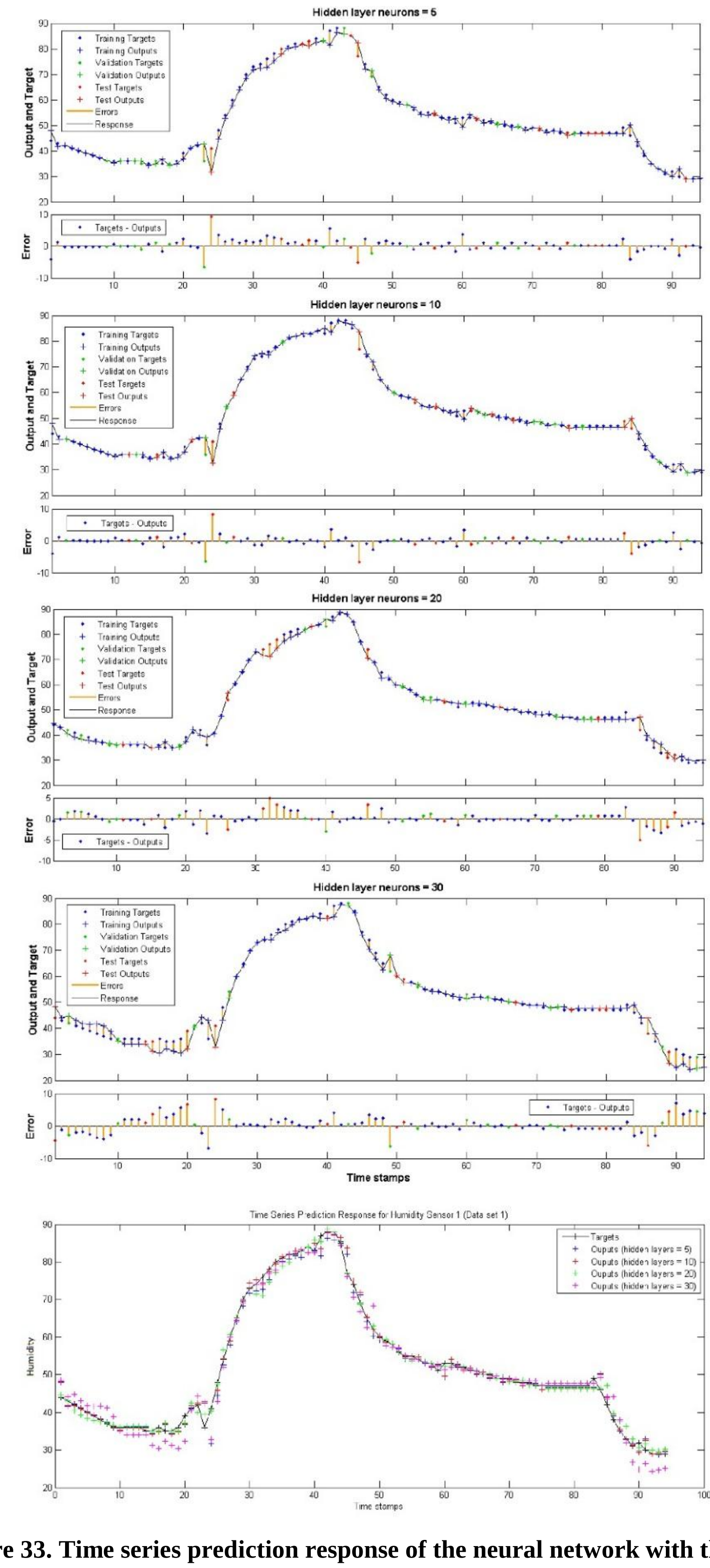

Figure 33. Time series prediction response of the neural network with the error plots for Humidity Sensor 1, Data 1 (Training data = 70%, Validation and Test = 15% each) by varying the size of hidden layer of the network

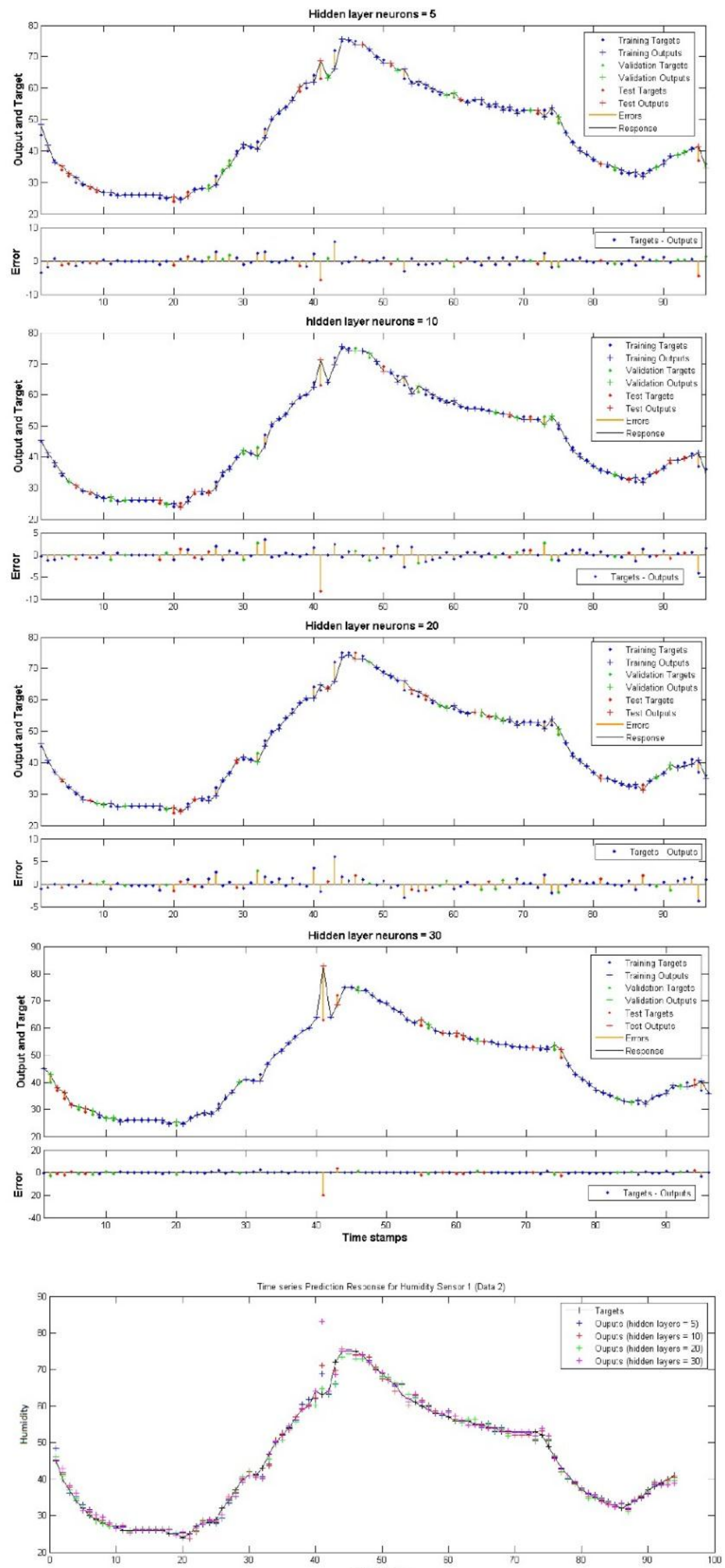

Figure 34. Time series prediction response of the neural network with the error plots for Humidity Sensor 1, Data 2 (Training data = 70%, Validation and Test = 15% each) by varying the size of hidden layer of the network

Figure 36 and Figure 37 represent the network prediction response for the temperature sensor dataset 1 and dataset 2 respectively. The response for the data acquired from temperature sensor placed inside environment chamber is shown in Figure 38.

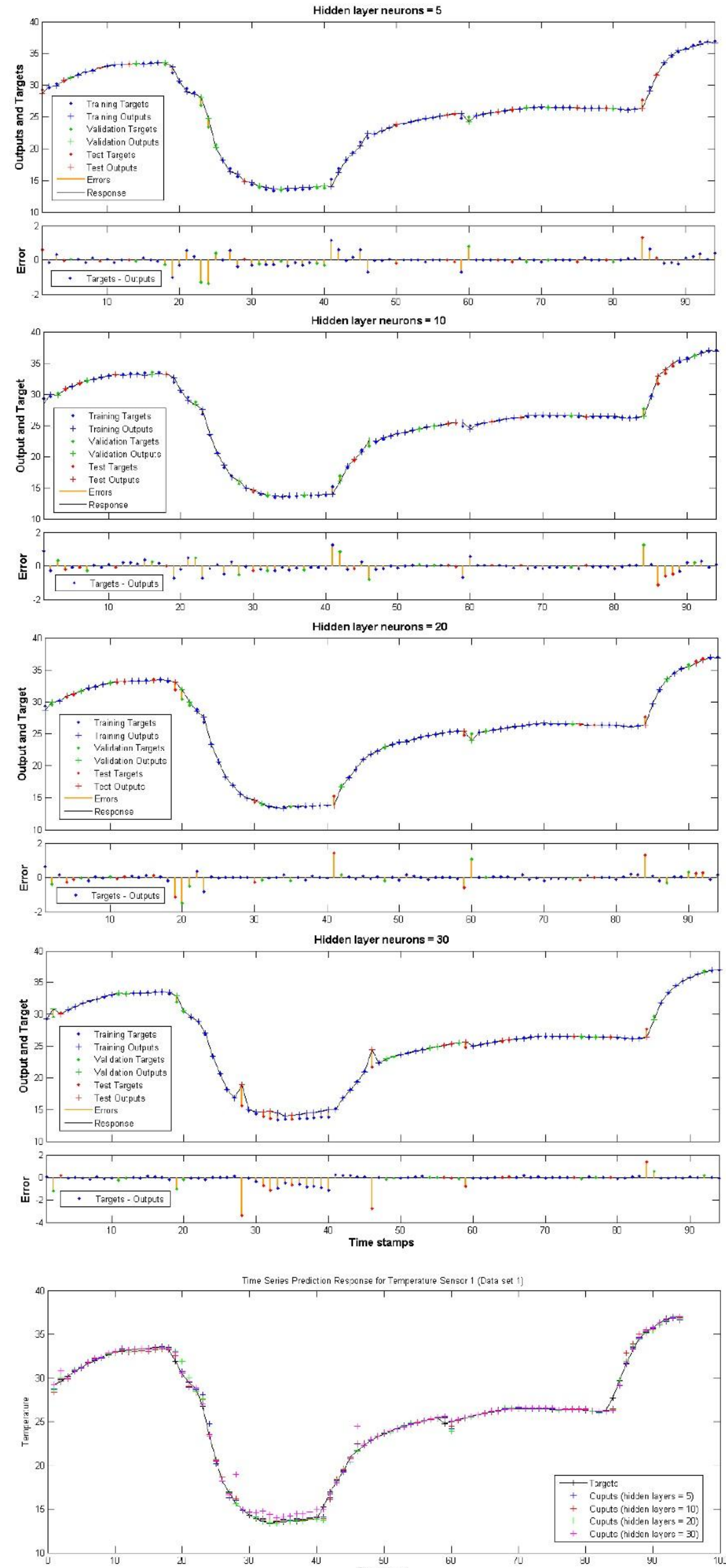

Figure 35. Time series prediction response of the neural network with the error plots for Temperature Sensor 1, Data 1 (Training data 70%, Validation and Test 15% each) by varying the size of hidden layer of the network

