# Dynamic learning of the environment for eco-citizen behavior

Davide Guastella


# THÈSE

**En vue de l'obtention du**

## DOCTORAT DE L'UNIVERSITÉ DE TOULOUSE

**Délivré par :** *l'Université Toulouse III - Paul Sabatier (UT3 Paul Sabatier)*
Cotutelle internationale *Università degli Studi di Catania*

---

**Présentée et soutenue le** *14/12/2020* **par :**

**Davide Andrea GUASTELLA**

---

**Dynamic Learning of the Environment for Eco-Citizen Behavior**

---

**JURY**

| | | |
|---|---|---|
| MARIE-PIERRE GLEIZES | Professeur d'Université | Co-Directrice |
| MASSIMO COSSENTINO | Directeur de recherche | Co-Directeur |
| VALÉRIE CAMPS | Maître de Conference | Co-Encadrante |
| CESARE FABIO VALENTI | Maître de Conference | Co-Encadrant |
| JEAN-PAUL JAMONT | Professeur d'Université | Rapporteur |
| ANDREA OMICINI | Professeur d'Université | Rapporteur |
| LAURENT VERCOUTER | Professeur d'Université | Examinateur |
| GIANCARLO FORTINO | Professeur d'Université | Examinateur |

---

**École doctorale et spécialité :**
*MITT : Domaine Mathématiques : Mathématiques appliquées*
**Unité de Recherche :**
*Institut de Recherche en Informatique de Toulouse (IRIT)*
**Directeur(s)/Encadrant(s) de Thèse :**
*Valérie Camps*, *Massimo Cossentino*, *Marie-Pierre Gleizes* et *Cesare Fabio Valenti*
**Rapporteurs :**
*Jean-Paul Jamont* et *Andrea Omicini*

II

Davide Andrea Guastella

# DYNAMIC AND REAL-TIME LEARNING OF THE ENVIRONMENT FOR ECO-CITIZEN BEHAVIOR IN SMART CITIES

Thesis Supervisors **Marie Pierre Gleizes**, Full Professor, Université Toulouse III Paul Sabatier
**Valérie Camps**, Associate Professor, Université Toulouse III Paul Sabatier
**Cesare Valenti**, Associate Professor, Università degli Studi di Palermo
**Massimo Cossentino**, Senior Research Scientist, Consiglio Nazionale delle Ricerche (CNR)

THE development of sustainable smart cities requires the deployment of Information and Communication Technology (ICT) to ensure better services and available information at any time and everywhere. As IoT devices become more powerful and low-cost, the implementation of an extensive sensor network for an urban context can be expensive. This thesis proposes a technique for estimating missing environmental information in large scale environments. Our technique enables providing information whereas devices are not available for an area of the environment not covered by sensing devices [1]. The contribution of our proposal is summarized in the following points:

- limiting the number of sensing devices to be deployed in an urban environment;
- the exploitation of heterogeneous data acquired from intermittent devices;
- real-time processing of information;
- self-calibration of the system.

Our proposal uses the **Adaptive Multi-Agent System** (AMAS) [53] approach to solve the problem of information unavailability. In this approach, an exception is considered as a Non-Cooperative Situation (NCS) that has to be solved locally and cooperatively. HybridIoT exploits both homogeneous (information of the same type) and heterogeneous information (information of different types or units) acquired from some available sensing device to **provide accurate estimates in the point of the environment where a sensing device is not available**. The proposed technique enables estimating accurate environmental information under conditions of uncertainty arising from the urban application context in which the project is situated, and which have not been explored by the state of the art solutions [2]:

- **openness**: sensors can enter or leave the system at any time without the need for any reconfiguration;

- **large scale**: the system can be deployed in a large, urban context and ensure correct operation with a significative number of devices;
- **heterogeneity**: the system handles different types of information without any *a priori* configuration.

Our proposal does not require any input parameters or reconfiguration. The system can operate in open, dynamic environments such as cities, where a large number of sensing devices can appear or disappear at any time and without any prior notification. We carried out different experiments to compare the obtained results to various standard techniques to assess the validity of our proposal. We also developed a pipeline of standard techniques to produce baseline results that will be compared to those obtained by our multi-agent proposal [4].

# REMERCIEMENTS

Cette aventure qu'est la thèse s'approche finalement à sa conclusion : une épreuve d'effort, de sacrifice, de dévouement qui est résumée dans ces pages. Comme je ne suis pas une personne très bavarde, je me limiterai à remercier dans ces lignes toutes les personnes qui m'ont permis d'arriver à ce résultat.

Tout d'abord, je remercie Andrea Omicini et Jean-Paul Jamont d'avoir accepté d'évaluer cette thèse, ainsi que les membres du jury, Giancarlo Fortino et Laurent Vercouter.

Je tiens à remercier mes chers compagnons d'aventure (AKA les SMACkers) : Maxime, Guilhem, Kristell, Augustin, Bruno, Walid, Nicolas, Valérian, et tous les autres, permanents et non, vieux et moins vieux. Merci à Valérie, pour ta rigueur et l'immense travail de relecture que t'as fait pour les articles ainsi que pour cette thèse et pour m'avoir soutenu et accompagné (avec beaucoup de patiente de ta part) vers ce résultat. Merci à Pierre et Marie-Pierre, plus que des simples encadrants, votre sourire, votre disponibilité et votre cordialité m'ont rassuré et ont tout à fait rendu le travail de la thèse bien agréable. Je remercie Massimo et Cesare, véritables guides spirituels depuis longtemps désormais. Je ne pourrai jamais vous remercier suffisamment pour tout ce que vous avez fait pour moi, pour m'avoir soutenu (et supporté), pour vos précieux conseils. Je ne serais pas ici sans votre soutien.

Un grand remerciement va à ma famille pour son énorme soutien pendant ces années passées loin de chez eux.

Pour finir, merci à Giulia : merci de m'avoir toujours soutenu pendant ces années vécues loin de toi, de m'avoir soutenu dans les moments les plus sombres de ce chemin, de m'avoir toujours montré le bon côté des choses, de rendre harmonieux ce monde chaotique. Je ne serai jamais capable d'exprimer tout mon amour pour toi, pour cela je dois m'appuyer sur les mots de quelqu'un autre :

*I feel wonderful,*
*Because I see the love light*
*in your eyes,*
*And the wonder of it all,*
*Is that you just don't realize*
*how much I love you*
E. Clapton

# Contents

# Part I

# Context Domain and State of The Art

Chapter 1

# Introduction

THE increasing diffusion and accessibility of the *Internet Of Things* (IoT) sensors enabled cities to become urban sensing platforms [105]. Data acquisition, through these platforms, enables cities to become ”Smart”, using environmental information collected in a participatory way to improve services and reduce their ecological footprint [40]. Environmental information includes information about air, water, soil, land, flora and fauna, energy, noise, waste and emissions, but also information about decisions, policies and activities that affect the environment [72].

A wide instrumentation of the environment through sensing devices could help to assess important information such as pollution [69], hydrological forecasting [128] or traffic estimation [60]. However, in large-scale contexts is difficult to deploy a large number of devices so as to enable sufficient informative coverage through the urban environment; this is due to the high costs of maintenance and installation of the network infrastructure [60]. For this reasons, it is necessary to conceive effective solutions that enable leveraging the potential of data perceived through a wide range of sensing devices.

Thanks to their increasing computational power and accessibility, smart devices can be exploited to make data acquisition in cities a participatory activity. This is the key concept of *Mobile Crowd Sensing* (MCS), which leverages device mobility and sensing capabilities, as well as human collaboration and intelligence to distributively perform tasks and provide cost-efficient applications and services [79]. For example, residents can use MCS for in-building navigation, real-time public transportation information, for cooperating with local administrators and policy-makers in activities such as reporting damages to public facilities. Moreover, the government departments can use MCS as a useful tool to monitor, manage, and upgrade the city infrastructures efficiently [79]. For that, MCS enables integrating different types of smart devices into a large scale sensing infrastructure. However, smart devices can embed a limited set of sensors: in this case, it is necessary to compensate for the lack of data through a mechanism for estimating the missing information.

## 1.1 Contribution

This thesis presents the HybridIoT system, based on a *Multi-Agent System* (MAS), that enables coping with the lack of environmental information at a large scale. A MAS is a system composed of multiple interacting and autonomous entities known as *agents*, each one acting and sensing within a common environment. Agents have a partial view of their environment; they act jointly to produce a result for a goal that cannot be achieved individually. Due to the distribution of tasks within the agents and the possibility to decentralize control and decision, a MAS is more suitable to model and simulate complex systems than traditional approaches [99]. It offers a framework to model, study, and control complex systems with a bottom-up approach by focusing on the entities and their interactions to solve a wide variety of problems [150].

The novelty of our contribution lies in:

- providing a solution for estimating missing environmental information in local parts of the environment not sufficiently covered by sensing devices through the exploitation of historical data perceived by sensors;
- limiting the installation of a large number of sensing device thanks to the use of virtual sensors;
- proposing a technique to estimate missing data of a particular type (in terms of unit, scale, and information) using the information of the same type and/or information of different types.

The benefits of the HybridIoT compared to the state of the art techniques consist of the following properties:

- *openness*: the system allows the sensors to enter or leave the system at any time without compromising its operation. Current state of the art solutions make use of mathematical or artificial intelligence techniques that prevent real-time operation and do not allow sensors to enter or leave the system;
- *heterogeneity*: HybridIoT exploits both homogeneous and heterogeneous information coming from mobile and intermittent sensors in order to provide accurate estimates whereas sensing devices are not available. We say a group of information to be homogeneous if it is composed of information of the same type, composed of information of the same nature. Contrarily, a group of heterogeneous information contains information of different types, not necessarily correlated;
- *large-scale*: the multi-agent approach distributes the computation among several computational entities (agents) that operate in local parts of the environment. This approach ensures that the system can operate independently of the number of sensing devices involved in its operation.

- *configuration*: the system does not require any parameters that depend on the application context or the information perceived by the sensors. HybridIoT does not require any particular configuration;

- *cost-effectiveness*: the use of virtual sensors, on which the HybridIoT system is based, makes it possible to avoid the installation of a large number of physical sensors. This reduces the installation and maintenance costs of the sensor network;

- *privacy*: it refers to the ability of the system to avoid the diffusion of personal data. This is important when using devices such as smartphones, or in general, devices that use personal information. HybridIoT does not make use of any personal data; instead, only the environmental information is used for the estimation.

In a large-scale environment, several thousand devices could face unpredictable situations in which information has to be estimated and provided to users. Moreover, having a large amount of heterogeneous information is useful in different application contexts: at the urban level, a deep knowledge of environmental dynamics can be useful to governments to provide better services to citizens (e.g. by observing air quality, pollution levels); at extra-urban level, observing heterogeneous environmental information can help to understand and possibly anticipate phenomena such as floods or fires. In these contexts, it is required to have an infrastructure composed of a large number of reliable sensors to ensure a continuous and effective observation of the environment. The implementation of a system for estimating missing information at large-scale requires high computational power and efficient communication infrastructure. Our proposal allows reducing the need for high computational time through a distributed computation paradigm based on AMAS.

## 1.2 Manuscript Organization

This manuscript is organized as follows:

- **Section I**: Context domain and state of the art

  - (Chapter 2) This chapter provides an overview of the smart city, discusses its importance in urban and technological development and its different areas of application. Later we focus on some of the main challenges that concern the domain in which our proposal takes place. The chapter ends with a theoretical problem statement to formalize the problem that this thesis address.
  - (Chapter 3) This chapter describes the principal methods for estimating missing information in urban and large scale context. We study these state of the art methods according to the properties of openness, large scale, and heterogeneity, addressed by our proposal.

  - (Chapter 4) This chapter provides an introduction to multi-agent systems and then to the adaptive multi-agent systems which is a pertinent approach to tackle problems such as unexpected behaviors, the integration in open and heterogeneous environments.

- **Section II**: Contribution

  - (Chapter 5) This chapter concerns the contribution of this thesis. Our proposal, called HybridIoT, is a multi-agent system that implements a technique for estimating missing environmental information. The proposed method addresses the properties of large scale, openness and heterogeneity. The definition of our proposal follows the guidelines of the ADELFE methodology for the development of adaptive multi-agent systems.

- **Section III**: Evaluation

  - (Chapter 6) This chapter describes the real dataset used for the experiments, which is not pre-processed by the proposed technique. Then the chapter describes the evaluation techniques used to evaluate our proposal under different experimental conditions and the obtained results.

Chapter 2

# Introduction to Smart cities

Objectives of this chapter:

- Introduce the smart city as a tool to face the development and needs of modern society
- Identify the application domains of the smart city
- Review the main definitions of the smart city
- Identify the economic and social value created by smart cities
- Identify the current challenges for the development of smart cities

In the olden and not-so-olden days, technological advances have accompanied the development of human settlements to manage the increasing complexity of society demands. This phenomenon has intensified considerably during the period of nineteenth-century industrialization, allowing many cities around the world to become landmarks in technological and social development. At that time, technological development was strictly linked to the exploitation of resources such as steam power and electricity. The urban expansion was not driven only by new machines that amplified our physical might, but also by inventions that multiplied our ability to process information and communicate quickly over great distances. For example, the telegraph revolutionized the way of communicating: it has facilitated communication between private individuals and industry, accelerating social and technological progress and reducing, although virtually, the distances between individuals.

Today communication has gone far beyond these objectives, enabling real-time interaction between people in every corner of the world and making telematic means accessible to a large part of the world's population. The miniaturization of communication devices, their increasing computational power and low-price allow having tools that do more than just communication tasks:

understanding the users' lifestyle to suggest purchases, remind events, meetings, but also observing the environment, getting the status of traffic. In less than 100 years, communication devices have become truly personal assistants.

Communication is just one of the examples of technological advancement that has led to the development of modern cities. It emerged that the growth of cities and the technological progress are strongly linked [134]. The continuous urban development, the increasingly complex needs of the society and ever-increasing urbanization have led today to a need for technological means to guarantee better services to citizens. Governments aim at interacting in a simpler and efficient way with citizens, police authorities to guarantee greater security and also to exploit energy resources in a responsible and sustainable way. These needs led to the definition of the concept of "*Smart City*".

Smart cities can be seen in two different ways:

- [a smart city] defines how cities can be conceived and developed in order to guarantee a high quality of life and intelligent use of energy resources;
- [a smart city should] integrates new technologies with existing infrastructures to improve the functioning of cities by enhancing their efficiency and improving their competitiveness.

In both cases, the smart city allows cities to integrate in a transparent and ubiquitous way the industrial and technological advances so that cities can face a sustainable development and reduce the environmental impact of human activity while ensuring a good quality of life.

The smart city answers, through technological means, to problems of resource optimization, efficient governance, better interactions with citizens. Briefly, the smart city affects all aspects of the urban society: where there is now waste (in terms of time and/or resources), there will be efficiency. Where there are volatility and risk, there will be predictions and early warnings. Where there are crime and insecurity, there will be watchful eyes. Where you now stand in line, you will instead access government services online [134].

This technological revolution makes it possible to exploit better governance over cities whose population raised dramatically over the last century. Smart city initiatives have a significant impact on various aspects: governance, people and communities, economy, natural environment, and built infrastructure.

Different cities have started implementing their smart projects in different ways. Because the concept of *smart city* is not related to any particular project, each city has an enormous set of possible actions to be carried out, but which to choose? Figure 1 shows the map of cities all over the world awarded for their smartness by the Intelligent Communities Forum [33].

Developing a smart city is a complex task because it involves several dimensions: technology, citizens, public and private organizations, urban vision, security. Consequently, the possible scenarios related to the smart city context are particularly wide and in continuous evolution. Moreover, smart initiatives interest cities all over the world, with very deep differences with each

Figure 2.1: Intelligent cities in the world [33]

other: cultural, economic, social. Every city wants both to apply a shared smart city idea and to pursue its own specific goals [34].

Even if the majority of the urban population lives in metropolises, the main focus of urban research tends to be in medium-sized cities. Medium-sized cities, which have to face the competition of the larger metropolises on corresponding issues, appear to be less well equipped in terms of critical mass, resources and organizing capacity. To enforce the development and achieve a good position, cities have to identify their strengths and chances for positioning and ensuring the comparative advantages in certain key resources against other cities of the same level [83].

The growth of cities leads to a demand for more energy resources and more efficient services for citizens. In this context, smart city initiatives play a major role in the development of urban contexts that allows to guarantee a high level of quality of life and to use environmental resources in a respectful manner.

This chapter provides a brief introduction to the smart city. The rest of this chapter is organized as follows: section 2.1 presents the main application domains in which the concept of smart city takes place. Section 2.2 presents some of most common definitions of smart city according to the academic and industrial world. Section 2.3 discusses the importance of the socio-economic return that smart initiatives should bring to both attract investments and produce value for societies. Section 2.4 discusses some of the main challenges in the development of a smart city. Finally, section 2.5 briefly introduces the AI, its evolution and its role in assisting human activities. Section 2.6 discusses how AI integrates into smart cities.

## 2.1 Smart City Application Domains

Smart initiatives focus on exploiting technological means to provide better services to citizens, ensure a good quality of life and drive sustainable urban development. Sánchez-Corcuera et al. [120] separate the application domains of smart city in four categories:

- business-related domain,

- citizens-related domain,
- environment domain and
- government domain.

### 2.1.1 Business-Related Domain

The business-related category includes all the applications that make use of information technologies to promote the development and growth of companies and industries. The most common application is the advertisement, where technological means are used to promote products, services and reach a wide range of the population. For example, the diffusion of mobile devices such as smartphones have allowed the spread of advertising through services such as social networks. The information that can be collected from users, such as their habits, searches, purchases, are used to provide targeted advertising to increase the likelihood that advertisements will be attractive. On the other hand, this constitutes an ethical issue that concerns the invasive use of technological means to acquire personal data from citizens. Advertising must be carried out in a user-oriented and non-invasive manner while being aware of the privacy of users.

### 2.1.2 Citizens-related Domain

Citizens-related domain concerns all the aspects of citizens' lives that can be assisted by technological means. Education, urban services, health, transport are some of the services that concern this domain. In healthcare service, the use of *Information and Communication Technologies* (ICT) leads to the concept of electronic health (e-health), which contributes to reducing costs and increases the efficiency of medical treatments. In this domain, the last few years have seen the development of integrated systems that allow continuous and precise monitoring of patients' health remotely, using non-invasive sensors and monitoring devices. In 2020, the advent of the CoVid-19 pandemic enabled an extensive use of this type of technology to avoid overcrowding in hospitals by doing telematic consultations but also to do contact tracing to identify the persons who may have come into contact with an infected person. This is the case of multiple application deployed in different countries such as France (StopCovid), Italy (Immuni), Germany (Corona-Warn-App). The use of technologies based on sensors and applications integrated with smartphones has made it possible to calculate more quickly and effectively models of contagion used to prevent the diffusion of the virus [112]. Using mobile and ubiquitous devices in healthcare domain has enormous advantages such as monitoring capabilities, wide availability and immediacy that contribute to monitor efficiently the health state of patients [126].

In the domain of transport, recent development in ICT enabled the integration of technological devices in urban contexts to avoid delays, bottlenecks, pollution, accidents, and a continuous decline in the quality of life for citizens in high polluted areas [124]. The integration of ICT

with the transportation infrastructure enables a better, safer traveling experience with Intelligent Transportation Systems (ITS) [60], which enables continuous observation of the environment.

### 2.1.3 Environment domain

Smart environment applications have as objective the minimization of the ecological and energy footprint of cities. Sánchez-Corcuera et al. [120] identified the following sub categories for this domain: building, housing, pollution control, public space, renewable energy, smart grid, waste management, and water management.

Buildings are the place where citizens spend most of their time, to live but also to work. Consequently, buildings are considered as the largest consumers of energy, especially in large cities [48]. Smart building initiatives have as objective the definition of methods and tools to reduce energy consumption while ensuring a good level of comfort for users. Therefore, buildings are important testbeds for the implementation of smart technologies. The use of intelligent systems allows automating *Heating, Ventilation and Air-Conditioning* (HVAC) systems to reduce energy consumption by ensuring a good level of comfort for occupants. HVAC systems run on schedules that are meant to address these properties, generally by using IoT sensors that allow observing the environment and users' habits.

Smart grid networks constitute an evolutionary step for power supply networks. A smart grid is an enhancement of the traditional power grid, used to carry out power from a few central generators to a large number of users or customers [44]. These networks improve efficiency and reliability and provide an uninterrupted energy supply to homes and businesses [82]. The increase in the size of the cities and the subsequent demographic increase requires adequate support for the electricity grid to meet the needs of citizens. This does not necessarily involve the production of a large amount of energy, but rather the use of different energy sources as well as the redistribution of the electrical load to ensure good coverage without energy losses.

### 2.1.4 Government domain

Smart government domain refers to city monitoring, e-government, emergency response, public safety, public service, and transparent government services. Academic researches suggest the importance of comprehensive governance by both local and central governments, aiming at designing an urban smart strategy [32]. Smart government applications use ICT to better interact with their citizens, taking advantage of all available data to solve issues involving services and infrastructures. Therefore, governments need to involve citizens to provide them with better services. In recent years the government domain has driven the spread of public urban databases containing information on different aspects of the city (mobility, environment, citizens). The technology advances, involving the diffusion of miniaturized devices capable of acquiring data from the environment, is contributing to the production of these databases containing impressive amounts of data that

have the potential to help us to better understand complex social problems as well as to improve government relationships with citizens, private organizations, and other governments [94]. On the one hand, these databases allow citizens to have accurate and up-to-date information about the city. On the other hand, the information is useful for experts to evaluate improvement measures to make cities attractive. Moreover, governments must establish guiding principles of openness, transparency, participation, and collaboration to manage and facilitate the flow of information acquired in the urban context to address the development of a smart city [6].

### 2.1.5 Discussion

From the discussed domains, which are just some of the applications of the smart initiatives, it emerges that ICT plays a key role in the development of smart initiatives. Smart City applications run on top of the ICT infrastructure that puts a bridge between citizens and governance to improve city services.

The achievement of smartness goals requires extensive use of information acquired by different, heterogeneous technologies: in the traffic domain through ICT devices for acquiring traffic information to improve traffic conditions, in the medical domain to provide accurate diagnoses; in the environmental domain to provide precise indications on the state of the environment, to predict harmful phenomena. Therefore, it is necessary to exploit ICT extensively through the use of sensing devices to acquire a large amount of information. Nevertheless, as sensing devices operate independently, it is necessary to provide mechanisms of coordination and integration of information that allows creating an accurate knowledge base that can contribute to the achievement of smartness goals. However, instrumenting a large scale environment with many sensing devices can be expensive (in terms of both installation and maintenance), therefore it is necessary to conceive non-intrusive solutions to provide information without deploying a large number of sensing devices.

## 2.2 Defining the Smart City

Defining the smart city is difficult because there is no accepted definition and because of the variety of application domains in which the concept of ”smart” is applied, as discussed in the previous section. The same concept can vary from city to city, initiative to initiative. It seems that every city all over the world, across continents and independently from dimension, culture, economic situation, considers important to be smart [33].

The difficulty to define a Smart City regards mainly two aspects [27]: (*i*) the adjective ”smart” depends on the meaning we attribute to this word. This is mainly due to the variety of application domains in which the concept of the smart city takes place, as different typologies of the smart city exist, (*ii*) the label ”smart city” is a fuzzy concept and it is used in ways that are not always

in accordance each other. There are many cities that define themselves as smart because of some technological characteristics, but without referring to a standard meaning.

Townsend [134] defines smart cities as "*places where information technology is combined with infrastructure, architecture, everyday objects, and even our bodies to address social, economic, and environmental problems*". As technology must accompany the urban development of cities, the most important question to answer would be "what do we expect from a smart city". The development of the cities passes from different aspects such as mobility, energy, health, environment... therefore the concept of the smart city is strongly ambiguous; various definitions differentiate between several applications areas [2].

Dameri [33] identifies the most common definitions of the smart city according to different academic and industrial points of view:

- A city that *monitors and integrates conditions of all of its critical infrastructures*, including roads, bridges, tunnels, rails, subways, airports, seaports, communications, water, power, even major buildings, can better optimize its resources, plan its preventive maintenance activities, and monitor security aspects while maximizing services to its citizens.
- A city to be smart when investments in human and social capital and traditional (transport) and modern (ICT) communication infrastructure fuel sustainable economic growth and high quality of life, with a wise management of natural resources, through *participatory governance*.
- The smart city is the use of smart *computing technologies* to make the critical infrastructure components and services of a city—which include city administration, education, healthcare, public safety, real estate, transportation, and utilities—more intelligent, interconnected, and efficient.
- (Smart cities are about) leveraging *interoperability within and across policy domains of the city* (e.g. transportation, public safety, energy, education, healthcare and development). Smart City strategies require innovative ways of interacting with stakeholders, managing resources and providing services.
- A city in which it can *combine technologies* as diverse as water recycling, advanced energy grids and mobile communications in order to reduce environmental impact and to offer its citizens better lives.
- A city where social and technological infrastructures and solutions facilitate and *accelerate sustainable economic growth*. This improves the quality of life in the city for everyone.
- A city that uses *Information and Communication Technology* (ICT) to *sense, analyze and integrate the key information of core systems in running cities*.

Due to the heterogeneity of definitions, a unique definition would not be sufficient to give a global meaning to the concept of smart city. On the one hand, the term *smart* is generic and it is not

possible to give a global meaning to what represents a smart city; on the other hand, the initiatives are specific in their domain of application, preventing the formalization of the concept of the smart city. For this reason, academics and organizations prefer to define the smart city according to the domain of interest.

Dameri [31] analyzes five macro-areas of interest in which smart initiatives can be grouped. This would avoid looking for a unique definition of a smart city, but rather analyzing five macro-areas of interest for multiple domains:

- *Intelligent city*: it is able to produce knowledge and to translate it into unique and distinctive abilities; this city is smart because it is able to create intellectual capital and to ground development and well-being on this intellectual capital;
- *Digital city*: it is a wired, digitalized city, using ICT both for data processing and for information sharing, but also to support communication and Web 2.0 democracy;
- *Sustainable city*: it aims to become a "green city" by using technology to reduce pollution, to produce energy and to improve the efficiency of buildings;
- *Technocity*: it uses the technology to improve the efficiency and effectiveness of its infrastructures and services: it focuses its smart projects on urban space quality, mobility, public transports, logistic.
- *Well-being city*: it uses technology as a mean to improve the quality of life for citizens. Better services, climate management, noise reduction in working places are some examples where well-being concept takes place.

These concepts do not represent disjoint areas of analysis and still share some common aspects. Therefore, a smart initiative can be identified in one or more of the above definitions; for example, a smart initiative may pursue a goal that concerns both sustainable and well-being city. The previous macro-areas highlight the possibility of extending smart initiatives to different aspects of the urban context. The design and implementation of smart initiatives involve significant investments for cities.

Regardless of their type, smart initiatives must create both economic and social value. The following section discusses the value created from the use of smart initiatives in urban contexts.

## 2.3 Smart City Value

The definitions of smart city discussed in the previous section reveal that there is a significant increase in smart initiatives in both academic and industrial communities. Smart initiatives attempt to create socio-economic value in the context of urban development through technological innovations. Creating value by smart city initiatives is mandatory when applying for funding, especially for the

European Union. This is one of the biggest challenges for smart city projects. If these projects are not able to boost economy or even be economically successful to be able to become autonomous, smart city projects will always have to rely on governmental support and funding [11]. City leaders are struggling to identify the true sources of value that novel ICT can generate for their municipalities; in fact, it is difficult to transform a high-level concept into actionable, effective solutions that deliver measurable value to the citizenry. This is in part due to the nature of the city itself, which is an enormously complex and open-ended system, with many force fields simultaneously influencing its form and continuously evolving [28].

To create value through smart city initiatives, technological systems must allow redefining both institutions and urban environments in an accessible and transparent way. Generally, the use of smart solutions makes it possible to use the knowledge acquired from observing information disseminated on an urban scale to improve the quality of life so that cities become attractive for citizens and industries. An attractive city generates investments and consequently important revenues generated by an increase of population and industries. Because cities must continuously adapt to population growth by providing solutions that ensure a good quality of life for its citizens, smart initiatives must solve efficiently problems that directly affect citizens, such as traffic, energy, pollution and water quality. These aspects are also near to the idea of a green city: the environmental topic is an important part of the smart city goals [34].

Indeed, it is important to define what does public value means: it interests the **economical aspects** of the community and involves different persons or organizations in different ways, each one with their needs and expectations. Creating public value in a smart city means to put together a large set of variables and to compose them into a well-defined general framework, able to collect the needs, the expectations and the perception of citizens respect to the smart city for their daily life [27].

The use of technology itself is not sufficient to justify the development of a smart city: technology provides a means to achieve smartness goals. The development of smart initiatives requires significant investment and because of these entities that invest in these initiatives and that aim for an economic return. To define the smartness objectives of an initiative, especially in an urban context, it is important to interact directly with citizens, asking what they expect from a smart city. It follows that **citizens** are at the center of the development of smart cities and it is necessary to involve them in these activities. Modern social media and communication technologies constitute an efficient channel of communication between citizens and institutions. These communication channels can also be used to notify citizens about services or possible problems that have arisen in the city. In this way, individuating smartness objectives from citizens' opinions can be done efficiently.

Once the smartness objectives are defined, they must be measured. Measuring the public value created and supplied thanks to a smart city program is a complex task. This is primarily because smart city benefits are not defined, not measured and not communicated [31].

A smart city that creates value for the community must take into account different aspects such as the effectiveness, the environment and the innovation [27]:

- *Effectiveness* means the capacity of a city to supply effective public and private services to several subjects, such as citizens, companies, not-for-profit organizations;
- *Environmental consideration* regards the increasing impact that large cities have on the environmental quality of urban areas. One of the main pillars of smarter cities is to prevent further environmental degradation.
- *Innovation* means that a smart city should use all the new and higher available technologies to improve the quality of its core components, to deliver better services and to reduce its environmental impacts. Technology is, therefore, a central aspect of a smarter city, used for implementing smart initiatives for the quality of life in the city.

Different challenges must be undertaken to develop an efficient system that meets the previous goals and can create not only innovation but also socio-economic value. In general, each smart initiative differs by application domain, so as the challenges. The following section discusses some of the challenges that must be addressed in smart initiatives that focus on sustainable and intelligent cities.

## 2.4 Challenges in Smart cities

In recent years, the issue of sustainable development of cities has become of great importance due to an increase in population and the consequent need to ensure services and good quality of life while exploiting resources in a rational way. According to the *World Health Organization* (WHO), the proportion of people living in an urban environment will grow significantly by $2050$. It is challenging for cities to accommodate such a large amount of population. They must deal with issues such as congestion and increasing demand for resources including energy, water, education and health-care services.

In the last decade, the concept of **sustainable city** became increasingly important. This is primarily caused by rapid urbanization processes that must be supported by government institutions in order to ensure a high-quality level of life for the citizens by also taking care of the environment. In fact, the climate change primarily driven by human-caused greenhouse gas, will yield warmer temperatures than the previous $150$ years, and possibly warmer than at any time in the last $2000$ years [42], making the earth a hostile place in which man can live. A smart city must address a sustainable development of urban context to avoid carbon dioxide ($CO_2$) emission, reducing air pollution and improving the quality of life of citizens [100].

The development of smart environments introduces several challenges regarding the integration of heterogeneous systems and technologies, scalability, reliability and functional extensibility. In

particular, functional extensibility refers to the problem of adding new services to the set of existing ones already working in a given home environment [26].

One of the most important challenges to consider for addressing sustainable development is to **integrate technological tools** that enable precise and large-scale analysis of environmental information that can be used by experts to manage resources efficiently. It is not just a matter of acquiring data, rather creating useful knowledge from a huge amount of data collected from different, heterogeneous sources of data. A sustainable city is instrumented with ICT infrastructures for optimizing the resource distribution, preventing resource outages, ensure easy and rapid maintenance actions, and so on. To do this, **a large variety of interconnected sensing devices must be exploited in cities to acquire information from the urban context**.

The increasing diffusion and accessibility of the *Internet of Things* (IoT) sensors enabled cities to become urban sensing platforms [105]. The development of IoT systems such as the smart city, its management as well as its integration in real applications is complex and challenging, thereby requiring suitable models, methods/techniques and technologies. Different solutions have been developed to face challenges such as physical device virtualization, decentralized entity management, and guideline identification. However, these solutions tend to tackle different specific issues, typically one at a time, without providing a full-fledged methodology to support the entire IoT system development process, from analysis to implementation [47]. Data acquisition through IoT platforms allows cities to become smart, using data collected in a participatory way to improve services and reduce their ecological footprint [40]. The IoT aims at providing a global infrastructure for the information society, enabling advanced services by interconnecting physical and virtual things based on existing and evolving interoperable information and communication technologies. IoT provides a lower layer made up of the individual devices and their communication and computing capabilities. The main challenge of the IoT is to achieve **full interoperability** of interconnected devices while guaranteeing the **trust, privacy and security** of communications [108]. The integration of IoT technologies in the urban contexts requires coordination of devices through intelligent systems known as *Ambient Systems*, having numerous mechanisms that rule the behavior of the environment. *Ambient Intelligence* provides ambient systems with mechanisms necessary to carry out reasoning activities using a representation of the environment perceived by IoT devices. Ambient systems are designed to provide adapted services that respond to an individual, collective, and social requirement. The term *environment* refers to a physical space enriched with sensors and computational entities that are seamlessly and invisibly interwoven. To be considered as smart or intelligent, an environment needs to be associated with a representative description that can be constructed from the perceptions of the *ambient components*. The interactions of ambient components enable a smart city to enhance its services such as transports, health, cultural events and so on. Nevertheless, avoiding the installation of new components in an ambient system to provide precise everywhere and anytime information on the environment is a difficult task [2].

*Ambient systems* represent a key solution to integrate different technologies and provide an

IoT infrastructure to be aware of the environment, doing pervasive computing and profiling through human-centric computer interactions [138]. Today there is a growing interest to support technologies to support not only smart cities but also smart regions. Moreover, the deployment of applications in large scale environments is a difficult task due to a variety of constraints (e.g. energy, communication, computing capacities, mobility, autonomy) [4].

The implementation of a sustainable smart city must consider the complexity of the physical world: a city is characterized by multiple dynamics, non-linear relations, feedbacks, unpredictability, strong inertia... where the non-linearity leads to the impossibility to plan all the consequences of a change, even small. Therefore, the smart city is not static, it evolves with citizens, with policies, with its environment, and must be resilient in order to quickly answer to new challenges [55]. A single centralized technological solution would be inefficient because of the complexity of the physical environment; for this reason, it is expected that different technologies are integrated to create a solution that can cope with the complexity of the environment and respond effectively to the needs of both citizens and experts. Therefore, smart cities become *Systems of Systems* (SoS) aiming at bringing significative benefits to government, society, economy, and environment, as well as providing a complete, holistic view of the city. In this vision, each of such constituent systems might be SoS themselves, thereby increasing the matters of scale and complexity in the design, engineering, and operation of these systems [23].

## 2.5 Introduction to Artificial Intelligence

For over two thousand years scientists and philosophers have questioned how the human mind works. This question is still unanswered today, despite technological advances. Some philosophers have picked up the computational approach originated by computer scientists and accepted the idea that machines can do everything that humans can do [97].

In the scientific community, Alan Turing was one of the first scientists to advance the idea that machines could have some sort of "**intelligence**". In his lecture at the London Mathematical Society on February 20, 1947, Turing outlined the meaning of what today we define as Artificial Intelligence: "*What we want is a machine that can learn from experience*" [135]. Even then, Turing believed that computers should have some kind of intelligence more than being barely capable of performing calculations. As a matter of fact, at that time computers have been conceived for supporting users in doing long and tedious calculations in a short time. This process was pursued by executing a set of ordered instructions specified by the user so that the calculation is done correctly. Moreover, the user was conscious that the calculation has been done by the machine in a mechanical way, without any personal judgment guided by the "spirit", so the user knows perfectly the steps that lead the machine to its result. The user entered in the machine the detailed instructions, designed by himself.

The idea of Turing was that machines could be able not only to simulate human activities but also

to **learn**: he thought that modifying a computer's instructions was a process close to the learning process of human, like a student that learns from the teacher. In this way, machine processes would be no longer deterministic; they would be influenced by past experiences, judgments, and the instinct that make the human intellect unique. This means the possibility that machines can no longer perform a calculation in the same way that they were programmed, thus exposing them to errors. However, it is precisely the ability to err and refute based on failed experiences that allowed man to evolve [68]. The question that openly posed itself was, therefore: what would intelligence serve?

Since 1950, the industrial revolution aimed at replacing the human workforce with the machine to gain time and produce more in less time. To address this goal, a remarkable effort has been done in AI research. Although the definition of AI is very broad, it can be defined as "the study of making computers do things that the human needs intelligence to do" [96]. According to this definition, a machine must have intelligence, the capacity to think about what it does. In order to think, someone or something has to have a brain, or in other words, an organ that enables someone or something to learn and understand things, to solve problems and to make decisions. Thus, we can further define intelligence as "the ability to learn and understand, to solve problems and to make decisions" [97].

The development of AI tries to conceive methods to provide intelligence to machines. This concerns not only the computational ability but the possibility for the machines to carrying out heterogeneous activities by adapting to changes in the surrounding environment, improving their capabilities according to their own experiences and cooperating with other machines.

Table 2.1 groups some of the main definitions put together according to four approaches to AI. According to these definitions, a human-centered approach must be an empirical science, involving hypothesis and experimental confirmation, while a rational approach involves a combination of mathematics and engineering. Each group has both disparaged and helped the other [118]. The way Turing thought about the IA consists in designing systems that act like humans.

## 2.6 Artificial Intelligence for the Smart City

The diffusion of smart initiatives is motivated by an increase in population in centers of economic and industrial interest, which makes the management of urban resources challenging to avoid the deterioration of the environment and ensure a good quality of life for citizens. As cities are becoming digitized through the installation of sensors, computational cores and different telecommunication systems, AI makes it possible to collect real-time data to provide a deeper understanding of how cities evolve, adapt and respond to various conditions [9]. The main objective of AI systems is to create intelligent machines and through this, to understand the principles of intelligence. AI system is suitable when a direct mathematical relationship cannot be established between cause and effect. Artificial intelligence system models capture the uncertainty between real-life cause and effect scenarios by incorporating available knowledge with probabilities and probability inference

Table 2.1: Some definitions of artificial intelligence, organized into four categories [118].

| **Systems that think like humans** | **Systems that think rationally** |
|---|---|
| "[The automation of] activities that we associate with human thinking, activities such as decision-making, problem solving, learning ..." (Bellman, 1978) | "The study of mental faculties through the use of computational models." (Chamiak and McDermott, 1985) |
| "The exciting new effort to make computers think... machines with minds, in the full and literal sense." (Haugeland, 1985) | "The study of the computations that make it possible to perceive, reason, and act." (Winston, 1992) |
| **Systems that act like humans** | **Systems that act rationally** |
| "The art of creating machines that perform functions that require intelligence when performed by people." (Kurzweil, 1990) | "Computational Intelligence is the study of the design of intelligent agents." (Poole et al., 1998) |
| "The study of how to make computers do things at which, at the moment, people are better." (Rich and Knight, 1991) | "AI...is concerned with intelligent behavior in artifacts." (Nilsson, 1998) |

computations [5]. The processing of data through AI can ensure the better provision of liveability dimensions; through cleanliness, health and conducive environments for people to live and work without the urban challenges of pollution and congestion. It is further believed that, through this technology, the built environment can digitally support intelligent and responsive services both conveniently and in real-time [9].

