Doctoral Dissertation
Doctoral Program in Control and Computer Engineering (33.rd cycle)

Sensor-based ICT Systems for Smart Societies

Edoardo Giusto

* * * * * *

Supervisor
Prof. Maurizio Rebaudengo

Doctoral Examination Committee:
Prof. Antonio Liotta, Referee, Free University of Bozen-Bolzano (IT)
Prof. Julio Pérez Acle, Referee, Universidad de la República de Uruguay (UY)
Prof. Enrico Natalizio, Technology Innovation Institute (UAE)
Prof. Giovanni Pau, Università di Bologna (IT)
Prof. Luca Sterpone, Politecnico di Torino (IT)

Politecnico di Torino
September 2021

I hereby declare that, the contents and organization of this dissertation constitute my own original work and does not compromise in any way the rights of third parties, including those relating to the security of personal data.

. . . .

Edoardo Giusto

Turin, September 2021

Summary

We are living today in the Internet of Things (IoT) era, with a multitude of connected devices around us. These devices can be sensors, sampling every possible aspect of our life, actuators, taking an action depending on our input or automatic decision, or interfaces, such as computers and smartphones, presenting valuable information in a way that enriches or simplifies our everyday tasks.

The field of IoT has been growing steadily since its conception, 20 years ago, thanks to the technology improvements which made the devices used in this domain cheaper and cheaper, enlarging the spectrum of possible users.

The key of the IoT framework are the *data*. Data are the new gold or new oil of the Smart Societies era. Data can be used to better understand the current situation of a certain environment or device under test, so that it is possible to take the corresponding action in order to perform better at some particular task or improve life quality conditions for humans.

To build a scientifically sound and efficient IoT device is not a simple task. It may seem easy to retrieve, collect, store and analyze this kind of data. In reality, these are tasks that require much attention from various perspectives.

In the literature it is possible to find a plethora of articles trying to do so, but failing in one or more (or all) of these aspects. First, these data have to be gathered by a sensor with a certain accuracy and has an output consistent with the environmental conditions (i.e., it must not produce spurious data). Moreover, depending on the desired quantity to sample, one needs also to determine the correct sampling frequency, in order not to violate the Sampling Theorem. Second, according to the constraints of the environment and the requirements in terms of transmission timing, the data transfer protocol has to be carefully chosen. Third, the data have to be stored in a certain structure depending on usage requirements (CSV files, databases, Distributed Ledger Technology, etc.). This thesis reports a detailed analysis of the most critical issues in the design and implementation of an IoT device, taking into account: the correctness of the data gathered; the optimization of the power consumption in relation to the signal to be sampled; the analysis of different wireless technologies that can be used to transmit the gathered data; the conjuncture of IoT and Blockchain paradigms for the sake of data authenticity and safe storage of sensible data. As a side topic, this thesis

also discusses another breakthrough in the Computer Science domain: Quantum Computing. This new computing paradigm is changing the landscape of a variety of human endeavors, from computing, to physics, to chemistry. The basic concepts are described, along with the possibilities this mindset entails and the consequences of such advanced computing power, which could impact directly the IoT/Blockchain implementation discussed.

Summarizing, the main contributions of this thesis are the following:

- *Ensuring scientifically-sound data exploiting low-cost sensors*
  The topic is addressed by focusing on the case study of air pollution monitoring. It is reported the design and development of a Particulate Matter monitoring station via a rigorous evaluation of sensors and calibration procedures. The resulting platform serves as proof-of-concept for the creation of densely deployed networks of low-cost sensors to work in synergy with reference-grade devices in order to create finer-grade air pollution maps.

- *Evaluation of wireless protocols for the task of indoor transmission of IoT data*
  The topic is addressed by considering the task of indoor thermal monitoring. In this context, two widespread wireless technologies, RFID and Bluetooth, have been compared both theoretically and via a series of meticulous in-field tests to assess their operating range, transmission efficiency, scalability, and resilience to interference.

- *Integration of IoT and Blockchain domains*
  This thesis reports the integration of the Blockchain Distributed Ledger Technology in the context of IoT. This implementation serves as the missing block with which it is possible to make IoT devices communicate with an open blockchain. This could enable true transparency in several manufacturing processes.

- *Analysis of new possibilities and threats posed by Quantum Computing*
  As a side topic, this thesis addresses the emerging field of Quantum Computing. The possible applications of such a new paradigm are reported, along with new threats posed to encryption schemes used today.

Acknowledgements

I would like to thank the people that helped me in all possible ways in this long journey.

First, thanks to my supervisor, Maurizio, who gave me the opportunity to pursue such degree and guided me through it, with precious suggestions valid for many situations.

Thanks also to the rest of my research group, Bartolomeo, Filippo, Renato, Stefano, Antonio, Pietro, Gustavo for all aid in channeling ideas into implementation, up to the paper form.

Thanks to the whole CAD Group for being a strange heap of smart people.

Thanks to Mauro and Serena, two brilliant robot-making expats.

Thanks to Mohamad, never-ending source of support in practical and life problems. I hope you get all the certificates you might need.

Thanks to Donatella and Daniele for the chance you gave me by bringing me into this world.

Thanks to Martina, my little Sista. I feel you wherever you are.

Special thanks to Lu, who pushes me to be the best version of myself I can be. The version I want to be.

Che figata UNIX!

Bartolomeo Montrucchio.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

We are living in the era of the Internet of Things, or Internet of Everything [1], surrounded by devices connected to the internet and performing a type of task. These devices help us in everyday activities and can produce precious information, which can be exploited for decision-making or lifestyle adaptations. The definition of IoT goes back to a presentation Kevin Ashton gave in 1999 at a Procter and Gamble event in the context of supply chain management [2]. As of today, the field of application has expanded to healthcare, utilities, transport, home automation, battlefields, environmental monitoring, and so on [3].

The IoT has made the domains of Computer Science and Engineering evolve constantly since its inception, becoming every day less *desktop* and more pervasive, *ubiquitous*. Computing itself is nowadays ubiquitous.

Embedded, or Ubiquitous computing is pervasive, yet not easy to see. One may not think it has a direct impact on our lives since we cannot fully appreciate its presence, but indeed it has an enormous impact. The aim of embedded computing is to make life easier, safer, and more secure for all mankind, regardless of whether the individuals are aware of it or not. The result is that these devices and frameworks are going to be much more than simply "autonomous" systems. They are going to yield valuable and(or) necessary information to us humans, like the possibility to gather data in real-time, crunch it, and act accordingly.

Having internet-connected devices enables the possibility to delegate certain activities - onerous, dangerous, or repetitive - to a dedicated device. It is a common belief that in the future the houses we live in will be fully autonomous and requiring little to no housekeeping at all.

The rising ramp for the spread of IoT platforms was made possible by three different factors:

- The diffusion of open wireless protocols - Bluetooth, Bluetooth Low Energy (BLE), Radio Frequency IDentification (RFID), Wi-Fi.
- The shift from IPv4 to IPv6.

- The constant cost, size, and energy consumption reduction for all the components.

As introduced before, the use case application spectrum for IoT is very large. Tons of devices are being used every day to record a plethora of parameters of the environment: temperature, relative humidity, sun exposure, atmospheric pressure, air quality, etc. Logged data can be leveraged to create models to represent physical phenomena and predict future conditions, from basic weather forecasts to more catastrophic events. This is the reason why there are sensors placed everywhere in the world, monitoring our seas and rivers, the movement of the soil, the air we breathe.

1.1 Wireless Sensor Networks

The IoT in reality consists of a set of devices brought together to create Wireless Sensor Networks (WSNs).

Sensing/actuating devices in WSNs are called *nodes*. A single node of the network is usually built in a simple architecture, like the one shown in Fig. 1.1. It is composed of a core board, a battery, a sensing device, a memory, and a radio transceiver.

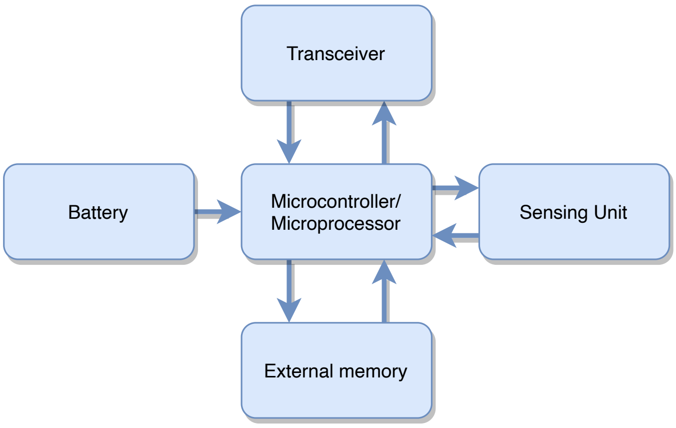


Figure 1.1: Sensing Node Architecture.

Since these kinds of devices are relatively small, the most critical resources are energy and memory. These constraints limit the frequency of the core board, which operates at a much lower frequency with respect to the central processing units of modern personal computers and workstations.

There are several kinds of network architectures, the most common ones are the *star* and *mesh* network topologies, shown in Figg. 1.2a, 1.2b.

The choice of the network topology depends on the constraints of the final system, there is no a priori best possible choice. The star network is simpler and

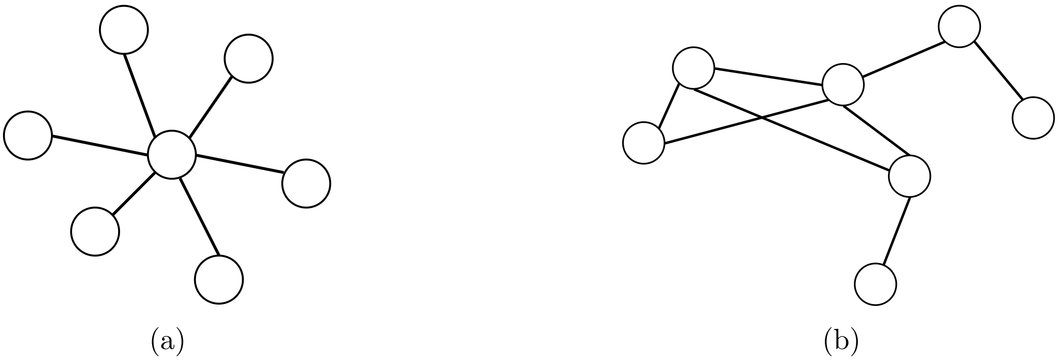


Figure 1.2: Network topologies: star network (a), mesh network (b).

faster since all sensing nodes are connected to the *sink* node. It is possible to synchronize the transmission times for the various nodes in such a way that in the network each node has a transmission time slot allocated. This could reduce the noise and thus the need for retransmission, resulting in lower power for the sensing node. This architecture, however, introduces a single point of failure for this kind of topology, since the other nodes could not connect to the internet without that central missing node. On the other hand, a mesh network is theoretically more resilient, since there could be more than one sink nodes so that if one of them is out of order for a certain reason, the packets can be redirected to the other gateway node. This kind of architecture is characterized by a multi-hop routing since a certain packet has to perform several jumps - hops - from a node to the following, in its path to reach the gateway. This increased resilience is paid for by the increased time for a package to reach the destination and the increased power for all nodes for transmission of it (or re-transmission in case of noise).

1.2 Questions

In the context of IoT, this thesis addresses various aspects which are sometimes overlooked when creating an IoT sensing device or a whole Wireless Sensor Network. We want to answer a series of questions:

- Are we sampling a certain signal at the correct frequency?
- Does the sensor we use produce scientifically valid data?
- Is it possible to reduce the energy consumption and the logging of data in certain circumstances? What are the consequences of this?
- What is the best wireless protocol to transmit sensed data?
- Are these data authentic?

- Now that we were able to sample and transmit these precious data, how do we store them in a way that they cannot be tampered with/altered?

In parallel with the work trying to answer these questions, a new topic of interest emerged on the panorama: Quantum Computing. This new computing paradigm has the potential to shake much of the knowledge we have built up to now, the consequences of which are still to be understood.

In the following subsections, we are going to give an overlook of all chapters included in this thesis.

1.2.1 Data Acquisition

Talking about data acquisition, it is not possible to start somewhere else than the *Sampling Theorem*, formulated throughout the history by Harry Nyquist, Claude Shannon, E.T. Whittaker, Vladimir Kotelnikov, which reads - in Shannon's formulation [4]:

Theorem 1. *If a function f(t) contains no frequencies higher than* ***B*** *Hz, it is completely determined by giving its ordinates at a series of points spaced* ***1/(2B)*** *seconds apart.*

Reversing this sentence, if we need to sample a certain signal, we have to do so at a frequency that is at least double the frequency of the signal itself. This may not seem so significant in the IoT domain, but it actually depends on how frequently the quantity you want to measure changes rapidly in time. There are certain applications - and we will be seeing them throughout the whole dissertation - in which sampling at a frequency that is lower than the defined threshold may yield severe consequences for the monitored product or system.

As previously mentioned, one of the forces driving the spread of the IoT is the low cost of these devices. Today it is possible to buy a certain sensor for a handful of dollars/euros. However, what kind of precision characterizes these sensors? Shall we take for granted the specifications of the datasheet? How do these devices age? Before deploying on the field a sensing device embedding a certain sensor, we have to actually evaluate up to what degree of confidence we can use that particular sensor for that specific task.

The case study chosen for this topic is **air pollution monitoring**.

Air pollution is a critical phenomenon impacting many cities all around the world. It has seen growing attention in recent years thanks to the awareness of individual people and to the effort of influential figures.

Causes of air pollution are diverse: fossil fuel burning for vehicular transportation, coal energy plants, factory production, waste incineration, man-made controlled burn for cattle farming, and so on, and so forth.

One of the most harmful resulting materials of these activities is *Particulate Matter (PM)*, which is fine dust that can be split into two main categories, depending on the diameter:

- PM_{10} made of particles smaller than $10\mu m$.
- $PM_{2.5}$ made of particles smaller than $2.5\mu m$.

The smaller this dust is, the deeper it can get into the lungs, causing a series of respiratory problems, and also cancer [5]. Recent studies have also linked PM pollution to an increase of the severity of Covid-19 health-related issues, among which, death [6].

The monitoring of PM concentration in the air is historically performed by environmental agencies exploiting high-precision techniques based on β-attenuation monitoring and gravimetric detection, as specified by the European Union in the 2008/50/CE Directive[7].

However, these high-precision sensors are also very expensive to build and maintain, with the result of a low spatial coverage due to the sparsity of the network.

A different approach is to use low-cost commercial devices to build monitoring stations with the aim of creating much denser air pollution maps. Such high-granularity maps could give a more detailed report of the ongoing situation throughout the entire territory under test and near sensitive places, such schools and hospitals. Having such maps could also be exploited by municipalities to address particular locations and possibly plan urban development according to air pollution levels.

This new opportunity caught the interest of academia and several projects of citizen science.

In certain cases, the mantra of these implementations is *low cost at all costs.* Unfortunately, this can be greatly misleading, since the produced data are very far from the actual PM concentration in the air, resulting in a wrong report of the current situation. It is of central importance to carefully evaluate the performance of these cheap devices and try to compensate for their technology imperfections by taking into account the boundary conditions in which they operate.

In parallel to the calibration task, we also wanted to address the possible concurrent reduction of the energy consumed, the data gathered and transmitted, the aging of sensors.

The deployment on an IoT system often entails the unmanned operation of the device itself, i.e. it should be able to be as energy-efficient as possible in order to have the least human intervention for maintenance or none at all.

The reduction of power consumption comes at the cost of a reduction of the time in which the sensors are active. This directly translates to a reduction of the volume of the gathered data. Please keep in mind that this is possible only in particular situations, where the frequency of the signal to be sampled is very low. In the case of PM levels, this subsampling could be carried out in environments with low time variability, such as those away from vehicular traffic and other polluting sources. If such conditions are in place, a reduction of the operating time of the sensor is possible, with several advantages: the reduction of logging of redundant information; the consequent reduction in the memory needed to store that information or the need to transmit it; last consequence is the reduction of the strain on the mobile parts of the sensors, with the outcome of an increased lifetime and reduced maintenance needed. A great reduction in consumption could also result in the possibility for the device to autonomously harvest energy from the sun by means of solar panels and store it in a battery for applications in remote areas or where a power socket is too far away.

In chapter 2 we thoroughly analyze the applicability of light-scattering sensors to the detection of the level of PM_{10} and $PM_{2.5}$ in the air. These sensors are put to work in an open-air environment in the immediate surrounding of a reference station of the Regional Agency of Environmental Protection (ARPA). We carried out a long data acquisition campaign, 5 months during autumn and winter, to follow the decline of temperature and the rise of pollution due to home heating. We did not only log the values of Particulate Matter at high frequency (1 sample per second), but also the temperature, relative humidity, and pressure of the surrounding environment. These data enabled us to perform different calibrations on the sensors and to study the effect of the boundary conditions on these low-cost devices. It turns out that relative humidity is the factor most affecting the behavior of these sensors, and a calibration using a Multivariate Linear Regression model is able to compensate for the error in measurement with respect to the ARPA reference station. The developed mobile air pollution monitoring stations, equipped with calibrated sensors, have also been exploited to perform several tests to assess the usefulness of such devices in real-time use case applications. The resulting system proved to be very efficient in detecting the concentration of PM in the air not only in a steady situation, but also in the presence of fast transient phenomena, which could not have been detected by high-precision sensors, due to intrinsic construction limitations. Such a system could therefore be leveraged in the implementation of a wide scale PM monitoring project carried out by a city municipality.

On the other hand, the analysis on the power consumption optimization is carried out by means of studying the impact of applying a duty cycle on the operation of the PM sensors. We want to assess if, under certain conditions, it is possible to reduce the operating time of the sensors and how great an impact this activity has on the quality of the gathered data. Our analysis explores several possible duty-cycle sets, taking into account the information in the datasheet of the sensors in the eyes of a critic. Our results show that it is possible to reduce the duty-cycle in such a way that the energy required for the operation is more than halved, without a significant impact on the quality of the gathered data. This applies of course to deployments in which a fast transient of the PM in the air is absent, given the distance from the traffic or other polluting agents.

1.2.2 Data Transmission

Now that we gathered the required data fulfilling certain specifications, it is time to transmit these data from the sensing site to the elaboration site.

However, how does one decide which is the best way of transmitting a piece of certain information? Well, it depends.

There are several ways to transmit data: copper wires, optical fiber cables, and wireless channels.

In the IoT domain, wireless communications are preferred, since this solution avoids the hassle of cable placement, which can be difficult or even impossible in certain environments.

There are however several different modes of realizing wireless transmission, the most widespread of which is radiofrequency.

In the radio frequency domain, there are many wireless communication protocols, operating at different frequencies, over different ranges, transmitting at different bit rates and with different latencies. Some protocols are simpler, others are more complex and require more electronic circuits and different antennas to be operational.

Therefore there is no one-size-fits-all wireless protocol for wireless data transmission.

We, users, are used to thinking of wireless communication as the only three protocols we use every day, so cellular connection (3G, 4G LTE), the WiFi network at home or in the office, Bluetooth to connect to headphones or to play music from a speaker. Indeed, these connections are a significant part of the whole data transmission we actually see in the first person. However, there are yet a lot of transmissions that we do not see since they are not carried out by our phone or laptop.

In Chapter 3 we evaluate the applicability of two of the most widespread low power wireless protocols - RFID and Bluetooth - to the task of indoor comfort monitoring. The working environment is thus identified as an office building, where people work and have to be productive, and to do so, they have to feel good in that environment. It should be not too hot or too cold, not too dry or too wet. Of course, granularity plays a role in this, so how densely do we place the sensors in the environment. However, in certain cases too high granularity could mean redundancy, that can be avoided in non-mission-critical situations such as the one considered here. A dense-deployed network of temperature and relative humidity sensors can effectively and inexpensively evaluate different ambiances inside a building and give feedback to the Heating, Ventilation, Air Conditioning (HVAC) system, which could also take into account user preferences. RFID and Bluetooth, while being wireless standards used in commercial electronics, result to be well suited to be employed in IoT systems and HVAC systems in this particular case.

1.2.3 Data Authenticity and Beyond. "Hello, Quantum World!"

Now that we have shown how it is possible to sample and transmit the data, there are some other questions left: are we sure that the transmitted data are authentic? How do we store these data in such a way that they cannot be tampered with?

In a world where, as we have shown, it is very easy to produce a numeric value representing a quantity that can be sampled from the environment with little precision, we could be flooded by meaningless, spurious data. It is crucial to actually determine if a certain sensor has a certain level of precision since it would be wrong to implement some sort of decision-making mechanism based on data that are not representative of the physical world.

It is a crucial requirement that these data stay secure, since they may be important for decision-making tasks - theoretically at building-, city-, state-level - or log health parameters of individuals. In both these cases and in other situations it is crucial for these data to be kept as they are and not modified by any malicious entity.

A possible way to ensure that these data are authentic and tamper-proof is to use Distributed Ledger Technology (DLT). The interest for humanity for DLTs sparked right after the publication of the original Bitcoin paper with the obscure signature of Satoshi Nakamoto [8] in 2008. After the advent of Bitcoin, several different implementations of Blockchain have been proposed, targeting not only crypto-currency and finance applications but also automation and certification of business processes [9]. Several studies nowadays are exploring the possibility of applying the BC technology for product certification processes, especially in the food industry, and most of them believe that this technology would represent a powerful partner for solving problems related to consumer's trust in food products and to implement a traceability system able to find possible sources of contamination in-time, thus reducing the risks for citizens' health [10].

Chapter section 4.1 reports an in-depth analysis of the problems related to the implementation of DLTs in the IoT framework, with the presentation of a Blockchain-enabled IoT gateway that can be used to make the sensors sign the transactions directly and send them to a remote node for the transaction to be inserted in the chain.

As previously introduced, a side topic in this Doctoral Degree was the exploration of the fascinating new topic of *Quantum Computing*. This new computing paradigm presents new possibilities and threats to the world as we know it.

Quantum Computing is expected to be a breakthrough not only in the computing field per se, but also in all the other fields which intensively use computing power to carry out calculations and simulations. This is due to the radically different way in which the information is represented inside a quantum computer. The base of computation, the *qubit* (instead of a simple bit), presents two crucial characteristics: *superposition* and *entanglement*. In some way, superposition means that the qubit is both 0 and 1 at the same time (but actually it is not like that). This results in the fact that, if you have an array of n qubits, you are actually handling 2^n states altogether, hence the so-called *quantum speedup*. Entanglement instead means that two or more qubits are linked together in a peculiar quantum-mechanical way. In fact, they have formed a bond, a special way to influence each other, even if they are placed at a great distance. Albert Einstein referred to this phenomenon as *"spooky action at a distance"*.

Thanks to this *speedup*, quantum computers are well suited for optimization problems, such as the *Travelling Salesperson Problem* or the *Max-Cut* problem. The main goal of optimization problems is finding the best solution to a problem according to some criteria. The complexity and amount of data involved in solving them require the definition of more efficient solution approaches. The usual quantum computation approach to optimization problems is to formulate them as minimization problems, where the optimal solution corresponds to the minimum of a cost or an error function.
As of today, two approaches are possible:

- **quantum gate array**, where all operations are implemented as sequences of quantum gates, each one associated to a particular unitary evolution of the quantum system;

- **quantum annealing**, a procedure for finding the global minimum of a given function over a given set of candidate solutions, associated to the states of a quantum system, mainly used for combinatorial optimization problems, where the search space is discrete with many local minima.

It is our intention to explore these possibilities regarding the possible applications of quantum computing to the data analysis task in the activities we are dwelling in, mainly the air pollution monitoring one.

The computational formalism for quantum computing was mainly defined in the second half of the 1990s, when the quantum circuit model was formalized [11], [12] and Lov Grover and David Shor proposed their quantum algorithms for searching elements in unsorted databases [13] and prime factoring [14] respectively, which were proved less computationally expensive than their classical counterparts. These proofs of quantum advantages with respect to classical computers forced the necessity of fabricating real quantum hardware.

Unfortunately, this new computing power poses a great threat to security as we know it. Shor's prime factoring algorithm could, if executed on a QC with a sufficient number of qubits, break the RSA cryptosystem, based on the Public/Private-key pair. This possibility created an immediate call to arms by all cryptographers around the world to design a quantum-resistant cryptosystem. It is a common opinion that such a cryptosystem will find an application in securing Wireless Sensor Network communications someday [15].

The prospects of the adoption of QC are discussed more thoroughly in Chapter section 4.2, with analysis in particular related to the tampering of Distributed Ledger Technologies.

QC has actually already found *unusual* applications, such as in music composition. During IBM Qiskit Camp Europe 2019, the author of this thesis, with other colleagues of Politecnico and other institutions, developed *Quantum synth: a quantum-computing-based synthesizer*, which is an interface for controlling sound synthesis parameters encoded on the basis states of a quantum computer [16]. This sound synthesis is obtained from the potential measured outcomes of a quantum circuit. The interface has been designed to be used by music performers and composers in their creative process, and as a resource to both learn Quantum Computing and analyze the intrinsic noise of real quantum hardware. The project found great success during the event, and was awarded "Community Choice Award" Prize.

1.3 Thesis Contributions

Summarizing, this thesis presents the following contributions:

- Ensuring scientifically sound air pollution data via thorough quality assessment of low-cost sensors.
- Evaluation of wireless protocols for the task of indoor transmission of IoT data.
- Integration of IoT and Distributed Ledger Technology frameworks.
- Analysis of the possible breakthrough in the field of computing thanks to Quantum Computing and evaluation of threats to classical encryption schemes and data vaults.

Chapter 2

Acquiring data

Air quality, especially particulate matter, has recently attracted a lot of attention from government, industry, and academia, motivating the use of denser air quality monitoring networks based on low-cost sensing strategies. However, low-cost sensors are frequently sensitive to aging, environmental conditions, and pollutant cross-sensitivities. These issues have been only partially addressed, limiting their usage.

In this chapter, we report the development of a low-cost particulate matter monitoring system based on special-purpose acquisition boards, deployed for monitoring air quality on both stationary and mobile sensor platforms. We explore the influence of all model variables, the quality of different calibration strategies, the accuracy across different concentration ranges, and the usefulness of redundant sensors placed in each station. The collected sensor data amounts to about 50GB of data gathered in six months during the winter season. Tests of statically immovable stations include an analysis of accuracy and sensors' reliability made by comparing our results with more accurate and expensive standard β-radiation sensors. Tests on mobile stations have been designed to analyze the reactivity of our system to unexpected and abrupt events. These experiments embrace traffic analysis, pollution investigation using different means of transport, and pollution analysis during peculiar events.

With respect to other approaches, our methodology has been proved to be extremely easy to calibrate, to offer a very high sample rate (one sample per second), and to be based on open-source software architecture. The gathered database and developed software are available as open-source in [17].

In addition to the calibration of sensors, this chapter addresses challenges related to IoT devices in the domain of air pollution monitoring: reduction of consumed energy; reduction of data logged and transmitted; aging of sensors. This study aims to understand if it is possible to reduce the duty cycle of an air pollution monitoring sensor and still get meaningful data on the general behavior. This has advantages from all perspectives: it enables less logging of redundant information;

it reduces the strain of the sensor to extend its lifespan and to reduce maintenance costs; it reduces the energy consumed by the sensor, essential in battery-powered devices.

Some of the work described in this chapter has been previously published in [18],[19] and some are currently being reviewed at an international conference on the topic.

2.1 On the calibration of sensors

This chapter section reports the design and development of an air pollution monitoring station, its calibration against reference values, and the performance analysis in an extensive series of tests. The analysis shows the high applicability of such a platform for the creation of fine-grained pollution maps, for the benefit of municipalities and of citizens.

2.1.1 Introduction

Conventional approaches to the task of air quality monitoring are based on very sparse networks of static reference-grade detectors, like the one in Fig. 2.1.

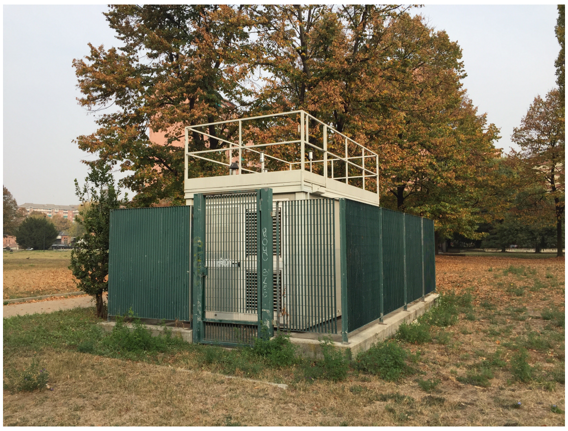
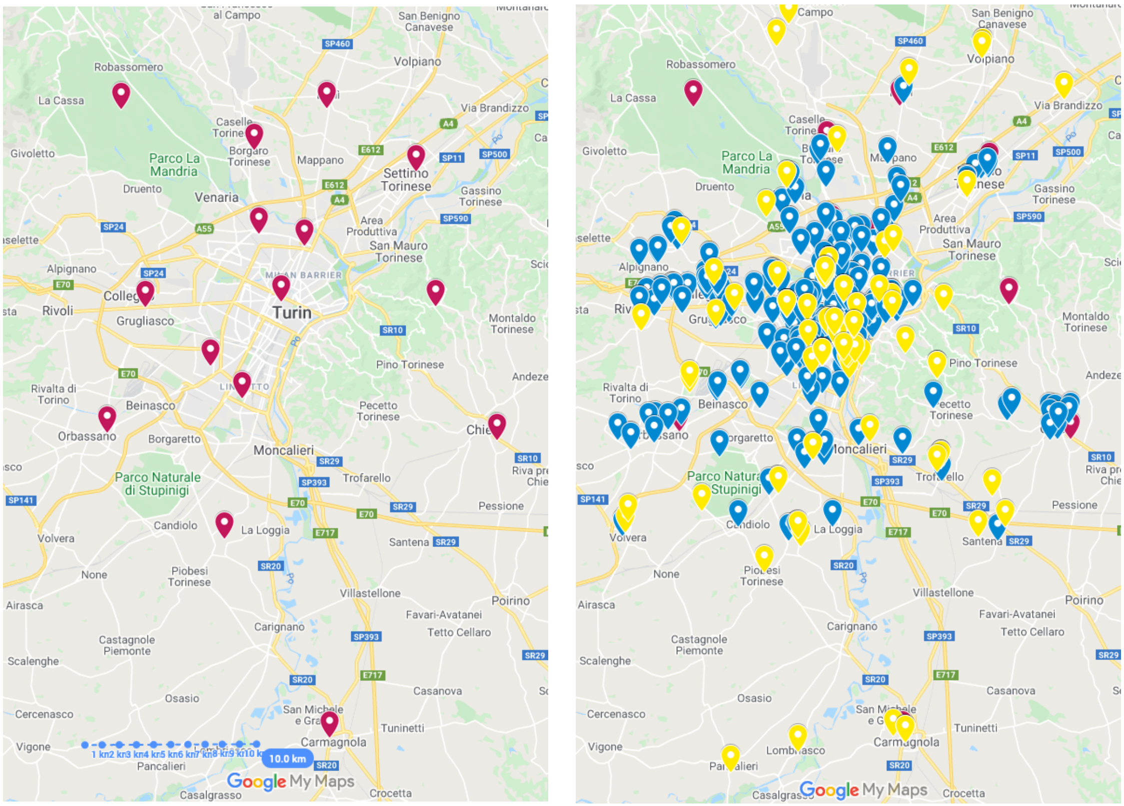


Figure 2.1: The ARPA Rubino monitoring station.

The spatial coverage of these networks has been limited by the high cost of instrumentation. In Fig. 2.2 a Map of the Metropolitan Area of Turin is displayed. On the left (Fig. 2.2a) there are the locations of ARPA monitoring stations, while on the right (Fig. 2.2b) it is possible to find the locations of sensitive places, such as schools, hospitals, and care homes.

From the one side, these micro-balance Particulate Matter (PM) monitoring stations are very accurate, but they are large and cost on the order of $5K - 30K$€ [20].



(a) (b)

Figure 2.2: The Metropolitan Area of Turin. Map of only ARPA monitoring stations (red pins) (a), Map of the location of schools (blue pins) and hospitals/care homes (yellow pins) (b).

On the other one, portable light-scattering based PM detectors have varying accuracy and costs between $300 - 2K$ € [21]. Moreover, air pollutant concentrations often exhibit significant spatial variability depending on local sources and features of the built environment, which may not be well captured by the existing sparse monitoring networks. As a consequence, there has recently been a significant increase in developing and applying low-cost sensor-based technology, which could enable much denser air quality networks at a comparable cost to the existing ones (see, for example, Giusto et al. [18]). These sensing nodes can be adopted not only in city-wide applications, but they can also be used in more strategic, location-aware deployments [22] or even to monitor the conservation state of historic buildings [23], [24]. Furthermore, mobile monitoring enables participatory sensing approaches, which in turn are well-suited to address many of the above aspects, as they intrinsically involve empowering citizens by providing individuals with low-cost measurement devices. Unfortunately, mobile sensing devices have also several drawbacks. Among these disadvantages, we recall their necessity to be battery-supplied and their limited processing capability. These limitations present challenges that can be overcome only adopting persistent engineering solutions [25], [26].

Following the previous considerations, in this chapter section we present the design, assembling, and evaluation of a low-cost, open-source air quality system which is based on special-purpose acquisition boards, deployed on stationary and mobile platforms, devised for participatory sensing strategies. In order to do so, we follow article 18.5 of Italian Decree 155/2010 on the dissemination of air quality data, which absorbs EU directive 2008/50/CE. Therefore, we declare, in the acknowledgment section, that our data cannot be considered as official.

First of all, we describe the design of our sensor platform. Each station contains 4 particulate matter (PM_{10} and $PM_{2.5}$) sensors plus single sensors for temperature, humidity, and pressure. Each platform is powered by a Raspberry Pi Zero Wireless board. The cost for the entire board, considering sensors, electronic components and enclosure is around 200€, while the reference grade devices have a price in the range $5K - 30K$€[20]. This configuration enables our application to work with a quite high sampling rate (i.e., one sample per second) which is able to accurately follow sudden atmospheric phenomena. We deployed about 100 sensors, both on stationary and mobile platforms, for a period of 5 months, from October 2018 to February 2019. We focused upon the city center of Turin (located in the north-west region of Italy) inside and outside the limited traffic area to reflect the variation in traffic density, driving speed, and street configuration. For the sake of completeness, we also included in our experiments a recreational area (i.e., a park) with very low traffic density at its border. The monitoring hours cover the entire day, with specific experiments running from 11.00 a.m. to 05.00 p.m. The collected database amounts to about 50GB of data, corresponding to about $700 \cdot 10^6$ data tuples.

Secondly, we describe the entire software architecture. Our application manipulates a heterogeneous set of input data coming from our sensor stations and the reference platforms. The public air quality station uses expensive instruments delivering very accurate measures. Unfortunately, these samples are available only from very sparse locations and they are gathered only once per hour. We also collect public weather data to study the relationship between the air quality as assessed by our sensor stations and other weather information. To store multiple kinds of data sources in a uniform format, we store them in a mySQL database.

Thirdly, once the database has been collected, we concentrate on data calibration and data validation. As far as calibration is concerned, accurate and precise calibration models are particularly critical to the success of dense sensor networks deployed in urban areas of developed countries. In these situations, pollutant concentrations are often at the low end of the spectrum of global pollutant concentrations, and poor signal-to-noise ratio and cross-sensitivity may hamper the ability of the network to deliver reliable results. Keeping in mind this consideration, we perform our calibration phase with great care, using both the Multivariate Linear Regression Model and the Random Forest machine learning algorithm [27], [28]. The latter one, to the best of our knowledge, has been rarely applied to low-cost air quality monitor calibration. Moreover, we compare the results gathered with different calibration windows and we validate them over different periods to assess the quality of the result in time.

Finally, we perform some mobile tests. These include testing a semaphore capability of managing traffic and the connection with pollution, the pollution level to which dwellers are exposed during their daily mobility (using different means of transport), and the effect of sudden events on the air quality (such as wind and New Year's Eve fireworks). Even if the results gathered in these scenarios often seem to confirm common conjectures on pollution levels, in some specific case they represent unexpected situations deserving further analysis.

Contributions

To sum up, the main contributions of this chapter section are the following:

- The implementation of a special-purpose designed air quality monitoring station, hosting several low-cost sensors, with simple hardware and software architectures, and a quite high sample rate (one sample per second).
- A careful deployment of our monitoring stations in the city center, for a period of over 5 months during fall and winter seasons, when air pollution is higher due to home heating. This provides valuable training and testing data for our models, enabling a long-term evaluation of the entire system.
- Results of several calibration strategies with related validation data over different periods of time.
- Several experiments to investigate the air quality on mobile and dynamic-related events. These experiments are made possible by the high sample rate of our platform, well suited to analyze fast transient phenomena such as the pollution variation at the traffic light, or the dynamic pollution variation during New Year's fireworks.
- A completely open-source software architecture. The entire hardware architecture, software implementation, and the entire dataset collected during 5 months are made available [17].

Roadmap

The rest of this chapter section is structured as follows. Section 2.1.2 describes the related works on the topic of air quality monitoring. Section 2.1.3 presents an overview of our system hardware and software architecture. Section 2.1.4 describes our acquisition methodology and our measurement sites. Section 2.1.5 and Section 2.1.6 focus on the calibration and validation strategy adopted for our sensor stations. Sections 2.1.7 and 2.1.8 illustrate the data gathered by stationary and mobile station boards, respectively. Section 2.1.9 reports some final considerations and discussion on the lessons that can be learned from our analysis. Finally, Section 2.1.10 concludes the chapter section with some closing remarks, giving hints on possible future works.

2.1.2 Related Works

Urban air pollution has attracted great attention in recent years as it has been shown to be of a significant threat to the safety of city dwellers. Today, air pollution concentrations are usually monitored by environmental or government authorities using networks of fixed monitoring stations. Fixed stations obtain flawless air quality data, as they can provide very accurate measurements at the deployment locations. Nevertheless, these stations usually require significant investments in terms of cost and human resources to be built, operated and maintained. Thus, several low-cost alternatives have been proposed in recent years.

Randall [29] demonstrates that coarse-grained information about the air quality of the Earth's surface can be obtained by remote sensing using satellites. Although a large-scale area can be easily covered by only one satellite, the accuracy of this strategy highly depends on factors like weather conditions and land-use characteristics. Following Kawamoto et al. [30], a satellite-routed sensor system can increase accuracy, as data can be accumulated by a large number of sensor terminals, then gathered by the satellite, and finally transferred to the ground station. The problem of data collisions may be solved by adopting a "divide and conquer" approach to collect data on demand. The method achieves efficient data collection from numerous sensor terminals and minimizes all operational delays in the system. Nevertheless, the cost of any solution exploiting satellites remains extremely high.

In order to find a cheaper alternative, many recent works concentrate on deploying low-cost sensors. A large number of publications have reported the use of stationary or mobile laboratories with low-cost sensors to collect air quality data for specific purposes. For example, distributed or mobile personal measurement devices equipped with cheap commercial off-the-shelf dust sensors can reach meaningful accuracy at a cost one to two orders of magnitude lower than the one of the current hand-held solutions [31]. The same study also shows that participatory sensing, where co-located measurements are shared across different devices, can help reaching a high measurement accuracy. Moreover, the participatory sensing paradigm includes also subjective perceptions, such as posts by citizens on Online Social Network (OSN) platforms, which can enrich the mere sampling of the data [32]. Overall, two main research topics can be identified: Managing a distributed network for local sensing and developing low-cost sensors of air pollutants. These topics are deeply investigated in the recent scientific literature, as discussed in the following paragraphs.

A network of vehicles carrying sensors for flexible air quality monitoring is called a *vehicular sensor network* (VSN). For example, a VSN may consist of a set of cars equipped with gas sensors, wireless connection, and GPS receivers. Gathering big data efficiently in such densely distributed sensor networks is challenging. Adjusting the data sampling rates of cars can balance monitoring accuracy and communication cost as it prevents the transmission of similar data collected from close positions, thus alleviating network congestion [33]. Moreover, the wireless transmission should be optimized to avoid excessive energy consumption. The network is usually divided into sub-networks because of the limited wireless communication range. In this case, a proper clustering algorithm increases energy efficiency of data gathering by managing the mobile sink routing in the sub-networks [34]. Evidence shows that there are other successful implementations of VSNs aside from cars. For example, bicycles can carry sensors for air pollution monitoring [35], [36]. It is shown that even a limited set of mobile measurements makes it possible to map locations with systematically higher or lower ultra-fine particles and PM_{10} concentrations in urban environments. Unfortunately, the use of semi-professional equipment to monitor PM_{10} levels makes this experiment unsuitable for low-cost urban sensing scenarios. A different implementation exploits smart devices integrating sensors to build an architecture for people-centric environmental sensing platforms [37]. Smart objects and virtual node technology establish closed loops of interactions between people and physical devices. By aggregating on-demand user data from smart devices, it is possible to measure the space-time distribution of particulate matter. A case study of particulate matter exposure in New York City illustrates the potential application of such a system.

There are several issues in developing low-cost sensors for air pollution management in cities, among which, reliability, sensitivity, selectivity (different gases can contribute to the response of the sensors), stability, and longevity of operation before replacement [38]. Low sensitivity and poor signal quality can be addressed with sliding window and a low pass filter [39]. This approach is adopted in a real case study, where a wireless network of low-cost particle sensors is deployed in a woodworking shop. Data quality can be improved by identifying outliers in raw measurement data and inferring anomalous events. This task can be achieved by means of an anomaly detection framework composed of four modules [40]: Time-sliced anomaly detector (detecting spatial, temporal, and spatio-temporal anomalies in real-time sensor measurement data stream), a real-time emission detector (detecting potential regional emission sources), a device ranker (providing a ranking for each sensing device), and a malfunction detection (identifying malfunctioning devices).