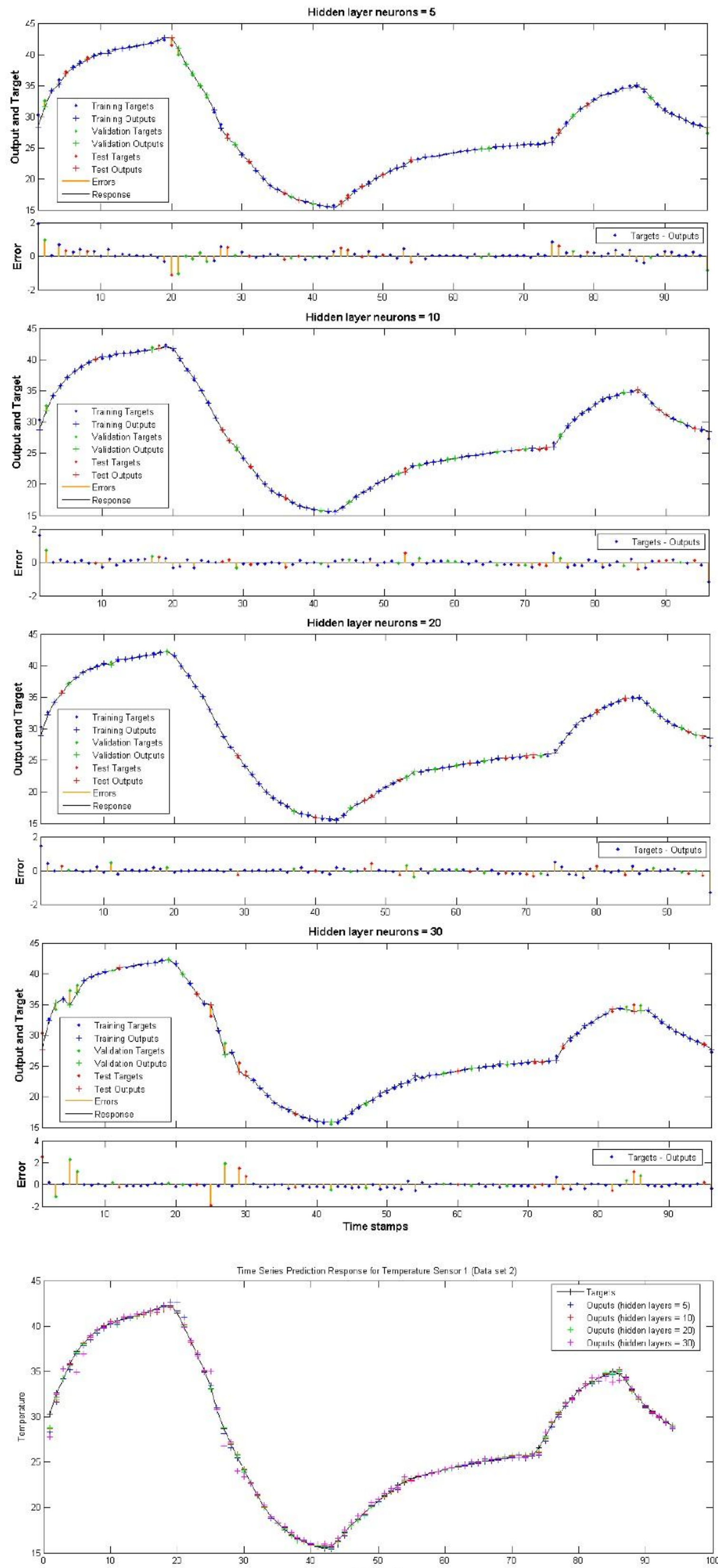

Figure 36. Time series prediction response of the neural network with the error plots for the Temperature Sensor 1, Data 2 (Training data 70%, Validation and Test 15% each) by varying the size of hidden layer of the network

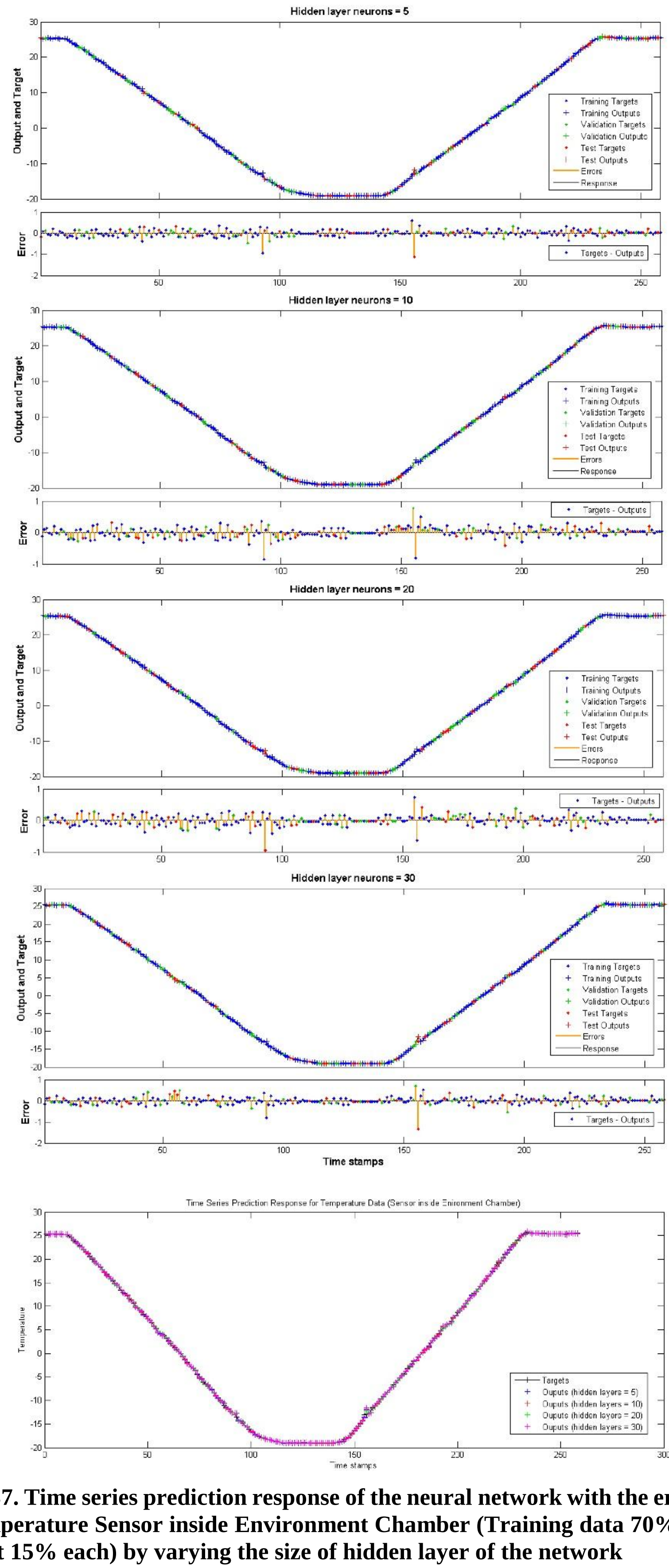

Figure 37. Time series prediction response of the neural network with the error plots for the Temperature Sensor inside Environment Chamber (Training data 70%, Validation and Test 15% each) by varying the size of hidden layer of the network

The mean squared error (MSE) for each of the sensor's data is plotted against variable number of hidden layer neurons, presented in Figure 39.

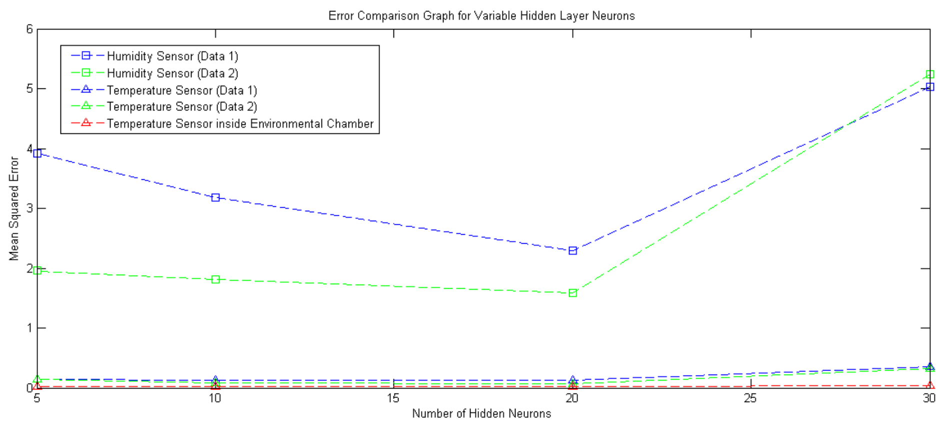

Figure 38. Mean Squared Error plot for each of the sensor's data against different number of hidden layer neurons used in the network

We recorded the network prediction response for each of the sensor's datasets. If we consider the network performance with a particular architecture, it can be seen that the network response is almost similar for each of sensor's datasets. The network shows good prediction even with fewer numbers of hidden neurons (like in case of 5 hidden neurons). With the increase in hidden layer size and the maximum accuracy (corresponding to minimum MSE) is achieved with 20 hidden layer neurons. In the Figure 39 (MSE plot), the continuous error reduction can be noticed up to 20 hidden neurons. Later by selecting 30 hidden neurons, the mean squared error is increased. The problem of overfitting occurred here due to over-parameterization. This overfitting response can be traced in all the sensor's datasets with the choice of 30 hidden layer neurons. On the basis of aforementioned results it can be concluded that among the selected choices, 20 is a good number for hidden neuron. This network architecture estimation was carried out by using two previous outcomes in a NAR system. We wondered what could happen if we changed the number of inputs (delayed outcomes). In this regard, we kept the hidden layer neurons equal to 20 (as estimated), and varied the number of inputs to the network. We changed the number of inputs (subsequent output delays) as 1, 2, 3 and 4. We calculated the results in the same manner as in the previous section. The response of the network is recorded for each of the sensor's data and comparative results are presented by using different number of inputs to the network. Figures 40 and 41 demonstrate the comparative results for humidity sensor, for dataset 1 and dataset 2 respectively. Comparative results for temperature sensor dataset 1 and 2 are presented in Figure 42 and 43 respectively. Figure 44 shows the results for temperature sensor of environmental chamber.