A smart city exploits synergistically multiple technologies; this is of primary importance in large-scale applications where the computation where the achievement of smartness objectives requires the use of diverse computational systems that operate according to mutual interactions to produce a result that depends on the computation of the individual parts. This arising complexity motivates the distribution of computation among different computing devices, thus focalizing more on how they interact. These aspects are addressed through models such as ubiquitous and autonomous computing.

- "*The Ubiquitous Computing era will have lots of computers sharing each of us. Some of these computers will be the hundred we may access in the course of a few minutes of Internet browsing. Others will be embedded in walls, chairs, clothing, light switches, cars - in everything. Ubiquitous Computing is fundamentally characterized by the connection of things in the world with computation. This will take place at a many scales, including the microscopic*" [144].

- [Autonomous systems are] "*computing systems that can manage themselves given high-level objectives from administrators. These systems manage themselves according to an administrator's goals*" [76].

The development of smart cities is based on the observation and analysis of data acquired

through a large number of IoT devices available in the urban environment. To make sense, the role of AI is to reason on and to coordinate the data observed by IoT devices while taking into account different constraints such as processing power, memory, and delay in real-time applications.

The main challenge of IoT is to achieve full interoperability of interconnected devices while guaranteeing the trust, privacy and security of communications [108]. These interconnected devices become more unobtrusive and thanks to their embedded sensors they can perceive the physical environment in which they are situated. *Ambient Intelligence* (AmI) provides ambient systems with the mechanisms necessary to carry out environmental reasoning using a representation of the environment perceived by IoT devices. Ambient systems are designed to provide adapted services that respond to individual, collective, and social requirements. The term *environment* refers to a physical space enriched with sensors and computational entities that are seamlessly and invisibly interwoven [109]. To be considered as smart or intelligent, an environment needs to be associated with a representative description that can be constructed from the perceptions of the *ambient components*. The interactions of ambient components enable a smart city to enhance its services such as transports, health, cultural events and so on. This is possible thanks to the analysis of data acquired not only from the physical environment but also from the analysis of citizens' behavior. Ambient components perform computation using different information collected from the environment. However, their computation depends on what is done and the information perceived in a precise moment and in a precise environmental context. The property of adapting to different situations is known as *context-awareness*.

Ambient intelligence involves the following three properties: ubiquity, connectivity and intelligence [140]. According to ubiquitous computing, computational units are embedded in all surrounding and everyday devices, functioning invisibly. Human users may not necessarily be aware of the existence of embedded devices and computations occurred behind the scene [141]. Connectivity involves the efficient use of networks to support the interoperability of connected ambient devices. As ambient systems are designed to be deployed in large-scale contexts such as cities, ambient devices can be present in a significant quantity. For this reason, it is necessary to dispose of an effective infrastructure that ensures connectivity to the services offered by the smart city and therefore to ambient components.

Software agents offer features that answer to key needs in Ambient Intelligence (AmI). Agents are characterized primarily by autonomy, proactivity and reasoning mechanisms: autonomy enables agents to act and take decisions with or without information from other agents, and to easily adapt to changing contexts. Reasoning and proactivity allow agents to act before the user requires it explicitly, making the AmI system be more useful and appear more intelligent. Especially coupled with mechanisms of self-organization, using multi-agent systems can be effective to deliver a high level of decentralization thanks to the autonomous computation of software agents [101]. Agents can integrate intelligence mechanisms necessary for the implementation of ambient systems. The intelligence can be appropriately distributed among agents and allows a punctual and real time

analysis of the information collected from the physical environment. Intelligence can be attributed in different ways, according to the application domain, through AI techniques.

Ambient intelligence deals with data acquired in large scale environments. Because smart cities are increasingly instrumented with sensors capable of acquiring information from the environment, ambient systems must be able to manipulate a large amount of information. Thanks to their autonomous computation, agents can be employed in the context of smart cities in order to allow a distributed acquisition of information associated with intelligence mechanisms.

### 2.6.1 Discussion

It is clear from the discussion presented in the previous section that the link between ambient intelligence, AI and software agents is close: the former provides intelligence mechanisms in distributed environments, that is, in contexts where large amounts of data are perceived by devices present in the urban environment. These devices, known as ambient components, are heterogeneous and can perceive different types of information.

The ambient systems, therefore, have the role of creating useful knowledge for society by exploiting the large amount of information perceived through perception devices. This is possible today also thanks to the low cost of devices able to perceive the environment; on the one hand, this allows ambient systems to have a large amount of information that can be used to build a precise urban knowledge base, On the other hand, ambient systems need to be able to handle huge amounts of data efficiently.

In this thesis, ambient intelligence is used to manipulate environmental data in order to estimate missing information where *ad hoc* sensors are not available.

We provide a working definition of a smart city for the purposes of this thesis:

*The smart city is an urban settlement that uses intelligent systems capable of operating in open, dynamic and heterogeneous environments to leverage the data acquired in a participative way from IoT devices to achieve a sustainable development of the city.*

In this thesis, we consider a city as smart because it holds a large quantity of information perceived by heterogeneous sensing devices whose information can be used to improve the quality of life of citizens.

This thesis presents the HybridIoT system to cope with the lack of environmental information in the urban context through an estimation technique that integrates heterogeneous data acquired from different sensors. Our proposal allows reducing the number of physical sensing devices while ensuring that information is available at any time and anywhere. HybridIoT can be deployed in large-scale contexts and ensures data accessibility even if devices enter or leave the system at any time.

The macro-areas of interest in which our proposal takes place are: intelligent city, sustainable city and well-being city. In the first case, our proposal allows acquiring and integrating information

through ICT tools to produce an urban knowledge base; sustainable because the environmental knowledge enables experts to optimize resource consumption. Moreover, the proposed technique allows limiting the number of sensors to be installed, limiting the production costs but also the energy consumption, installation and maintenance costs related to the sensors network. Finally, we consider as smart the well-being city because the knowledge produced can be used by experts to improve both services offered to citizens and their quality of life.

The value determined by the implementation of the proposed technique is mainly economic: on a large scale, a large number of sensors would be needed to ensure a wide information coverage, so that a large amount of information can be used by government agencies to offer better services to citizens. Rather than installing a large number of sensors, our proposal allows reducing the maintenance and installation costs of sensors by using only those already present in the infrastructure, estimating accurately the information in the points where sensors are not available.

## 2.7 Conclusion

This chapter provided a general overview of the smart city, its application domains and different factors that need to be considered in the implementation of a smart initiative. The socio-economic value is the most important factor to address a sustainable development of a urban context: the implementation of a smart initiative must create benefits not only the scientific community but also the whole social community, creating attractiveness and moving cities and economic centers towards sustainable development while ensuring a high quality of life for citizens.

The next chapter introduces the problem of estimating missing values and discusses the principal solutions present in the literature. These solutions are afterward discussed according to the properties of openness, heterogeneity, large-scale introduced in the previous chapter; these properties constitute the key-advantages of our proposal.

Chapter 3

# Estimating missing information in Smart Cities

---

Objectives of this chapter:

- Discussing the role of big data in the smart city
- Introducing the *Artificial Intelligence* (AI) as the main tool to analyze and exploit big data for the implementation of smart cities
- Analyzing the main techniques for estimating missing data

---

SMART cities play a key role in transforming urban contexts by improving different aspects of the life of their citizens, such as environment, transportation, health, energy, and education. The main challenge for the implementation of a smart city is to make extensive use of data acquired on an urban scale. The amount of data acquired on an urban scale is growing significantly today, mainly because of the accessibility and low cost of devices capable of acquiring data from the environment. A large number of sensing devices (devices capable of sensing the environment) disseminated in the urban context generates an enormous volume of data or *big data*, that is at the core of the services rendered by the IoT. Big data offers the potential for the city to obtain valuable insights from a considerable amount of data collected through various sources [65]. For example, obtaining information from weather data can be beneficial for agricultural development and to inform the people in advance about the possible hazardous conditions. Data acquired from *Global Positioning System* (GPS) devices can be used to monitor traffic conditions to avoid delays, bottlenecks, and accidents. In buildings, data acquired from both environmental conditions and users' habits can be used to control *Heating, Ventilation and Air-Conditioning* (HVAC) systems to optimize energy consumption while ensuring a good quality of life for users. The analysis of big

data, therefore, remains the primary activity to ensure that the data obtained on an urban scale can be used to create social and economic value for the society.

To pursue the objectives of optimizing and improving services and consumption in cities, it is necessary not only to collect a large amount of data but also to extract useful knowledge to achieve the objectives of smartness. In this context, AI is needed to analyze, extract knowledge and reason on big data so that smart city stakeholders can decide on appropriate actions to pursue. For this to be possible, IoT and AI must operate jointly to migrate from systems of connected objects towards systems of connected intelligence. On the one hand, IoT provides means to easily acquire information from the urban context; on the other hand, AI provides tools to analyze and extract knowledge from data.

This chapter discusses the application of AI to the estimation of missing information in smart cities.

The remainder of this chapter is organized as follows:

Section 3.1 introduces the problem of estimating missing data in the smart city. Section 3.2 briefly presents the bibliographic study that we pursued to find the articles presented. Section 3.3 discusses the main techniques for estimating missing information. Section 3.4 analyzes the presented methods according to different fundamental properties, necessary for the development of efficient systems for the smart city.

The next section introduces the problem of estimating missing information in the smart city.

## 3.1 Estimating Missing Information in Smart Cities

The estimation of missing values in dataset began to develop in the 1970s as a technique to address data incompleteness. There are several reasons why it is necessary to perform this task in the smart city domain: the presence of incorrect data, non-functioning (or not available) sensors and/or the need to estimate and infer data more accurately using available data [92]. Although the lack of data can be expected in many applications, addressing the estimation of missing data is a challenge as in numerous domains it must be done in real-time and on-demand.

The estimation of missing information is mainly based on the use of mathematical models and AI techniques. Because estimated information can be used in decision-making processes, the estimation techniques must produce accurate results. Therefore, the estimates must be as close as possible to the real values to be able to make effective decisions. In large-scale contexts where estimates need to be provided at real-time, the techniques for estimating missing information needed to address issues such as **accuracy** and **timing** in an **efficient manner**.

For example, at the city scale, available information can be disseminated in environments with different physical characteristics, the sensors themselves may be inaccurate, subject to environmental factors or unpredictable failures. A system for estimating missing information must be able not only to gather all the available information from such a large context but also to provide an

accurate estimate by considering the characteristics of the environment. This is possible by using AI techniques for reasoning on data and evaluating estimates using only some of the information in the environment. This is motivated by the fact that in large scale contexts, environmental dynamics can vary considerably in different points of the environment. This implies that the system must be able to determine a subset of the information present in the environment to calculate the estimates, which may involve high computational costs. On the one hand, using a small amount of information could lead to the inability of the system to extract accurate information. On the other hand, using too much information would make the system unable to provide a precise estimate because the data would differ too much and the system would not be able to extract precise information.

There are numerous reasons why data are missing, including instrument malfunction and incorrect reporting. In order to design an efficient method to handle missing data, it is useful to understand why data can be missing [63]. This can be done by observing data distributions in order to understand whether the missing data occur regularly or not, and whether the missing data have a correlation with the available data. To this end, Rubin [116] classified missing data problems into three categories:

- *Missing Completely At Random* (MCAR): in this case, the missing samples are unrelated to the observed data. Therefore, complexities that arise because data are missing can be ignored.
- *Missing At Random* (MAR): the probability for a sample of being missing is the same within groups defined by the observed data.
- *Missing Not At Random* (MNAR): means that the probability for a sample of being missing varies for reasons that are unknown to us.

Rubin's distinction is important for understanding why some methods will not work [136], therefore this classification is relevant to conceive an efficient method to estimate missing information.

### 3.1.1 Discussion

This thesis deals with a system called HybridIoT for estimating missing environmental information. The system has been designed to operate according to the following properties:

- data are correlated during consecutive time intervals (for example: events generated from weather conditions may produce unpredictable variations);
- the sources of information can appear or disappear without any prior notification;
- the lack of data cannot be predicted in any way.

Following Rubin's classification presented in the previous section, **the problem addressed by this thesis falls into the MNAR category**: the probability of an information of being missing varies for reasons that are unknown to us, but during consecutive time intervals the data are correlated.

In the urban context in which the proposed solution is configured, the development of an effective system for estimating missing information must consider the following difficulties:

- **large amount of data**: only the data coming from a limited number of sensors should be analyzed when estimating information in a local part of the environment. Moreover, considering all the available data is not useful and can be computationally expensive;
- **data filtering**: the information perceived by the sensors may be noisy and should not be taken into account in the estimation process;
- **real-time**: analyzing and filtering data is not easy when you consider several thousands of perceived data at the city level.

The following section presents the bibliographic study used to find the articles presented.

## 3.2 Bibliographic Study

The concise bibliographic study presented in this section is motivated by the large presence in the scientific literature of articles about the estimation of missing information in smart cities. This study was carried out to motivate the context of application in which this thesis is situated by showing its wide presence in the scientific literature. Although this chapter describes some of the techniques present in the state of the art for the estimation of missing values, this study allows the reader to appreciate the large number of academic papers treating the domain of estimation in the smart city and in particular in the domain of missing information estimation.

This bibliometric study is carried out by referring to publications for which the main topic is the smart city. The publications are found in different academic databases: Google Scholar, IEEE Xplore, SpringerLink, ScienceDirect. Despite the large number of publications within the smart city context, this thesis deals only with those that focus on the estimation or integration of information at a large scale, these two topics being of central interest.

Figure 3.1 shows the growth during the last 9 years in the number of articles concerning the smart city and missing information estimation. The online academic engine *Dimensions.AI* [39] has been used to evaluate this analysis. Different keywords have been used together with "smart city": estimation, forecasting and regression.

From Figure 3.1 it can be seen that the trend of the keyword regression and forecasting curves coincide. This may be due to the presence of both keywords in the papers; regression is frequently used as a forecasting technique.

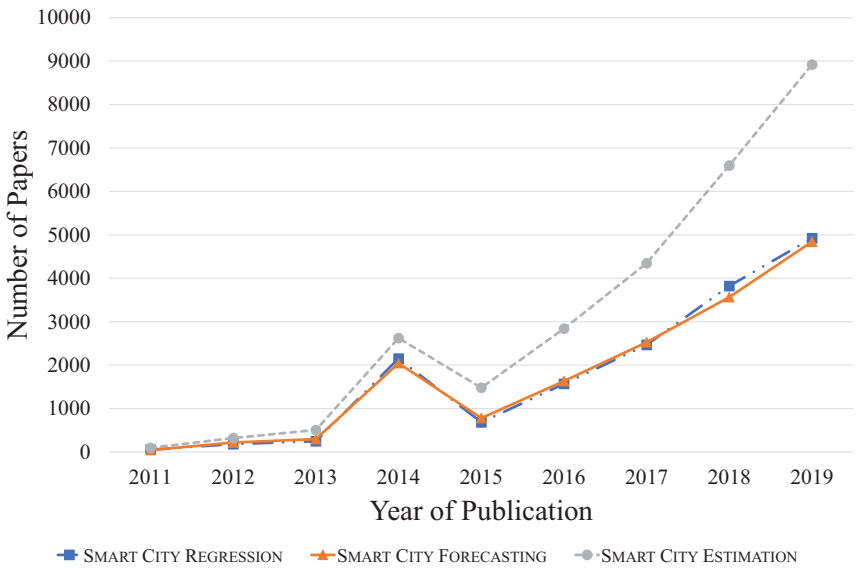



Figure 3.1: trend of the number of papers published in the last 9 years on the smart city and about "regression", "forecasting" and "estimation".

The development of an intelligent ambient system for the smart city must respect several properties that allow the resolution of complex and distributed problems, regardless of the technology used to develop the system: the **use of a large number of heterogeneous sensing devices** that may enter or leave the system at any time as well as the **management of the connections** and the **exchange of information** between sensing devices. Also, in urban contexts, ambient systems must adapt to the evolving dynamics of the environment so that they can act properly on it.

To allow an ambient system to be context-sensitive and effective for the smart city, the properties of openness, heterogeneity, large-scale, configuration, cost-effectiveness and privacy discussed in chapter 1 must be addressed. Moreover, the same properties have been used to evaluate the state of the art solutions for estimating missing information described in the following section.

## 3.3 Existing Solutions for Estimating Missing Information

This section presents the main solutions for the estimation of missing values in the smart city. The presented methods are grouped according to the technology used to estimate missing information.

### 3.3.1 Regression Techniques

This section presents the main state of the art solutions for estimating missing information on a large scale using mathematical regression techniques. Regression methods can be properly calibrated to provide accurate estimates of missing information. However, using a single regression model has disadvantages because the introduction of a new sensing device requires re-calibration of the entire

model. Likewise, significant and unpredictable perturbations on local parts of the environment would require appropriate calibration of the global model.

**REGRESSION, Spatial distribution of air pollutants in urban environments**

**Presentation** Weather stations are highly reliable and able to accurately measure a wide range of air pollutants but their installation and maintenance costs make them an unfeasible choice to raise a sufficient informative covering of the environment. Hasenfratz et al. [64] propose using a mobile measurement system capable to reduce the need for a huge number of fixed sensors at the city scale. More precisely, the system consists of ten sensor nodes installed on top of public transport vehicles, which cover a large urban area on a regular schedule. Throughout more than two years, over $50$ million ultra-fine particle measurements used to produce accurate pollution maps with high spatio-temporal resolution. The pollution maps have been produced using a *Land-Use Regression* (LUR) model. The resulting pollution maps are used to analyze how much the inhabitants of Zurich (Switzerland) can reduce their exposure to ultra-fine particles by taking a path where the concentration of the pollution is limited. This enables citizens to take a healthier route, and organization to take initiatives to reduce pollution.

The development of high-resolution pollution maps through LUR models is based on two steps:

1. linear regression is used to find dependencies between explanatory variables (e.g., population density, traffic volume, and terrain elevation) and pollution levels;

2. the relationships between the previous variables are used to predict concentration levels at locations using the available data. A Generalized Additive Model (GAM) is used to construct a LUR model. A GAM is a generalized linear model in which the linear predictor depends linearly on environmental variables.

Data has been divided into different time scales (yearly, seasonal, monthly, biweekly, weekly, daily, and semi-daily), resulting in $989$ temporal models.

**Motivation** We reported this study as it uses mobile sensors deployed on an urban scale to monitor environmental information (specifically, the air quality). The regression technique proposed by the authors enables calculating accurate maps of urban pollution that are valuable to the general public as well as to environmental scientists and epidemiologists to shed more light on the adverse health effects of ultrafine particles [64].

**Analysis** The combination of regression and accurate data acquired from *ad hoc* sensing devices specifically built for the application gives good results considering the application domain. However, there is no possibility to use any generic device to acquire the data. The openness challenge is not fulfilled as authors do not assume that devices can enter or leave the system at any time, rather they assume that they are always available. The heterogeneity is partially fulfilled as different types of information are employed in the regression model. Nevertheless, the type of information is

known in advance, so the learning model knows specifically how each information has to be treated according to its type.

#### REGRESSION, A Simple Flood Forecasting Scheme Using Wireless Sensor Networks

**Presentation** Seal et al. propose a model to predict flood in rivers using simple and fast calculations to provide real-time results [122]. Floods are responsible for the loss of precious lives and the destruction of large amounts of property every year, especially in the poor and developing countries.

The proposed sensor network system architecture consists of sensor nodes, computational nodes and a central monitoring office. Sensors nodes collect data from direct perception of the environment, computational nodes implement the prediction algorithm and the monitoring office verifies the results from computational nodes, implements a centralized version of the prediction algorithm as a redundancy mechanism, issues alerts and initiates evacuation procedures. Different types of sensors are required to sense water discharge from dam, rainfall, humidity, temperature, etc. The data collected by these sensors are used in the flood prediction algorithm performed by computational nodes, which have sufficient computational power to implement the prediction model in a distributed manner.

A linear regression method and a quadratic fit function are used to predict future values. The computational nodes pursue prediction and communicate the results to the monitoring office, where a human operator decides to eventually initiate evacuation procedures based on the prediction results.

**Motivation** We report this study because the proposed application shows how the real-time acquisition and processing of information acquired by sensors deployed on a large-scale can predict floods, this prediction being useful to save the lives of people who may be affected by the flood.

**Analysis** The solution relies on a network of sensors known in advance and whose operations and information are handled hierarchically by central nodes. In the proposed solution, nodes have communication between themselves for detecting malfunctioning.

The main drawbacks of the proposed solution are the scalability, not treated by the authors, and the openness: although data is collected and processed online, adding a large number of sensors could compromise the performance of the proposed solution. Moreover, the solution does not allow the integration of information whose type is different from those supported, as this would require a change to the proposed prediction procedure.

#### REGRESSION, Fusing Incomplete Multisensor Heterogeneous Data to Estimate Urban Traffic

**Presentation** Shan et al. [123] authors propose a robust fusing method to improve the accuracy of traffic state estimation using incomplete multi-sensor data. Using a *Multiple Linear Regression* (MLR) model on three road segments, they analyze the historic GPS data detected from taxies on those segments and extract the spatio-temporal correlation of traffic states (such as speed).

In the first step, the MLR model extracts the spatio-temporal correlations of traffic states for the three road segments. The speed correlations of road segments in candidate region scales are extracted from GPS data because the GPS data collected by probe vehicles are the most complete in different types of multi-sensor heterogeneous traffic data. Based on the correlations, the missing speed data of the target road segment can be estimated without deploying a particular type of sensor using the correlated road segments deploying that type of sensor. This estimated speed is fused with GPS data to achieve a more accurate traffic state. Then, the candidate correlated roads are chosen according to the number of connected road segments deploying the type of sensor, and their distribution. After determining the candidate region scales, the speed of the target road segment can be estimated using the data collected on input and output road segments.

**Motivation** We reported this study as it constitutes an important advance in estimating missing information on an urban scale. More specifically, the most important aspect that we report is the ability of the estimation technique proposed by the authors to integrate heterogeneous information to provide accurate estimates of missing values.

**Analysis** The authors affirm that the parallel implementation of their solution can facilitate traffic state estimation on massive incomplete data to achieve real-time performance. Nevertheless, the scalability of the solution is limited, as the time required to calculate an estimate depends on the number of computational nodes present in the system. For an estimation time of fewer than $500$ seconds, at least $32$ computational nodes are required. Moreover, it does not seem possible to add new computational nodes in real-time in the system, so that the problem of openness remains unsolved.

The problem of heterogeneity is partially satisfied because the information used is known *a priori*.

### REGRESSION, Evaluating the Health State of Urban Areas Using Multi-source Heterogeneous Data

**Presentation** Tomaras et al. [133] propose ”HELIoS” (HEalthy LIving Smart), a framework that combines multiple heterogeneous sources of data such as urban traffic and pollution data to diagnose the health state of urban areas in a smart city. The developed solution is currently deployed in the City of Dublin and aims at monitoring diverse data coming from city-wide infrastructures and recognize in real-time abnormal events of interest such as traffic conditions and the air pollution levels. In the proposed solution, traffic data are received from various voluminous sources, including cameras CCTV, static loop sensors and bus sensors that measure the traffic flow, the data are possibly noisy, with missing values and measurement errors. In Dublin, the air quality is measured city-wide through 6 stations deployed in the city center and its suburban areas to provide air pollution information to its citizens. The data are acquired each hour from $3402$ heterogeneous fixed sensors deployed in different urban areas.

The proposed system is based on the use of a regression model that combines different environ-

mental indices such as air quality, road saturation, environmental conditions, defined through a series of equations, to accurately classify the city area into health levels.

The proposed regression model is divided into three steps:

- **Data aggregation**: the data measurements for each sensor are aggregated by computing the mean value of the degree of saturation metric every hour. In this way, the data are transformed into a time series with 24 measurements per day.
- **Feature selection**: three different approaches are used: (*i*) most correlated, which selects the most similar pollution measurement, (*ii*) spatially close, which selects the nearest sensors and (*iii*) all the sensors, which includes all the sensors without any restriction.
- **Regression**: a machine learning approach has been used to estimate the pollution data using the available information from sensors and the historical data of observed pollution measurements. During the training process, the goal is to identify the model parameters that result in the minimum estimation error. A variety of regression methods has been used and, finally, the one that performs better (in terms of minimization between data and predicted outcome) is chosen for the pollution estimation task. Three different regression models have been used to predict the health state of urban areas:
  - **Support Vector Regression** (SVR): a regression variant of the support vector machine classifier;
  - **Random forest**: a set of multiple decision trees where each tree is trained using a different subset of the training set;
  - **Gaussian process**: a non-linear and non-parametric model that is an extension of the multivariate Gaussian distribution for an infinite collection of real-valued variables.

**Motivation** We report this study as it highlights the importance of integrating heterogeneous information to estimate relevant information in the urban context such as the urban traffic, the pollution and to diagnose the health state of urban areas in a smart city.

**Analysis** The traffic data are collected in the period between November and December 2015. The pollution data are acquired by sensors deployed in the urban area of Dublin. The system does not provide any resilience mechanism that allows sensors to enter or leave the system at any time. Therefore, if any sensor is not available due to an unpredictable cause, such as a malfunction, the system has to reconfigure itself. For this reason, the challenge of openness is not addressed.

#### REGRESSION, A Refinement of Lasso Regression Applied to Temperature Forecasting

**Presentation** Spencer et al. [129] propose a technique based on regression to predict the temperature in a smart home equipped with 88 sensors and 49 actuators. Sensors are capable of perceiving every quarter-hour heterogeneous information such as temperature, wind speed, light, $CO_2$. This solution

uses the R library $glmnet$, containing a package that fits generalized linear models via penalized maximum likelihood.

Authors overcome forecast inaccuracy that arises from the "one standard error" heuristic (1SE) in lasso regression, a regression analysis that enhances the prediction accuracy through a variable selection and regularization. They propose a refinement of lasso regression called *midfel*, based on linear regression. In lasso regression analysis, the relations between variables generate a curve that best fits particular data. Unlike lasso regression, the midfel refinement of lasso regression also uses the shape of the error curve and it is particularly effective when the forecast model is based on a large amount of data.

**Motivation** We report this study as the technique proposed by authors enables saving energy in a smart building by controlling heating, ventilation, and air conditioning equipment while achieving comfort for occupants. The technique proposed by the authors uses historical data (up to 24 hours) to forecast accurate temperature information.

**Analysis** The experiments have been conducted on a fixed number of sensors. The proposed solution does not foresee the use of sensors that can enter or leave the system, so the challenge of openness is not addressed.

The scalability is not satisfied because an implementation on an urban scale would imply that the model can process a significant number of variables that derive from a large number of sensing devices; this does not seem to be possible because a linear regression model would require a considerable amount of computational time.

### 3.3.2 Neural Network Techniques

This section presents the main state of the art solutions for estimating missing information on a large scale using neural networks.

#### NN, real time Forest Fire Detection Using Wireless Networks

**Presentation** Yu et al. [152] propose a real-time forest fire detection method by using wireless sensor networks and neural networks. The goal of this work is to predict forest fire promptly and accurately in order to minimize the loss of forests, wild animals, and people in the forest fire. The proposed solution makes use of a large number of sensor nodes deployed in the forest and organized into clusters so that each node has a corresponding cluster header. The clusters are formed according to parameters such as the proximity between nodes. The clustering technique has the objective of reducing energy consumption and ensure load balancing in network transmission. Figure 3.2 shows the proposed sensor network paradigm.

Sensor nodes can measure environment temperature, relative humidity and smoke. Also, they are equipped with a GPS module so that the position of the sensors is known. Every sensor node sends measurement data and their location to the corresponding cluster head. The corresponding

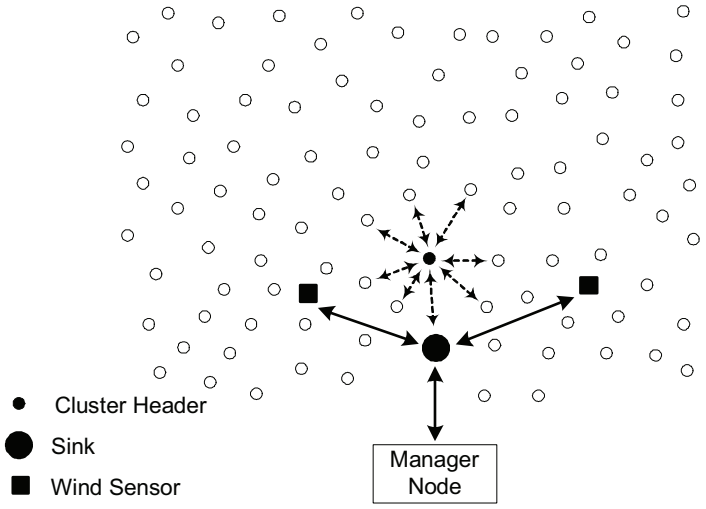



Figure 3.2: The sensors network for real time fire detection (Yu et al. [152]).

cluster header calculates the weather index $F$ using a neural network method (which structure is similar to a feed-forward neural network), then it further sends the weather index to the manager node via the sink node. A sink node collects and processes data (Figure 3.2). The weather index is calculated as follows:

$$\begin{aligned} &f(T_{max24}w_1, T_{min24}w_2, T_{avg}w_3, H_{max24}w_4, H_{min24}w_5, H_{avg}w_6) = \\ &T_{max24}w_1 + T_{min24}w_2 + T_{avg}w_3 + H_{max24}w_4 + H_{min24}w_5 + H_{avg}w_6. \end{aligned} \tag{3.1}$$

where $T_{max24}$ and $T_{min24}$ denote the highest temperature and the lowest temperature in the past $24$ hours respectively. $H_{max24}$ and $H_{min24}$ denote the highest relative humidity and the lowest relative humidity in the past $24$ hours respectively. $T_{avg}$ and $H_{avg}$ denote current average temperature and the average relative humidity respectively.

The sink is connected to a manager node via a wired network. A few wind sensor nodes are manually deployed over the forest and connected to the sink via wired networks to detect wind speed. All the cluster headers periodically calculate a weather index F that is sent to the manager node, responsible for evaluating the presence of fire danger using the received weather indexes.

**Motivation** We report this study because the authors focus on real-time detection of forest fire using the information acquired at a large-scale. The technique proposed by the authors enables grouping sensors according to properties such as their proximity. Grouping the sensors can be beneficial in providing accurate and local information on the presence of fire.

**Analysis** In a critical application context such as fire detection, a fault-tolerance mechanism should be required to enable the system to detect and eventually solve problems that arise from sensors malfunctions. In the case of the proposed solution, a malfunction of a sensor in the network would require, once the problem has been identified, a reconfiguration of the nodes, their

connections and also a reconfiguration of the neural network used to determine the presence of fires. The openness property can overcome this problem, allowing devices to enter or leave the system in a transparent way, without compromising its operation.

The scalability is partially satisfied. The proposed method has been validated on fields of different sizes (from $500m^2$ to $860m^2$) using a different number of nodes (from $50$ to $150$).

**NN, Short term traffic flow prediction for a non urban highway using Artificial Neural Network**

**Presentation** Kumar et al. [81] apply an Artificial Neural Network (ANN) for short term prediction of traffic flow using past traffic data. The model incorporates traffic volume, speed, density, time and day of the week as input variables. The speed of each category of vehicles was considered separately as input variables in contrast to previous studies reported in the literature which consider the average speed of combined traffic flow. The main objective of the proposed model is to investigate the stability and efficiency of neural networks for short-term prediction of traffic volume in the case of mixed Indian traffic flow conditions.

A Multilayer Perceptron (MLP) network has been used for prediction of traffic flow data $15$ minutes in the future using the past $45$ minutes data. $160$ data sets have been taken for analysis, each of which contains $19$ features: day of the week, time of the day, the category of vehicles, the corresponding average speed of vehicles and traffic density were among these features. The whole database was divided into three parts for training, cross-validation and testing in the ratio $60$, $15$, $25$ percent respectively. First, $45$ minutes data (extracted in intervals of fifteen minutes) of each hour were used for the training/validation stage and the last $15$ minutes data for testing purposes. Thus $96$, $24$, $40$ data sets were used for training, cross-validation and testing the ANN models respectively.

**Motivation** We report this article as it shows how neural networks, in particular the multilayer perception networks, can be employed for short term traffic flow prediction up to 15 minutes.

**Analysis** As stated by the authors themselves, one of the major limitations associated with ANN modeling is its black box type nature which prevents the system to find the mutual interrelation between the variables in ANN modeling.

The proposed model has been validated on a limited set of previously observed data. Moreover, the proposed model does not allow processing large amounts of urban data in real-time and also it is not possible to consider the acquisition from intermittent sensing devices that can enter or leave the system at any time, without causing a reconfiguration of the system. Thus, the openness challenge has not been addressed.

The heterogeneity is partially satisfied as the solution employs a limited set of information which type is known in advance. Also, the scalability is not addressed as the data are acquired from an area of limited size.

**NN, Indoor Air-Temperature Forecast for Energy-Efficient Management in Smart Buildings**

**Presentation** Aliberti et al. [8] present a methodology for indoor air-temperature forecasting obtained by exploiting a Non-linear Autoregressive neural network. The authors designed, trained and validated this neural network with a dataset consisting of six years of indoor air-temperature values.

The autoregressive neural network is basically a multilayer perceptron exploiting a high number of regressors to predict the temperatures in a time interval of 15 minutes, allowing to study the trend of air-temperatures up to about two hours onwards. The employed neural network architecture characterized by *(i)* one hidden layer of neurons with hyperbolic tangent activation function and (*ii*) an output layer with a linear activation function. The network is subjected to a training phase that allows determining a mapping from the set of training data to the set of possible weights. In this way, the network can produce prediction, to be compared to the true output.

**Motivation** We report this article as the proposed technique enables predicting (up to 3 hours) accurate temperature values using both historical data and data acquired by IoT sensing device using a multilayer perception network and multiple regressors. This article shows the ability of standard techniques to use environmental data to estimate information in different environments of a building.

**Analysis** The proposed method does not consider the possibility that the sensors present in the building may encounter problems. In this case, a reconfiguration of the system would be necessary to ensure its correct functioning. Therefore, the challenge of openness is not met. The heterogeneity is not addressed because in the proposed model only the temperature trend is considered. Moreover, the scalability does not seem to be addressed: the proposed model is specifically designed to work in the application context proposed by the authors. It is therefore not possible, unless of modifying the model, to deploy the system on an urban scale.

**Soft Sensor, Soft Sensor with Deep Learning for Functional Region Detection in Urban Environments**

**Presentation** Ma et al. [88] proposed a technique to identify the functional region of subway stations by combining different types of information. Urban functional regions can be defined by some types of activities or spatial interactions that may occur in a region. The contribution is as follows:

- integrated data from the *Online to Offline* (OTO) e-commerce platform data, *Smart Card Data* (SCD) and Point Of Interest (POI) data to reduce the impact of a lack of data diversity. OTO is an emerging e-commerce model where consumers are able to search for and buy services or products online, and then consume them in an offline store [104]. The combination of multiple information enables obtaining more rigorous results.
- constructed a passenger flow feature map that contains specific information of each subway station and can reflect their characteristics.

- created a hybrid neural network approach that allows input vectors and maps to simultaneously consider the features of data and the need for data.

The proposed technique consists of two stages:

1. the SCD feature is extracted and converted to a feature map, and a ResNet model is used to get the output.
2. the POI and OTO features are extracted, and a deep neural network with stacked autoencoders (SAE–DNN) model is used to get the output, identifying functional regions of subway stations.

The outputs of the two stages are connected and a SoftMax function is used to make the final identification of the functional region.

**Motivation** We report this article as the technique proposed by the authors uses the information acquired on an urban scale to determine functional areas for subway stations (the residential region, the work region, the recreation region, and the transportation region) using a soft-sensor approach. This article shows the variety of fields of application in which the estimation of information can be useful in the development of smart initiatives.

**Analysis** The technique makes use of a real dataset known before the execution. The method does not consider data introduced during the system functioning, noisy or missing data. The openness property is not addressed: the use of new devices requires a system configuration. The heterogeneity property is partially satisfied: the technique uses different types of information, although their type is known in advance. A proper configuration of the system is necessary to introduce new types of information.

#### ANN-Based Soft Sensor to Predict Effluent Violations in Wastewater Treatment Plants

**Presentation** Wastewater treatment plants (WWTPs) reduce water's pollutant products, which are harmful to the environment at high concentrations. Pisa et al. [110] proposed an artificial neural network (ANN)-based soft sensor in which a Long-Short Term Memory (LSTM) network is used to generate predictions of nitrogen-derived components, specifically ammonium ($S_{NH}$) and total nitrogen ($S_{Ntot}$). $S_{Ntot}$ is a limiting nutrient and can therefore cause eutrophication, while nitrogen in the $S_{NH}$ form is toxic to aquatic life. These parameters are used by control strategies to allow actions to be taken in advance and only when violations are predicted.

The main contributions of this work are:

- design of a soft sensor based on ANNs (LSTM structures) to predict WWTP's effluent concentrations.
- treatment of online data as the unique source of information to predict effluent limits in real time.

- application of a data preprocessing techniques to improve the LSTM predictions.
- development a prediction system able to obtain a low error.

Soft sensors' input and output data consist of online measurements with the exception of total nitrogen ($S_{Ntot,e}$). To predict effluent values with a prediction horizon of 4 hours, the authors used data collected over $10$ hours, using data windows containing 4 samples collected for each hour.

**Motivation** We report this article as it shows how neural networks can be employed in critical domains such as the prediction of nitrogen-derived components in wastewater treatment plants. The technique proposed by the authors is an effective tool especially if coupled with measuring instruments to provide accurate indications in real-time on the status of treated water.

**Analysis** Due to the application context in which the proposed technique takes place, the information must be acquired from devices specifically configured by the expert. Although the openness property is not respected, this is not particularly beneficial in the application context. The property of heterogeneity is partially satisfied: the information is of different types but the estimation of nitrogen-derived components requires a specific technique to combine information such as ammonium and nitrogen.

### 3.3.3 Gradient Boost Technique

This section presents a state of the art solution for estimating missing information on a large scale using boosting techniques, which combine multiple learning algorithms into a single learner [121]. Further details on the gradient boost techniques are reported in Appendix A.

#### A gradient boosting method to improve travel time prediction

**Presentation** Zhang et al. [155] propose a *Gradient Boosting regression tree Method* (GBM) to predict travel time on a freeway stretch by considering all relevant variables derived from historical travel time data.

The real-word travel time data used in the proposed solution aggregate traffic data from probe vehicles and traditional sensor sources. Probe vehicles utilized include: taxis, airport shuttles, service delivery vans, long-haul trucks, consumer vehicles, and GPS enabled consumer smartphones and so on. Traffic sensors range from inductive-loop detectors, radar sensors, to toll tag readers.

The data present only a percentage of $1\%$ of missing samples; given the small amount of missing values, this study simply replaced the missing values with the mean of its closest surrounding values.

The ten variables that are used as input to predict travel time at time step $t$ are as follows: $\text{TT}_{t-1}$, $\text{TT}_{t-2}$, $\text{TT}_{t-3}$ are the three most recent travel time observations at time steps $t-1$, $t-2$ and $t-3$, $\Delta\text{TT}_{t-1} = \text{TT}_{t-1} - \text{TT}_{t-2}$ is the growth rate over two consecutive time steps $t-1$ and $t-2$, time of day is represented by every five minutes time step indexed from $1$ to $288$, the week is indexed from

0 to 6 to represent from Sunday to Saturday, the day is the day when the observation is detected (from 1 to 31), and the month is the month information for the observation (from 1 to 12).