Finally, particular attention must be paid to calibration, as this is a necessary step to obtain accurate measures. Zimmerman et al. [28] propose a multi-pollutant sensor package, which measures CO, NO_2, O_3, and CO_2, on which they compare three different calibration methods: Laboratory uni-variate linear regression, empirical multiple linear regression, and machine-learning-based calibration models using random forests. The evaluation reveals that only the sensors calibrated with random-forest approach meet the US EPA Air Sensors Guidebook [41] recommendations of minimum data quality for personal exposure measurement. A similar study by Bigi et al. [27] investigates the medium-term performance of a set of NO and NO_2 electrochemical sensors using three different calibration approaches: Multivariate linear regression, support vector regression, and random forest. The behavior of the sensing devices over time and after relocation was studied. It was noted that the performance of many algorithms strongly depends on the comparability of calibration and on the deployment area. The suitability of the devices for mapping intra-urban pollution gradients of NO and NO_2 was also studied. The devices could not reliably map small intra-urban gradients, thus they are not suitable for cleaner urban areas. Nevertheless, they can quantitatively resolve intra-urban concentration gradients on an hourly basis in higher polluted cities. Time and cost of the calibration of low-cost sensors can be reduced by firstly selecting sensors with similar responses. Then, a single on-site calibration for one sensor could be used for all sensors as the computed percent differences in the field are similar to laboratory results [42].

Although mobile sensing can be successfully applied to measure the concentration of gases, such as ozone [43], and ultra-fine particles [44], it is more frequently adopted in monitoring air pollutants like $PM_{2.5}$. For example, AirCloud is a cloud-based monitoring system for $PM_{2.5}$ concentration using affordable sensors [45]. At the front-end, two types of Internet-connected particulate matter monitors are adopted, i.e., AQM and miniAQM, with a mechanical structure optimized for inlet air-flow. On the cloud side, a novel air quality analytical engine calibrates the sensed data to improve accuracy. Overall, the project enables the adoption of low-cost sensors based on light scattering for public air quality monitoring. In Mosaic, another low-cost urban $PM_{2.5}$ monitoring system based on mobile sensing, monitoring nodes are first built with a novel constructive airflow disturbance design based on a carefully tuned airflow structure and a GPS-assisted filtering method [46]. Then, the buses are used for system deployment are selected by a novel algorithm that achieves both high coverage and low computation overhead.

2.1.3 System Overview

This section includes a description of our architecture from several points of view, going from the hardware and software architecture, to the communication protocols.

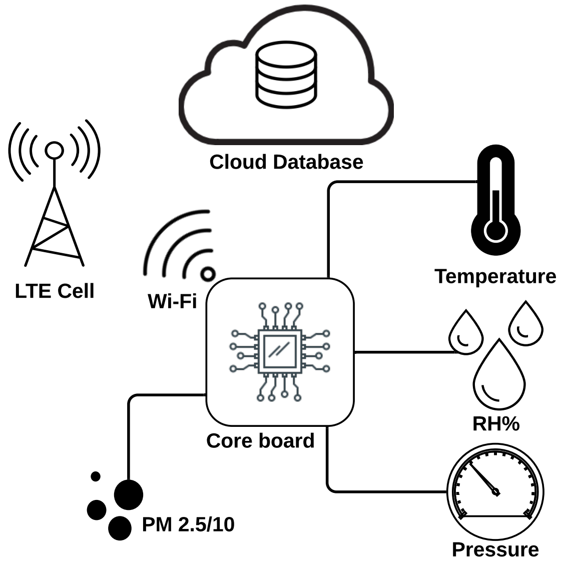
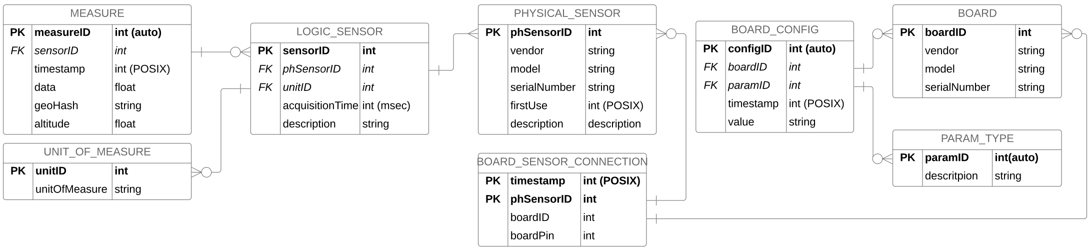


Figure 2.3: Architecture of the proposed system. The data coming from the sensors are first stored in Raspberry Pi, and then transferred to a remote server over the Wi-Fi network.

Hardware Architecture

We target the following key characteristics for our system: (1) rapid and easy prototyping capabilities, (2) flexibility in connection scenarios, and (3) cheapness but also robustness of components. As each board has to include a limited number of modules, to facilitate our prototype development, we select the Raspberry Pi (RPi) [47] single-board computer as the monitoring board. Due to our constraints in terms of cost, size, and power consumption, we chose the Zero Wireless [48] version based on the ARM® 11 microprocessor. A graphical representation of the architecture is available at Fig. 2.3.



MEASURE		
PK	**measureID**	**int (auto)**
FK	*sensorID*	*int*
	timestamp	int (POSIX)
	data	float
	geoHash	string
	altitude	float

UNIT_OF_MEASURE		
PK	**unitID**	**int**
	unitOfMeasure	string

LOGIC_SENSOR		
PK	**sensorID**	**int**
FK	*phSensorID*	*int*
FK	*unitID*	*int*
	acquisitionTime	int (msec)
	description	string

PHYSICAL_SENSOR		
PK	**phSensorID**	**int**
	vendor	string
	model	string
	serialNumber	string
	firstUse	int (POSIX)
	description	description

BOARD_SENSOR_CONNECTION		
PK	**timestamp**	**int (POSIX)**
PK	**phSensorID**	**int**
	boardID	int
	boardPin	int

BOARD_CONFIG		
PK	**configID**	**int (auto)**
FK	*boardID*	*int*
FK	*paramID*	*int*
	timestamp	int (POSIX)
	value	string

BOARD		
PK	**boardID**	**int**
	vendor	string
	model	string
	serialNumber	string

PARAM_TYPE		
PK	**paramID**	**int(auto)**
	descritpion	string

Figure 2.4: ER-diagram of the database, in Crow's Foot notation.

The basic operating principle of the system is the following:

- The data gathered from the sensors are stored in the MicroSD card of the

RPi board.

- At certain time intervals, the RPi tries to connect to a Wi-Fi network and, if such a connection is established, it uploads the newly acquired data to a remote server.

- The creation of the Wi-Fi network is achieved using a mobile phone set to operate a personal hotspot, while on the remote server the database storing all performed measurements resides.

Software Architecture

Wi-Fi connectivity was one of the requirements for the system, but at the same time, the system itself should not produce unnecessary electromagnetic noise, possibly impacting the operating ability of the host's appliances. To reduce the time in which the Wi-Fi connection was active, the Linux OS was set to activate the specific interface at predefined time instants to connect to the portable hotspot. Once connected to the network, the system performs the following tasks:

- Synchronization of the system and Real-Time Clock with a remote Network Time Protocol (NTP) server [49],

- Synchronization of the local samples directory with the remote directory residing on the server.

The latter task is performed using the UNIX `rsync` utility, which has to be installed on both machines.

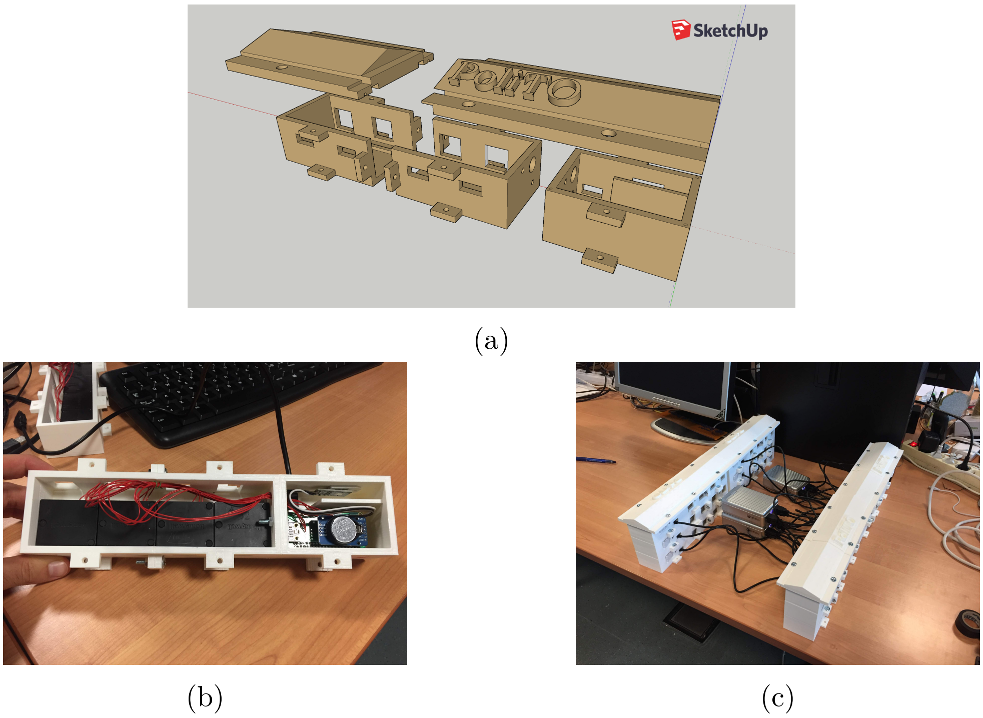
To gather data from the sensors, a `C` program has been implemented, which runs continuously on a separate process reading from each physical sensor plugged on the board and writing on the MicroSD card. It has to be noted that for what concerns the PM sensors, since the UART communication had to take place using GPIOs, a `Pigpiod` daemon [50] has been exploited to create digital serial ports over the Pi's pins.

The directories on the remote server are a simple copy of the MicroSD cards mounted on the board. Data in these directories have been inserted in a MySQL database with the structure depicted in Fig. 2.4.

Mechanical Design and Hardware Components

In order to easily stack more than one device together, a 3D printed modular case has been designed. Several enclosing frames can be tied together using nuts and bolts, with the use of a single cap on top. Fig. 2.5 shows the 3D board design, together with the final sensor and board configurations.

Each platform is equipped with 4 PM sensors (a good trade-off between size and redundancy), 1 Temperature (T) and Relative Humidity (RH) sensor, and 1



(a)

(b)

(c)

Figure 2.5: Board design, sensors, and final measuring station. The stackable modular 3D printed case (a), a single sensor board with 4 sensors (b), the set of boards used during the calibration phase (c).

Pressure (P) sensor. As our target is to capture significant data sampling for the particulate matter we adopt the following sensors:

- The Honeywell® HPMA115S0-XXX [51] as PM sensor. As one of our targets is to evaluate these sensors' suitability for air pollution monitoring applications, we insert 4 instances of this sensor in every single platform. This sort of redundancy allows us to detect strange phenomena and to avoid several kinds of malfunctions, making more stable the overall system.
- The DHT22 [52] as temperature and relative humidity sensor. This is very widespread in prototyping applications, with several open-source implementations of its library, publicly available on the internet [53].
- The Bosch® BME280 [54] as a pressure sensor. This is a cheap but precise barometric pressure and temperature sensor which comes pre-soldered on a small PCB for easy prototyping.

The system also includes a Real Time Clock (RTC) module for the operating system to retrieve the correct time after a sudden power loss, i.e., the DS3231 module. The

DS3231 communicates via *I2C* interface and has native support from the Linux kernel.

As a last comment, a Printed Circuit Board (PCB) was designed to facilitate connection and soldering of the various sensors and other components.

2.1.4 Data Acquisition and Measurement Sites

Our data acquisition campaign was carried out during Autumn and Winter seasons, from October 2018 to February 2019, in the city of Turin in north-western Italy. We deployed our stations to include a wide range of environments, such as residential boroughs, commercial areas, and parks. Each location corresponds to a particular GPS coordinate, POI (Point Of Interest), and sensor readings. Fig. 2.6 shows a map of the city of Turin. It represents the deployment positions of the stationary stations of some of our sensors and the paths followed by a few other sensor platforms during the analysis of dynamic and mobile events. Blue pins represent stationary sensors outside and inside the limited traffic zone. The orange pin represents the traffic light position in which we performed some dynamic analysis. The green path shows the roadway followed by our sensors on a bus, in a car, on a bicycle, and on foot. The red pin indicates the position of the ARPA reference station.

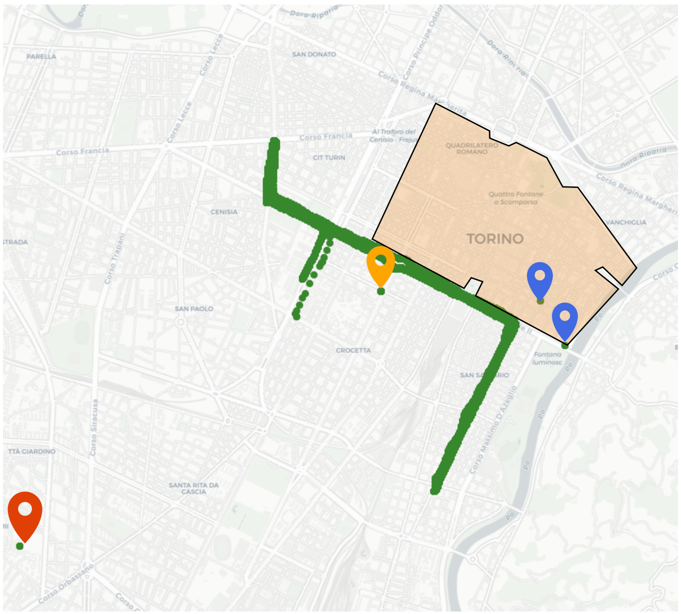
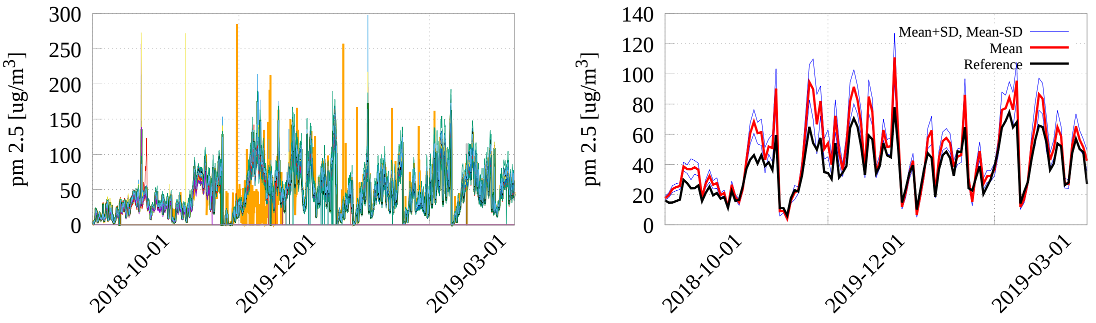


Figure 2.6: The pins represent the following objects: The ARPA reference station (red), the stationary sensors outside and inside the limited traffic zone (blue), the traffic light (orange), the green path shows the roadway followed by our sensors during the dynamic analysis.

Fig. 2.7 focuses on the $PM_{2.5}$ pollutant levels registered by all our stationary sensors, when placed on top of the ARPA reference station, over a period of 5 months. The actual location of the sensors is $1.5 - 2m$ far from the air inlet of the reference grade device on premises. Such a distance can be considered negligible since the monitoring station is located in a public park, far from vehicular traffic. Fig. 2.7a plots the readings of all sensors before calibration but with all plots vertically shifted to start from the same initial position. Plots consider our original data sample rate, i.e., one reading per second. This sample rate is definitely high, allowing us to analyze pollution levels both statically and dynamically, in terms of air pollution spatio-temporal variation. Fig. 2.7b represents the reference sensor readings with the mean values of all our sensors and the mean standard deviation, i.e., the mean values incremented and decremented by the standard deviation. To make the graph more readable than the one of Fig. 2.7a, we report daily averaged values. Fig. 2.8 reports the same data of Fig. 2.7b, i.e., the daily mean of the reference readings and of our sensor readings, within a scatter X-Y plot. The coefficient of determination (as computed by the `SciKitLearn` Python library [55] `r2_score`) is equal to 0.8267.



(a)

(b)

Figure 2.7: Collected data for all statically deployed sensors (in the ARPA station) measuring $PM_{2.5}$ (48 overall) for a period of 5 months (from October 2018 to February 2019). All sensors are uncalibrated but their initial offset is modified to make graphs coincide in the origin. Fig. 2.7a reports the time series for all sensors. Fig. 2.7b plots the reference, the mean and the mean standard deviation (mean ± SD) for all our sensors.

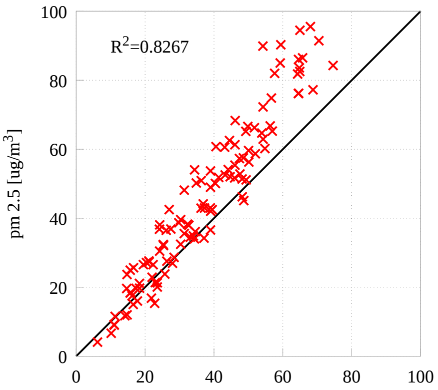


Figure 2.8: Scatter plot comparing the reference data with our sensor mean represented in Fig. 2.7b.

2.1.5 The Calibration Phase

The concentration of PM in the air and its measurement are affected by the weather conditions [56]. For instance, high values of relative humidity (RH%) make the particles tend to aggregate to water in the air, making them look bigger than they actually are. On the other hand, when it is raining or there is wind, the air gets cleaned, since the particles are taken to the ground or brought away. Low-cost light-scattering sensors are prone to such cross-sensitivities with other ambient variables [28], while the reference-grade devices implement different measuring techniques which, by construction, are not affected by such phenomena. Given the limitation of low-cost devices, one of the main primary requirements in open-air measurement is their calibration.

In general, linear regression has been identified as the main technique used for low-cost sensors calibration, since temperature and relative humidity follow linear patterns. Multivariate linear regression (MLR) analysis has been used to investigate several aspects of the air pollution over the years. For example, Chaloulakou et al. [57] used the regression models to investigate the complex relationships between the meteorological and time period parameters as factors controlling the PM levels. However, even if a sensor is calibrated, non-linearities sometimes appear due to the impurity and aging of low-cost sensing techniques. In these cases, accurate and precise calibration models are particularly critical, and there has been increasing interest in more sophisticated algorithms for low-cost sensor calibration [27]. Moreover, as reported by several researchers [58]–[60] artificial neural networks may give more accurate results than the multivariate linear regression model, mostly for PM_{10} forecasting, even though the difference is often not remarkable. As a consequence, we experimented with three different calibration methods, namely Multivariate Linear Regression (MLR), Random Forest (RF), and the Support Vector

Regression (SVR) model. As the last two methods delivered close results, we just concentrate on MLR and RF in the sequel.

Calibration Strategies

The Multivariate Linear Regression (MLR) Model

In Multivariate Linear Regression Models, regression analysis is used to predict the value of one or more responses from a set of predictors. Let $(x_1, x_2, \ldots, x_n)$ be a set of predictors (dependent variables) believed to be related to a response (independent) variable Y. The linear regression model for the j-th sample unit has the form

$$Y_j \quad = \quad \beta_0 + \beta_1 \cdot x_{j1} + \beta_2 \cdot x_{j2} + \ldots + \beta_r \cdot x_{jr} + \epsilon_j$$

where ϵ_j is a random error and the β_i are unknown (and fixed) regression coefficients. The value β_0 is the intercept. With n independent observations, we can write one model for each sample unit or we can organize everything into vectors and matrices as:

$$Y \quad = \quad X \cdot \beta + \epsilon.$$

The training data are used to calculate the model coefficients, and the model performance is evaluated on withheld testing data. Separate MLR models are usually developed for each sensor and each measure.

The Random Forest (RF) Model

A Random Forest Model is a machine learning algorithm for solving regression or classification problems. It works by constructing an ensemble of decision trees using a training data set. The mean value from that ensemble of decision trees is then used to predict the value for the new input data.

To develop a random forest model, we must specify the maximum number of trees that make up the forest, and each tree is constructed using a bootstrapped random sample from the training data set. The origin node of the decision tree is split into sub-nodes by considering a random subset of the possible explanatory variables. The training algorithm splits the tree based on which of the explanatory variables in each random subset is the strongest predictor of the response. This process of node splitting is repeated until a terminal node is reached.

The user can specify the number of random explanatory variables considered at each node, the maximum number of subnodes or the minimum number of data points in the node as the indication to terminate the tree.

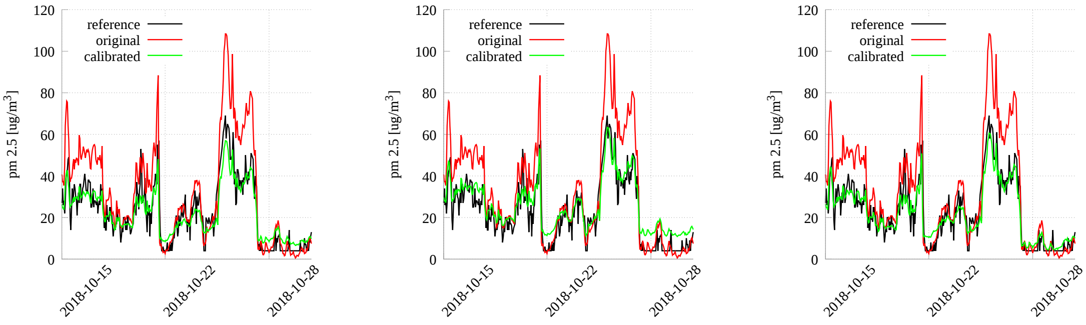
Our Calibration Process

As $PM_{2.5}$ is heavily influenced by meteorology factors, we exploit the dependencies between the sensor error and meteorology factors. More specifically, the

Honeywell® HPMA115S0-XXX Particulate Matter sensor has a relatively high precision ($\pm 15 \mu g/_{m^3}$ from 0 to 100 $\mu g/_{m^3}$), considering the extremely low price and the technology used. Nonetheless, it is required to calibrate the collected data to remove possible offsets and linearity errors.

For that reason, during the calibration period, all sensors have been placed near the stationary ARPA station in the city of Turin, which exploits the β-radiation technology to provide high precision measures. ARPA provides hourly average data which have been used as a reference for all data collected from the sensor boards. Please note that hourly average data, obtained with β-radiation approach, are fully consistent with gravimetric sensor measurements.

As far as the boards are concerned, we first apply different filters to remove outliers and possible undesired data. Then, we compute the window average with variable width to smooth the samples. After that, we group the values collected on a per-second basis to have hourly measures directly comparable with the ARPA values. Most samples fall in the $\pm 15 \mu g/_{m^3}$ range, which is reasonable considering the sensitivity of the sensor. Finally, we perform calibration on the hourly average samples. As previously introduced in this section, we consider Multivariate Linear Regression and Random Forest. Nevertheless, we apply three types of MLR, i.e., using only the temperature, only the humidity, and both temperature and humidity. In all cases, the calibration using RF considers both temperature and humidity.



(a) (b) (c)

Figure 2.9: Time series comparing the reference, the raw, and the calibrated data using sensor 34, randomly selected. Calibration is performed using MLR. The different plots show MLR with different dependent variables, namely temperature (a), humidity (b), and temperature plus humidity (c).

2.1.6 The Validation Phase

Validation Strategies

The way to quantify the accuracy of a fitting model is by minimizing some error function that measures the misfit between the output and the response function for any given value of the data set. In the following, we will use several metrics defined as in the SciKitLearn Python library [55]. We will indicate with y_i the true value of the i-th sample, $\hat{y}_i$ the corresponding predicted value, and $\bar{y}$ as the mean of the true samples:

- The coefficient of determination is the proportion of the variance in the dependent variable that is predictable from the independent variable:

$$\mathrm{R}^2(y,\hat{y}) \quad = \quad \frac{\sum_{i=0}^{n_{\text{samples}}-1}(\hat{y}_i-\bar{y})^2}{\sum_{i=0}^{n_{\text{samples}}-1}(y_i-\bar{y})^2} \tag{2.1}$$

- The mean squared error (MSE) measures the average of the squares of the errors. It is the second moment (about the origin) of the error and thus incorporates the variance of the calibration curve:

$$\mathrm{MSE}(y,\hat{y}) \quad = \quad \frac{1}{n_{\text{samples}}} \cdot \sum_{i=0}^{n_{\text{samples}}\,-1} (y_i-\hat{y}_i)^2\,, \tag{2.2}$$

- The Mean Bias Error (MBE) is usually adopted to capture the average bias in a prediction.

$$\mathrm{MBE}(y,\hat{y}) \quad = \quad \frac{1}{n_{\text{samples}}} \cdot \sum_{i=0}^{n_{\text{samples}}\,-1} (y_i-\hat{y}_i)\,. \tag{2.3}$$

- The root mean square error (RMSE) allows comparing different sizes of data sets, since it is measured on the same scale as the target value. It is obtained as the square root of the MSE, i.e.,

$$\mathrm{RMSE}(y,\hat{y}) \quad = \quad \sqrt{\mathrm{MSE}(y,\hat{y})}. \tag{2.4}$$

- The CRMSE is the Root Mean Square Error (RMSE) corrected for bias, i.e., it is defined as:

$$CRMSE \quad = \quad RMSE \cdot sign(\sigma_{model} - \sigma_{reference}) \tag{2.5}$$

  where σ is the standard deviation of the measure.

- The correlation coefficient (Pearson product-moment correlation coefficient) is defined as the covariance of the variables divided by the product of their standard deviations.

$$\rho_{y,\widehat{y}} \quad = \quad \frac{cov(y,\widehat{y})}{\sigma_y \cdot \sigma_{\widehat{y}}} \tag{2.6}$$

Our Validation Process

To test the performance of the two different calibration models, we first calibrate our sensors using the data collected in the first 2 weeks of October 2018. Then, we validate these calibration methods using samples collected in the last 2 weeks of the same month. In this period, we compare the concentrations obtained after the calibration with the measured reference concentrations.

For the sake of simplicity, Fig. 2.9 shows our results for one single sensor (sensor 34), randomly selected, using the MLR model. For this model, the three plots present the data gathered using as dependent variable only the temperature, only the humidity, and both the variables as free variables. For all graphics, the calibrated plot is far more stable than the original one, but there is no clear winner among the three strategies.

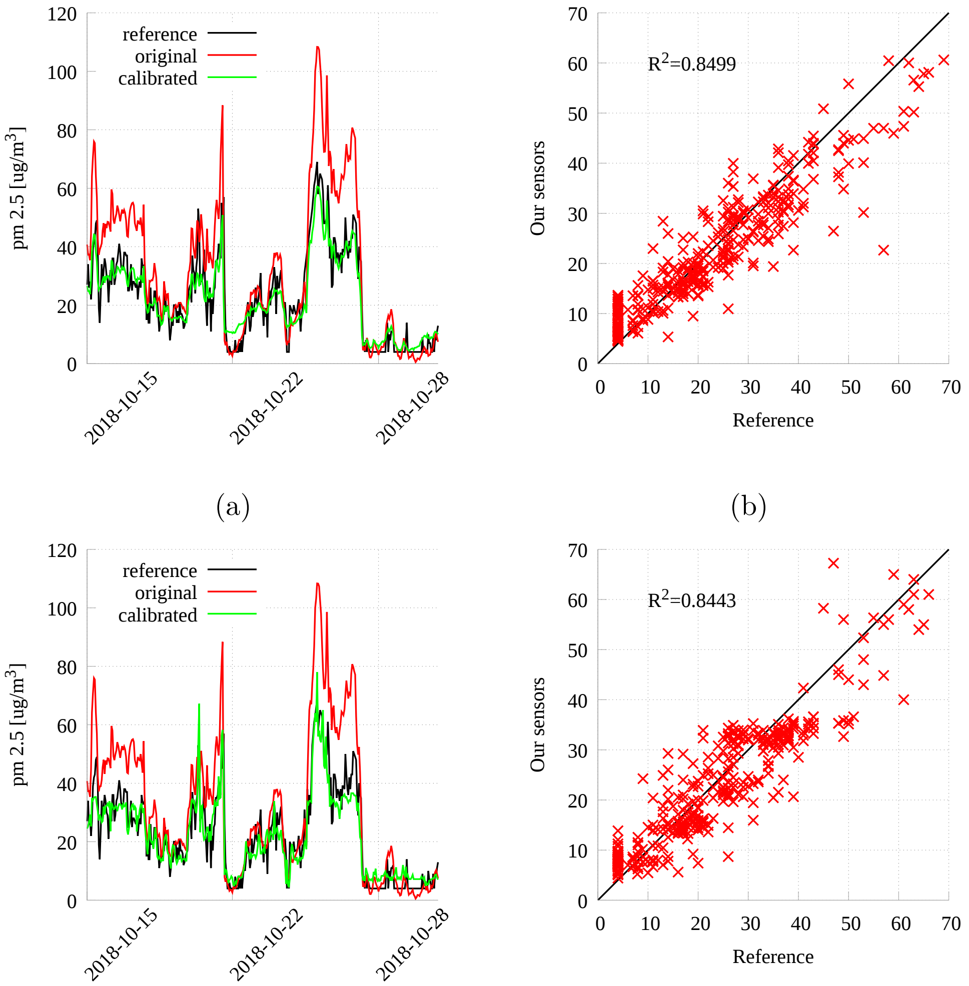
To deepen our analysis, Fig. 2.10 compares the MLR model with the RF one (again using sensor 34). As for Fig. 2.9 calibration is performed during the first 2 weeks of October and validation during the last 2 weeks. In this case, we use both temperature and humidity as free variables. The charts report time series (Fig. 2.10a and 2.10c) and scatter plots (Fig. 2.10b and 2.10d). Somehow unexpectedly (please see Zimmerman et al. [28]) our results show no advantage for the RF model with respect to the MLR one. On the contrary, the MLR model seems to outperform the RF model.

To better evaluate our results and to better assess the overall model performance, we performed calibration and validation tests for longer periods. A secondary target of this analysis is to find the best trade-off between the calibration effort and the error obtained. We consider calibration periods varying from the 2 weeks used so far up to 12 weeks, starting in October and ending in November 2018. In all the cases, the validation period has been selected in December 2018.

		2 weeks			6 weeks			12 weeks		
		323 samples			840 samples			1851 samples		
		T	H	T+H	T	H	T+H	T	H	T+H
LR	RMSE	19.03	14.17	14.42	18.96	12.58	14.70	13.49	10.04	11.66
	Correlation	0.89	0.88	0.89	0.89	0.88	0.89	0.88	0.88	0.89
RF	RMSE	25.41	21.77	23.50	25.28	18.75	19.37	13.85	12.63	11.33
	Correlation	0.82	0.80	0.78	0.77	0.82	0.80	0.85	0.88	0.89

Table 2.1: Comparing different calibration techniques over different calibration periods (2, 4, and 6 weeks, respectively), adopting the RMSE (Root Mean Square Error) metric. We consider all stationary sensors.

Table 2.1 reports the RMSE (i.e., the Root Mean Square Error, computed as defined by Equation 2.4) and the data correlation (computed following Equation 2.6)



(a) (b)

(c) (d)

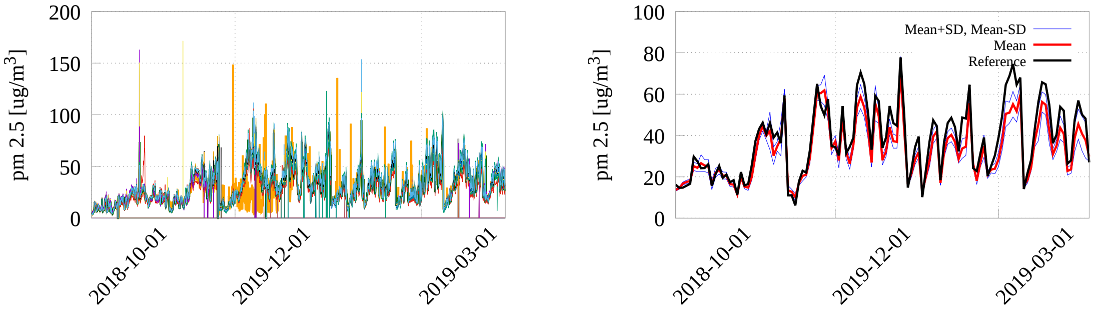
Figure 2.10: Comparing different calibration techniques on our sensor network using sensor 34, randomly selected. The graphics compare linear regressions, with temperature and humidity as the dependent variables (a and b), versus random forest (c and d). The figure reports time series (a and c) and scatter plots (b and d).

for a representative sensor with different calibration periods (2, 6, and 12 weeks, respectively).

Final Considerations

MLR has been used during calibration because it is the strategy historically adopted for this phase and it is the one that delivers the best results in our environment. Anyway, we also experimented with the RF and the Support Vector Regression (SVR) models. These latter methodologies are supposed to perform better with non-linear measures [27], [28] and they delivered very similar results. Actually, the results we gathered using RF look slightly disappointing in our environment. In our opinion, this may be due to three different reasons. First of all, an important difference between MLR and RF is that MLR allows to extrapolate outside the range of its input data-set, while the estimates provided by RF can only be within the bounds of the calibration space. This behavior can motivate our results as in our framework short calibration periods have been followed by long working sessions with wider measure variations. This consideration is strengthened by the data of Table 2.1, which show how the computed errors (RMSE and correlation) seem to decrease the longer the calibration period. Secondly, we focused on particulate matter, whereas Bigi et al. and Zimmerman et al. [27], [28], who reported favorable results for nonlinear models, concentrated on CO, NO_2, O_3, and CO_2. We can then conclude that MLR works at its best for our low-cost sensors measuring particulate matter. Thirdly, many works estimating the concentration of PM_{10} obtained a much lower accuracy than the one gathered by us. For example, Alam et al. [60] claimed a R^2 coefficient equal to 66% in the best case. On the contrary, we obtained values close to 90% for the same metric. Thus, with this level of accuracy, nonlinearity effects may have a much lower impact and RF may not be able to improve the results obtained with MLR.

Overall, another remarkable result is that we obtained a better performance of the regression process using only the relative humidity instead of both relative humidity and temperature. This makes it possible to simplify the hardware (and software) structure of our platform by removing the temperature and pressure sensors, as they do not provide any significant advantage when performing calibration. It is also important to note that measuring temperature is not easy in our case, because the sensors present cross-heating, mainly due to the heat produced by the particulate matter sensors, which use more electrical power. Better results in measuring temperature could be obtained by putting a separate sensor in a different box, but this would negatively impact on the size and portability of the global system. Atmospheric pressure is simpler to be measured, but it is less significant than temperature. Relative humidity is by far the most important parameter for calibrating PM sensors. Furthermore, relatively short calibration periods (2 to 4 weeks) perform quite well compared to considerably longer periods (10 to 12 weeks), thus reducing the necessity of performing long calibration sessions.



(a)

(b)

Figure 2.11: Collected data for all statically deployed sensors (in the ARPA station) measuring $PM_{2.5}$ (48 overall) for a period of 5 months (from October 2018to February 2019). All sensors are calibrated during the first two weeks of October, using MLR with humidity as the dependent variable. Fig. 2.11a reports the time series for all sensors. Fig. 2.11b plots the reference, the mean and the mean standard deviation (mean ± SD) for all our sensors.

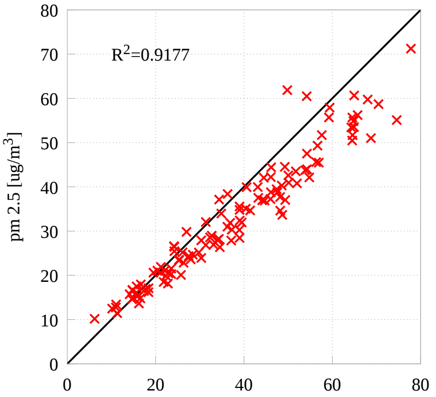


Figure 2.12: Scatter plot comparing the reference data with our sensor mean represented in Fig. 2.11b.

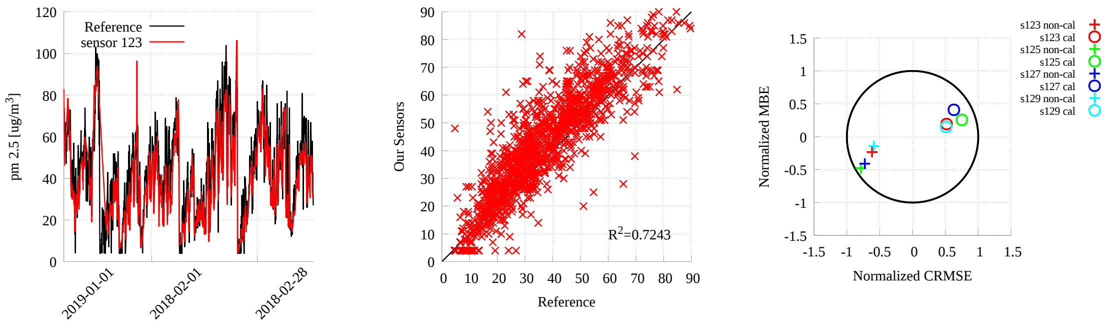
2.1.7 Stationary Platforms

Once the previously described calibration and validation phases have been performed, we analyze air quality and intra-urban pollutant gradients over long periods of time.

Following Fig. 2.7, Fig. 2.11 and Fig. 2.12 focus on the $PM_{2.5}$ air pollutant levels registered by all our stationary sensors, placed in the ARPA station, over a period of 5 months. All sensors have been calibrated using MLR with humidity as the dependent variable. The graphics report: The time-series plot for all sensors with one reading per second (Fig. 2.11a), the mean and mean standard deviation using daily averaged values (Fig. 2.11b), and the scatter X-Y plot (Fig. 2.12) comparing the reference values with our mean ones. Compared with Fig. 2.7, Fig. 2.11 shows much lower peaks, a much lower deviation, and more regular behavior. The coefficient of determination based on the data plotted in Fig. 2.11b is 0.9177, higher than the one computed for Fig. 2.7b.

Fig. 2.13 compares our readings with the reference one for a period of about 2 months. Given a randomly selected platform, we first calibrated its 4 sensors (namely sensors 123, 125, 127, and 129). We adopt the MLR calibration model, with relative humidity as the dependent variable, over 2 weeks. Then, we analyze the behavior of the 4 sensors over 12 weeks. Fig. 2.13a reports the time-series, and Fig. 2.13b the corresponding scatter plot, for sensor 123. As it can be easily noticed, the readings and fluctuations are very similar to the reference ones for the entire period. The other sensors are not reported for the sake of brevity, but they show a very similar behavior. This can be verified by the target diagram [61] of Fig. 2.13c, which considers all 4 sensors. In a target diagram, the x axis indicates the CRMSE (computed as in Equation 2.5) and the y axis the MBE (please refer to Equation 2.3), both normalized by the standard deviation of the reference $\sigma_{reference}$. As a consequence, the vector distance between any given point and the origin is the RMSE normalized by the standard deviation of the reference measurements. Again, all sensors deliver points within the unit circle both before and after calibration, with the second set of points closer to the origin than the first one.

To analyze intra-urban pollutant gradients, Fig. 2.14 shows a test performed with two platforms (each one including 3 sensors) placed at the border and inside the Limited Traffic Area (LTA) in the city of Turin. This test has been repeated for two days, on December the 3rd, and on December 4th, 2018, from 4:00 p.m. to 4:30 p.m. (local time). These dates have been selected due to the fact that they were at the beginning of an emergency traffic control, which has been in progress until December 8th, due to unhealthy pollution levels in the previous days. The time is representative of the rush hour, with significant traffic increments with respect to other hours. At first glance, it can be noted that albeit all sensors of each station delivered very similar readings, the measurements carried out inside the LTA are significantly higher than those collected outside. A few considerations may address this issue. The location outside LTA is indeed a very crowded crossing, but it is located in an open area, in the immediate proximity of the Po river, the longest river in Italy. These factors may greatly impact the concentration of PM in the air since the area is very wide and the river could induce current flow just by its own motion. On the contrary, the location inside the LTA is represented



(a) (b) (c)

Figure 2.13: Time-series (a) and scatter plots (b) for the data coming from sensor 123 over 2 months. Calibration was performed with MLR, with humidity as the dependent variable, for a period of 2 weeks. The target diagram of figure (c) show the 4 sensors (123, 125, 127, and 129) of the board, considering both uncalibrated and calibrated measures.

by the crossing of two narrow streets. This situation, combined with the height of the surrounding buildings (up to 7-story), could greatly reduce air circulation, resulting in concentration values that are higher with respect to the outside location. It has to be pointed out that this test does not undermine the effectiveness of LTA regulations in reducing pollution concentrations in the urban center. Nevertheless, it gives some insights and hints on possible studies which could be carried out to increase the level of understanding of the phenomena happening in city areas.

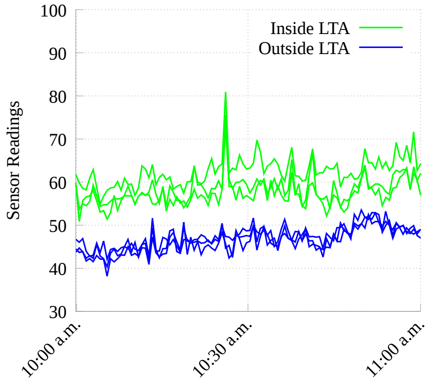
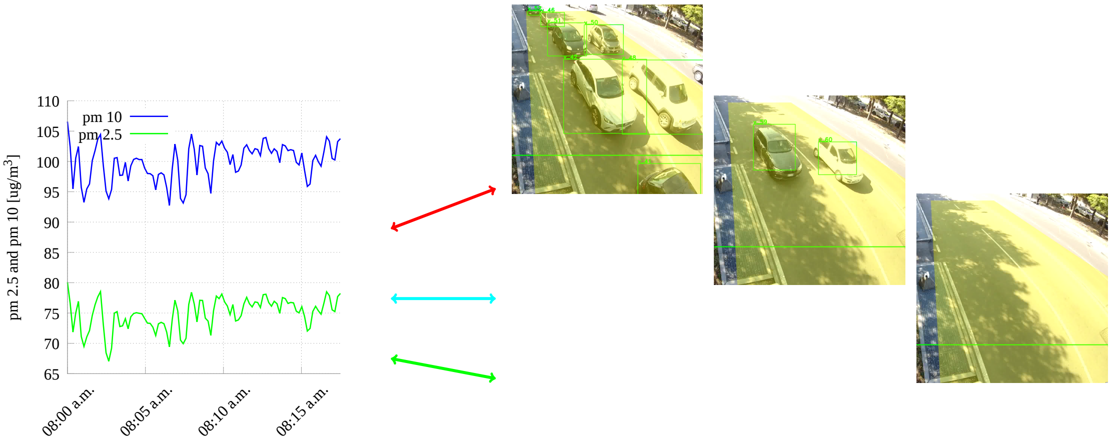


Figure 2.14: A time-series comparison between two different sites, one inside and the other one outside the Limited Traffic Area.