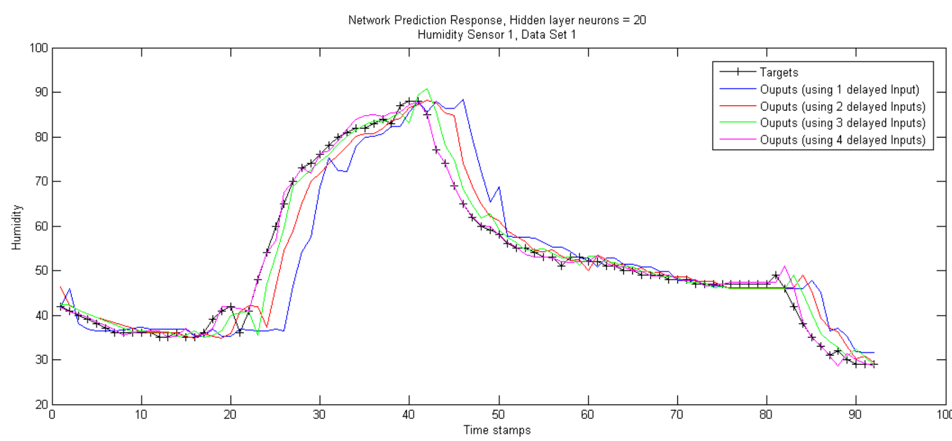

Figure 39. Network prediction response with 20 hidden neurons by varying the number of inputs, Humidity sensor 1, dataset 1

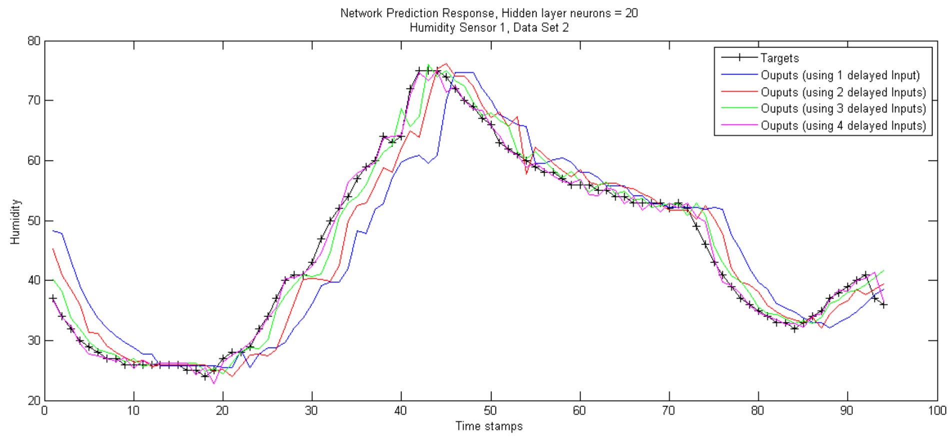

Figure 40. Network prediction response with 20 hidden neurons by varying the number of inputs, Humidity sensor 1, dataset 2

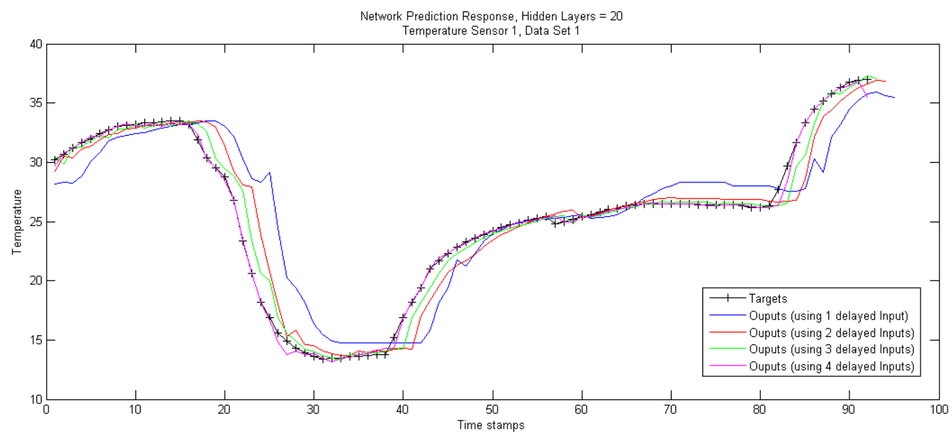

Figure 41. Network prediction response with 20 hidden neurons by varying the number of inputs, Temperature sensor 1, dataset 1

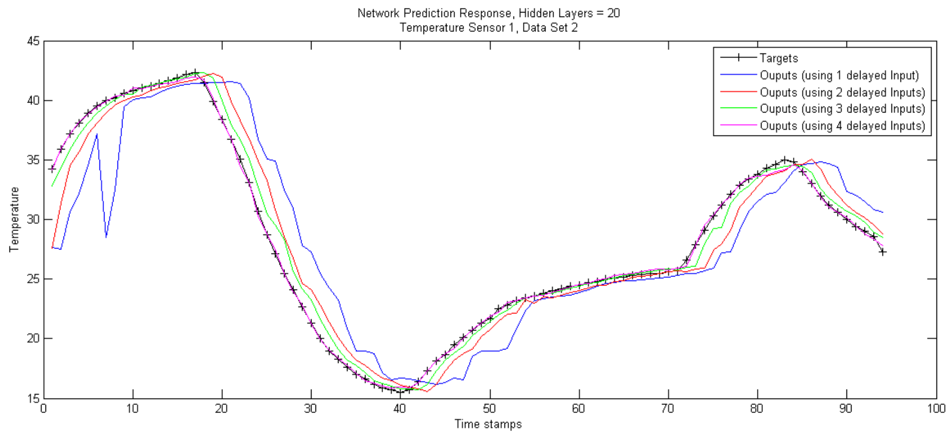

Figure 42. Network prediction response with 20 hidden neurons by varying the number of inputs, Temperature sensor 1, dataset 2

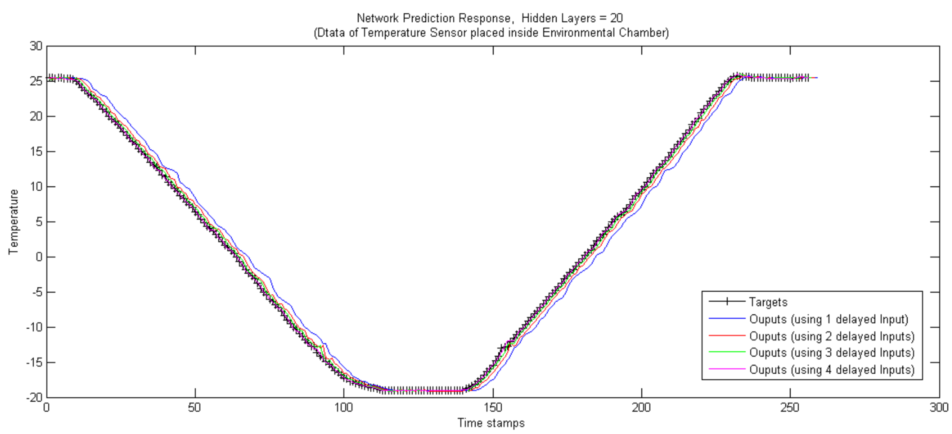

Figure 43. Network prediction response with 20 hidden neurons by varying the number of inputs, Temperature sensor kept inside environmental chamber.

Evidently, network prediction response has been improved for all sensor's datasets by feeding more information to the network. The best response is given by the network when 4 delayed targets are being used as input to the network i.e. more information, better learning and higher accuracy. We calculated the MSE for each of the sensor's data against variable number of network inputs presented in Figure 45.

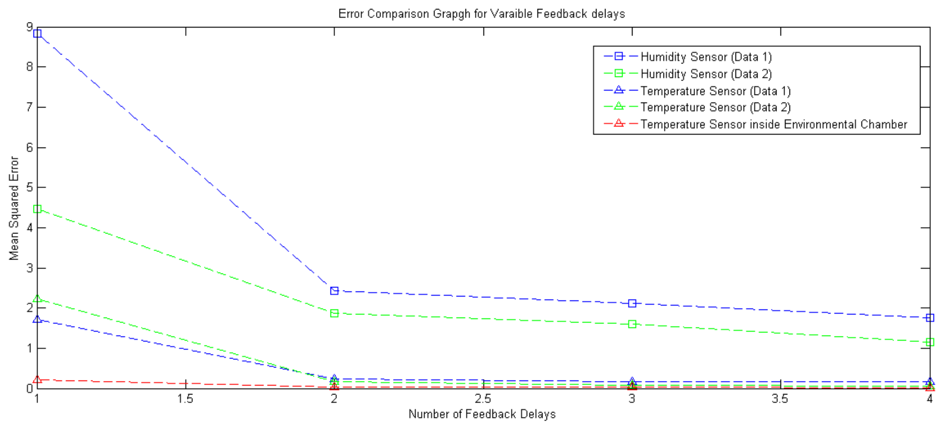

Figure 44. Mean Squared Error plot for each of the sensor's data against different number delayed outcomes used as input to the network

In Figure 45, it can be observed that the error reduces continuously with the increase in number of inputs. A large gradient in the error can be observed upon changing the number of inputs from 1 to 2. However, there is a slight reduction in error with further increase in number of inputs. Therefore by keeping the number of inputs more than 2 did not significantly improve the accuracy. This typical behavior can be observed for all the datasets. On the other hand, the network showed higher error for humidity sensor's data due to its wider range. Hence for the same number of samples, the network performed better for temperature data. After estimating a choice for number of hidden layer neurons earlier equal to 20, we are now interested in finding the best tradeoff between the number of hidden layer neurons and the number of inputs to the network which lead to an optimum solution.