**Motivation** We report this article as it shows how the gradient boosting methods can be employed for the travel time prediction. Moreover, this study shows the capability of the gradient boosting model of handling different types of input variables and modeling nonlinear relationships between variables.

**Analysis** The authors state that there is also a need to consider the trade-off between prediction accuracy and computational time. Since a large number of trees are being fitted, model complexity also increases and requires more computational time. If the number of sensors increases, and consequently the amount of data, the number of regression trees may increase considerably and consequently a high computation time can be necessary. Therefore the scalability challenge could not be considered as addressed for this solution.

The authors do not consider the possibility of processing data in real-time and they use the information on the travel time in the observed road segments. The challenges of heterogeneity and openness are therefore not addressed.

### 3.3.4 Combined Techniques

This section presents the main state of the art methods for estimating missing information on a large scale by combining different AI methods.

#### REGRESSION+NN, Granger-Causality-based air quality estimation with spatio-temporal heterogeneous big data

**Presentation** Zhu et al. [156] deal with city-wide air quality estimation with limited available monitoring stations that are geographically sparse and discovering Regions Of Interest (ROI) to produce an accurate air quality map for the city of Shenzhen, China. The idea behind the proposed solution is to estimate the air quality at locations not covered by monitoring stations. They propose to analyze the temporal dependency and spatial correlation between urban dynamics data, such as meteorology and traffic. These dependencies, or causalities, are expressed based on Granger causality, which represents the causality between two time series in a regression manner and determines a "Granger" cause if one time series can successfully predict another.

The prediction of air quality data is divided into five stages: the first stage interacts with input data flows, from both online and historical data; the second stage deals with non-causality detection, which is based on the spatio-temporal extended Granger causality model. Granger causality test analyzes the causalities between urban dynamics and the air quality. A causality between two time series enables to determine if a time series can successfully predict another series. The causality between two time series is calculated as follow:

$$Y\left(s_y;t;c_j\right)=\sum_{k=1}^{L_j}a_kY\left(s_y;t-k;c_j\right)+\sum_{i=1}^{N}\sum_{k=0}^{L_i}b_{ik}X\left(s_x;t-k;c_i\right)+\sum_{i=1}^{N}\sum_{k=0}^{L_i}r_kZ_{i,t-k}+\xi_t \tag{3.2}$$

where $\xi_t$ are uncorrelated random variables with zero mean and variance $\sigma^2$, $L$ is the number of time-stamps, vectors $\mathbf{a}=\{a_k\}$ and $\mathbf{b}=\{b_k\}(k=1,2,...,L)$ are the correspondent weights for two processes $\{X_t\}$ and $\{Y_t\}$.

The authors aim to generate an air quality map for the city of Shenzhen to help people understand better about urban dynamics. However, processing the massive volume of urban dynamics data poses a challenge. To overcome this problem, the authors propose an approach to discover the *Region Of Influence* (ROI) by selecting data with the highest causality levels spatially and temporally, thus to process "part" of the data instead of "all" the data. ROI are defined as a set of grids that have the top $10\%$ highest causality considering a precise urban dynamic.

In the third phase of the solution, a causality factor is calculated for each grid. The fourth phase of the solution combines ROI detection with historical air quality estimation, which gradually trains the parameters of the neural network used for predicting the air quality. After inferring the air quality index in each grid, in the last phase of the solution, a fine-grained air quality map is generated.

**Motivation** We report this article as it shows how it is possible to combine different techniques, precisely the regression and the neural networks, to estimate environmental information on an urban scale, in parts of the environment not covered by monitoring stations by combining heterogeneous information.

**Analysis** The proposed solution does not take into account the openness, as data are acquired from a limited number of fixed stations. Therefore, it seems to not be possible to employ sensors that can enter or leave the system at any time. The problem of heterogeneity is partially satisfied: the predicted information on air quality derives from the integration of different types of information (air quality, meteorology, traffic). Nevertheless, the authors do not foresee the possibility to estimate types of information different from air quality.

#### REGRESSION+NN, Machine learning methods to forecast temperature in buildings

**Presentation** Mateo et al. [93] apply different machine learning techniques for predicting the temperatures in different rooms. The data used have been obtained by simulating, through the software TeKton 3D, a building located in the province of Málaga (Spain). The obtained results demonstrate the validity of these techniques for predicting temperatures and, therefore, for the establishment of optimal policies of energy consumption.

The authors used the following methods to predict temperature values:

- *Autoregressive model*: it considers a certain number of past samples to predict a temperature value;
- *Multiple Linear Regression (MLR)*: it fits a data model that is linear in the model coefficients, which are typically estimated by least-squares, and can approximate both lines and polynomials, among other linear models.
- *Multilayer Perceptron (MLP)*, which produces a non-linear mapping of a set of input data to one or more outputs in an adaptive way, that is accomplished by learning from examples. Typically, the learning is done by the back-propagation algorithm or one of its variants.
- *Extreme learning machines*: these are feedforward neural networks that can be used for regression with one or multiple layers, where the parameters of hidden nodes (not just the weights connecting inputs to hidden nodes) need not be tuned. The hidden nodes are randomly assigned, never updated or inherited from the nodes of the previous layer. The weights of the output weights are learned once and represent the amount of learning of the model.

The authors also make use of different clustering techniques to help to arrange and combine samples in the initial population according to their similarities, to remove outliers and to provide more homogeneous datasets. This clustering step, applied before building the prediction model, can improve the overall performance.

**Motivation** We report this article as it provides an overview of different methods applied in the context of the estimation of environmental information in different rooms of a building.

**Analysis** The heterogeneity challenge is partially satisfied as the information used is of different types. However, the types of information are known in advance and the introduction of a new type of information would imply a modification in the prediction model.

The performance of the proposed system is based on the processing of the available data, and the use of data acquired in real-time is not foreseen. Moreover, the used sensing devices are always available, so the challenge of openness is not addressed.

The scalability is not addressed because the model has been deployed in a limited application context. Its application in an urban environment would require significant computational resources to ensure that the estimated information is available in a short time.

### ALL Machine Learning Algorithms for Short-Term Load Forecast in Residential Buildings Using Smart Meters, Sensors and Big Data Solutions

**Presentation** Oprea and Bara [103] propose a framework to determine the load profiles and forecast the electricity consumption for residential buildings for the next 24 hours. The proposed methodology for Short-Term Load Forecast (STLF) consists of three stages that imply data pre-processing, validation, storing, further processing and analysis with ML algorithms.

In the first stage, data is gathered from smart metering devices, combined with meteorological data that were collected from the weather sensors or web APIs. To validate the readings, an *Extract, Transform and Load* (ETL) process is applied and then the data is loaded into a NoSQL database.

In the second stage, data analysis is performed to identify the most significant attributes that influence electricity consumption. Also, the $k$-means clustering method is used to group the electricity consumers into consumption groups with similar behaviors. Both clusters and the most significant attributes are considered as input for the machine learning algorithms.

In the third stage, the STLF for the next $24$ hours is performed. As a novelty, a Feed-Forward Artificial Neural Network (FF-ANN) algorithm is proposed with an enhanced learning method by introducing back-tracking for adjusting the learning rate and reducing the computational time especially in case of large data-sets. To compare its performance, six other powerful and competitive machine learning algorithms are implemented, i.e. FF-ANN with enhanced gradient descend methods.

The training method relies on seven machine learning algorithms (i.e. three FF-ANN algorithms, Non-linear AutoRegressive with eXogenous (NARX), Deep Neural Network (DNN), Gradient Tree Boosting (GTB) and Random Forests (RF)) that simultaneously run to select the best performing algorithm and estimate the electricity consumption for the next day.

To obtain the best results, in the proposed methodology, all seven ML algorithms are simultaneously executed and the best performing algorithm in terms of accuracy is automatically selected to forecast the electricity consumption.

**Motivation** We report this article because it enables estimating the power consumption using simultaneously multiple machine learning techniques to obtain accurate estimates.

**Analysis** The authors used for the training and validation of the proposed method a data set containing about $6$ millions of information acquired in $2015$. Although the sensors were installed the previous year, the authors did not consider the data from $2014$ because some records were missing or inconsistent; to perform correctly, the framework must be given in input with a consistent dataset. The framework, therefore, does not satisfy the openness property because it is not able to handle new input at any time without the need for any reconfiguration during its operation.

The proposed framework uses different types of information (temperature, wind speed, humidity, precipitation, atmospheric pressure) that are used only for estimating the electrical load. The heterogeneity challenge is therefore partially addressed because it is not possible to estimate other types of information beyond the electrical load unless the system is specifically configured (by performing a training phase for the machine learning algorithms).

## 3.4 Discussion

Table 3.1 lists the described methods along with their strength and weak points according to the following properties: openness, large-scale, heterogeneity. We used four indicators to depict

strength and weak points of each described method: (++) a challenge has been discussed and authors describe a precise method to address it, (+) a challenge has been discussed and addressed but authors did not provide a precise explanation of the solution, (-) the challenge has been mentioned but not addressed, (– –) the challenge was neither mentioned nor discussed.

Table 3.1: Comparison of state of the art solutions for estimating missing information.

| Technology | Authors | Domain | Openness | Heterogeneity | Large-Scale |
|---|---|---|---|---|---|
| Regression | Hasenfratz et al. (2015) | Environment | - - | ++ | + |
| | Seal et al. (2012) | | - - | - - | + |
| | Shan et al. (2016) | Urban Traffic | - - | ++ | + |
| | Tomaras et al. (2018) | Smart home | - - | + | + |
| | Spencer et al. (2018) | | - - | - | - - |
| NN | Kumar et al. (2013) | Urban Traffic | - - | ++ | + |
| | Yu et al. (2005) | Environment | - - | - | - - |
| | Ma et al. (2020) | | - - | ++ | + |
| | Pisa et al. (2019) | | - - | ++ | - - |
| | Aliberti et al. (2018) | Smart Home and Building | - - | - - | - - |
| Regression+ NN | Zhu et al. (2015) | Environment | - - | ++ | + |
| | Mateo et al. (2013) | Smart Home and Building | - - | - | - - |
| Gradient Boost | Zhang et al. (2015) | Urban Traffic | - - | - - | - - |
| Combined | Oprea and Bâra (2019) | Smart Home and Building | - - | ++ | - - |

We do not report the properties of privacy, proactivity and economy in table 3.1 because the discussed methods do not address these properties.

## 3.5 Conclusion

This chapter provided a general overview of the problem of estimating missing information. The state of the art solutions were presented and discussed according to the properties of openness, heterogeneity, scalability. We have reviewed that the systems discussed do not allow to satisfy the property of openness, and therefore the ability to manage intermittent sensors, i.e. that can enter or leave the system at any time. The heterogeneity is partially satisfied by some solutions, as they manage heterogeneous information but whose type is known in advance.

The proposed HybridIoT system in this thesis addresses these properties to be met, which makes our proposal a significant technological advance over the state of the art solutions. HybridIoT is based on the multi-agent paradigm, which is introduced in the next chapter.

Chapter 4

# Multi-Agent Systems

Objectives of this chapter:

- Introducing multi-agent systems
- Presenting the properties of self-organization and cooperation for agent-based methods
- Presenting the adaptive multi-agent systems for solving complex problems in smart cities

CITIES are truly complex systems because of their evolving dynamics, unpredictability and non-linear relations between urban and environmental phenomena. A technological means that allows cities to be ”smart” must satisfy these properties. Ambient systems are capable of responding to the dynamic evolution of the environment. In this context, citizens themselves become part of the systems, as they are capable of providing information about their activities through mobile devices such as smartphones.

Engaging the user into the ambient system has an important drawback: ambient systems must face the challenge of the diffusion of sensing devices and consequently the huge amount of information that can be collected and analyzed. A centralized system for collecting and analyzing data would require considerable computing power given the amount of information that can be collected on a city scale. Furthermore, due to a large number of sensing devices present on an urban scale, it is difficult for a centralized system to respond adequately to the continuous evolution of the environment and be resilient, anticipating all possible malfunctioning situations.

In the context of missing information estimation, mobile and intermittent IoT devices can be used jointly with *ad hoc* sensing devices to ensure a large informative coverage in the urban environment. But how mobile, intermittent sensors can be integrated into an efficient infrastructure for providing environmental information at large-scale, reducing the overhead due to centralized computing and providing sufficient information coverage in cities? Using mobile sensing devices implies an

ambient system for estimating missing information to be resilient, to be capable of considering the mobility of sensors, their openness (intended as the capacity of managing sensing devices that can enter or leave the system at any time) and ensuring that information is available even in case of sensors unavailability. Therefore, it is difficult to conceive a system that responds adequately to such complex dynamics. Anticipating all possible system functionalities is difficult, especially when such a system has to be deployed into cities. Moreover, in such a context, the lack of transparency of machine learning-based solutions is an unacceptable condition: understanding the outcome or the behaviour of machine learning technique can be difficult [25].

A relevant approach to address such complex behavior is the **Multi-Agent Systems** (MAS), that allows IoT devices to become "intelligent" agents, capable of perceiving the environment, to analyze data and to respond adequately to unpredictable situations that could potentially prevent the sensing devices to provide information.

This chapter presents multi-agent systems, and in particular the adaptive multi-agent systems. After the presentation of this approach, its relevance will be argued with respect to the properties required for the development of the smart city described in the previous chapter.

This chapter is organized as follows: section 4.1 introduces the multi-agent systems for developing distributed applications to handle the complexity of open and evolving environments. Section 4.2 discusses the concept of cooperation in multi-agent systems. Section 4.3 introduces the **Adaptive Multi-Agent System** (AMAS) approach. This approach allows designing multi-agent systems able to solve complex problems in a bottom-up way thanks to a cooperative behavior between agents. Section 4.4 shortly introduces the ADELFE methodology, which contains useful guidelines for designing AMAS and verifying their pertinence concerning a problem.

## 4.1 Multi-Agent Systems

In traditional information systems, programs are developed to respond deterministically to all possible anomalous situations that the system may encounter. This implies that the designer can determine the entire set of anomalous situations for the system. If, however, the system runs into an unpredicted anomalous operation, the system may crash, which can also lead to the loss of relevant data. For an increasing number of applications, such malfunctions cannot be allowed. On the one hand, today's applications are becoming more and more complex, which makes it difficult to identify all possible anomalous situations that a system may encounter. On the other hand, for an increasingly large number of applications, we require systems that can decide by themselves what they need to do to satisfy their design objectives. Such computer systems are known as **software agents** or simply **agents**. Agents that operate in rapidly changing, unpredictable, or open environments, where there is a significant possibility that actions can fail are known as intelligent agents, or sometimes autonomous agents [145].

The MAS paradigm offers an intuitive and natural way to solve a complex problem through

a distributed computation between interacting autonomous agents that have specific and limited tasks. Agents operate jointly to achieve a global objective that cannot be pursued individually. Agents have different skills and social behaviors to interact with other agents. When interacting, agents can autonomously constitute organizations in which they operate jointly to contribute to a common goal that cannot be achieved individually.

The MAS paradigm defines a model for software systems composed of two components: a set of agents and an environment.

### 4.1.1 Agents

Although there are different definitions of agent in the literature, we report the one provided by Weiss [145]:

**Definition 1.** *An agent is a computer system that is situated in some environment, and that is capable of autonomous action in this environment to meet its design objectives.*

In this thesis we define agents as autonomous computational entities located in an environment, they have operational and social capabilities. The set of actions available to an agent allows it to modify autonomously the environment in which it is situated. Agents generally have limited capacity, that is, an agent cannot solve the goal alone and can work with other agents to do so. Agents are able to create social relationships with other agents which results in a coordinated, joint activity that allows a group of agents to contribute to a common goal.

Agents have a partial view of the environment in which they are situated (the context in which the agents act; can be either virtual or physical). As a result, they have limited control, they can only influence a part of the environment (including agents) through their actions. Agents are also capable of interacting with other agents by engaging in a social activity that we all engage in every day of our lives: cooperation, coordination, negotiation, etc. [150].

From this definition we deduce some fundamental characteristics of the agents:

- agents are autonomous, there is no need for external interventions to decide their behavior;
- agents can be situated into an environment: *situatedness* is the property of being potentially influenced and, in turn, potentially capable of affecting someone or something [52];
  - we make a distinction between *physical* and *social* environment: the first is the real environment from which agents perceive data through sensing devices, while the second is the "virtual" environment constituted of agents and where interactions between agents take place.
- agents can evolve in their environment, adapting and acting in it according to environmental conditions;

- agents have social skills, can communicate, form organizations and act jointly. Generally, agents do not own all the skills necessary to pursue the goal the multi-agent system is designed for, but rather they have a limited set of skills and knowledge;
- agents have a partial view of the environment, consequently, their action affects a limited part of the environment. This implies that a modification of a part of the environment could not be perceived by all the other agents.

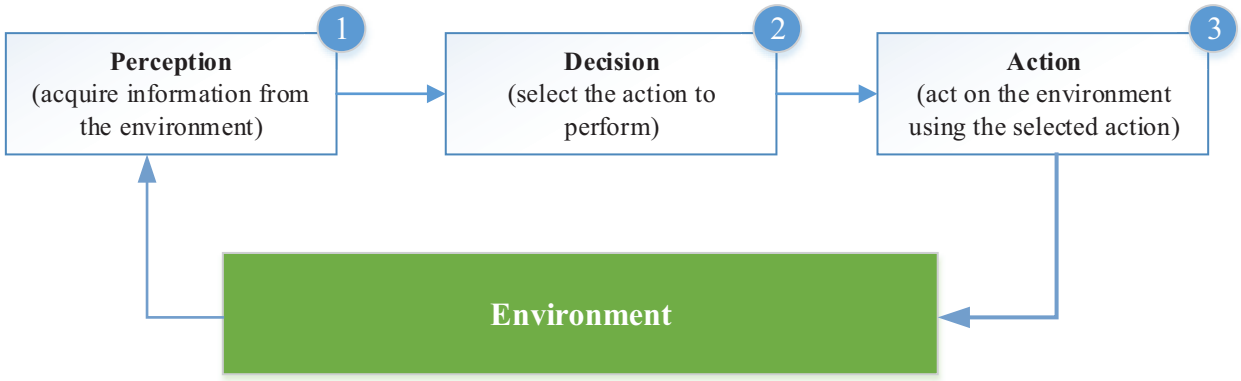



Figure 4.1: An agent's lifecyle.

Each agent has one or more objectives that can be expressed in different ways, such as mathematical functions or through predicates. The first case is useful when agents deal with numerical information, as in the case of estimating missing information. The second case is widely used in the intelligent management of services that cannot be expressed through mathematical models, such as web or urban services [119]. Goals can also be distinguished into hard goals and soft goals: hard goals describe services the MAS is expected to deliver whereas soft goals refer to nonfunctional properties such as performance or excellence issues [18].

An agent performs a continuous life cycle that can be divided into three parts [118] (Figure 4.1):

- **perception** (step ❶), in which an agent acquires information about a limited part of the environment through sensors;
- **decision** (step ❷), in which an agent decides the action to be taken according to its perception of the environment. Agents have a variable number of actions that can be carried out;
- **action** (step ❸), in which the agent acts and modifies the local part of the environment in which it is situated.

A multi-agent system is then composed of a set of agents interacting with each other, each agent having its own objective and its own knowledge and skills. The behavior of the multi-agent system results from the local behaviours of each of its agents and their interactions.

### 4.1.2 Environment

Weyns et al. [147] defined the environment as "*a first-class abstraction that provides the surrounding conditions for agents to exist and that mediates both the interaction among agents and the access to resources*". By stating that the environment is a first-class abstraction, authors stress the fact that the environment is an independent entity in MAS systems. The environment is the part of the world with which the agents interact, in which the effects of the agents will be observed and evaluated. The environment provides a medium for sharing information and mediating coordination among agents. As a mediator, the environment not only enables interaction, it also constrains it. [147].

Agents must face the problem of deciding which of their actions should be performed to best satisfy their goals. The complexity of the decision-making process can be affected by several different environmental properties [145]. Russell and Norvig [118] discuss five fundamental characteristics that designers should consider when modeling an environment for an agent:

- *accessible/inaccessible*: in an accessible environment, agents can access all information regarding the state of the environment at any time. Complex environments (including, for example, the physical world and the Internet) are inaccessible. The more accessible an environment is, the simpler it is to build the agent to operate in it;

- *discrete/continuous*: an environment is discrete if a limited set of actions can be performed in it. A typical example of a discrete environment is the chess table: at a given time instant, only a limited set of actions can be performed in a cell. On the contrary, the physical world is continuous;

- *deterministic/non-deterministic*: in a deterministic environment, the result of an operation has a precise result that can be predicted in advance. On the contrary, in a non-deterministic environment, the same operation can produce diverse results in different contexts. The physical world can be regarded as non-deterministic. Modeling a physical world, considering only a subset of its properties, is a challenging task for agent designers;

- *static/dynamic*: A static environment does not change over time unless it is subject to the external actions of an agent. A dynamic environment evolves independently of the actions of the agents present in the system. The physical world is a highly dynamic environment.

- *episodic/non-episodic*: the result of the execution of a set of actions is independent of the execution of the same actions at an earlier time instant. In other words, an agent chooses the set of tasks to execute at a specific time not considering the effects they will have on the execution of future tasks.

### 4.1.3 Self-Organization

There are two primary ways to approach problem-solving: top-down and bottom-up. In the first case, the problem is well formalized, we can define a function to minimize or maximize to solve a problem, an objective function, or a goal to achieve. In this case, it is possible to decompose the problem into many sub-problems, each of which is delegated to different sub-components of a system. The solutions produced by the sub-components are combined so that the final result expected by the system is reordered. Not all problems can be formalized precisely, so the top-down approach is not pertinent. For instance, it is possible to define the tasks of the sub-systems but not a global function that defines the global behavior of the system. Therefore, it is preferable to adopt a reverse strategy, i.e. the bottom-up strategy, by attributing to the various sub-systems a local behavior that ensures that the overall functionality emerged is satisfactory to address the problem. The bottom-up strategy is particularly suitable for solving complex problems, whose functional objectives cannot be expressed formally. Some of these problems are present in nature, for example in flocks of birds or schools of fish. In these cases, individuals have limited capabilities but their interaction enables the emergence of complex patterns. Therefore, what are the rules that govern the dynamics between individuals, and how their interaction leads to an emergent behavior? A broad answer to these questions resides in a phenomenon known as *self-organization*.

Self-organization depicts an artificial or natural phenomenon that allows the spontaneous creation of patterns in space and/or time through the interaction between individuals in a group. Self-organization is closely related to the concept of "emergence", that is the creation of patterns that individuals can only determine through interaction with other individuals [37, 38]. Self-organization has brought attention to researchers during the mid-twentieth century in different fields of studies, notably biology and computer science. During the l970s and l980s, the increasing computing power made it possible to initiate a new research field that attempted to create artificial life through the use of computer systems to demonstrate self-organization in ecological and evolutionary contexts. Only during the 1990s, it was stressed the role of self-organization in computer systems as well as in biology and ecology [58].

Self-organization enables complex patterns to appear without any explicit control or constraints imposed from outside the system. In other words, the organization of individuals is based on local interactions between its components, often indirect, and are carried out through the environment [38].

In computer science, self-organization has been used for solving complex computational problems. In particular, self-organization finds its practical application in smart cities [98], complex systems in continuous evolution, where it is not possible to specify *a priori* the functional objectives of the system and to take into account parameters regarding, for example, information coming from citizens. A top-down approach would not result suitable unless the application domain is strongly constrained and patterns are known in advance. In a smart city context, it is preferable to adopt a bottom-up strategy, through the self-organization of intelligent devices so that the objectives of the

smart city can emerge autonomously. Two questions derive from this statement: how to implement self-organization? How is it possible to specify the interaction protocols between individuals in a system?

## 4.2 Cooperation

A way to design a multi-agent system to solve complex problem through a bottom-up approach is to use cooperative agents. For that, it is pertinent to attribute to agents local behaviors that ensure that the global functionality of the system (intended as the orchestration of the local activities of the agents) satisfies the definition of the problem. Ferber [46] provides the following definition of cooperative agents: *several agents will be said to be cooperating, or to be in a situation of cooperation, if one of the two conditions is verified:*

1. *adding a new agent enables increasing the performance of the group;*

2. *agents act to avoid or resolve potential conflicts.*

The performance of a group reflects the ability of agents to perform the task that must be accomplished by the group.

The use of the cooperative approach requires the analysis of the interactions that may occur between agents. In general, the interactions between agents can be analyzed at the macro and micro level [46]: at the macro level, we find the interactions that interweave the activity of a collective of agents, while at the micro level we analyze the individual interactions between agents. The relationship between micro-situations and macro-situations is expressed as the relationship of the parts to the whole: the macro-situation, being the result of the interactions at level micro, introduces a set of problems whose resolution depends on the production of several micro-situations that are themselves problem-bearing [46].

For reasons regarding technological limits, nowadays, multi-agent systems are developed in such a way to address a given problem, thus agents cannot overcome the limits of the problem itself. This implies that the interactions are constrained by the defined problem. When problems are complex, it is not possible to specify a formal objective function: this implies that it is not possible to analyze the interactions at the macro level to define a global way to solve the problem through the joint actions of a group of agents. For this reason, the cooperation enables addressing this challenge through a resolution process that considers the individual relationships of the agents (at the micro level) that lead to the emergence of an organization able to solve the defined problem; therefore, also a global behavior emerges from this process [53]. To realize cooperation, agents act altruistically to contribute, through their knowledge and skills, to the resolution of a complex goal that cannot be pursued individually by agents.

## 4.3 Adaptive Multi-Agent Systems

The development of a computational system requires the designer to properly implement functions that realize its nominal behavior and mechanisms to deal with any exceptions that may occur during the system's operation. By specifying *a priori* a model for a system that will have to deal with unexpected events, the space of possibilities can be constrained [22]. Unexpected behaviors of the system not considered in the design phase can compromise the system during its operation and potentially the data it manipulates. This may arise not only from human design errors but also from the complex nature of the software. For example, consider the design of a software system with the following specifications [22]:

- The environment of the system is dynamic, making it ineffective to enumerate exhaustively all the situations the system may encounter.
- The system is open and therefore dynamic because it is constituted of a shifting number of components.
- The task the system has to achieve is so complex that we cannot guarantee a perfect design.
- The way by which the system may achieve the task it has been assigned is difficult or even impossible to apprehend globally by the designer.

Anticipating all possible exceptions is extremely complex, especially because of the properties of openness and dynamic environment. Therefore it is necessary to have a methodological approach that can guide towards the realization of systems able not only to operate autonomously but also to adapt to changes in the environment and to evolve with it, ensuring that the functioning of the system converges towards the desired result. This challenge has been encompassed by the *Adaptive Multi-Agent Systems* (AMAS) theory as well as the ADELFE methodology, which enable the design of systems to perform complex tasks through mechanisms of cooperation between software agents.

The AMAS approach allows the design of multi-agent systems through a bottom-up resolution process where the agents collectively solve a problem. All the agents participate in solving complex problems without deliberately lying or being malicious [53]. Although this statement may result obvious, as the system designer does not intentionally create behaviors that could compromise the proper functioning of the agents. In fact, as explained below, it is the interactions between the agents that can lead to anomalous behaviors.

If every agent has the capability to locally rearrange its interactions with others, this ability of self-organization at the lowest level permits changes in the global function without coding this modification at the upper level of the system. In this way, self-organization allows the system to solve complex problems, whose global function is impossible to describe in a formal way. Consequently, this function has then to emerge at the macro level (the system level) from the

interactions at the micro level (component level). Moreover, this global function cannot be known at agent level, and agents just needs some local criteria to rearrange their interactions [13].

Using the AMAS approach to solve complex problems means that, as mentioned earlier, the designer cannot specify a global objective function for the MAS, thus it is not possible to specify a function shared among agents that defines the goal to pursue. A theoretical foundation to answer this issue is the functional adequacy: a system is said to be functionally adequate if it executes the function which it has been designed to [22].

The evaluation of the functional adequacy of a system is delegated to an external entity that observes the behavior of the components. In MAS, this evaluation has to be addressed by agents on their own activities, as they do not have any knowledge about the global goal to be achieved. Therefore, the observation of the components of the system cannot be delegated to an external observer, but rather it is the agents that individually, based on their observations and feedback from the environment evaluate their behavior according to a local criterion.

The AMAS approach enables to guarantee the convergence towards a desired global function. This is obtained by the following theorem [22]: *For any functionally adequate system in a given environment there is a system having a co-operative internal medium which realizes an equivalent function*. A cooperative internal medium system is a system composed of parts which are always in an interactive way with its own environment [56]. A system with cooperative internal medium has only cooperative exchanges with its environment because these exchanges are a subset of its parts interactions. Systems with cooperative internal medium have these properties [22]:

- A cooperative system in the environment is functionally adequate, which implies that the system has only beneficial activities for its environment.
- There is no need to formally specify a global goal for the system. Agents act so as to keep their behavior cooperative according to their skills, representations of themselves, other agents and environment.
- The feedback concept is not constraining in this theory because the system must only evaluate if the changes taking place in the medium are cooperative from its point of view without knowing if these changes are dependent on its own past actions.

An AMAS system is characterized by the following points [54]:

- the system is plunged into a dynamical environment;
- the system realizes a function;
- the system is composed of interacting autonomous agents;
- each agent of the system realizes a partial function. The global function is the result of the composition of the partial functions of the agents and of their interactions;

- the organization of the system determines the result of the system. The composition of partial functions is determined by relations that link agents i.e. the organization.

### 4.3.1 Non-Cooperative Situations

Cooperation is the key mechanism through which AMAS can be implemented to solve complex problems. The functional adequacy ensures that for a calculable problem there exists a MAS whose agents cooperatively find a solution. The non-adequacy of the system comes from the existence of non-cooperative interactions within the system. In open systems, it is not possible to anticipate cooperative behaviors that do not contribute to the global functioning of the system, either actions that could be deleterious. To reach a functionally adequate state, agents need to locally detect the failures in cooperation by changing their behaviors.

An AMAS has to be conceived so that agents can do the best they can when they encounter difficulties. These difficulties can be viewed as exceptions in traditional programming. From an agent point of view, we call them Non-Cooperative Situations (NCS) or cooperation failures. A NCS is defined as follows [53]:

**Definition 2.** *An agent is in a Non-Cooperative Situation (NCS) when:*

- *a perception (either from the environment or other agents) is not understood or is ambiguous;*
- *perceived information does not produce any new decision;*
- *the consequences of its actions are not useful to others.*

Given the previous definition, seven generic types of NCS that an agent may encounter have been identified. These NCS can be triggered by the interaction of an agent either with another agent or with the environment:

- **Perceive**:
  - *incomprehension*: the agent cannot extract the semantic contents of a received stimulus;
  - *ambiguity*: the agent extracts several interpretations from a same stimulus;
  - *incompetency*: the agent cannot benefits from the current knowledge state during the decision;
- **Decide**:
  - *unproductiveness*: the agent cannot propose an action to do during the decision;
- **Act**:
  - *concurrency*: the agent perceives another agent which is acting to reach the same world state;

  - *conflict*: the agent believes that the transformation it is going to operate on the world is incompatible with the activity of another agent;
  - *uselessness*: the agent believes that its action cannot change the world state or it believes that the results for its action are not interesting for the other agents.

NCSs must be identified during system design and solved through specific functions. A cooperative agent has two types of behavior: the nominal behavior, which describes the local functionalities of the agents, and cooperative behavior, that consists in a set of actions that enables the agent to prevent and solve eventual NCS. The cooperative behaviour of the agents is the basis of self-adaptation in AMAS: the system tends to reach the functional adequacy as the NCS are solved or avoided.

To solve NCSs, agents carry out so-called "cooperative" actions that consist in locally adjust their behavior. Three types of cooperative actions have been identified [21]:

- *Tuning*: the agent adjusts its internal parameters.
- *Reorganization*: the agent changes the way it interacts with its neighbourhood,i.e it stops interacting with a given neighbour, or it starts interacting with a new neighbour, or it updates the confidence given to its existing neighbours.
- *Openness*: the agent creates one or several other agents, or deletes itself.

Identifying the nominal and cooperative behavior of agents is delegated to the system designers, who must propose appropriate mechanisms. To this end, the ADELFE methodology introduced in the following section guides designers in the conception and the development of cooperative AMAS.

### 4.3.2 Criticality

When an agent determines a NCS, a criticality value associated with the agent is calculated. For an agent, criticality represents the distance between its current situation and the its own goal. It enables an agent to determine the relative difficulty of agents in its neighborhood and improves cooperation. The evaluation methods and calculation of the criticality are specific to each type of agent [17]. An AMAS agent can determine NCSs and its criticality in several ways [85]:

- according to a priori determined criteria, depending on the agent's perceptions, that allow the agent to interpret the criticality of NCSs.
- according to the feedback transmitted between agents. An agent can inform other agents about the encountered NCS and the criticality associated with the NCS. An issuing agent can potentially attach the information useful for the resolution of this NCS.

- according to service requests transmitted between agents. An agent may need the skills of another agent in order to maintain a cooperative activity. The agent then attaches to its request for service the criticality of the NCS which it would encounter if it did not obtain satisfaction.
- In order to pursue its local goal, an agent may require the skills of another agent. To this end, the agent associates to its request for cooperation the criticality that would result if it does not get help.

Cooperative agents attempt to reduce their criticality of agents that encountered some NCS while avoiding that the process of helping the agent does not cause the criticality of other agents to raise. The criticality is notably employed to determine which NCS to resolve first.

## 4.4 Designing an AMAS with ADELFE

Today several agent-oriented methodologies exist. Some of the most recognized methodologies are ASPECS [30], PASSI [29], TROPOS [19], GAIA [153], Prometheus [148] and so on. Differently from these methodologies, the AMAS approach focuses on the design of cooperative multi-agent systems. ADELFE is the french acronym for "*Atelier de Développement de Logiciel à Fonctionnalité Emergente*" which can be translated by *Toolkit for Designing Software with Emergent Functionalities*.

ADELFE is based on the *Rational Unified Process* (RUP) to guide the designer during the 5 phases of AMAS design [17]:

- **WD1** - Preliminary requirements: this phase represents a description of specifications between customers, users and designers. The result of this phase is a document containing a precise description of the problem without using any particular modeling language.
- **WD2** - Final requirements: in this work definition, the preliminary requirements are transformed to use cases. Also, this work definition characterize the environment of the system, by identifying the entities that interact with the system and the constraints on these interactions.
- **WD3** - Analysis: this involves the identification and the definition of the different entities of the system by evaluating their pertinence to the AMAS theory. The result of this phase is a comprehensive description of agents and their interaction.
- **WD4** - Design: in this phase are defined the interaction protocols between agents as well as their behaviors (nominal and cooperative). This activity results in a complete characterization of the multi-agent system.
- **WD5** - Implementation: implementation of the framework and agent behaviours.

ADELFE constitutes a fundamental tool for the design of AMAS. The Appendix B provides further details on the ADELFE methodology. A complete description of the ADELFE methodology can be found in [17].

## 4.5 Discussion

The concept of the smart city arises from the need to find a solution to rapid population growth and the risks this entails for a city, economic risks such as unemployment, or physical risks such as over-pollution [74]. MAS together with IoT constitutes an effective technological means for addressing the objective of smartness by enabling the design of networks of intercommunicating devices that responds to the needs of citizens both individually and as a whole and also by monitoring with sensors the levels of pollution, traffic, noise, etc [62, 74]. A smart city can be instrumented with a large amount of sensing devices constantly collecting information about actions that happen in the city, humidity sensors, temperature sensors, noise, pollution, etc. All these sensors are part of a data collection system that will be responsible for processing information quickly and intelligently [74]. Through the MAS technology, sensing devices embody not only data acquisition capacities but also mechanisms of coordination and are capable of act jointly. The increasing complexity of the urban society and its constantly evolving dynamics makes it necessary for smart cities to set goals that are difficult to formalize for computer software, even for a MAS. This is because of the nature of the physical environment in which the MAS is immersed. For example, the learning activity in ambient systems is a complex activity: the task to learn is not known in advance, can vary over time, new tasks can be introduced, etc. Moreover, their non-linearity (that is, the inability to define through simple mathematical rules the behavior of the system), the high number of components and their interactions, their unpredictability, make an extremely complex task the specification of a single computation model [137].

The AMAS approach aims at solving problems in dynamic nonlinear environments by a bottom-up design of cooperative agents, where cooperation is the engine of the self-organization process [53]. The AMAS approach is a pertinent solution to address the smartness goals of cities by exploiting agents that are delegated with simple tasks; the cooperation allows the agents to solve complex problems that cannot be pursued individually by the agents.

## 4.6 Conclusion

This chapter introduced the AMAS as an approach for solving complex problems in dynamic, open environments through autonomous agents. AMAS are suitable for solving problems that cannot be precisely formalized through a bottom-up approach, where agents organize autonomously their operation to solve cooperatively a problem that cannot be pursued individually. Autonomous agents have social skills, thanks to which they can operate jointly by cooperating, acting together and compensating their local skills so that global patterns can emerge.

Autonomous agents have two behaviors: nominal behavior and cooperative behavior. The nominal behavior consists of a set of skills that are performed locally by an agent in its environment to pursue its goal. The cooperative behavior allows agents to perform jointly different operations that can contribute to the emergence of patterns that cannot be produced individually. The joint

execution of different actions leads to the emergence of patterns that cannot be produced by individual agents.

It is a difficult task to anticipate, at design time, how agents have to interact in case of expected behaviors. In this case, the interactions between agents enable the emergence of cooperative behavior that could correct eventual exceptions. Indeed, there is no global control of the system. In itself, the emergent organization is an observable organization that has not been given first by the designer of the system [53]. Therefore, the agents assume a cooperative behavior to resolve any unpredictable situations (NCS).

When conceiving an AMAS, the designer must define the nominal behaviors of the agents. Besides this, the designer must describe the cooperative behaviors of the agents, based on the possible malfunctioning that can occur during the nominal behaviors. When an agent encounters a NCS, it solves this exception by acting cooperatively.

Seven NCS have been identified and can be triggered by the interaction between agents. These NCS are sufficiently generic to cover a large variety of problems. System designers can use these generic NCS to its problem to define the behavior that agents must assume in case of unpredictable behavior.

# Part II

# Contribution

Chapter 5

# Estimating Missing Information in Smart Cities: HybridIoT

Objectives of this chapter:

- Introducing the HybridIoT system for estimating missing information in smart cities
- Describing how agents in HybridIoT implements the estimation technique
- Describing how agents adapt to the environmental context to determine how to estimate the missing information
- Describing how agents use historical data to improve the estimation task

THE previous chapter introduced the MAS paradigm and the AMAS approach to solve complex problems in a bottom-up way through the self-organization of cooperating agents that operate jointly to solve problems that cannot be solved individually. Also, a brief description has been reported on the ADELFE methodology, conceived to guide the designers in the conception and development of complex systems using the AMAS approach.

This chapter discusses the HybridIoT estimation technique for estimating missing environmental information on a large-scale. The description of the system will be characterized by the objective of the system, the environment and the behavior of the agents. This description, following the ADELFE approach described in the previous chapter, enables justifying the use of the AMAS approach.

## 5.1 Problem Statement

This section presents the formal problem statement concerning the application of this thesis. The formulation presented is taken from Lehmann and Casella [84].