(a) (b) (c) (d)

Figure 2.15: Time-series comparing different traffic conditions at a traffic light. Three different conditions (high traffic (b), average (c) and low traffic (d) levels) repeat themselves continuously. The test has been performed the 8th of January 2020.

2.1.8 Mobile Platforms

In parallel to the systematic data acquisition and calibration campaign, we carried out targeted experiments to investigate the robustness and accuracy of the system in dynamic use cases. As our architecture is able to collect one sample per second, we are able to analyze sharp events in terms of air pollution spatio-temporal variation. This study also allows us to investigate to what extent a limited set of mobile measurements permit us to draw conclusions on global urban air pollution.

Traffic Light Case Study

Fig. 2.15 analyzes the pollution levels registered in close proximity of a traffic light within the city center at rush hour. It reports the level of particulate matter registered for a period of 15 minutes (from about 08:00 to about 8:15 a.m.), including several traffic-light cycles. The time-series shows a wave behavior essentially highlighting 3 different traffic conditions at the traffic light. The higher levels, around 102 $\mu g/_{m^3}$ for PM_{10} and 77 for $PM_{2.5}$, were recorded with a long queue during red traffic lights. Intermediate levels, around 97 $\mu g/_{m^3}$ for PM_{10} and 72 for $PM_{2.5}$, were recorded with average traffic, i.e., a few cars passing by. The lower levels, with a concentration of around 93 $\mu g/_{m^3}$ for PM_{10} and 69 for $PM_{2.5}$, were recorded with no traffic. Overall, the data show high variability in time, as it can be expected due to the vehicles' stop-and-go at the traffic light [62], and our platforms demonstrate a very good reactivity to transient events.

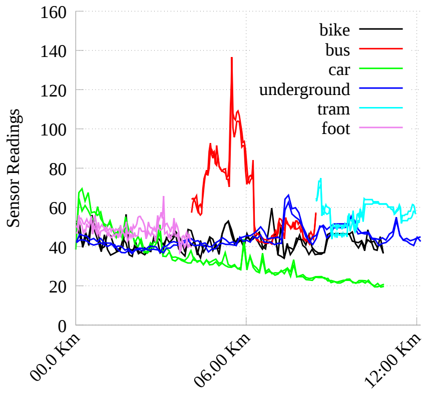


Figure 2.16: A time-series comparison among different means of transport along the same path (reported in Fig 2.6).

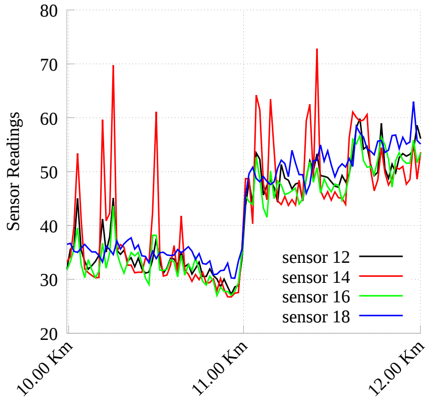


Figure 2.17: Effect of opening a window (halfway on the x-axis) on a car traveling along the green path of Fig. 2.6.

Citizens' Mobility Case Study

Following [63], Fig. 2.16 focuses on pollution levels recorder with portable platforms using different means of transportation along the same route. The path followed, i.e., a 4 Km path crossing the city center of Turin, is represented by the green route in Fig. 2.6. The different concentrations are plotted with distinct colors depending on the transportation mean. The two transport means which registered the highest pollution concentrations are the tram and bus. Both of them carry dozens of passengers and run in the street, alongside the usual fossil fuel-propelled vehicles. Analyzing the bike plot, it is possible to guess that the fluctuations of the pollution level mainly depend on the varying number of cars and trucks the user had to follow along her path. Anyhow, if we compare the bike plot with the one

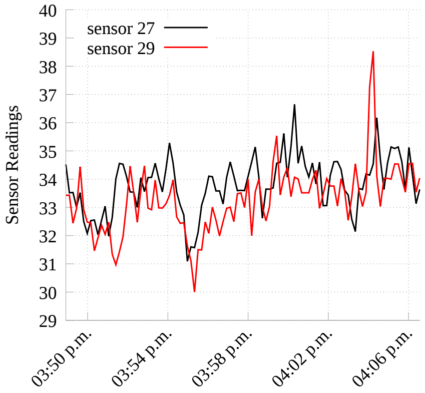


Figure 2.18: Registered pollution on a bike running at different speeds (from about 8 Km/h to about 27 Km/h).

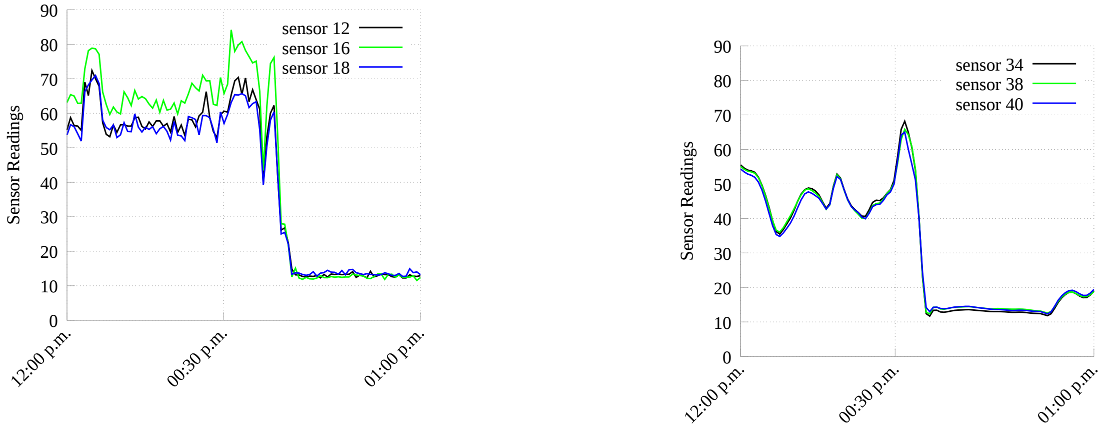
obtained on foot, the bike user is exposed to a lower pollution level. The data gathered underground are almost constant, thanks to the air recirculation and filtering devices used in metro stations and metro trains.

The plot of the car shows concentration levels in the cabin gradually decreasing, due to the car's air recirculation active. From the driver's point of view, the car looks like the best mean of transport among the ones analyzed. To further analyze this issue, Fig. 2.17 shows the effect of opening the car window during the previous experiment. It is possible to observe an abrupt increase of pollutants in the middle of the graph.

Finally, Fig. 2.18 shows the pollution registered on a bike trip, running the bike at different speeds. The target was to verify whether the bicycle speed has some influence on the pollution level to which the rider is exposed. The bike was running at about: 8 Km/h from 15:50 to 15.54, 13 Km/h from 15:54 to 15.58, 23 Km/h from 15:58 to 16.02, and at 27 Km/h from 16:02 to 16.06. The graphic shows that there is no much difference in the pollution level encountered while running the bike at different speeds.

Wind analysis

Fig. 2.19 compares the pollution before, during, and after the föhn wind arrived in the city of Turin, on October 28th, 2018. This brought an increase in the temperature, but also naturally helped cleaning the air in the city. We follow the event with 2 stations, generating the graphics in Fig. 2.19a and 2.19b respectively. The two readings are very coherent, considering that Fig. 2.19a reports one sensor readings per second, whereas Fig. 2.19b considers hourly averaged values.



(a) Platform A (sensors 12, 16, and 18).

(b) Platform B (sensors 34, 38, 40).

Figure 2.19: Time-series representing the pollution variation when the wind comes on two stations each one equipped with three $PM_{2.5}$ sensors.

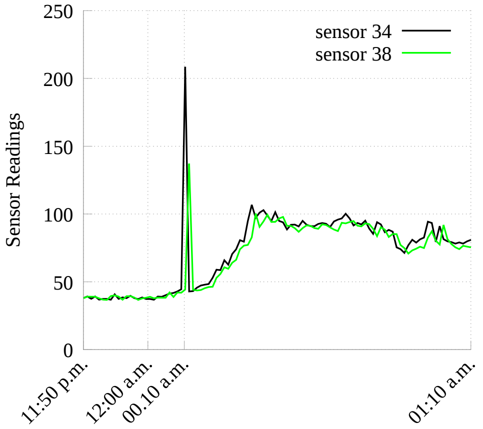


Figure 2.20: Pollution level during the New Year Firecrackers (data measured near the ARPA station).

New Year's eve fireworks

Fig. 2.20 shows the pollution level during the New Year Firecrackers in Turin. The quick rise of PM concentration is reported about 10 minutes after midnight, with a peak of about 4 times higher than the average value before the event. Moreover, the level remains twice as higher than before the event for more than 1 hour. From this plot it is possible to deduce that a certain amount of firecrackers were lit in the immediate vicinity of the stations, giving the first peak. The fast increase is then due to diffusion of dust from the city center to the suburbs, where the sensors were located. After reaching this maximum value, the level of pollution slowly starts decreasing again thanks to the diffusion.

2.1.9 Data Analysis and Lessons Learned

Since our system has been designed to appropriately scale to difficult environments (like factories), it is important to underline the practical difficulties we encountered to design and to build it. Developing the hardware and software system has required a considerable team effort in terms of man-hours. From the hardware point of view, using more than one PM sensor for each station (in our case 4) has been very useful to better compare the behavior of each low cost sensor. The high number of sensors stacked in each board is for sure a point of strength of this study when compared with previous ones, in particular for its use over very long time periods. From the software point of view, the database grows very fast and it requires a powerful computer to be managed efficiently. To have a fast response to SQL queries, we used a Dell computer with 20 cores, 384 GB of internal memory, and some TB of hard-disk. Smaller computers would have potentially implied very slow query evaluation. Many different discussions can be done using the data and the system. The main focus here is on the system itself and on some feasibility studies in order to demonstrate how well and accurately it works, and how reliable it is. We plan to use the system for specific purposes, like environment studies of air quality in interior settings, in addition to all uses already introduced here. For that reason, all data (i.e., the entire database), software, and hardware designs are available on IEEE *DataPort* in Open Access mode [17] in order to make easier further evaluations and facilitate comparisons with other approaches.

As far as the calibration phase is concerned, we can make the following observations. Even if RF has the ability to build a nonlinear regression model and has been proved to be superior to MLR in handling measures on CO, NO_2, O_3, and CO_2, in our analysis MLR performs better. This somehow motivates its wide use to calibrate low-cost sensors. The difference between MLR and RF is that MLR allows to extrapolate outside the range of its input data-set, while the estimates provided by RF can only be within the bounds of the calibration space. This behavior, due to the intrinsic nature of RF, based on tree manipulations, can partially motivate our data, as short calibration periods have been followed by long working sessions with wider measure ranges. Another important aspect of our calibration phase is that MLR proved to be more accurate when using only the relative humidity as dependent variable instead that the humidity and temperature together or only the temperature. This was somehow unexpected, and it may lead to a possible simplification of the hardware and software platforms, which may avoid collecting and storing data coming from the temperature and pressure sensors. As previously stated, simplifying the hardware platform may be very useful and measuring the temperature is difficult for the cross-heating effect between sensors due to the electrical power used by the PM sensors.

Another important aspect we focused on was to understand whether the tested sensor units are appropriate to capture particulate matter concentration with high resolution. From that point of view, our results are comfortable, as our platform may significantly improve our ability to resolve spatial and time heterogeneity in air pollutant concentrations. We have to consider that numerous factors are involved in the variation of the atmospheric pollution, and often the relationship between the intensity of emissions produced by the polluting source and the resulting pollution is not immediate. For that reason, our sample rate (one sample per second) may even be considered higher than necessary to capture even dynamic phenomena. Among the future works, we would like to mention the necessity to evaluate the best possible trade-off between the sample rate, the dimension of the data-base (that can become really large), and the ability to follow high gradients.

As our system is based on boards including 4 sensors, another feature of our platform is the combination of information coming from a multitude of sources. There are several issues that arise when fusing information from multiple sources, but the most fundamental one is to combine the information in a coherent and synergistic manner to obtain a robust, accurate and reliable description of the quantities of interest. One of the feature we are working on at the moment is to model the inherent uncertainties in the sensor measurements due to sensor aging, and to spurious data readings. Even if there is little literature regarding recent advanced fusion techniques and emerging applications, we are developing a fusion strategy based on Bayesian methods that can identify the inconsistency in sensor data so that spurious data can be eliminated from the sensor fusion process.

2.1.10 Conclusion

We approach the challenging problem of accurate and affordable $PM_{2.5}$ monitoring by orchestrating several low-cost sensor units deployed within an urban scenario. More specifically, we first carefully design and build our sensor stations with a high level of sensor redundancy (4 particulate matter sensors are embedded in every station). Then, we placed them on the field and we field-calibrated them. Finally, we deployed them to measure the air quality on stationary monitoring sites and during specific and sudden events.

From a broad perspective, our findings investigate whether the adopted sensors are fit for the intended purpose and the intended environment. Given their low cost, it is possible to deploy a large number of monitoring stations throughout a city providing a spatial dense coverage. Sensor redundancy may help in reducing errors of the overall system. They proved to be extremely easy to calibrate, as a short calibration time (less than 2 weeks) and simple calibration methods (MLR) are sufficient. As they offer a very high sample rate (one sample per second) they are able to follow sudden changes in the environment, providing a temporal dense coverage. One of our goals is that the developed system can be used by researchers and practitioners in order to estimate the level of air pollution and investigate the general behavior of particulate matter.

As future work, we hope to use our sensor stations to further improve our model and to solve a number of environmental problems, such as identifying pollution sources and air quality prediction. Moreover, we would like to better analyze the issue of sensor aging and de-calibration and better use the redundancy present in each monitoring station to make each station more reliable and resilient. Furthermore, we would like to improve the experiments in the Participatory Sensing direction, enabling university students to install a monitoring station at their home.

2.2 Analysis of Duty-Cycle operation

This chapter section reports the analysis on the impact of changing duty cycle in light scattering PM sensors. We have to keep in mind that this kind of activity is not useful in certain cases, such as when the time variability of the PM concentration in the air is too quick to be followed, for example in high-traffic locations. Nevertheless, such an approach can indeed be implemented in situations where the PM concentration is sufficiently low. The analysis reports that in such situations this technique can yield great advantages w.r.t. energy saving and sensor lifespan extension.

2.2.1 Introduction

As introduced in the previous section, low-cost portable PM monitoring devices based on the light-scattering-particle (LSP) are surging. Their operation consists of a small chamber with an air inlet and an air outlet. The sensor pulls air via a fan into a detection chamber where a low-power light source (laser) illuminates the airflow. According to the size and concentration of the particles, the light is scattered and detected by a photo-diode.

Air pollution concentration shows large spatial variability, depending on a series of factors, among which the presence of polluting sources and the structure of buildings in a certain area[64]. This is the reason for which more granular air pollution monitoring activities should be put in place. Such activities could be carried out by exploiting portable, cheap but reliable devices to reduce the installation costs while improving the coverage and pervasiveness of the network.

In this domain, an IoT deployment comes as an alternative to sense and provide a reliable PM concentration map. However, the designing of IoT hardware devices entails several constraints to be considered. Some of them are network availability, power limitations, and data reliability. LSP sensors can provide values in periods around 1 second, and constant measurements can increase the amount of data to process and transmit. Moreover, the fan embedded in the sensor consumes most of the power. The majority of the sensors used to measure PM_{10} and $PM_{2.5}$ have a current consumption close to 90 mA [65]. A deployment with limited network throughput or power budget for data transmission represents a trade-off for battery autonomy.

Different energy management schemes and techniques were explored in the literature to prolong battery autonomy. In the case of hungry sensors as LSP, methods such as *data reduction*, *data prediction*, or *sleep/wake-up (duty cycle)* have been proposed [66], [67]. The *duty cycle* method also presents the most energy-saving for LSP sensors; they can reduce their power consumption to at least $20mA$ by shutting off the fan or the heater, and this is desirable for mobile or battery-dependent devices.

However, LSP sensors face some challenges in data accuracy [68]. The first one is related to the light scattering effect, where it is correlated to the particles' composition and shape, and also environmental factors such as temperature and relative humidity. Therefore, some experiments showed better precision performance with in-situ calibration [69]. Other error sources are the aging and drift produced by dust deposits inside the air chamber that restrain the light intensity over the photodiode or diminish the light power [70]. So, shutting-off the fan also cuts the airflow inside the detection chamber, and it can increase the stacking of matter, and this affects the measurement accuracy.

This chapter section analyzes the effect of accuracy drift, and error in LSP sensors with different duty cycles in a real PM measurement scenario to determine the measurement repeatability and the impact of the measurement degradation compared with an always-on LSP sensor.

This chapter section is organized as follows: Section 2.2.2 gives an overview of the related works in this domain; Section 2.2.3 describes the experimental setup, giving details about the sensor selection, HW and SW architecture developed, methodology adopted, and measurement location; 2.2.4 analyses the results obtained; Section 2.2.5 draws conclusions and gives insights about possible future extensions of this work.

2.2.2 Related Works

The literature on low-cost air pollution monitoring is very rich [17], [19], [71]–[78]. Several of these papers carry out the measurement of air pollution either in an indoor environment, an outdoor one, or both. Among them, some evaluate the performance of different sampling frequencies, also relating this to the energy consumed while operating the sensors. A few try to address the issue of energy optimization in the sense of reducing the power consumption and its implication on the accuracy of the sensors. Some articles describe different sampling periods or the use of a type of duty cycle to prolong the battery life. Despite doing this, to the best of our knowledge, no paper describes the effect on the accuracy of measurements by implementing long intervals between subsequent samples. Only some of these papers mention this as a possible future work.

In [73] the authors design a power management technique to schedule the operation of energy-hungry sensors and wireless transceivers. Their contribution to the energy consumption reduction is three-fold: at the sensor level, operating the gas sensor in a pulse-like mode and performing early detection of safe concentration conditions; at the node level, managing deep sleep and duty cycled activity depending on people presence in the area; at the network level, by enhancing the lifetime of each node and exploiting information of neighbor nodes. Unfortunately, this paper does not report any analysis on both the sampling rate used to operate the sensors and the energy efficiency.

In [74], authors propose a simple WSN-based air quality monitoring system for industrial and urban areas. This work introduces an energy efficient routing protocol named CPAS (Clustering Protocol of Air Sensor network), in which multiple base stations periodically transmit data between them. However, the paper does not provide any justification for the sampling rate chosen, picking the sensing period in the range $200 - 300s$.

In [75], authors present the design, implementation, and evaluation of Mosaic, a mobile sensing system for low-cost urban air quality monitoring. This device embeds a PM 2.5 sensor and an anemometer to take into account the air flow at speed when mounted on public transportation vehicles. The paper is very interesting and presents a novel algorithm to select the buses, achieving high coverage with a low computational overhead. However, the energy autonomy of the system and the precision of the model is not clearly described.

Authors of [76] investigate the use of WSN for air pollution monitoring projects in Mauritius. They designed an aggregation algorithm to extrapolate data from the large number of sensors deployed in the island. This algorithm aims to merge data while eliminating duplicates, filtering out spurious sensor readings and summarizing them in order to reduce the amount of data transmitted to the sink in order to save energy. Moreover, to further increase the energy reduction, they also implemented a power management scheme in the hierarchical routing protocol that made the nodes go to sleep during idle time. Despite all this, we have to keep in mind that, in a crowd sensing application, the data availability depends on the gathering data rate. This means that a real-time pollution map could not display the pollution estimation, also using mobile nodes over multi-hop WSNs, since the data will be available only when a certain sensing device reaches the range of a collector node. Moreover, this paper does not address any kind of any measurement to assess the energy efficiency it tries to reach.

In [78], authors present a data reduction method via dynamic subsampling of the measured variable, performing data fusion among several sensors for the same variable, and data scaling depending on the variables range. These improvements are aimed to reduce the power consumption of the device both for the sensing and the transmission of data. They claim to be able to reduce the data gathering to only 4% of the raw data without significant impact on the time variability of the pollution concentration. However, it is not very clear how they describe the noise model of their system. Moreover, they use raw data from a single LSP sensor, without comparing those measurements to other instances of the same sensor.

In [77], authors propose a duty-cycling methodology for an air pollution monitoring system. This methodology explores the reduction of energy consumption by using a machine learning algorithm that allows predicting values during periods of inactivity of the sensor. The results presented by the authors suggest excellent results in terms of energy reduction. However, they do not specify if the test scenarios present high concentration variations and remark that this model requires constant training. Hence, it is not clear if this process requires keeping the sensors active or if the system is resilient.

The present work investigates the impact of a duty-cycling methodology in urban environments using different estimation periods of particulate matter under a resilient scheme. That allows us to maintain accurate readings if a sensor presents a temporary failure (jamming of some material inside the sensing chamber).

2.2.3 Experimental Setup

The following experiment was designed to measure PM_{10} and $PM_{2.5}$ values over an urban scenario to determine the effect of on sensor accuracy of the implementation of a duty-cycle. This section explains the PM sensor selection, the hardware and software setup, and the data acquisition campaign.

Sensor Selection

There are different low-cost LSP sensor models to consider. Sensors with a heater resistor require a fixed orientation, usually a vertical position, to establish the airflow. This restriction increases the complexity for the creation of an enclosure, even more if such sensor is utilized in a mobile device. Another limitation of heater resistors is the stabilization time. This is the time needed to heat up the resistor, establish the airflow, and provide an accurate measurement. This time is close to one or two minutes. Therefore, the duty cycle application in these sensors will be limited to periods of measurement longer than 2 minutes.

On the other hand, fan-based LSP sensors provide faster stabilization times, close to $10s$, and more tolerant with respect to orientation changes. Another interesting feature is the inclusion of digital outputs to deliver the measurements, permitting to use less electronics to adapt the signal to different processing units.

Based on the previous analysis, the *HPMA115S0-XXX*[51] sensor was chosen to deliver the experiment. It is a fan-based LSP and provides lower stabilization times ($< 6\ s$) and a *UART* interface to get the measurement information and control the fan state. This sensor presents high consistency and coherence [18] and can be used as a power benchmark with the prototype shown in [19]. The sensor provides a concentration sensing range of 0 $\mu g/m^3$ to 1,000 $\mu g/m^3$ for PM_{10} and $PM_{2.5}$ values. Its features include a serial communication protocol to deliver both PM values and manage the fan state (on/off) and two measurement modes:

- Auto-send mode: (Default) PM measurement is delivered every second.
- Query-mode: PM measurement is requested through a query message.

Hardware architecture

The hardware setup presented in this chapter section is an evolved version of the prototype presented in the previous section [19]. The main board was replaced by a Pycom FiPy board [79] which embeds an Expressif ESP32 [80] microcontroller unit (MCU) as CPU. This MCU implements different energy modes to reduce the power consumption, and it can achieve 25 μA consumption in ultra-low-power mode. In terms of communications, the ESP32 MCU embeds WiFi (802.11 b/g/n at 2.4 GHz) and Bluetooth (v4.2 and BLE) communications. Additionally, the FiPy board adds Sigfox/LoRA and LTE-M (CAT-M1 and NB-IoT) radio modules. These five communication technologies permit the device a wide range of application capabilities in an IoT context.

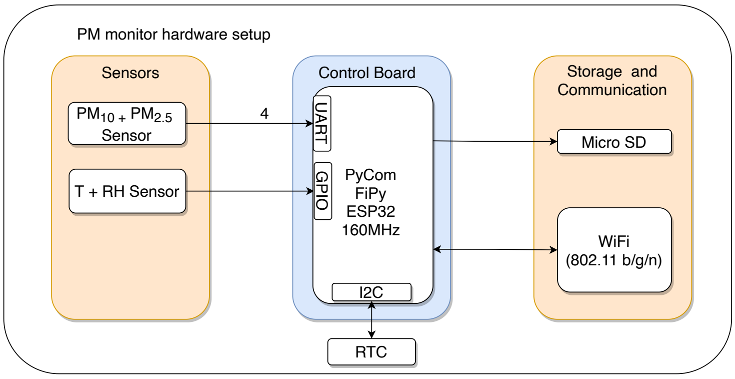


Figure 2.21: Hardware architecture.

The hardware architecture is shown in Fig. 2.21. It has attached the following peripherals to sense and register data relevant to particulate matter:

- Four *HPMA115S0-XXX* PM sensors attached to the Fipy board UART. The FiPy only has 2 UART ports assignable to any free GPIO pins. One of them is used as a programming port, and in this experiment, it is used as a debugging and control port. The second one is assigned to communicate with the PM sensors in a bidirectional mode. Therefore, the UART assignation is controlled via software to manage and measure each Honeywell PM device (see Fig. 2.22).

- One *DTH22* connected via one-wire protocol to sense temperature and relative humidity.

- One external *DS3231* RTC module is connected via the I2C protocol to retrieve the current time. The RTC date-time is updated via NTP protocol at booting if a known WiFi connection is available.

With respect to the prototype and results presented in [19], the atmospheric pressure sensor was not included in the current architecture, since it does not have a significant impact on the PM concentration levels.

In terms of energy, its current consumption is less than 1 mA and could be negligible compared to the CPU and PM sensor consumption. Regarding the data storage task, the Fipy board embeds only 8 MB of flash storage; this is inadequate to store PM values for long measurement periods. Consequently, a micro-SD module was then added via a Pycom expansion board and accessed via SPI protocol. All data was stored in the micro SD in CSV format, and the internal memory of the Fipy was used only to store the device firmware.

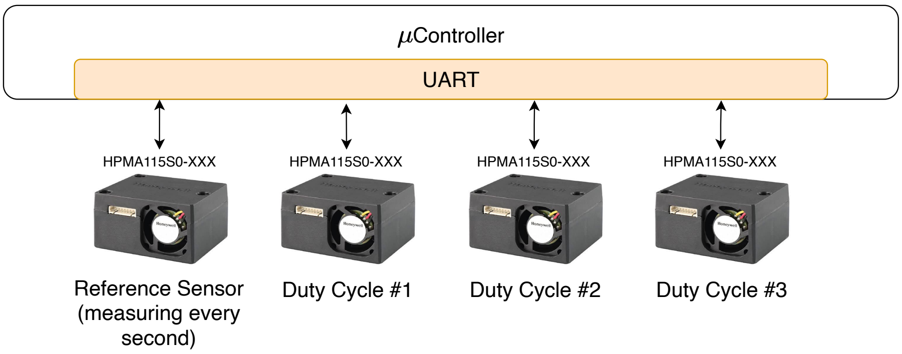


Figure 2.22: Experiment setup.

Software architecture

The Fipy development board implements an open-source MicroPython [81] port which permits direct control over the MCU and its peripherals via Python 3.4 syntax. This port also adds hardware libraries to handle all hardware modules encapsulated in the development board and communications API to enable IoT functionalities such as network control and remote management.

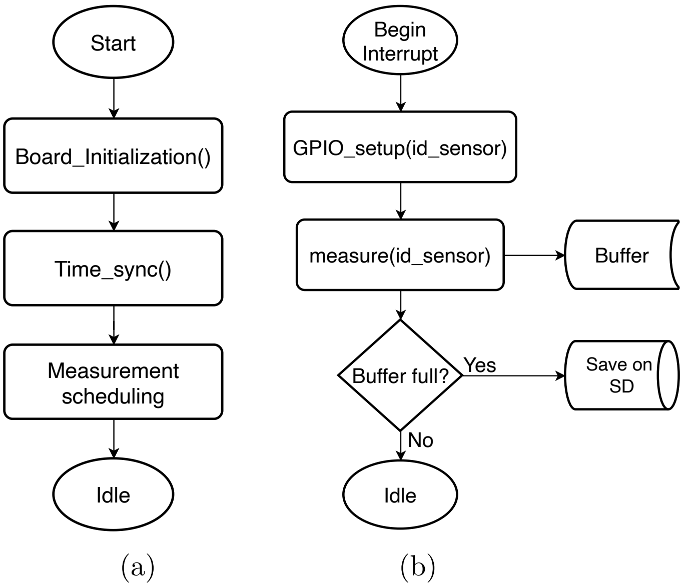


Figure 2.23: Interrupt creation (a) and interrupt handling (b).

The software implementation flow can be resumed by Fig. 2.23a. At the first step, the `board_initialization()` process enables the WiFi connection and RTC peripherals required for data login. At the next step: `time_sync()` sub-process, if the WiFi connection is successful, the internal ESP32 RTC time is synchronized via the NTP protocol and subsequently in the external RTC. If the connection or synchronization is not successful, the time value stored in the external RTC is used.

Finally, the `Measurement_scheduling` process; its purpose is to check the micro SD storage availability and to mount it. Afterwards, it proceeds with the setup of all sensors to their initial state, which in the case of PM sensors, are configured in a low-power state (fan off), and the measurement interruptions are scheduled. It is important to note that the MicroPython implementation used by Pycom queues the interrupts in order of arrival, so the interrupt routines must be short to guarantee the sensing periods.

Each sensor runs its interrupt routine so that different measurement periods can be evaluated for each sensor. The measurement routine can be summarized in Fig. 2.23b. Since the ESP32 has only one UART interface available, this must be assigned to the sensor indicated via software in the `GPIO_setup` process according to the `sensor_id` parameter. In the case of the DTH22 sensor, this sub process is not required since it uses a different communication interface. Subsequently, the measurement process is performed and stored in a memory buffer which is intended to avoid multiple write cycles on the micro SD card to prolong its life. In case the buffer reaches its maximum predefined capacity, the measurements are stored on the micro SD.

Methodology

The experiment consists of two phases. The first one estimates the coherence amongst sensors and defines a set point for evaluating the impact on the accuracy of the sensors due to the different duty-cycle operation. Each sensor operates by taking a measurement every second and with the fan always on. All sensors values are compared to determine the sensor difference and coherence between measurements. In the second phase, the four sensors are deployed in the same environment, taking one sensor as a reference, which will continuously perform the measurements at a rate of one measurement per second. The other three sensors are deployed using different measurement periods to sample every 10 seconds, 30 seconds, and 60 seconds. Those periods are the most significant estimation periods found in the literature. In both phases, all sensors measure PM_{10} and $PM_{2.5}$ values.

	Phase 1				Phase 2							
	Experiment 1				Experiment 2				Experiment 3			
Sensor	t_{st}	t_m	t_{off}	T	t_{st}	t_m	t_{off}	T	t_{st}	t_m	t_{off}	T
#1 (ID: 33, 34)	-	1	0	1	-	1	0	1	-	1	0	1
# 2 (ID: 35, 36)	-	1	0	1	6	3	1	10	-	3	0	10
# 3 (ID: 37,38)	-	1	0	1	6	3	21	30	10	3	27	30
#4 (ID: 39, 40)	-	1	0	1	6	3	51	60	10	3	47	60

Table 2.2: Duty-Cycle times adopted.

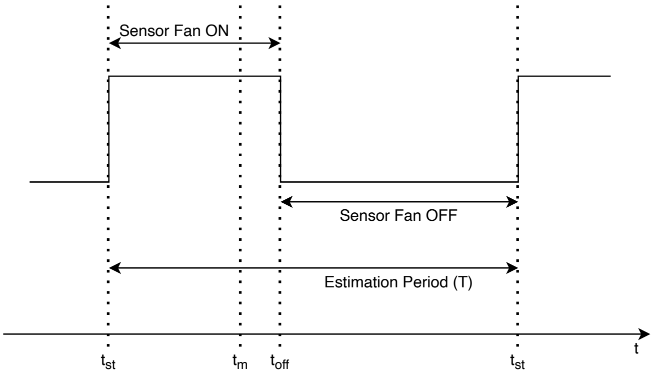


Figure 2.24: Signal period estimation example.

The measurement under duty cycle requires different sensor stages to get a correct measurement over an estimation period (T): the sensor goes into an active mode where the fan is turned on, but no measurement is made until the stabilization time is reached (t_{st}). Once it is reached, it performs the measurement during a measurement period (t_m) and returns an average of these measurements; finally, the sensor goes inactive, in this (t_{off}) the sensor does not perform any measurement, and its fan is turned off. In the inactive (fan off) state, the power consumption is the lowest. The different phases of the PM measurement can be seen graphically in Fig. 2.24. The PM sensor manufacturer specified a stabilization time of 6 seconds. However, in some initial tests, we detect some erroneous measurements. Therefore, a longer settling time of 10 seconds was also used to avoid stabilization transients.

Measurement Location

The measurement campaign was carried out in the city of Turin, Italy, during the winter of 2021. The experiment aims to expose the measurement device to the typical pollutants found in an urban environment. For this purpose, the device was located at the intersection with high traffic flow, where several types of vehicles are moving (small, public transport, and cargo vehicles).

Given the current restrictions due to the *SARS-CoV-2* pandemic, a point close to a large food market was established because it is less susceptible to mobility restrictions decreed by the local authorities.

2.2.4 Results and Discussion

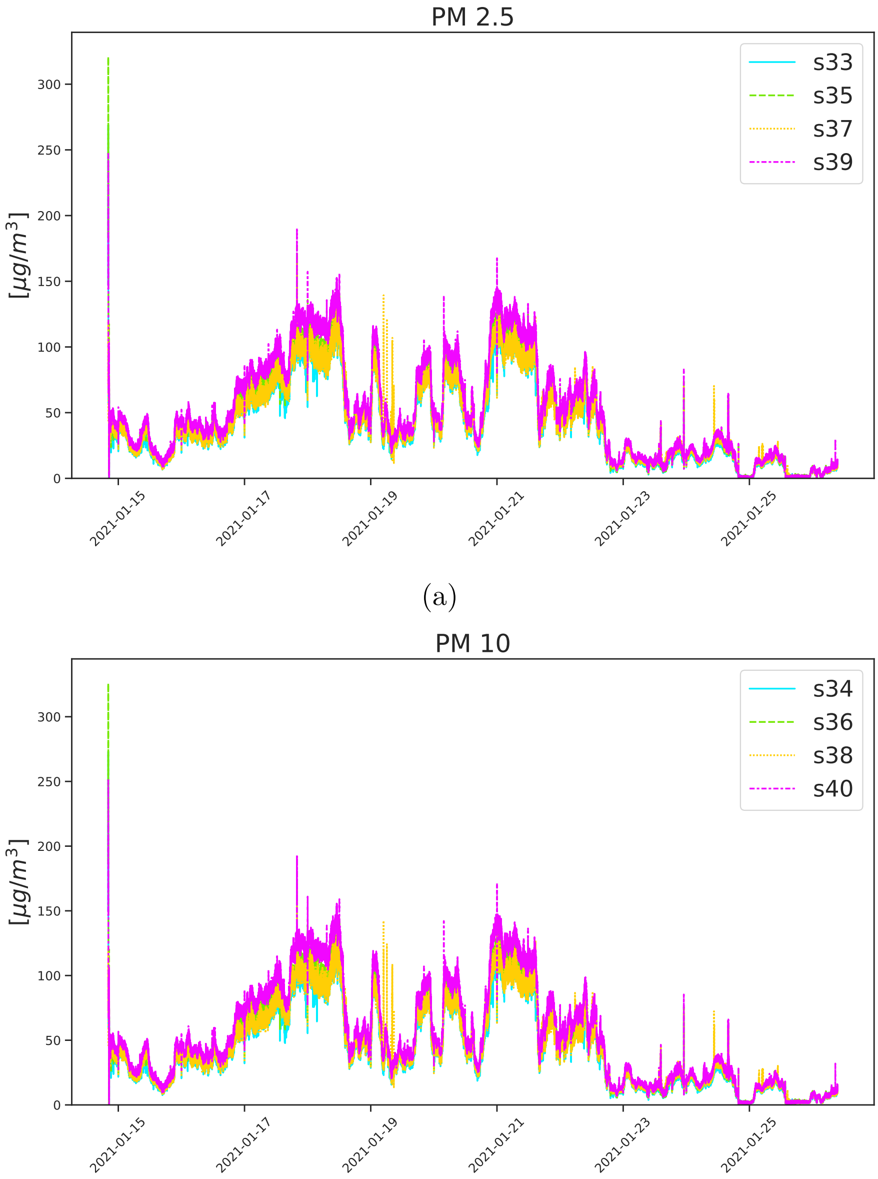
In this section we are going to analyze the data gathered in three experiments over two phases. The first phase is used to establish a reference point for the experiments in the second phase.

Phase 1, Experiment 1

In this Phase we checked that the sensors were behaving in a similar manner among each other.

The data gathered by the four $PM_{2.5}$ (a) and PM_{10} (b) values are shown in Fig. 2.25, after the application of a filter to discard Gaussian noise and to smooth the signal. In this first part of the experiment, the four sensors have been put to work at the maximum sampling frequency (1 sample per second) in order to understand their mutual behavior and correlation. In Fig. 2.26a the correlations (also showing the average) between the sensor chosen as reference and the other ones are reported. The mean correlation factor does not fall below the value of 96%, indicating that these sensors are highly dependable with respect to each other.

For the same time period, Fig. 2.26b plots the Mean Absolute Error (MAE) for the sensors with respect to the reference (also displaying the average). This average error is at maximum $10\mu g/m^3$, which is coherent with that declared by



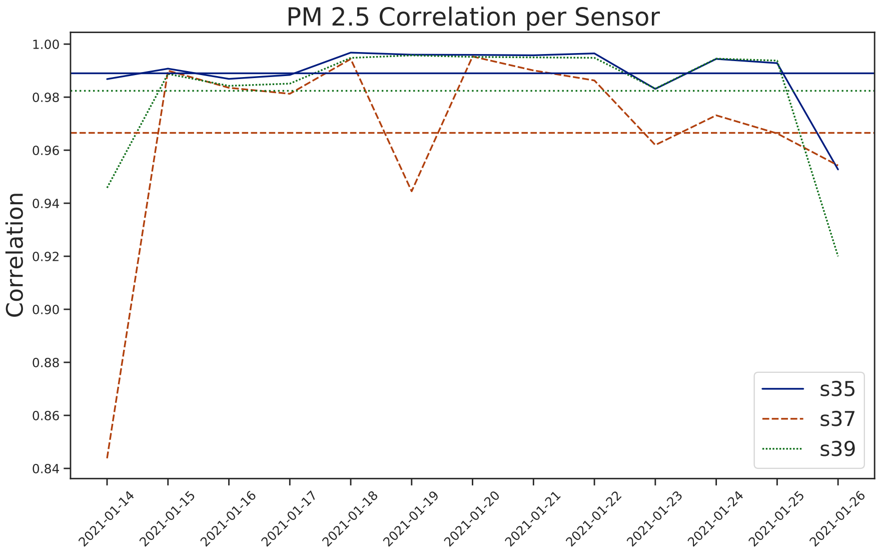
(a)

(b)

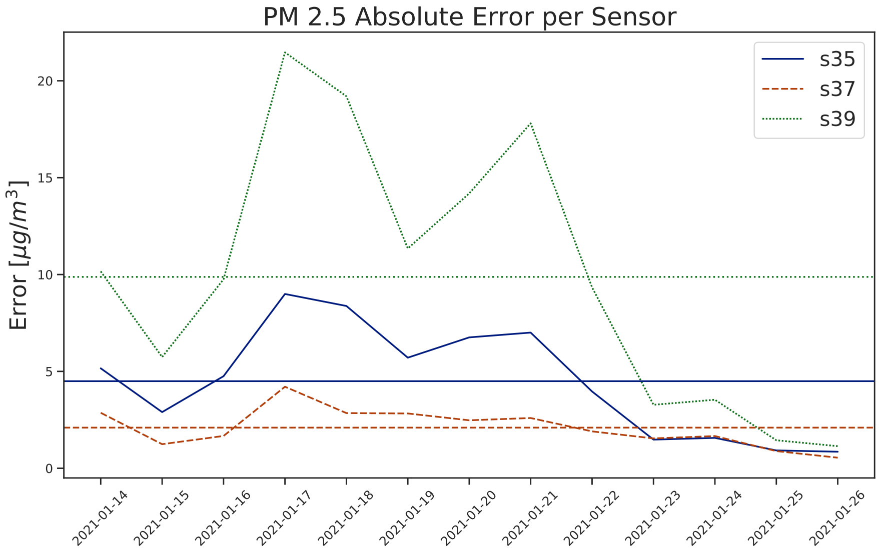
Figure 2.25: Experiment 1: $PM_{2.5}$ (a) and PM_{10} (b) Values at one sample per second and fan always-on.

the manufacturer in the data sheet and also reasonable given the low cost of these devices.