To estimate the existing margin of error in the network setup, we calculated the Mean Absolute Percentage Error (MAPE) as;

$$MAPE = \frac{1}{n}\sum_{k=1}^{n}\left|\frac{T_k - P_k}{T_k}\right| \times 100 \qquad (12)$$

Where Tk represents the target value for timestamp k, and Pk represents the corresponding predicted value, and k = 1,2,3,…n for n number of samples.

The MAPE plot against variable number of network inputs for all datasets is presented in Figure 46. The minimum percentage error was recorded when 4 inputs were fed to the network. Once again it can be observed that reduction in MAPE by choosing more than 2 inputs is very small. Thus, restricting the number of inputs to 2 with 20 neurons in the hidden layer seems to be a favorable solution to achieve the optimized and computationally efficient solution.

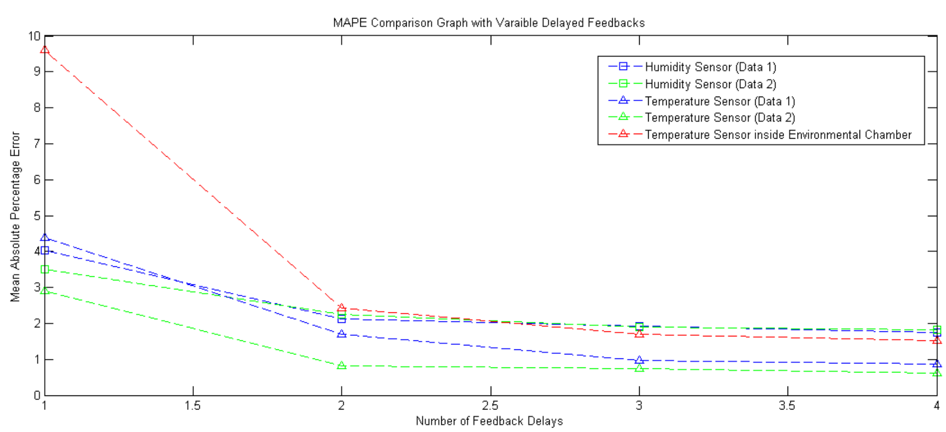

Figure 45. MAPE plot for each of the sensor's data against different number of network inputs

We also calculated the error margin by reducing the size of the network. Figure 47 shows MAPE calculated by choosing different sizes of hidden layer of the network. It is noted that although the error is higher with fewer hidden neuron (5 neurons), however the error gradient by switching from 5 to 20 remains significantly low. With these results we can make a general conclusion that in case of larger tolerable error margin, even a network with few hidden neurons is selectable.

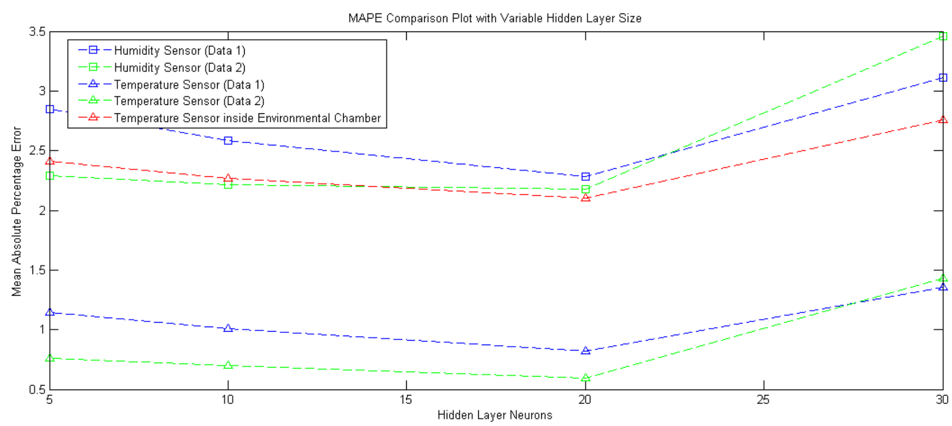

Figure 46. MAPE plot for each of the sensor's data against different number of hidden layer neurons used in the network

Chapter 5

Discussion and Conclusion

This work investigates the possibility of reducing the amount of communications and subsequently the power consumption of a sensor network by elaborating on both data and sensor. Since reading from a Bluetooth-based acquisition system is one of the most expensive task in terms of power consumption in WSNs, energy saving can be obtained by timely replacing read data with predicted data. Reducing the number of measurements or keeping sensor in sleep mode could be beneficial also in general networks, considering that it allows saving power or memory. An innovative and general method is discussed in this this research to determine when, how and which sensor has to be interrogated. The major work is based on a data prediction approach.

1. Bluetooth-based Acquisition System

This phase of the research investigates a low power wireless sensor network to acquire temperature and humidity values from a number of devices built using low cost and low power components, and the Bluetooth communication system for the transmission of the acquired environmental data to a smartphone using an application that automatically connects with them.

The performances of the system were tested by a laboratory test which used a climate chamber for modifying the environmental conditions. Results showed that all the devices connected to the wireless sensor network, correctly reported the environmental condition created by the climatic chamber, and the application for the smartphone properly acquired and stored the data sent by the Bluetooth connection. The environmental data was continuously acquired by the smartphone application during the first experiment without losing absence of signal from any of the devices. This trend indicates that wireless sensor network, is working properly.

Results from the second experiment demonstrated that the current consumption of the Bluetooth based temperature and humidity acquisition system is 1μA while waiting for a transmission and on average 27mA during the transmission to a smartphone. It was calculated that, using a time interval of 15 minutes between two measures and the CR2247 battery, the device can transmit data to a smartphone for more than 4 months. This approach is useful in monitoring climate conditions for small environments, such as laboratories, home rooms, medical spaces, etc., and turn on alarms when the conditions change or go over fixed thresholds. Another possible application of the system presented is the detection of fire in small environments.

2. Using a MLP

Data prediction is also applied in [73], where data is predicted and streamed only when the mismatch with respect to the acquired measurement is higher than a threshold. A similar approach is adopted in [74], where Kalman filters are applied for the prediction. These methods are more useful to conserve bandwidth than to reduce battery consumption. Indeed, the sensors waste energy to perform predictions and continuous sampling, so that they cannot be switched off, but the computational load and cost of each node are increased. Otherwise, only the base station is used in [11] to perform the prediction, not the nodes. However, the sensors are periodically interrogated to test if the predicted value is sufficiently accurate. Another method to reduce power consumption of a WSN is data aggregation [75]. It is an application-specific technique considered in most cases where data is transferred between intermediate or neighboring nodes.

The proposed algorithm estimates a prediction uncertainty for each sensor in the network during the monitoring. A specific sensor is interrogated when its uncertainty is above a threshold, which can be selected by the user (allowing, for example, to fit better a specific dataset or to impose a deeper undersampling of a sensor). The algorithm to estimate the sensor uncertainties is based on a tool for data forecasting. It is used to estimate the rate of increasing to the prediction error and the future dispersion of the predictions due to the uncertainty contained in the available data (due to the finite precision of the sensors or to the errors cumulated by iterating the prediction). Two contributions (related to the predicted errors and to the dispersion of the predictions) are given the same weight and are linearly combined for the estimation of the uncertainty associated to the sensor.

The method was tested on both simulated and experimental data. Simulated data were examined to analyze the algorithm correctness. Different simulations were considered, in order to analyze the following properties:

1. The performance of the proposed method on non-stationary signals
2. The ability of integrating the information from correlated measurements
3. The management of chaotic versus periodic signals

The method adapted the sampling rate to the properties of a non-stationary signal, so that more samples were required for the portions of the signal with higher frequency. Moreover, when applied to two correlated signals, the method improved the performances with respect to the case in which it was applied on the two signals separately. Finally, the method required more samples to describe a chaotic system than a simple periodic one. All these results are in line with our expectations and confirm the reliability of the proposed method.

Two applications of the method on experimental data are also provided. When applied to meteorological data, the method was able to reduce the number of acquired samples with low estimation errors. More samples were recorded from the sensor monitoring the wind velocity, which provided a very erratic signal, with respect to temperature, pressure, and humidity, which showed regular and correlated variations. Notice that only a representative application is here considered: for practical applications, as only average information on wind velocity is usually of interest, subsequent measured or estimated samples could be averaged, further reducing the data to be effectively transmitted. This outdoor application is in line with the results of the application of our method to indoor environmental data from a WSN.

Considering the power consumed by the sensors during transmission and while being in the idle state, some considerations could be made on the power that could be saved using our algorithm to reduce the number of measurements (see Appendix B). Considering the indoor application, a reduction of the 50% of samples (getting an estimation error of about 35%,) allows to decrease the power consumption of about 7.5%; for the outdoor application here considered, data could be reduced to 70% (guaranteeing an estimation error lower than 20 %); thus, by scaling the acquisition and sampling times, a 10% of power saving could be obtained.

The results of the application of our method appear to be promising in light of the basic and general method considered within the frame of this research. Following the same ideas, more sophisticated methods could be developed, in order to better fit specific applications. For example, only the last two (measured or predicted) samples are here considered as the inputs of the prediction algorithm. This choice is due to the general applications discussed here, where four different datasets were processed by the same algorithm. However, different inputs can be chosen (e.g., the average values of data on long periods, often used in meteorological forecasting applications, or delayed samples with an optimally chosen delay, or simply more than two values could be used from each sensor; the methods of time series embedding [76] could be used to support a proper selection of the optimal delay and of the number of delayed values to characterize better each sensor). Moreover, a simple MLP was used for data prediction (see Appendix C). Different alternative methods relevant to the main idea of this paper could be applied instead of the proposed method. For example, different neural networks or fuzzy rule-based systems can be used [77]. Also, a single MLP is used here to predict all the measurements of the sensors, but different MLPs could be used, one for each sensor. The method estimates the uncertainty of the predicted measurements as the average of two contributions: different combinations can fit specific applications better. Moreover, a linear increase of the prediction error, including a memory term, is assumed here, but a more sophisticated (nonlinear, adaptive) algorithm can be introduced in the future to estimate better the raise of the prediction error over the time.