The statistical problem addressed in this thesis concerns the theory of estimation, one of the most common forms of statistical inference. Given:

- $x[t]$ is the observation at time $t$,
- $x = (x[1], x[2], ..., x[T])$ is a vector of $T$ observation samples,
- $\theta$ is the parameter vector of interest belonging to a sample sub-space $\Omega$, and
- $p(x; \theta)$ is the mathematical model (i.e. a probability density function) parametrized by $\theta$.

Suppose that $g$ is a real-valued function defined over the parameter vector $\theta \subset \Omega$ and that we would like to know the value of $g(\theta)$. Unfortunately, $\theta$, and hence $g(\theta)$, is unknown. However, the data can be used to obtain an estimate of $g(\theta)$, a value that one hopes will be close to $g(\theta)$. The problem is to find a function of the $N$-point data set which provides an estimate of $\theta$, that is:

$$\hat{\theta} = g(x = \{x[1], x[2], ..., x[T]\}) \tag{5.1}$$

We are interested in finding an estimator function that calculates estimates punctually, that is, using a collection of sample data to calculate a single value at a given time instant. In this case, we say that the problem concerns the **point estimation theory**.

The problem is, therefore, the determination of a suitable estimator, which can be defined as follow:

**Definition 3.** *An estimator is a real-valued function $\delta$ defined over the sample space. It is used to estimate an estimand $g(\theta)$, a real-valued function of the parameter $\theta$.*

It is hoped that $\delta(X)$ will tend to be close to the unknown $g(\theta)$, but such a requirement is not part of the formal definition of an estimator. The value $\delta(x)$ taken on by $\delta(X)$ for the observed value $x$ of $X$ is the estimate of $g(\theta)$, which corresponds to the "correct" value for the unknown value. Because $\delta(X)$ is a random variable, we shall interpret this to mean that it will be close on the average. To make this requirement precise, it is necessary to specify a measure of the average closeness of (or distance from) an estimator to $g(\theta)$. Examples of such measures are

$$P(|\delta(X) - g(\theta)| < c) \quad \text{for some} \quad c > 0 \tag{5.2}$$

and

$$E|\delta(X) - g(\theta)|^p \quad \text{for some} \quad p > 0. \tag{5.3}$$

Consequently, estimating an information causes $\delta(X)$ to generate a loss of information that can be measured as follows:

$$L(\theta, d) \geq 0 \quad \text{for all} \quad \theta, d \tag{5.4}$$

and

$$L[\theta, g(\theta)] = 0 \quad \text{for all} \quad \theta, \tag{5.5}$$

so that the loss is zero when the correct value is estimated. The accuracy, or rather inaccuracy, of an estimator $\delta$ is then measured by the risk function

$$R(\theta, \delta) = E_\theta\{L[\theta, \delta(X)]\} \tag{5.6}$$

where $X$ is a vector of observations stochastically drawn from a population, $E_\theta$ is the expectation over all population values of $X$.

This problem has no *exact* solution. It is possible to reduce the risk at any given point $\theta_0$ to zero by making $\delta(x)$ equal to $g(\theta_0)$ for all $x$. There exists no best estimator that minimizes the risk for all values of $\theta$. To avoid this difficulty, the class of estimator can be restricted by excluding the estimators that too strongly favor one or more values of $\theta$ at the cost of neglecting other possible values. In this ways, only the estimators that behave in a "generic" manner are kept. This can be achieved by requiring the estimator to satisfy some condition which enforces a certain degree of impartiality. One such condition requires that the bias $E_\theta[\delta(X)] - g(\theta)$, also called the *systematic error*, of the estimator $\delta$ be zero, that is, that

$$E_\theta[\delta(X)] = g(\theta) \quad \text{for all} \quad \theta \in \Omega \tag{5.7}$$

This condition of unbiasedness ensures that, in the long run, the estimator provides correct answers "on the average".

### 5.1.1 Discussion

In the context of this thesis, we suppose that one or more observations at time instant $t$ may be unavailable. Based on the samples previously observed by sensing devices, we propose an estimation technique that is deployable at a large-scale and addresses the properties of openness and heterogeneity. Sensing devices are defined as follows:

**Definition 4 (Sensing Device).** *A sensing device is any physical instrumentation that embeds sensors capable of detecting events and changes in its environment. Sensing devices can be either fixed or mobile: fixed sensing devices are installed in specific positions and enable continuous monitoring of the environment. They acquire information continuously and store these in dedicated databases. Mobile sensing devices can embed a variable number of sensors: the information can be acquired on-demand by the user (the owner of the device) or using a fixed schedule.*

The developed approach is based on the AMAS approach presented in the previous chapter; this approach allows the sensing devices to perform autonomous computation in their environment (locally), without being influenced in any way by other sensing devices. Moreover, the software agents

composing the designed AMAS improve their estimation function according to the information they perceived; this allows calculating accurate estimates at any time.

The scientific contribution of the proposed approach lies in an estimation technique for environmental information according to three different schemas:

- using the historical data [1];
- using the information acquired by nearby, homogeneous sensing devices [2];
- using heterogeneous information (that is, information of different types and/or units) [3].

## 5.2 General Functioning of HybridIoT by Example

This section describes a representative case study that will be used to present the general functioning of the HybridIoT estimation technique and how it works under different conditions. In the rest of the chapter, we assume that there is no difference between **sensors** and **sensing devices**. Moreover, we assume that sensing devices are agentified, thus are associated with agents that can estimate data in case of temporary unavailability of the sensing device and to learn the dynamics of the environment in which the device is situated.

In the scenario, illustrated in Figure 5.1, there are two rooms including three sensing devices: two temperature sensors, one for each room ($S_1$ and $S_3$) and a humidity sensor in the corridor ($S_2$). All three sensors installed are supposed to be fixed at a given position and their functioning is autonomous. The rooms are separated by walls that are not known by the agents, nor manually specified by any configuration. In case a sensor is unable to provide any information by direct observation of the environment, the associated agent cooperates with the other agents using their perceived information.

Each agent is able to learn environmental dynamics from information obtained by direct observation. In some cases, agents use the learned information to estimate missing information.

In the following examples, we suppose that sensing devices are capable of perceiving their environment and that each one is associated each one with an AMAS agent capable of processing the environmental information and estimating missing information when needed.

### 5.2.1 First case: no other sensor available

Let $S_1$ be an agentified sensing device. Its associated agent **learns the dynamics of the local part of the environment**. Suppose that $S_1$ encounters a problem and that it is no longer able to provide temperature values (Figure 5.2). Because there is no other agent in the proximity of $S_1$ (the agents are aware of the local environment in which they operate), then the agent associated with $S_1$ must find a way to provide the information to the user despite its unavailability. The agent associated with $S_1$ uses the information previously observed (during $S_1$ functioning) to provide an estimate for the

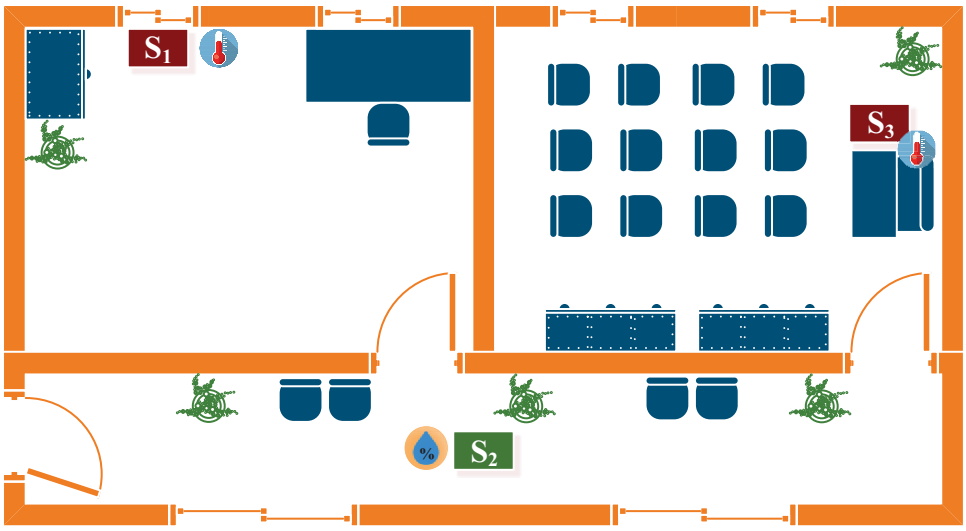



Figure 5.1: Case study illustration. Three sensors are deployed into a building: two temperature sensors ($S_1$, $S_3$) and a humidity sensor ($S_2$).

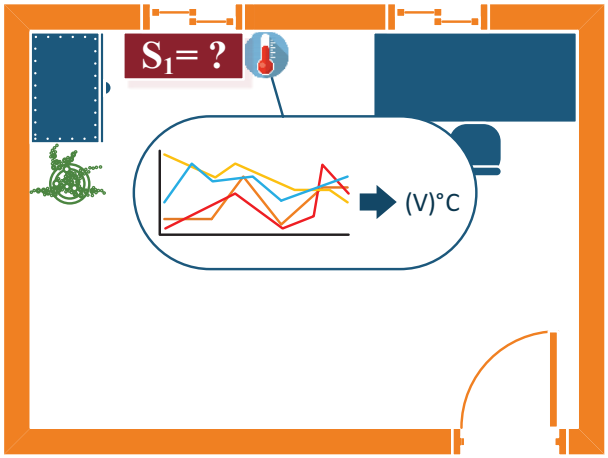



Figure 5.2: In case of an unpredictable event that prevents $S_1$ to provide temperature values and no sensor is available in the proximity, the agent associated with $S_1$ calculates estimates by using the values previously perceived.

missing values. The agentification of sensing devices provides resilience and learning mechanisms to solve problems such as the temporary unavailability of the sensors. In this example, the local computation of the agent provides accurate estimates of missing values when $S_1$ is not able to provide any information.

### 5.2.2 Second case: some sensors are present in the proximity of $S_1$

This example shows how HybridIoT addresses the **openness property**. Suppose the agent associated with the sensing device $S_1$ must estimate the missing information as in the previous example

and it estimates the missing information using the information previously observed (Figure 5.3). During this operation, some sensing devices perceiving the same type of information as $S_1$ are supposed to be in the proximity of $S_1$. In this case, $S_1$ adapts its behavior (therefore how it estimates the missing information) to the environmental context and calculates an estimate for the missing information by cooperating with the agents associated with the sensing devices of the same type (that is, perceiving the same type of information).

The agents address the openness property as they can process new inputs from sensing devices that enter the system at any time. Therefore, the functioning of each agent is independent of each other. When cooperating with other agents, $S_1$ gives more importance to the real information perceived by the nearby sensing devices (as they are likely to perceive similar information) than the information observed previously (because the current environmental dynamics may be significantly different from the one previously observed).

In this example, **multiple agentified sensing devices are considered**. Each agent operates locally in its part of the environment and operates independently from the other agents. The local and autonomous computation of the agents enables the technique to be scalable.

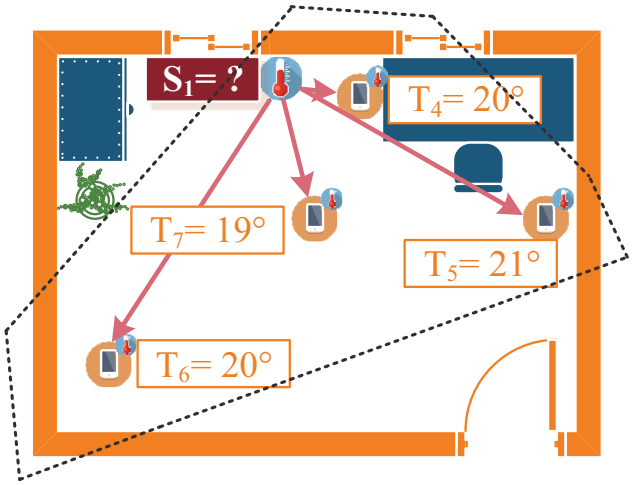



Figure 5.3: Agent $S_1$ uses the information obtained cooperatively from the agents available in its local part of the environment to estimate the missing values.

### 5.2.3 Third case: some sensors perceiving information of different types are present

This example shows how HybridIoT is able to estimate missing information by **integrating heterogeneous information**. Suppose the agent associated with the sensing device $S_2$, perceiving humidity values, must estimate missing information (Figure 5.4). If there is no agent in its environment, the agent associated with $S_2$ can rely on its knowledge[1] to calculate the estimates for the missing value.

[1]Here we refer to knowledge as the set of information perceived by sensing devices, then collected and processed by the associated agents.

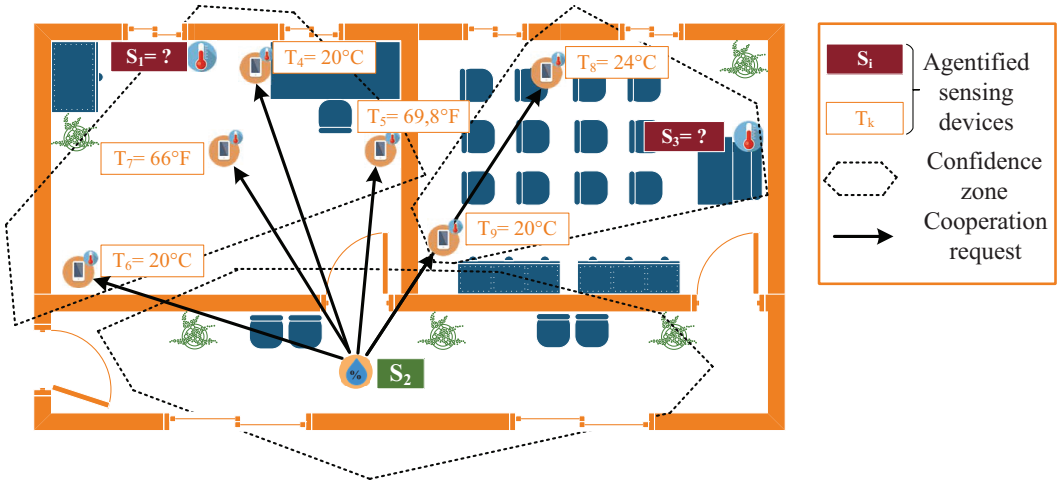



Figure 5.4: Agent $S_2$ estimates missing humidity values by cooperating with agents perceiving heterogeneous information.

If some sensing devices are available in the environment but not in the local part observed by $S_2$, this one can cooperate with other agents, that perceive different types of information, to estimate the missing information. In this way, the agents overcome the lack of homogeneous sensors. Therefore, $S_2$ estimates the missing information through a cooperative resolution process that is based on the exploitation of the environmental information observed by agents perceiving heterogeneous information.

As in the previous example, the agents associated with the sensing devices operate autonomously in their local part of the environment. Even if the estimation process is pursued between agents associated with sensing devices perceiving **different types of information**, the same technique can be deployed at **large-scale** as do not make any assumption on the scale or the type of environment in which sensors are installed.

### 5.2.4 Discussion

The illustrated example shows how the estimation technique implemented by the HybridIoT system enables addressing the properties of **openness**, **heterogeneity** and **large-scale**. The instantaneous adaptation of agents in case new devices entering the system does not affect the functionality of the available agents. The agents can use new inputs to calculate any missing value estimates without any reconfiguration. The agents use heterogeneous information whose type is not known in advance to ensure that the information is provided even in case of a lack of homogeneous sensors. The multi-agent approach used to implement the proposed estimation technique allows the agents to pursue autonomous computation so that the estimation technique can be used in **large-scale contexts**.

## 5.3 System Objective

The novelty of the proposed approach lies in:

- estimate missing information in points in the environment where few sensing devices are present. In this case, the advantage proposed technique is twofold:
  - to provide a resilience mechanism to allow the information to be available in case of sudden malfunctioning of sensing devices and
  - to reduce the number of sensing devices to deploy in an environment;
- learn from raw data without feedback from the end-user;
- exploiting environmental information from available devices to avoid the installation of a large number of *ad hoc* sensors;
- integrating heterogeneous environmental data.

HybridIoT leverages on two types of estimation:

- **endogenous estimation**, relying on information of the same type, same scale and unit as the sensor that must estimate an information. The endogenous estimation consists of two further types of estimation:
  - **using historical data**: the estimation is done by exploiting the information previously acquired by sensors;
  - **using nearby sensors data**: the estimation is done by exploiting the information of the same type (and same unit) acquired by nearby sensors';
- **exogenous estimation**, estimating missing information by integrating heterogeneous information perceived from different data sources.

To motivate the use of exogenous estimation, consider an urban sensing platform used to monitor the environment. In such a context, sensors can be of different types, they can acquire heterogeneous information or data of the same type using different units. We hypothesize that the abundance of different information is crucial for both AI as well as data analysis techniques to extract useful knowledge from the urban environment to achieve the smartness goal of a city. The exogenous estimation is beneficial in the smart city context for three aspects:

- enables integrating heterogeneous information to estimate missing values and to overcome the lack of sensors of the same type;
- the integration of heterogeneous information enables defining a coherent representation of the environment to monitor and to understand its evolution;

- the integration of heterogeneous information enables extracting correlations between semantically different information.

Exogenous estimation enables leveraging the large amount of information perceived by heterogeneous devices to provide wide information coverage. This ensures that users can access accurate and timely information on the state of the environment, so that experts can use the information to improve services offered to citizens.

In the context of the development of a framework to support smart city initiatives, HybridIoT aims at improving the economic and sustainable aspects. The installation, maintenance and operation of *ad hoc* sensing devices are the main costs for the deployment of an urban sensing infrastructure. That's why the proposed estimation technique enables reducing the number of sensors in a large-scale context by exploiting the information acquired from some sensing devices and providing accurate estimates whereas sensing devices are momentarily not available (due to unpredictable malfunctioning or not present).

## 5.4 System Requirements

The requirement steps of the ADELFE methodology are devoted to the establishment of requirements and are usual in software development methodology. The steps consist in a description of the problem domain to be solved, as well as a specification of the final user needs, the functional and non-functional requirements [115]. Therefore the environment and the entities situated in it are identified. This phase results in an identification of the functionalities required by the user and a complete definition of the elements that make up the system to be done, so that it can be analyzed later to verify the adequacy of the AMAS approach to the problem posed.

Functional requirements refer to the constraints related to the function of the system [20]:

- HybridIoT must adapt to unexpected and unpredictable changes in the environment related to the changes in environmental conditions but also to the changes in the configuration of the devices (such as their location or their availability);
- HybridIoT must be able to learn the evolutionary dynamics of the environment by observing the information acquired directly from the environment;
- HybridIoT must provide estimates in a short amount of time (from seconds to minutes).

Non-functional requirements are constraints on the operation of the system that are not related directly to its functioning [20]. We identified the non-functional requirements as the properties addressed by HybridIoT that arise from the complexity of the environment in which the system operates. These properties have not been simultaneously explored by the state of the art solutions:

- *Openness*: refers to the capacity to handle new input at any time without the need for any reconfiguration while ensuring its operation;

- *Large-scale*: refers to the capacity to be deployed in large, urban contexts where thousands of devices can be present;
- *Heterogeneity*: refers to the capacity of handling different types of information without any *a priori* configuration.

HybridIoT satisfies these properties, making it a significant advance in the state of the art for estimating missing information. Heterogeneity enables integrating different types of devices that could perceive several types of environmental information. HybridIoT is capable to reason on these sensing devices to accurately estimate missing information. Openness enables sensing devices to enter or leave the system without the need for any re-configuration.

The system requirement phase also establishes the limits of the systems to be designed. This includes identifying the entities involved in the system operation, defining the environment, the entities situated within it and its properties.

The actors involved in the functioning are solely the users, without any difference in role or responsibility. A user's objective consists of interacting with the system to request environmental information; the requests can occur at any time and everywhere. The interactions between the user and the system only include the requests for particular types of information, and the related responses by the system, which in turn consist of environmental data.

Due to the constantly evolving and unpredictable dynamics of the urban contexts, sensors are not always available to provide the information to the user. This is why the sensors must be endowed with adaptation capabilities so that they are capable of facing unpredictable events that prevent them from providing information.

We identify the following failure situations:

- the user has requested information from a device that cannot provide the desired information;
- there is no sensor at the point where information was requested;
- sensors in the proximity provide information different from that required.

Besides, according to Russel and Norvig's definition, the environment in which sensors are plunged is:

- **inaccessible** because knowing all about the environmental information is difficult;
- **discrete** because the number of distinct perceptions and actions is limited;

## 5.5 System Analysis

The analysis phase of the ADELFE methodology aims at identifying the system structure and justifying the AMAS adequacy. This phase enables analyzing the domain characteristics, determine

the agents and validate an AMAS approach at the global and local levels [17]. The AMAS adequacy process consists of dealing with the AMAS principles; this concerns the identification of cooperative agents, how they determine the cooperation failures that can occur between entities, and then to define the agents regarding the results of previous steps. The verification of the adequacy of the AMAS is crucial for the ADELFE methodology. Indeed not all the applications require the AMAS approach for their realization. Figure 5.5 shows the flow of tasks necessary to verify the MAS adequacy according to the ADELFE methodology.

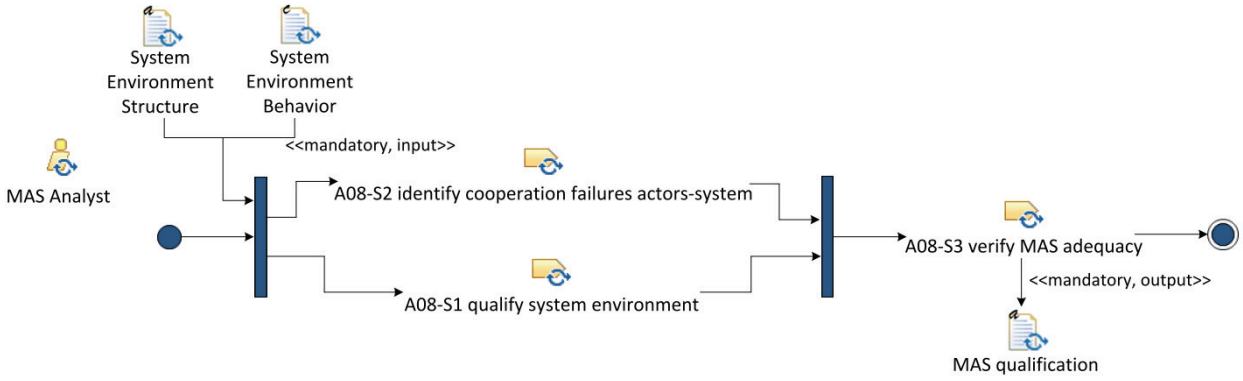



Figure 5.5: Flow of tasks necessary to verify the MAS adequacy according to the ADELFE methodology.

The entities of an AMAS are differentiated into two categories [17]:

- **Active entity**: an active entity is given behavioral autonomy, allowing it to change state without necessarily interacting with another entity. An active entity can interact with other entities, possibly through communication mediums;
- **Passive entity**: passive entities are associated with resources or data. Passive entities have no behavior, they can only be perceived and eventually modified by active entities. A state transition can only be the result of an interaction with another system component.

In HybridIoT, we identify three types of entities: information, sensors and ambient devices. The information, which can be real or estimated, is the result that must be presented to the end-user. It does not contain any indication of whether it has been estimated or not. The sensors are responsible for acquiring the information. The ambient devices are any device that contains sensors and can be queried by the end-users.

We consider sensors and ambient devices as active entities. The former because they perceive information autonomously, they need interaction and negotiation capabilities to ensure that estimates can be calculated when information is not available. Ambient devices, to which we can associate IoT devices, smartphones, connected cars, in general, all devices associated with environmental

sensors, are considered as active because they are linked to users, who may change their mind which implies a change in the state of the related entities.

The ADELFE methodology provides a tool based on a questionnaire to assess the adequacy of the AMAS approach to our problem (Figure 5.6). This adequacy is studied both at the system level and the local level (the level of the different parts composing the system) [12].

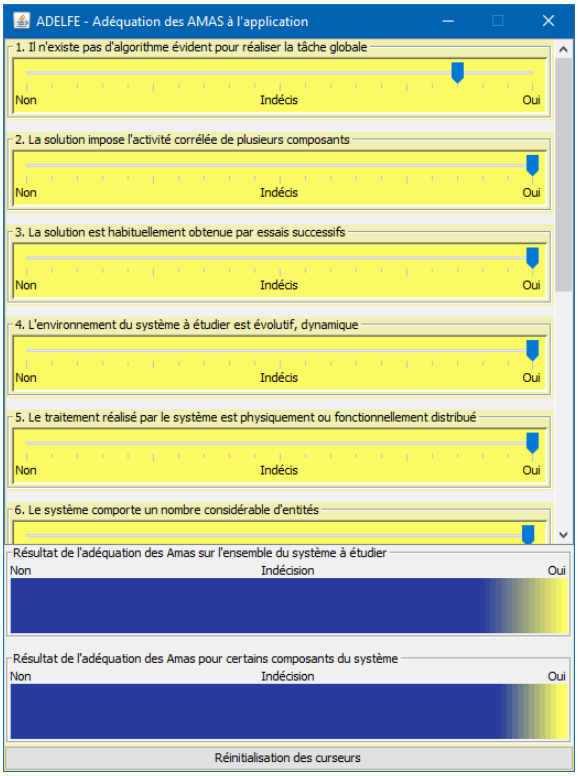



Figure 5.6: ADELFE tool used to assess the pertinence of the AMAS approach for the HybridIoT system.

The two main activities of the analysis phase consist in verifying the adequacy of AMAS at global and local levels [17]:

- **Global level**: In this activity, the adequacy at the global level is studied to answer the question *"is an AMAS required to implement the system?"*. This is done through several simple questions related to the global functioning of the group of agents in HybridIoT [13]:

  1. Is the global task incompletely specified? Is an algorithm a priori unknown?
  2. Is the correlated activity of several entities needed to solve the problem?
  3. Is the solution generally obtained by repetitive tests? Are different attempts required before finding a solution?
  4. Can the system environment evolve? Is it dynamic?

5. Is the system functionally or physically distributed? Are several physically distributed components needed to solve the global task? Or is a conceptual distribution needed?
6. Does a great number of components needed?
7. Is the studied system non-linear?
8. Finally, is the system evolutionary or open? Can new components appear or disappear dynamically?

- **Local level**: In this activity, the AMAS adequacy is studied at the local level to determine the agents that need to be implemented as an AMAS. At the component level, three more criteria are used [13]:

1. Does a component have only a limited rationality?
2. Is a component "big" or not? Is it able to do many actions, to reason a lot? Does it need significant abilities to perform its own task?
3. Can the behavior of a component evolve? Does it need to adapt to the changes of its environment?

The following subsection discusses both global and local levels analysis, then the identified agents involved in HybridIoT operation.

### 5.5.1 Global Level Analysis

**Is the global task incompletely specified? Is an algorithm *a priori* unknown?** The objective of the proposed method is to estimate environmental information in local parts of the environment not sufficiently covered by sensing devices using heterogeneous information which type is not known in advance; the information is acquired from a set of sensors not known in advance. HybridIoT can be effectively considered as an ambient complex system. The following points clarify why the addressed problem is complex:

- the physical environment in which the system operates respects the given definition of complexity: it is **heterogeneous**, its evolution is **non-linear** (that is, its evolution cannot be determined only by its current state and it is not proportional to the data input), and **non-deterministic** (the consequences of performed actions in the physical environment could not be determined in advance with certainty).
- the dynamism of IoT devices involved in the system operation process makes it complex to achieve the system global goal. The dynamism is linked to the nature of the connected objects, not always available, some of them are mobile and heterogeneous. These factors make the interoperability of IoT devices difficult to realize in an ambient system.

- The system has to make decisions in a dynamic, constantly evolving environment, so it is difficult to foresee all the possible behaviors that the system has to take in every possible situations. Ambient systems, therefore, have a certain form of rationality or intelligence, which enables them to learn from environmental observations, to observe the dynamics, and to respond accordingly to situations that cannot be predicted in advance.

The requirements of the proposed technique make it impossible to specify a global, formally defined objective that can be used for the development of the proposed approach in a top-down manner.

**Is the correlated activity of several entities needed to solve the problem?** The unavailability of a device in providing the information requested by the user requires a mechanism of interaction between a group of sensors to provide an estimate of the information requested, calculated through the information acquired by a collective of available sensors.

**Is the solution generally obtained by repetitive tests? Are different attempts required before finding a solution?** The estimate of missing information is calculated in almost instantaneous time based on the available information, no repetitive tests are required.

**Is the system functionally or physically distributed? Are several physically distributed components needed to solve the global task? Or is a conceptual distribution needed?**

The system is functionally and physically distributed, composed of different agents each one associated with a physical device that perceives environmental information. Thanks to the properties of the AMAS approach, the proposed estimation technique is deployable in a physically distributed environment where sensing devices are coupled with intelligent agents providing estimates in case of missing information.

**Does a great number of components needed?** The proposed technique enables operating despite the number of devices present that may vary during the operation. There is no need for a large number of components as the proposed estimation technique relies on the available sensors to calculate the estimates for missing information.

**Finally, is the system evolutionary or open? Can new components appear or disappear dynamically?** To address the openness property, it is necessary to calculate estimates through the information coming from devices that can enter or leave the system at any time. HybridIoT enables addressing the openness property.

### 5.5.2 Local Level Analysis

In the second iteration of analysis we examine the local level to know if some components will need to be recursively viewed as AMAS. If some components need to evolve or adapt themselves, it is rather sure that they will be implemented as AMAS. We have therefore repeated the analysis phase of ADELFE to determine the agents composing an AMAS sensing device: the agents need data to ensure its proper operation, therefore, we define the data agent associated with each observation

coming from the environment at a precise moment in a discrete-time interval. Data perceived in consecutive time instants are aggregated to determine windows that accurately describe the environmental evolution in a discrete time interval. We then identify the **Ambient Context Agent** (ACA) as responsible for providing environmental information at any time from either direct observation or estimation and learning the environment dynamics by grouping information into data windows known as **Ambient Context Windows** (ACWs), each one composed of multiple **Context Entries**. We report the definitions of both ACW, context entry and ACA respectively:

**Definition 5** (**Ambient Context Window**). *An Ambient Context Window (ACW) $C_t$ contains homogeneous environmental information perceived in a discrete time interval $T = [t - \delta, t],\ t - \delta < t$. An ACW has $|C_t| = |T|$ homogeneous **context entries**, one for each time instant. Each ACW $C_t$ is associated with an index $t$ that corresponds to the time instant in which the information at time $t$ has been perceived.*

**Definition 6** (**Context Entry**). *A Context Entry $E_t^i \in \mathbb{R}$ is a numerical information perceived at time $t \in T$, where $T$ is the time window of the $C_i$. The value of a context entry can be any type of environmental information such as temperature, humidity, lightness etc.*

**Definition 7** (**Ambient Context Agent**). *An Ambient Context Agent (ACA) can be associated with a sensing device; its goal is to provide environmental information in a local part of the environment. Thanks to cooperative behavior, ACAs are capable of providing estimates even if ad hoc sensors are unavailable. Each ACA is characterized by a knowledge base containing ACWs. Each ACW belongs to a unique ACA.*

We also define a second type of agent, called **Real Sensor Agent** (RSA). Each RSA is associated with a unique sensing device and is defined as follows:

**Definition 8** (**Real Sensor Agent**). *A Real Sensor Agent (RSA) is any physical instrumentation that can provide accurate environmental information value (such as a temperature). A RSA is not associated with any confidence zone[2]. Its goal is to provide environmental information to ACAs. RSAs have also the capacity of learning from the environment by creating ACWs related to the observed information.*

The difference between ACA and RSA is that the former can exist without being associated with sensing devices; the RSAs cannot exist unless they are associated with the sensing devices. The idea behind the use of ACAs and RSAs is that we suppose that the environment is instrumented with *some* sensors. Without sensors, ACAs cannot provide an estimate, therefore these would be useless. For this reason, we suppose that some sensors are available in the environment and are associated with RSAs. The choice to use ACAs or RSAs depends on the context in which HybridIoT is used. Because ACAs can be used as virtual sensors (that is, not agentifying sensing devices), they must rely on the information acquired from reliable sensing devices associated with RSAs. For that, it is not possible to use only ACAs.

Having defined both ACA and RSA, from this point we will refer to **agent** without any difference between **ACA or RSA**.

[2] A confidence zone defines the neighborhood in which the agent bases its reasoning. See Definition 9.

**Does a component have only a limited rationality?** Agents operate in the local part of the environment in which they are situated and observe the evolutionary dynamics of the environment. Their rationality is limited because the environmental dynamics observed by some agents are different from those observed by other agents.

**Is a component "big" or not? Is it able to do many actions, to reason a lot? Does it need significant abilities to perform its own task?** Yes, the ACAs have to perceive data each one through a sensing device, adjust their neighborhood according to the information they perceive, and estimate data in case of an unpredictable NCS. These tasks can be more or less complex depending on the amount of knowledge acquired by the agents. Moreover, the previous tasks are done differently depending on the type of information perceived.

**Can the behaviour of a component evolve? Does it need to adapt to the changes of its environment?** ACAs adapt to the environmental context in which they are situated by determining the strategy for estimating missing information. Both ACAs and RSAs adapt to the environmental context in order to learn the evolving dynamics of the perceived information.

### 5.5.3 Agents' Characterization

The agents in HybridIoT provide a means to observe the environment, learn its dynamics and estimate missing information where *ad hoc* sensors are not available (for ACAs). Agents can be associated with sensing devices or placed in points of the environment where sensors are not available, thus acting as **virtual sensors**. In this way, it is possible to address the lack of sensors in large-scale environments to ensure sufficient information coverage.

Agents adapt to the environment in which they are situated by learning from observation and interacting cooperatively with other agents without any external intervention or configuration. By learning from direct observation of the environment, the agents acquire knowledge that can be used to estimate missing information. This knowledge consists of the information observed by the sensors during their operation.

The relevance of the AMAS approach is particularly reflected in the way the agents interact. An agent may encounter problems during its interactions with other agents or with the environment: the interaction protocol may not be respected or the interaction itself may be a source of errors or failures that may be the result of unpredictable events, due to the dynamic nature of the environment or sudden sensing devices' malfunctions. Such unpredictable events result in exceptions known in the AMAS theory as *Non-Cooperative Situations* (or NCS).

To characterize the agents in HybridIoT, the following criteria are discussed for each entity:

- is it autonomous?
- does it pursue a local goal?
- does it have to interact with other entities?

- does it possess a partial view of the environment?
- does it possess certain negotiating skills?
- does the entity act in a dynamic environment?
- do the entity carries out a function?
- are the entity likely to face failures in cooperation?
- do the entity deal with Non-Cooperative Situations?

The advantage of the AMAS is based on the cooperation between agents to find an acceptable solution to a complex problem. In the case of HybridIoT, we are interested in estimating missing information of environmental type (representable in numerical form), using information from a large number of devices. The reasons why the AMAS approach is relevant in the context of the proposed approach can be summarized as follow:

- because we aim at providing a general estimation method for any kind of numerical information, it is not possible to specify *a priori* the way agents will estimate information in case a real sensor is not available. Therefore, agents cannot know what information will have to be integrated to calculate a precise estimate, neither the function used to estimate a value;
- when information is not available, ACAs cooperate to provide an estimate by exploiting their knowledge, which consists of a set of information describing the evolution of the previously observed environmental values. Because a large number of sensing devices may be available in a large-scale context, it is not possible to specify the interactions between all sensors to decide with which agents should cooperate: this would lead to a significant computational overhead due to the significant amount of information to be processed and the necessary coordination between sensing devices.

The following subsections describe the ACAs and the RSAs according to the properties previously presented.

**Ambient Context Agent (ACA)**

ACAs can be associated with sensing devices: in other words, ACAs can agentify physical sensing devices or not. In the first case, the role of the ACAs is to provide a mechanism of resilience and learning to sensing devices to provide estimates in a short amount of time whereas the sensing devices are not able to perceive the physical environment. In the other case, the ACA can provide accurate estimation in points of the environment where sensing devices are not present through cooperation with the other agents (either RSAs and ACAs).

The objectives of an ACA are triple:

1. to provide accurate environmental information through estimation or direct observation of the environment;

2. to modify its **confidence zone**, a region of the physical space centered around the position of the ACA and whose devices within it provide information coherent with that perceived by the ACA. The confidence zone is defined as follows:

   **Definition 9.** *[**Confidence Zone**] a confidence zone is an $n$-side polygon associated with an ACA. An ACA uses a confidence zone to group sensors whose perceived information follows similar dynamics. The confidence zone is modified in real-time to keep within the region only the devices that provide coherent information.*

3. to learn the dynamics of environmental information.

The first two objectives are interdependent and cannot be pursued individually by ACAs. The ACAs cooperate with other agents to provide accurate estimates when sensors are not available. During the cooperation phase, an ACA discriminates the information obtained from other agents to exclude those providing information that is not relevant for the calculation of the estimate; as a matter of fact, some sensors can provide information that could lead to an imprecise estimate. As a result of this process, the ACAs dynamically modify their confidence zones to include the agents that provide appropriate values for the calculation of the estimate and to exclude agents that do not provide pertinent information. A confidence zone can be modified only if an information, either real or estimated, is available.

Each ACA operates independently from other entities of the system: this means that the ACAs do not depend on other components of the system, either internals or externals, to pursue their goals. Therefore, the ACAs are autonomous in their decisions. The goal of an ACA is local as it operates in a limited part of the environment. This suggests that in case of unavailability of sensors, the ACAs cooperate with a limited set of ACAs in their proximity. The ACAs are also responsible for learning the evolving dynamics of the environment in which these are situated by assembling ACWs containing data acquired in discrete time intervals.

The proposed approach is able to cope with unpredictable exceptions that may prevent the ACAs from perceiving information from the environment. These exceptions, known as *Non-Cooperative Situations* (NCS), are solved through the joint activity of agents. The interactions between agents require negotiation skills for the agent that has encountered an NCS; it has to be able to establish the most useful information to achieve its goal. These properties ensure that ACAs address the properties of the AMAS approach.

Figure 5.7 shows the activity diagram depicting the interactions between the ACAs and the other entities.