These graphs clearly show that these sensors are well correlated among each other, being able to follow the pattern of PM in the same way throughout the whole experiment. Since the behavior is the same for both pollutants, we decided



(a)



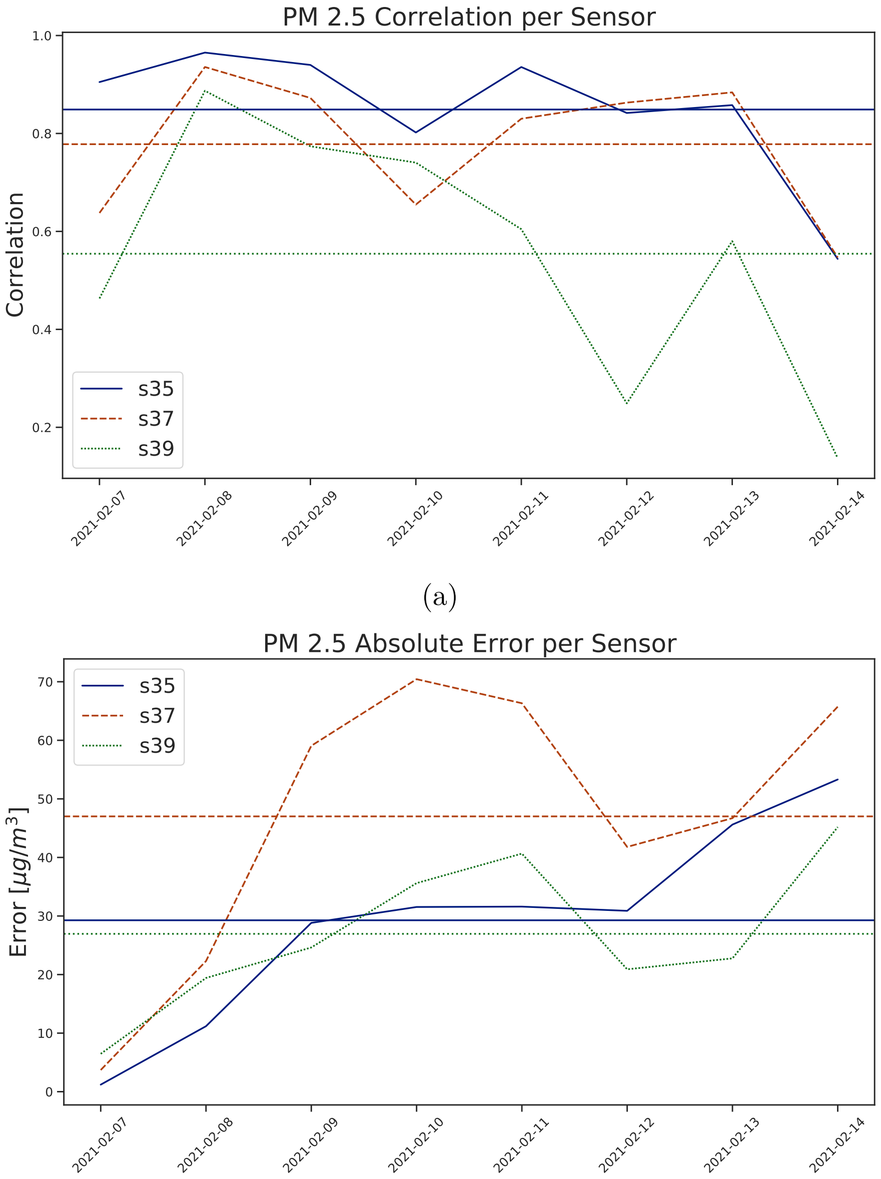
(b)

Figure 2.26: Correlation and MAE between sensors for $PM_{2.5}$ measurement with one sample per second and fan always-on.

to report only the $PM_{2.5}$ plots in the following analysis.

Phase 2, Experiment 2

In this phase, we started changing the duty cycle of the sensors depending on the different configurations displayed in Table 2.2.



(a)

(b)

Figure 2.27: Correlation and MAE Absolute error between sensor for $PM_{2.5}$ measurement ($t_{st} = 6s$).

In *Experiment 2*, the sensors where set up to operate according to several possible duty cycles. In this experiment, the sensors have been powered on and a time of $t_{st} = 6s$ elapsed between the fan-on command and the start of the measurements. Three subsequent measurements have been gathered from the sensors and then averaged. We chose a time of $t_{st} = 6s$ after fan-on since it is the time specified in the data sheet of the sensor after which there is a stable output. Plots in Fig. 2.27a and 2.27b show the correlations and the MAE respectively for such an experiment. As can be deduced by these graphs, the correlations and the MAE are much worse with respect to *Experiment 1.* This means that probably a settling time of 6 seconds is not enough to actually have a reliable output coming from these sensors.

Phase 2, Experiment 3

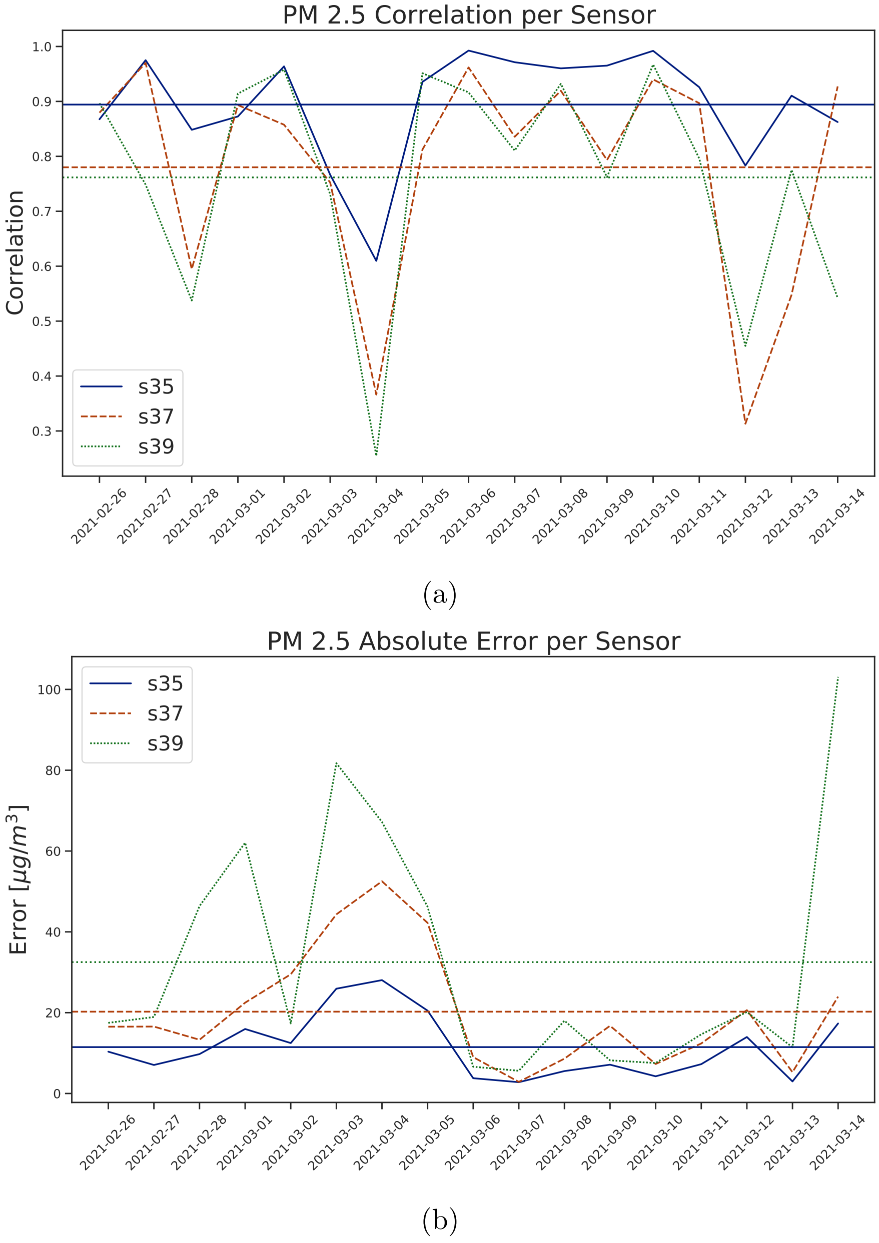
In *Experiment 3*, the duty-cycle of the sensors was changed again, as in Table 2.2. In this experiment, the start-up time of the sensors was enlarged to $t_{st} = 10s$, $4s$ larger than the time specified in the data sheet. Graphs in Fig. 2.28a and 2.28b report the correlations and the MAE (per day and the average) respectively. The improvement by employing such a duty cycle is immediate. The correlations return to high percentages and the MAE is not optimal, but reasonable due to the reduced number of samples gathered. It is also worth noticing that in this last experiment, the minimum average MAE is around $10\mu g/m^3$, while in the previous experiment the minimum was of $27\mu g/m^3$ circa.

Energy Consumption

To evaluate the energy reduction compared to the device presented in [19], the consumption measurements of the device shown in this study were performed using the different duty cycles presented in Table 2.2. In these measurements, all sensors operated with the same estimation parameters to perform the respective comparison. The results are summarized in Table 2.3.

	Watt-Hour	milliAmpere-hour (mAh)
Reference Device [19] 1 sample every second	$\approx 2{,}2Wh$	$\approx 460mAh$
Without Duty-Cycle 1 sample every second	$\approx 1{,}8Wh$	$\approx 380mAh$
Without Duty-Cycle 1 sample every 10 seconds	$\approx 1{,}8Wh$	$\approx 380mAh$
With Duty-Cycle $T = 30s$ $t_{st} = 10s$ $t_m = 3s$	$\approx 1{,}26Wh$	$\approx 256mAh$
With Duty-Cycle $T = 60s$ $t_{st} = 10s$ $t_m = 3s$	$\approx 1{,}06Wh$	$\approx 214mAh$

Table 2.3: Energy consumption benchmark (Hour average)



(a)

(b)

Figure 2.28: Correlation and MAE Absolute error between sensor for $PM_{2.5}$ measurement ($t_{st} = 10s$).

Discussion

Taking into account the plots in Fig. 2.26, we can see that if the sensors are continuously sampling and the fan is always on, the correlation and error are stable for the time of the test. Fig. 2.27 shows that applying the duty cycle we can see a faster degradation of the signal in function of the elapsed time. This is why we increased the stabilization time for the sensors as in Fig.2.28. We can see that in this way we have a degradation rate of the signal with respect to the situation with the sensors with the fan always on. We could say that the time window selected gives some clues about the impact on accuracy drift, but is not enough to draw definitive conclusions on the matter. We are planning longer experiments to address the topic thoroughly, which will be included in future works.

Given the above results, one evident finding is that using a micro-controller, the energy level required to perform the same tasks is reduced by approximately 18%. However, it can be identified that the reduction in the measurement frequency to one sample every 10 seconds, although it does not represent any decrease in energy consumption, does present a low MAE, within the tolerance order specified by the manufacturer. This decrease in the estimation rate will decrease the amount of data generated and, at the same time, reduce the energy consumption required for its transmission.

For the case in which the duty cycle is used under a rate of one sample every 30 seconds, it is identified that, in specific periods, MAE and correlation are low.

Reviewing the data, we find that stable concentration values are presented in these periods, so depending on the dynamics of the site it would be possible to further reduce the energy consumption. Finally, for $T = 60s$, the accuracy is also linked to the variability of the pollution concentration. On days with stable PM concentration (weekend days), the correlation of the sensors is high. This mode could be used as a low-power mode applied in environments or hours with meager human traffic.

2.2.5 Conclusions

Recent years have seen a growing interest in air pollution related topics thanks to the general awareness and the influence of certain public figures. Air pollution monitoring in academia has recently sparkled thanks to the reduction of the cost of Particulate Matter portable sensors. In this domain, there are certain engineering challenges to overcome: the reduction of power consumption, the reduction of data gathering and need of transmission, the aging of such low-cost devices. In this chapter subsection, we address all these issues by analyzing the impact of changing the duty cycle for the operation of Light-Scattering-Particle devices. These measurements carried out sustain our claim that it would be possible to change the duty-cycle of Particulate Matter sensors in such a way that the number of samples to be taken drastically reduces, without losing too much information with respect to a sensor sampling continuously. Naturally, following this sampling reduction comes a reduction of the power consumed by the sensors, and since the sensors are not operating, the moving parts inside them tend to age more slowly, with the advantage of an increased lifetime and less maintenance needed.

Chapter 3

Transmitting data

In this chapter, we are going to tackle the problem of data transmission in the IoT framework. Differently with respect to the previous chapter, in here we focus on another common task in IoT applications, the monitoring of indoor comfort.

Indoor thermal monitoring is a crucial requirement for home automation.

In this context, IoT provides new opportunities for a dense and/or large scale distribution of sensors, which have to gather data to effectively control the Heating, Ventilation and Air Conditioning (HVAC) system.

Several wireless technologies can be exploited for this scope. However, they involve different benefits and drawbacks. In particular, this chapter focuses on Radio Frequency Identification (RFID) and Bluetooth, which are two well-known wireless technological standards suitable for pervasive IoT systems. These technologies are discussed and compared from several points of view. It has been taken into account the easiness of development in terms of time and effort, flexibility, reliability, battery life, and cost of the system. A theoretical analysis highlights their benefits for the application context and evaluates if they are applicable to dense and large scale monitoring systems. The theoretical results are supported by an experimental analysis based on the implementation and test of two different systems, the one using RFID and the other using Bluetooth technology.

Some of the work described in this chapter has been previously published in [82],[83].

3.1 Introduction

IoT-based applications are currently studied and exploited in many sectors, such as health-care [84], [85], autonomous vehicles [86], and environmental monitoring (e.g., air [87], water [88] and fire monitoring [89]). Home automation is the field studying the exploitation of technology for quality of life improvement inside buildings [90], [91].

The smart environment paradigm can be applied to various kinds of environments: private houses, offices, laboratories, factories, etc. Automated systems can control many appliances inside a smart environment, for instance, door opening and locking. One of the most important aims is controlling the Heating, Ventilation and Air Conditioning system (HVAC), which is directly responsible for the wellness of the user inside the building. The aim of this kind of systems is to automatically control temperature ($°C$) and relative humidity ($RH\%$) in order to maximize the comfort for the users while reducing as much as possible the energy consumption.

The American Society of Heating, Refrigerating and Air-Conditioning Engineers defined thermal comfort as “the condition of mind that expresses satisfaction with the thermal environment and is assessed by subjective evaluation” [92].

To automatize the HVAC process, sensors are needed to numerically measure the environmental conditions, while actuators are needed to perform the required actions. A fundamental block of this process is the measurement of temperature and relative humidity parameters in the environment. A related issue is the deployment density of sensors, which depends on the specific environment and on the granularity of the HVAC system. The actual configuration of the environment has to be carefully taken into account when determining the deployment locations of the sensors. For sure, the denser is the deployment, the higher is the accuracy of the measurement, but redundancy could be avoided in such non-mission-critical applications. Wireless sensors can indeed be exploited to effectively build a pervasive distributed monitoring system.

Traditionally WSNs are composed of low-cost devices, featuring: a low-frequency microcontroller, a RAM memory comprised between $2kB$ and $32kB$, a limited capability to no Operating System. The majority of these devices use IEEE 802.15.4 standard for communication [93]. Buildings-oriented WSNs should fulfill a series of characteristics [94]: high correct transmission rate, low power requirements, reconfigurability, and scalability.

Although many papers consider the use of thermal monitoring systems, high-density wireless implementations are still an emerging application and there are no stable guidelines about what should be the best enabling technology. In this chapter we investigate the use of widespread communication systems for commercial electronics applied to the task of thermal monitoring. Radio Frequency IDentification (RFID) and Bluetooth technologies have been selected even if not traditionally strictly applied to WSNs. These technologies are both widespread, cheap, low power, and already used in similar applications.

The RFID technology was originally designed to discern among many instances of the same object in a quick way using wireless communication. As technology scaled, it has been possible to integrate sensors directly in the system hardware. This led to the growing use for various applications in different fields: smart buildings [95] access control [96], supply chain traceability [97], and healthcare [98].

RFID transponders are small, cheap and not power-hungry. Moreover, the presence of RFID tags integrated with sensors makes RFID a proper candidate for dense thermal monitoring tasks.

The Bluetooth® technology aims at providing an inexpensive, low-power, short-range radio-based interface between wireless devices. It is usually used to set up a Personal Area Network (PAN), where a number of slave devices are connected to a master. From the first specifications by the Bluetooth Special Interest Group (SIG) formalized in 1998, this technology went through many improvements. In 2010, the 4.0 protocol version introduced Bluetooth Smart, defining Classic Bluetooth, Bluetooth High Speed and Bluetooth Low Energy (BLE). BLE is a subset of Bluetooth $v4.0$ with an entirely new protocol stack for rapid build-up of simple links. The latest specifications have reached version 5, presented in June 2016. Bluetooth is largely used in IoT applications. It provides a good communication range and requires low-power modules that can be integrated into embedded devices, as required by a dense monitoring system. To evaluate the possible benefits for thermal monitoring applications and to evaluate the applicability to dense and large scale systems, RFID and Bluetooth are carefully analyzed and compared.

The analysis covers many aspects: flexibility, reliability, battery life, and cost of the system. Moreover, two prototype implementations of a thermal monitoring system have been done by using RFID and Bluetooth, respectively. The implementations have been tested to validate the results of the theoretical analysis. Each application location involves specific issues and requirements. Therefore, the purpose of the performed tests is not to provide the exact performance of the considered technologies, but to observe their general behavior in a real scenario and suitability for a specific HVAC application.

The rest of the chapter is organized as follows. In Sect. 3.2, related works are described. Sect. 3.3 presents the theoretical analysis and comparison while in Sect. 3.4 a practical comparison is carried out. Sect. 3.5 presents the system implementation and describes the performed tests. Finally, conclusions are drawn in Sect. 3.6.

3.2 Related Works

Two main approaches exist for the task of thermal monitoring: wired or wireless. The wireless one is often preferred, since it is easier to deploy wireless sensors instead of wiring a particular environment.

Wireless sensor networks (WSNs) are made up by several (generally) low-cost devices, able to collect certain data related to the environment and to send these data via wireless communication. Wireless devices operating in this framework take the name of *sensor nodes*.

Applications of WSN nodes to HVAC monitoring and control systems are already employed nowadays. In [99], an algorithm is proposed to forecast the temperature evolution of an indoor environment exploiting data coming from wireless nodes. An artificial neural network was implemented using low-cost, low resource nodes, based on an on-line learning approach, without the use of a historical database, due to the limitation of the hardware and the system in general. Nevertheless, the presented model is really effective in forecasting the evolution of temperature even after a short period of time in an unknown environment.

A key issue in this domain is related to the power supply mechanism of the nodes. If they are supplied by a battery, this could need replacement or recharging connected to an electric socket. To avoid this hassle, a self-powered wireless node has been proposed in [100], applied to HVAC monitoring task. These kinds of devices are shown able to easily integrate with pre-existent energy saving systems, also supporting diagnostics via dedicated alerts, all this requiring almost no intervention at all from the users.

WSNs are also exploited for more than a single purpose at a time. In [101], the main task is the monitoring of indoor air quality, but the network is not limited to the use of air quality sensors. Instead, pervasive data about temperature are collected at the same time. In this work, the devices used as sensing nodes were the widespread Waspmotes, manufactured by Libelium™.

RFID technology is not commonly applied to the task of thermal monitoring for HVAC systems. The original purpose of this technology was related to object identification. Nevertheless, nowadays several examples exist of RFID tags also embedding temperature sensors, which are mainly used to track the cold chain of perishable goods. In [102] a general-purpose CMOS temperature sensor is proposed and carefully described. In [103] instead, a temperature sensor tag is proposed to be installed directly inside the concrete walls of buildings during construction.

In [104], a hybrid temperature monitoring system has been presented. RFID tags and WSN nodes are combined to sense the temperature in a refrigerator chamber. Used tags are HF semi-passive ones, with short communication range. WSN nodes are able to sample and transmit the data continuously, while the RFID tags are manually read by an operator.

RFID technology has been applied to HVAC automation in [105]. This study examined the impact of human presence detection inside a particular environment. Both stationary and mobile locations of people were detected with fairly high precision. Such self-adaptive system is demonstrated to effectively reduce the energy consumption of a building.

Furthermore, commercially available purely passive tags exist, for instance, the ones manufactured by RFmicron™ [1] or Farsens™ [2]. Since this kind of tags are battery-free, there is no way for them to unmannedly sample the environment in which they reside. The user has to manually query the tag every time a new data has to be collected.

Regarding Bluetooth®, there are not many scientific articles on the use of such devices to sense temperature and relative humidity. There are instead several using Bluetooth® devices in HVAC systems to detect the occupation of rooms, such as [106] and [107]. The former is cheaper, leveraging the users' mobile phone presence and Bluetooth® functionality, while the latter requires the installation of additional hardware for effective triangulation of the signal to determine the position of the devices.

Not strictly related to indoor thermal monitoring, [108] proposes a multi-hop real-time communication protocol leveraging commercial BLE devices forming a mesh network for industrial applications. Such an approach could be implemented where there are strict requirements for what concerns the timing of the thermal monitoring task.

In [109] a forecast system for indoor temperature is presented. This has been achieved using sensors inside and outside an electrical appliance (refrigerator) and outdoor temperature. The proposed solution is demonstrated to be very effective in predicting the room temperature in order to reduce not only the power consumption of the electrical appliance itself, but also the data that can be used for finer tuning, exploiting the relation with the outdoor temperature. This could potentially lead to a decrease in power consumption in entire buildings or even in cities.

A framework for the development of indoor flexible temperature monitoring is presented in [110]. In this work, a mesh network is realized using off-the-shelf components, also implementing an aggressive synchronization scheme to minimize power consumption. The system has been effectively deployed in a real use case for a year-long continuous data acquisition campaign.

3.3 Technological Background

Both RFID and Bluetooth Low Energy solutions are characterized by a low energy consumption and a moderate data rate, usually up to a few hundred *kbps*. These characteristics make them suitable for environmental monitoring. Just like every other wireless communication technology, RFID and Bluetooth® have to deal with possible interference. The denser is the communication medium, the more difficult the communication results. Furthermore, for what concerns RFID

[1] http://rfmicron.com/temp-sensor/

[2] http://www.farsens.com/en/products/battery-free-rfid-sensors/temperature/

technology, the presence of metal in the medium or other radio communications, such as GSM [111], can interfere with their transmission. In this section, a theoretical description of the two technologies will be performed.

3.3.1 RFID

An RFID system is made by one or more *tags* and one or more *receivers.* The reader/receiver queries tags via wireless communication. Tags are thus composed by a logic element, a memory element and a radio frequency antenna. It is possible to distinguish the kinds of tags depending on the different power supply mechanisms: *passive*, *semi-passive* and *active. Passive* tags do not have a battery, they are usually simply used for identification purposes (for example: tracking in the supply chain). They are powered only by the wireless communication of the reader. The electromagnetic radiation gives enough power to the tag to wake up and communicate the required information. *Semi-passive* tags include a battery to supply the sensors, while the communication with the reader is still achieved in a passive way. *Active* tags instead are equipped with a battery and are able to autonomously sample the sensors and ensure longer communication ranges with respect to passive ones. In addition, once properly programmed, they can autonomously initiate data sending.

RFID systems are subject to many interference issues. Water and metal can block the communications. Moreover, other RFID signals or other wireless transmissions operating at a close frequency can affect the communication. In particular, a reader can be prevented from receiving the low-power answers of passive tags. The mutual interference among RFID devices is called collision [112]. A tag-to-tag collision corresponds to two tags simultaneously answering to a reader. A reader-to-tag collision corresponds to a tag simultaneously queried by two readers. A reader-to-reader collision involves a reader that with its high power transmissions prevents another reader from receiving the answer of a tag. RFID tags are generally manufactured in a simple and quite robust shape. Active tags are usually built with non-rechargeable batteries, thus the life expectancy of the device is directly related to its battery. In a network composed of this kind of tags, if one of them ceases functioning the system does not stop, thus this loss is not critical. Instead, the reader ceasing functioning actually stops the system and it has to be substituted. A recap of the various frequencies at which RFID systems work can be found in Table 3.1.

Table 3.1: RFID Frequencies.

	Frequency	**Range**	**Active/Passive**
Low Frequency (LF)	125 - 134 KHz	1-10 cm	Passive
High Frequency (HF)	13.56 MHz	10 cm - 1 m	Passive
Ultra High Frequency (UHF)	433 MHz	1 - 100 m	Active
	866 - 868 MHz in EU, 902 - 928 MHz in USA	1 - 12 m	Passive
Microwave	2.45 - 5.8 GHz 3.1 -10 GHz in USA	1 - 2 m Up to 200 m	Active

3.3.2 Bluetooth®

Bluetooth® is one of the main wireless technology standards operating over short distances. Bluetooth® operates at frequencies between 2,402 and 2,480 MHz, in the globally unlicensed Industrial, Scientific and Medical (ISM) band. Data to be sent is divided into packets and then transmitted on one of 79 1MHz-wide channels. The channel is changed 1,600 times per second according to the adaptive frequency hopping mechanism, used to overcome interference problems. Bluetooth Low Energy uses $2MHz$ channel spacing, for a total of 40 channels.

The heart of the Bluetooth® specification is the Bluetooth® protocol stack. By providing well-defined layers of functionality, the Bluetooth specification ensures the interoperability of Bluetooth devices and encourages the adoption of Bluetooth technology. The layers range from low-level radio link to applications and profiles. Lower layers handle packet management and physical data transmission; the upper layers provide communication APIs to applications. The developed application is directly interfaced with the Generic Access Profile (GAP) APIs. GAP is a top layer in the host protocol stack that defines how BLE devices behave in standby and connecting states to maintain interoperability with peer devices. GAP also describes discovery, link establishment, and security procedures.

Figure 3.1 visually shows the GAP layer in relation to other layers in the software hierarchy.

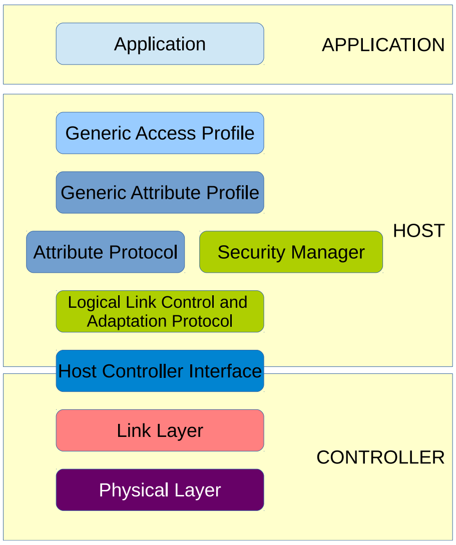


Figure 3.1: The Generic Access Profile (GAP) layer within the BLE protocol stack.

The following GAP roles are defined in the BLE specifications:

Table 3.2: BLE specifications: GAP roles.

GAP Role	Description
BROADCASTER	A device that only sends advertising events
OBSERVER	A device that only receives advertising events
PERIPHERAL	A device that accepts the establishment of an LE physical link using the connection establishment procedure
CENTRAL	A device that supports the Central role initiates the establishment of a physical connection

Bluetooth® devices send advertising packets (PDUs) to broadcast data on one or more channels, and to allow other devices (scanners) to find and connect to them. The advertising data consists up to 31 bytes of user configurable data. An additional 31 bytes can be sent as a scan response to a scan request.

For what concerns batteries powering Bluetooth® devices, there is no such mainstream approach between rechargeable/non-rechargeable or interchangeable/non-interchangeable. Different requirements for a special application could result in a different trade-off on these features. Taking into account the robustness of the overall system, it greatly depends on the network architecture. The single node ceasing functioning could have relative importance to the operation of the overall system. If a star network is chosen, a critical situation happens in case the sink node fails. Instead, if a mesh approach is in use, the failure of gateway or hop nodes could be compensated with a reconfiguration of the network.

3.4 Comparative Analysis

There are a few but critical requirements for the deployment of a thermal monitoring system: transmission range, scalability, power consumption, resilience to interference, and cost of the system. This subsection will focus on the characteristics of the system concerning these requirements, highlighting the pros and cons for the adoption of one of the two systems. In Table 3.3 a qualitative evaluation of the addressed features is presented.

3.4.1 Transmission Range

Typically, active RFID systems operate on a range of $100m$ in open air environments. The network organization strictly follows the star distribution paradigm, with no possibility of network reconfiguration. For what concerns BLE devices instead, the point-to-point transmission range is more than $100m$, depending on the transmission power.

3.4.2 Scalability

The concept of scalability concerns both the number of devices in the system and the area it has to cover. Cover the area with RFID devices would mean to add in the system more readers, with the overhead in terms of hardware cost for the additional readers. In case the packet sent by a tag is read by more than one reader, the software should simply discard redundant information, which means to map every tag to one single reader within its range. For what concerns Bluetooth® , a similar approach can be used. In addition, this kind of devices could be programmed to create a mesh network. In this case, the multi-hop communication removes the need for additional readers.

3.4.3 Interference Resilience

As previously said, the RFID technology is actually subject to several kinds of interference, which are not avoidable due to the definition of the protocol. The collisions that affect RFID are effectively managed by specific protocols. The most common is the tag-to-tag collision. However, there are many protocols that allow identifying in a short time all the tags [113], [114]. Reader-to-tag and reader-to-reader collisions are a problem only within environments with many readers. Moreover, this issue can be managed by existing protocols [115], [116]. However, a degradation of the transmission efficiency is possible, especially in dense applications. BLE instead is built on top of the Bluetooth® protocol, which is self-reconfigurable over a series of channels inside its operative frequency range. Thus, even if it works over a usually congested frequency spectrum (since it is the same frequency also used by IEEE 802.11 standard), it is able to actively avoid collisions.

3.4.4 Costs

A generic RFID reader costs around 400$. Active RFID sensor tags sampling temperature and relative humidity can be purchased for 50$.

For what concerns Bluetooth®, an off-the-shelf or (semi-)custom approach would provide suitable options. Bluetooth tags range around 10$ each, including a microcontroller, the radio interface, and the sensors. The Bluetooth® receiver can be based on a smartphone, a PC, or a single-board computer such as Raspberry Pi.

Table 3.3: RFID and Bluetooth® comparison table.

	RFID	**Bluetooth®**
Range	Medium	High
Scalability	Medium	High
Interference resilience	Medium	High
Cost	Low	Medium
Power consumption	Low	Medium

3.5 Experimental Analysis

This section carries out an experimental analysis of two systems, chosen as a representative for RFID and Bluetooth® technology. The section is divided into three subsections, two characterizing the two different architectures and one describing all experimentally performed tests.

3.5.1 RFID Architecture

This subsection presents all the components of the chosen RFID architecture, which is composed of a *reader*, a *tag*, a *programmer* and also describes the *communication protocol*. For the purposes of this study, it has been chosen to use a complete suite of hardware manufactured by ELA Innovation S.A[3].

Reader

Among the possible readers, the SCIEL READER WF2 was selected for ease of connection. It is an active RFID reader operating at 433 MHz, also featuring

[3] Ela Innovation S.A. https://elainnovation.com/

a WiFi interface IEEE 802.11 b/g/n ($2.4GHz$). It has a dedicated $9 - 48VDC$ power supply. Its rugged enclosure ensures a wide operating temperature range: $-20°C$ to $+60°C$. At the first start, the reader creates its own network as an access point. So as to interact with other devices, the user can then connect the reader to another existing Wifi network or use the one just set up. To communicate with the reader, a series of software tools are available for free download, only for Windows® platforms. The communication is practically implemented as *serial over IP* protocol. The software has a graphical user interface guiding the user to easily set and send the selected settings to the reader. Depending on these, the reader opens a TCP socket port on its IP address and writes accordingly the data coming from the tags. In order to collect these data, a software program has to continuously read and store them for further processing.

Puck RHT tag

The *Puck RHT* are rugged tags specifically designed for environmental monitoring applications[4]. *RHT* stands for *Relative Humidity (RH)* and *Temperature (T)*, specifying the kinds of sensors which are included on the tag. The maximum querying range reported in the datasheet is $150m$ in the open field. The minimum interval between the queries is $200ms$, and the maximum is $10Hrs$. The transmission of T and RH frames happens delayed only by $35ms$ apart, first T and then RH. The life expectancy of the onboard non-replaceable battery strictly depends on the query rate at which it is programmed; with a 3.6 VDC coin battery it may last up to 10 years at the least frequent query rate, meaning the power consumption in this case is really low. Moreover, since the RFID tag has sensors embedded inside the enclosure, the authors reasonably expect the manufacturer designed the measurement in such a way to compensate for the possible noise due to self-heating of the device.

Relative humidity sensor

- Range: 0 to 100% RH;
- Resolution: 0.04% RH;
- Accuracy: ±2% RH max from 20% to 80%, ±5% RH max from 0 to 100%;
- Hysteresis: ±1% RH;

[4] Puck RHT, Wireless relative humidity and temperature sensor. https://elainnovation.com/puck-rht.html

Temperature sensor

- Range: $-40°C$ to $+125°C$;
- Resolution: $0.0625°C$;
- Accuracy: $\pm.4°C$ max from$0°C$ to $60°C$; $\pm1.2°C$ for the remaining range;
- Dimensions: $57x18$ mm;
- Weight $36g$.

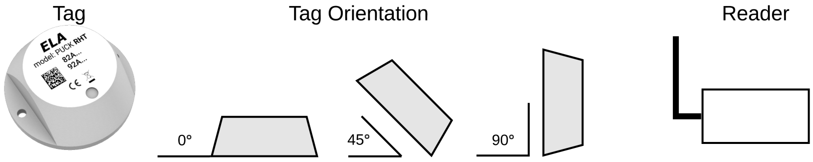


Figure 3.2: Tag Orientation

Programmer

The last component needed for the operation of the system is the tag programmer [5]. It is not only able to program the tags for different query times, but also it can read nearby passive tags up to $15cm$ distant. Furthermore, it supports batch programming for large sets of tags and it has to be connected via USB cable to a PC running a specific program. The programmer is able to read a tag's id and sampled data, program it, activate or deactivate it, and read its software version. It also features the possibility of calibrating the temperature, i.e., inserting an offset to correct its measurements.

Communication Protocol

The query is asynchronous in the sense that the tag autonomously initiates the communication, signalling the reader that it has to transmit the data at a time which depends on the previously set frequency. The reader then receives this kind of packet:

$$[AAxxxxxxLL]$$

where:

[5] SCIEL PROG IR tag programmer. https://elainnovation.com/sciel-prog-ir.html

- [**-**] characters are used as delimiters;
- **AA** is a byte (2 ASCII characters) expressing signal strength;
- **xxxxxx** is the actual payload, 12 bits ID code + 12 bits measured value;
- **LL** is a byte (2 ASCII characters) defining the reader ID.

In this configuration, the whole packet transmitted is 12 bytes long.

3.5.2 Bluetooth® Architecture

In order to test the Bluetooth® Low Energy (BLE) solution, the STM32 NUCLEO low-cost prototyping system from STMicroelectronics was used. The NUCLEO hardware and software ecosystem provides an easy way to develop and evaluate embedded system applications. The STM32 NUCLEO board is equipped with an STM32 microcontroller and Arduino-compatible connectors for external peripherals, and also includes a debugger that can be connected to a PC by means of an USB cable. Then, X-NUCLEO expansion boards can be plugged on top of the base microcontroller board to add functionalities such as sensors (inertial, environmental, etc.), wireless connectivity (e.g., WiFi, BLE and LoRa®) and actuators (motor control, audio amplifier, etc.). Library drivers and code examples are available in order to accelerate and simplify development. The same architecture is used for both the sensor nodes and the data collector, and includes the NUCLEO-L152RE microcontroller board and the X-NUCLEO-IDB05A1 BLE expansion. Each sensor node is equipped also with a DHT22 relative humidity and temperature sensor, while the collector is connected to a host PC by means of a USB cable, but a WiFi expansion board might be employed to provide wireless connectivity (such as X-NUCLEO-IDW04A1).

Microcontroller

For the current experiment, due to the limited computational demand of the application and the necessity of extending the battery lifetime with the minimization of energy consumption, the NUCLEO-L152RE board was selected, which hosts the ultra-low power STM32L152RE microcontroller. Its main features are the following:

- 32-bit architecture (ARM® Cortex® -M3) clocked at up to $32MHz$
- $512KB$ Flash, $64KB$ RAM
- 11 peripheral communication interfaces (USB 2.0, USART, SPIs, I2Cs)
- 11 programmable timers

- 12-bit ADC and DAC
- $1.65V$ to $3.6V$ power supply
- customizable low-power modes, including $195\mu A/MHz$ Run mode, $290nA$ Standby mode ($1.11uA$ with active Real Time Clock - RTC) - at $3.3V$.

The board is operational in the $-40°\ C$ to $85°\ C$ temperature range and power can be supplied either by the host PC through a USB cable or by an external source (cable or battery).

Bluetooth® Low Energy connection

The X-NUCLEO-IDB05A1 board hosts a SPBTLE-RF Bluetooth® Smart 4.1-compliant module integrating a BlueNRG-MS chip and a chip antenna. Its key features are:

- BLE Master and Slave mode supported also simultaneously
- Embedded protocol stack (GAP, GATT, SM, L2CAP, LL, RFPHY)
- Tx power: $+4dBm$
- Rx sensitivity: $-88dBm$
- SPI interface, with programmable interrupt and reset
- Operating supply voltage from 1.7 to $3.6V$
- Current consumption of $1.98uA$ (standby), $0.850mA$ (advertising), 0.105 (connection), $7.72mA$ (host in scan mode) - at $3.3V$.

The operational temperature range is between $-40°\ C$ and $85°\ C$, and power is provided by the microcontroller board.

DHT22

The DHT22 [6] is a low-cost temperature and relative humidity sensor. It is widely used in IoT projects thanks to its ease of use and availability of high-level code libraries. The sensor has to be supplied with a voltage of $3.3\ V - 5.5\ V$; the data pin needs a pull-up resistor of $1\ K\Omega$ to be functioning properly; the highest current consumption is required during the conversion of the data, and it is around $2.5\ mA$, while it draws 40 to $50uA$ in standby; T readings range from -40 to

[6] DHT22 relative humidity and temperature sensor.
https://www.sparkfun.com/datasheets/Sensors/Temperature/DHT22.pdf

$80°C$, with an accuracy of ±0.5° C; RH readings range from 0% to 100% with an accuracy of 2% to 5%; sampling frequency is around 0.5 Hz, so new data are ready every 2 seconds. The DHT22 sensor is connected via a $15cm$ wire to the main board in order to avoid a significant impact of device operational heating on its measurement.

3.5.3 Tests

To evaluate the quality of the systems taken into consideration, a series of tests have been carried out. It has to be specified that *quality* here encompasses both the transmission efficiency and the accuracy in measurements.

Transmission Efficiency

In order to quantify the transmission efficiency, the two systems have gone through a test in which they were put to work in several settings, changing the device orientation and distance from the reader/receiver.

The tags were programmed to transmit at an interval of $1.1s$, and a time interval of 3 minutes was defined. In Figure 3.3 the result of the test is summarized. As the distance from the reader grows higher, the efficiency lowers. Increasing the number of tags used, the number of received packets subsequently increases, but the efficiency gets lower, since a higher number of packets were expected to arrive. The average efficiency for this test is in the order of 60%. It has to be pointed out that the tag orientation with respect to the reader antenna actually plays a role in the data collection percentage. The 45° orientation was the one to attain the highest efficiency, probably due to the reflection of the electromagnetic signals inside a closed environment. In contrast, 90° orientation was the one attaining the lowest efficiency. Notwithstanding the different tag orientations, almost all curves coincide in the right side of the graph, which means the efficiency when 10 tags were used is practically the same in every configuration.

The Bluetooth® system has been put to work in the following configuration: 6 devices in *Server mode* and one device in *Client mode*. *Server* means that this kind of devices periodically (every $1s$) send a broadcast packet, while the *Client* continuously receives and parses all packets sent in broadcast by all other devices communicating in the same technology. This kind of operating mode is known as *advertising* mode; no handshake is performed and no direct communication channel is established between *Server* and *Client*. The BlueNRG-MS device can be programmed both for the broadcast period and the actual broadcast message being sent. In this way, it is possible to periodically change the broadcast packet to include the new data just sampled by the sensors. In order to distinguish the packets coming from NUCLEO boards, the broadcast package is thus composed:

$$< MAC\_address > NUCLEO < temperature >< relative\_humidity >$$

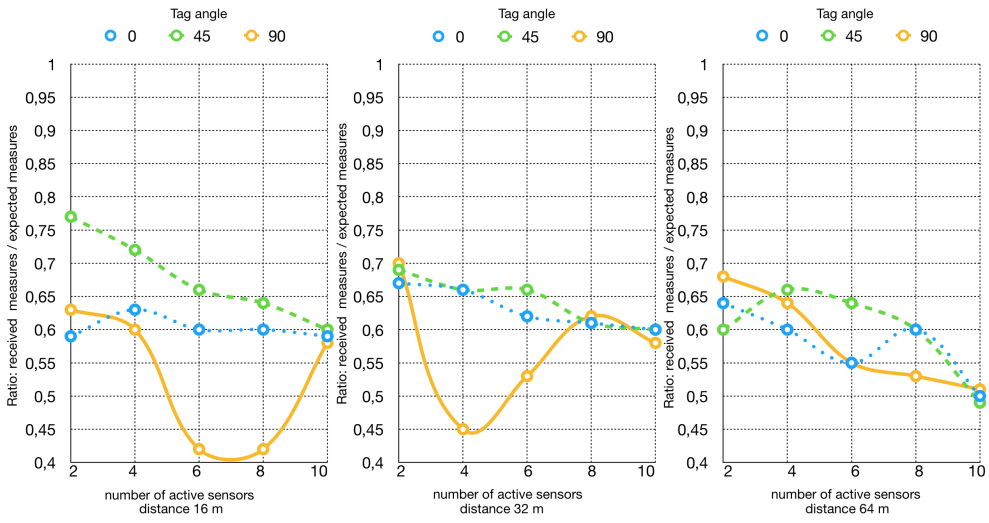


Figure 3.3: RFID reception rate taking into account tag orientation

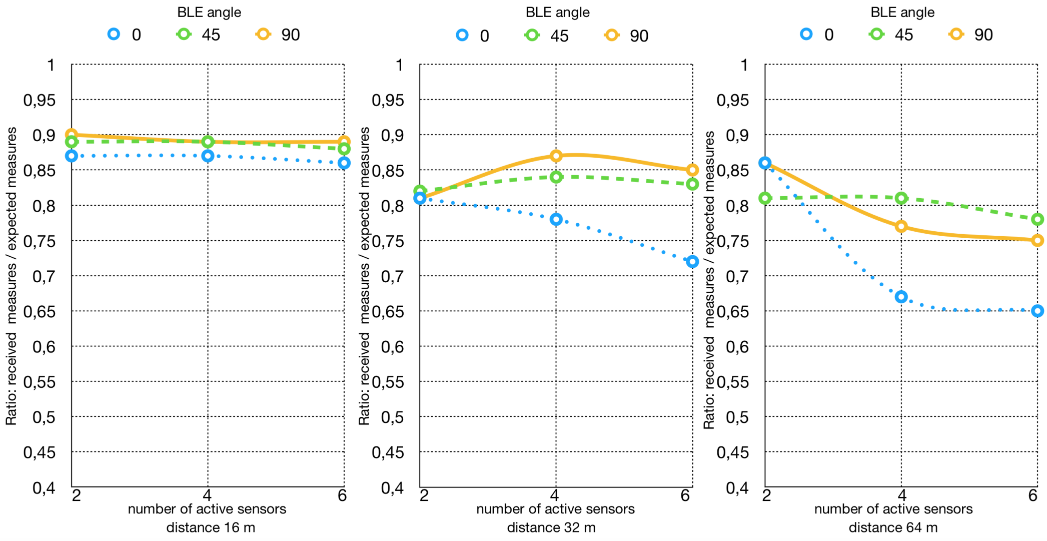


Figure 3.4: Bluetooth® reception rate taking into account tag orientation

In this way, the *Client* can actively distinguish and keep only the packets coming from the *Server*, discarding messages coming from other kinds of devices. The kept packages are forwarded via UART (Universal Asynchronous Receiver Transmitter) communication to a PC in which a C program is listening on the port and writing to a file. Thanks to the possibility of choosing the 1 s interval, there was no need to define a time window for the test, in contrast with the RFID one. Therefore, each *Server* device was programmed to send 200 packets (running for 200 s) and then enter the sleep state. This number has been chosen in order to put the two systems in the most similar configuration, expecting almost the same total number of packets at almost the same frequency.