3. Using NAR ; High Reduction

We devise a data driven approach to reduce power consumption in wireless sensor networks. The method is based on the prediction of sensed data using non-linear autoregressive neural networks. Evaluation is also performed using the actual data obtained from temperature and humidity sensors. The performance of the network is assessed under different conditions. In fact, we feed two delayed targets to the network and change the size of the hidden layer. The results imply that the most accurate forecast is obtained with 20 hidden layers. Another observation pertains to the number of inputs. The results show that increasing the number of inputs does not lead to a significant decline in the error rate.

The experiments conducted in this research indicate that our method substantially reduces power consumption in wireless sensor networks. Implementing the proposed method in real-life sensor networks will help prevent unnecessary sampling and, in turn, reduces energy and costs. There is still much to be done. More theoretical work on the proposed method and characterizing the environments for which the method leads to satisfactory results deserve future research. However, in our second step in the last phase, we achieved higher reduction on data, consequently on power consumption with reducing error as well.

Also here, we used the neural network for time series prediction of the sensor's data. Hereby, we selected the NAR which performs time series prediction by using the target values at subsequent delayed time stamps as inputs, and predicts the value at the current time stamp. Initially, we analyzed the performance (based on MSE) with the aforementioned data to estimate a good network architecture with optimum choice of hidden layer neurons. The network performance was better for temperature sensor data, set 1 and set 2, which correspond to indoor environment readings and range between 15C and 42C (Δ=27C). There was no significant improvement recorded in accuracy with the increase in hidden layer size. The network showed good performance even with fewer hidden layer neurons. On the contrary, the error recorded for humidity sensor data (ranges 20-90, Δ= 70) was higher. The range of humidity data is quite larger than that of temperature with the same number of samples. Consequently for humidity sensor, the network performed better with larger hidden layer size. The environmental chamber's sensor data contains large number of samples with linear change in temperature, so the network outperformed for this data even with the smallest network size.

Later, the data was used to analyze the network performance by altering the size of input data fed to the network. While keeping the hidden layer size equal to 20, the network was fed with 1, 2, 3 and 4 subsequent delayed targets alternatively, and the performance was recorded. Continuous reduction in the error was observed as a result, however it wasn't significantly reduced by using more than two inputs. With 20 hidden neurons and feeding 2 delayed target values as input, we calculated the MAPE to estimate the error margin projected by the network. The network showed the MAPE upto 1.6% for the indoor temperature sensor and 2.2% for the indoor humidity sensor. At the same time, the error margin for the sensor in the chamber was recorded as 2.4%. It may be thus concluded that the network provides lower error even with small architecture. The reduction in percentage error by increasing the hidden layer size is less than 1% for humidity sensor, and even lesser for temperature sensor. Hence, keeping the number of inputs to 2 with 20 hidden neurons produces the optimized results. However, the network can be restricted to

5 hidden layer neuron by compromising 1% of error. This can be adopted to improve computational efficiency in the case where there is a large error margin is allowed.

Regarding the power saving, it is obvious that less communication with the sensor corresponds to more power saving. By selecting the optimum network architecture with 20 hidden neurons, test data prediction is done for every sample by using two delayed feedbacks as input. Hence to predict a target value (of temperature or humidity) with the presented accuracy, the network requires targets at previous 2 timestamps. In this way 66.6% is the communication time, while 33.3% is the prediction time. In view of the change in temperature in an indoor environment is small, it is possible to minimize communication with the sensor to save more power. In case of predicting temperature once in 8 timestamps, the communication time is 25% (2/8*100) at t1 and t2. The idle time is 75% (6/8*100), prediction at t3 and no communication for t4-t8. For the humidity sensor where there is a large range of observations, the idle time can be reduced to maintain the accuracy. Hence we can conclude that for the temperature sensors, upto 75% of power can be saved with an error margin of 2.6% (1.6% by network + 1% of the sensor). Similarly for the humidity sensor, 66% of power can be saved for predicting once in 6 timestamps, with an error margin of 3.2% (2.2% by network + 1% of the sensor). For the temperature sensor inside the environmental chamber, 75% of power can be saved within an error margin of 3.6% (2.6% by network + 1% of the sensor).

References

1. Rone Ilidio da Silva, V Del Duca Almeida, Andre Marques Poersch, and José Marcos S Nogueira. Wireless sensor network for disaster management. In Network Operations and Management Symposium (NOMS), pages 870–873. IEEE, 2010.

2. Dalei Wu, Song Ci, Haiyan Luo, Yun Ye, and Haohong Wang. Video surveillance over wireless sensor and actuator networks using active cameras. Automatic Control, IEEE Transactions on, 56(10):2467–2472, 2011.

3. Sunyoung Kim and Eric Paulos. InAir: sharing indoor air quality measurements and visualizations. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, pages 1861–1870. ACM, 2010.

4. Siamak Aram, Amedeo Troiano, and Eros Pasero. Environment sensing using smartphone. In Sensors Applica- tions Symposium (SAS), 2012 IEEE, pages 1–4. IEEE, 2012.

5. Siamak Aram, Amedeo Troiano, Francesco Rugiano, and Eros Pasero. Low power and bluetooth-based wireless sensor network for environmental sensing using smartphones. In Artificial Intelligence Applications and Innovations, pages 332–340. Springer, 2012.

6. Jeffrey Goldman, Katie Shilton, Jeff Burke, Deborah Estrin, Mark Hansen, Nithya Ramanathan, Sasank Reddy, Vids Samanta, Mani Srivastava, and Ruth West. Participatory sensing: A citizen-powered approach to illuminating the patterns that shape our world. Foresight & Governance Project, White Paper, pages 1–15, 2009.

7. Abdellah Chehri, Paul Fortier, and P-M Tardif. Security monitoring using wireless sensor networks. In Communication Networks and Services Research, 2007. CNSR'07. Fifth Annual Conference on, pages 13– 17. IEEE, 2007.

8. Giuseppe Anastasi, Marco Conti, Mario Di Francesco, and Andrea Passarella. Energy conservation in wireless sensor networks: A survey. Ad hoc networks, 7(3):537–568, 2009.

9. Hnin Yu Shwe, Xiao Hong Jiang, and Susumu Horiguchi. Energy saving in wireless sensor networks. Journal of Communication and Computer, 6(5):20–27, 2009.

10. Amjad Abu-Baker, Hong Huang, and Satyajayant Misra. Maximizing lifetime sequences of wireless sensor networks powered by renewable energy. Sensor Review, 34(4):374–381, 2014.

11. Mario Cordina and Carl James Debono. Maximizing the lifetime of wireless sensor networks through intelligent clustering and data reduction techniques. In Wireless Communications and Networking Conference, 2009. WCNC2009. IEEE, pages 1–6. IEEE, 2009.

12. Tomoya Enokido, Ailixier Aikebaier, and Makoto Takizawa. Computation and transmission rate based algorithm for reducing the total power consumption. Journal of Wireless Mobile Networks, Ubiquitous Computing, and Dependable Applications (JoWUA), 2(2):1–18, 2011.

13. Siamak Aram, Amedeo Troiano, Francesco Rugiano, and Eros Pasero. 2014. Mobile environmental sensing using smartphones. In: Eren, H. and Webster, J.,Measurement, Instrumentation, and Sensors Handbook, Second Edition: Electromagnetic, Optical, Radiation, Chemical, and Biomedical, Measurement. Boca Raton, FL: CRC Press, Ch. 73, 2014. 15th January 2014.

14. Siamak Aram, Luca Mesin, and Eros Pasero. Improving lifetime in wireless sensor networks using neural data prediction. In 2014 World Symposium on Computer Applications & Research - IEEE International Conference on Information and Intelligent Systems.18-20 Jan 2014, Sousse.

15. Luca Mesin, Siamak Aram and Eros Pasero. A neural data-driven approach to increase wireless sensor networks' lifetime. In 2014 World Symposium on Computer Applications & Research – IEEE International Conference on Artificial Intelligence.18-20 Jan 2014; Sousse.

16. Luca Mesin, Siamak Aram, and Eros Pasero. A neural data-driven algorithm for smart sampling in wireless sensor networks. EURASIP Journal on Wireless Communications and Networking, 2014(1):1–8, 2014.

17. Siamak Aram, Ikramullah Khosa, and Eros Pasero. Conserving energy through neural prediction of sensed data. Journal of Wireless Mobile Networks, Ubiquitous Computing nnd Dependable Applications, 6(1), 2015.

18. Barnard Etienne. Optimization for training neural nets. IEEE Transaction on Neural Networks 3(2): 232–240, 1991.

19. TWS Chow and CT Leung. Nonlinear autoregressive integrated neural network model for short-term load forecasting. In Generation, Transmission and Distribution, IEE Proceedings-, volume 143, pages 500–506. IET, 1996.

20. Timo Paananen. Smartphone cross-platform frameworks: A case study. Bachelor's Thesis, Jamk University of applied sciences 2011.

21. Maximiliano Firtman. Programming the Mobile Web. O'Reilly Media. 2010.

22. Wenxi Chen, Daming Wei, Xin Zhu, Masayuki Uchida, Shuxue Ding, and Michael Cohen. A mobile phone- based wearable vital signs monitoring system. In Computer and Information Technology, 2005. CIT 2005. The Fifth International Conference on, pages 950–955. IEEE, 2005.

23. E Staples. EnviSensor: Environmental Data Collection via Participatory Sensor Networks Utilizing Mobile Devices. Master's Thesis, University of Edinburgh, 2011.

24. Jin-Shyan Lee, Yu-Wei Su, and Chung-Chou Shen. A comparative study of wireless protocols: Bluetooth, UWB, ZigBee, and Wi-Fi. In Industrial Electronics Society, 2007. IECON 2007. 33rd Annual Conference of the IEEE, pages 46–51. IEEE, 2007.

25. Saadi Boudjit, Nadjim Chelghoum, Miloud Otsmani, and Salim Allal. Multi-sensors' data gathering management system for a wireless health monitoring platform. In Proceedings of the First ACM MobiHoc Workshop on Pervasive Wireless Healthcare, page 4. ACM, New York. 2011.

26. Alan Lapedes and Robert Farber. Nonlinear signal processing using neural networks: Prediction and system modelling. Technical report LA-UR-872662, Los Alamos Nat. Lab. 1987.

27. Eugen Diaconescu. The use of NARX neural networks to predict chaotic time series. WSEAS Transactions on Computer Research, 3(3):182–191, 2008.