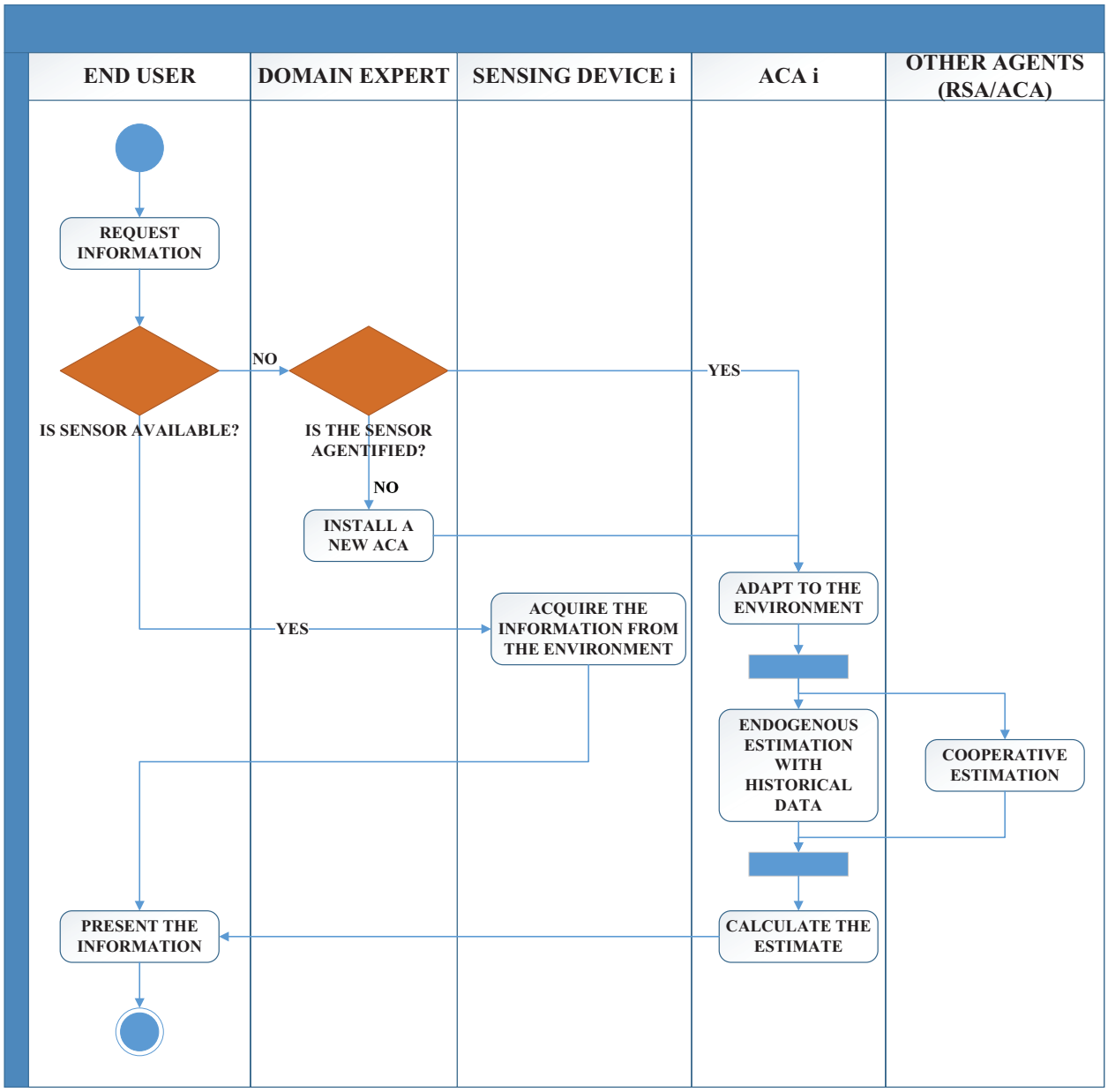



Figure 5.7: Activity diagram showing the interactions between the ACAs and the other entities.

**Real Sensor Agent (RSA)**

Unlike ACAs, RSAs are associated with physical sensing devices available in the environment (that is, with devices that can provide real information by direct observation of the surrounding environment) but they do not provide any estimation. RSAs operate autonomously, thus independently from other components of the system, by perceiving information and learning the evolving dynamics of the environment. The operation of a RSA depends only on the availability of the device with which the agent is associated. Therefore, a RSA can exist only if it is associated with a sensing

device.

RSAs pursue two goals:

- **perceiving the the local part of environment in which the RSAs are located and learning its evolving dynamics**: to pursue learning in a dynamic environment where RSAs are plunged, each RSA builds ACWs that describe the evolution of perceived information in discrete time intervals.
- **cooperating with ACAs**: the RSAs cooperate with the ACAs by providing these with its perceptions to help the ACAs in pursuing their goal, which consists of providing accurate estimates in case of device unavailability. The RSAs do not have any negotiation skills, as their cooperative behavior consists solely in helping the ACAs by providing information.

Because RSAs are associated with sensing devices, a malfunction of the latter would consequently cause the RSAs to fail to operate properly. This includes problems such as the interruption of the network connection or insufficient battery level. These problems are not treated in this thesis as we assume that RSAs are only able to operate thanks to the proper functioning of the sensing device with which they are associated. For this reason, RSAs are not designed to deal with NCSs; we say that RSAs are autonomous agents, but not AMAS.

Figure 5.8 shows the activity diagram depicting the interactions between the RSAs and the other entities.

## 5.6 Environment

Once the objectives of the HybridIoT system has been defined and the use of the AMAS approach for the problem addressed is justified, we define the environment in which the agents operate. The environment in HybridIoT can be characterized by the following properties [118]:

- The environment is **dynamic**: the active entities present in HybridIoT environment have their dynamics. These can be either fixed or mobiles (in case of smartphones), they can interact with the environment and other entities and modify their behavior according to the evolution of the environment.
- The environment is **discrete**: because sensors perceive data at discrete time intervals. The fact that the environment is discrete does not constitute a limitation, but this property reflects the way the devices perceive information from the environment.
- The environment is **partially observable**: a sensor cannot have a visibility of the entire environment. Consequently, actions are performed and have an effect on a local part of the environment. Also, actions performed on local parts of the environment do not influence the activity of other sensors.

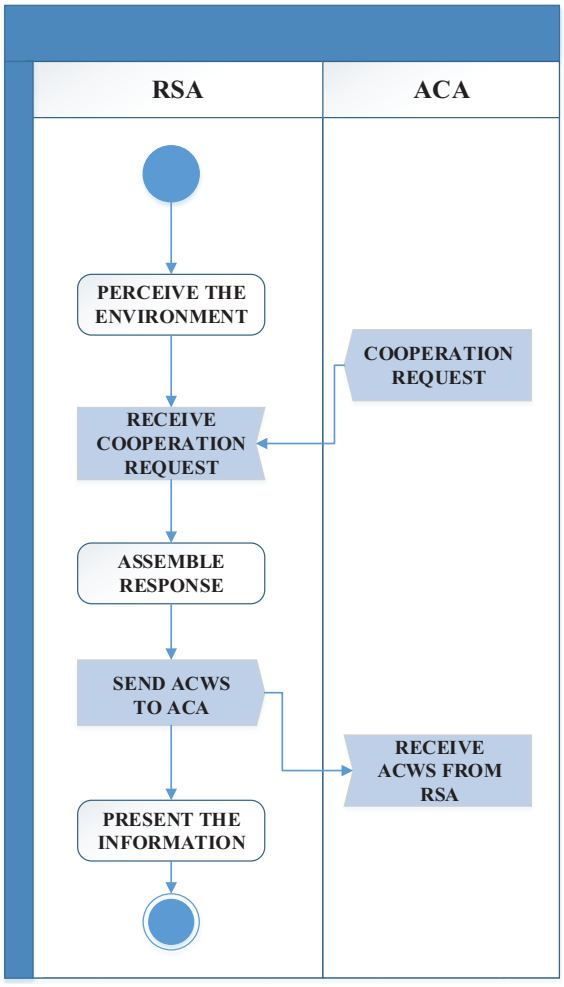



Figure 5.8: Activity diagram showing the interactions between the RSAs and the other entities.

- The environment is **episodic**: each action of an agent depends on the actions performed previously. When an ACA calculates an estimate at time $t$, the resulting estimate can be used at time $t + k$ when a further missing information needs to be calculated, thus influencing the calculation of the latter estimate.

## 5.7 Nominal Behavior

The nominal behavior of agents in HybridIoT consists of providing environmental information at any time and everywhere.

During the operation of agents, there is no external control over their functioning, neither they have any initial knowledge about the information perceived by sensors.

### 5.7.1 Agents Nominal Behavior

This section describes the behavior of an agent in HybridIoT. Agents have two types of behaviors that can be separated into nominal and cooperative, according to the AMAS theory. The nominal behavior consists of a set of skills needed by agents to pursue their goals. The cooperative behavior enables the agent to self-adapt to abnormal, unpredictable situations [17].

#### ACAs Nominal Behavior

The nominal behavior of an ACA in HybridIoT consists of four goals:

- **perceiving the environment**: this activity is carried out through physical sensors, capable of acquiring information from direct observation of the environment. Sensors have a limited capacity range and may be subject to unpredictable events that lead to errors in measurement, or even malfunctioning that avoids sensors to provide information. Agents in HybridIoT do not make any assumption on the hardware, on the information perceived and the eventual mobility of a sensor;
- **modifying the confidence zone**: the confidence zones define the local part of the environment observed by the ACAs and their rough area of relevance. The ACAs modify the shape of their confidence zone to group together the agents whose perceived information follows similar dynamics. The agents within a confidence zone (either ACAs or RSAs) provide reliable values used to estimate missing information in case of unpredictable events. The confidence zone is defined as a polygon centered in the position where the corresponding agent is situated. When ACAs are created, each one is associated with a new confidence zone; the ACAs modify their confidence zones according to the values perceived by the other agents (either ACAs or RSAs) that are situated inside the confidence zones;
- **online environment learning**: agents are capable of learning the dynamics of the environment without pre-processing data. Each agent exploits data windows containing consecutive information in time to find recurrent dynamics in the historical data. The data windows, or ACWs, have a variable quantity of values depending on the variability of the observed data;
- **presenting the information to the user**: the result of the system is the information presented to the end-user, who must not be capable of determining whether the information has been perceived by an *ad hoc* sensor or estimated. Therefore, the operation of HybridIoT is transparent to the end-user.

#### RSAs Nominal Behavior

The nominal behavior of a RSA consists of acquiring information through direct observation of the environment; the RSAs are associated with *ad hoc* sensors and contrarily to ACAs they do not

estimate any information. Because the objective of the RSAs is to provide environmental information to the ACAs in the system, they do not have confidence zones.

The RSAs learn the dynamics of the environment by determining ACWs; these can be provided to the ACAs in proximity that use the context windows to calculate accurate estimates in case of unpredictable events that prevent ACAs to provide environmental information.

RSAs have no cooperative behavior as their objective is only to provide information acquired by the sensors to the ACAs that have to estimate missing information. Moreover, the only considered failures of RSAs are related to the sensing devices with which agents are related, such as a low battery or network issues. These failures are not considered because RSAs only operate when the sensing devices can perceive the environment and send these to ACAs.

Algorithm 1 describes the behavior of a RSA. At line 2, the RSA perceives the physical environment through an *ad hoc* sensor. At line 5 the RSA checks if it is situated within the confidence zone of the $\text{ACA}_i$ and this ACA has encountered an incompetence NCS (represents the inability of the sensor to provide information, this NCS described in Section 5.8.1): if so, the RSA provides the ACA with its perception (line 6). Finally, at line 8 the RSA adds the perceived information $p$ to the current ACW. The process of determining ACWs is described in Section 5.10.

**Algorithm 1** RealSensorAgent

1: {—*perceive*—}
2: $p \leftarrow$ perceiveFromSensor()
3: {—*decide and act*—}
4: **if** inConfidenceZone($\text{ACA}_i$) $\wedge$ incompetenceNCS($\text{ACA}_i$), $\forall$ $\text{ACA}_i$ **then**
5: $\text{ACA}_i \leftarrow$getACA()
6: sendPerception($p$, $\text{ACA}_i$)
7: **end if**
8: addToACW($p$)

### 5.7.2 Discussion

This section discussed the nominal behavior of the agents composing the HybridIoT system. Both RSAs and ACAs act autonomously on the local part of the environment in which they are situated; they can perceive the environment and observe its dynamics. The environment responds to the characteristics of a real one: dynamic, constantly evolving, unpredictable. This enables HybridIoT to be employed in a real urban context.

The nominal behavior of both ACAs and RSAs is adequate to provide information in large-scale environments. Nevertheless, the proposed approach must consider unpredictable sensing device malfunctions, as well as the possibility for new sensing devices, perceiving information which type is not known in advance, to enter or leave the system at any time. The AMAS methodology provides the necessary tools to address these challenges and implementing the wished estimation technique.

The next section discusses the cooperation behavior of agents, used to address unpredictable in which information is missing.

## 5.8 Cooperative Behavior

This section discusses the non-cooperative situations, i.e. exceptions that are encountered by ACAs during their nominal operation, that are solved thanks to the joint activity of the agents. The idea behind the cooperative behavior is that an ACA cooperates with other agents that could compensate, through their perception, the action of the ACA that encountered an unpredictable event.

### 5.8.1 Non-Cooperative Situation

#### Incompetence NCS

**NCS description**: an ACA cannot provide any environmental information because of an unpredictable, unexpected problem;
**Detection**: the ACA attempts to query the associated sensing device to provide information to the end-user. In case the device is not present or unavailable due to an unexpected malfunction, the ACA states that an incompetence NCS occurred as it is unable to fulfill its goal of providing environmental information.
**Resolution**: the resolution of the incompetence NCS requires the adaptation of the ACA to the environmental context at the time instant in which it encounters the NCS. The adaptation process enables the ACA to determine the appropriate estimation mechanism to be used to estimate the missing information. The ACAs solves incompetence NCS by pursuing two types of estimation processes:

- **endogenous estimation by historical data**: the ACA exploits the information previously observed and calculates the estimate by exploiting its knowledge, which consists of the information previously perceived from the environment;
- **endogenous estimation by confidence zone**: the ACA cooperates with the agents within the confidence zone that perceive the same type of information as the ACA that encountered the NCS;
- **exogenous estimation**: the ACA cooperates with the agents that perceive heterogeneous information and are situated beyond its confidence zone.

#### Uselessness NCS

**NCS description**: an ACA cannot help the agent that encountered an incompetence NCS because its perceived information is significantly different.

**Detection**: when an ACA is unable to provide information from direct observation of the environment, it cooperates with other available agents to estimate the missing information. The cooperative resolution process consists in an exchange of information between the agents so that the ACA that encountered the incompetence NCS can make use of the data provided by agents to calculate the estimate. Because a large number of agents may be present, not all the agents provide useful information for the ACA that encountered a NCS. This is due to the fact that the agents are situated in different physical environments, therefore the observed environmental dynamics are also different. For this reason, the ACA must discriminate the information that has been received cooperatively from the agents to avoid the introduction of noise that could compromise the accuracy of the estimate. Through this process of discrimination, agents are able to determine a level of mutual confidence: the agents whose information will be discarded will have a lower level of confidence than those that provide information relevant to the calculation of the estimate.

**Resolution**: to solve a uselessness NCS, the ACA that encountered an incompetence NCS ignores the agents that provide information which, numerically, is significantly different from those perceived by the ACA that encountered the NCS. The ACA acts in such a way to cooperate with the agents that perceive similar environmental dynamics.

## 5.9 Estimation Procedure

This section presents the estimation technique pursued by agents. Figure 5.9 shows the main steps of the proposed technique, which is pursued by ACAs to provide environmental information and estimates in case of unavailability of physical sensing devices.

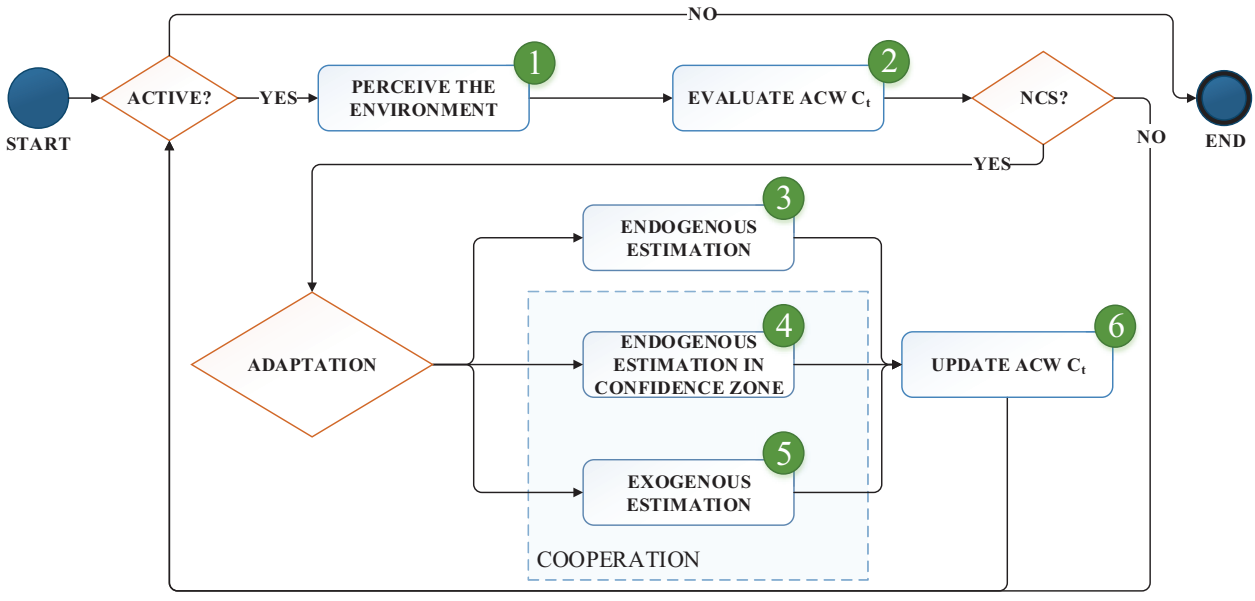



Figure 5.9: Main steps of the proposed estimation technique.

Let $ACA_i$ be the $i$-th ACA available in HybridIoT. If the $ACA_i$ agentifies an existing sensing device, then it is able to perceive the environment, thus a new information is presented to $ACA_i$ (step ❶). Following, a new ACW is created and associated with the perceived information (step ❷). The new ACW is added to the set of ACWs of the ACA. Each ACW contains a variable number of entries; Section 5.10 describes the process pursued by agents for determining ACWs of dynamic size.

If the information at time $t$ is not available due to the unavailability of the device or even a missing device, the $ACA_i$ is faced to an *incompetence NCS* as it is not able to provide any information. ACAs try to solve incompetence NCS by adapting to the environment condition and pursuing one of the three available estimation methods presented afterward (steps ❸, ❹ and ❺).

We identified three conditions according to which ACAs are able to adapt for estimating missing information:

1. *ACAs are missing* (step ❸): in this case, a unique ACA cannot pursue a cooperative behavior and must exploit its knowledge to estimate missing information. ACAs embody learning skills: they are aware of the environmental context in which they are situated, perceive its characteristics, and learn its evolutionary dynamics. The learning phase, which coincides with the automatic determination of ACWs, enables agents to determine recurrent and precise information that can be exploited to estimate missing values. When there is no ACA to cooperate with, the ACAs use the ACWs in their knowledge base to estimate the missing information.

2. *ACAs are available in the local part of the environment observed by the ACA* (step ❹): in this case, the ACAs estimate the missing information using the values provided by the other agents, within the confidence zone, that perceive the same type of information. In this case, the agents cooperate to provide the ACA (that encounters an incompetence NCS) with useful information for estimating the missing information.

3. *ACAs providing heterogeneous information are available beyond the local part of the environment observed by ACAs* (step ❺): to overcome the lack of sensing devices perceiving the same type of information as the ACAs that encounter an incompetence NCS, ACAs exploit heterogeneous information to estimate the missing values. Moreover, there are no ACAs available to perform a local estimation within the confidence zones. Therefore, in this case, ACAs pursue a cooperative behavior with agents perceiving heterogeneous information to estimate the missing values.

The last two estimation methods (steps ❹ and ❺) involve the cooperation between ACAs to solve the incompetence NCSs. The idea behind the cooperative behavior is that $ACA_i$ cooperates with other ACAs, regardless of the type of perceived information, that could help $ACA_i$ to estimate a missing information. The cooperation requires negotiation skills between ACAs: not all the agents

provide useful information for estimating a missing information, therefore ACAs discriminate between those that provide pertinent information from those providing values considered as *outliers*. This enables ACAs to provide accurate estimates.

The last step (step ❻) consists of updating the knowledge base of the agent: the estimate (or real perception) is used to determine the ACW that represents the information. To do this, ACAs determine ACWs of dynamic size, as discussed in section 5.10.

We highlight the advantages of the proposed estimation technique compared to the state of the art methods:

- **openness**: the ACAs do not need any prior knowledge on the available devices in the environment. They can adapt to the environmental context in which they are situated and resolve any unpredictable situations that could lead to the unavailability of the information;
- **heterogeneity**: the ACAs can perceive environmental information whose types are not defined in advance. This enables using the same estimation technique in environmental contexts where different types of information are perceived by sensors, without any particular configuration;
- **large-scale**: the ACAs pursue a local computation. They operate autonomously, therefore the computation of an ACA does not affect the others. This enables distributing the estimation technique in large-scale environments.

The rest of this section is organized as follows: section 5.9.1 presents the endogenous estimation scheme using the historical data (step ❸). Then, section 5.9.2 presents the estimation scheme that exploits the information perceived by agents within the confidence zone of the ACA that encountered a NCS of incompetence (step ❹). Finally, section 5.9.3 presents the exogenous estimation scheme that enables ACAs to estimate missing information through the integration of heterogeneous information (step ❺).

### 5.9.1 Endogenous Estimation by Historical Data

This section describes the endogenous estimation scheme where ACAs rely on homogenous data to provide estimates for missing information.

The endogenous estimation of missing information is divided into two steps:

1. **Cooperative weights evaluation**: ACAs evaluate the weights to be used for estimating missing information;
2. **Missing information calculation**: ACAs estimate missing information using the weights evaluated previously.

Figure 5.10 shows the main steps of the endogenous estimation performed by ACAs. The *Endogenous Estimation* part in this diagram is contained in the step ❸ of Figure 5.9.

If an incompetence NCS occurs and no other ACAs are available in the proximity of the agent, then the ACA pursues an endogenous estimation by using the information previously perceived. This process includes two steps: the evaluation of the weights that have been calculated using the previously perceived information (step ❶) and the calculation of the estimate (step ❷).

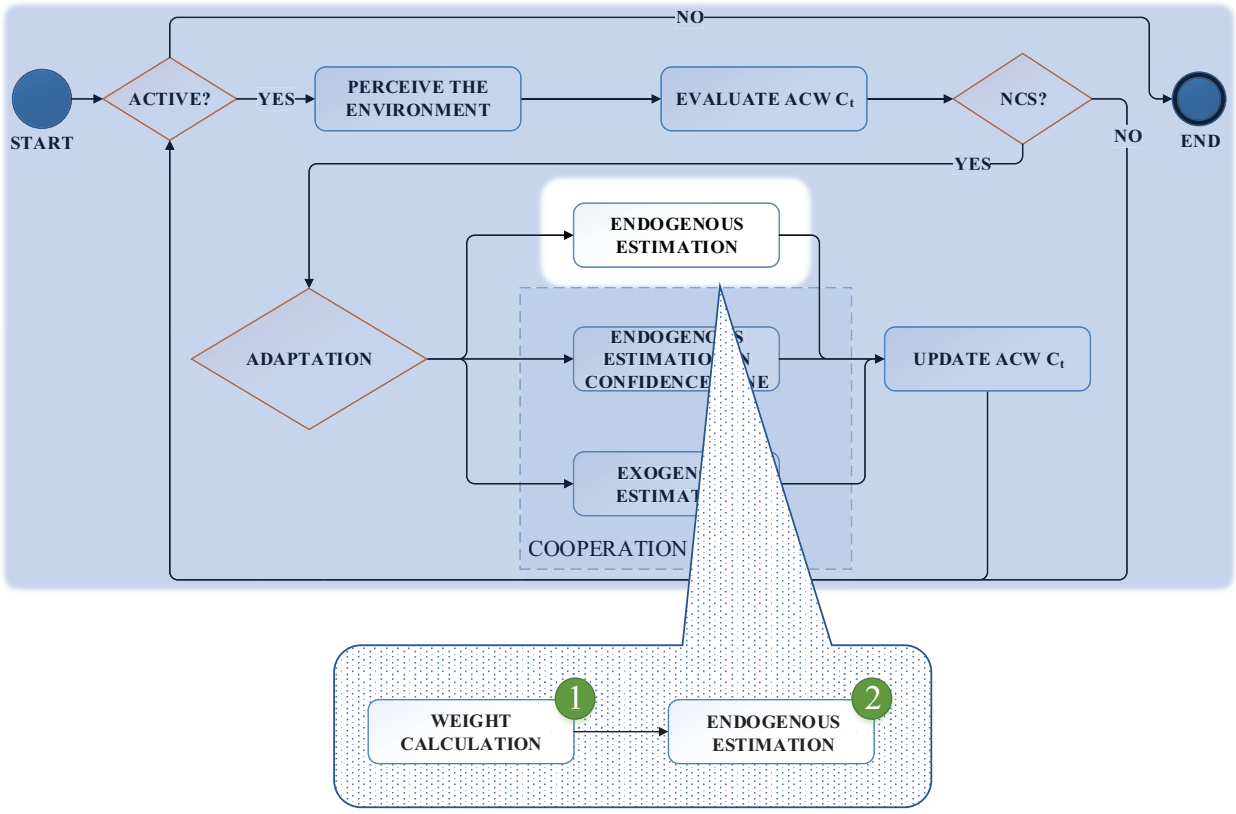



Figure 5.10: Main steps of the endogenous estimation process performed by ACAs.

### Weights Calculation

Let $t$ be the time instant at which the information needs to be estimated by $\text{ACA}_i$. For the sake of simplicity, we assume that $\text{ACA}_i$ has assembled the ACWs for the information before $t$.

The $\text{ACA}_i$ evaluates a subset $\xi^{(t)}$ containing the ACWs that minimize the distance $d(C_t, C_k), \forall C_k \in \xi, k \neq t$ where $C_t$ is the ACW containing the information to be estimated at time $t$ and $C_t \notin \xi^{(t)}$.

The $\text{ACA}_i$ evaluates a weight $w_t$ obtained as a weighted average of the difference between the last two values of each context $C_k \in \xi^{(t)}$. The distances between $C_t$ and the ACWs in $\xi$ are used to calculate the weight $w_t$: the smaller the distance between an ACW $C_k \in \xi^{(t)}$ and the ACW $C_t$, the more likely the missing information at time $t$ will be similar to the last value of the ACW $C_k$.

The distance $d$ between two ACWs is defined as follows:

**Definition 10 (ACW Distance).** *the distance between two ACWs is defined as the absolute difference*

*in time between the context entries divided by the number of entries $\gamma$ of the two ACWs. The smaller the difference is, the more similar two ACWs are. The context distance between two ACWs $C_t$ and $C_k$ is defined by the following formula:*

$$d\left(C_t, C_k\right) = \frac{\sum_{l \in [1,\gamma]} |E_l^t - E_l^k|}{\gamma} \tag{5.8}$$

*where $\gamma = |C_t| = |C_k|$, $l$ is an index in the range $[1, \gamma]$, $E_l^t$ and $E_l^k$ are the context entries of index $l$ in the ACW $C_t$ and $C_k$ respectively.*

*The calculation* (5.8) *of the distance $d$ is generic as it does not consider the unit of the information used; therefore it can be used for any type of numerical environmental information.*

*The distance $d$ satisfies the following properties:*

- $d(C_t, C_k) \geq 0$,
- $d(C_t, C_k) = 0 \iff C_t = C_k$,
- $d(C_t, C_k) = d(C_k, C_t)$,
- $d(C_t, C_k) \leq d(C_t, C_p) + d(C_p, C_k)$ *where $C_t$, $C_k$, $C_p$ are ACWs relative to time instants $t$, $k$, $p$ respectively.*

*The distance $d$ corresponds to the L1-norm. Figure 5.11 shows two ACWs and the surface between them, which area is used as similarity measure.*

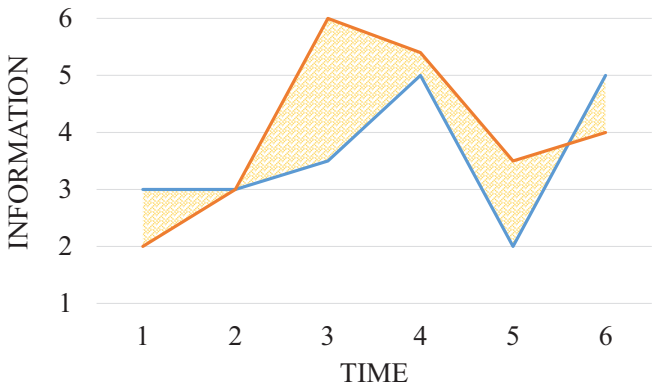



Figure 5.11: Surface between two ACWs (orange and blue) which area is used to compare their similarity.

The calculation of $w_t$ is the average of the differences of the last two context entries for each ACW $C_k \in \xi$ by using the distance $d(C_k, C_t), \forall C_k \in \xi$. The value $w_t$ is calculated in such a way to be independent from the sampling rate of the data. The weight $w_t$ is computed as follow:

$$w_t = \frac{\sum_{C_k \in \xi^{(t)}} \left( E^k_\ell - E^k_{\ell-1} \right) \cdot d\left(C_t, C_k\right)}{\sum_{C_k \in \xi^{(t)}} d\left(C_t, C_k\right)} \tag{5.9}$$

where $C_t$ is the ACW containing the information to be estimated at time $t$, $C_k \in \xi^{(t)}$, $|\xi^{(t)}| = 10$, is the $k$-th most similar ACW to $C_t$ and $E^k_\ell$ and $E^k_{\ell-1}$ are respectively the $\ell^{th}$ and $(\ell-1)^{th}$ context entries of the ACW $C_k \in \xi^{(t)}$, namely the last two context entries of $C_k$.

We observed through experiments that HybridIoT is capable of estimating accurate information even with a limited number (10) of ACWs. This is why the set $\xi^{(t)}$ contains a maximum of 10 ACWs. Moreover, the use of a limited number of ACWs is advantageous because it enables avoiding noise in the estimation process.

**Estimating Missing Information**

Let $C_j$ be the ACW of the ACA$_i$ containing the information to be estimated at time $t$. The estimated context entry $E^j_t$ is computed as follows:

$$E^j_t = E^j_{t-1} + w_t \tag{5.10}$$

where $E^j_t$ is the estimate for the missing information at time $t$, $E^j_{t-1} \in C_j$ is the last information perceived by ACA$_i$, $w_t$ the weight obtained through a cooperation behavior between ACA$_i$ and the other available agents. The weight $w_t$ is calculated by Equation (5.9). Equation (5.10) considers the last information acquired because we assume that environmental information perceived in consecutive temporal instants do not present relevant changes.

If there is no other agent in the neighborhood of the ACA, then $E^j_t$ is the estimated value for the missing information at time $t$. Finally, the ACW $C_j$ is updated by adding the context entry $E^j_t$.

**Scenario Example**

Consider the scenario in Figure 5.12 where the sensing devices $S_1$, $S_2$ and $S_3$ are agentified; each one is therefore associated with a unique ACA operating autonomously and locally. Suppose that the sensing device $S_1$ has to estimate the missing information at time $t$ due to a sudden device malfunction and ACA$_1$ is the agent associated with $S_1$; suppose that ACA$_1$ has previously collected a significative quantity of information from the environment; the ACA$_1$ disposes of a set of data windows representative of the environmental dynamics observed in the room where $S_1$ is located. Having this knowledge, ACA$_1$ can then estimate missing information at time $t$ using the previously observed information. The estimate is calculated using the previously observed environmental dynamics to calculate the difference that the estimated information will have with the last information perceived by the device.

The estimation process is carried out by ACA$_1$ as follows: ACA$_1$ finds the most similar data windows to the one containing the information to be estimated. The resulting data windows are

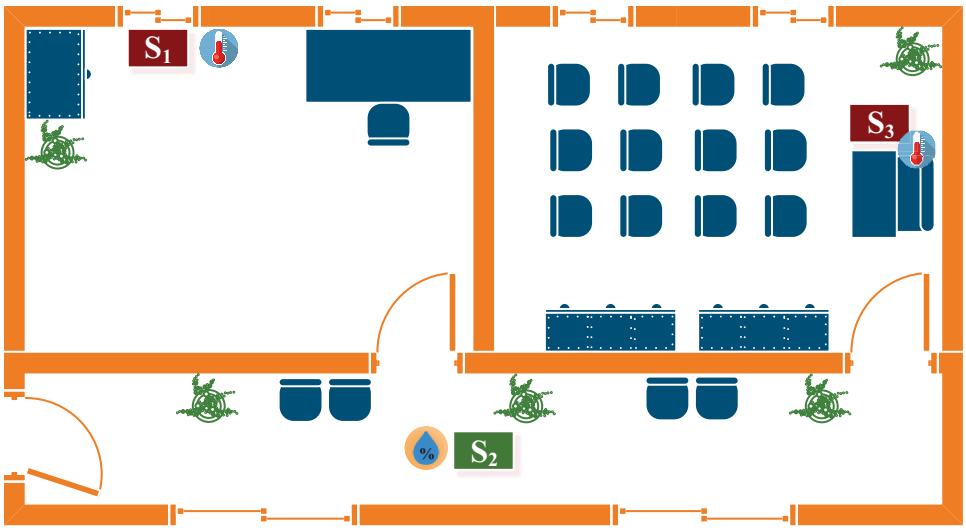



Figure 5.12: Case study illustration. Three sensors are deployed into a building context: two temperature sensors ($S_1$, $S_3$) and a humidity sensor ($S_2$).

used to indicate how the information to be estimated varies from the last perceived value.

The solving process for estimating the temperature at time $t$ is roughly the following:

1. $ACA_1$ encounters an incompetence NCS at time $t$. Let $ACW_t$ be the data window containing the information to be estimated;

2. $ACA_1$ determines a set $\xi^{(t)}$ of ACWs that are the most similar to $ACW_t$ (the Equation (5.8) is used to calculate the distance between the ACWs);

3. the ACWs in $\xi^{(t)}$ are used to calculate a weight $w_t$ by using the Equation (5.9);

4. $ACA_1$ estimate the missing information by the Equation (5.10).

For the sake of example, let $ACW_t$ be the context window in which information at time instant $t$ has to be estimated because the device associated with the ACA is not available. Let $ACW_k$ and $ACW_p$ be two other ACWs whose values are similar to those in $ACW_t$.

Figure 5.13 shows the trend of the temperature values of the three ACWs. Also, Table 5.1 reports the temperature values of the ACWs. $ACW_k$ and $ACW_p$ have a similar trend to $ACW_t$: this makes these ACWs pertinent for estimating the missing information for $ACW_t$. This is possible thanks to the distance measure in Equation (5.8), which allows finding ACW whose trend is similar to that of an arbitrary context window.

The weight $w_t$ for this example is calculated as follows (by using the Equation (5.9)):

$$w_t = \frac{(0,2 \cdot 0,090) + (-0,7 \cdot 0,107)}{0,090 + 0,107} = -0,288.$$

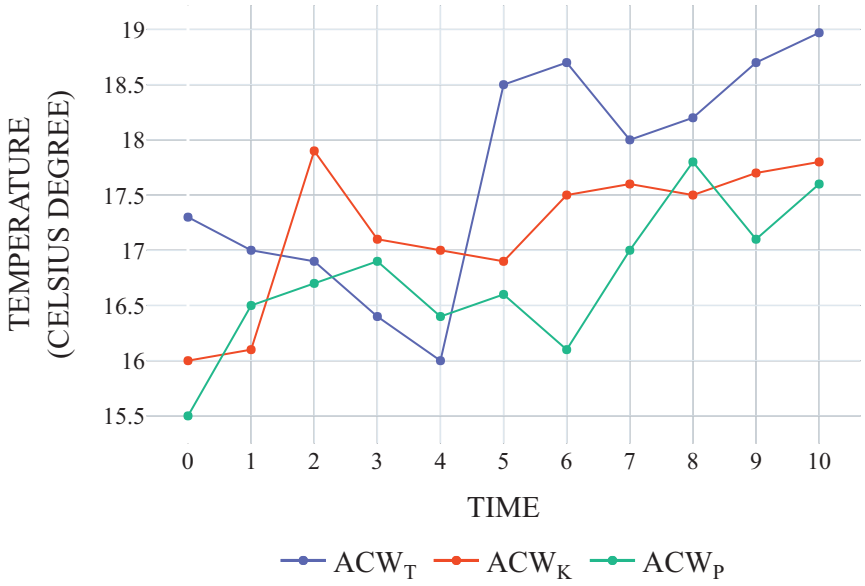



Figure 5.13: Plot of the temperature values in the ACW$_t$, ACW$_k$ and ACW$_p$, owned by the ACA$_1$.

Therefore, the estimate for the missing value at time $t$ is calculated as the sum of the information at time $t-1$ and the weight $w_t$:

$$e = E_t^l + w_t = 18,4°\text{C}$$

where $E_t^l = 18,7°\text{C}$ is the last observed context entry in ACW$_t$.

Table 5.1: The temperature values of the ACWs shown in Figure 5.13.

| | **Temperature Value** | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| **ACW$_t$** | 17,3 | 17 | 16,9 | 16,4 | 16 | 18,5 | 18,7 | 18 | 18,2 | 18,7 | **?** |
| **ACW$_k$** | 16 | 16,1 | 17,9 | 17,1 | 17 | 16,9 | 17,5 | 17,6 | 17,5 | 17,7 | |
| **ACW$_p$** | 15,5 | 16,5 | 16,7 | 16,9 | 16,4 | 16,6 | 16,1 | 17 | 17,8 | 17,1 | |

The endogenous estimation technique enables agents to estimate missing information using previously observed information. We consider the endogenous estimation as a simple resilience mechanism for sensing devices that enables providing estimates of environmental information in a short amount of time in case of sudden and unpredictable unavailability of the devices.

In the case study discussed, we assumed that ACA$_1$ has enough information to allow it to calculate an accurate estimate for the missing information. But what would happen if the ACA$_1$ does not have enough ACWs in its historic? Assuming other devices are present in the immediate proximity of ACA$_1$, is it possible to estimate accurate values through a mechanism of cooperation between the agents associated with the sensing devices? This is the idea behind the estimation in the confidence zone, introduced in the following section.

### 5.9.2 Endogenous Estimation by Confidence Zone

Suppose an $ACA_i$ agentifies the $i$-th available sensing device in the environment and that at time $t$ the sensing device is unable to perceive the local part of the environment; therefore the $ACA_i$ encounters an incompetence NCS. If some ACAs are available in the confidence zone of the $ACA_i$, then the ACA adapts its behavior to pursue an estimation by cooperating with the agents situated inside the **confidence zone (both ACAs and RSAs)**. This type of estimation can be pursued when the nearby agents perceive the same type of information as the one that encountered the incompetence NCS; we assume that unless there are natural or artificial barriers that could lead to variations in the observed data, near agents perceive similar environmental values. The confidence zones enable ACAs to **partition the environment according to the similarity of data perceived by sensors**. In this way, the ACAs are able to group the sensors that provide useful information for the calculation of the missing information [2].

Figure 5.14 shows the main steps of the endogenous estimation by the confidence zone performed by ACAs.

The $ACA_i$ that encountered an incompetence NCS exploits the information provided by the agents within its confidence zone (step ❶). The agents are autonomous and aware of the state of the local part of physical environment; they send their perceptions to the $ACA_i$ that has encountered an incompetence NCS (step ❷). In this way, the agents cooperate to help the $ACA_i$ in the estimation of the missing information.

The $ACA_i$ evaluates the pairs of agents that are used to calculate an estimate of the missing information (step ❸). For each pair of agents, the $ACA_i$ calculates a unique **data field**, which represents an estimate for the missing value. A data field is defined as follows:

**Definition 11 (Data Field).** *A data field $\gamma$ between two sensors (either ACAs or RSAs) is a vector field in the Euclidean space. Each point is associated with a vector that is oriented towards the sensor which provides a higher data value; the magnitude is the value of the gradient between the data perceived by the sensors. Figure 5.15 shows an example of a data field between two sensors $S_j$ and $S_k$ at the point where the sensor $S_i$ is situated.*

Once the $ACA_i$ have calculated a data field for each pair of agents within its confidence zone, it calculates the estimate for the missing value at time $t$ by weighting the data fields provided by agents (step ❹). Finally, the shape of the confidence zone of the $ACA_i$ is adjusted to keep inside only the agents that provide **coherent values** (step ❺). The values considered as coherent are calculated **using a thresholding technique that discards the values that are distant from other observations**.