In Figure 3.4 the result of the test is summarized. At 16 m, the efficiency is almost 90% for all the three possible orientations $(0°, 45°, 90°)$, with no significant variation as the number of devices increases. The efficiency at 32 and 64 m can be considered stable for what concerns the $45°, 90°$ orientation as the number of devices increases. On the other hand, the orientation at 0° is the one achieving the least efficiency, slightly higher than 70% at 32 m and slightly lower than 70% at 64 m. Comparing the graphs for the two systems, the reader can see that the Bluetooth® system performs better in every possible configuration. It is actually able to guarantee a higher packet reception rate at any of the considered transmission ranges, also maintaining a certain stability, in contrast with the behaviour of the RFID system, which is less stable depending on the configuration.

Long-range Test

To study the maximum transmission ranges of the two systems in a real scenario, a long-range test has been performed, which occurred in an open area.

For what concerns the RFID system, 10 sensor tags have been programmed for $1.1s$ query time. As in the previous experiment, the considered time window was of 3 minutes. The tags were placed with the top pointing right towards the reader, while the reader antenna was oriented pointing first to the sky and then in the direction of the tags. Actually, as it can be seen from Figure 3.5, there is not much difference between the two configurations. In one of the configurations, it is possible to observe a small increase from 16 m to 32 m. This behavior is due to the inconstant interference generally present in real locations. Therefore, it does not mean that better transmissions are possible at 32 m, while that within a short range the distance does not affect significantly the transmission efficiency. This conclusion is also supported by the results of the indoor test, which shows small fluctuations in the efficiency level between 16 and 32 m, without an evident pick. Instead, when attempting the communication at $128m$, a dramatic loss of performance can be observed. The configuration with the reader antenna in the horizontal position was only able to attain 15% efficiency, while the one with the antenna in vertical position was actually 0%.

For the Bluetooth® test, the *Server* devices have been set up to for a 1 s interval between consecutive broadcast messages. As to determine the orientation of the tags ensuring the highest reception rate, several tries have been performed (antennas at 90°, 45°, 0°), for both kinds of devices. The resulting configuration was 45° for the *Client* device (facing upwards) and 0° for the *Server* devices. Figure 3.6 plots the results obtained for this test. As it can be inferred starting from the test at $16m$, the BlueNRG equipped devices are able to ensure a high reception rate, approximately 90%, which is 38% more with respect to the RFID case. At

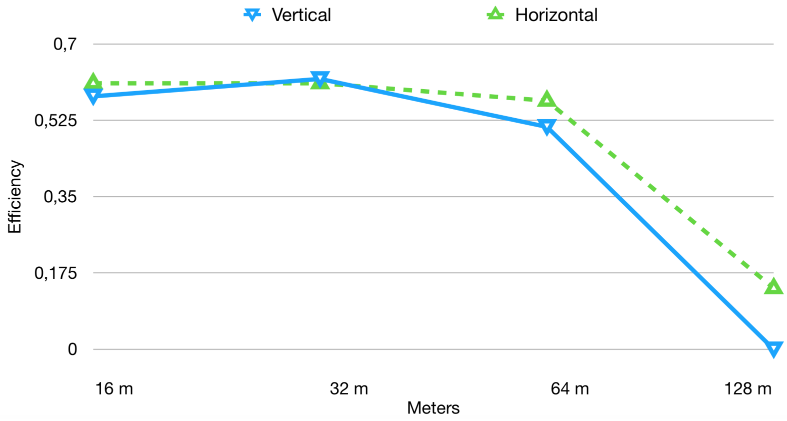


Figure 3.5: RFID data reception rate in long-range tests.

128m the reception rate was above 25%, while for the RFID system was only 17%. It was then decided to continue increasing the range at 32m steps, until a minimum reception rate threshold was no more satisfied. This threshold was set to 5%, as minimum requirement to distinguish between effective and not effective communication. Taking this threshold as a reference, the test can be considered successful up to 226m range, which is an interesting result compared with the 128m reached by the RFID system.

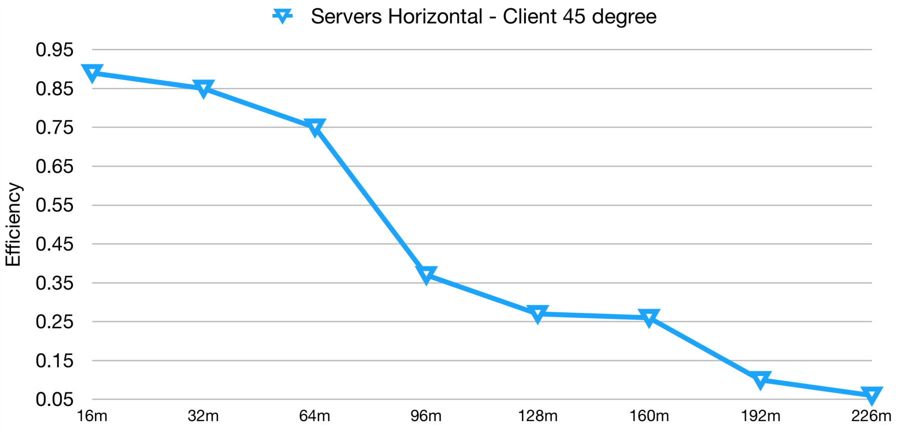


Figure 3.6: Bluetooth® data reception rate in long-range tests.

Accuracy of measurements

A series of tests have been carried out to esteem measurement accuracy for temperature and relative humidity. For what concerns the RFID system, the sensor tags have been compared with the values coming from a DHT22 sensor, which was used as a reference.

The measurement accuracy test has been conducted using a refrigerator. The test lasted for several hours, during which the refrigerator has been powered on and let reach stationary conditions at a certain temperature. It has then been powered off and let heat back to room temperature. In the following subsections, 6 hours of the heating phase are depicted and analyzed, for both systems. In both cases, the sampling interval has been set at 1 *s*, even if given the time constant of temperature and relative humidity a sampling interval of 1 minute is completely appropriate.

In a test of RFID sensor tags, 10 PUCK RHT tags have been used. Both the RFID tags and one DHT22 sensor have been placed inside the cited refrigerator. Plots for temperature and relative humidity can be seen in Figg. 3.7 and 3.8. The plots represent the data after the application of a 5 sample window median filter. It is possible to understand, from both figures, that the sensors embedded on the PUCK RHT tags are coherent with each other and clearly respect the tolerance specified in the datasheet.

To numerically quantify the correlation among the different tags, the correlation factors for every couple of sensors have been produced. Tables containing these values are presented in 3.4 and 3.5. The more the values tend to 1, the more the two sensor tags produce a similar value.

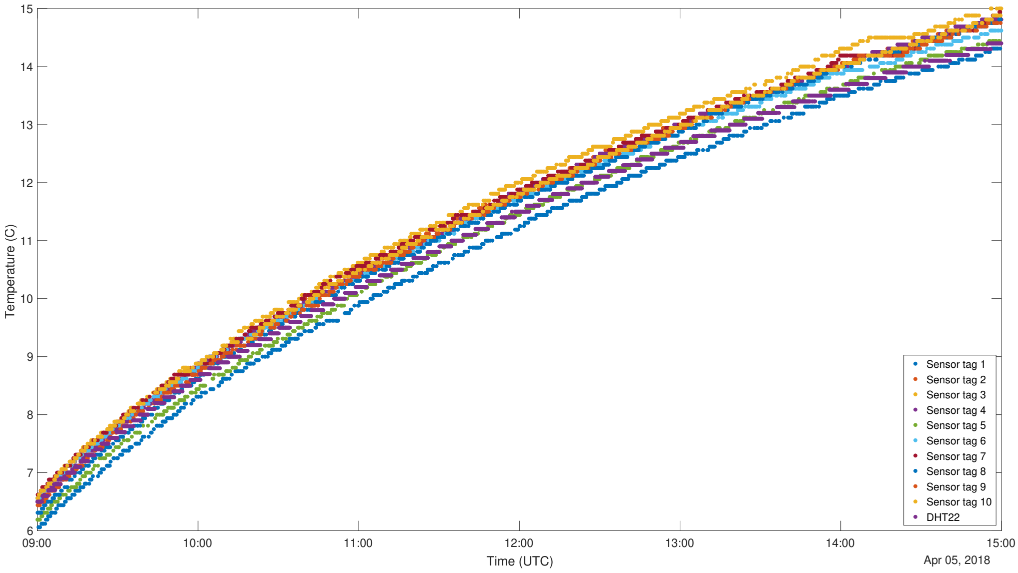


Figure 3.7: RFID Refrigerator test - Temperature

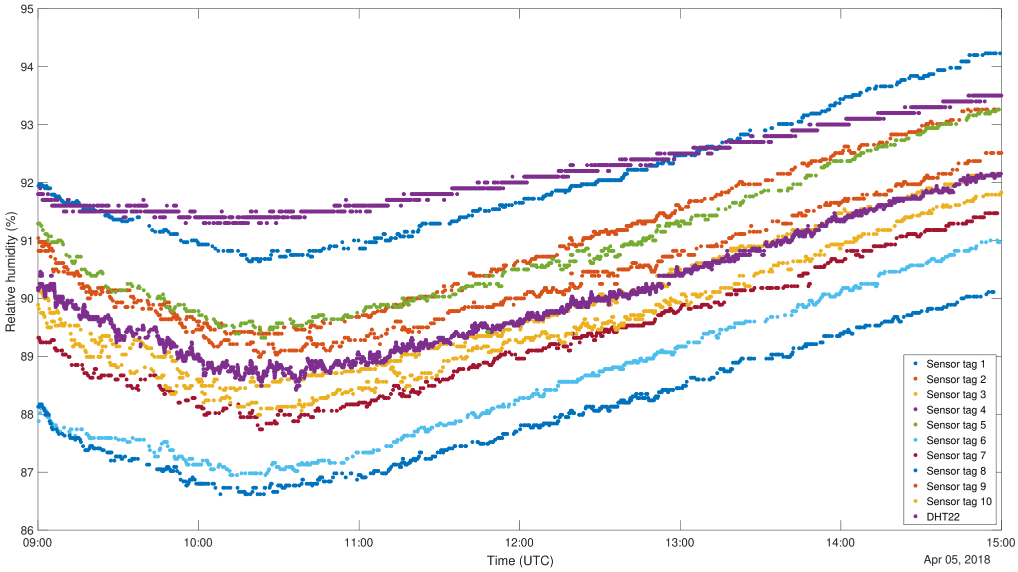


Figure 3.8: RFID Refrigerator test - Relative humidity

Table 3.4: Correlation among different instances of RFID tags - Temperature (C).

	Tag 1	**Tag 2**	**Tag 3**	**Tag 4**	**Tag 5**	**Tag 6**	**Tag 7**	**Tag 8**	**Tag 9**
Tag 2	0.99974								
Tag 3	0.99976	0.99982							
Tag 4	0.99979	0.99979	0.99964						
Tag 5	0.99985	0.99991	0.9998	0.99978					
Tag 6	0.99985	0.99978	0.99955	0.99982	0.99986				
Tag 7	0.99951	0.99969	0.99986	0.99959	0.99972	0.99961			
Tag 8	0.99986	0.9998	0.99982	0.99975	0.99979	0.99985	0.99968		
Tag 9	0.99973	0.99975	0.99985	0.99973	0.99985	0.99983	0.9997	0.99961	
Tag 10	0.99972	0.99976	0.99958	0.99975	0.99972	0.99982	0.99947	0.99988	0.99966

Table 3.5: Correlation among different instances of RFID tags - Relative Humidity (%).

	Tag 1	**Tag 2**	**Tag 3**	**Tag 4**	**Tag 5**	**Tag 6**	**Tag 7**	**Tag 8**	**Tag 9**
Tag 2	0.99448								
Tag 3	0.99853	0.99513							
Tag 4	0.9925	0.98212	0.99273						
Tag 5	0.98851	0.98609	0.99606	0.99162					
Tag 6	0.9957	0.99676	0.99838	0.96458	0.98923				
Tag 7	0.99547	0.99694	0.9971	0.99035	0.99011	0.99689			
Tag 8	0.99583	0.99803	0.99623	0.98767	0.98235	0.98543	0.99258		
Tag 9	0.98334	0.92486	0.99371	0.99169	0.9886	0.98401	0.99578	0.96842	
Tag 10	0.99684	0.98727	0.98511	0.99276	0.98118	0.97046	0.99631	0.99179	0.98741

To carry out this test for the other system, 6 NUCLEO boards, equipped with BLE shield and DHT22 sensor, have been programmed to act as *Server* and placed inside the cited refrigerator. Outside of the refrigerator, another device, programmed as a *Client*, was placed in order to gather all acquired data. Figg. 3.9 and 3.10 show the plots of the gathered data during the 6 hours ascending phase of the refrigerator. It has to be noted though that the two experiments have been carried out in different times of the year, thus the room temperature resulted in being different. In fact, when the RFID test was performed, the air conditioning cooling system was working. This is the reason why the final temperatures for the two tests are not the same. The different setting of the experiment does not invalidate the correctness of its outputs. The graphs for the temperatures are definitely well aligned, and this can also be said for what concerns relative humidity, taking into account the usual uncertainty in measuring this kind of environmental data. Also in this case correlation factors for sensor couples have been produced. These data are organized in distance Tables 3.6 and 3.7. Also in this case correlation values are extremely high, confirming the correspondence among the different instances of the sensor.

Table 3.6: Correlation between different instances of DHT22 - Temperature sensor.

	Sens. 1	**Sens. 2**	**Sens. 3**	**Sens. 4**	**Sens. 5**
Sens. 2	0.99982				
Sens. 3	0.99913	0.99955			
Sens. 4	0.99917	0.99941	0.99953		
Sens. 5	0.99987	0.99984	0.99923	0.99911	
Sens. 6	0.99948	0.99976	0.99983	0.99965	0.99955

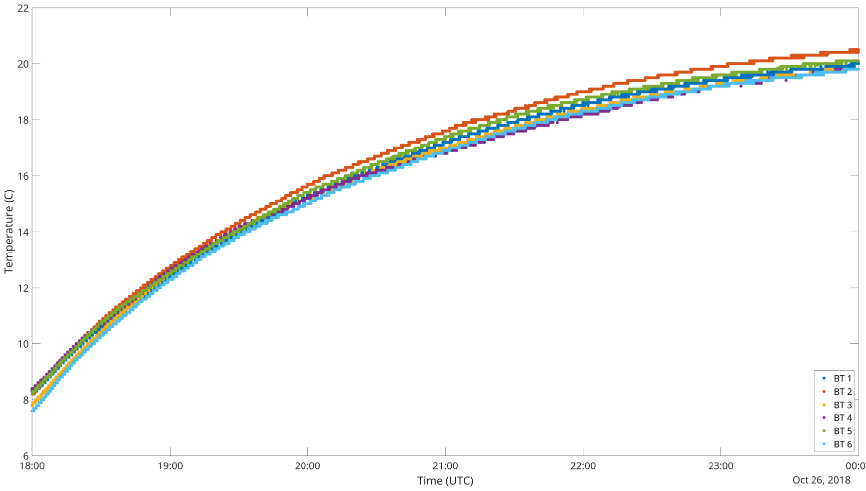


Figure 3.9: DHT22 Refrigerator test, Temperature (C).

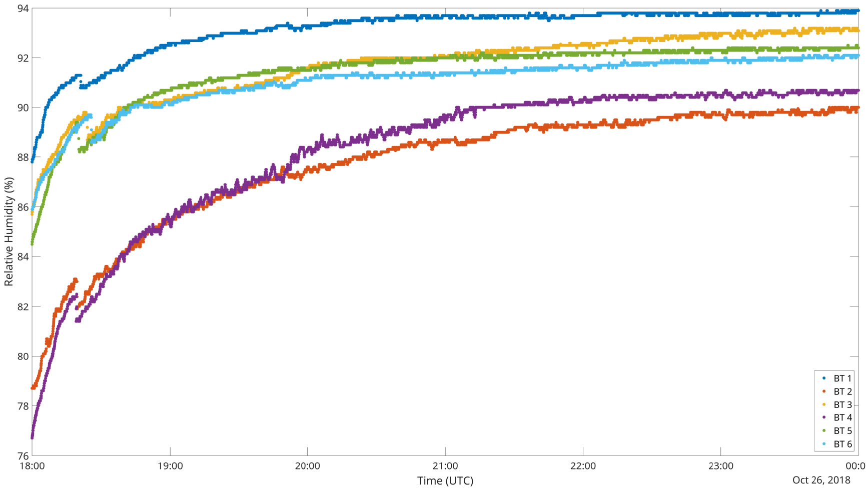


Figure 3.10: DHT22 Refrigerator test, Relative Humidity (%).

Table 3.7: Correlation between different instances of DHT22 - Relative Humidity sensor.

	Sens. 1	**Sens. 2**	**Sens. 3**	**Sens. 4**	**Sens. 5**
Sens. 2	0.97215				
Sens. 3	0.95111	0.9876			
Sens. 4	0.97429	0.99622	0.9858		
Sens. 5	0.99417	0.97092	0.95288	0.97182	
Sens. 6	0.97742	0.98093	0.98119	0.98095	0.98258

Having used the DHT22 sensor as a reference, a comparison between the measurements of the sensor itself and of the sensor tags is due. As previously done between instances of RFID sensor tags and DHT22, correlation factors and correlation graphs have been produced comparing the measurements of the two kinds of devices. These graphs are represented in Figure 3.11. From these graphs it is possible to infer the high correlation between the devices to support the suitability of the use of this kind of RFID sensor tags for indoor thermal monitoring. For both graphs in Figure 3.11, a little shift of values can be appreciated due to the physical construction of the specific couple of sensors. The absolute value of the shift is nevertheless comprised inside the tolerance limits given by the datasheets (maximum ±5% for both DHT22 and RFID sensor tags). To further reduce such shift, a specific calibration would be required, but for this kind of application it would represent an additional cost for each sensor that would eventually be too expensive.

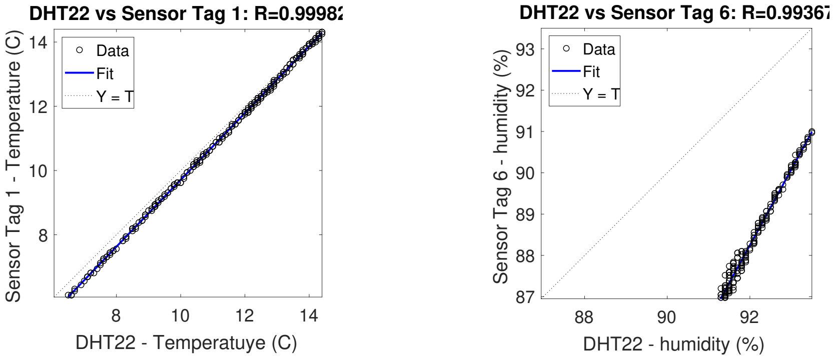


Figure 3.11: Correlation between DHT and Sensor tag sensors - Temperature and Relative Humidity.

Point to point calibration

To further prove the accuracy of the used devices and their applicability for the purpose of this paper, their sampled values have been compared against the ones produced by a professional sensing device. The chosen sensor is the HygroPalm *HP22* by Rotronic®, with a HygroClip2 HC2-S sensing probe. This probe has an accuracy of ±0.8% for relative humidity and ±0.1 K for temperature.

The test was executed in three different runs in three different conditions for what concerns temperature and relative humidity conditions. The graphs in Figure 3.12 show the plots of the three tests. Graphs on the top concern measurements of temperature, while the ones at the bottom concern measurements of relative humidity. Graphs on the left represent data gathered by the RFID sensor tags, while those on the right represent data gathered by DHT22 sensors connected to Bluetooth®modules. For all graphs, the reference measurement gathered by the

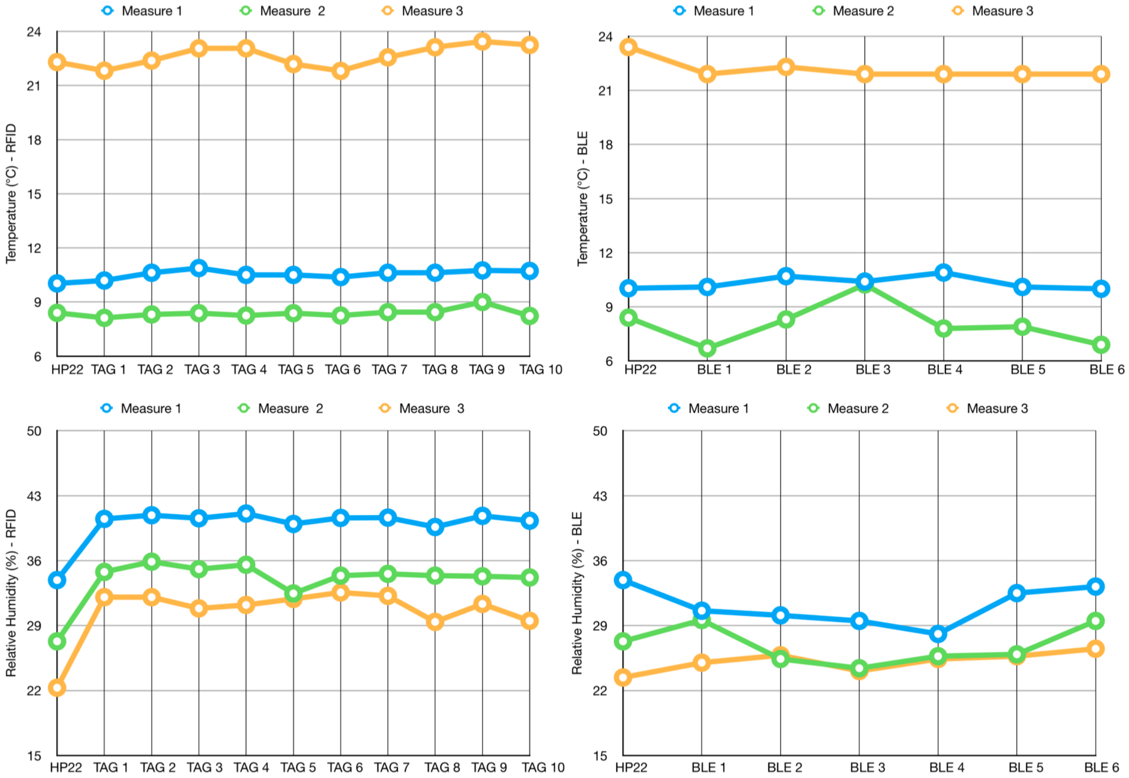


Figure 3.12: Point to point calibration test.

HP22 sensor is the first value on the left. As already pointed out in the previous subsection, the temperature data are well aligned, while the relative humidity ones clearly show an offset. Nevertheless, they are also characterized by a strong correlation.

Power Consumption Estimation

Some reasoning has to be performed for what concerns the power consumption of the NUCLEO BLE devices. Let us logically split the consumption into two parts:

- microcontroller with sensor
- BLE module

Once every minute, the microcontroller reads the sensor and programs the BLE module to broadcast the acquired temperature and humidity data. As it can be seen from Figure 3.13, this process takes around $23ms$. This has been achieved setting a GPIO pin to a high value only during the active phase. The current consumed by the microcrontroller and the sensor in this active phase is around $12mA$. During the off phase, the microcontroller current is negligible, while the DHT22 sensor instead keeps draining a significative amount of current. The average current consumed by these modules in the off-phase is around $45\mu A$.

The BLE module then uses a single advertising message sent on two channels for transmitting the data to the receiver. After this, the whole system enters a low-power mode. Figure 3.14 is an excerpt of the BlueNRG current consumption estimation tool[7]. This figure shows the current consumed only by the BLE shield to transmit a 27-byte packet over 2 channels at the maximum power of $7dB$. This operation lasts for $2.5ms$, during which the device consumes $7.4mA$. During the rest of the time, it is in low power mode, consuming only $1.7\mu A$.

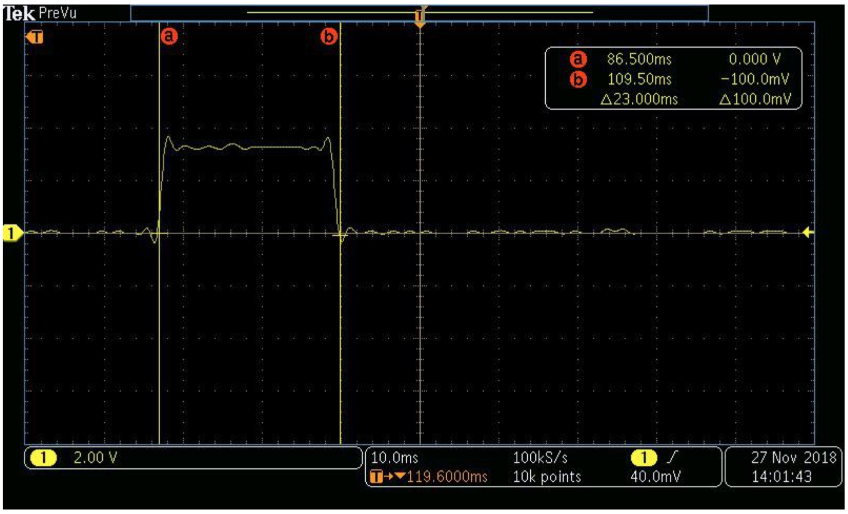


Figure 3.13: Microcontroller active time.

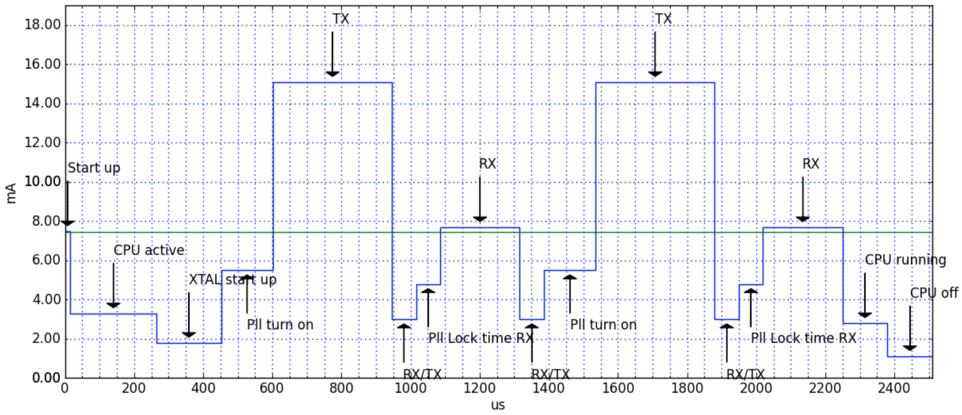


Figure 3.14: BLE current consumption plot.

[7] https://www.st.com/en/embedded-software/stsw-bnrg001.html

In this configuration, a 1000mAh coin battery (such as the CR2477) could last for more than 2 years. It has to be noted that the highest consumption is given by the DHT22 sensor in sleep mode for the majority of the time. Possible solutions to overcome this useless current drain could be: the use of another kind of sensor; inserting a transistor in the design of the custom design to totally cut off the power on the sensor when it is not required.

3.5.4 Case Study

The case study test has been carried out having the two different systems sampling together at the same time in the same location for every couple (one RFID device and one Bluetooth® device). This test took place inside the Department of Control and Computer Engineering at Politecnico di Torino, Italy. Figure 3.15 shows the map of the sensed area with the deployment locations of the couples of sensors. In total, 6 couples of sensors have been deployed: 6 RFID and 6 Bluetooth® devices. Two couples of sensors have been placed inside Laboratory room number 2, while the other couples have been deployed in the surrounding areas. The RFID receiver and the NUCLEO *Client* device have been placed in Laboratory 2, near the deployment location of devices #3. All laboratories in the map do not have direct exposure on the exterior of the building, on one side there is a corridor, while on the other there is a gap between the window glasses which serves as an insulator in case the outside windows are open. During the tests, windows on the outside were open, meaning that the temperature and relative humidity values were actually the open air ones.

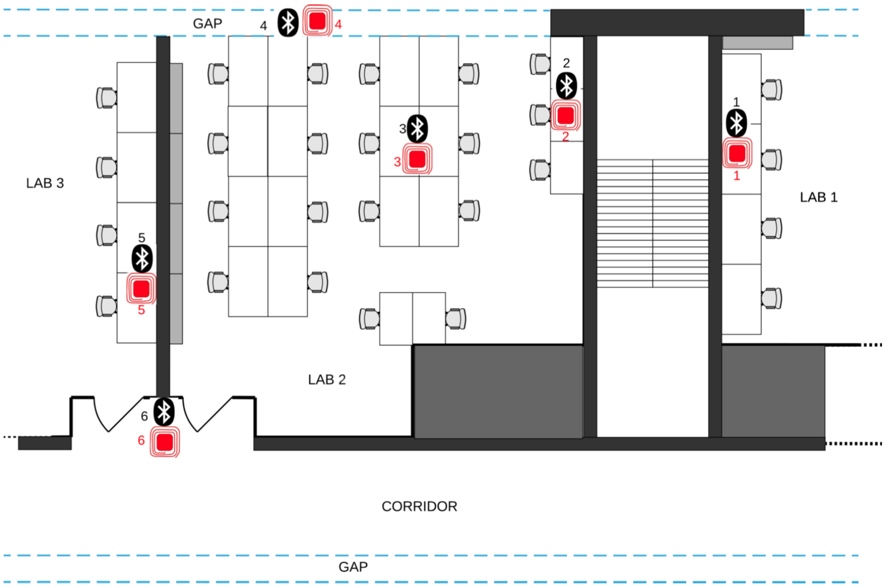
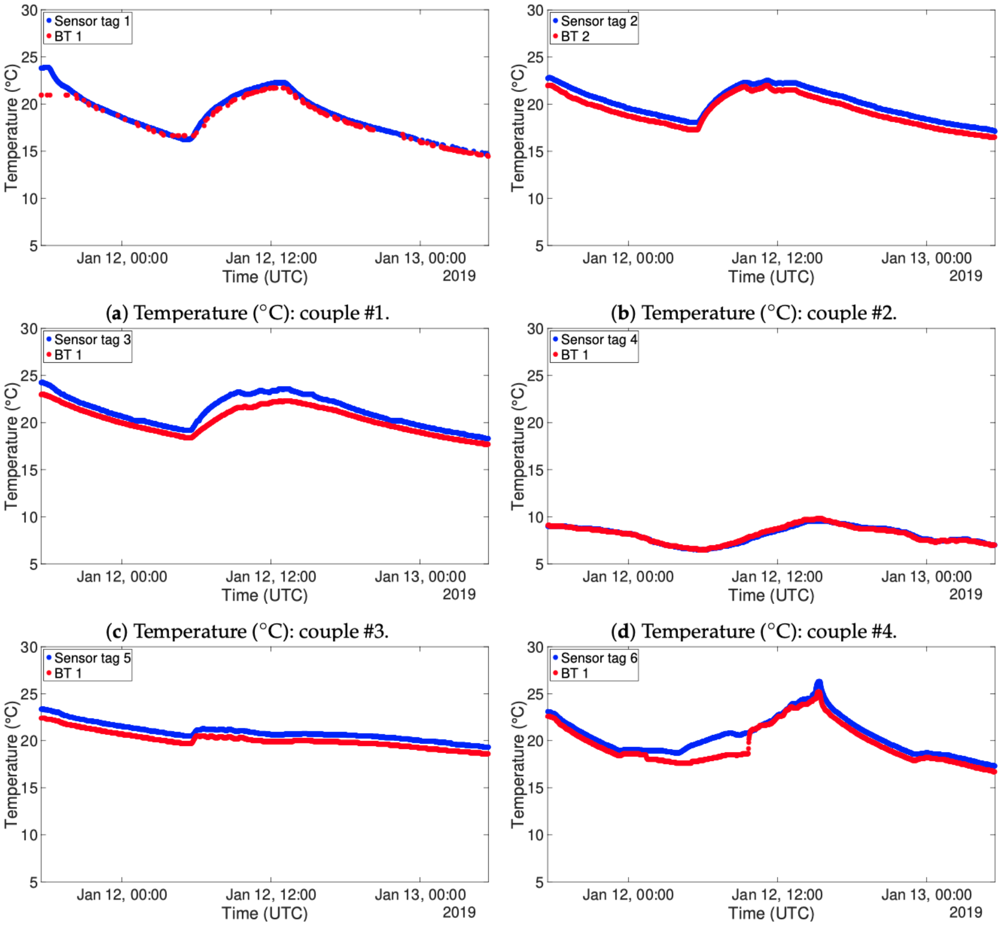


Figure 3.15: Case study plant.

For this test, the devices have been programmed for a sampling interval of one minute, since it is reasonable to expect that possible variations are not instantaneous. The continuous curves show that the system has been able to receive a congruous amount of data, despite walls and interference coming from other sources of electromagnetic signals. The only couple of devices which suffered slightly more from this point of view was the first one, due to the fact was the farthest from the reader devices and the signal had to pass through several walls and metallic stairs. The gathered data are presented in Figure 3.16 for what concerns temperature, and in Figure 3.17 for what concerns relative humidity. Couples of sensing devices have been depicted using the same color, but with a different shape: circles have been used for RFID devices, while squares for Bluetooth® devices. In this way, the comparison between the two measurements is straightforward. The alignment for both temperature and relative humidity is clear from both the figures. The one related to relative humidity actually displays the couples of sensing devices with an offset between the two, but their evolution throughout time is the same. As previously said, this kind of offset could be corrected with a calibration of the devices. Nevertheless, the sensors sampled in accordance with their accuracy ranges. It has to be noted that the couple of devices #6 has been placed in such a place that there was sunlight hitting directly the devices. This fact has a direct impact on the temperature sensing, as it can be inferred by the peak registered on Jan 12th around 16 PM.

3.6 Conclusions

In this chapter, an investigation on wireless technologies applied to the task of indoor comfort monitoring is proposed. This task falls into the IoT/Smart Societies scenario, in which the measurement of indoor temperature and relative humidity are demanded to autonomous devices able to control the HVAC system in such a way to reduce power consumption for the related appliances. Wireless technologies have been chosen due to their ease of deployment and scalability with respect to wired ones. This study focused on two different systems, a commercial one and a semi-custom one. The former exploits RFID technology, while the latter exploiting BLE technology. The two systems have been carefully compared taking into account several key requirements that must be fulfilled to guarantee the correctness and dependability of the overall apparatus. The two technologies have been first described and analyzed from a theoretical point of view, generically considering their operating range, scalability, resilience to interference, and costs of deployment. The two actually chosen systems have then been described and a comparison between them has been performed through the analysis of the outcome of a series of tests. These tests showed that both systems are able to sample the needed data with an uncertainty not invalidating the correctness of the measurement. For what concerns



(a) Temperature (°C): couple #1.

(b) Temperature (°C): couple #2.

(c) Temperature (°C): couple #3.

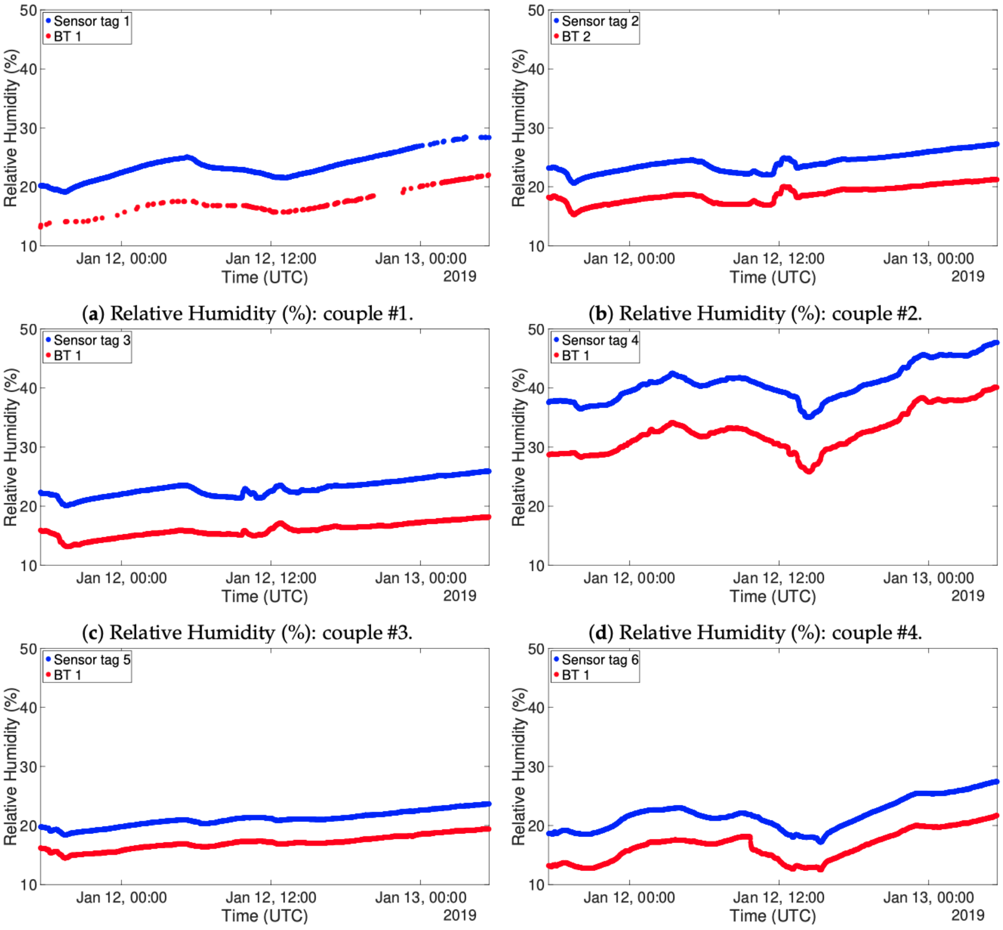
(d) Temperature (°C): couple #4.

(e) Temperature (°C): couple #5.

(f) Temperature (°C): couple #6.

Figure 3.16: Temperature (C) measurements: RFID (circles) and BT (squares).

the transmission range and data reception ratio, the BLE system usually performs better in both the tasks. A final case study has been carried out, with comparable outcomes for both the technologies. The main difference resides in the scalability of the system. The RFID one, once reached the maximum transmission range, requires the deployment of an additional reader. The BLE one instead could be set to form a mesh network, but this configuration has the downside of increased power requirements. Concluding, there is no prominent technology to choose for what concerns the task of thermal monitoring, a trade-off has to be carefully performed case by case. For example, if the final network configuration is not expected to change, probably an RFID system could be preferred, otherwise, in a dynamic environment, the reconfigurability of the BLE comes in handy.



(a) Relative Humidity (%): couple #1.

(b) Relative Humidity (%): couple #2.

(c) Relative Humidity (%): couple #3.

(d) Relative Humidity (%): couple #4.

(e) Relative Humidity (%): couple #5.

(f) Relative Humidity (%): couple #6.

Figure 3.17: Relative Humidity (%) measurements - RFID (circles) and BT (squares).

Chapter 4

Data Security

The next point in the discussion agenda of this thesis is the topic related to the genuine data creation and its safe storage in order to protect it from tampering.

In this domain, we are seeing the rise of Blockchain and other Distributed Ledger Technologies (DLT). Several studies about the possibility of exploiting such technology in different application domains have been conducted. Most of these studies highlighted the benefits that the use of the blockchain could bring in those contexts where integrity and authenticity of the data are a crucial ingredient, for instance, in situations related to regulations about consumers' healthcare. In such cases, it would be important to collect data, coming in real-time through the sensors and then store them in a DLT in such a way that they become immutable. In this chapter, we report the design and development of a software framework allowing the conjunction of an Ethereum-based blockchain with the IoT framework. The proposed solution represents an alternative way for integrating a wide category of IoT devices without relying on a centralized intermediary and third-party services. To better illustrate possible real applications, a case study on food chain traceability in the Industry 4.0 domain is presented.

Unfortunately, the new race for securing private/sensible data sees a new threat appearing on the horizon: Quantum Computing. It is theorized that sufficiently large Quantum Computers would disrupt the cryptographic protocols used nowadays, with catastrophic consequences. This chapter aims to investigate the real threats for blockchain due to the advent of Quantum Computing and review post-quantum DLT solutions for traceability applications.

Some of the work described in this chapter has been previously published in [117] and [118].

4.1 Data Authenticity and Safe Storage

This chapter subsection describes the design and development of an IoT gateway device which is able to directly sign transactions and insert them in a public blockchain. Such a device enables the possibility of identifying the entity actually sampled a certain data and inserted in the blockchain.

4.1.1 Introduction

During the last ten years, the industrial processes commonly applied in any business field, from manufacturing to agriculture and many others, have been subject to a radical evolution led by the arising of new technologies interconnecting the digital and the physical world and making up the so-called Cyber-Physical-Systems (CPSs). CPS consists of a system of collaborating computational elements controlling physical entities, in which mechanical and electronic systems are embedded and networked using software components. They use a shared knowledge and information of processes to cooperate for accomplishing a specific task [119], [120]. The trend of embedding interconnected electronic devices, capable of interacting with the environment in which they live in and enabled to communicate over the Internet, has spread not only in the industrial and manufacturing fields but also in other business sectors such as agriculture, breeding, and food transformation processes [121]. The resulting scenario is known under the context of Internet-of-Things (IoT), which can be defined as a digital overlay of information over the physical world. Each object (referred to as a *"thing"*) connected in such a network is uniquely identifiable and able to sense and react with the environment, as well as with other users or objects [2], [122]. The IoT devices market nowadays represents a large portion of the whole Information-Technology (IT) and electronic devices market with more than \$235 billion spent in 2017, and it is expected to double in 2021, growing up to \$520 billion [123].