28. Holger Kantz and Thomas Schreiber. Nonlinear time series analysis, volume 7. Cambridge university press, 2004.

29. TWS Chow and CT Leung. Nonlinear autoregressive integrated neural network model for short-term load forecasting. In Generation, Transmission and Distribution, IEE Proceedings-, volume 143, pages 500–506. IET, 1996.

30. James K Lein. Environmental sensing: analytical techniques for earth observation. Springer Science & Business Media, 2011.

31. Hussein, K. Remote Sensing of the Environment, PDF file, Cooperative Institute for Research in Environmental Science, University of Colorado at Boulder, Boulder, CO. 2006.

32. Bureau for Africa's Office of Sustainable Development. Environmental Guidelines for Small-Scale Activities in Africa: Environmentally Sound Design for Planning and Implementing Development Activities, 2nd ed., Bureau for Africa's Office of Sustainable Development. 2007.

33. Zheng, Pei, and Lionel Ni. Smart Phone & Next Generation Mobile Computing, Elsevier Oxford, U.K. 2010.

34. Wenxi Chen, Daming Wei, Xin Zhu, Masayuki Uchida, Shuxue Ding, and Michael Cohen. A mobile phone- based wearable vital signs monitoring system. In Computer and Information Technology, 2005. CIT 2005. The Fifth International Conference on, pages 950–955. IEEE, 2005.

35. Hendika Sugiharto. Current and future mobile platforms. SNET Master Project, 2010.

36. Nicholas D Lane, Emiliano Miluzzo, Hong Lu, Daniel Peebles, Tanzeem Choudhury, and Andrew T Camp- bell. A survey of mobile phone sensing. Communications Magazine, IEEE, 48(9):140–150, 2010.

37.Tathagata Das, Prashanth Mohan,Venkata N Padmanabhan, Ramachandran Ramjee, and Asankhaya Sharma. Prism: platform for remote sensing using smartphones. In Proceedings of the 8th international conference on Mobile systems, applications, and services, pages 63–76. ACM, 2010.

38. Tanzeem Choudhury, Sunny Consolvo, Beverly Harrison, Jeffrey Hightower, Anthony LaMarca, Louis LeGrand, Ali Rahimi, Adam Rea, G Bordello, Bruce Hemingway, et al. The mobile sensing platform: An embedded activity recognition system. Pervasive Computing, IEEE, 7(2):32–41, 2008.

39. Min Mun, Sasank Reddy, Katie Shilton, Nathan Yau, Jeff Burke, Deborah Estrin, Mark Hansen, Eric Howard, Ruth West, and Peter Boda. PEIR; the personal

environmental impact report, as a platform for participatory sensing systems research. In Proceedings of the 7th international conference on Mobile systems, applications, and services, pages 55–68. ACM, 2009.

40. Mohammad Faiz Liew Abdullah and Lee Mei Poh. Mobile robot temperature sensing application via blue- tooth. International Journal of Smart Home, 5(3):39–48, 2011.

41. Radek Kuchta, Radimir Vrba, and Vladimir Sulc. Smart platform for wireless communication-case study. In ICN, pages 117–120, 2008.

42. Alberto Rosi, M Berti, N Bicocchi, G Castelli, A Corsini, M Mamei, F Zambonelli, et al. Environmental monitoring and task-driven computing. IEEE Pervasive Computing, 9:48–49, 2010.

43. Qingshan Shanand David Brown. Wireless temperature sensor using Bluetooth. Proceeding IWWAN, London, UK, 2005.

44. Ricardo Carrano, Diego Passos, Luiz Claudio Schara Magalhaes, and Célio VN Albuquerque. Survey and taxonomy of duty cycling mechanisms in wireless sensor networks. Communications Surveys & Tutorials, IEEE 16(1): 181-194, 2014.

45. Ganesan, Deepak, Alberto Cerpa, Wei Ye, Yan Yu, Jerry Zhao, and Deborah Estrin. Networking issues in wireless sensor networks. Journal of Parallel and Distributed Computing. 64(7): 799-814. 2004.

46. Warrier Ajit, Sangjoon Park, Jeongki Min, and Injong Rhee. "How much energy saving does topology control offer for wireless sensor networks?–A practical study." Computer Communications. 30 (14):2867-2879. 2007.

47. Biljana Stojkoskaand, Kliment Mahoski. Comparison of different data prediction methods for wireless sensor networks. CIIT, Bitola. 2013.

48. Amol Deshpande, Carlos Guestrin, Samuel R Madden, Joseph M Hellerstein, and Wei Hong. Model-driven data acquisition in sensor networks. In Proceedings of the Thirtieth international conference on Very large data bases-Volume 30, pages 588–599. VLDB Endowment, 2004.

49. David Chu, Amol Deshpande, Joseph M Hellerstein, and Wei Hong. Approximate data collection in sensor networks using probabilistic models. In Data Engineering, 2006. ICDE'06. Proceedings of the 22nd International Conference on, 2006.

50. Bhargav Kanagal, and Amol Deshpande. Online filtering, smoothing and probabilistic modeling of streaming data. In Data Engineering, 2008. ICDE 2008. IEEE 24th International Conference on, pages 1160-1169. IEEE, 2008.

51. G.E.P. Box and G.M. Jenkins. Time series analysis: forecasting and control. Holden-Day series in time series analysis and digital processing. Holden-Day, 1976.

52. Daniela Tulone, and Samuel Madden. PAQ: Time series forecasting for approximate query answering in sensor networks. In Wireless Sensor Networks, pages 21-37. Springer Berlin Heidelberg, 2006.

53. Daniela Tulone and Samuel Madden. An energy-efficient querying framework in sensor networks for detecting node similarities. In Proceedings of the 9th ACM international symposium on Modeling analysis and simulation of wireless and mobile systems, pages 191–300. ACM, 2006.

54. Qi Han, Sharad Mehrotra, and Nalini Venkatasubramanian. Energy efficient data collection in distributed sensor environments. In Distributed Computing Systems, 2004. Proceedings. 24th International Conference on, pages 590-597. IEEE, 2004.

55. Samir Goel and Tomasz Imielinski. Prediction-based monitoring in sensor networks: taking lessons from mpeg. ACM SIGCOMM Computer Communication Review, 31(5):82–98, 2001.

56. Yann-Ae¨l Le Borgne, Silvia Santini, and Gianluca Bontempi. Adaptive model selection for time series prediction in wireless sensor networks. Signal Processing, 87(12):3010–3020, 2007.

57. Silvia Santini and Kay Romer. An adaptive strategy for quality-based data reduction in wireless sensor networks. In Proceedings of the 3rd International Conference on Networked Sensing Systems (INSS 2006), pages 29–36, 2006.

58. Biljana Stojkoska, Dimitar Solev, and Danco Davcev. Data prediction in WSN using variable step size LMS algorithm. In SENSORCOMM 2011, The Fifth International Conference on Sensor Technologies and Applications, pages 191–196, 2011.

59. Biljana Risteska Stojkoska, Dimitar Solev, and Danco Davcev. Variable step size LMS algorithm for data prediction in wireless sensor networks. Sensors & Transducers. Volume 14. 2012.

60. Oded Maron and Andrew W Moore. The racing algorithm: Model selection for lazy learners. Springer, 1997.

61. William Wu-Shyong Wei. Time series analysis. Addison-Wesley publication, 1994.

62. Chong Liu, Kui Wu, and Min Tsao. Energy efficient information collection with the ARIMA model in wireless sensor networks. In Global Telecommunications Conference, 2005. GLOBECOM'05. IEEE, volume 5, pages 2470–2474. IEEE, 2005.

63. Guiyi Wei, Yun Ling, Binfeng Guo, Bin Xiao, and Athanasios V Vasilakos. Prediction-based data aggregation in wireless sensor networks: Combining grey model and Kalman filter. Computer Communications, 34(6):793–802, 2011.

64. Feng Xia, Zhenzhen Xu, Lin Yao, Weifeng Sun, and Mingchu Li. Prediction-based data transmission for energy conservation in wireless body sensors. In The International Workshop on Ubiquitous Body Sensor Networks (UBSN), in conjunction with the 5th Annual International Wireless Internet Conference (WICON), Singapore, March 2010, 2010.

65. Young Soo Suh. Send-on-delta sensor data transmission with a linear predictor. Sensors, 7(4):537–547, 2007.

66. Jialiang Lu, Fabrice Valois, Mischa Dohler, and Min-You Wu. Optimized data aggregation in WSNs using adaptive ARMA. In Sensor Technologies and Applications (SENSORCOMM), 2010 Fourth International Conference on, pages 115–120. IEEE, 2010.

67. Surender Kumar Soni, N Chand, and P Singh. Reducing the data transmission in WSNs using time series prediction model. In Signal Processing, Computing and Control (ISPCC), 2012 IEEE International Conference on, pages 1–5. IEEE, 2012.

68. Edward N. Lorenz. Deterministic nonperiodic flow. Journal of the atmospheric sciences. 20(2): 130-141. 1963.

69. Amedeo Troiano, Eros Pasero, and Luca Mesin. In the Field Application of a New Sensor for Monitoring Road and Runway Surfaces. Sensors & Transducers. 10(2): 71–83. 2011.

70. Simon Haykin. Neural Networks: A Comprehensive Foundation. Prentice Hall PTR, Upper Saddle River, NJ, USA, 2nd edition, 1998.

71. Ananth Ranganathan. The levenberg-marquardt algorithm. Tutoral on LM Algorithm, pages 1–5, 2004.

72. Jorge J Moré. The levenberg-marquardt algorithm: implementation and theory. In Numerical analysis, pages 105–116. Springer, 1978.

73. Vibhore Kumar, Brian F. Cooper, and Shamkant B. Navathe. Predictive filtering: a learning-based approach to data stream filtering. In Proceeedings of the 1st international workshop on Data management for sensor networks: in conjunction with VLDB 2004, pages 17-23. ACM, 2004.

74. Ankur Jain, Edward Y. Chang, and Yuan-Fang Wang. Adaptive stream resource management using Kalman filters. In Proceedings of the 2004 ACM SIGMOD international conference on Management of data, pages11-22. ACM, 2004.

75. Jamal N. Al-Karaki, Raza Ul-Mustafa, and Ahmed E. Kamal. Data aggregation in wireless sensor networks-exact and approximate algorithms. In High Performance Switching and Routing, 2004. HPSR. 2004 Workshop on, pages 241-245. IEEE, 2004.