The described process takes place locally between agents within a specific confidence zone and is independent of the operation of other agents.

This type of estimation enables the agents in HybridIoT to provide information in a short amount of time. Thanks to the ability of ACAs to provide information in a short time, either estimated or

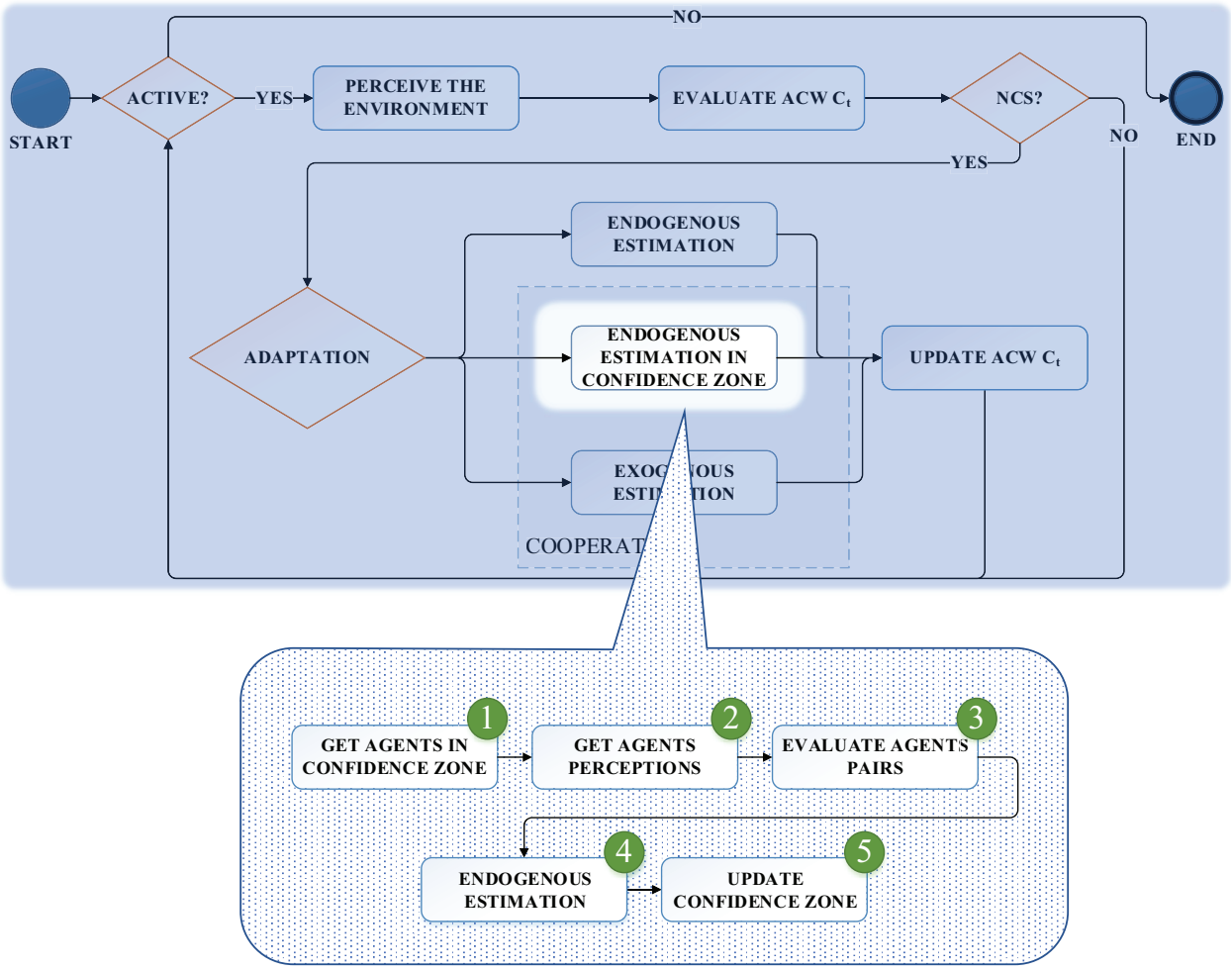



Figure 5.14: Main steps of the endogenous estimation process performed by cooperation between ACAs in the same confidence zone.

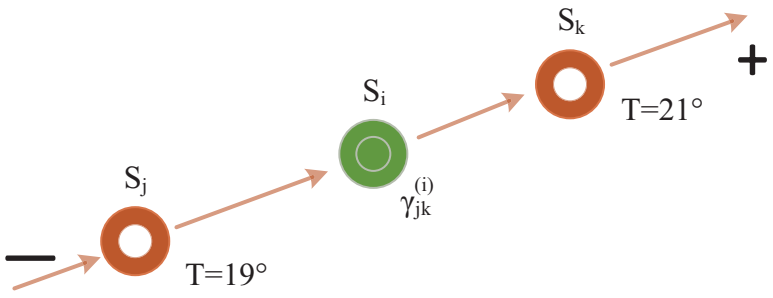



Figure 5.15: The data field $\gamma_{jk}^{(i)}$ at the point where the sensing device $S_i$ is situated is calculated as the value of the gradient between the values perceived from $S_j$ (the lower value) and $S_k$ (the higher value).

acquired from the environment, the end-user is unable to perceive the **difference between an ACA and a physical sensing device**.

Algorithm 2 describes the behavior of the $ACA_i$, precisely when a NCS that is solved by an endogenous estimation between the agents present in the confidence zone.

**Algorithm 2** Estimation in Confidence Zone

```
1: {—perceive—}
2: p ← perceiveEnvironment()
3: {—decide and act—}
4: if NCS then
5:    R ← getAgentsInConfidenceZone()
6:    P ← getPerceptionsOfAgents()
7:    D ← evaluateAgentsPairs(R, P)
8:    p ← calculateEstimation(D)
9:    updateconfidenceZone(D)
10: end if
11: updateACW(p)
```

The algorithm begins by perceiving the environment (line 2): this includes not only the direct observation of the environment in case a sensing device is available, but also the identification of the agents present in the confidence zone.

If the $ACA_i$ encounters an incompetence NCS and other sensing devices that perceive the same type of information as the $ACA_i$ are present in its confidence zone, then a cooperative estimation process is started (lines 5-9). The first step of the cooperative process consists of retrieving the list $R$ of agents inside the confidence zone of the $ACA_i$ (line 5). The agents within the confidence zone send their last perceptions to the $ACA_i$. At line 6 the $ACA_i$ receives the perceptions of the other agents in its confidence zone. Then, at line 7 the $ACA_i$ determines the pairs of agents according to their alignment to the $ACA_i$. For each pair of agents obtained, the $ACA_i$ calculates a data field using the received agents' perceptions.

At line 8 the $ACA_i$ calculates the estimate for the missing information at time $t$ by using the data fields in $D$. Finally, the shape of the confidence zone is modified (enlarged or reduced) according to the data fields in $D$ (line 9). A simple thresholding technique allows determining the pairs of agents to exclude from the confidence zone, thus considered as **outliers**. The ACAs do not have any knowledge about the optimal shape of their confidence zone, but rather reason on the perceptions and the geographical position of the agents to determine if they must be excluded or not from its confidence zone.

The last step of the $ACA_i$ consists of updating the current ACW by adding the estimated information (line 11). This process is described in detail in Section 5.10.

The next subsections describe the operations carried out during the cooperative processes and the updating of the confidence zones. For that, we suppose that $ACA_i$ is the agent that encountered an incompetence NCS and must estimate the missing information at a given time instant, and that

$AG_j$ is an arbitrary agent available in HybridIoT (either RSA or ACA).

### Determining Agents Pairs

Algorithm 3 determines the pairs of agents involved in the cooperative estimation process. At line 1 the algorithm initializes a map $C$ of pairs of agents: for each record, the key is a pair of agents, the value is their collinearity to the ACA that must estimate the information. The collinearity is defined as follows:

**Definition 12** (Collinearity)**.** *Three or more points $P_1, P_2, P_3, ...$ are said to be collinear if they lie on a single straight line $L$, as shown in Figure 5.16.*

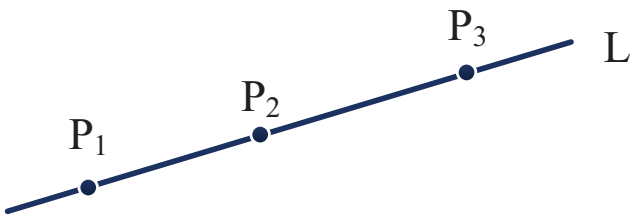



Figure 5.16: Three points $P_1, P_2, P_3$ collinear to a line $L$.

*Three points $x_i = (x_i, y_i, z_i)$ for $i = 1, 2, 3$ are collinear if the ratios of distances satisfy*

$$\begin{vmatrix} x_1 & y_1 & 1 \\ x_2 & y_2 & 1 \\ x_3 & y_3 & 1 \end{vmatrix} = 0 \tag{5.11}$$

*or, in expanded form,*

$$x_1(y_2 - y_3) + x_2(y_3 - y_1) + x_3(y_1 - y_2) = 0. \tag{5.12}$$

---

In the context of ACAs, we are interested in how much pairs of ACAs are collinear. To do this, we employ a function that calculates the degree of collinearity between three points. This can be easily interpreted as the distance of a point to the line that passes through the other two points. More precisely, given three points $P_1 = (x_1, y_1)$, $P_2 = (x_2, y_2)$, $P_3 = (x_3, y_3)$, the collinearity value is calculated by the following function:

$$f(P_1, P_2, P_3) = x_1(y_2 - y_3) + x_2(y_3 - y_1) + x_3(y_1 - y_2). \tag{5.13}$$

The value of the function $f$ provides a measure of how collinear the points are to a line $L$.

At line 2 the algorithm initializes an empty list $SP$ of pairs of agents. At line 3 the algorithm evaluates the list $agList$ of agents that provided a perception to the ACA that encountered the NCS. This list is sorted in increasing order of the distance between the agents (the Euclidean distance has been used). The loop 4-10 is executed until the list $agList$ is not empty. At line 5, the ACA retrieves the agent $AG_j$, that is its nearest agent, and it is removed from $agList$. At line 6, the ACA calculates the collinearity according to its position, the position of $AG_j$ and the remaining agents in $agList$. The ACA chooses the $AG_k$, that is, the agent that minimizes the collinearity value calculated using Equation 5.13. The collinearity value obtained is normalized in the range $[0, 1]$, where $0$ indicates the worst aligned pair regarding the ACA, while $1$ the most aligned. If one or more agents are likely to have the same collinearity, the farthest is chosen. The $AG_k$ is then removed from the list $agList$ (line 7) and paired to $AG_j$. The pair ($AG_j$, $AG_k$) is then added to the map $C$ together with the collinearity value (line 8). At line 9 the pair $(AG_j, AG_k)$ is added to the end of the list $SP$.

**Algorithm 3** evaluateRSAPairs

**Require:** the map $P$ of perceptions received by agents within the confidence zone
1: $C \leftarrow \emptyset$ {Map of collinear pairs within the confidence zone. The key is a pair of agents, the value is the related weight}
2: $SP \leftarrow \emptyset$ {List of pairs of agents. The list is sorted according to the order of evaluation.}
3: $agList \leftarrow$ sort(keySet($P$))
4: **while** $|agList| > 0$ **do**
5: $AG_j \leftarrow$ removeFirst($agList$)
6: $AG_k \leftarrow$ mostCollinearAG($agList$,$AG_j$, $ACA_i$)
7: remove($AG_k, agList$)
8: $C(AG_j, AG_k) \leftarrow$ dist($\overline{AG_jAG_k}, ACA_i$)
9: add($SP, (AG_j, AG_k)$)
10: **end while**

The next subsection describes how the $ACA_i$ calculates a data field for each pair of agents within its confidence zone.

### Determining Data Fields

Algorithm 4 determines the data fields for each pair of agents evaluated by algorithm 3. The algorithm starts by initializing an empty map $D$ containing, for each pair of agents, the value of the data field in the point where the ACA that encountered the NCS is situated. The loop 2-7 is executed for each pair $p$ of collinear agents in the map $C$. At lines 3 and 4 two variables are initialized, containing the values $v_a$ and $v_b$, respectively the last perceptions of agents $AG_i$ and $AG_k$. At line 5 the data field is calculated at the point where the ACA that encountered the NCS is situated.

A data field $\gamma$ between two sensors is a vector field in the Euclidean space; the magnitude is the value of the gradient between the data perceived by the sensors. The data field between two agents $AG_j$ and $AG_k$ in the point where $ACA_i$ is situated is calculated as follows:

$$\gamma_{jk}^{(i)} = v_j + (v_k - v_j) \cdot \frac{d(\text{AG}_j, \text{ACA}_i)}{d(\text{AG}_j, \text{AG}_k)} \tag{5.14}$$

where $v_j$ and $v_k$ are the last information perceived by $\text{AG}_j$ and $\text{AG}_k$ respectively, $d(\text{AG}_j, \text{ACA}_i)$ is the euclidean distance between $\text{AG}_i$ and the $\text{ACA}_i$, $d(\text{AG}_j, \text{AG}_k)$ is the euclidean distance between $\text{AG}_j$ and $\text{AG}_k$. Equation (5.14) calculates the data field at the point where the $\text{ACA}_i$ is situated taking into account the information perceived by the two agents and their distance from the $\text{ACA}_i$.

At line 6 the data field value $\gamma_{jk}^{(i)}$ is added to the map $\Gamma$ with the pair $p = (\text{AG}_j, \text{AG}_k)$ as key. Finally, the map $\Gamma$ containing the data fields is returned as the output of the algorithm (line 8).

**Algorithm 4** evaluateDataFields

**Require:** the map $P$ of perceptions received by agents
**Require:** the map $C$ of collinear pairs within the confidence zone; the key is a pair of agents, the value is their correlation
1: $\Gamma \leftarrow \emptyset$ map of data fields; the key is a pair of agents, the value is the data field between the agents in the pair and $\text{ACA}_i$
2: **for all** pair $p = (\text{AG}_j, \text{AG}_k) \in \text{keySet}(C)$ **do**
3: $v_j \leftarrow P(\text{AG}_j)$
4: $v_k \leftarrow P(\text{AG}_k)$
5: $\gamma_{jk}^{(i)} = v_j + (v_k - v_j) \cdot \frac{d(\text{AG}_j, \text{ACA}_i)}{d(\text{AG}_j, \text{AG}_k)}$
6: $\Gamma(p) \leftarrow \gamma_{jk}^{(i)}$
7: **end for**
8: **return** $\Gamma$

Once the $\text{ACA}_i$ calculated the data fields for the pairs of agents inside its confidence, it calculates the estimates by combining the data fields using the algorithm 4. The next subsection describes the process of calculation of the estimate.

### Calculating the Estimate

Algorithm 5 estimates the missing information using the data fields calculated using Algorithm 4. The algorithm takes as input a map $\Gamma$ of data fields and a map $C$ of collinearities between the pairs of agents within the confidence zone of $\text{ACA}_i$. At line 1, the $\text{ACA}_i$ calculates the estimate as a weighted sum of the data field values by the collinearity values of the pairs of agents. The estimate is calculated as follows:

$$est = \frac{\sum_{p \in \text{keyset}(C)} \Gamma(p) \cdot C(p)}{\sum_{p \in \text{keyset}(C)} C(p)}. \tag{5.15}$$

The Equation (5.15) returns a mean of the data fields weighted by the collinearity of the agents with respect to $\text{ACA}_i$. We suppose that agents with a higher collinearity value provide more accurate information as the related gradient provides accurate values if the $\text{ACA}_i$ is situated in the same direction of the gradient and near the two sensors.

At line 2 the estimate is added to the list of perceptions of the $\text{ACA}_i$.

**Algorithm 5** evaluateEstimation

**Require:** the map $\Gamma$ of data fields
**Require:** the map $C$ of collinear pairs; the key is a pair $p$ of agents, the value is the collinearity between the agents in $p$.
1: $est \leftarrow \frac{\sum_{p \in \text{keyset}(C)} \Gamma(p) \cdot C(p)}{\sum_{p \in \text{keyset}(C)} C(p)}$
2: addPerception$(est)$

The last step of the estimation procedure consists of comparing the estimate $est$ to the perception of the other agents. The more the estimate is close to the perception of an agent $\text{AG}_k$, the more this last is considered as pertinent for helping the $\text{ACA}_i$ in estimating missing information.

### Confidence Zone Update

Algorithm 6 updates the confidence zone of the $\text{ACA}_i$. This operation modifies the shape of the polygon representing the confidence zone according to the data fields associated with the pairs of agents previously assembled by the $\text{ACA}_i$. The idea behind the algorithm is to exclude progressively (reducing the polygon) the sensors that perceive values not coherent with the majority of the agents while trying to explore the environment (enlarging the polygon) to search for new agents capable of values useful for the estimation process. The $\text{ACA}_i$ updates the confidence zone by small steps at each iteration; therefore, the polygon is modified incrementally in time.

The algorithm takes as input a list $SP$ of pairs of ACAs and a map $\Gamma$ of data fields. The objective of this algorithm is to discriminate which sensors to exclude or include from the confidence zone. To do this, the algorithm calculates a threshold value using the calculated data fields in $\Gamma$.

In the first step, the algorithm sorts the list $SP$: the pairs of agents in $SP$ are sorted according to the distance between their data fields and the median data field $\widetilde{\gamma} \in \Gamma$ (line 5). Then, the algorithm calculates the value $\delta$ as the distance between the data fields of the first ($\Gamma(SP_1)$) and third ($\Gamma(SP_3)$) pairs of sensors in $SP$ (which has been previously sorted) (line 6).

At line 7 two threshold values $th^+$ and $th^-$ are calculated. The threshold values are calculated as the sum (for $th^+$) and difference (for $th^-$) of the first data field $\Gamma(SP_1)$ and the value $\delta$ multiplied by a constant $\omega$:

$$th^{\pm} \leftarrow \Gamma(SP_1) \pm \delta \times \omega \tag{5.16}$$

The constant factor $\omega$ is equal to $2.5$; we verified this value experimentally. The value of $\omega$ is chosen according to the application domain: if HybridIoT is used with outdoor sensing devices whose information may vary significantly, then $\omega$ can be increased accordingly. On the other hand, if HybridIoT is used with indoor sensing devices perceiving values that vary less significantly, it would be appropriate to reduce the parameter $\omega$ to avoid including irrelevant values for the

estimation. Although the parameter $\omega$ is fixed, we assume that a learning mechanism could be defined to allow the ACAs to determine autonomously this parameter. Nevertheless, this goes beyond the purposes of this thesis.

At line 8 the algorithm groups the pairs of sensors that provide data fields considered as outliers, that is, whose values are outside the range $[th^-, th^+]$. At line 9 the algorithm determines the set $K$ of pairs of sensors to be kept inside the confidence zone, thus providing pertinent data fields for estimating the missing information. The set K is obtained as the set difference between $SP$, containing all the pairs, and $O$, containing the pairs considered as outliers.

Finally, the algorithm respectively enlarges (line 10) the confidence zone towards the direction of the sensors in $K$ and reduces (line 11) the confidence zone to exclude the sensors in $O$.

Figure 5.17 illustrates the idea behind the presented algorithm: all the pairs of sensors for which perceptions are outside the range $[th^-, th^+]$ are considered as outliers.

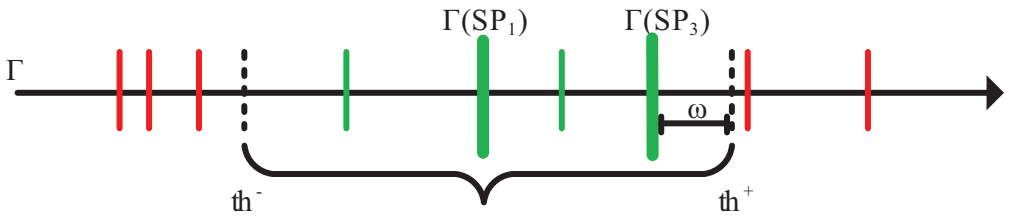



Figure 5.17: Evaluation of the pairs of agents to be excluded from the confidence zone of the $ACA_i$. The green values refer to the pairs to be kept inside the confidence zone, while the red values refer to the pairs to be excluded.

We consider the median data field as the reference data field for separating relevant and non-relevant pairs of sensors. We use the median value $\tilde{\gamma}$ as the reference data field as it is less affected by outlier values. Let us consider an arbitrary set containing a significant number of data fields; the median value represents the most relevant data field and the center of the distribution of the data field values. Therefore, the data field values that are not relevant have a significantly different value from the median data field.

**Scenario Example**

In the scenario illustrated in Figure 5.1 there are two rooms and a corridor including respectively two temperature sensors and a humidity sensor. These sensors are supposed to be fixed. We also consider that in the rooms there are several mobile devices able to perceive the temperature ($T_{4-7}$, $T_{8-9}$), and that the fixed sensors $S_1$, $S_2$ and $S_3$ are respectively associated with $ACA_1$, $ACA_2$ and $ACA_3$. The ACAs are initially associated with confidence zones of the same size as the environment in which the associated devices are located, as showed in Figure 5.18.

The solving process for estimating the temperature for $S_1$ and $S_2$ is roughly the following:

**Algorithm 6** updateConfidenceZone

**Require:** the list $SP$ of pairs of agents.
**Require:** the map $\Gamma$ of data fields for the pairs in $SP$
1: **if** $|SP| < 3$ **then**
2: **return**
3: **end if**
4: $\tilde{\gamma} \leftarrow$ median($\Gamma$)
5: sortBy($SP$, $\Gamma$, $\tilde{\gamma}$)
6: $\delta \leftarrow |\Gamma(SP_1) - \Gamma(SP_3)|$
7: $th^{\pm} \leftarrow \Gamma(SP_1) \pm \delta \times \omega$
8: $O \leftarrow \{p : p \in SP \wedge \Gamma(p) < \Gamma(SP_1) - th^{-} \vee \Gamma(p) > \Gamma(SP_1) + th^{+}\}$
9: $K \leftarrow SP \setminus O$
10: enlargeTowards($AG_k$), $\forall$ $AG_k \in K$
11: reduceFrom($AG_k$), $\forall$ $AG_k \in O$

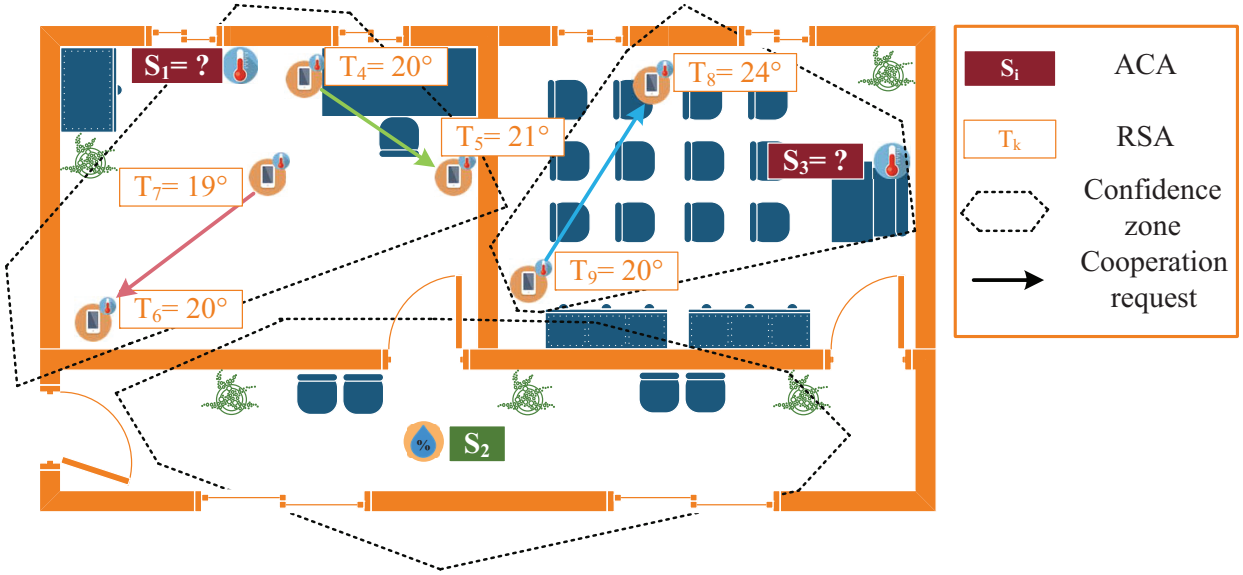



Figure 5.18: Case study illustration. The dashed zones represent the confidence zones of VSAs $S_1$, $S_2$ and $S_3$.

1. both $S_1$ and $S_2$ cannot perceive the environment because of an unexpected anomalous event. Therefore the associated $ACA_1$ and $ACA_2$ encounter an incompetence NCS.

2. the RSAs associated with sensors send their perceptions to the agent associated with the confidence zone they are in. Thus $T_4$ to $T_7$ send each one their perceptions to $S_1$, $T_8$ and $T_9$ to $S_3$.

3. The ACAs determine pairs of agents within their confidence zones. They first sort the agents according to increasing Euclidean distance, then each agent in order is coupled with the most aligned to the ACA that encountered a NCS. The pairs ($T_4$, $T_5$) and ($T_7$, $T_6$) are formed by $S_1$,

the pair ($T_9$, $T_8$) by $S_3$.

4. Each ACA evaluates the data field provided by the determined pairs of agents.

5. The data fields are used to estimate the missing information.

Let us now consider a numerical example using the temperature values shown in Figure 5.18 and suppose that the pairs of RSAs are already formed as in the same figure. The calculation of the data fields $\gamma^{(1)}_{6,7}$ and $\gamma^{(1)}_{4,5}$ for the pairs of RSAs $(T_6, T_7)$ and $(T_4, T_5)$ respectively, results in two values:

$$\gamma^{(1)}_{6,7} = T_6 + (T_7 - T_6) \cdot \frac{d(T_6, ACA_1)}{d(T_7, T_6)} = 18.66°C$$

for the pair $(T_6, T_7)$, and

$$\gamma^{(1)}_{4,5} = T_4 + (T_5 - T_4) \cdot \frac{d(T_4, ACA_1)}{d(T_5, T_4)} = 20.37°C$$

for the pair $(T_4, T_5)$. Once the data fields have been calculated, the estimate for the $ACA_1$ associated with $S_1$ is calculated as follows:

$$\frac{18.66 \cdot 380.23 + 20.37 \cdot 86.81}{380.23 + 86.81} = 18.97°C$$

The values $380.23$ and $86.81$ represent the collinearity between the agents. Finally, the confidence zone is updated by performing the thresholding technique described in Algorithm 6. Because the number of pairs in this example is $2$, the confidence zone is not updated.

The estimation of missing values through cooperation in the confidence zone is a complex resolution process due to the interdependence of the tasks performed by the ACAs. An ACA estimates missing information through cooperation with agents in the local environment where it is located (that is, the confidence zone): ACAs calculate the missing information as accurately as possible by differentiating between relevant and irrelevant information. This operation requires a continuous exploration of the environment in which the ACAs are situated. ACAs modify their confidence zones to exclude sensors that provide information not relevant to the calculation of estimates; contrarily, the confidence zones are enlarged towards the sensors that provide information relevant to the calculation of estimates: this enables the ACA to continuously explore the environment to determine the sensors that can contribute to the estimation process. Nevertheless, modifying the confidence zone requires the values perceived (or estimated) by the agents, either real or estimated. Therefore, the two tasks (estimation and confidence zone modification) contribute to the endogenous estimation goal, but cannot be separated.

The next section presents the exogenous estimation, used to estimate missing information by exploiting heterogeneous information provided by agents that are situated outside the confidence zones of ACAs.

### 5.9.3 Exogenous Estimation

The exogenous estimation enables addressing the lack of a sufficient number of sensing devices providing homogeneous information by integrating heterogeneous data for calculating the estimates for missing values. The idea behind the exogenous estimation is that the values perceived by different sensing devices can follow similar dynamics under the same environmental condition. The exogenous estimation enables ACAs to overcome the lack of homogeneous agents, that is, perceiving the same type of information. Therefore, thanks to exogenous estimation agents are capable of exploiting a large amount of information from heterogeneous sensing devices to estimate accurately the missing information [3].

The exogenous estimation method is divided into three steps:

1. **Evaluation of the set of cooperating agents**: the ACAs determine the set of agents with which cooperate to estimate missing information;
2. **Cooperative evaluation of the set of weights**: the ACAs cooperatively evaluate the weights to be used for estimating the missing information. A weight is a numerical quantity, calculated cooperatively by the agents, that is added to the last information perceived by the ACA that has encountered an incompetence NCS;
3. **Evaluation of the estimate**: the ACAs calculate the missing information by weighting the result of the estimate (calculated using the historical data) by the weights obtained cooperatively from the other ACAs.

Figure 5.19 shows the main step of the exogenous estimation performed by ACAs.

If the information at time $t$ is not available due to the unavailability of the sensing device, the $\text{ACA}_i$ that agentifies the sensing device encounters an incompetence NCS as it is not able to provide any information. The $\text{ACA}_i$ pursues the exogenous estimation if (*i*) there is no agent inside the confidence zone of $\text{ACA}_i$ and (*ii*) some agents perceiving heterogeneous information are available beyond the confidence zone of the $\text{ACA}_i$. The $\text{ACA}_i$ solves the incompetence NCS by calculating an estimate using the historical data of $\text{ACA}_i$ (step ❶), then cooperating with agents (if any) that perceive heterogeneous information and are situated beyond the confidence zone of $\text{ACA}_i$ (step ❷).

The exogenous estimation of the missing information considers the last perceived information at time $t-1$, weighted by a value $w_t$ obtained from a cooperative process among ACAs. The value $w_t$ is added to the information perceived at time $t-1$ because we consider that consecutive environmental information does not vary significantly unless there is noise or unpredictable environmental factors. The weight $w_t$, therefore, represents the variation that the estimated information at time $t$ assumes with respect to the last information perceived at time $t-1$.

Figure 5.20 shows an example of how cooperation takes place between ACA ① and other ACAs perceiving different types of information. The ACA ① has encountered an incompetence NCS as it is not able to provide information. The ACA ① performs an endogenous estimation by cooperating

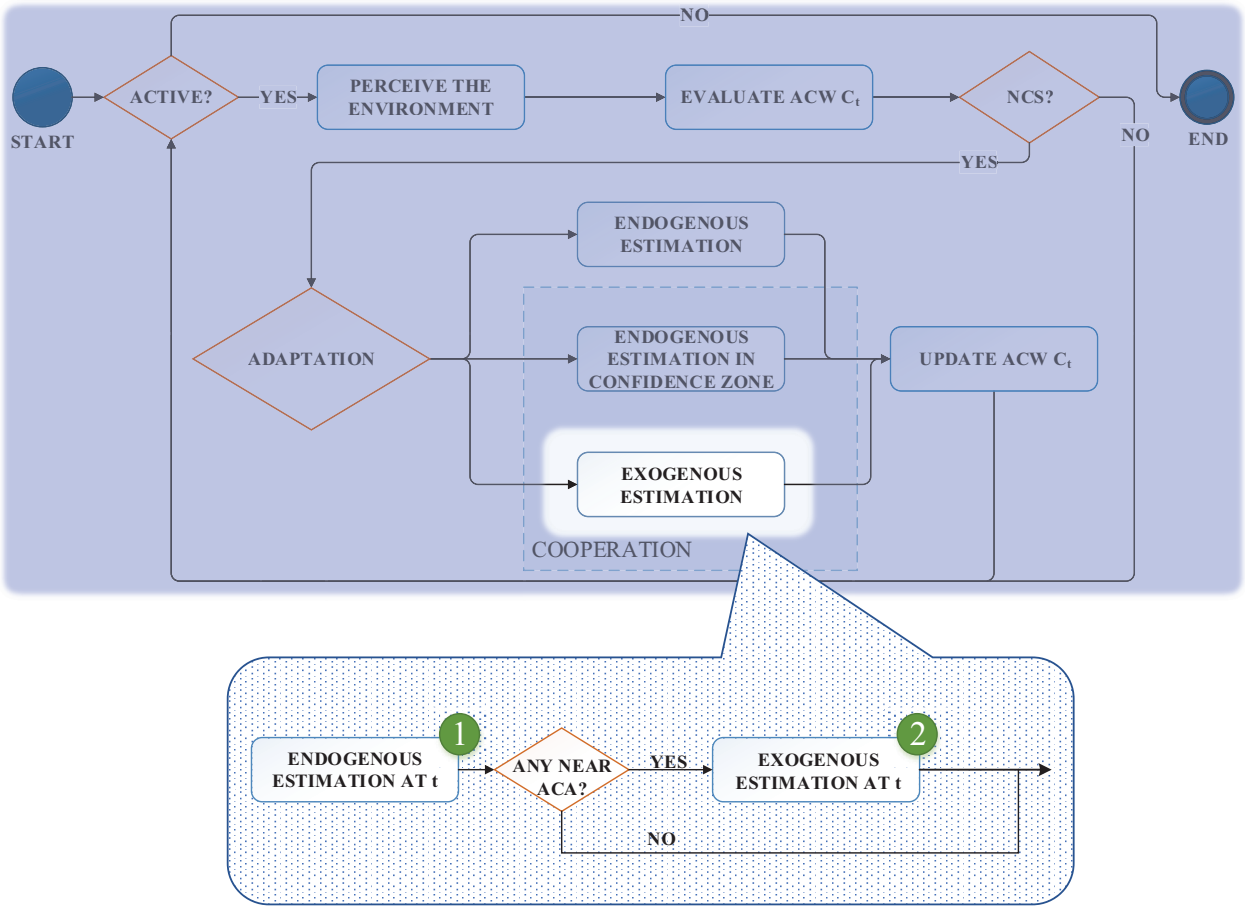



Figure 5.19: Main steps of the exogenous estimation process performed by ACAs.

with ACAs ② and ③ (step ❶). Following, an exogenous estimation is performed by cooperating with ACAs ④, ⑤, ⑥ and ⑦ (step ❷), that perceive information of different types with respect to ACA ①. Once the ACA ① obtained the weights through cooperation, some ACAs can encounter a uselessness NCS as their cooperation is not helpful to provide an estimate. This is because the ACA ① discriminates the obtained weights to be used in order to calculate an accurate estimate.

#### Evaluation of the Set of Cooperating Agents

The exogenous estimation is based on a cooperative process between different ACAs. However, a large number of agents can be present in the environment, some of which are not relevant for the calculation of the estimate. The $ACA_i$ uses one of the following criteria to choose the ACAs to cooperate with:

- **Nearest ACAs**: the $ACA_i$ cooperates with the nearest ones in the environment (the Euclidean distance is used). The Nearest ACAs criterion enables to choose ACAs that are in the immediate proximity of the $ACA_i$ that has encountered a NCS. The use of this criterion can be very

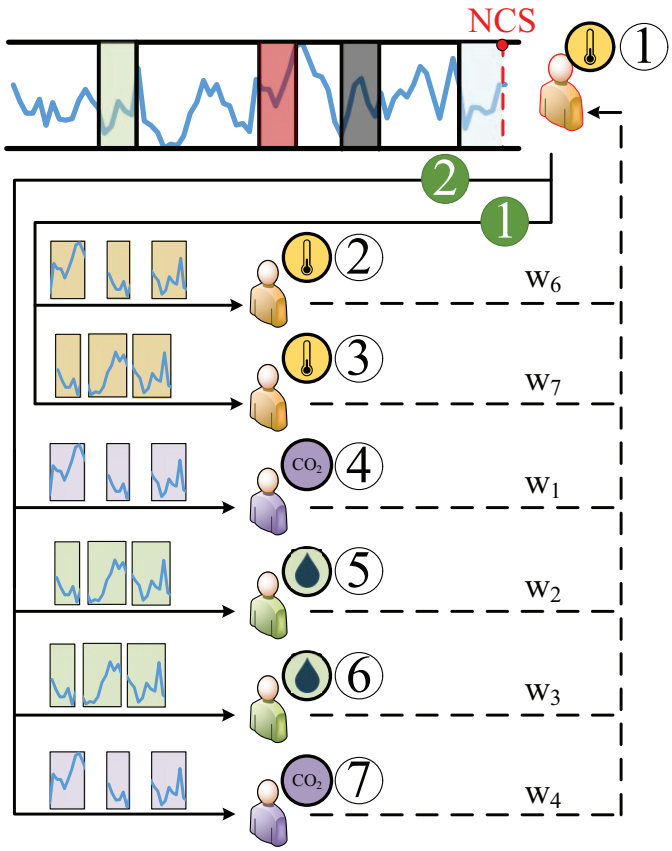



Figure 5.20: ACA ① cooperates with other agents by performing both endogenous and exogenous estimations. The weights obtained cooperatively are used to provide an estimate of the missing value. Steps ❶ and ❷ are the same as in Figure 5.19.

efficient in open environments where no barriers are present (such as weather stations). For example, this criterion enables to choose ACAs both inside and outside a building because of their proximity, even if they could perceive information that differs significantly.

- **Most confident ACAs**: the $\text{ACA}_i$ cooperates with the agents that have the highest degree of confidence. The most confident criterion enables selecting a set of agents that do not necessarily have to be in the immediate proximity: agents are chosen according to the similarity between the perceived information.

### Cooperative Evaluation of the Set of Weights

Let $t$ be the time instant at which the $\text{ACA}_i$, associated with the $i$-th sensing device, has to estimate the information, and $\Upsilon^{(t)}$ be the set of agents the $\text{ACA}_i$ cooperates with, evaluated according to one of the previous criteria.

Let $\xi^{(t)}$ be the set of ACWs chosen by the $\text{ACA}_i$ to estimate the missing information at time $t$, $\Sigma^{(t)} = \{\sigma_1, \sigma_2, ..., \sigma_k\}$ with $k = |\xi^{(t)}|$ the indexes of the ACWs chosen by $\text{ACA}_i$, where $\sigma_i \in \Sigma^{(t)}$ is

the index of the ${\sigma_i}^{th}$ ACW in $\xi^{(t)}$. In the exogenous estimation process, the set $\Sigma^{(t)}$ is communicated to the other agents in $\Upsilon^{(t)}$. In this way, the ACA$_i$ provides each agent in $\Upsilon^{(t)}$ with an indication of which are the temporal instants associated with ACWs that are similar to the ACW $C_t$, that is, the ACW containing the information to estimate at time $t$. Each agent in $\Upsilon^{(t)}$, therefore, evaluates the distance between the ACW observed at the time instant $t$ and the ACWs whose indices are indicated by the set $\Sigma^{(t)}$. Each agent calculates a weight by using the Equation (5.9).

The cooperation between agents, therefore, yields a set $W$ of weights, one for each agent in $\Upsilon^{(t)}$. The weights in $W$ are calculated by agents that perceive both homogeneous and heterogeneous information.

**Calculation of the Estimate**

Once the ACA$_i$ obtains a set $W$ of weights through cooperation with the other agents, it evaluates one estimate for each weight $w \in W$, resulting in a set of estimates. To avoid taking into account information that is not relevant to the output estimate, the ACA$_i$ evaluates a histogram $H$ obtained from an empirical cumulative distribution of the estimates. The histogram $H$ includes the estimates obtained from both endogenous and exogenous estimations. The ACA$_i$ selects the bin which average value is closest to the estimate obtained by the endogenous process, then it calculates the average value of the elements within the bin. The result of the average operation is returned as the estimate for the missing information at time $t$.