Besides IoT, another technology arose and has grown even faster: the Blockchain. A Blockchain (BC) is a distributed-ledger based on a peer-to-peer (P2P) network, in which participants, called *nodes*, agree on a unique version of the distributed data storage through a shared consensus mechanism. All information stored inside the ledger is digitally signed employing cryptographic primitives and data-authenticity is guaranteed using asymmetric key pairs. The first definition of BC has been given in November 2008, in a white-paper titled "Bitcoin: a Peer-to-Peer Electronic Cash System" sent to the subscribers of a mailing list about cryptography by a mysterious author known under the pseudonym of Satoshi Nakamoto [8].

In this chapter section, a possible integration strategy that combines both IoT and BC for improving business processes is presented. The proposed architecture is able to fully exploit the concept of a public Blockchain. In fact, IoT devices are able to directly sign transactions and store them in the BC in such a way that there is no third party involved in the process, which could lead to a series of undesirable situations. The final result of this study is a *blockchain-enabled gateway* for IoT devices to be integrated in Industry 4.0 domain and in many other scenarios in which the combination of both technologies brings advantages to both business processes and customer satisfaction. In particular, the selected use case in which the architecture has been applied and tested is related to the agri-food supply chain, known also as *food-chain*.

The remainder of this chapter is organized as follows. Section 4.1.2 gives a high-level overview of the basic elements of a Blockchain, with a specific focus on Ethereum in section 4.1.2. Section 4.1.3 describes the current landscape of IoT-BC integration, highlighting its main weaknesses. In section 4.1.4, presents the proposed integration strategy, along with the implementation details in Section 4.1.5. Section 4.1.6 depicts a simulation of a use-case application related to the cold-chain monitoring, also describing the main issues related to the agri-food supply chain. Section 4.1.7 displays and comments the results obtained with the proposed architecture. In Section 4.1.8, Conclusions are drawn, also proposing possible future extensions.

4.1.2 Background

The decentralized network proposed by Nakamoto [8] is based on a Peer-To-Peer architecture, where all the nodes have the same functionality and provide the same services without the distinction of roles as it happens in a Client-Server architecture. Each peer in the network, known as a **node**, stores locally a copy of all the history of the **transactions** published on the network. A Blockchain can be formally defined as a structured database of **blocks**, each one containing several transactions, linked together by including in each block the cryptographic hash of the prior block in the chain. This allows us to check the integrity of the whole chain going backward until the genesis block, which is the very first block of the chain. Sometimes separate blocks can be produced concurrently, leading to the creation of a temporary fork of the chain. To solve this problem, any BC defines an algorithm for scoring different versions of the history so that only the one with the highest score is considered valid and selected over the others. The mechanism used for verifying new transactions to be added to the chain is called **mining**, and not only makes the users agree on a unique version of the chain but at the same time allows to create new coins that can be spent in the network, exploiting a **reward mechanism**.

Blockchain principles

As previously said, the fundamental elements of a Blockchain are blocks and transactions. The need for each node of the network to locally store the entire chain implies the usage of an efficient data structure. This data structure has to occupy a small amount of memory while guaranteeing that the data stored in the chain are immutable and tamper-proof. In order to be scalable, in Bitcoin BC, the hash of each block is the hash of its **block-header** (a data structure containing a timestamp, a nonce and the hash of the previous block) and the root hash of a multi-level data structure known as Merkle Tree. A graphical representation of a blockchain block linking mechanism can be seen in Fig. 4.1. A **Merkle-Tree** is a binary tree composed of a set of nodes with a large number of leaf nodes containing the underlying data at the bottom. Starting from the leaves, each intermediate node stores the hash of its two children up to the root node, that is at the top of the tree.

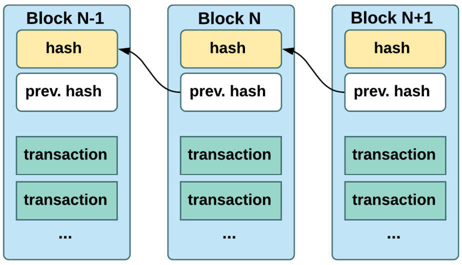


Figure 4.1: Blockchain blocks linking concept.

The hash mechanism behind this structure, combined with the shared consensus, ensures that the entire chain can not be changed or altered. However, nowadays the effectiveness of this protocol is argued not to be sustainable for the long-term. Indeed, as of June 2020, the total space occupation of the Bitcoin Blockchain is roughly 270 GB and it keeps growing quite fast. This factor renders unfeasible the implementation of a node on devices with limited storing and computing capabilities, such as mobile or embedded devices [124].

Digital-signature and Elliptic-Curve Cryptography

Asymmetric key cryptography, also known as **public-key cryptography**, is a cryptographic scheme that makes use of a pair of large numbers (*keys*) that are somehow related together but they are not identical. One of the keys is referred to as *public*, because the owner shares it with other parties. It is used by the message's sender to encrypt the message to be sent to the other party. The other key is referred to as *private*, because it has to be known only by the owner and not shared with any other parties. It is used for decrypting a message received over an insecure channel.

This kind of mechanism works under three fundamental assumptions:

- Private and public keys should be related, but at the same time it should be unfeasible to obtain the private key knowing only the public one,
- The decryption function gives the correct result if and only if the correct private key (i.e. the one related with the public key used for the encryption) is given and for no other different value,
- Private keys should be kept secret, never shared with anybody else.

Asymmetric cryptography protocol implementations differ mostly in the algorithm used for the key pairs generation and the encryption/decryption functions. For example, the RSA algorithm is based on the *large integer factorization problem* and, simply speaking, uses the product of two large prime numbers as part of the public key, and derives the private key from the same number as well. The encryption strength relies mostly on the size of the key, with its robustness increasing exponentially with such a parameter. Actually, RSA [125] keys can be typically 1024 or 2048 bits long, but experts believe that RSA 1024 is near to be cracked and it should not be considered secure anymore, especially with the advent of quantum computing [126]. We will dig deeper on this topic in the next chapter.

Another implementation of the public-key cryptography protocol relies on the algebraic structure of *elliptic curves* over finite fields and is known as Elliptic Curve Cryptography (ECC). It is based on the *elliptic curve discrete logarithm problem* that consists in finding the discrete logarithm of a random elliptic curve element concerning a publicly known base point. The approach of using elliptic curves in cryptography was proposed for the first time in 1986 by Victor Miller [127] and Neal Kolbiz [128], but it became widely used starting from 2004. It is used in Blockchain implementations such as Bitcoin and Ethereum as a Digital Signature scheme [129], [130]. The security of ECC stands in the fact that, roughly speaking, given a point $k * G$ on the curve, it is difficult to go back to the original value of k. This is known as the **Discrete Logarithm Problem**, which is said to be one of the NP-hard problems. Compared to RSA, ECDSA offers the same level of security, but with smaller keys.

Ethereum

Ethereum is an Open-Source project aiming at the creation of a public BC for distributed computing. It was born in 2013 thanks to Vitalik Buterin, a Canadian developer and researcher in the field of cryptocurrencies. Through a crowdfunding campaign, he was able, in 2014, to implement his idea, published and available online the following year [131].

The project intended to create an alternative protocol for building decentralized applications (DApp), allowing a different set of trade-offs to be used in those applications where rapid development time, security and the interaction between different distributed applications or systems is crucial. Following this idea, Ethereum provides a fundamental layer composed of a Blockchain and a Turing-complete programming language so that anyone can write its own DApp, implementing its arbitrary rules for ownership, transaction formats, and state-transition functions. The most interesting and explored functionality of this new architecture are the so-called *Smart-Contracts*. *Smart-Contracts* can be seen as cryptographic entities containing a value, that can be represented by a certain amount of currency or information or both, unlocked only if and only if an application-defined set of conditions are met.

To achieve the goal of creating a simple framework for building powerful decentralized applications, the basic philosophy pursued in Ethereum development is based on a few principles:

- **Simplicity**: the protocol has to be as simple as possible, even at the cost of some inefficiency in terms of execution time or data storage. The idea is that an average programmer should be able to follow and implement the entire specification.
- **Universality**: Ethereum should not provide features. Instead, it has to provide a complete fundamental layer upon which any kind of application can be easily built.
- **Modularity**: the framework has to be structured in modules as separate as possible.
- **Agility**: the protocol's architecture should not be extremely fixed, and any meaningful further proposal bringing significant benefits or improving scalability and sustainability is accepted.
- **Non-discrimination** and **non-censorship**: the protocol should not attempt to actively restrict or prevent specific categories of usage.

Ethereum's fundamental currency is *Ether* (*ETH*) and is used to pay for *gas*, which represents the unit of computation used in a transaction and other state transitions, such as Smart-Contract execution.

Smart-Contracts

The term Smart-Contract (SC) was introduced for the first time in the 1990s by Nick Szabo, a computer scientist and cryptographer, who realized that the Distributed Ledger Technology can be exploited for securing relationships over a network [132]. Bitcoin was the first Blockchain implementation to support a basic form of Smart-Contract through its scripting language, which was however limited. The Ethereum project, on the other hand, was born with Smart-Contracts in mind, aiming to realize a framework for DApps running over a BC ledger in a peer-to-peer network. Just as the term suggests, Smart-Contracts are somehow related to “normal" contracts, with the difference that the latter ones are signed by partaking entities and enforced by the law, whereas the former ones are digitally signed and define relationships among parties exploiting cryptographic mechanisms. Basically, a Smart-Contract is a self-executing application that runs on top of the BC and is therefore immutable. It can be seen as a complex *if-then* statement that is executed if and only if a set of conditions is met. Smart-Contract can either produce an output or even trigger the execution of another Smart-Contract for performing a sophisticated task. Because of their decentralized nature and the possibility to exploit BC’s features for handling trust-less contexts, SCs are becoming an attractive topic for various kinds of business fields spreading from finance, insurance, manufacturing, and many others.

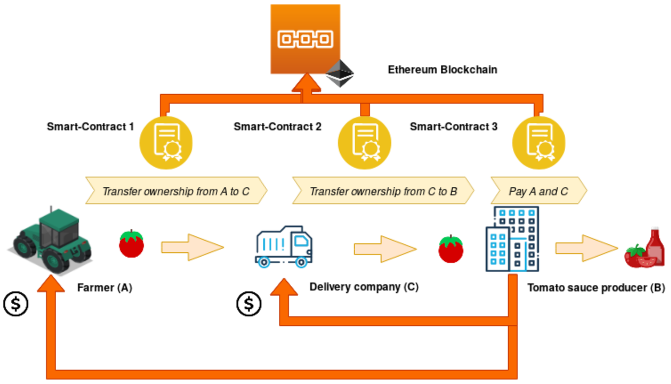


Figure 4.2: Application of Smart Contracts to traceability in the food-chain domain (Image created using https://draw.io).

To better understand how SCs work and how big their potential can be if exploited in some application domain, it is worth describing an example use case (Fig. 4.2). Let us assume that the farmer A sells tomatoes to company B, which then uses them for producing tomato sauce. The transfer of tomatoes from A’s farm to B’s warehouse is in charge of a third delivery company C. The actual most common scenario would be the one in which A, B, and C have their own digital business layer, generally made of a centralized data storage (in a local database or using some cloud services) and their own software products that deal with such data structures for registering shipping, invoices, inventory and so on. Moreover, this software may differ between one company from another, making it difficult to automatize their cooperation. Let us assume now that the same business scenario was managed through the use of Smart-Contracts and a Blockchain in a Peer-To-Peer network in which all of the three companies participate. A smart DApp developer would implement three Smart-Contracts able to interact together:

- A first Smart-Contract that can handle the transfer of ownership of the batch of tomatoes from A to C, ensuring that the latter would be responsible for any loss or damage during shipping.
- Another Smart-Contract that can handle the transfer of ownership from the delivery company to recipient B.
- A third Smart-Contract, triggered by the previous one, that handles payments between the parties in something similar to the following statement: *If tomatoes are delivered without any damage a certain amount of funds will be transferred to both the selling and shipping companies. If tomatoes are damaged then company C has to refund the selling company, and so the proper amount of crypto-money will be transferred from C's account to A's one.*

Thanks to the BC, all operations occurred between the parties will be registered in the ledger forever, and so it is straightforward to think of more complex mechanisms to be built upon the simple scenario described above (insurance for the delivery process, supply-chain traceability, etc.).

Quadrans: the Industrial Blockchain

The Quadrans [133] platform is an open-source and public Blockchain network that runs Smart-Contracts and decentralized applications. The first implementation of this platform, born in 2012 as a fork of Ethereum, has been conducted under the Foodchain S.p.A. brand, and its initial goal was to enable traceability, transparency, and authenticity of information within agri-food supply chains. In August 2018, Quadrans Foundation was officially formed and acknowledged by Swiss local authorities. The goal of the foundation is to guarantee continuous advancement in technology in an ethical and publicly open way. Quadrans' infrastructure is public, and its open-source blockchain has been designed to overcome scalability issues and to reduce the instability of operational costs affecting other existing blockchains. The current protocol, used by Quadrans, ensures a high throughput (average block-time is about 5 seconds) that can maintain at the same time both high security and decentralization grades and supports all Ethereum's core features, such as EVM (Ethereum Virtual Machine) and Smart-Contracts. Having reached these goals, Quadrans represents a valuable solution for effectively applying blockchain technology in many business scenarios.

4.1.3 Literature review

Having given enough details on the basic principles of BC's technology, it is possible to introduce the real goal of this chapter. As stated in the introduction, the project aims to find a new method of interaction between the BC infrastructure and the world of internet-connected devices.

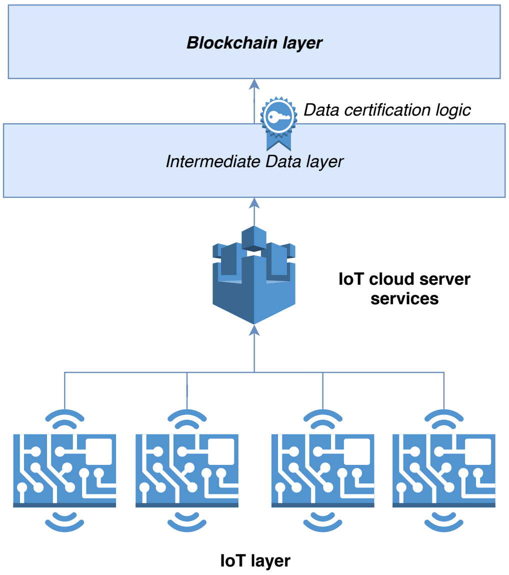


Figure 4.3: The common architecture used for storing data collected from IoT devices into the blockchain (Image created using https://draw.io).

The most common solution applied for this scenario consists of a Client-Server communication (as in Fig.4.3) between a central server and the IoT devices. The server is in charge of (a) gathering the data streams collected from the field and (b) storing these data, or part of them, in the Blockchain. Even if this solution is starting to be applied in different proof-of-concept and business applications, this has some point of weakness:

- The central server may become a single-point-of-failure that can inhibit some data to be stored in the BC when it is in a fault condition,
- Assuming that redundancy is applied, in order to minimize the probability of failure, there is still the presence of a type of centralization that makes difficult the interaction between cooperative business parties,
- The data which is then sent to the BC is not signed-in-place, a way in which its authenticity and integrity can be guaranteed to start from the source, but it is signed only when it is received by the central server before posting it into the Blockchain,

- Nevertheless, such kinds of systems are now mostly realized using cloud solutions in the IoT field. Experts believe that the cost of such services is going to increase rapidly and to become a significant component of business costs for most companies.

The surveys in [124] and [134] present extensive studies of these and other issues. The authors of [134] describe the issue of storage size in relation to an application in the IoT field. The solution we present solves this issue by implementing light-nodes that act as a gateway. These light-nodes only perform the tasks of signing and sending their own transactions. They do not participate in the consensus mechanism (there is no mining), and thus they do not have to locally store the whole blockchain.

A relevant paper on the quest for decentralization, secure data storage and IoT compatibility can be found in [135]. This paper describes a model to handle data-streams coming from IoT devices with a decentralized access mechanism control. This solution has some undesired aspects though. It uses an intermediate level between the cloud and the BC, meaning that the data are not set directly to the BC by the device. Moreover, it uses the Bitcoin BC, even if in its VirtualChain version, supposedly lighter and faster.

The solution described in [136] seems more focused on performing a type of action based on the conditions of Smart-Contracts on the Ethereum Blockchain. It has interesting propositions regarding the use of smartphones to set/update conditions for Smart-Contracts, and the possibility of IoT devices to act according to these conditions. Also, some pieces of Smart-Contracts are shown in the paper. Despite these characteristics, the whole architecture is not clearly described, especially for what concerns the communication with Ethereum blockchain, where the blockchain itself resides, and the signing functionality of the IoT device.

In [137], BPIIoT is presented. The authors refer to it as a key enabler for Cloud-Based Manufacturing, enabling peers of a peer-to-peer decentralized network to interact among them without a trusted intermediary. They also implement Smart-Contract handling on a BeagleBone single-board computer (SBC) which seems to be connected to an Arduino board for sensing capabilities. However, also this paper does not give many details about the implementation of such a system, especially regarding the connection with the Ethereum blockchain and the signing capabilities.

Authors in [138] present AgriBlockIoT, a decentralized traceability system for the Agri-Food supply chain management. AgriBlockIoT is nicely organized in layers and can support either Ethereum or Hyperledger Sawtooth. The paper also presents an interesting *from-farm-to-fork* use case, depicting all actors, materials, and tasks involved in such a process. Our approach can be considered an evolution of this, since we remove the need of running a full node of the blockchain in the stakeholder's local network. Moreover, the use of IoT devices is only simulated and not practically implemented.

A blockchain–IoT-based food traceability system (BIFTS) is proposed in [139]. Not only it does integrate BC and IoT, but also proposes an appealing quality decay evaluation of perishable goods based on fuzzy logic. A critical point in this architecture is the fact that, to get to the BC, the data have to pass through a cloud server. This represents in any case a type of centralization, which we would like to avoid. In fact, it is not possible, inside the cloud, to authenticate the data with respect to a certain device.

In [140], authors present a blockchain-based fair nonrepudiation service provisioning scheme for Industrial IoT scenarios. It is a complex and interesting framework in which the BC acts as an evidence recorder and a service publication proxy. A limitation of this proposal is that their devices communicate through a BC node, like the first intermediate solution proposed later in Section 4.1.4.

The perspective presented in [141] is really compelling. Their solution features the direct signing capabilities of devices, which is one of our aims, and the offloading of communication with the BC to a proxy service. The framework is built in such a way that the proxy server itself cannot forge the devices' signatures, thus maintaining the authenticity of the data. Also, the applicability to the cold-chain use case is the same as our approach. However, their solution relies on the use of Hyperledger Fabric as a permissioned BC, which is not public by definition. A permissioned BC implies the presence of a certain authority issuing credentials and certificates, which is indeed a form of centralization. Moreover, the SDK proposed does not seem ready yet to target very constrained IoT devices built upon a microcontroller.

The advantages of the proposed solution are the following:

- Implemented devices do not perform mining and do not need to store the whole BC, making this solution suitable for low-end hardware (low-power and low-cost),
- The use of a BC (Quadrans) directly derived from Ethereum and natively supporting smart-contracts, with average block times of around 5 s,
- IoT and BC levels communicate directly, without a cloud intermediary,
- Every device has an associated *wallet*, meaning that it is possible to guarantee principles of data-authenticity and non-repudiation since the device itself signs the data before sending it to the BC.

4.1.4 Proposed Architecture

The proposed architecture is intended for solving the weaknesses presented above, and possibly to introduce some improvements to the described context.

It consists of enabling commercial or ad-hoc Internet-of-Things devices to communicate directly with the BC infrastructure through a *gateway* device to which they are connected using wired or wireless protocols. Such a gateway device should be able to sign-in-place the data sent from the IoT data-logging device and then to directly push on the BC. The most straightforward method of doing so would be to let the gateway be a full node of the chain (i.e., a node in the peer-to-peer network that can store and verify the whole chain and keep it updated too). However, this is unfeasible as the target application context lies in the embedded systems domain, which commonly consists of limited computing capability devices, for sure equipped with not enough storage to maintain an entire copy of the chain.

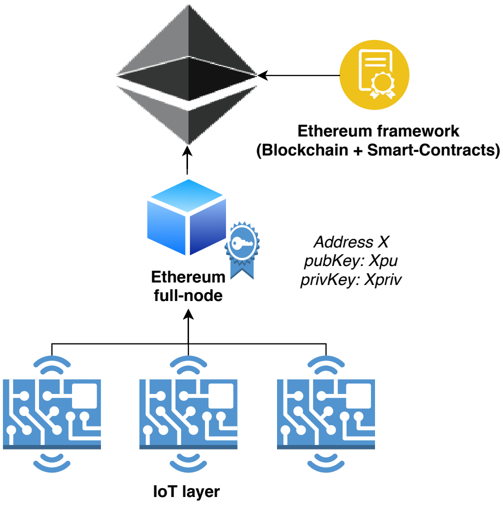


Figure 4.4: A representation of the first explored solution. IoT devices communicate with the local full node which is in charge of signing and broadcasting transactions for them. However, the IoT layer remains agnostic of the existence of the Blockchain. It simply sends data to a local centralized entity. (Image created using https://draw.io)

The first intermediate solution (Fig. 4.4) that has been explored is based on an architecture where the IoT layer communicates directly with an Ethereum full node placed within the same Local-Area-Network in which the devices are connected. Such a local node is then in charge of signing and broadcasting the transactions requested by the devices, as well as keeping their private key storage and mapping (between IoT device ID and private key). The implementation of this solution resulted to be fast and straightforward, because the only part of which one has to deal with requires only the communication mechanism between the local BC node and the IoT devices, defining the data model of the message to be exchanged and eventually some encryption mechanism to secure the communication channel.

However, this solution suffers from the same problems related to centralization and security as the common case previously described. Nevertheless, it would only be worse if no mechanism for device authentication was implemented for the communication with the local full node.

Therefore, the path followed in the project exploits the Remote-Procedure-Call (RPC) protocol [142] for enabling an embedded device (acting as a gateway) to trigger the execution of some procedure on a remote server exposing such service. Such a device has to be physically placed in the very near proximity of the IoT device (eg., as expansion hardware for the device) or even be the IoT device itself. The RPC server exposes to calling clients (i.e. IoT gateways) the Web3 Ethereum API (Application Programming Interface), that allows to interact with Ethereum's BC, so that any embedded device can access the information stored in the chain, exchange coins (eg. for implementing machine-to-machine payments) and call the execution of Smart-Contracts for implementing complex logic on top of the blockchain layer. Moreover, exploiting the events mechanism of Smart-Contract [143], and the possibility to search for those events in the chain by selectively download block headers through RPC calls, it could even be possible to trigger IoT devices directly within distributed applications and keep the history of such requests. To apply this solution, however, there should be one strict requirement above the others: *the gateway must be identifiable on the networks through its address, and should also be able to sign the transaction locally and off-line, using its private-key, before sending it to the RPC server.* This functionality is essential to avoid the possibility that the transaction is maliciously manipulated when it is sent to the server.
The final overall proposed architecture is shown in Fig. 4.5.

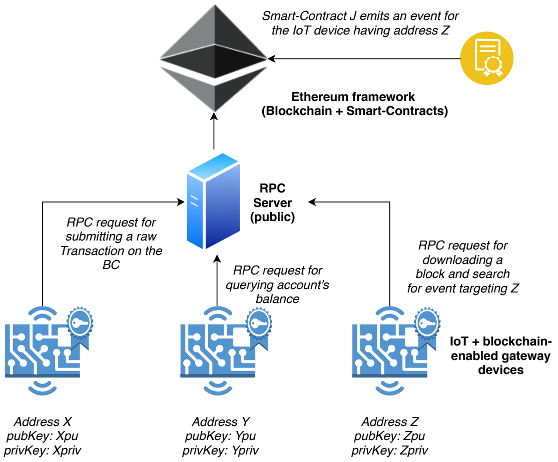

Smart-Contract J emits an event for the IoT device having address Z
Ethereum framework (Blockchain + Smart-Contracts)
RPC Server (public)
RPC request for submitting a raw Transaction on the BC
RPC request for querying account's balance
RPC request for downloading a block and search for event targeting Z
IoT + blockchain-enabled gateway devices
Address X pubKey: Xpu privKey: Xpriv
Address Y pubKey: Ypu privKey: Ypriv
Address Z pubKey: Zpu privKey: Zpriv

Figure 4.5: A representation of the final proposed architecture. Each IoT device has its own gateway and can sign transactions locally and offline. Each of them is also identified within the blockchain through its address and can be thus a target for possible Smart-Contract events. (Image created using https://draw.io)

4.1.5 Software architecture

Most of the project complexity has been demanded to the software layer, meaning that the vast majority of the operations are performed by the CPU without any additional special-purpose peripherals or hardware accelerator.

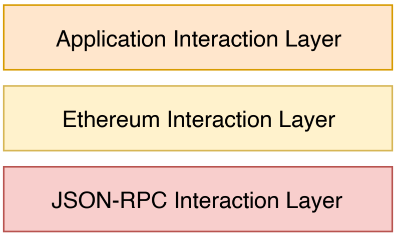

Application Interaction Layer
Ethereum Interaction Layer
JSON-RPC Interaction Layer

Figure 4.6: Representation of the Software Architecture (Image created using https://draw.io)

The developed software framework is intended to be as easy and user-friendly as possible. In this way, it can be used by the majority of IoT developers, with minimal knowledge of the details behind the Blockchain architecture. Following this philosophy, the resulting framework can be represented as a layered architecture (Fig.4.6) where the top-most software components are those providing the interaction services to the common Internet-of-Things application.

The JSON-RPC layer

The Remote-Procedure-Call is based on the Client-Server model and is used in those contexts where one program wants to request a service from another one, executed in a remote host. An RPC call is a synchronous event requiring the caller program to be suspended until the remote procedure returns its result. An example of this behaviour can be found in Fig. 4.7. JSON-RPC represents a state-less and light-weight implementation of an RPC. It is an extension of the original RPC protocol, which uses the JSON data-format for remote-procedures, parameters and results encoding. [144].

In the developed framework, JSON-RPC Client is the software component in charge of performing JSON-RPC calls over the TCP socket to which the RPC server is listening. Its implementation consists of a queue, where other tasks push a data structure for each RPC-Call that they request. This data-structure is used to model the standard fields of a Remote-Procedure-Call and, in addition to those fields, another one is provided to store the current state of the call. Each call, in this implementation, can assume the following logical states:

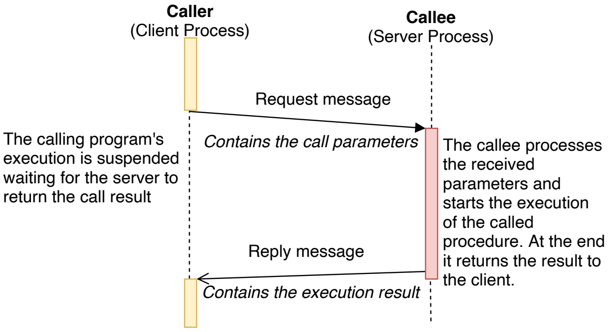

Caller
(Client Process)
Callee
(Server Process)
Request message
Contains the call parameters
The calling program's execution is suspended waiting for the server to return the call result
The callee processes the received parameters and starts the execution of the called procedure. At the end it returns the result to the client.
Reply message
Contains the execution result

Figure 4.7: A sequence diagram showing the flow of execution of a remote-procedure-call. (Image created using https://draw.io)

- `QUEUED`: it is the first state assumed by a call-object. It describes a call that has been pushed into the queue by a caller task and it is waiting to be executed,
- `SENT`: the call has been sent to the RPC server successfully,
- `WAIT_RESPONSE`: the client is waiting for the server response for this call,
- `SUCCESS`: the call has been executed successfully and its result is now available,
- `ERROR`: an error occurred during the call processing,
- `DELETED`: the call has been removed from the queue.

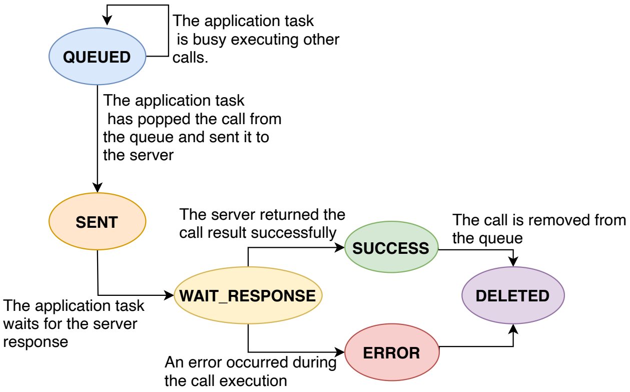

QUEUED
The application task is busy executing other calls.
The application task has popped the call from the queue and sent it to the server
SENT
The application task waits for the server response
WAIT_RESPONSE
The server returned the call result successfully
SUCCESS
The call is removed from the queue
DELETED
An error occurred during the call execution
ERROR

Figure 4.8: The state-transition diagram for a call-object. (Image created using https://draw.io)

A graphical representation of the state-transition diagram for a call can be seen in Fig. 4.8. The call queue is accessed in a circular-buffer manner and its dimension, in terms of the number of calls, is static and does not rely on dynamic memory allocation. The module's user can then set the dimension according to the application requirements, through a C-language `define` pre-compiler instruction and select therefore the optimal trade-off between memory occupation and performance. Summarizing, the `JSONRPC_Client` interface provides functions for the upper layers , allowing the caller to:

1. Push a new RPC call in the queue,
2. Retrieve the status of a call previously submitted,
3. Retrieve the result of an executed call,
4. Delete a call,
5. Check the queue status (empty/full).

Besides, the module implements the application-task function (`JSONRPC_Task`) in charge of maintaining the queue, converting calls-object into JSON, sending calls over the TCP socket, retrieving the response, and parsing it from JSON. The sequence diagram in Fig. 4.9 illustrates an example of an RPC call execution.

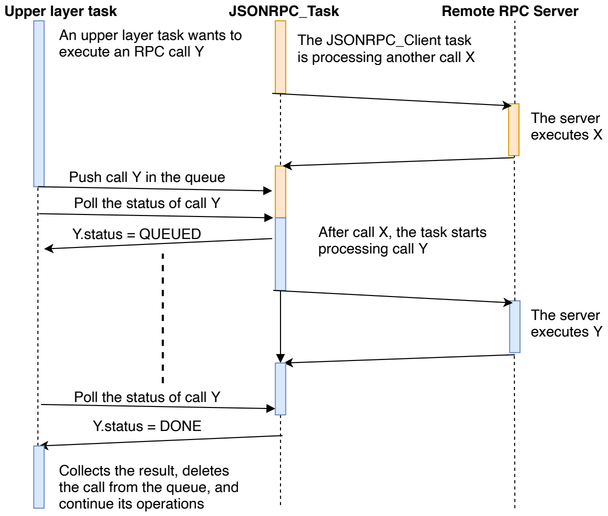

Upper layer task
JSONRPC_Task
Remote RPC Server
An upper layer task wants to execute an RPC call Y
The JSONRPC_Client task is processing another call X
The server executes X
Push call Y in the queue
Poll the status of call Y
Y.status = QUEUED
After call X, the task starts processing call Y
The server executes Y
Poll the status of call Y
Y.status = DONE
Collects the result, deletes the call from the queue, and continue its operations

Figure 4.9: A sequence diagram illustrating how the developed JSONRPC client interacts with upper layers requesting a remote call to be executed. (Image created using https://draw.io)

The Ethereum interaction layer

The Ethereum interaction layer is one of the frameworks that implements a subset of the functions provided by the Web3 API. Each of the provided functions is in charge of encoding the necessary parameters for the correspondent RPC call to be executed, as well as adapting the returned result into user-friendly C-language types.
The provided functions allow the caller to check the status of an account (i.e., nonce and balance), perform transaction and message calls, and request a specific set of blocks from the chain. The fundamental data structure of this layer is the `WEB3_ETH_OBJ` that holds the information about the RPC call object as well as the state of the Web3 call. The implementation of these functions follows the state-machine model,with a given web3-call evolving through the following states:

- `CALL`: Parameters are encoded into JSON format and the proper RPC call is pushed into the queue of the JSONRPC client module,
- `WAIT`: Polling on the call-object status to check when the result is ready or an error is returned,
- `DONE`: The result is ready to be processed,
- `ERROR`: An error occurred.

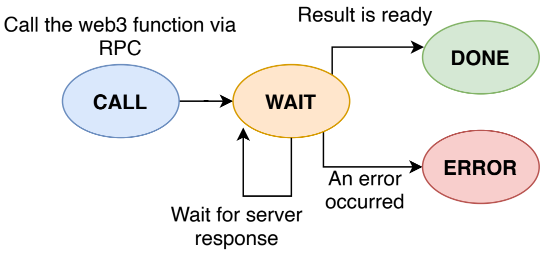


Figure 4.10: The state-transition diagram for a web3 call in the developed framework. (Image created using https://draw.io)

A graphical representation of the state-transition diagram for a Web3 call can be seen in Fig. 4.10.

The Application interaction layer

The Application interaction layer represents a wrapper of the Ethereum interaction layer in the sense that it hides the details for performing specific operations on the blockchain (eg., signing a transaction before sending it as raw). It wraps all functions of the layer below and performs the preprocessing and postprocessing

operations needed for some of them.
Again, they are internally implemented as state-machines, cycling through the following possible states:

- `WEB3TASK_INIT`: In this state, all preprocessing is performed before the web3 function is called,
- `WEB3TASK_WAIT_RESPONSE`: The state machine waits for the web3 call to be executed and returns its result to perform the post-processing operations,
- `WEB3TASK_IDLE`: Nothing to be done,
- `WEB3TASK_ERROR`: An error occurred.

4.1.6 Use case application

This chapter presents a use case covering the supply chain of fish and seafood products. The use case has been simulated in collaboration with Foodchain S.p.A., a company whose mission is to enhance the traceability process of food's supply chain through the Blockchain technology.

To simulate the use case scenario, an IoT device has been built and programmed in such a way that it can autonomously communicate with the Blockchain by directly signing its own transactions. Moreover, to corroborate the usefulness of the system, an actual use case scenario has been depicted later in this Section.

For what concerns the Blockchain side, Foodchain S.p.A. has developed a platform, based on Quadrans' infrastructure, that provides services for both companies and final buyers. Companies can use the platform as a common Enterprise Resource Planning (ERP) software registering products, batches, and processesand any information related to the product's lifecycle (documents, certificates, media, etc.). Such information is immutably registered into the BC, on top of which the platform lies, using Smart-Contracts. At the end of the product's transformation process, the platform generates some kind of Smart-Label (like a QR code or an RFID tag) that uniquely identifies the product and can therefore be scanned by the product's buyer for viewing the entire traceability process. Another important feature provided by the platform is that it enables many companies and producers to cooperate together within the network, allowing them to follow the transformation cycle of a product even when it passes through the production and distribution chains of different actors.

Food-chain issues

Recent studies have demonstrated that customers' loyalty to food companies and in all products they sell has drastically decreased year after year. In 2018, the Center for Food Integrity published its annual report about consumers' habits and

feelings when buying food. The investigation showed that an increasing number of customers are concerned about what they eat, thus they are asking for more information about the food they buy to choose the healthy one [145]. However, most of them do not trust the information written behind the product's package, especially when it does not directly come from farmers, but it goes through some transformation processes instead. Besides, the awareness of consumers about the damages caused to the environment by some farming and breeding processes is encouraging them to not buy entirely certain categories of products [146].
The lack of trust in food producers is the consequence of a minimal customers' knowledge about the whole supply chain and also the fear of a globalized market that may allow fraudulent producers to sell dangerous or counterfeited products. In particular, this last aspect is critical both for consumers, who are subjected to risks related to their health, and "honest" producers, who not only suffer the economical damages but are also affected by the side-effects of the distrust of buyers. In the year 2018, the Italian Central Inspectorate of Quality Protection and Fraud prevention of food products (ICQRF) has registered an increment of 58%, compared to the previous year of crimes related to food frauds, seizing about 17.6 tons of goods, for an equivalent value of over 37 million USD. The wine sector resulted to be the most affected by this phenomenon [147].

The concept of Food Trust

From the previous analysis, it is clear that consumers' confidence in the agri-food industry is the result of their concerns or certainty on various aspects related to the product they intend to buy. One of the most important factors to restore a good level of reputation of the food industry is to guarantee **food safety**, which is every day undermined by the numerous cases of food hazards. The lack of information about transformations and events under which the final product has gone through during the supply chain poses the basis for a serious alarm about the health of citizens and, despite the efforts of government in control and prevention, the current infrastructure turns out to be unsuitable and inefficient in the fight against this phenomenon. Another important aspect of food trust is related to the concepts of **food-integrity** and **food-authenticity**, which have a significant impact both on the healthcare and the economy of countries. Food-Integrity defines the state of being undiminished and unaltered with respect to its original nature, which means that no unapproved and undeclared transformations and alterations have been made on such products. The Food-authenticity definition, instead, is more related to economical aspects, like frauds, that generally do not represent a risk for the health of consumers. However, food fraud and counterfeiting are hurting the markets of many countries, causing losses of billions of euros per year. The most subjected products to counterfeiting threats are those that earn a great economical value because of their territory of origin or brand, like oil, wine, and cheese. In

July 2016 the European Union Intellectual Property Office (EUIPO), has published research about the economical costs of intellectual property infringement in spirits and wine. The results showed enormous damage to the legitimate industries, with estimated losses of approximately 1.3 billion euros of revenue annually [148]. Last but not least component of the concept of food trust is linked to customers' attention, concerning the sustainability of processes and the agri-food chain. Consumers are becoming ever more aware of the role they play in the game for building a sustainable food industry. Most of them, especially the young ones, base their decision making when buying a product not only on the "classical" factors (price, nutritional value, brand, etc.) but also on the impact that the product they buy has on the environment. The expansion of this new category of customers, known as *Ethical Consumers*, is leading this new market sector to grow very fast, registering in 2015 a growth of 5.3% and 9.7% in 2017 (UK Market) [149], [150].

Combining IoT and Blockchain to empower the traceability process

There is a strong necessity for an enhancement in the monitoring and certification processes to overcome these issues. In the actual situation, most of the compliance data and information are audited by trusted third parties and stored either on paper or in centralized databases. However, these approaches suffer from many informational problems, such as the high cost and inefficiency of paper-based processes, resulting in fraud, corruption and error happening both on paper and using IT systems.

Thus, there is a need for a new trustworthy approach for registering information about the food chain. The availability of this information to customers is finding a potential strong partner in the arising Blockchain technology. Indeed, the application of the BC technology in the agri-food sector could lead not only a better and more trustworthy traceability process, but it also could represent a powerful partner in the assessment of the product's value for the buyer. For these reasons, there are actually several experiments and proof-of-concepts aiming to find a way to apply the BC technology into this business field. This potential can be truly unlocked by enabling the blockchain platform to collect, and register on the ledger, data gathered directly from the environment (such as farm fields, food transformation chains, logistic processes, etc.) employing IoT devices [138].

System architecture

The scenario in which the use case is applied is about the cold chain monitoring during logistics operation. In such a context, the IoT device is assumed to be installed into a refrigerated truck that transports seafood from a harbor's warehouse to the final fish shop. The device can connect to the Internet through a mobile

network connection and is in charge of monitoring the fridge's temperature and truck's position employing a temperature sensor and a GPS module respectively. As soon as the data are collected, the device then sends them to a proper Smart-Contract, in charge of storing them in the BC, using the IoT-blockchain gateway presented in this chapter.
The gateway's hardware configuration used for this specific simulation consists of:

- A PIC32MX 32-bit microcontroller
- A 256-byte EEPROM (used for storing device keys in the developed Proof-of-Concept, keeping in mind that in a real-life application this should become a secure-storage element to avoid the stealing or alteration of the device's key)
- A GSM Module for communicating with the remote RPC server
- A GPS module
- A Temperature sensor

This simple architecture has been chosen to show the limited requirements of this approach. The implemented code is for sure hardware-dependent, but the general architecture is independent from the hardware chosen, and applicable to any kind of similar device.

Moreover, the simulation has been carried out using the Quadrans Testnet blockchain, accessed through its public RPC node, reachable at the address:

```
rpc.testnet.quadrans.io
```

. The communication with the remote RPC is based on HTTPS protocol, which ensures a higher layer of security within the interaction between the IoT device and the remote RPC server.

Monitoring the cold chain of fish products

The boats of Fresh-Fish's fleet are equipped with our device, able to periodically record GPS coordinates. This makes it possible to have a complete history of the route navigated by every boat. When the boats arrive at the harbor, fishermen store the catch in the company's warehouses. Here, each batch of fish (e.g., one kilogram of mussels) is registered on Food-Chain's platform, through its web interface, and is identified with a unique ID that is written into an RFID tag. Then, according to fish shop daily requests, each batch is charged into refrigerator trucks to be delivered to its final destination. When a batch is charged on the truck, the device installed on the truck scans the RFID tag on the batch package so that the device knows for *which product it is going to start the monitoring process. Once the truck starts its delivery path, the IoT device starts to periodically monitor the temperature and the GPS position of the truck and sends data to the proper smart contract for being registered, providing also the IDs of the batches under the current monitoring process. Eventually, when the truck reaches its final destination, batches' tags are scanned again for registering that the delivery has been completed and that the recipient retailer is now the owner of such products. At this point, the retailer shares the information with the customers by tagging its product with a specific QR code, that represents an entry point to access the information stored in the BC. The customers can verify the complete story of the mussels and check if any violation of the cold-chain parameters has occurred.*

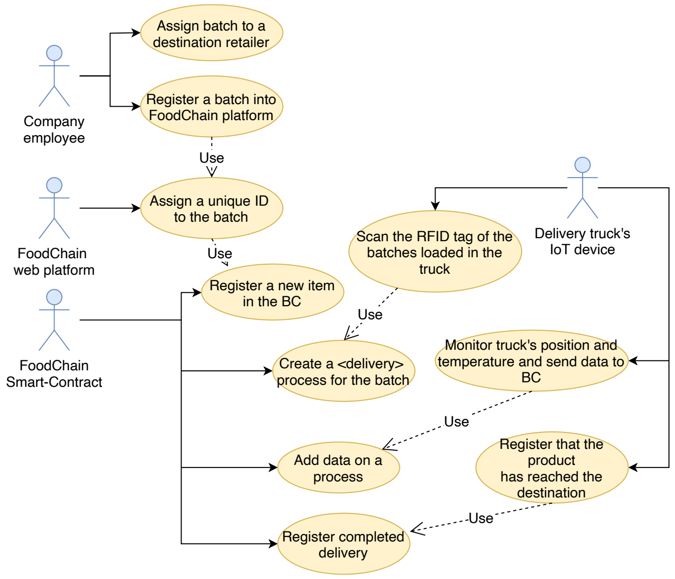

Company employee
Assign batch to a destination retailer
Register a batch into FoodChain platform
Use
FoodChain web platform
Assign a unique ID to the batch
Use
Register a new item in the BC
FoodChain Smart-Contract
Delivery truck's IoT device
Scan the RFID tag of the batches loaded in the truck
Use
Create a <delivery> process for the batch
Monitor truck's position and temperature and send data to BC
Use
Add data on a process
Register that the product has reached the destination
Use
Register completed delivery

Figure 4.11: The UML use case diagram of the system. (Image created using https://draw.io)

A UML diagram of the use case is depicted in Fig. 4.11.