76. Holger Kantz and Thomas Schreiber. Nonlinear Time-series Analysis. Journal of Biological Physics. 24(1):79-80. 1998.

77. B. Meenakshi, and P. Anandhakumar. Optimization of energy consumption by fuzzy based routing in wireless sensor network. IOSR Journal of Electronics and Communication Engineering. 2(1): 31-35. 2012.

Appendix A

Bluetooth as communication Protocol

The next section of the thesis compares and analyzes a variety of protocols in order to identify the most effective and efficient protocol. An overview of the protocols embraced by the scope of research at hand will be provided below. The overview shows that among all the alternatives, Bluetooth is best matched to the objectives of this study.

i. Wired connections

Generally speaking, poor connectivity problems which arise in the course of interaction between the smartphone and related sensor may be potentially solved with the aid of wired connections. However, even though wired connection is considered as more or less resultant technique, its stability remains dependent on the individual interface and other peculiarities of the smartphone.

For instance, the capacity of the smartphone to connect to other devices externally is still rather limited with regard to the fact that it only has a single USB port.

Therefore, USB hubs will be inevitably required should there be a need to enhance the smartphone's external connectivity. Besides, modern smartphones almost never incorporate any interface (i.e. RS232 serial ports), partly for the reason that such interface would demand incorporating extra hardware.

In view of the above, wired connection objectively appears as an overly complex method for Smartphone applications. Even though this way is rather inconvenient, it may be applied in the research purposes or for development of new prototypes. Wired connections typically rely on SPP (Serial Port Profiles) based on UART standards. The aforementioned protocols were as well adopted in the *aBluSen,* which was devised within the frame of the present research but wirelessly. Another distinguishing characteristics of wired connections is the speed of data processing, which may reach around 920.16 kbits/sec. Such speed is considered sufficient enough for performing operational tasks of the sensing applications.

ii. Wi-Fi

Unlike wired connectivity and Bluetooth, Wi-Fi (IEEE 801.2) protocols necessarily include complete stack of TCP/IP. Since TCP/IP is also commonly used for Ethernet and tools related to it, we can infer some similarity between Wi-Fi and other applications enabled by Wireless Local Area Networks (WLANs) and Wide Area Networks (WANs). In this case, location and placement of the sensor merely plays a supplementary role. To illustrate, hypothetically TCP/IP connectivity may allow global access to the Wi-Fi-enabled sensor provided that its IP number is recognizable by WLAN. Consequently, it is no longer necessary to make any physical connection between the devices or conduct any immediate management tasks. In comparison to Bluetooth, Wi-Fi protocols also have the following advantages:

1) Compatibility with more advanced encryption techniques;

2) Faster data processing;

3) More robust external connections achieved by reliance on TCP/IP tools.

iii. ZigBee

Stages of designing sensors usually envisage early assessment of applicable wireless components. Accordingly, ZigBee protocols, as one of the available alternatives, suffice for the purposes of this preliminary analysis. In terms of their operational principles, ZigBee protocols slightly resemble RS-232 and Bluetooth interfaces. Additionally, point-to-point connection established with the aid of ZigBee covers up to 1-km distance. Nonetheless, Zigbee is a niche instrument which demands individually selected tools. Accordingly, inherent smartphone abilities may not be sufficient to support a ZigBee-based network.

iv. Modems

Potentially, a sensor may be upgraded by installation of a modem for data processing. In this respect, there exist two types of suitable data modems:

1) Modems which operate on a landline;

2) Modems which establish connection via mobile technology.

Such type of modems is capable of establishing communications as listed below:

1) Via switched circuit

2) Via dialing a phone number

3) Via exploiting the in-built capacities of trafficking the data (reliance on TCP/IP networking, accessing the Internet).

Both types of modems above may apply when measurement stations and motes controlled need from the distance. As opposed to that, technical impediments render these tools impractical for individual sensing systems.

v. Bluetooth

Having rapidly gained popularity in the sphere of smartphones, Bluetooth tools appear to be well suited for performing individual sensing tasks. Notwithstanding positive feedbacks on adoption of Bluetooth technology, connection along with the communicational protocols should be established solely on an ad hoc basis. Communications handled via Bluetooth may be one-way and two-way. In the first case the sensor carries out its activities autonomously. By contrast, in the second case the sensor commences its work upon receiving the command from external sources. Evidently, these commands are transferred to the sensor through Bluetooth. Referring to the means of enabling Bluetooth connectivity, the following methods should be considered:

Bluetooth-operated devices may be connected between each other mainly through the following ways:

1) Serial Soft Profiles;

2) TCP/IP packet-based for configuring relevant applications with Bluetooth;

3) Profiles for running specific devices, such as cameras, scanners, applications, headsets etc.

Correspondingly, majority of the applications tuned for sensing implement Bluetooth SPP owing to its compatibility with proprietary formatting tasks. To understand the distinctive characteristics of Bluetooth, more detailed analysis of Bluetooth-enabled connections is required. At that, several important points were identified, these ones in particular:

a) In all transmissions except for the multicast ones Bluetooth solution may connect transmitter to the receiver only on a point-to-point basis;

b) Multi-sensors may be potentially applied to a single mobile device/smartphone;

c) Bluetooth-based nodes strictly requires Bluetooth-based receivers, otherwise the connection will not be established.

d) Default abilities of hardware used for Bluetooth transmissions determine whether multi-connections can be initiated on the same device.

In total, point-to-point connections may offer a number of prospective benefits, such as enhanced confidentiality and protection of the data involved in sensor transmissions. Privacy of the information is safeguarded by narrowing the transmission ranges down to areas in close proximity to the transmission base. This solution can be an asset for such technologies as vehicle sensors, applications used for military purposes, telemetry and other areas.

vi. Conclusion

The results of comparative analysis between Wi-Fi, ZigBee and Bluetooth are contained in Table 4. Evidently, low cost, moderate complexity, ubiquitousness, accessibility, reasonable energy costs and robust security mechanism allows us to conclude that Bluetooth module works best for the sensor transmissions especially for this thesis scenarios. Even though ZigBee has lesser energy costs, Bluetooth matches to the requirements of most smartphones, while ZigBee cannot be utilized by a smartphone without installation of rare hardware which may not be easily found in the market.

Table 4. Wireless communication protocols comparison

Standard	**Bluetooth**	**ZigBee**	**Wi-Fi**
IEEE spec.	802.15.1	802.15.4	802.11a/b/g
Max signal rate	1 Mb/s	250 Kb/s	54 Mb/s
Complexity	Complex	Simple	Very Complex
Nominal range	10 m	10 - 100 m	100 m
Number of RF channels	79	1/10; 16	14 (2.4 GHz)
Max number of cell nodes	8	> 65000	2007
Encryption	E0 stream cipher	AES block cipher	RC4 stream cipher (WEP),AES block cipher
Authentication	Shared secret	CBC-MAC (ext. of CCM)	WPA2 (802.11i)
Data protection	16-bit CRC	16-bit CRC	32-bit CRC
Security protocols/Models	Security Modes	High security mode" and "Standard security mode"	WEP and WPA
Power [Wi2] (The current consumptions - mW) — **transmit (TX)**	54	24.7	219
Power [Wi2] (The current consumptions - mW) — **Receive (RX)**	47	27	216
Cost	Low	Low	High

Appendix B

BluSen Source Codes for Windows;

Application for data acquisition system

Here are only some parts of the source code that may help readers and developers to improve the applicational implementation of this work in the future. Application part of this work is developed in both Android and Windows operating systems. Figure 48, 49, 50 and 51 shows different features of the application which allow

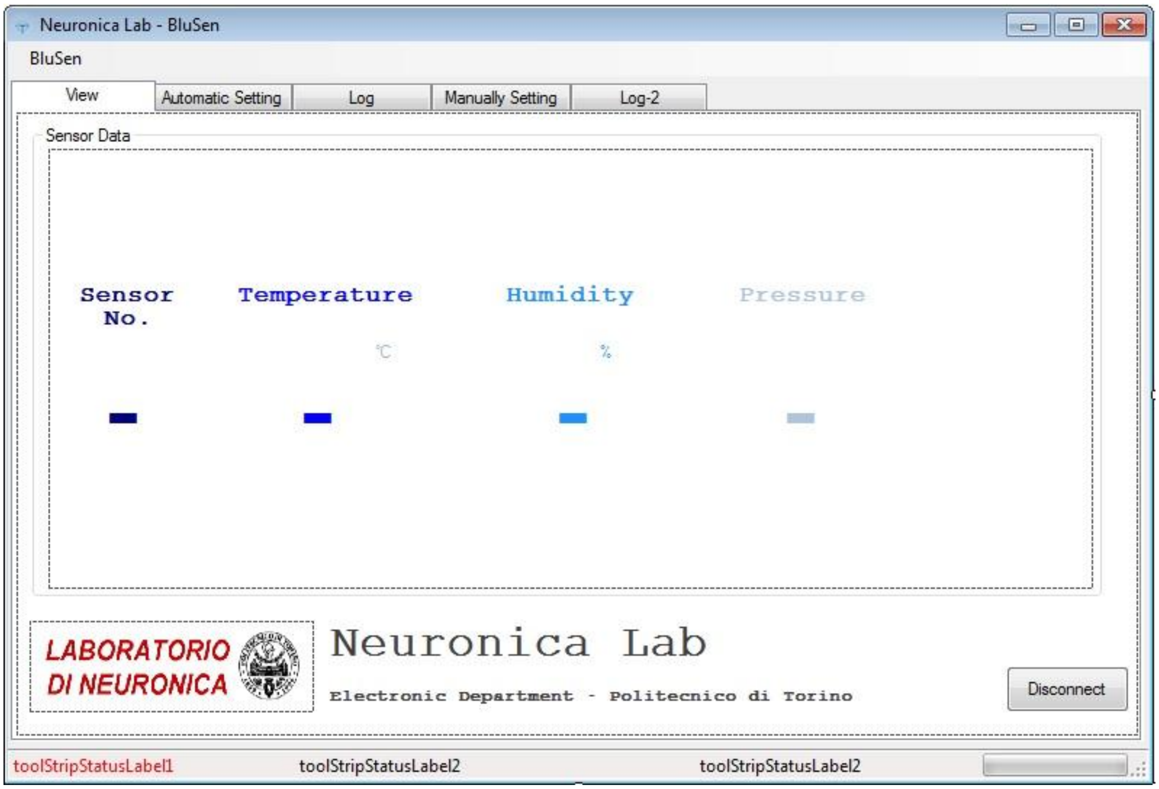

Figure 47. The interface is dynamic. It makes the application compatible with number of connected sensors to present the sensed information dynamically.

handling communication to a number of available or pre-configured sensors. Windows codes that are written by C# .net are listed as following:

Eight parts of the source codes are presented here for readers who want to develop similar applications. Some of the important components and classes (`WidcommAPI` and `WidcommStack`) are needed to be added to the project.