Let $E_{end}$ be the estimate obtained by ACA$_i$ through the endogenous estimation process (calculated using Equation (5.10)). Let $f$ be a function that returns the mean of the values contained in a bin $H_k$ of the histogram $H$ such that the difference between $H_k$ and the endogenous estimate $E_{end}$ is minimized:

$$f(H_k) = |\overline{H_k} - E_{end}| \tag{5.17}$$

where $\overline{H_k}$ is the average of the values in the $k$-th bin of the histogram of frequencies $H$. The ACA$_i$ evaluates the estimate $E_{exo}$ as follow:

$$E_{exo} = \frac{\text{argmin}_k\, f(H_k) + E_{end}}{2}. \tag{5.18}$$

The average operator in (5.18) enables to weigh equally the results obtained from endogenous and heterogeneous estimations.

Figure 5.21 resumes the cooperative estimation between ACA$_i$ and the other agents. For the sake of simplicity, we show one lane for all the agents with which the ACA$_i$ cooperates, as the cooperative behavior is identical among the agents.

### Scenario Example

In the scenario illustrated in Figure 5.1 there are two rooms and a corridor including respectively two temperature sensors and a humidity sensor. These sensors are supposed to be fixed.

In the first scenario described in Section 5.9.1, we supposed that the available sensing devices are capable of calculating estimates, through their associated ACAs, by using the information perceived previously. In the scenario described in Section 5.9.2, we used the information perceived by the sensing devices within the confidence zone of each ACA that encounters an incompetence NCS. In both cases, ACAs perform an endogenous estimation as we supposed that agents perceive the same type of information.

In a real context, the available sensing devices can perceive heterogeneous information. Moreover, the sensing devices can perceive information of the same type but using different units and scales. The HybridIoT system allows to exploit a multitude of sensing devices in the environment independently of the type of information perceived and factors such as unit or scale.

Consider the scenario in Figure 5.22 (previously described in Section 5.9.2). Here we suppose that two sensors face an unpredictable event that avoid them to provide the information. In this

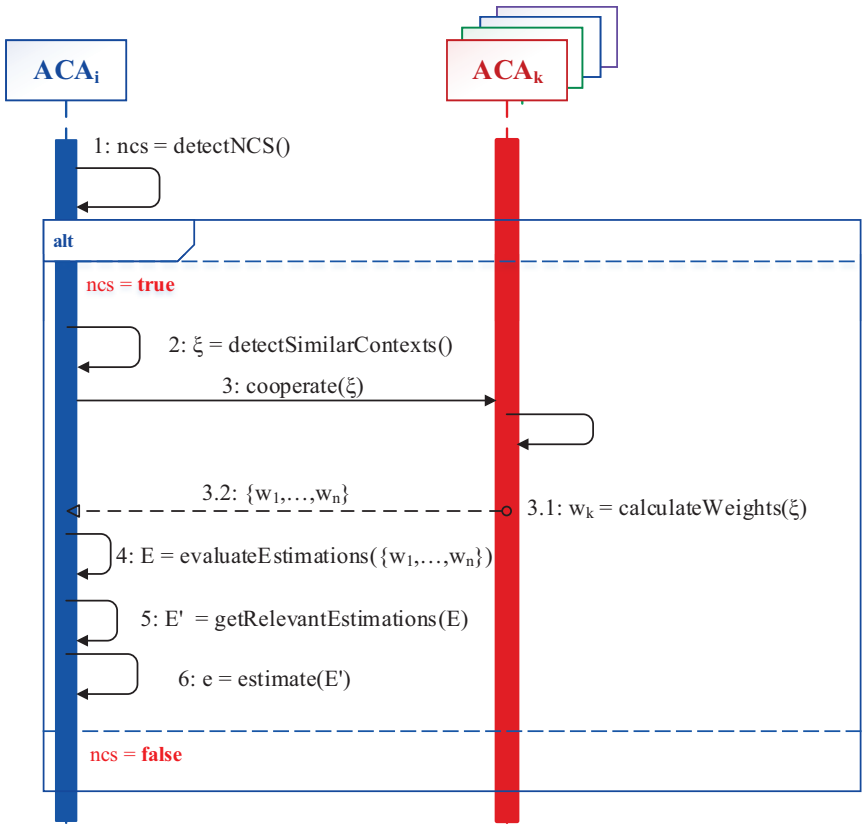



Figure 5.21: Steps of the cooperation between the ACA that detected a NCS and the subset of neighbors agents with which the ACA cooperates.

case, the ACAs associated with the sensing devices encounter an incompetence NCS that is solved by a cooperative process between agents of the same type within the confidence zones of agents. However, if $S_2$ encounters an incompetence NCS it is not able to provide any estimation for the following reasons:

- there is no sensor of the same type as $S_2$, that is, perceiving information of the same type,
- there is no sensor within the confidence zone of $S_2$.

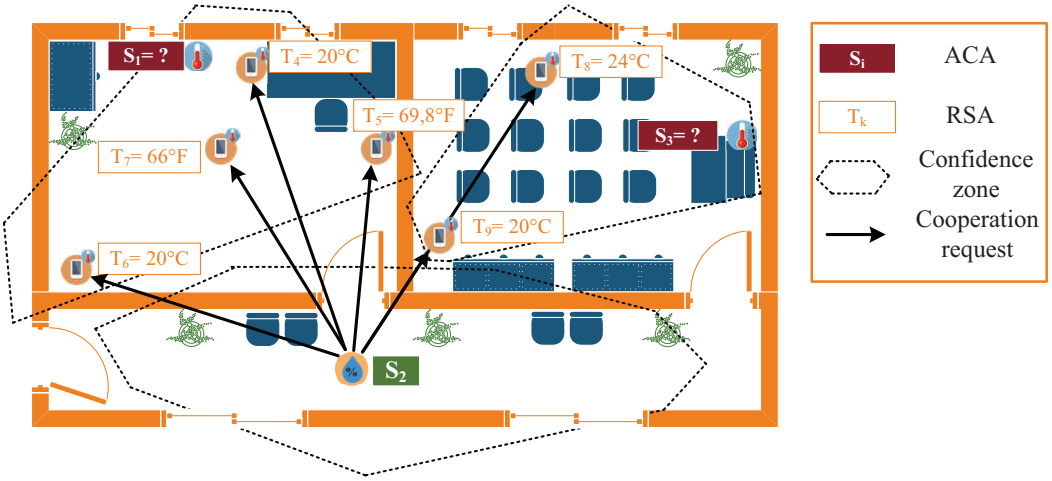



Figure 5.22: Case study illustration. The sensors $S_2$ is unable to provide any information, thus it has to cooperate with the other agents.

In this scenario, we suppose that $S_2$ has to estimate the missing information by means of exogenous estimation thanks to a cooperative resolution process between agents situated outside the confidence zone of $S_2$. Because no agents are within the confidence zone of $S_2$, the ACA that agentifies $S_2$ cooperates with the other agents, independently from the type of the perceived information (as depicted by the arrows in Figure 5.22). Moreover, in this scenario, we suppose that the sensing devices perceive not only different information (temperature and humidity) but also the same type of information using different units (Fahrenheit and Celsius degree).

To estimate the missing value at time $t$, the $ACA_2$ (related to $S_2$) searches in its history for ACWs that are similar to the last observed $ACW_t$ containing the information to be estimated (Table 5.2).

Following, $ACA_2$ cooperates with the other ACAs to estimate the missing value. The other sensing devices perceive temperature information in Fahrenheit and Celsius degree. Although the information perceived by these devices cannot be used directly to estimate a humidity value for $ACA_2$, they act cooperatively to allow $ACA_2$ to estimate the missing information.

The cooperative estimation process starts from the $ACA_2$, which encountered an incompetence NCS at time $t$. The $ACA_2$ evaluates the agents to cooperate with according to one of the criteria described in Section 5.9.3 (nearest or most confident agents). In this scenario, we assume that all the

Table 5.2: Most similar ACWs to $ACW_t$ of $ACA_2$, which has to estimate the information at time $t+1$. The $t$-th value is on the right side of the table.

| | | $\mathbf{S_2}$ | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| % | $\mathbf{ACW_{18}}$ | 64 | 69 | 66 | 88 | 88 | 76 | 65 | 57 |
| | $\mathbf{ACW_{26}}$ | 67 | 65 | 70 | 88 | 86 | 69 | 67 | 62 |
| | $\mathbf{ACW_{33}}$ | 64 | 71 | 70 | 88 | 86 | 75 | 65 | 60 |
| | $\mathbf{ACW_{59}}$ | 67 | 68 | 71 | 93 | 88 | 71 | 65 | 62 |
| | $\mathbf{ACW_t}$ | 64 | 68 | 63 | 86 | 82 | 71 | 65 | 61 |

agents are involved in the cooperative estimation process. The $ACA_2$ begins by evaluating the most similar context windows to $ACW_t$ as described in the endogenous estimation process. Let $\xi$ be the set of ACWs evaluated by the $ACA_2$. The set $\xi$ is then sent to the other agents: each agent involved in the cooperative resolution process compares its own ACW at time $t$ with the ACWs present in its history, associated with the time instants that identify the ACWs present in set $\xi$. The cooperative process results in a set $W$ of weights that the $ACA_2$ uses to calculate different estimates, one for each weight $w \in W$ calculated by the cooperating agents. Each estimate is calculated as the sum of a weight $w \in W$ and the information at time $t-1$ perceived by the $ACA_2$. This process results in a set $E$ of estimates calculated using the weights provided by the agents. The $ACA_2$ then selects the most relevant estimate using a frequency histogram of the estimates in $E$. The final estimate is calculated using the Equation (5.18).

Let us consider the $ACA_2$ to be in an incompetence NCS, thus it has to estimate the information at time $t+1$. Table 5.2 lists the $ACW_t$ and the most similar ACWs at times $(18, 26, 42, 59)$. These ACWs are chosen to minimize the distance $d$ (Equation (5.8)) with the $ACW_t$. The $ACA_2$ calculates an endogenous estimation at first, as described in Section 5.9.1. The result of the endogenous estimation is $55,85\%$. Then, the $ACA_2$ cooperates with the other agents available in the environment by providing them with the set $\xi = \{ACW_{18}, ACW_{26}, ACW_{33}, ACW_{59}\}$. In turns, the other agents evaluates their ACWs at the time instants $18$, $26$, $33$ and $59$ as indicated by $\xi$. Table 5.3 reports the ACWs returned by other agents.

Table 5.3: The ACWs evaluated by the cooperating agents at the time instants indicated by the ACWs in the set $\xi$. T4, T6, T8 and T9 perceive temperature values in degree Celsius, while T5 and T7 perceive temperature values in degree Fahrenheit. The $t$-th value is on the right side of the table.

| | | **T4** | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| °C | $\mathbf{ACW_{18}}$ | 21,66 | 20,28 | 20,58 | 24,94 | 24,17 | 21,90 | 25,70 | 20,21 |
| | $\mathbf{ACW_{26}}$ | 22,63 | 22,29 | 24,59 | 24,77 | 21,12 | 22,94 | 22,67 | 23,88 |
| | $\mathbf{ACW_{33}}$ | 24,26 | 24,53 | 21,66 | 24,08 | 23,93 | 20,98 | 20,71 | 22,99 |
| | $\mathbf{ACW_{59}}$ | 25,76 | 22,04 | 23,51 | 21,34 | 24,51 | 21,53 | 23,04 | 24,19 |
| | $\mathbf{ACW_t}$ | 23,19 | 24,67 | 20,92 | 22,74 | 23,83 | 21,73 | 21,53 | 22,07 |
| | | **T6** | | | | | | | |
| | $\mathbf{ACW_{18}}$ | 25,35 | 25,76 | 23,28 | 20,83 | 20,90 | 21,55 | 25,04 | 21,53 |

°C

| | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| | $\mathbf{ACW}_{26}$ | 24,89 | 21,46 | 25,58 | 22,10 | 21,18 | 21,51 | 23,70 | 22,84 |
| | $\mathbf{ACW}_{33}$ | 22,11 | 24,98 | 23,51 | 23,30 | 25,50 | 21,72 | 24,54 | 24,52 |
| | $\mathbf{ACW}_{59}$ | 22,28 | 23,41 | 20,46 | 20,32 | 23,18 | 24,68 | 25,60 | 20,78 |
| | $\mathbf{ACW}_{t}$ | 21,95 | 22,54 | 21,69 | 25,25 | 25,75 | 24,03 | 21,34 | 24,68 |
| | | **T8** | | | | | | | |
| °C | $\mathbf{ACW}_{18}$ | 23,41 | 22,82 | 20,07 | 22,02 | 20,97 | 24,77 | 21,87 | 23,17 |
| | $\mathbf{ACW}_{26}$ | 20,99 | 23,61 | 21,58 | 23,92 | 24,14 | 24,49 | 22,70 | 20,50 |
| | $\mathbf{ACW}_{33}$ | 21,37 | 25,48 | 20,91 | 24,95 | 23,23 | 25,98 | 20,47 | 22,66 |
| | $\mathbf{ACW}_{59}$ | 20,64 | 25,77 | 20,03 | 24,65 | 24,90 | 25,21 | 20,51 | 22,40 |
| | $\mathbf{ACW}_{t}$ | 20,63 | 20,54 | 22,64 | 23,11 | 21,44 | 24,17 | 24,01 | 24,05 |
| | | **T9** | | | | | | | |
| °C | $\mathbf{ACW}_{18}$ | 21,56 | 24,80 | 22,59 | 25,46 | 21,09 | 21,58 | 20,87 | 20,82 |
| | $\mathbf{ACW}_{26}$ | 25,22 | 23,48 | 23,30 | 20,87 | 25,12 | 23,73 | 22,11 | 23,08 |
| | $\mathbf{ACW}_{33}$ | 22,41 | 20,46 | 21,44 | 20,74 | 21,10 | 21,44 | 22,50 | 20,30 |
| | $\mathbf{ACW}_{59}$ | 25,42 | 25,67 | 22,95 | 22,94 | 22,03 | 25,40 | 22,22 | 20,67 |
| | $\mathbf{ACW}_{t}$ | 22,05 | 23,64 | 21,15 | 24,43 | 21,46 | 25,50 | 21,61 | 24,59 |
| | | **T5** | | | | | | | |
| °F | $\mathbf{ACW}_{18}$ | 77,77 | 78,58 | 72,74 | 69,20 | 70,79 | 72,41 | 74,42 | 70,83 |
| | $\mathbf{ACW}_{26}$ | 74,51 | 75,68 | 70,39 | 69,27 | 71,20 | 71,44 | 72,58 | 73,48 |
| | $\mathbf{ACW}_{33}$ | 68,92 | 70,83 | 76,65 | 68,32 | 78,03 | 75,89 | 73,28 | 74,25 |
| | $\mathbf{ACW}_{59}$ | 70,56 | 72,96 | 78,40 | 73,91 | 73,63 | 70,50 | 73,28 | 74,74 |
| | $\mathbf{ACW}_{t}$ | 74,50 | 77,89 | 72,99 | 72,98 | 71,48 | 73,09 | 69,90 | 73,11 |
| | | **T7** | | | | | | | |
| °F | $\mathbf{ACW}_{18}$ | 75,33 | 72,27 | 71,97 | 78,67 | 68,41 | 77,56 | 77,86 | 76,60 |
| | $\mathbf{ACW}_{26}$ | 69,07 | 70,83 | 71,62 | 75,34 | 69,47 | 75,79 | 69,15 | 75,06 |
| | $\mathbf{ACW}_{33}$ | 73,34 | 76,41 | 75,72 | 77,76 | 77,62 | 71,61 | 75,55 | 70,14 |
| | $\mathbf{ACW}_{59}$ | 68,33 | 76,04 | 73,40 | 73,18 | 77,77 | 74,59 | 74,67 | 77,28 |
| | $\mathbf{ACW}_{t}$ | 72,18 | 68,01 | 72,58 | 76,32 | 76,48 | 68,39 | 75,79 | 69,65 |

Once evaluated the ACWs at the time instants indicated by the context windows in $\xi$, each cooperating agent evaluates a weight $w$ as indicated by Equation (5.9). Then, the weights are used by $ACA_2$ to calculate one estimate for each weight. Table 5.4 lists the weights evaluated by the cooperating agents and the estimate calculated by $ACA_2$ using the weights. The error column refers to the absolute difference between the estimate calculated by each cooperating agent and the estimate calculated by $ACA_2$ through the endogenous estimation process.

The $ACA_2$ Finally chooses the estimates provided by T7 as the most confident, as it provides an estimate that is the closest to the value obtained through the endogenous estimation process. Finally, $ACA_2$ evaluates the exogenous estimation as in Equation (5.18):

Table 5.4: Weights calculated by cooperating agents. Column $E$ contains the estimation calculated by the $ACA_2$ by using the weights obtained cooperatively with other agents. The error column refers to the absolute difference between the estimation in $E$ and the estimate obtained by the endogenous estimation process of $ACA_2$.

| | | Weights | E | Absolute Error |
|---|---|---|---|---|
| % | **T4** | -0,69 | 61,31 | 5,45 |
| | **T6** | 2,51 | 64,51 | 8,65 |
| | **T8** | 1,82 | 63,82 | 7,96 |
| | **T9** | 3,34 | 65,34 | 9,48 |
| | **T5** | -0,54 | 61,46 | 5,60 |
| | **T7** | -5,68 | 56,32 | 0,46 |

$$E_{exo} = \frac{56,32 + 55,85}{2} = 56,08\% \tag{5.19}$$

where $56,32$ is the result of the exogenous estimation, $55,85\%$ is the result of the endogenous estimation calculated by $ACA_2$.

## 5.10 Dynamic Ambient Context Window Evaluation

This section describes how agents determine ACWs with a variable number of entries, which has an impact on the calculation of estimates [1].

HybridIoT determines ACWs of variable size to capture data fluctuations and automatically determine ACW without specifying either their length or the frequency of data acquisition from sensing devices. This allows the system to integrate sensing devices that have different acquisition frequencies without any configuration.

Consider two temperature datasets, one containing daily data and the other data perceived every $30$ seconds. In the first case, data perceived daily could vary significantly from sample to sample, so it is necessary that the ACWs have a limited size so that the variance does not assume high values. In this case, windows containing a considerable amount of information would not identify the information associated with a precise time instant due to the high number of fluctuations in the data. On the contrary, in the second case, it is necessary to have ACWs containing a significant amount of information: the information does not vary significantly from sample to sample, so the variance would be too low if an ACW contains few samples. Therefore, in both cases, it would be necessary to specify a size for ACWs that reflect the information and the acquisition frequency of the sensing device. In a large-scale environment such as a city, using an open system such as HybridIoT where an unspecified number of sensing devices can enter or leave the system at any time, **it is not possible to specify an ACW size** for each ACA associated with a sensing device. It is therefore necessary for the ACAs to determine autonomously the size of each ACW depending on the information perceived.

To explain the calculation of dynamic size ACWs, consider two examples ACWs referring to the same time instant $t$, but of different sizes: $C_{t4}$, $C_{t7}$, containing respectively $4$ and $7$ entries in the time interval $[t-4, t]$ and $[t-7, t]$. Here we suppose that the ACA has previously collected (through a sensing device) enough information to calculate both ACWs. The agent evaluates the estimate of information at $t$ by simulating its absence using both $C_{t4}$, $C_{t7}$. This results in two different estimates that are compared to the value that is perceived by the sensing device. The ACW minimizing the bias, that is the discrepancy between the estimate and the real value, is defined as **representative of the information at** $t$.

In HybridIoT, the ACAs calculate up to $15$ different ACWs for specific information. Let $\xi_t$ be the set of $15$ ACWs determined by an arbitrary ACA. Each ACW $C \in \xi_t$ is compared with those of the same ACA and with those of the other agents. Since an ACW of size $\ell$ can be compared with a set $\xi$ containing only ACWs of size $\ell$, therefore each ACW $C \in \xi_t$ can be compared with separate sets of ACWs, each one containing different ACWs of different sizes. For example, let $C^{(\ell)} \in \xi_t$ be an ACW of size $\ell$, then $C^{(\ell)}$ is compared to a set $\xi_t^{(\ell)}$ containing only ACWs of size $\ell$. After a comparison with different sets, the ACA chooses the one that minimizes the bias: in other words, the agents choose, among the ACWs already present in the history, those presenting similar dynamics. This allows for obtaining accurate estimates.

After experimentations, we observed that the amount of information depends on the data acquisition frequency of the sensor. For example, if a sensing device perceives information at intervals of 10 seconds, it is necessary to have a sufficient amount of information as the variability between consecutive information is low. Contrarily, if a sensing device perceives information daily, the variability between samples consecutive in time can be high; therefore, it is necessary to use a small number of values to assemble an ACW. ACAs capture the variability between consecutive values to assemble ACWs that are representative of the information without any configuration.

Figure 5.23 shows an example of how dynamic size ACWs are calculated by ACAs. Starting from the information at time instant $t$, the ACA evaluates a set $\xi_t$ containing $6$ ACWs that refer to the same information at time $t$ (the blue cases). Once the set $\xi_t$ has been calculated, each ACW $C \in \xi_t$ is used to estimate the information at time $t+1$. The information perceived by the real sensor at time $t+1$ is then compared with the estimates provided by each ACW $C \in \xi_t$. The error is then calculated as the absolute difference between the estimates and the real value. The ACA, finally, chooses the ACW that minimizes the error between the estimate and the real value.

## 5.11 Conclusion

This chapter presented HybridIoT, a MAS based solution to estimate missing heterogeneous information addressing the properties of openness and large-scale computation. Using HybridIoT it is possible to avoid the installation of a large number of sensing devices by using virtual devices able to provide accurate information estimates where *ad hoc* sensing devices are not available. This leads

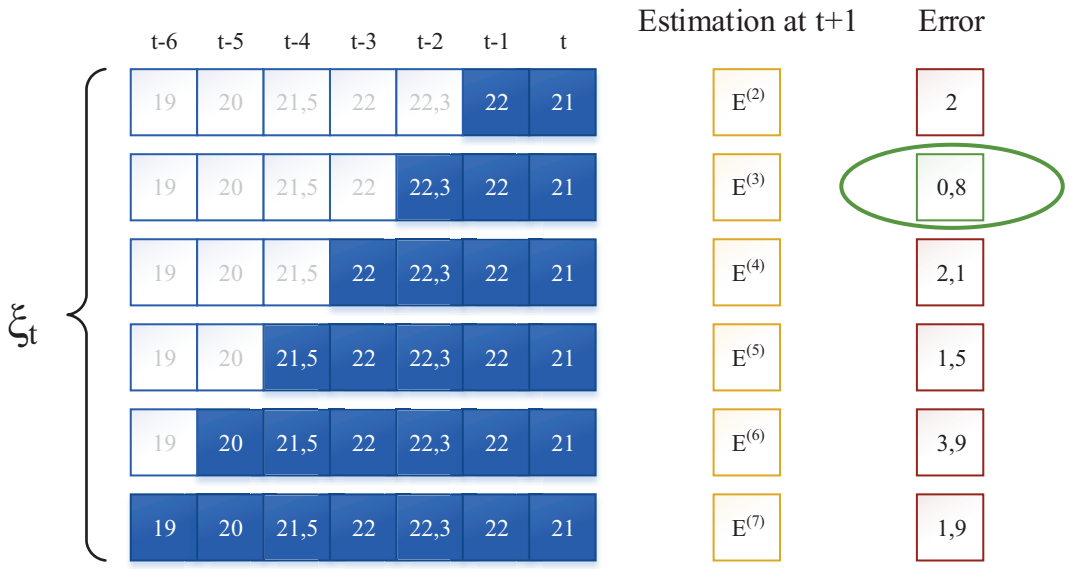



Figure 5.23: The ACW of size 3 is chosen as representative of the information at time $t$ as it enables providing an estimate $E^{(3)}$ at $t+1$ whose difference from the real value (0.8) is minimized.

to relevant cost savings due to the low number of sensors to be installed and the management costs related to the sensing devices network. In this way, it is possible to reach a significant compromise between an extensive instrumentation of cities and the availability of information at any time and everywhere.

The advantages of HybridIoT over the state of the art are heterogeneity, openness and scalability. Heterogeneity enables to integrate different types of devices that could perceive different types of information. Openness enables sensing devices to enter or leave the system without the need for any re-configuration. Cooperation enables agents to exploit the data acquired from a multitude of devices to estimate missing information.

HybridIoT is generic as it does not require any input parameters or configuration. The system can operate in open, dynamic environments such as cities, where devices can appear or disappear without any prior notification. Agents are able to provide estimates in almost instantaneous time. The state of the art techniques used to compare the proposed approach are among the most common and powerful methods to carry out regression over time series. However, these techniques do not appear to be able to operate in open environments where devices may appear or disappear unexpectedly and do not make use of heterogeneous information which type is not known *a priori*.

# Part III

# Evaluation

Chapter 6

# Experimental Results

Objectives of this chapter:

- Describing the real dataset used for the experiments
- Describing the evaluation methods and the state of the art solution used to compare the proposed approach
- Discussing the results obtained by HybridIoT

AFTER discussing HybridIoT and the estimation technique, this chapter presents the validation of HybridIoT on a real weather dataset. Figure 6.1 shows the order in which the experiments will be presented in this chapter. Section 6.2 presents a pipeline of standard techniques for estimating missing values on a large scale. The results obtained from this pipeline will be analyzed and used to compare the results obtained by HybridIoT. Afterwards, the results obtained by HybridIoT will be analyzed on the presented dataset. The results obtained from endogenous estimation will be compared with those obtained by standard techniques available in the KNIME platform and with the results obtained from the pipeline of standard techniques. Finally, section 6.5 presents the results obtained from exogenous estimation, that is, obtained from the integration of heterogeneous information. The results of the exogenous estimation will be analyzed and compared with those obtained from the endogenous estimation in order to show the validity of the proposed method and the benefits compared to endogenous estimation.

The following section 6.1 presents the real dataset used to evaluate the proposed approach. We describe the type of information, its size, the frequency of acquisition and where data have been acquired; the $k$-fold validation technique is therefore described. This evaluation technique is used to validate the obtained results. Then, we provide some details of the machine used to validate the proposed solution.

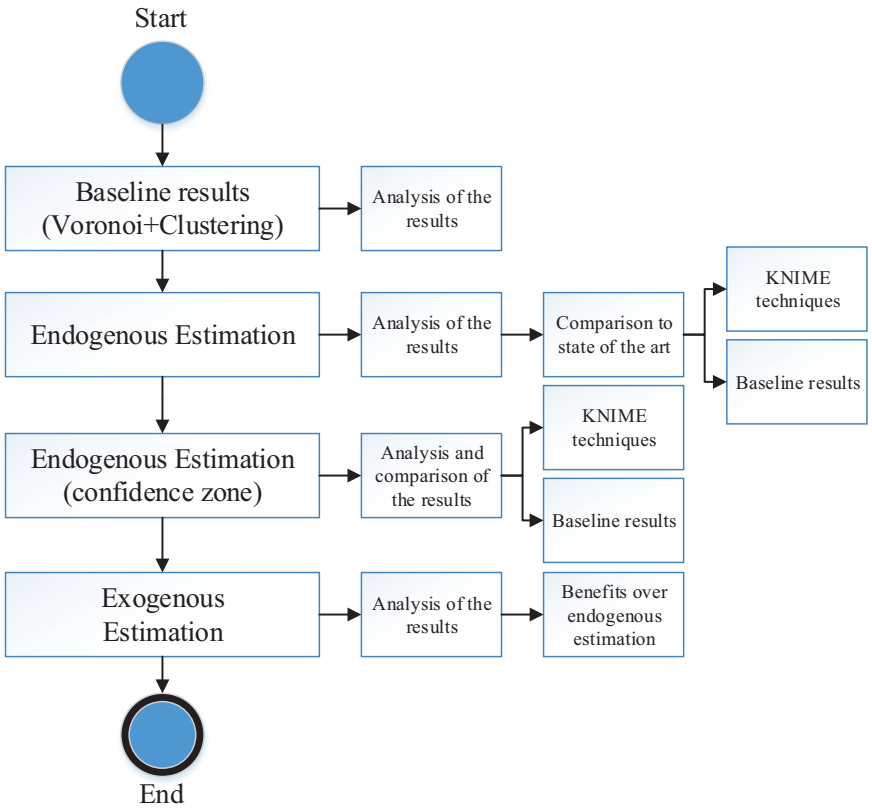



Figure 6.1: The order according to which the experiments are presented in this chapter.

## 6.1 Dataset

The dataset used to validate the proposed method contains real environmental information acquired in the Emilia Romagna region in Italy [4]. In Emilia Romagna region, the prevailing climate is temperate subcontinental, with hot and humid summers followed by cold and harsh winters. Orographically the region is divided almost symmetrically between the Po Valley and the hills and mountains of the northern Apennines.

We considered the average daily air temperatures at 2 meters of altitude, the average daily solar irradiance, the average daily wind speed at 10 meters of altitude and the average daily air humidity at 2 meters of altitude. Data have been collected in 196 days from September 8 2017 to April 25 2018.The days when some stations were not operational are not considered. We used two versions of this dataset for the experimentations: (*i*) a version containing all the types of information previously listed, that consists in an array of 8112 numerical values acquired from 9 weather stations; (*ii*) a version containing only temperature values, which results in an array of 15680 numerical values acquired from 80 weather stations. This difference in terms of the number of samples and stations, between the two datasets versions, is because for different days not all the information has been acquired correctly by each station.

Figure 6.2 shows the distribution of mean temperature values for the first version of the dataset,

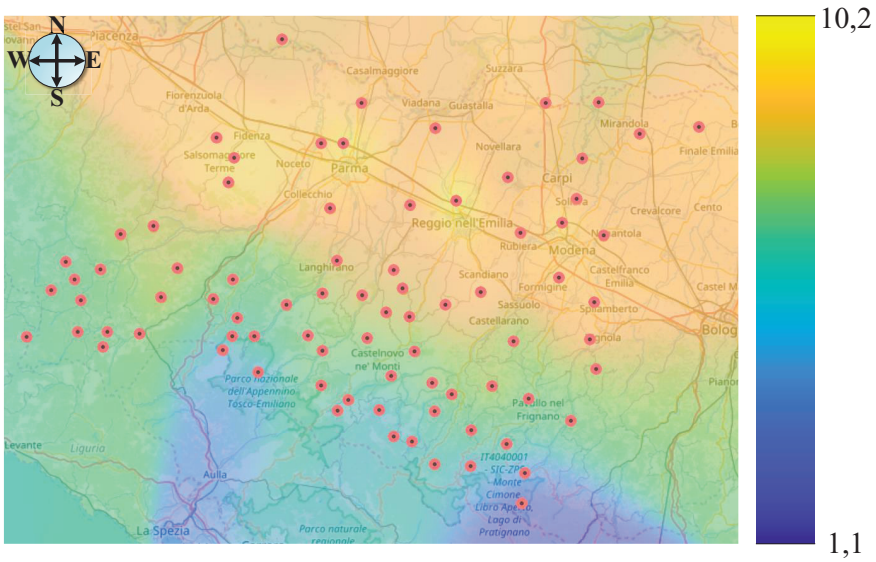



Figure 6.2: Distribution of the temperature values (degree Celsius) on the Emilia-Romagna region (obtained through OpenStreetMap [102]) obtained from the dataset of 80 stations. The $x$-axis reports the longitude, the $y$-axis the latitude.

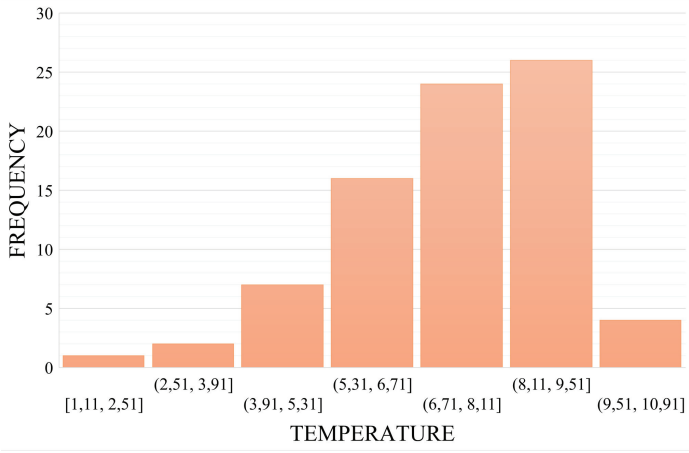
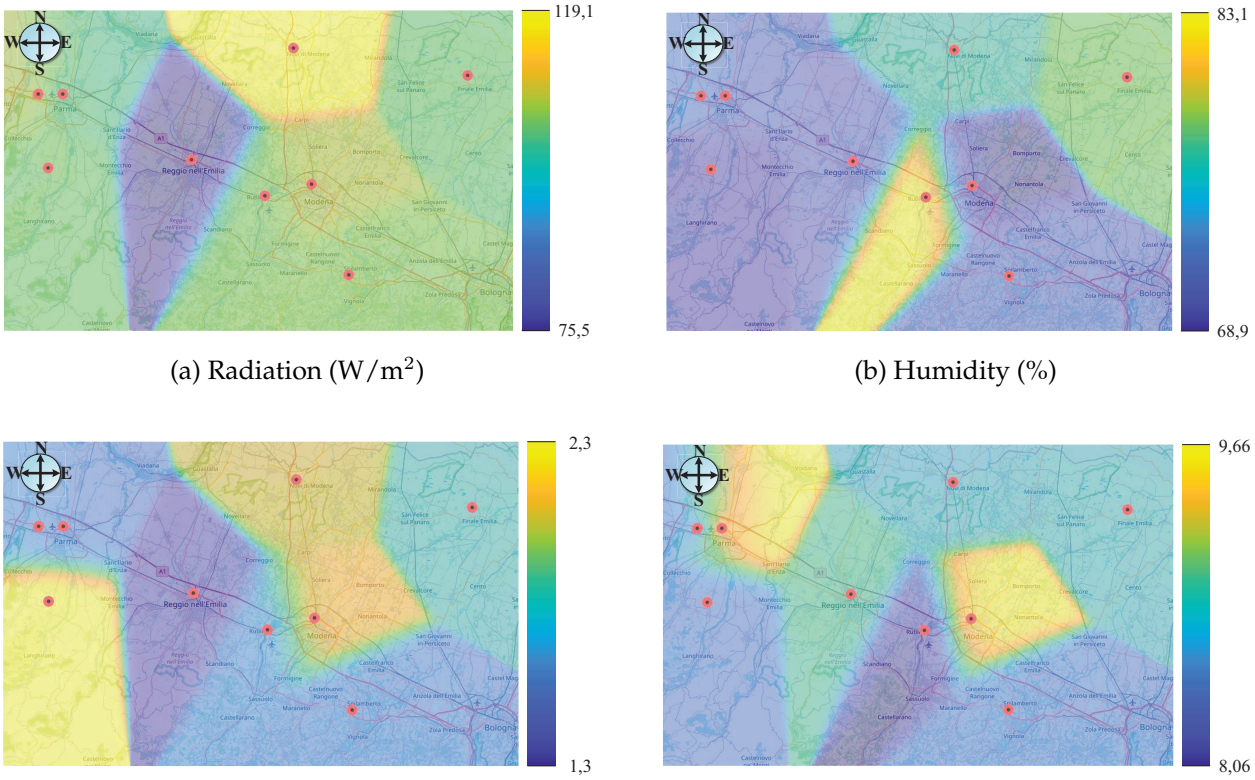



Figure 6.3: Histogram of frequencies for the temperature values from the dataset of 80 weather stations.

obtained through a standard normalized convolution, a non-direct methodology widely used for filtering incomplete or uncertain data which is based on the separation of both data and operator into a signal part and a certainty part [4]. The figure shows that temperatures are uniformly distributed over the territory, in urban areas (center, north-west) temperatures are more intense than in the rest of the region, especially in the south-east part, closer to the Mediterranean Sea.

Figure 6.3 shows the histogram of temperature value frequencies for the first version of the dataset, containing information acquired by the 80 stations. The histogram shows that on average the temperature perceived by the stations in the region is between 6.72°C and 9.51°C.

Figure 6.4 shows the distribution of radiation, relative humidity, wind speed, and temperature for the dataset obtained from 9 weather stations. Here can be observed that the wind speed is more intense in the south-east part of the region, which is closer to the Mediterranean sea, and as in Figure 6.2, the temperatures are higher in urban areas.




(a) Radiation ($W/m^2$)

(b) Humidity (%)

(c) Wind speed (m/s)

(d) Temperature (°C)

Figure 6.4: Distribution of radiation, humidity, wind speed and temperature values in the Emilia-Romagna region (obtained through OpenStreetMap [102]). The red markers represent the position of the weather stations. The $x$-axis reports the longitude, the $y$-axis the latitude.

We do not report the histograms of the values for the dataset obtained from the 9 stations because the information content is limited and the graphs would be superfluous.

## 6.2 Comparison Solution - Voronoi Tessellation and Hierarchical Clustering

This section describes a simple yet effective methodology to estimate missing environmental information by cluster analysis and normalized convolution (see Figure 6.5) [4].

We have developed a pipeline of standard techniques for estimating missing values that uses standard methods such as hierarchical clustering and Voronoi tessellation [4]. The use of this technique allows obtaining results that we use as a **baseline to validate the results obtained by**

**HybridIoT**. The developed pipeline has been evaluated on the regional dataset described in the previous section, using temperature information.


Figure 6.5: Sketch of the main steps of the methodology based on Voronoi tessellation and hierarchical clustering.

### 6.2.1 Agglomerative Hierarchical Clustering

In a clustering problem we are given a dataset of "points" and the goal is to partition this dataset into a finite set of disjoint subsets (i.e. clusters) such that the union of all subsets covers the whole dataset. A high quality clustering is one in which the points in any particular subset are more similar to each other than the points in other subsets [78].

One common technique for clustering is known as agglomerative hierarchical clustering; this approach starts considering each single point into a single cluster and then it iteratively merges the closest pair of clusters according to some similarity criteria until all points belong to one cluster. The main drawbacks of this approach are that the points that have been incorrectly grouped at an early stage cannot be reallocated subsequently and different similarity measures among the clusters can lead to completely different results [51].

The hierarchical methods group training points into a typical tree structure known as *dendrogram* which represents a sequence of nested clusters constructed top-down or bottom-up. The root of the tree represents the cluster that includes the whole dataset of points while each leave has one point. By cutting the tree at a certain level we obtain a clustering into disjoint groups (Figure 6.6) [113].

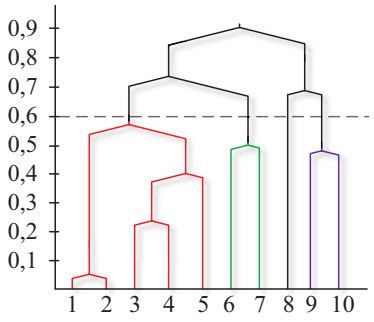



Figure 6.6: Colors represents the clusters obtained by thresholding the dendrogram at $0.6$ (the dashed line).

A major challenge in cluster analysis is determining the optimal number of clusters. There are two principal approaches to face this problem. One common technique depends on plotting

an optimization parameter against a number of clusters and on choosing the number which corresponds to a large change in the parameter value. A second method seeks large changes in inter-group fusions where fusion distances are generally determined with the aid of the dendrogram. Both these techniques depend on the determination of relatively large changes in an index rather than its minimization or maximization and therefore require human interpretation and subjective analysis of what should be considered a "large change" [35]. In the *gap statistic method* [132] we suppose that we clustered the set of points into $k$ clusters $C_1, C_2, \ldots, C_k$. Let

$$D_r = \sum_{i,j \in C_n} d_{ij}$$

be the sum of pairwise distances for all points in the cluster $r$ where $|C_r| = n_r$ and let

$$W_k = \sum_{r=1}^{k} \frac{D_r}{2||C_r||}$$

be the variance that represents the dispersion of the points within the $k$ clusters. The idea of gap statistic is to compare the dispersion of the points within each of the $k$ clusters to its expectation under an appropriate null reference distribution [132].