A description of a scenario for the use-case involving the IoT device is given in Table 4.1.

4.1.7 Results and comments

The proposed system tackles the two fundamental issues in the food chain domain: food integrity and food safety. These are the crucial parameters that efficiently protect customers from fraud or health-related problems, with the positive side effect of helping them to become loyal to a certain vendor. This permits a shift in the paradigm from a trust-based situation to a trust-less scenario. In the former, the buyer is led to put trust in a certain vendor because of an ad campaign, while in

Step	Description
Pre-Condition	*The item X exists and a <delivery>process p has been activated for it*
1	Read Temperature from temp. sensor
2	Get the position from the GPS module
3	Encode data in the proper format required by the target Smart-Contract
4	Build the transaction object
5	Sign the transaction using the device's private key stored in the EEPROM
6	Send the signed transaction to the BC through an RPC call
8	Wait for the server to reply with the hash of the transaction submitted on the chain
Post-Condition	*Gathered data are now immutably stored in the BC for the item with id X.*

Table 4.1: Use case scenario description

the latter the vendor shares with her all information about the product, that cannot be tampered thanks to the Blockchain technology. The use-case demonstrated how the fish tracking process, carried out combining both Blockchain and IoT devices, can enrich the amount of information that is shared with final consumers guaranteeing their authenticity and integrity without relying on a centralized trust authority.

Moreover, from the logistic company point-of-view, publishing data about their food shipping publicly and with transparency can become a good marketing tool for acquiring new customers interested in both shipping their products and, at the same time, sharing data about shipping quality with their final consumers.

Last, but not least, the data-authenticity and data-immutability properties implemented by the blockchain can make the information registered by IoT devices valuable in legal trials, in case of debate about food-safety issues or even issues related to the shipment insurance.

Table 4.2 presents a structured comparison between the proposed solution and the other papers referred in the literature and described before. As the table shows, despite the fact that also other implementations in literature could be theoretically

Reference	**IoT hw required**	**Full node required**	**Cloud required**	**Kind of BC**	**On-device signing**	**Directly applicable to cold-chain monitoring**
[135]	n/a	yes	yes	Custom, based on bitcoin	n/a	no
[136]	high-end/SBC	n/a	n/a	Ethereum	n/a	yes, but requires high-end device
[137]	high-end/SBC	yes	no	Ethereum	yes	yes
[138]	n/a	yes	no	Ethereum and Hyperledger	n/a	no
[139]	low-end	n/a	yes	n/a	n/a	no
[140]	high-end/SBC	yes	no	Ethereum	n/a	no
[141]	high-end	no	no	Hyperledger	yes	yes, but on a permissioned BC
This paper	yes	no	no	Quadrans	yes	yes

Table 4.2: Comparison between the proposed paper and the cited literature, with a focus on cold chain monitoring.

applied to the cold-chain monitoring task, the one presented here is the only one able to fulfill all requirements together: actually working on resource-constrained IoT devices; the lack of need of a local full node of the BC; the lack of an intermediate and possibly insecure cloud platform; operating with a public domain blockchain, without the need for trust-certifying parties; employing devices able to directly sign their own transactions.

To estimate the performance and the effectiveness of the presented system, the following metrics have been considered:

- Hash Time (T_{hash}): the time needed by the microcontroller to hash the transaction, previously encoded as byte-stream using Recursive-Length-Prefix (RLP) algorithm. This metric is related to the amount of data that is included in the transaction.
- Signature time (T_{sign}): the time needed by the microcontroller for executing the ECDSA algorithm and generating a valid signature for the transaction's hash. Since the signature is always computed over the hash of the transaction (having a fixed length of 32 bytes), this metric is not related to the amount of data making up the transaction, but it gives a good index for evaluating the performance of the hardware on which the system is based.
- Confirmation time ($T_{confirmation}$): the time needed by the BC network to validate the transaction sent by the IoT device. This metric depends only on the blockchain layer and is strictly related to the average block-time of the network.

		Min.	***Avg.***	***Max.***
T_{sig}	**[ms]**		*15*	
T_{hash}	**[ms]**	*2*	*3*	*5*
$T_{confirmed}$	**[s]**	*12*	*16*	*18*

Table 4.3: Use case performance metrics

These three metrics have been chosen since their sum represents the least possible time interval that must elapse between the moment the data are gathered by the device (i.e. an on board sensor) and the moment these data become immutable in the blockchain.

From Table 4.3 it can be noticed that, as expected, the time required by the microcontroller for running the ECDSA algorithm is constant, and represents the major component of the total time needed by the gateway for preparing and issuing a transaction ($T_{transaction} = T_{sign} + T_{hash}$, in first approximation). At the moment, both hashing and ECDSA operations are implemented in software, but it would be worth exploring possible implementations of the system that can rely on hardware accelerators (maybe implemented on an FPGA connected to the micro-controller) which can speed up the execution of such tasks. Another important aspect is related to the time needed by the BC network for validating the transaction. It can be observed that such time goes in the order of 10-20 seconds, which reflects the average block-time of Quadrans Testnet that is about 15 seconds. Even if Quadrans Mainnet, the one used in production contexts, has a block-time of 5 seconds that would result in a 3x speed-up, it can still represent a barrier for some time-critical applications in automation and automotive fields, for example the control of a motor/robot or the automatic emergency braking of a car. However, in these particular mission critical applications, the use of a BC does not bring any benefit to the system architecture. This aspect goes beyond the goal of this paper, but trying to improve and reduce this time can become a valuable research case for the Quadrans Foundation, because a further improvement in this direction will open the possibility to adapt the presented gateway also in scenarios where time constraints are more strict.

4.1.8 Conclusions and Future Works

Summarizing, the integration between two emerging and disruptive technologies such as Internet-of-Things devices and BC-based infrastructure for decentralized applications can bring benefits in a very wide variety of fields, spanning far beyond the simple use case presented in this document. The proposed and developed system gives a proof-of-concept of a possible path to follow for a successful and efficient interaction between Internet-of-Things devices and decentralized applications based on a Blockchain ledger. The *blockchain-enabled gateway* presented in this chapter satisfies the essential requirements for performing secure and reliable data operation onto the BC infrastructure and, at the same time, provides both hardware and software interfaces for easy and seamless integration even with already developed applications. From the side of the decentralized applications, the possibility to add functionalities that can use data coming directly from the environment, without doubting over their integrity or authenticity, represents a powerful tool that can be exploited to enhance a lot of processes. Finally, for business actors, the services offered by the resulting kind of systems allow them to run a new form of marketing, focused on product and processes transparency. However, the possible application of this promising technology combination can go even further from traceability and business processes, and be applied for example to healthcare, infrastructure, services and so on.

Another important feature provided by the designed architecture, and which should be worth exploring better by developing proper use cases and DApps, is the one allowing Smart-Contracts running on the BC to asynchronously trigger some action on specific target IoT devices, identifiable on the network through their public address. Shortly, the idea would be to implement some logic on the IoT devices in order to periodically download the latest N blocks in the chain and search for some emitted event which is targeted to them. However, further studies should evaluate how these concepts can be applied in soft or hard real-time embedded systems, for example starting from the time-critical applications cited above. An exhaustive analysis should take into account not only both the delay of the network and the architecture itself, but also the delay related to the block-time parameter of the BC. Probably, explorations in this direction may lead to the development of

IoT-oriented Blockchain infrastructures that optimize their parameters to be more adaptable to those contexts where timing constraints are critical and sometimes restrictive.

In conclusion, with respect to the currently applied solutions, the one presented in this document allows to develop decentralized applications of any kind that could interact directly with the Cyber-Physical-System, keeping costs low because they do not need to rely on any other service for accessing to the distributed network and, at the same time, satisfying the constraints about data-integrity and data-authenticity.

4.2 Quantum Computing - new opportunities and threats

The concept of data authenticity and safe storage in the IoT domain is a novelty. Despite being such a new theme, it is already endangered by the advent of Quantum Computing. This chapter section draws an overlook of the new paradigm of Quantum Computing, starting with the basic introductory concepts, up to how such paradigm poses a menace to the topic of interest of the chapter, the safe storage of data.

Possible attack scenarios to the cryptosystems used nowadays are reported, along with alternative quantum-proof possible industrial solutions.

4.2.1 Introduction

Global scale of industrial production has reached dimensions for which it is difficult for a human being to keep the pace at which the data have to be recorded [151]. This process has necessarily been automatised with the help of a plethora of different technologies. This task automatization is crucial in traceability applications, which find natural employment in the automotive domain, both from a supply chain perspective and from a vehicle management perspective. There is also the need for storing these traceability data in a decentralized way[152], either to avoid a single point of failure or to ensure the data cannot be altered by some malicious entity. Blockchain and other Distributed Ledger Technologies (DLTs) are then the way to go if such needs are in place[153].

As previously said, the blockchain is a distributed ledger of timestamped records (transactions), built in such a way that it is tamper-proof.

The integrity of blockchains relies basically on two points: *a)* the signature scheme encrypting it and *b)* the fact that one malicious actor is supposed not to be able to control the majority of the network's computing power.

Several DLTs also feature the possibility of implementing Smart Contracts, which are pieces of code that can automatically enforce actions depending on the status of the ledger.

All these amenities and commodities provided by DLTs could cease with the advent of large Quantum Computers - i.e., with a sufficient number of qubits to become dangerous. This kind of devices could, for instance, be used to break the RSA cryptosystem or to take over the entire ledger network thanks to their ability to solve mathematical problems [154].

Several companies and research centers are trying to address this issue by designing and implementing quantum-resistant DLT.

The rest of the chapter is organized as follows: in Section 4.2.2 some traceability examples are described; in Section 4.2.3 some information about quantum computers are provided; in Section 4.2.4 the possible attacks of quantum computers to blockchains are described; in Section 4.2.5 some possible solutions are presented; in Section 4.2.6 some implementations of these solutions are reported; in Section 4.2.7 conclusions are drawn.

4.2.2 Traceability scenarios

In this section, two example applications of DLTs for traceability-related purposes are presented.

Smart contracts - Traceability

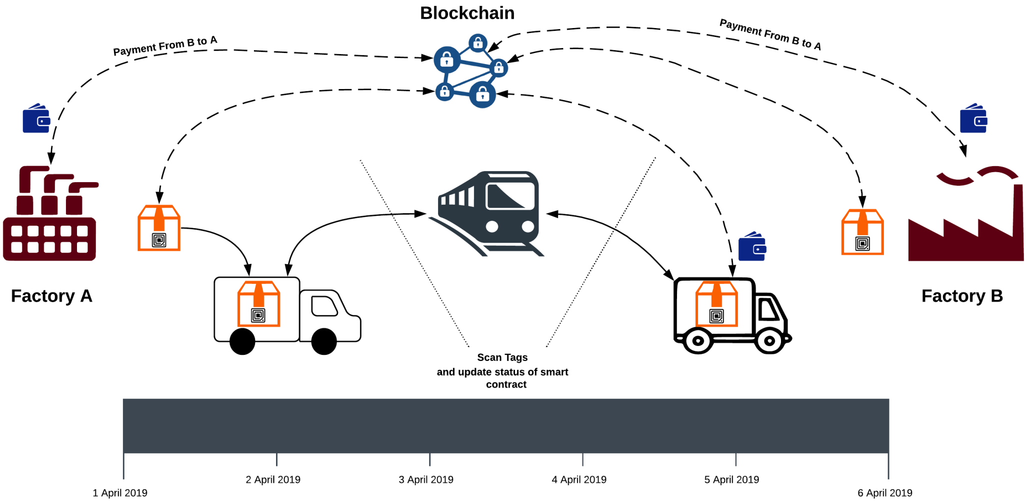


Figure 4.12: Traceability example[155]

The main branch of application is related to the realization of smart contracts[132]. Smart contracts are in practice pieces of code, programs, written in some programming languages. This kind of tool first appeared in the Ethereum [156] blockchain. Smart contracts are written in such a way that the set of conditions in the program matches the clauses making up the actual contract written and signed on paper.

Among the sectors in which blockchain is very useful, there is the **supply chain**. Goods can be tracked from raw material to finished product (as in Fig. 4.12). Furthermore, the inclusion of smart contracts makes it possible to directly and automatically pay the company which is the source of some goods, just by interrogating the incoming deliveries of boxes identified by a tag of some sort. In this way, the company in charge of the delivery can be automatically paid. Moreover, there is an added advantage in the single update of a shared ledger, instead of updates to several separate databases which can get into conflict.

For all final products, the **list of components** can be kept up to date and a single component can be traced back up to the original manufacturer (i.e. parts in a car).

This data has for sure also direct implications in **insurance applications**. In fact, tags identifying goods are read throughout the entire supply chain, passing the responsibility of the transported load from entity to entity.

Digital wallet / digital twin

One of the main fields of application for blockchain technology is the automotive sector.

The term *digital wallet* is intended as a wallet owned not by a physical person, but by a device, which could in principle be a vehicle, having the ability to automatically pay some other device in an automatic way for some service they may provide. It is necessary that some sort of digital information is matched to the car, with a *logical id* stored in the blockchain which binds to a physical vehicle in the real world.

There are **insurance** applications also in the automotive domain, as the automatic payment of the insurance fee for the car and the automatic update of the ledger related to the vehicle and insurance contractor in case of accident or selling. On a daily basis, a moving vehicle could **autonomously pay** for the parking spots it occupies, for the tolls encountered on the road or for the access to some restricted areas, which are ever more common in city centers due to pollution control. Moreover, such a blockchain-enabled car can autonomously pay for **refueling** (as in Fig. 4.13), whether the fuel is petrol or diesel or electricity. In case of need, the car could take care of paying planned or unplanned **repairs**, while keeping the ledger up to date in case of component substitution. Furthermore, it has to be noted that money flow could happen also in the other way. Cars can actually **make money**

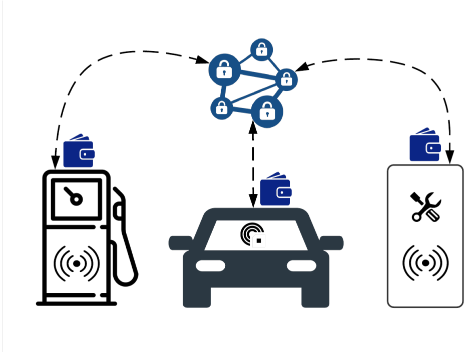


Figure 4.13: Car automatic payment[155]

by selling services, such as deliveries, rides, or ads on the bodywork.

4.2.3 Quantum Computing

Quantum Computing (QC) is a new computing paradigm, aiming to leverage the principles of Quantum Mechanics to perform calculations.

Since it was born as a pure theoretical matter of study, it fell in the interest of researchers solely in the areas of physics and mathematics.

When IBM (International Business Machine) introduced its first quantum computer in 2016, the audience of such machines started growing and including computer science and engineering researchers.

As of today, there are countless big and small companies and start-ups trying to build a quantum computer: Microsoft, Intel, Rigetti, Xanadu etc.

The basic unit of information is the quantum bit or *qubit*. It is a two-level quantum mechanical system, which show interesting and unique features:

1. *Superposition*: a combination of states.

2. *Entanglement*: two entangled particles bind together to form a system. They can interact with each other even if they are far apart from one another and their states are intertwined.

Whereas a classical bit can assume the values 0 and 1, the quantum bit is formalized as a bi-dimensional array. The two basis states are $|0\rangle$ and $|1\rangle$, expressed in "ket" or Dirac notation:

$$|0\rangle = \begin{bmatrix} 1 \\ 0 \end{bmatrix} \qquad \text{and} \qquad |1\rangle = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$$

These two are orthonormal bases defining the vector space. When we say that a qubit is in *superposition*, its state is defined by a linear combination of these two basis states:

$$|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$$

where α, $\beta \in \mathbb{C}$ represent the probability amplitudes, which have to obey the rule:

$$|\alpha|^2 + |\beta|^2 = 1$$

where $|\alpha|^2$ is the probability that the qubit will be measured in the state of $|0\rangle$ and $|\beta|^2$ is the probability that it will be measured in the state of $|1\rangle$.

A qubit in superposition ($|+\rangle$) is represented as:

$$|\psi\rangle = \frac{|0\rangle+|1\rangle}{\sqrt{2}}$$

However, this state cannot be not measured experimentally. When the measurement is performed, the state of the qubit will collapse to 0 in 50% of the times, and to 1 the other 50% of the time. This is because the probability amplitude of both states is $|\frac{1}{\sqrt{2}}|^2 = 0.5$.

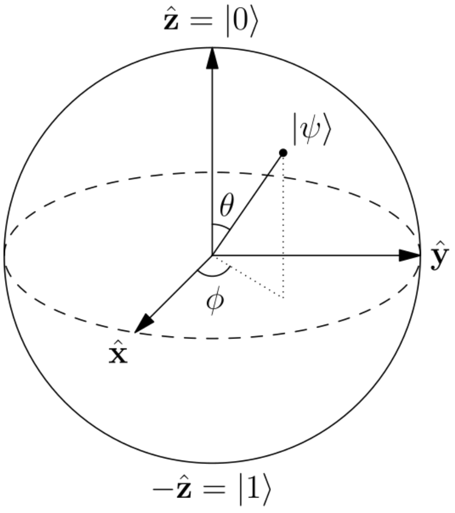


Figure 4.14: The Bloch sphere, a geometric representation of a two-level quantum system (credit: Glosser.ca).

A graphical representation of a qubit can be seen in Fig. 4.14, the so-called Bloch Sphere. The qubit has the $|0\rangle$ state to the north pole, while the $|1\rangle$ state lies at the south pole.

The actual state of the qubit is represented by the vector $|\psi\rangle$ which can take any value on the surface of the sphere.

The state of q qubit can be changed by applying operations called *gates*.

Standard gates include operations such as X, Z, H, Y, and similar ones. These are represented as matrices, which are multiplied by the array representing the state of the qubit.

A quantum computer made of n qubits is able to handle a space state of 2^n states.

At the starting of the execution of a quantum program, the problem to be solved is encoded in those n qubits. The execution of the program consists in changing the probability amplitudes of the qubits in such a way that the ones encoding the solution of the problem are increased.

4.2.4 Possible attack scenarios to the Blockchain

As pointed out in [154], there are several possible attack scenarios on distributed ledgers, which can be categorized by the operating principle on which it is aimed to: attacks on *Proof-of-Work* or attacks on *Signatures*.

Attacks on Proof-of-Work

Exploiting superposition, Grover's search algorithm [13] on a quantum computer can allow performing PoW quadratically faster with respect to a classical computer approach. Grover's algorithm, given a function and its output, can invert that function, finding the input generating that output. It is theoretically possible that a single large quantum computer, applied to the PoW task, would be able to take over the entire ledger network. Such QC could be able to validate transactions faster than the rest of the network, successfully performing *double-spending* transactions [157]. Of course, this scenario would not be possible if more than one miner had access to superior computing capabilities.

Attacks on Signatures

As said, public/private key cryptosystems in the majority of DLTs rely on some mathematical function, such as the *integer factorization problem*, which is very difficult for a classical computer to solve, but that would not be the case for a quantum computer, as Peter Shor demonstrated in 1994[158]. Indeed, a quantum computer executing Shor's algorithm could compute the private key associated with a public key (i.e. the address in the Bitcoin network), opening the path for several attacks, for instance:

- **Address reuse** - To spend cryptocurrency, addresses have to be revealed. Once revealed, an attacker could use the public key to retrieve the private one. For the sake of security, a new pair of public and private key should be generated for every transaction, which has a small, but not negligible cost.
- **Transaction in progress** - When a transaction is issued, the public key is revealed. If a quantum attacker is able to retrieve the associated private key before the transaction is accepted and stored on the blockchain, it could effectively steal all the remaining crypto-money on the compromised account.

4.2.5 Possible solutions to the Quantum Threat

In this section are presented some countermeasures for quantum attacks on *PoW* and on *signatures*.

Disclaimer: at the time of writing, there are yet no standards for Post-Quantum Cryptography. The National Institute of Standards and Technology (NIST) is at the second call of the standardization process[159].

Countermeasures on Proof-of-work

Alternative implementations of PoW could be proposed, as pointed out in [154], which could be:

- Flexibly difficult - adapting to the network load, as in the original Bitcoin blockchain;
- Asymmetric - difficult to solve for one node, but easy to be checked by all the other nodes;
- No quantum advantage - there should not be any quantum computing algorithm able to make this problem simple.

Another viable solution could be to shift to some other consensus algorithm, such as *Proof-of-Stake*[160].

Countermeasures on Signatures

Critical characteristics of DLT signature schemes are [154][161][162]:

- *Size* of signatures and public keys - since they have to be stored somewhere;
- *Time* - required to check the signatures.

Signatures should be strong as possible from the security point of view, and as small as possible to reduce the storage size and the checking time.

The best performing in this sense are *hash*-based and *lattice*-based cryptosystems, having the smallest dimension for the sum of signature and public key length[154].

4.2.6 Quantum-safe DLTs

The most relevant and most adopted DLTs, which are designed to include quantum-resistant features are: *QRL*, *Corda* and *IOTA*. Again, the reader has to keep in mind that still no standard exists at the moment [159].

QRL

QRL stands for Quantum Resistant Ledger [163]. QRL secures its signature against quantum computing power using a hash-based scheme, which is organised as an asymmetrical hypertree. It uses W-OTS+ (a variant of Winternitz One Time Signature scheme) and has the ability to quickly sign transactions. This ledger has been designed to be a public blockchain, with a cryptocurrency perspective. There should be a minimum fee for issuing a transaction, but its load should be floating and set by the market demand. This ledger is thus mining based, with a planned block time of 60 seconds, which is fair compared to the 10 minutes of the Bitcoin blockchain. The cryptocurrency proposed is the *quantum*, which has the *Shor* as fraction (10^{-9}). Transactions fee should be small and calculated in *Shor* units. QRL is for sure a very interesting open-source starting to create a large fanbase. It has to be noted although that at the moment, this project does not include Smart Contract capability, which is crucial for the the traceability of tomorrow.

Corda

Corda is an open-source, distributed ledger platform developed by R3 company [164]. It secures signatures using a tree-like hash-based scheme, as we have seen with QRL, but using W-OTS-T (another variant of the W-OTS scheme). It has been specially designed with financial applications in mind, but it can also be used in a variety of other fields, such as insurance, government and supply chain. It is a permissioned blockchain, meaning that only entitled actors can participate in the network for added security. This case is particularly useful within or in between companies who signed some deals. Speaking of which, Corda has a crucial feature regarding Smart Contracts. In parallel with the actual smart contract code, developed either in well-known Java or Kotlin, it can natively support also legal prose contracts, in such a way that they could be complementary to classical paper contracts. The adjunct factor is that this kind of contract is signed by all participants and safely stored in the tamper-proof blockchain. In order to ensure the fulfilment of the contract regulations, observer nodes could be included as supervisors, upon the access permission to the ledger has been granted. The ledger is updated using transactions, which change its state. Transactions have to be *valid*, meaning their smart contract code runs successfully and has the needed signatures, and *unique*, there is no other transaction which evolves from a previous state. The consensus on the transaction is reached only by the parties involved in that transaction, i.e. only those parties share these data. It has to be noted that in this kind of ledger there is no concept of PoW, thus there is no way in which a quantum computer could attempt an attack on the mining capabilities of the network. Moreover, it does not use a native cryptocurrency, which means it is easily integrable in current financial applications without the hassle of forcing the involved parties into adopting another currency.

IOTA

The other option is not based on the blockchain, but on the concept of *tangle*. IOTA [165] is an open-source cryptocurrency for the Internet-of-Things (IoT) industry. The main feature of this novel cryptocurrency system is the so-called *tangle*, directed acyclic graph (DAG) for storing transactions. It is made to be used also by small nodes with not much computing power. In fact, it is best suitable in *Machine-to-Machine* (M2M) micropayment applications, where the transaction fee could be larger than the actual amount of exchanged money. This kind of approach could be very useful if applied to the various use cases described before.

In order to issue a transaction, a node has first to verify two previous transactions. Unlike in classical blockchain ledgers, where there are transaction issuing nodes and mining nodes (with great computing power), in IOTA it is sufficient to check two previous transactions, while yours will be checked by some subsequent one. In this way, the IOTA ledger results in being transaction fee-free, rebalancing the roles of the participants in the network.

It has to be noted that the IOTA network is asynchronous. Nodes in the network may or may not see the same set of transactions from which to choose the two to validate. Moreover, nodes do not have to reach a consensus on what transaction has the right to stay in the ledger. Thus, all transactions can be in the ledger, but only the ones fulfilling some particular requirements will be actually validated, others will be orphaned. The particular requirements are defined in the *tip selection algorithm*, which has a flexible metric helping in the choice. One of the basic ideas is: the more a node is involved in the verification, the less likely its transaction will be rejected. Thus, a node still has an incentive in participating in the network even if not issuing any transaction. IOTA developers present the definition of *weight* of a transaction, which is proportional to the work produced by the issuing node for it. The higher the weight, the more important the transaction.

Given its different structure, a tree instead of a chain, this kind of ledger is more flexible with respect to classical blockchain. Moreover, for what concerns its resistance to quantum threats, since there is no mining, such an attack could not be tempted. Instead, a *large weight* attack could be carried out, but it is sufficient to block this at the protocol level, imposing a limit on the maximum transaction weight. The time spent validating some transaction is not so different from the time required to perform several tasks to issue a transaction, thus there is not much advantage in using a quantum computer applied to this kind of ledger.

The foundation is intensively working on Qubic[166], a name which comes from quorum-based computation. This platform is not intended to simply handle smart contracts, but involves the use of oracle machines to bridge the physical and a parallel, logical world, also featuring the outsourcing of computing power for IoT devices to use it at disposal.

4.2.7 Conclusion

As discussed in the past sections, the blockchain approach and specifically smart contract capabilities have changed the aspect of supply chain future and payment for goods. These technologies have been safe enough until now but might not be so safe in the future.

We discussed the possible advantages of using a quantum computer to solve problems that could not be solved in the past in a very short time and how this becomes a threat for current DLTs. It is now the time in which quantum computers are actually rebooting computing and actions have to be taken now because the future may come sooner than we think. This is why some big enterprises and research centers are already trying to exploit these advantages riding this new wave.

As shown before, there are some companies which are offering the quantum-proof DLTs solutions based on quantum-resistant algorithms: QRL, Corda (R3), and IOTA. Moreover, some of them also provide Smart Contract capabilities which can be very effective for the presented use cases.

We foresee other possible alternatives will come out very soon to exploit the usefulness of blockchain and to tackle its threats.

Chapter 5

Conclusions and Outlook

This chapter summarizes the contributions of this thesis and delineates possible future research topics.

5.1 Contributions

We are living in the era of the Internet of Things, with smart devices producing large amounts of data that can be exploited in decision-making policies, to implement the Smart Societies concept. This thesis aims at tackling several issues that affect the design and implementation of an IoT system: the correctness of the data produced; the possibility of reducing power consumption; transmitting the data in the most suited way; authenticating the transmitted data and storing them in a safe way.

Such issues have been carefully addressed to draw design lines that must be fulfilled to implement a scientifically sound system for the Smart Cities of today and tomorrow.

The main contributions of this thesis are:

- **Ensuring scientifically sound air pollution data (Chapter 2)**
  Air pollution has reached the peak of attention in the last decades, due to the risk it poses for the environment and the human race. Historically, air pollution has been monitored by a sparse network of high precision, high cost devices. Fine-grained maps could be crucial in understanding the dynamics of such phenomena. This chapter presents the design, implementation, and thorough test of a low-cost mobile Particulate Matter sensing station. Such a system went through an extensive data acquisition campaign near a reference-grade sensor whose values were used to evaluate different calibration techniques. The system proved itself extremely useful in several real use case scenarios in situations characterized by rapid time and space variability of the air pollutants.

The system has also been evaluated from the perspective of varying the duty cycle of operation with the aim of reducing the power consumption, the data logged, and to extend the life of sensors.

- **Evaluating the best wireless protocol for the transmission of IoT data (Chapter 3)**
  This topic has been carried out in the home automation domain of HVAC (Heating, Ventilation, and Air Conditioning). The analysis was focused on trying to understand which is the best transmission protocol for the task of indoor comfort monitoring. To this end, two widespread technologies: RFID (Radio Frequency IDentification) and Bluetooth. These two technologies have been carefully compared not only theoretically, but also by means of extensive testing to assess their operating range, scalability, and resilience to interference. The analysis shows that there is no a priori best technology, but the choice between them greatly depends on the characteristics of the desired system. To the best of our knowledge, this is the first comparison between these two technologies in such depth.

- **Implementation of data authenticity and safe storage for the IoT (Chapter 4.1)**
  This work presents the integration of two disruptive technologies: IoT and Blockchain. This paves the way for IoT devices to work in synergy with decentralized applications running on top of public and open blockchain. The *blockchain-enabled gateway* proposed features the necessary hardware and software requirements to directly sign transactions to be stored in the chain. This guarantees the authenticity and immutability of the data. This step is a leap towards the transparency of data in sensitive domains, without the need for intervention of a third-party entity.

- **Analysis of new possibilities and threats posed by Quantum Computing (Chapter 4.2)**
  Started as a side topic, the new paradigm of Quantum Computing grew deep in our interest. In this work an analysis of this new computing paradigm are reported, with an analysis of possible applications and the threat that QC poses for the security as we know it. The last topic of this thesis is the analysis of the newly found vulnerabilities of Distributed Ledger Technologies (such as the blockchain). Attacking scenarios are explained and professed quantum-proof solutions already implemented are reported.

5.2 Possible future research topics

Given the wide spectrum of the discussed topics, the possibilities for improvements in this field are many. Here are reported some interesting ones:

- **Analysis of sensor redundancy**
  In the air pollution monitoring IoT system developed in this work are included 4 instances of a PM sensor. Such redundancy could be exploited in a way that it would be possible to identify when a sensor is misbehaving for a certain amount of time or irreparably broken. Another option could be to schedule alternative operating times to prolong the life of the overall system.

- **Implementation of PM participate sensing**
  It is possible to envision a deployment strategy with a revised version of the sensing board implementing Bluetooth and LoRa connectivity capabilities. Such devices could be deployed throughout the territory of a city, and leverage the Bluetooth connectivity of smartphones of supportive city dwellers to upload the data to a remote server. The LoRa connectivity capability could be exploited once in a while, given its limitations, to transmit the status of the board, and not the data itself.

- **Leveraging Quantum Computing for air pollution monitoring activities**
  The potential of QC could be exploited in activities which are already of our interest. A possible interesting topic in this field could be the evaluation of a quantum approach to the task of calibration of PM sensors. This task could be carried out by means of quantum machine learning techniques on gate-array or quantum annealing machines. This could potentially lead to a decrease in the time required to perform the calibration of sensors and improve their usability.

Chapter 6

Publications

Here are reported publications, grouped by genre, produced throughout the PhD degree at Politecnico di Torino.

6.1 Journal Papers

- E. Giusto, F. Gandino, M. Greco, M. Grosso, B. Montrucchio, and S. Rinaudo, “An Investigation on Pervasive Technologies for IoT-based Thermal Monitoring,” Sensors, vol. 19, no. 3, p. 663, Feb. 2019.
https://doi.org/10.3390/s19030663 [82]

- B. Montrucchio, E. Giusto, M. G. Vakili, S. Quer, R. Ferrero and C. Fornaro, "A Densely-Deployed, High Sampling Rate, Open-Source Air Pollution Monitoring WSN," in IEEE Transactions on Vehicular Technology, vol. 69, no. 12, pp. 15786-15799, Dec. 2020,
doi: 10.1109/TVT.2020.3035554.[19]

- J. Grecuccio, E. Giusto, F. Fiori, and M. Rebaudengo, “Combining Blockchain and IoT: Food-Chain Traceability and Beyond,” Energies, vol. 13, no. 15, p. 3820, Jul. 2020.
https://doi.org/10.3390/en13153820 [117]

- M. Collotta, R. Ferrero, E. Giusto, M. Ghazi Vakili, J. Grecuccio, X. Kong, I. You. A fuzzy control system for energy-efficient wireless devices in the Internet of vehicles. Int J Intell Syst. 2021; 36: 1595– 1618.
https://doi.org/10.1002/int.22353 [167]

- E. Giusto, M. G. Vakili, F. Gandino, C. Demartini and B. Montrucchio, "Quantum Pliers Cutting the Blockchain," in IT Professional, vol. 22, no. 6, pp. 90-96, 1 Nov.-Dec. 2020,
doi: 10.1109/MITP.2020.2974690. [118]

6.2 Conference Papers

- E. Giusto, F. Gandino, M. L. Greco, M. Rebaudengo and B. Montrucchio, "A dense RFID network for flexible Thermal Monitoring," 2018 6th International EURASIP Workshop on RFID Technology (EURFID), 2018, pp. 1-7, doi: 10.1109/EURFID.2018.8611649.[168]

- E. Giusto, R. Ferrero, F. Gandino, B. Montrucchio, M. Rebaudengo and M. Zhang, "Particulate Matter Monitoring in Mixed Indoor/Outdoor Industrial Applications: A Case Study," 2018 IEEE 23rd International Conference on Emerging Technologies and Factory Automation (ETFA), 2018, pp. 838-844, doi: 10.1109/ETFA.2018.8502644.[169]

- R. Ferrero, M. G. Vakili, E. Giusto, M. Guerrera and V. Randazzo, "Ubiquitous fridge with natural language interaction," 2019 IEEE International Conference on RFID Technology and Applications (RFID-TA), 2019, pp. 404-409, doi: 10.1109/RFID-TA.2019.8892025.[170]

- O. Costa Hamido, G. A. Cirillo, E. Giusto. 2020. Quantum synth: a quantum-computing-based synthesizer. In Proceedings of the 15th International Conference on Audio Mostly (AM '20). Association for Computing Machinery, New York, NY, USA, 265–268.
DOI: https://doi.org/10.1145/3411109.3411135 [16]

6.3 Datasets

- B. Montrucchio, E. Giusto, M. Ghazi Vakili, S. Quer, R. Ferrero, C. Fornaro, August 17, 2020, "A Densely-Deployed, High Sampling Rate, Open-Source Air Pollution Monitoring WSN", IEEE Dataport,
doi: https://dx.doi.org/10.21227/m4pb-g538. [17]

Bibliography

[1] I-SCOOP, *Industry 4.0: the fourth industrial revolution–guide to industrie 4.0*, 2017.

[2] K. Ashton, "That Internet of Things Thing", *RFID Journal*, 2009.

[3] H. Sundmaeker, P. Guillemin, P. Friess, and S. Woelfflé, "Vision and challenges for realising the Internet of Things", *Cluster of European research projects on the internet of things, European Commision*, vol. 3, no. 3, pp. 34–36, 2010.

[4] C. E. Shannon, "Communication in the Presence of Noise", *Proceedings of the IRE*, vol. 37, no. 1, pp. 10–21, 1949. DOI: 10.1109/JRPROC.1949.232969.

[5] G. B. Hamra, N. Guha, A. Cohen, F. Laden, O. Raaschou-Nielsen, J. M. Samet, P. Vineis, F. Forastiere, P. Saldiva, T. Yorifuji, *et al.*, "Outdoor particulate matter exposure and lung cancer: a systematic review and meta-analysis", *Environmental health perspectives*, 2014.

[6] X. Wu, R. C. Nethery, M. B. Sabath, D. Braun, and F. Dominici, "Air pollution and COVID-19 mortality in the United States: Strengths and limitations of an ecological regression analysis", *Science Advances*, vol. 6, no. 45, Nov. 2020, ISSN: 23752548. DOI: 10.1126/SCIADV.ABD4049. [Online]. Available: /pmc/articles/PMC7673673/%20/pmc/articles/PMC7673673/?report=abstract%20https://www.ncbi.nlm.nih.gov/pmc/articles/PMC7673673/.

[7] *EUR-Lex - 02008L0050-20150918 - EN - EUR-Lex.* [Online]. Available: https://eur-lex.europa.eu/legal-content/EN/ALL/?uri=CELEX%5C%3A32008L0050 (visited on 05/05/2021).

[8] S. Nakamoto, "Bitcoin: A Peer-to-Peer Electronic Cash System", Tech. Rep. [Online]. Available: www.bitcoin.org.

[9] J. Frizzo-barker, P. A. Chow-white, P. R. Adams, J. Mentanko, D. Ha, and S. Green, "Blockchain as a disruptive technology for business : A systematic review", *International Journal of Information Management*, no. April, 2019, ISSN: 0268-4012. DOI: 10.1016/j.ijinfomgt.2019.10.014. [Online]. Available: https://doi.org/10.1016/j.ijinfomgt.2019.10.014.

[10] F. Tian, "An agri-food supply chain traceability system for China based on RFID & blockchain technology", in *2016 13th international conference on service systems and service management (ICSSSM)*, IEEE, 2016, pp. 1–6.

[11] M. A. Nielsen and I. L. Chuang, *Quantum Computation and Quantum Information: 10th Anniversary Edition*. Cambridge University Press, 2010. DOI: `10.1017/CBO9780511976667`.

[12] A. Barenco, C. H. Bennett, R. Cleve, D. P. DiVincenzo, N. Margolus, P. Shor, T. Sleator, J. A. Smolin, and H. Weinfurter, "Elementary gates for quantum computation", *Phys. Rev. A*, vol. 52, pp. 3457–3467, 5 Nov. 1995. DOI: `10.1103/PhysRevA.52.3457`. [Online]. Available: `https://link.aps.org/doi/10.1103/PhysRevA.52.3457`.

[13] L. K. Grover, "A Fast Quantum Mechanical Algorithm for Database Search", in *Proceedings of the Twenty-eighth Annual ACM Symposium on Theory of Computing*, ser. STOC '96, Philadelphia, Pennsylvania, USA: ACM, 1996, pp. 212–219, ISBN: 0-89791-785-5. DOI: `10.1145/237814.237866`. [Online]. Available: `http://doi.acm.org/10.1145/237814.237866`.

[14] P. W. Shor, "Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer", *SIAM review*, vol. 41, no. 2, pp. 303–332, 1999.

[15] N. Nagy, M. Nagy, and S. G. Akl, "Quantum wireless sensor networks", in *International Conference on Unconventional Computation*, Springer, 2008, pp. 177–188.

[16] O. C. Hamido, G. A. Cirillo, and E. Giusto, "Quantum synth: a quantum-computing-based synthesizer", in *Proceedings of the 15th International Conference on Audio Mostly*, 2020, pp. 265–268.

[17] B. Montrucchio, E. Giusto, M. Ghazi Vakili, S. Quer, R. Ferrero, and C. Fornaro, *A Densely-Deployed, High Sampling Rate, Open-Source Air Pollution Monitoring WSN | IEEE DataPort*. [Online]. Available: `https://dx.doi.org/10.21227/m4pb-g538` (visited on 05/07/2021).

[18] E. Giusto, R. Ferrero, F. Gandino, B. Montrucchio, M. Rebaudengo, and M. Zhang, "Particulate Matter Monitoring in Mixed Indoor/Outdoor Industrial Applications: A Case Study", in *IEEE International Conference on Emerging Technologies and Factory Automation, ETFA*, vol. 2018-Septe, Institute of Electrical and Electronics Engineers Inc., 2018, pp. 838–844, ISBN: 9781538671085. DOI: `10.1109/ETFA.2018.8502644`.

[19] B. Montrucchio, E. Giusto, M. Ghazi Vakili, S. Quer, R. Ferrero, and C. Fornaro, "A Densely-Deployed, High Sampling Rate, Open-Source Air Pollution Monitoring WSN", *IEEE Transactions on Vehicular Technology*, pp. 1–1, 2020, ISSN: 0018-9545. DOI: `10.1109/TVT.2020.3035554`. [Online]. Available: `https://ieeexplore.ieee.org/document/9247550/`.

[20] N. Castell, F. R. Dauge, P. Schneider, M. Vogt, U. Lerner, B. Fishbain, D. Broday, and A. Bartonova, "Can commercial low-cost sensor platforms contribute to air quality monitoring and exposure estimates?", *Environment International*, vol. 99, pp. 293–302, 2017, ISSN: 0160-4120. DOI: `https://doi.org/10.1016/j.envint.2016.12.007`. [Online]. Available: `https://www.sciencedirect.com/science/article/pii/S0160412016309989`.

[21] "European Union Tech Report: Measuring air pollution with low-cost sensors. Thoughts on the quality of data measured by sensors", Tech. Rep. [Online]. Available: `https://ec.europa.eu/environment/air/pdf/Brochure%20lower-cost%20sensors.pdf`.

[22] R. Tse, D. Aguiari, L. Monti, G. Pau, C. Prandi, and P. Salomoni, "On assessing the accuracy of air pollution models exploiting a strategic sensors deployment", in *ACM International Conference Proceeding Series*, New York, New York, USA: Association for Computing Machinery, Nov. 2018, pp. 55–58, ISBN: 9781450365819. DOI: `10.1145/3284869.3284880`. [Online]. Available: `http://dl.acm.org/citation.cfm?doid=3284869.3284880`.

[23] R. Tse, D. Aguiari, K.-S. Chou, S.-K. Tang, D. Giusto, and G. Pau, "Monitoring cultural heritage buildings via low-cost edge computing/sensing platforms: the Biblioteca Joanina de Coimbra case study", in *Proceedings of the 4th EAI International Conference on Smart Objects and Technologies for Social Good*, 2018, pp. 148–152.