Source Code 1

```
using System.IO.Ports;
```

```
using System.IO;
using InTheHand.Net;
using InTheHand.Net.Sockets;
using InTheHand.Net.Bluetooth;
using System.Net.Sockets;
```

Two public variables are needed to be used for handling Bluetooth communication.

Source Code 2

```
public BluetoothClient bc2;
public  StreamReader rdr ;
```

A public class for handling device is needed as well: `public class Device` with the following subclasses:

Source Code 3

```
    public string DeviceName { get; set; }

    public bool Authenticated { get; set; }

    public bool Connected { get; set; }

    public ushort Nap { get; set; }

    public uint Sap { get; set; }

    public DateTime LastSeen { get; set; }

    public DateTime LastUsed { get; set; }

    public bool Remembered { get; set; }
```

Source Code 4

```
   public Device(BluetoothDeviceInfo device_info) {

       this.Authenticated = device_info.Authenticated;

       this.Connected = device_info.Connected;

        this.DeviceName = device_info.DeviceName;

        this.LastSeen = device_info.LastSeen;

       this.LastUsed = device_info.LastUsed;
```

```
        this.Nap = device_info.DeviceAddress.Nap;

       this.Sap = device_info.DeviceAddress.Sap;

      this.Remembered = device_info.Remembered;
     }
```

A function can be used as Bluetooth counter, allowing us to that let us to work with each set of Bluetooth information in a loop with the following variables:

Source Code 5

```
List<Device> devices = new List<Device>();
BluetoothClient bc = new BluetoothClient();
BluetoothDeviceInfo [] array = bc.DiscoverDevices(100, false, false, false,
true);
```

By checking the Bluetooth module which will be used for communication should not be NULL,

Source Code 6

```
if (bc2 != null)
            DisConnect();
```

Then we can have the following codes for making connection to the Bluetooth device:

Source Code 7

```
            serviceClass = BluetoothService.SerialPort;
            bc2 = new BluetoothClient();

            BluetoothAddress addr = BluetoothAddress.Parse(addt);
            BluetoothDeviceInfo device = new BluetoothDeviceInfo(addr);

            try
            {
              BluetoothEndPoint ep = new BluetoothEndPoint(addr,
serviceClass, 1);
              bc2.Connect(ep);

            }
```

If the Bluetooth is connected we can further make a line to read stream of data:

Source Code 8

```
                 if (bc2.Connected)
                 {
                     try
                     {
                         var peer = bc2.GetStream();
                         var rdr = new StreamReader(peer);
                         string line2;

                         try
                         {
                             line2 = rdr.ReadLine();
// Here we can work with read data
                          }

                      }
                  {
```

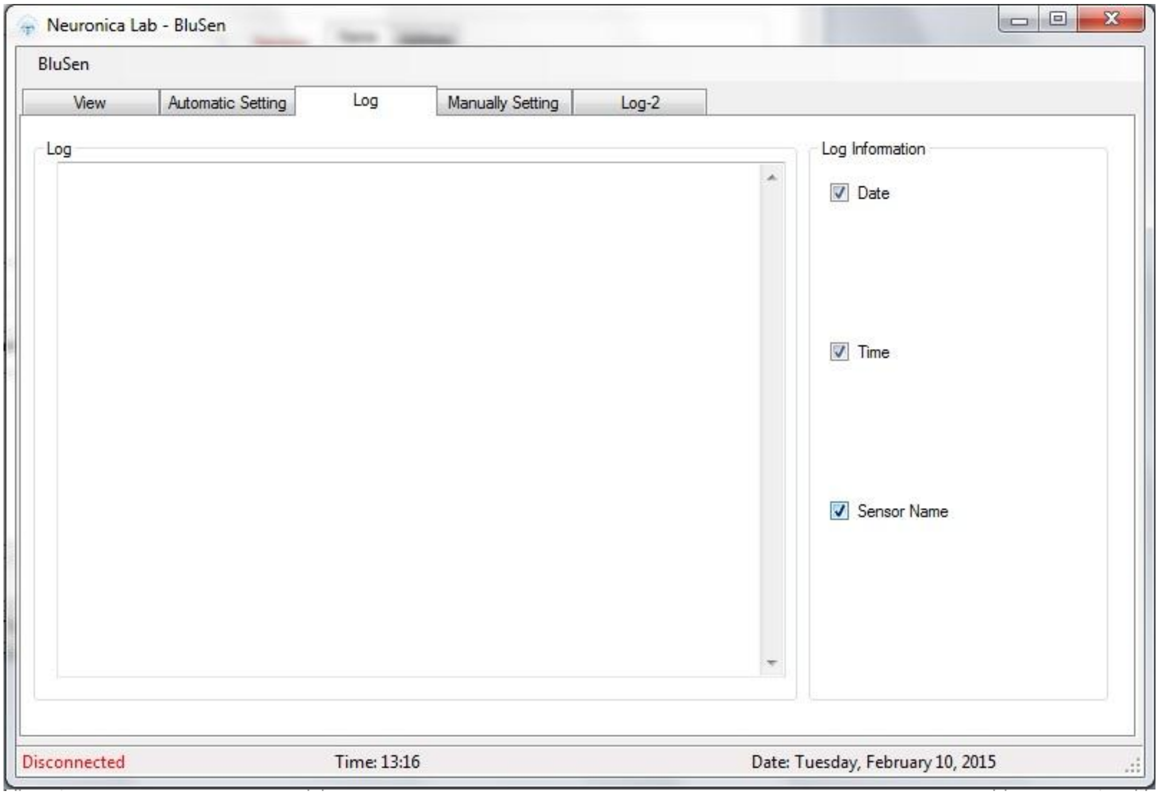

Figure 49. User can record logged data with more information such as data, time and sensor name

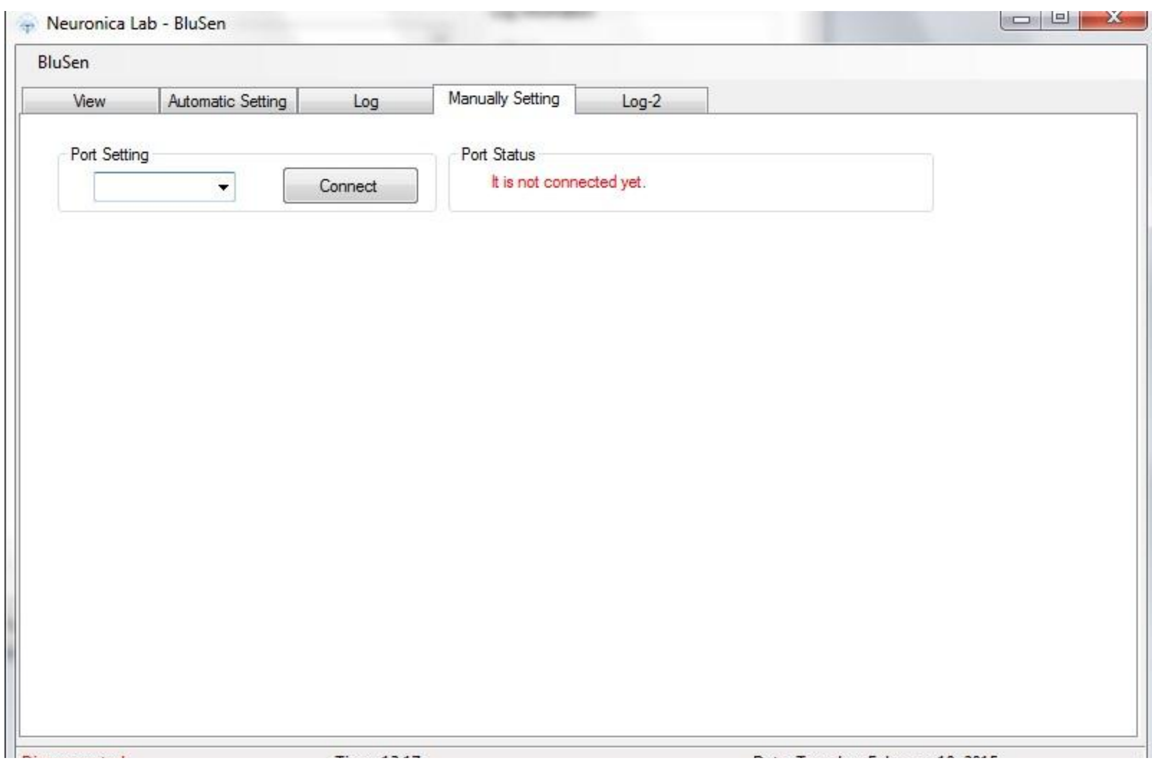

Figure 50. Port setting can be set up manually as well.

Appendix C

Prediction algorithm

A set of MLPs is considered to perform the prediction. Measurements from each sensor were interpolated at a constant time interval r. The value of the interpolated measurement in the most recent time and the delayed value of one sample interval are considered for each sensor as the inputs of the MLP. The data are divided into training (60% of data), validation (20% of data), and test sets (20% of data). A single hidden layer is used, which is sufficient to approximate any non-linear function (universal approximation property, [70]). The neurons in the hidden layer applies a sigmoidal activation function

$$f(x) = \left(1 + \exp(-x)\right)^{-1} \tag{13}$$

The number of neurons in the hidden layer is chosen in the range from 20 to 50. The output neurons have linear activation function. Each output neuron is used to predict the measurement of a specific sensor. MLPs are trained by modifying iteratively the weights and the bias in order to reduce the error in predicting the training data, applying the quasi-Newton algorithm [17] for a number of iterations in the range of 50 to 400 times. An optimal network is selected choosing the topology (i.e., the number of hidden neurons) and the parameters (synaptic weights and bias, after a specific number of iterations of the optimization algorithm) with best generalization performances (i.e., with minimum error in the validation set). The proper MLP provides a function estimating the relation between the available information (past and present measurements) and the subsequent measurements

$$\vec{x}(t+\tau) = \vec{F}(\vec{x}(t), \vec{x}(t-\tau)) \tag{14}$$

where $\vec{x}(t)$ indicates the set of sensor measurements (acquired or predicted) and $\vec{F}(\cdot)$ is the vector function predicting the future values from each sensor.