[24] R. Tse, M. Im, S.-K. Tang, L. Menezes, A. Dias, and G. Pau, "Self-adaptive Sensing IoT Platform for Conserving Historic Buildings and Collections in Museums", in *Proceedings of the 5th International Conference on Internet of Things, Big Data and Security*, SCITEPRESS - Science and Technology Publications, May 2020, pp. 392–398, ISBN: 978-989-758-426-8. DOI: `10.5220/0009470203920398`. [Online]. Available: `http://www.scitepress.org/DigitalLibrary/Link.aspx?doi=10.5220/0009470203920398`.

[25] D. J. Pagliari, M. Poncino, and E. Macii, "Energy-efficient digital processing via approximate computing", in *Smart Systems Integration and Simulation*, 2016, ISBN: 9783319273921. DOI: `10.1007/978-3-319-27392-1{\_}4`.

[26] Y. Chen, D. Jahier Pagliari, E. Macii, and M. Poncino, "Battery-aware design exploration of scheduling policies for multi-sensor devices", in *Proceedings of the ACM Great Lakes Symposium on VLSI, GLSVLSI*, 2018, ISBN: 9781450357241. DOI: `10.1145/3194554.3194588`.

[27] A. Bigi, M. Mueller, S. K. Grange, G. Ghermandi, and C. Hueglin, "Performance of NO, NO 2 low cost sensors and three calibration approaches within a real world application", *Atmos. Meas. Tech*, vol. 11, pp. 3717–3735, 2018. DOI: 10.5194/amt-11-3717-2018. [Online]. Available: https://doi.org/10.5194/amt-11-3717-2018.

[28] N. Zimmerman, A. A. Presto, S. P. N. Kumar, J. Gu, A. Hauryliuk, E. S. Robinson, A. L. Robinson, and R. Subramanian, "A machine learning calibration model using random forests to improve sensor performance for lower-cost air quality monitoring", *Atmos. Meas. Tech*, vol. 11, pp. 291–313, 2018. DOI: 10.5194/amt-11-291-2018. [Online]. Available: https://doi.org/10.5194/amt-11-291-2018.

[29] R. V. Martin, "Satellite remote sensing of surface air quality", DOI: 10.1016/j.atmosenv.2008.07.018. [Online]. Available: http://www.acd..

[30] Y. Kawamoto, H. Nishiyama, Z. M. Fadlullah, and N. Kato, "Effective data collection via satellite-routed sensor system ({\{}SRSS{\}}) to realize global-scaled {\{}I{\}}nternet of {\{}T{\}}hings", *IEEE Sensors Journal*, vol. 13, no. 10, pp. 3645–3654, 2013.

[31] M. Budde, R. El Masri, T. Riedel, and M. Beigl, "Enabling low-cost particulate matter measurement for participatory sensing scenarios", *Proceedings of the 12th International Conference on Mobile and Ubiquitous Multimedia (MUM)*, pp. 1–10, 2013. DOI: 10.1145/2541831.2541859.

[32] M. Sammarco, R. Tse, G. Pau, and G. Marfia, "Using geosocial search for urban air pollution monitoring", *Pervasive and Mobile Computing*, vol. 35, pp. 15–31, Feb. 2017, ISSN: 15741192. DOI: 10.1016/j.pmcj.2016.07.001.

[33] Y. C. Wang and G. W. Chen, "Efficient Data Gathering and Estimation for Metropolitan Air Quality Monitoring by Using Vehicular Sensor Networks", *IEEE Transactions on Vehicular Technology*, 2017, ISSN: 00189545. DOI: 10.1109/TVT.2017.2655084.

[34] D. Takaishi, H. Nishiyama, N. Kato, and R. Miura, "Toward energy efficient big data gathering in densely distributed sensor networks", *IEEE transactions on emerging topics in computing*, vol. 2, no. 3, pp. 388–397, 2014.

[35] J. Peters, J. Theunis, M. Van Poppel, and P. Berghmans, "Monitoring PM 10 and Ultrafine Particles in Urban Environments Using Mobile Measurements", *Aerosol and Air Quality Research*, vol. 13, pp. 509–522, 2013. DOI: 10.4209/aaqr.2012.06.0152.

[36] D. Aguiari, G. Delnevo, L. Monti, V. Ghini, S. Mirri, P. Salomoni, G. Pau, M. Im, R. Tse, M. Ekpanyapong, and R. Battistini, “Canarin II: Designing a smart e-bike eco-system”, *CCNC 2018 - 2018 15th IEEE Annual Consumer Communications and Networking Conference*, vol. 2018-Janua, no. 762013, pp. 1–6, Mar. 2018. DOI: `10.1109/CCNC.2018.8319221`.

[37] L. Yang, W. Li, M. Ghandehari, and G. Fortino, “People-centric cognitive internet of things for the quantitative analysis of environmental exposure”, *IEEE Internet of Things Journal*, vol. 5, no. 4, pp. 2353–2366, 2018, ISSN: 23274662. DOI: `10.1109/JIOT.2017.2751307`.

[38] P. Kumar, L. Morawska, C. Martani, G. Biskos, M. Neophytou, S. Di Sabatino, M. Bell, L. Norford, and R. Britter, “The rise of low-cost sensing for managing air pollution in cities”, *Environment International*, vol. 75, pp. 199–205, 2015. DOI: `10.1016/j.envint.2014.11.019`.

[39] J. Li, H. Li, Y. Ma, Y. Wang, A. A. Abokifa, C. Lu, and P. Biswas, “Spatiotemporal distribution of indoor particulate matter concentration with a low-cost sensor network”, *Building and Environment*, 2018, ISSN: 03601323. DOI: `10.1016/j.buildenv.2017.11.001`.

[40] L. J. Chen, Y. H. Ho, H. H. Hsieh, S. T. Huang, H. C. Lee, and S. Mahajan, “{ADF}: An Anomaly Detection Framework for Large-Scale {PM2.5} Sensing Systems”, *IEEE Internet of Things Journal*, vol. 5, no. 2, pp. 559–570, 2017, ISSN: 23274662. DOI: `10.1109/JIOT.2017.2766085`.

[41] *US EPA Air SensorsGuidebook*. [Online]. Available: `https://www.epa.gov/air-sensor-toolbox/how-use-air-sensors-air-sensor-guidebook`.

[42] S. Sousan, A. Gray, C. Zuidema, L. Stebounova, G. Thomas, K. A. Koehler, and T. Peters, “Sensor selection to improve estimates of particulate matter concentration from a low-cost network”, *Sensors*, vol. 18, no. 9, 2018, ISSN: 1424-3210. DOI: `10.3390/s18093008`.

[43] D. Hasenfratz, O. Saukh, S. Sturzenegger, and L. Thiele, “Participatory air pollution monitoring using smartphones”, *Proc. 1st Int’l Workshop on Mobile Sensing: From Smartphones and Wearables to Big Data*, pp. 1–5, 2012. [Online]. Available: `ftp://ftp.tik.ee.ethz.ch/pub/people/hdavid/HSST2012.pdf`.

[44] D. Hasenfratz, O. Saukh, C. Walser, C. Hueglin, M. Fierz, and L. Thiele, “Pushing the spatio-temporal resolution limit of urban air pollution maps”, in *2014 IEEE International Conference on Pervasive Computing and Communications, PerCom 2014*, 2014. DOI: `10.1109/PerCom.2014.6813946`.

[45] Y. Cheng, X. Li, Z. Li, S. Jiang, Y. Li, J. Jia, and X. Jiang, "AirCloud: a cloud-based air-quality monitoring system for everyone", *Proceedings of the 12th ACM Conference on Embedded Network Sensor Systems - SenSys '14*, pp. 251–265, 2014. DOI: 10.1145/2668332.2668346. [Online]. Available: http://dl.acm.org/citation.cfm?doid=2668332.2668346.

[46] Y. Gao, W. Dong, K. Guo, X. Liu, Y. Chen, X. Liu, J. Bu, and C. Chen, "Mosaic: A low-cost mobile sensing system for urban air quality monitoring", *Proceedings - IEEE INFOCOM*, vol. 2016-July, pp. 1–9, 2016, ISSN: 0743166X. DOI: 10.1109/INFOCOM.2016.7524478.

[47] *Raspberry Pi.* [Online]. Available: https://www.raspberrypi.org/.

[48] *Raspberry Pi Zero W.* [Online]. Available: https://www.raspberrypi.org/products/raspberry-pi-zero-w/.

[49] *pool.ntp.org.* [Online]. Available: https://www.pool.ntp.org/en/.

[50] *pigpio library.* [Online]. Available: http://abyz.me.uk/rpi/pigpio/index.html.

[51] *HPMA115S0 Particulate Matter Sensors - Honeywell.* [Online]. Available: https://sensing.honeywell.com/hpma115s0-particulate-matter-sensors.

[52] T. Liu, "DHT22 humidity and temperature module/sensor", Tech. Rep. [Online]. Available: https://www.sparkfun.com/datasheets/Sensors/Temperature/DHT22.pdf.

[53] *DHT22 library for Raspberry Pi.* [Online]. Available: https://github.com/technion/loldht22.

[54] *BME280.* [Online]. Available: https://www.bosch-sensortec.com/bst/products/allproducts/bme280.

[55] *API Reference — scikit-learn 0.21.3 documentation.* [Online]. Available: https://scikit-learn.org/stable/modules/classes.html#module-sklearn.metrics.

[56] X. Wang, X. Shen, J. Sun, X. Zhang, Y. Wang, Y. Zhang, P. Wang, C. Xia, X. Qi, and J. Zhong, "Size-resolved hygroscopic behavior of atmospheric aerosols during heavy aerosol pollution episodes in Beijing in December 2016", *Atmospheric Environment*, vol. 194, pp. 188–197, 2018, ISSN: 1352-2310. DOI: https://doi.org/10.1016/j.atmosenv.2018.09.041. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1352231018306411.

[57] A. Chaloulakou, P. Kassomenos, N. Spyrellis, P. Demokritou, and P. Koutrakis, "Measurements of PM10 and PM2. 5 particle concentrations in Athens, Greece", *Atmospheric Environment*, vol. 37, no. 5, pp. 649–660, 2003.

[58] M. Caselli, L. Trizio, G. De Gennaro, and P. Ielpo, "A Simple Feedforward Neural Network for the PM 10 Forecasting: Comparison with a Radial Basis Function Network and a Multivariate Linear Regression Model", *Water Air and Soil Pollution - WATER AIR SOIL POLLUT*, vol. 201, pp. 365–377, 2009. DOI: 10.1007/s11270-008-9950-2.

[59] K. Polat and S. Durduran, "Usage of output-dependent data scaling in modeling and prediction of air pollution daily concentration values (PM$_10$) in the city of Konya", *Neural Computing and Applications - NCA*, vol. 21, 2011. DOI: 10.1007/s00521-011-0661-z.

[60] M. S. Alam and A. McNabola, "Exploring the modeling of spatiotemporal variations in ambient air pollution within the land use reression framework: Estimation of PM{\$}{_}{\{}10{\}}{\$} concentrations on a daily basis", *Journal of the Air {\&} Waste Management Association*, vol. 65, no. 5, pp. 628–640, 2015. DOI: 10.1080/10962247.2015.1006377. [Online]. Available: https://doi.org/10.1080/10962247.2015.1006377.

[61] J. Jolliff, J. Kindle, I. Shulman, B. Penta, M. Friedrichs, R. Helber, and R. Arnone, "Summary diagrams for coupled hydrodynamic-ecosystem model skill assessment", *Journal of Marine Systems*, vol. 76, pp. 64–82, 2009. DOI: 10.1016/j.jmarsys.2008.05.014.

[62] C. Li and S. Shimamoto, "An Open Traffic Light Control Model for Reducing Vehicles' {CO2} Emissions Based on ETC Vehicles", *IEEE transactions on vehicular technology*, vol. 61, no. 1, pp. 97–110, 2011.

[63] J. Gulliver and D. Briggs, "Personal exposure to particulate air pollution in transport microenvironments", *Atmospheric Environment*, vol. 38, no. 1, pp. 1–8, 2004.

[64] A. Dobre, S. J. Arnold, R. J. Smalley, J. W. Boddy, J. F. Barlow, A. S. Tomlin, and S. E. Belcher, "Flow field measurements in the proximity of an urban intersection in London, UK", *Atmospheric Environment*, vol. 39, no. 26, pp. 4647–4657, 2005, ISSN: 13522310. DOI: 10.1016/j.atmosenv.2005.04.015.

[65] M. Jovašević-Stojanović, A. Bartonova, D. Topalović, I. Lazović, B. Pokrić, and Z. Ristovski, "On the use of small and cheaper sensors and devices for indicative citizen-based monitoring of respirable particulate matter", *Environmental Pollution*, vol. 206, pp. 696–704, 2015, ISSN: 18736424. DOI: 10.1016/j.envpol.2015.08.035.

[66] G. Anastasi, M. Conti, M. Di Francesco, and A. Passarella, "Energy conservation in wireless sensor networks: A survey", *Ad Hoc Networks*, vol. 7, no. 3, pp. 537–568, May 2009, ISSN: 15708705. DOI: 10.1016/j.adhoc.2008.06.003.

[67] F. Engmann, F. A. Katsriku, J. D. Abdulai, K. S. Adu-Manu, and F. K. Banaseka, “Prolonging the Lifetime of Wireless Sensor Networks: A Review of Current Techniques”, *Wireless Communications and Mobile Computing*, vol. 2018, 2018, ISSN: 15308677. DOI: 10.1155/2018/8035065. [Online]. Available: https://doi.org/10.1155/2018/8035065.

[68] L. Morawska, P. K. Thai, X. Liu, A. Asumadu-Sakyi, G. Ayoko, A. Bartonova, A. Bedini, F. Chai, B. Christensen, M. Dunbabin, J. Gao, G. S. W. Hagler, R. Jayaratne, P. Kumar, A. K. H. Lau, P. K. K. Louie, M. Mazaheri, Z. Ning, N. Motta, B. Mullins, M. M. Rahman, Z. Ristovski, M. Shafiei, D. Tjondronegoro, D. Westerdahl, and R. Williams, “Applications of low-cost sensing technologies for air quality monitoring and exposure assessment: How far have they gone?”, *Environment International*, vol. 116, no. April, pp. 286–299, 2018, ISSN: 18736750. DOI: 10.1016/j.envint.2018.04.018.

[69] A. C. Rai, P. Kumar, F. Pilla, A. N. Skouloudis, S. Di Sabatino, C. Ratti, A. Yasar, and D. Rickerby, “End-user perspective of low-cost sensors for outdoor air pollution monitoring”, *Science of the Total Environment*, vol. 607-608, pp. 691–705, 2017, ISSN: 18791026. DOI: 10.1016/j.scitotenv.2017.06.266. [Online]. Available: http://dx.doi.org/10.1016/j.scitotenv.2017.06.266.

[70] E. Austin, I. Novosselov, E. Seto, and M. G. Yost, “Laboratory evaluation of the Shinyei PPD42NS low-cost particulate matter sensor”, *PLoS ONE*, vol. 10, no. 9, 2015, ISSN: 19326203. DOI: 10.1371/journal.pone.0137789.

[71] D. Aguiari, G. Delnevo, L. Monti, V. Ghini, S. Mirri, P. Salomoni, G. Pau, M. Im, R. Tse, M. Ekpanyapong, and R. Battistini, “Canarin II: Designing a smart e-bike eco-system”, *CCNC 2018 - 2018 15th IEEE Annual Consumer Communications and Networking Conference*, vol. 2018-Janua, no. 762013, pp. 1–6, 2018. DOI: 10.1109/CCNC.2018.8319221.

[72] B. Dessimond, I. Annesi-Maesano, J. L. Pepin, S. Srairi, and G. Pau, “Academically produced air pollution sensors for personal exposure assessment: The canarin project”, *Sensors*, vol. 21, no. 5, pp. 1–18, Mar. 2021, ISSN: 14248220. DOI: 10.3390/s21051876. [Online]. Available: https://www.mdpi.com/1424-8220/21/5/1876.

[73] V. Jelicic, M. Magno, D. Brunelli, G. Paci, and L. Benini, “Context-adaptive multimodal wireless sensor network for energy-efficient gas monitoring”, *IEEE Sensors Journal*, vol. 13, no. 1, pp. 328–338, 2013, ISSN: 1530437X. DOI: 10.1109/JSEN.2012.2215733.

[74] S. Mansour, N. Nasser, L. Karim, and A. Ali, “Wireless sensor network-based air quality monitoring system”, in *2014 International Conference on Computing, Networking and Communications, ICNC 2014*, IEEE Computer Society, 2014, pp. 545–550. DOI: 10.1109/ICCNC.2014.6785394.

[75] Y. Gao, W. Dong, K. Guo, X. Liu, Y. Chen, X. Liu, J. Bu, and C. Chen, "Mosaic: A low-cost mobile sensing system for urban air quality monitoring", *Proceedings - IEEE INFOCOM*, vol. 2016-July, pp. 1–9, 2016, ISSN: 0743166X. DOI: 10.1109/INFOCOM.2016.7524478.

[76] K. K. Khedo, R. Perseedoss, A. Mungur, U. o. Mauritius, and Mauritius, "A Wireless Sensor Network Air Pollution Monitoring System", *International Journal of Wireless & Mobile Networks*, vol. 2, no. 2, pp. 31–45, May 2010. DOI: 10.5121/ijwmn.2010.2203. [Online]. Available: http://arxiv.org/abs/1005.1737%20http://dx.doi.org/10.5121/ijwmn.2010.2203.

[77] M. R. Chowdhury, S. De, N. K. Shukla, and R. N. Biswas, "Energy-Efficient Air Pollution Monitoring with Optimum Duty-Cycling on a Sensor Hub", in *2018 24th National Conference on Communications, NCC 2018*, Institute of Electrical and Electronics Engineers Inc., Jan. 2019, ISBN: 9781538612248. DOI: 10.1109/NCC.2018.8600133.

[78] J. Botero-Valencia, L. Castano-Londono, D. Marquez-Viloria, and M. Rico-Garcia, "Data reduction in a low-cost environmental monitoring system based on LoRa for WSN", *IEEE Internet of Things Journal*, vol. 6, no. 2, pp. 3024–3030, Apr. 2019, ISSN: 23274662. DOI: 10.1109/JIOT.2018.2878528.

[79] *FiPy - Pycom -Five Network Development Board for IoT*. [Online]. Available: https://pycom.io/product/fipy/.

[80] *The Internet of Things with ESP32*. [Online]. Available: http://esp32.net/.

[81] *MicroPython - Python for microcontrollers*. [Online]. Available: https://micropython.org/.

[82] E. Giusto, F. Gandino, M. L. Greco, M. Grosso, B. Montrucchio, and S. Rinaudo, "An investigation on pervasive technologies for IoT-based thermal monitoring", *Sensors*, vol. 19, no. 3, p. 663, 2019.

[83] E. Giusto, F. Gandino, M. L. Greco, M. Rebaudengo, and B. Montrucchio, "A dense RFID network for flexible Thermal Monitoring", in *6th International EURASIP Workshop on RFID Technology*, Sep. 2018, pp. 1–7.

[84] B. Benadda, B. Beldjilali, A. Mankouri, and O. Taleb, "Secure IoT solution for wearable health care applications, case study Electric Imp development platform", *International Journal of Communication Systems*, vol. 31, no. 5, e3499, DOI: 10.1002/dac.3499.

[85] M. Hemmatpour, R. Ferrero, F. Gandino, B. Montrucchio, and M. Rebaudengo, "Internet of Things for fall prediction and prevention", *Journal of Computational Methods in Sciences and Engineering*, vol. 18, no. 2, pp. 511–518, 2018.

[86] Y. Jeong, S. Son, E. Jeong, and B. Lee, "An Integrated Self-Diagnosis System for an Autonomous Vehicle Based on an IoT Gateway and Deep Learning", *Applied Sciences*, vol. 8, no. 7, 2018, ISSN: 2076-3417.

[87] Z. Idrees, Z. Zou, and L. Zheng, "Edge Computing Based IoT Architecture for Low Cost Air Pollution Monitoring Systems: A Comprehensive System Analysis, Design Considerations &amp; Development", *Sensors*, vol. 18, no. 9, 2018, ISSN: 1424-8220.

[88] B. Esakki, S. Ganesan, S. Mathiyazhagan, K. Ramasubramanian, B. Gnanasekaran, B. Son, S. W. Park, and J. S. Choi, "Design of Amphibious Vehicle for Unmanned Mission in Water Quality Monitoring Using Internet of Things", *Sensors*, vol. 18, no. 10, 2018, ISSN: 1424-8220.

[89] S. Kim, J. Cho, and D. Park, "Accelerated DEVS Simulation Using Collaborative Computation on Multi-Cores and GPUs for Fire-Spreading IoT Sensing Applications", *Applied Sciences*, vol. 8, no. 9, 2018, ISSN: 2076-3417.

[90] M. Alaa, A. A. Zaidan, B. B. Zaidan, M. Talal, and M. L. M. Kiah, "A review of smart home applications based on Internet of Things", *Journal of Network and Computer Applications*, vol. 97, pp. 48–65, 2017, ISSN: 1084-8045.

[91] N. N. W. Tay, J. Botzheim, and N. Kubota, "Human-Centric Automation and Optimization for Smart Homes", *IEEE Transactions on Automation Science and Engineering*, pp. 1–13, ISSN: 1545-5955. DOI: `10.1109/TASE.2018.2789658`.

[92] "Thermal environmental conditions for human occupancy", *ANSI/ASHRAE Standard 55-2013*, 2013.

[93] T. V. Chien, H. N. Chan, and T. N. Huu, "A Comparative Study on Hardware Platforms for Wireless Sensor Networks", *International Journal on Advanced Science, Engineering and Information Technology*, 2012, ISSN: 2088-5334. DOI: `10.18517/ijaseit.2.1.157`.

[94] A. Guerrieri, G. Fortino, and W. Russo, "An evaluation framework for buildings-oriented wireless sensor networks", in *Proceedings - 14th IEEE/ACM International Symposium on Cluster, Cloud, and Grid Computing, CCGrid 2014*, 2014, ISBN: 9781479927838. DOI: `10.1109/CCGrid.2014.99`.

[95] M. V. Bueno-Delgado, P. Pavón-Marino, A. De-Gea-García, and A. Dolón-García, "The Smart University Experience: An NFC-Based Ubiquitous Environment", in *2012 Sixth International Conference on Innovative Mobile and Internet Services in Ubiquitous Computing*, Jul. 2012, pp. 799–804. DOI: `10.1109/IMIS.2012.110`.

[96] B.-C. Chen, C.-T. Yang, H.-T. Yeh, and C.-C. Lin, "Mutual Authentication Protocol for Role-Based Access Control Using Mobile RFID", *Applied Sciences*, vol. 6, no. 8, 2016, ISSN: 2076-3417.

[97] F. Gandino, B. Montrucchio, and M. Rebaudengo, “A security protocol for RFID traceability”, *International Journal of Communication Systems*, vol. 30, no. 6, e3109, DOI: `10.1002/dac.3109`.

[98] M. Martínez Pérez, G. Vázquez González, and C. Dafonte, “Evaluation of a Tracking System for Patients and Mixed Intravenous Medication Based on RFID Technology”, *Sensors*, vol. 16, no. 12, 2016, ISSN: 1424-8220.

[99] J. Pardo, F. Zamora-Martínez, and P. Botella-Rocamora, “Online Learning Algorithm for Time Series Forecasting Suitable for Low Cost Wireless Sensor Networks Nodes”, *Sensors*, vol. 15, no. 4, pp. 9277–9304, 2015, ISSN: 1424-8220.

[100] A. Ruano, S. Silva, H. Duarte, and P. M. Ferreira, “Wireless Sensors and IoT Platform for Intelligent HVAC Control”, *Applied Sciences*, vol. 8, no. 3, 2018, ISSN: 2076-3417.

[101] M. Benammar, A. Abdaoui, S. H. M. Ahmad, F. Touati, and A. Kadri, “A Modular IoT Platform for Real-Time Indoor Air Quality Monitoring”, *Sensors*, vol. 18, no. 2, 2018, ISSN: 1424-8220.

[102] F. Deng, Y. He, B. Li, L. Zhang, X. Wu, Z. Fu, and L. Zuo, “Design of an Embedded CMOS Temperature Sensor for Passive RFID Tag Chips”, *Sensors*, vol. 15, no. 5, pp. 11 442–11 453, 2015, ISSN: 1424-8220.

[103] Y. Liu, F. Deng, Y. He, B. Li, Z. Liang, and S. Zhou, “Novel Concrete Temperature Monitoring Method Based on an Embedded Passive RFID Sensor Tag”, *Sensors*, vol. 17, no. 7, 2017, ISSN: 1424-8220.

[104] R. Badia-Melis, L. Ruiz-Garcia, J. Garcia-Hierro, and J. I. R. Villalba, “Refrigerated Fruit Storage Monitoring Combining Two Different Wireless Sensing Technologies: RFID and WSN”, *Sensors*, vol. 15, no. 3, pp. 4781–4795, 2015, ISSN: 1424-8220.

[105] N. Li, G. Calis, and B. Becerik-Gerber, “Measuring and monitoring occupancy with an RFID based system for demand-driven HVAC operations”, *Automation in Construction*, vol. 24, pp. 89–99, 2012, ISSN: 0926-5805.

[106] A. Corna, L. Fontana, A. A. Nacci, and D. Sciuto, “Occupancy detection via iBeacon on Android devices for smart building management”, in *Proceedings -Design, Automation and Test in Europe, DATE*, 2015, ISBN: 9783981537048. DOI: `10.7873/date.2015.0753`.

[107] M. D’Aloia, F. Cortone, G. Cice, R. Russo, M. Rizzi, and A. Longo, “Improving energy efficiency in building system using a novel people localization system”, in *EESMS 2016 - 2016 IEEE Workshop on Environmental, Energy, and Structural Monitoring Systems, Proceedings*, 2016, ISBN: 9781509023691. DOI: `10.1109/EESMS.2016.7504811`.

[108] L. Leonardi, G. Patti, and L. Lo Bello, "Multi-Hop Real-Time Communications over Bluetooth Low Energy Industrial Wireless Mesh Networks", *IEEE Access*, 2018, ISSN: 21693536. DOI: 10.1109/ACCESS.2018.2834479.

[109] P. L. Monteiro, M. Zanin, E. M. Ruiz, J. Pimentão, and P. A. d. C. Sousa, "Indoor temperature prediction in an IoT scenario", *Sensors (Switzerland)*, 2018, ISSN: 14248220. DOI: 10.3390/s18113610.

[110] M. Grosso, S. Rinaudo, E. Patti, and A. Acquaviva, "An energy-autonomous wireless sensor network development platform", in *Proceedings - 2018 13th IEEE International Conference on Design and Technology of Integrated Systems In Nanoscale Era, DTIS 2018*, 2018, ISBN: 9781538652916. DOI: 10.1109/DTIS.2018.8368569.

[111] R. Ferrero, F. Gandino, B. Montrucchio, and M. Rebaudengo, "Experimental investigation on the interference between UHF RFID and GSM", in *2015 International EURASIP Workshop on RFID Technology (EURFID)*, Oct. 2015, pp. 140–143. DOI: 10.1109/EURFID.2015.7332399.

[112] L. Zhang, R. Ferrero, F. Gandino, and M. Rebaudengo, "Investigation of Interference Models for RFID Systems", *Sensors*, vol. 16, no. 2, 2016, ISSN: 1424-8220. [Online]. Available: http://www.mdpi.com/1424-8220/16/2/199.

[113] I. E. de Barros Filho, I. Silva, and C. M. Viegas, "An effective extension of anti-collision protocol for RFID in the industrial internet of things (IIoT)", *Sensors (Switzerland)*, 2018, ISSN: 14248220. DOI: 10.3390/s18124426.

[114] L. Arjona, H. L. Simon, and A. P. Ruiz, "Energy-aware RFID anti-collision protocol", *Sensors (Switzerland)*, 2018, ISSN: 14248220. DOI: 10.3390/s18061904.

[115] R. Ferrero, F. Gandino, B. Montrucchio, and M. Rebaudengo, "Improving Colorwave with the probabilistic approach for reader-to-reader anti-collision TDMA protocols", *Wireless Networks*, 2014, ISSN: 10220038. DOI: 10.1007/s11276-013-0611-z.

[116] W. Yoon and N. H. Vaidya, "RFID reader collision problem: Performance analysis and medium access", *Wireless Communications and Mobile Computing*, 2012, ISSN: 15308669. DOI: 10.1002/wcm.972.

[117] J. Grecuccio, E. Giusto, F. Fiori, and M. Rebaudengo, "Combining Blockchain and IoT: Food-Chain Traceability and Beyond", *Energies*, vol. 13, no. 15, p. 3820, 2020.

[118] E. Giusto, M. G. Vakili, F. Gandino, C. Demartini, and B. Montrucchio, "Quantum Pliers Cutting the Blockchain", *IT Professional*, vol. 22, no. 6, pp. 90–96, 2020.

[119] J. Lee, B. Bagheri, and H. A. Kao, “A Cyber-Physical Systems architecture for Industry 4.0-based manufacturing systems”, *Manufacturing Letters*, vol. 3, pp. 18–23, Jan. 2015, ISSN: 22138463. DOI: 10.1016/j.mfglet.2014.12.001.

[120] J. Lee, M. Azamfar, and J. Singh, “A blockchain enabled Cyber-Physical System architecture for Industry 4.0 manufacturing systems”, *Manufacturing Letters*, vol. 20, pp. 34–39, Apr. 2019, ISSN: 22138463. DOI: 10.1016/j.mfglet.2019.05.003.

[121] R. Ferrero, F. Gandino, B. Montrucchio, and M. Rebaudengo, “A cost-effective proposal for an RFID-based system for agri-food traceability”, *International Journal of Ad Hoc and Ubiquitous Computing*, vol. 27, no. 4, pp. 270–280, 2018, ISSN: 17438233. DOI: 10.1504/IJAHUC.2018.090598.

[122] L. Atzori, A. Iera, and G. Morabito, “The Internet of Things: A survey”, *Computer Networks*, vol. 54, no. 15, pp. 2787–2805, Oct. 2010, ISSN: 13891286. DOI: 10.1016/j.comnet.2010.05.010.

[123] A. Bosche, D. Crawford, D. Jackson, M. Schallehn, and C. Schorling, “Unlocking Opportunities in the Internet of Things - Bain &amp; Company”, *Bain & Company*, 2018. [Online]. Available: https://www.bain.com/insights/unlocking-opportunities-in-the-internet-of-things/.

[124] M. Salimitari and M. Chatterjee, “A Survey on Consensus Protocols in Blockchain for IoT Networks”, *ArXiv*, Sep. 2018. [Online]. Available: http://arxiv.org/abs/1809.05613.

[125] R. L. Rivest, A. Shamir, and L. Adleman, “A Method for Obtaining Digital Signatures and Public-Key Cryptosystems”, *Communications of the ACM*, 1978, ISSN: 15577317. DOI: 10.1145/359340.359342.

[126] A. K. Fedorov, E. O. Kiktenko, and A. I. Lvovsky, “Quantum computers put blockchain security at risk”, *Nature*, vol. 563, no. 7732, pp. 465–467, Nov. 2018, ISSN: 0028-0836. DOI: 10.1038/d41586-018-07449-z. [Online]. Available: http://www.nature.com/articles/d41586-018-07449-z.

[127] V. S. Miller, “Use of Elliptic Curves in Cryptography”, in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 1986, ISBN: 9783540164630. DOI: 10.1007/3-540-39799-X{_}31.

[128] N. Koblitz, “Elliptic Curve Cryptosystems”, Tech. Rep. 177, 1987, pp. 203–209.

[129] P. D. Gallagher and C. Romine, “FIPS PUB 186-4 Digital Signature Standard (DSS) CATEGORY: COMPUTER SECURITY SUBCATEGORY: CRYPTOGRAPHY”, 2013. DOI: 10.6028/NIST.FIPS.186-4. [Online]. Available: http://dx.doi.org/10.6028/NIST.FIPS.186-4.

[130] G. Wood, "Ethereum: a secure decentralised generalised transaction ledger", *Ethereum Project Yellow Paper*, 2014, ISSN: 1098-6596. DOI: `10.1017/CBO9781107415324.004`.

[131] V. Buterin, "Ethereum White Paper", *Etherum*, 2014.

[132] N. Szabo, "Formalizing and securing relationships on public networks", *First Monday*, 1997, ISSN: 13960466. DOI: `10.5210/fm.v2i9.548`.

[133] D. Costa, F. Fiori, P. Milan, M. Sala, A. Vitale, and M. Vitale, "Quadrans White Paper", 2019.

[134] M. Wu, K. Wang, X. Cai, S. Guo, M. Guo, and C. Rong, "A Comprehensive Survey of Blockchain: From Theory to IoT Applications and beyond", *IEEE Internet of Things Journal*, 2019, ISSN: 23274662. DOI: `10.1109/JIOT.2019.2922538`.

[135] H. Shafagh, L. Burkhalter, A. Hithnawi, and S. Duquennoy, "Towards Blockchain-based Auditable Storage and Sharing of IoT Data", in *Proceedings of the 2017 on Cloud Computing Security Workshop - CCSW '17*, New York, New York, USA: ACM Press, 2017, pp. 45–50, ISBN: 9781450352048. DOI: `10.1145/3140649.3140656`. [Online]. Available: `http://dl.acm.org/citation.cfm?doid=3140649.3140656`.

[136] S. Huh, S. Cho, and S. Kim, "Managing IoT devices using blockchain platform", in *2017 19th International Conference on Advanced Communication Technology (ICACT)*, IEEE, 2017, pp. 464–467, ISBN: 978-89-968650-9-4. DOI: `10.23919/ICACT.2017.7890132`. [Online]. Available: `http://ieeexplore.ieee.org/document/7890132/`.

[137] A. Bahga and V. K. Madisetti, "Blockchain Platform for Industrial Internet of Things", *Journal of Software Engineering and Applications*, vol. 09, no. 10, pp. 533–546, 2016, ISSN: 1945-3116. DOI: `10.4236/jsea.2016.910036`. [Online]. Available: `http://www.scirp.org/journal/doi.aspx?DOI=10.4236/jsea.2016.910036`.

[138] M. P. Caro, M. S. Ali, M. Vecchio, and R. Giaffreda, "Blockchain-based traceability in Agri-Food supply chain management: A practical implementation", in *2018 IoT Vertical and Topical Summit on Agriculture - Tuscany, IOT Tuscany 2018*, Institute of Electrical and Electronics Engineers Inc., Jun. 2018, pp. 1–4, ISBN: 9781538669303. DOI: `10.1109/IOT-TUSCANY.2018.8373021`.

[139] Y. P. Tsang, K. L. Choy, C. H. Wu, G. T. S. Ho, and H. Y. Lam, "Blockchain-Driven IoT for Food Traceability with an Integrated Consensus Mechanism", *IEEE Access*, vol. 7, pp. 129 000–129 017, 2019, ISSN: 21693536. DOI: `10.1109/ACCESS.2019.2940227`.

[140] Y. Xu, J. Ren, G. Wang, C. Zhang, J. Yang, and Y. Zhang, “A blockchain-based nonrepudiation network computing service scheme for industrial iot”, *IEEE Transactions on Industrial Informatics*, vol. 15, no. 6, pp. 3632–3641, Jun. 2019, ISSN: 19410050. DOI: 10.1109/TII.2019.2897133.

[141] G. Dittmann and J. Jelitto, “A blockchain proxy for lightweight iot devices”, in *Proceedings - 2019 Crypto Valley Conference on Blockchain Technology, CVCBT 2019*, Institute of Electrical and Electronics Engineers Inc., Jun. 2019, pp. 82–85, ISBN: 9781728136691. DOI: 10.1109/CVCBT.2019.00015.

[142] B. J. Nelson, “Remote Procedure Call”, Ph.D. dissertation, Pittsburgh, PA, USA, 1981.

[143] J. Chow, *Technical Introduction to Events and Logs in Ethereum*, 2016. [Online]. Available: https://media.consensys.net/technical-introduction-to-events-and-logs-in-ethereum-a074d65dd61e.

[144] JSON-RPC, *JSON-RPC 2.0 Specification*, 2013. [Online]. Available: https://www.jsonrpc.org/specification%20http://www.jsonrpc.org/specification.

[145] Foodintegrity.org, “A Dangerous Food Disconnect When Consumers Hold You Responsible But Don’t Trust You”, Tech. Rep., 2018.

[146] J. Etienne, S. Chirico, K. McEntaggart, S. Papoutsis, and E. Millstone, “EU Insights – Consumer perceptions of emerging risks in the food chain”, *EFSA Supporting Publications*, 2018, ISSN: 23978325. DOI: 10.2903/sp.efsa.2018.en-1394.

[147] *Mipaaf - ICQRF - Report attività 2018.* [Online]. Available: https://www.politicheagricole.it/flex/cm/pages/ServeBLOB.php/L/IT/IDPagina/13602.

[148] *The economic costs of IPR infringement in spirits and wine.* [Online]. Available: https://euipo.europa.eu/ohimportal/en/web/observatory/ipr_infringement_wines_and_spirits.

[149] A. Hancoak, *Younger Consumers Drive Shift to Ethical Products*, 2017.

[150] V. G. Venkatesh, K. Kang, B. Wang, R. Y. Zhong, and A. Zhang, “System architecture for blockchain based transparency of supply chain social sustainability”, *Robotics and Computer-Integrated Manufacturing*, vol. 63, Jun. 2020, ISSN: 07365845. DOI: 10.1016/j.rcim.2019.101896.

[151] T. M. Fernández-caramés and P. Fraga-Lamas, “A Review on the Use of Blockchain for the Internet of Things”, *IEEE Access*, vol. 6, pp. 32 979–33 001, 2018, ISSN: 21693536. DOI: 10.1109/ACCESS.2018.2842685.

[152] G. Strawn, "Blockchain", *IT and Twenty-First Century Employment*, vol. 21, no. 1, pp. 91–92, 2019, ISSN: 1520-9202. DOI: `10.1109/MITP.2018.2879244`. [Online]. Available: `https://ieeexplore.ieee.org/document/8657387/`.

[153] K. Christidis and M. Devetsikiotis, "Blockchains and Smart Contracts for the Internet of Things", *IEEE Access*, vol. 4, pp. 2292–2303, 2016, ISSN: 21693536. DOI: `10.1109/ACCESS.2016.2566339`.

[154] D. Aggarwal, G. K. Brennen, T. Lee, M. Santha, and M. Tomamichel, "Quantum attacks on Bitcoin, and how to protect against them", pp. 1–21, 2017. DOI: `10.5195/ledger.2018.127`. [Online]. Available: `http://arxiv.org/abs/1710.10377%0Ahttp://dx.doi.org/10.5195/ledger.2018.127`.

[155] *Image created using Creative Commons icons found in www.onlinewebfonts.com/icon.*

[156] *Ethereum Project.* [Online]. Available: `https://www.ethereum.org/`.

[157] V. Gheorghiu, S. Gorbunov, M. Mosca, and B. Munson, "Quantum-Proofing the Blockchain", *Blockchain Research Institute (BRI)*, no. November, 2017.

[158] P. Shor, "Algorithms for quantum computation: discrete logarithms and factoring", in *Proceedings 35th Annual Symposium on Foundations of Computer Science*, IEEE Comput. Soc. Press, 1994, pp. 124–134, ISBN: 0-8186-6580-7. DOI: `10.1109/SFCS.1994.365700`. [Online]. Available: `http://ieeexplore.ieee.org/document/365700/`.

[159] *Round 2 Submissions - Post-Quantum Cryptography | CSRC.* [Online]. Available: `https://csrc.nist.gov/projects/post-quantum-cryptography/round-2-submissions`.

[160] S. King and S. Nadal, "PPCoin: Peer-to-Peer Crypto-Currency with Proof-of-Stake", in, 2012, pp. 1–6, ISBN: 9781450349741. DOI: `10.1017/CBO9781107415324.004`. [Online]. Available: `https://bitcoin.peryaudo.org/vendor/peercoin-paper.pdf`.

[161] K. Chalkias, J. Brown, M. Hearn, T. Lillehagen, I. Nitto, and T. Schroeter, "Blockchained Post-Quantum Signatures", 2018.

[162] *QRL - The Quantum Resistant Ledger.* [Online]. Available: `https://theqrl.org/`.

[163] P. Waterland, "Quantum Resistant Ledger (QRL)", Tech. Rep. October, 2016, pp. 1–15. [Online]. Available: `http://theqrl.org/whitepaper/QRL_whitepaper.pdf`.

[164] R. G. Brown, J. Carlyle, I. Grigg, and M. Hearn, "Corda : An Introduction", pp. 1–15, 2016.

[165] S. Popov, *The Tangle*, 2018.

[166] *Qubic: Quorum-based Computations - Powered by IOTA.* [Online]. Available: `https://qubic.iota.org/`.

[167] M. Collotta, R. Ferrero, E. Giusto, M. Ghazi Vakili, J. Grecuccio, X. Kong, and I. You, “A fuzzy control system for energy-efficient wireless devices in the Internet of vehicles”, *International Journal of Intelligent Systems*, vol. 36, no. 4, pp. 1595–1618, Apr. 2021, ISSN: 0884-8173. DOI: `10.1002/int.22353`. [Online]. Available: `https://onlinelibrary.wiley.com/doi/10.1002/int.22353`.

[168] E. Giusto, F. Gandino, M. L. Greco, M. Rebaudengo, and B. Montrucchio, “A dense RFID network for flexible Thermal Monitoring”, in *2018 6th International EURASIP Workshop on RFID Technology (EURFID)*, IEEE, 2018, pp. 1–7.

[169] E. Giusto, R. Ferrero, F. Gandino, B. Montrucchio, and M. Rebaudengo, “Advancements in Distributed Air Quality Monitoring Systems”, pp. 127–137, 2018.

[170] R. Ferrero, M. G. Vakili, E. Giusto, M. Guerrera, and V. Randazzo, “Ubiquitous fridge with natural language interaction”, in *2019 IEEE International Conference on RFID Technology and Applications (RFID-TA)*, IEEE, 2019, pp. 404–409.

This Ph.D. thesis has been typeset by means of the TEX-system facilities. The typesetting engine was pdfLATEX. The document class was `toptesi`, by Claudio Beccari, with option `tipotesi=scudo`. This class is available in every up-to-date and complete TEX-system installation.