### 6.2.2 Normalized Convolution

Dealing with irregularly sampled data due to the presence, for example, of noise or instrumental error is quite common in many scientific fields, such as astrophysics, geoscience, oceanography, telecommunications, remote sensing and medical imaging. In general, performing operations on incomplete or irregularly sampled data is a non-trivial task and therefore it is often required to reconstruct the irregularly sampled data or resample it onto a regular grid. These operations can be carried out through *direct* methods, which can involve the computation of irregularly sampled data in the frequency domain and the inverse transform to obtain a regularly spaced signal [91, 59, 24]. Normalized convolution is a *non-direct* methodology widely used for filtering incomplete or uncertain data which is based on the separation of both data and operator into a signal part and a certainty part. A map indicates the presence degree of a sample in a given position and, in particular, a binary map would indicate just the absence or presence of the signal. This approach was described for the first time by Knutsson and Westin for digital image analysis with a simple and fast implementation [77].

Let $S$ be the positive map that represents certain samples of bi-dimensional data $I$. If we indicate by $\{S \cdot I\}$ the pointwise product of $S$ and $I$ and by $\{K * I\}$ the usual convolution with a kernel $K$, then normalized convolution is defined by

$$NC(I, S, K) = \{K * S \cdot I\} \,/\, \{K * S\}$$

In other words, to reconstruct the data $I$ from its samples specified in $S$, we just have to weight $\{K * S \cdot I\}$ by the confidence $\{K * S\}$ of the results generated.

The kernel, centered in the origin, have the original and general form

$$K_{x,y} = \begin{cases} r^{-\alpha} \cos^{\beta}\left(\frac{\pi r}{2R}\right) & \text{if } r < R \\ 0 & \text{otherwise} \end{cases}$$

where $r = \sqrt{(x^2 + y^2)}$ denotes the distance from the neighbourhood center, $R$ denotes the maximum distance from the neighbourhood center, $\alpha$ and $\beta$ are positive integers. Usually, the parameters $\alpha = 0$ and $\beta = 2$ are used, thus obtaining the bi-dimensional raised cosine depicted in figure 6.7.

To avoid over-smoothing the output signal, $K$ should be big enough to contain just some pixels of the input signal. Vice versa, if the distance between the nearest samples in $S$ is greater than the size $R$ of $K$, then the reconstructed image will contain gaps. Without a priori information, $R$ is automatically set to the minimal distance among the available samples to reduce artifact effects along discontinuities: at least one point lies always within the radius [7].

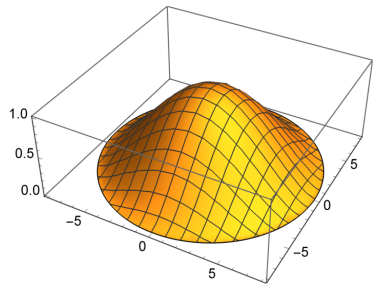



Figure 6.7: Graphic representation of the typical smoothing kernel, as described in [77].

It is noteworthy that the normalized convolution can be calculated through efficient lookup tables if we accumulate the contribution of the few known samples according to the reflected kernel's values about its center (Figure 6.8), instead of directly calculating the scalar product at each unknown point. Instead, this latter approach is preferable if we are interested in a single missing point, without considering the rest of the signal (Figure 6.9).

A variant of this algorithm is known as adaptive normalized convolution and it modulates both the size and the shape of the kernel $K$, according to the position of certain samples [106, 107]. In this case, implementing an optimized and efficient custom convolution routine can be quite difficult: a different filter should be arranged for each point of the output image and an estimate of the gradient of the whole signal is used to determine this proper kernel. Obviously, this gradient itself is just an approximation since it has to be computed from available samples specified by $S$. Actually, we preferred not to use this approach because it requires a considerable amount of computational time and usually its performances do not justify the enhancement of the final result due to the very few known samples.

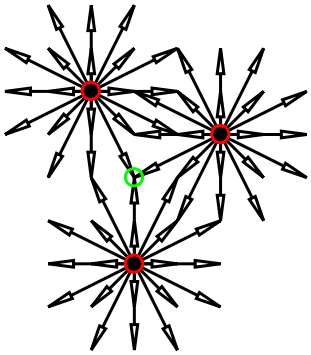


Figure 6.8: If very few sensors (marked in red) are available, it is convenient to calculate the values in all remaining points by indirect contributions of the actual temperatures: their precalculated products by the kernel are simply added to accumulation arrays.

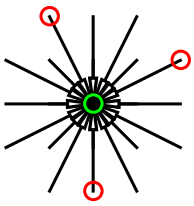


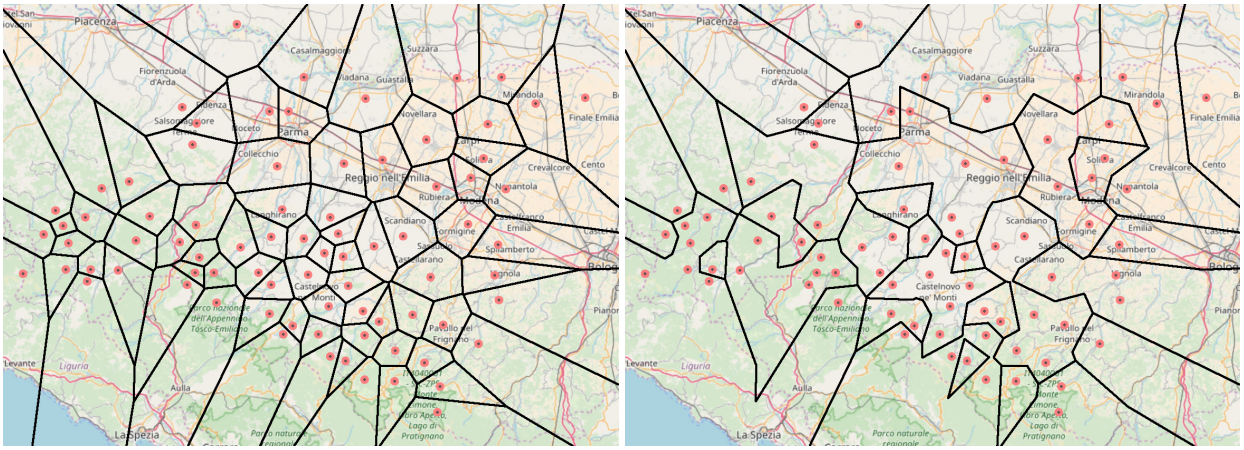
Figure 6.9: If we are interested in estimating the temperature in just one point (marked in green), the straightforward implementation of the convolution remains the fastest solution.

### 6.2.3 Analysis of the Results

The hierarchical clustering technique groups together these stations and therefore their corresponding tiles. The hierarchical clustering tree is created starting from the information $I$ of the dataset and then the optimal number $k$ of clusters is computed by using the gap statistic method. Stations belonging to the same cluster and in adjacent Voronoi tiles are grouped together: this refines the original clustering by separating eventual stations which act in the same manner though are too far one from each other (Figure 6.10). These new tiles are no more convex in general and contain stations with a similar behavior, normally due to the local orography.

The certainty map $S$ needed by the normalized convolution stores just the positions of the stations. In order to estimate the temperature value in a missing position, it is sufficient to apply the normalized convolution on the information map $I$ from the dataset and on the certainty map $S$, limited to the cluster containing the position (Figure 6.11).

The algorithm was coded in Matlab language without particular optimizations; nonetheless it takes about a second to cluster the whole dataset and it is practically instantaneous in computing the normalized convolution. It should be noted that the clusters should be computed from time to time and while it is appropriate for a server to update the cluster periodically, the normalized convolution can be efficiently calculated even by low-end mobile devices. That is, the certainty map $S$ can vary not only according to the response provided by the fixed stations, but it can be refined




(a) (b)

Figure 6.10: Superimposition of the Voronoi diagram (Figure 6.10a) and subsequent clusters (Figure 6.10b) on the map of Emilia-Romagna (obtained through OpenStreetMap [102]). The red dots indicate the weather stations.

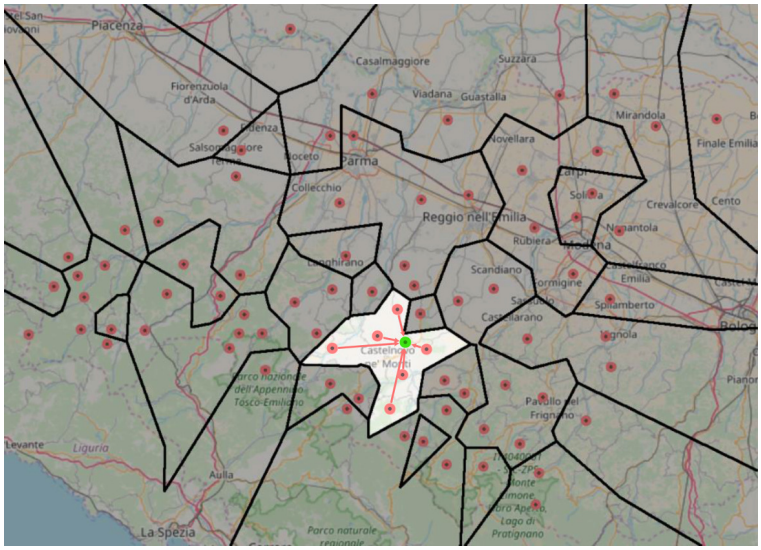



Figure 6.11: Only the stations within the same cluster (in detail) contribute to the normalized convolution, similarly to figure 6.9.

also through extemporaneous surveys.

To verify the correctness of the results we considered a leave-one-out cross validation: for each experiment a precise station has been removed at a time, in order to evaluate the estimation from the remaining stations. Figure 6.12 shows the average absolute error (in Celsius degree) of each station during the considered 196 days. The absolute average overall error is just 0.03°C.

The same normalized convolution can be effectively applied to determine the reliability of the proposed method on the entire lattice: it is sufficient to propagate the average error of each weather station to measure the correctness of the extrapolated measurements in any point without a real

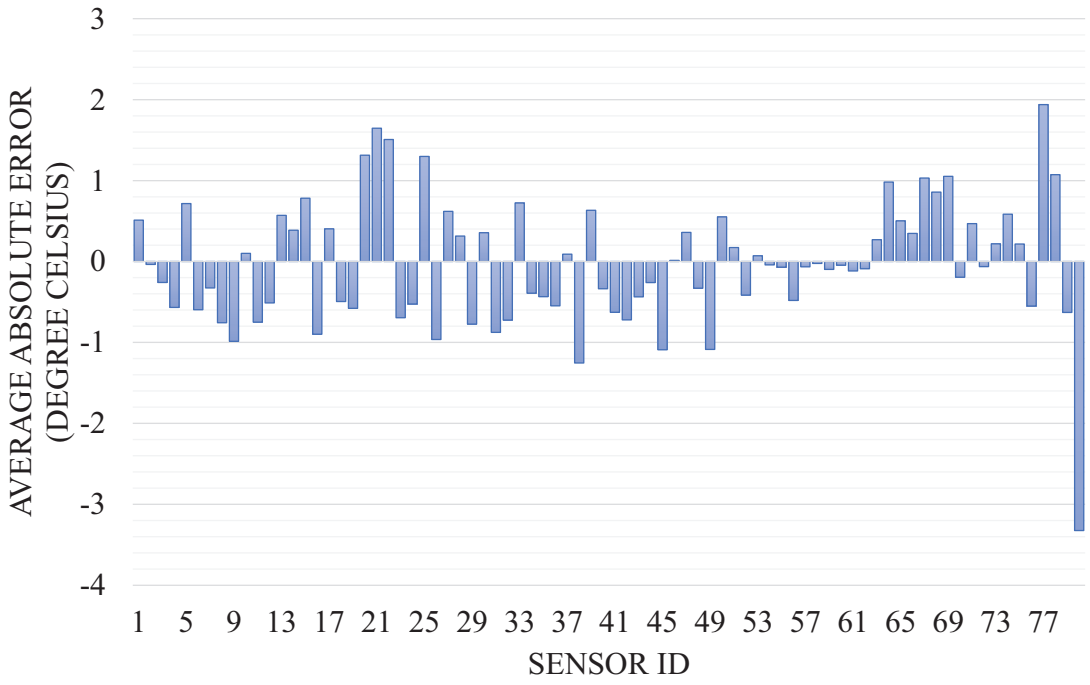



Figure 6.12: Error bar of deduced temperatures (degree Celsius) for each station, computed through the leave-one-out process.

station (figure 6.13).

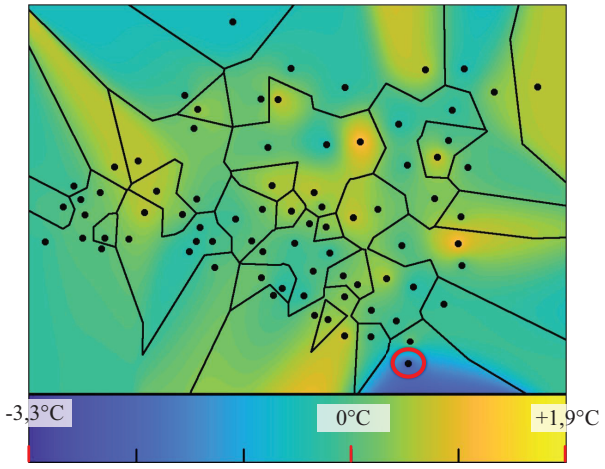



Figure 6.13: Expected error through normalized convolution. Most stations do not introduce any error (green); some stations report warmer values than the real temperatures (yellow); only one station indicates a lower temperature (blue). We highlight that this colder station (circled in red, #79 in Figure 6.12) is due to the absence of neighboring stations for a correct evaluation.

## 6.3 Endogenous Estimation

We carried out both cooperative and non cooperative evaluation: in the first case, the ACAs estimate missing information by cooperating with the other agents, while in the second case the agents use only their ACWs to estimate missing information. When using cooperation, the ACAs choose the ACAs with which it cooperates according to two criteria:

- *Nearest Agents*: ACAs cooperate with the nearest with respect to their position;
- *Most Confident Agents*: ACAs cooperate with those that have a high confidence value.

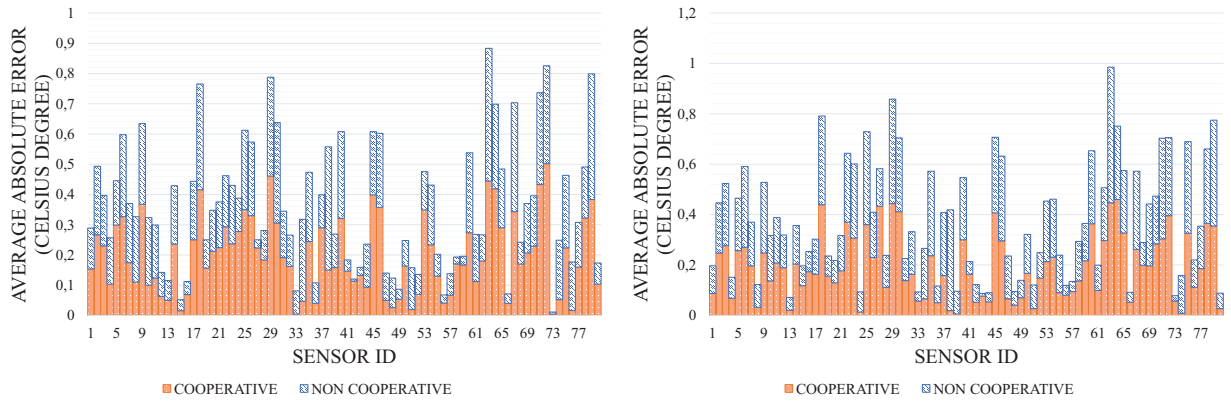
In order to assess the effectiveness of the cooperation, three evaluations have been carried out using different percentages of ACAs involved in the cooperative process: $25\%$, $50\%$ and $75\%$ respectively. The results for the cooperative scheme are calculated as the mean of the results obtained by these three evaluations.

### 6.3.1 Analysis of the Results

Figure 6.14 shows the average error, in degree Celsius, obtained by estimating temperature values through non cooperative and cooperative cases using both agents selection criteria described. For the non-cooperative case, the average error among the considered sensors is $0.074$°C, the standard deviation is $1.865$°C. For the cooperative case, the average error is $0.060$°C, the standard deviation is $1.970$°C using the most confident agents criterion. By using the nearest agents criteria, the average error is $0.081$°C, the standard deviation is $1.968$°C. Figure 6.14 shows that the results obtained by the cooperative method are comparable to those obtained by the endogenous (individual) estimation; this proves that the joint operation of a collective of autonomous agents can accurately estimate missing information. Moreover, the most confident criterion gives better results than the nearest criterion: cooperation between agents observing similar environmental dynamics enable agents to provide accurate estimates compared to the use of physically close sensors. This result proves that the proposed technique constitutes a step forward towards the development of large-scale ubiquitous systems for the estimation of missing values, where the distributed computation and the mutual interaction between devices allow obtaining significant results and addressing the challenges of openness and highly dynamic environment.

We evaluated the endogenous estimation scheme separately on different types of information to prove that the proposed technique is able to estimate information of different types without any configuration. Figure 6.15 shows the absolute error obtained from estimating solar radiation, average daily wind speed, and relative humidity. The error obtained is fairly similar among the different information because of the limited amount of information available and the limited number of variations in the observed data.

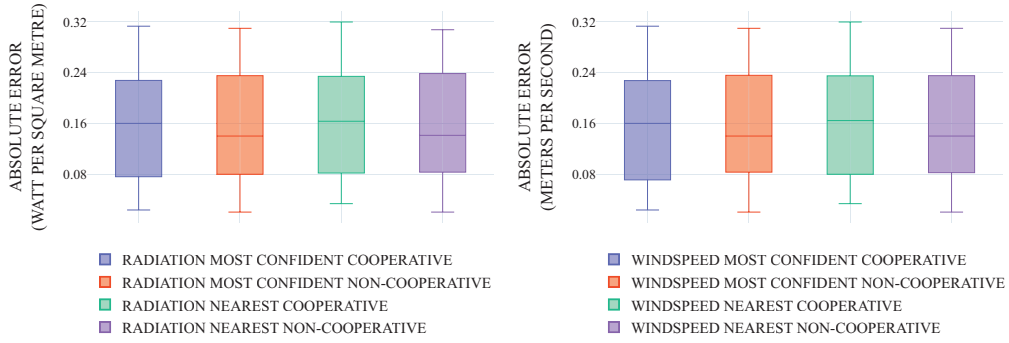
Table 6.1 summarizes the average absolute error obtained by using endogenous estimation on solar radiation, wind speed, and relative humidity.




(a) Nearest neighbors criterion case.

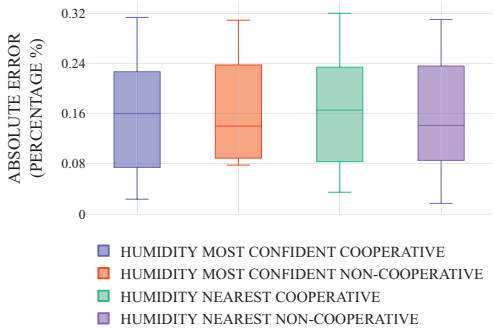
(b) Most confident neighbors criterion case.

Figure 6.14: Average absolute error (in degree Celsius) obtained from endogenous estimation, for both cooperative and non-cooperative cases. For the cooperative case, both nearest neighbors criterion (Figure 6.14a) and most confident neighbors criterion (Figure 6.14b) are used to evaluate the agents with which cooperate.




(a) Radiation

(b) Wind Speed




(c) Humidity

Figure 6.15: Box plot of absolute error obtained through the endogenous estimation, for both cooperative and non cooperative cases, using solar radiation, wind speed, and relative humidity. We used both the nearest neighbor criterion and most confident criterion.

Table 6.1: Average absolute error obtained by endogenous estimation.

| | Average Absolute Error | |
|---|---|---|
| | **Most confident agent strategy** | **Nearest agent strategy** |
| **Humidity (%)** | 0.16320 | 0.16240 |
| **Solar radiation (W/m$^2$)** | 0.15851 | 0.16265 |
| **Wind speed (m/s)** | 0.15888 | 0.16141 |

Although the information used is of a different type, with different ranges of values and different distributions, HybridIoT enables obtaining a low error without any configuration: it can intercept and learn the environmental dynamics to accurately estimate missing information, independently from its type. Two main consequences can be discerned from the obtained results: the technique (*i*) makes it possible to obtain precise estimates for different types of environmental information, and (*ii*) allows estimating of environmental information using information which type is not known *a priori*; this suggests that the user does not have to provide, in advance, any estimation or configuration mechanism depending on the type of data and new types of information can be introduced without any particular modification.

### 6.3.2 Comparison to the State of the Art

This section compares the results obtained by the endogenous estimation in HybridIoT with different standard techniques for estimating missing information. The section is structured as follows: Section 6.3.2 presents the KNIME platform used to evaluate different techniques for estimating missing information on the presented regional dataset. The results obtained by the techniques available in this platform are subsequently analysed in section 6.3.2.

#### Comparison Solution - KNIME Platform

The KNIME analytic platform is a free software that provides a graphical drag-and-drop environment where pipelines can be assembled by connecting nodes that perform data analysis tasks [15]. In KNIME, nodes are components represented as boxes having input and output ports. Each node transforms and processes data according to specific functionalities. Input/output connections ports allow data to flow through the pipeline. Figure 6.16 shows an example of workflow in KNIME.

The KNIME platform was chosen for the experiments because of its availability, ease of use and easy reproducibility of experiments that do not require any programming languages. Moreover, KNIME provides a large number of regression techniques, which allows a more exhaustive comparison with the proposed approach. The following regression techniques have been used for estimating missing information, the nodes being available on KNIME: linear regression, polynomial regression, random forest regression [67], fuzzy rules [16], gradient boost trees regression [50], Autoregressive Integrated Moving Average (ARIMA) [75], Pace regression [142], Radial Basis Function (RBF) [139] and isotonic regression [151]. Pace regression, RBF and isotonic regression nodes are available

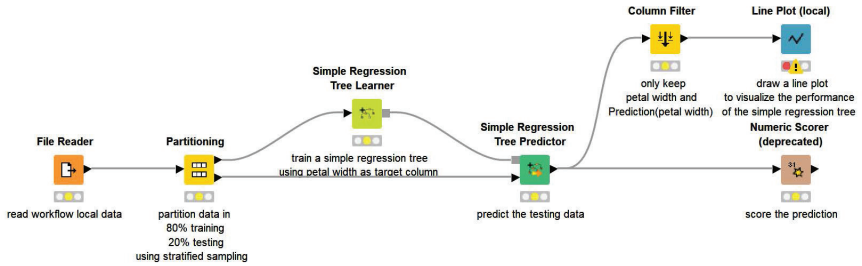



Figure 6.16: An example of workflow in KNIME

Table 6.2: Configuration of state-of-art techniques nodes available in KNIME.

| **Technique** | **Properties** |
|---|---|
| Linear Regression | No properties |
| Polynomial Regression | Maximum polynomial degree: 3 |
| Random Forest Regression | No properties |
| Fuzzy Rules | Missing Values: Best Guess |
| Gradient Boost Trees Regression | Missing Value Handling: XGBoost<br>Alpha: 1.0 |
| ARIMA | AR/I/MA order: 1<br>Estimation method: conditional likelihood [61] |
| Pace Regression | Estimator: ordinary least squares<br>Threshold value: 2 |
| RBF | Number of Gaussian basis functions: 2<br>Ridge factor for quadratic penalty on output weights: 0.01<br>Tolerance parameter for delta values: 1.0e-6<br>Scale optimization option: one scale per unit<br>Use conjugate gradient descent: true<br>Use normalized basis functions: true<br>Size of the thread pool: 1<br>Number of threads to use: 1<br>Use random number seed: true |
| Gaussian Processes | Level of Gaussian Noise with respect to transformed target: 1<br>Kernel used: polynomial |
| Isotonic Regression | No properties |

through the Weka data mining framework [149], which can be integrated with KNIME. The comparison with the state of the art does consider the solutions described in chapter 3 as authors do not provide any tool that allow a comparison with the data used in this thesis.

The evaluation using the state of the art techniques has been carried out using the default configuration for each node. Table 6.2 summarizes the default parameters used to configure the nodes in KNIME.

The $k$-fold cross-validation and an auto-regressive model are used to evaluate the accuracy of the state of the art techniques; the required nodes are available in KNIME, using a $k$ value of $5$, $10$ and $15$. An auto-regressive model assumes that the previously observed samples can be used to predict accurately the value at the next time step. For instance, we used $4$ samples to implement the auto-regressive model using KNIME.

There is no formal rule for choosing the value of $k$ to use for cross-validation. However, as $k$ gets larger, the difference in size between the training set and the resampling subsets gets smaller.

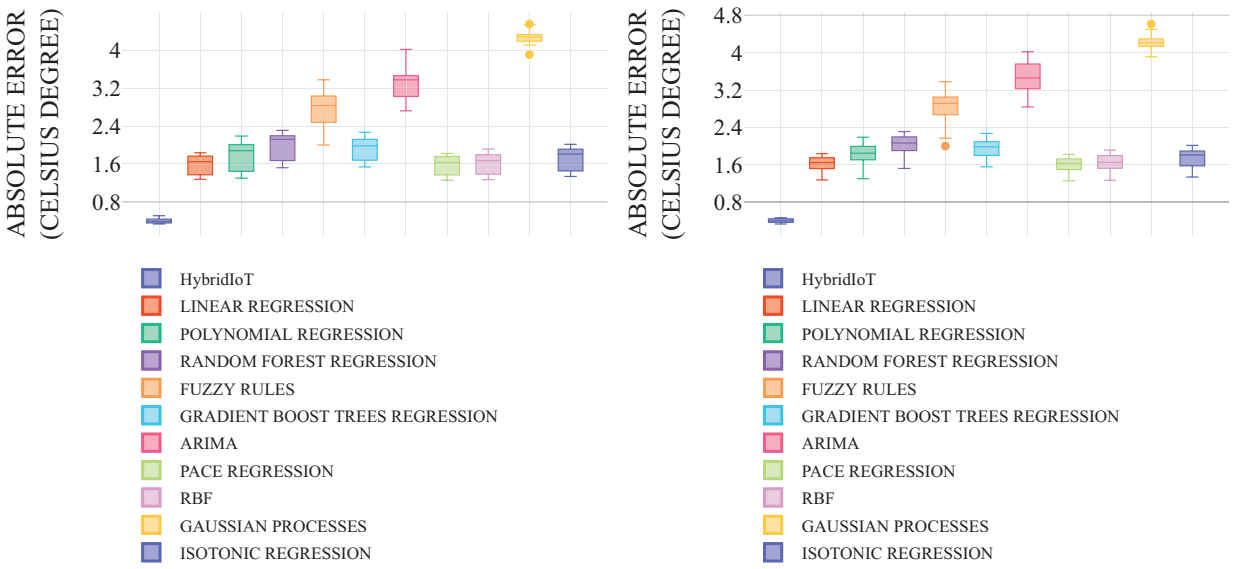
Consequently, the difference between the estimated and real values becomes smaller [80].

The results obtained by the endogenous estimation have been compared to state of the art techniques using the KNIME analytic platform. This free software provides a graphical drag-and-drop environment where pipelines can be assembled by connecting nodes that perform data analysis tasks [15]. In KNIME, nodes are components represented as boxes having input and output ports. Each node transforms and processes data according to specific functionalities. Input/output connections ports allow data to flow through the pipeline.

The evaluation using the state of art techniques has been carried out using the default configuration for each node, which parameters are listed in Table 6.2.

### Analysis of the Results

We used the 15 sensors that gave the worst results using the endogenous estimation in HybridIoT to evaluate the state of the art techniques. Figure 6.17 shows the box plots that depict the absolute error obtained by the state of art techniques, calculated as the average among the considered folds (5, 10 and 15).




(a) Nearest agents criterion case

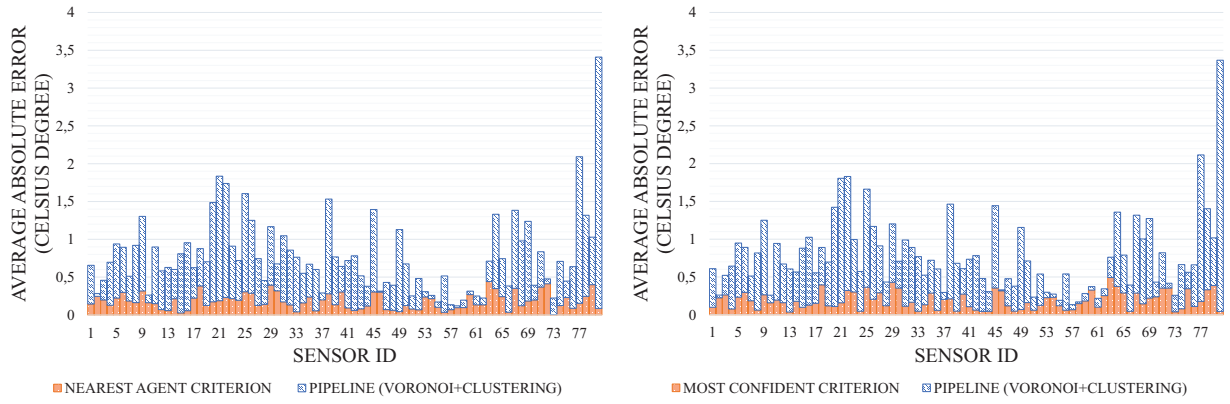
(b) Most confident agents criterion case

Figure 6.17: Box plot of absolute error (in degree Celsius) obtained by the state of the art techniques on the 15 sensors that gave the worst results using the proposed solution. Figure 6.17a compares the results of the proposed approach using the nearest agents criterion, Figure 6.17b compares the results of the proposed approach using the most confident agents criterion.

The results in Figure 6.17 are obtained from the dataset containing only temperature values. Although the average error obtained for each sensor is less than 1°C, the proposed cooperative approach outperforms the state of the art techniques. In fact, cooperation enables ACAs to use not only information in their historic, but also information coming from the other available agents.

Figure 6.18 shows the comparison of results obtained by the pipeline of standard techniques (Voronoi tessellation and hierarchical clustering) and HybridIoT using respectively the nearest agents and most confident agents criteria for the cooperative case. HybridIoT shows better results with respect to the pipeline using both agents' selection criteria (near agents and most confident agents). Moreover, we outline some fundamental differences from HybridIoT:

- HybridIoT can learn environmental dynamics and estimate missing information instantaneously in areas of the environment not sufficiently covered by sensors;
- HybridIoT enables introducing new sensing devices at any time;
- using the pipeline, a device that needs to estimate missing information must be located in a region where at least one working sensor must be present. In HybridIoT, an ACA can estimate missing information through cooperation but also using previously perceived values, thus overcoming the lack of sensors in the proximity of the ACA.




(a) Nearest neighbors criterion case.

(b) Most confident neighbors criterion case.

Figure 6.18: Average absolute error (in degree Celsius) obtained from the pipeline that uses Voronoi tessellation and Hierarchical clustering and HybridIoT, using nearest agent (Figure 6.18a) and most confident criterion (Figure 6.18b).

We presented the results obtained by the endogenous estimation scheme. We compared the results to those obtained by state of the art regression techniques to assess the accuracy of the results obtained. Also, we carried out a comparison using a pipeline of standard techniques including Voronoi tessellation and hierarchical clustering, on the same dataset. In this case, HybridIoT outperforms the results obtained by the pipeline.

## 6.4 Endogenous Estimation by Confidence Zone

As in the previous experimentations, the evaluation of the cooperative endogenous estimation technique by the confidence zone is pursued by using the leave-one-out validation: each real sensor

has been replaced by an ACA that provides estimates at the point where the real sensor is located. The experimentation has been repeated for each real sensor available in the dataset.

Because the ACAs modify continuously their confidence zones when pursuing this type of estimation, the results presented were obtained by using confidence zones containing the sensors that provide relevant values for the calculation of the estimates. Figure 6.19 shows the steps of the evaluation process for the endogenous estimation technique using the confidence zone.

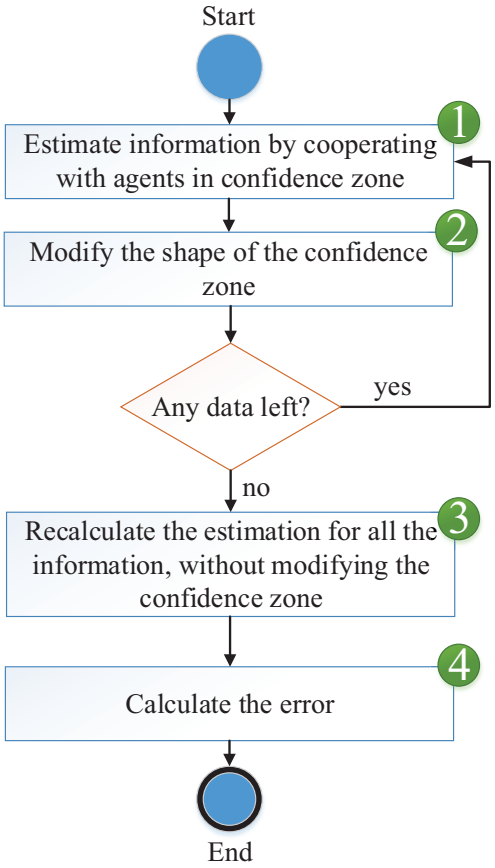



Figure 6.19: Main steps of the evaluation of HybridIoT for the endogenous estimation using the confidence zone.

In the first step of the validation, the information is estimated using the cooperative method involving the use of information acquired by sensors located within the confidence zone of the ACA that encountered an incompetence NCS (step ❶). The initial confidence zone is large enough to contain all the available sensors. For each estimate, the ACA changes the shape of its confidence zone according to the technique presented in the previous chapter (step ❷): the ACA enlarges the confidence zone towards the sensors that perceive relevant values for the estimation, while reduces the confidence zone in the direction of the others sensors. At the end of the estimation process, the shape of the confidence zone has been reduced to contain only a limited subset of the available sensors. According to the proposed technique, the sensors within the resulting confidence zone are

the most pertinent for the estimation of information at the location of the ACA that encountered an incompetence NCS. Figure 6.20 shows some examples of confidence zones containing the sensors relevant to the estimate calculation for ACAs that have encountered an incompetence NCS.

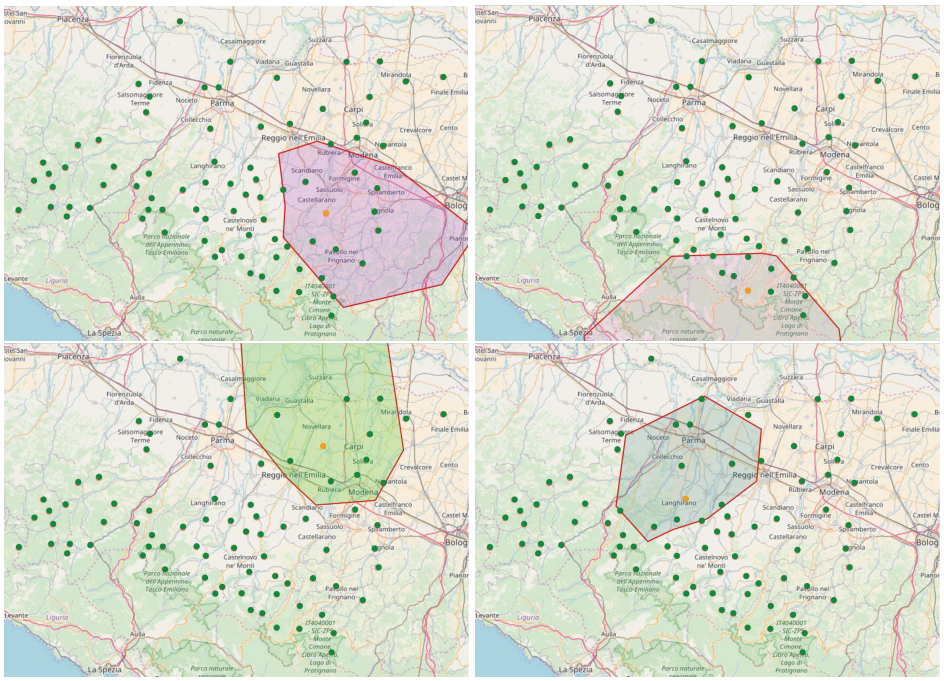


Figure 6.20: When estimating by cooperation with nearby agents, the ACA that encounters an incompetence NCS (the yellow marker) cooperates with the other agents (the green markers) and modifies its confidence zone according to the perceived values. Only the agents perceiving pertinent information are kept inside the confidence zone.

The information is then estimated once again, this time using the information acquired by the sensors within the reduced confidence zone (step ❸); in this step, the shape of the confidence zone remains unmodified. The last step of the validation is to evaluate the error related to the calculated estimated values (step ❹): for each sensor, this is calculated as the average of the differences in absolute value between the real information (the information perceived by the replaced sensor) and the estimated information (the information estimated by the ACA).

We applied the same estimation technique using both regional dataset containing:

- 80 stations and temperature values,
- 9 stations and heterogeneous information (temperature, humidity, solar radiation and wind speed).

### 6.4.1 Analysis of the Results

Figure 6.21 shows the absolute error obtained by the proposed approach, compared to the pipeline of standard techniques (Voronoi tessellation and hierarchical clustering). For the endogenous

estimation technique, the average absolute error is 1.17°C.

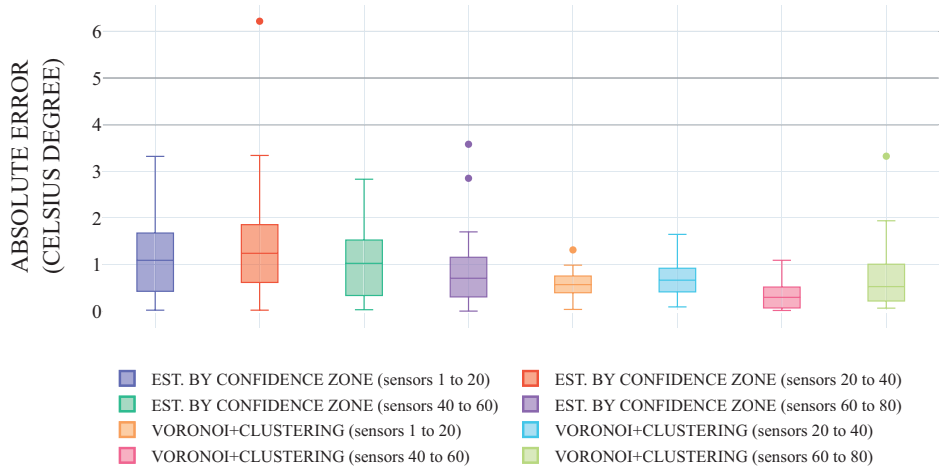



Figure 6.21: Results obtained by the proposed estimation technique using the confidence zone and the baseline results produced by the pipeline of standard methods (Voronoi+hierarchical clustering).

Compared to the pipeline of standard techniques, HybridIoT enables calculating estimates considering devices that can enter or leave the system at any time. Moreover, thanks to the local computation of ACAs, HybridIoT can be deployed in large scale contexts.

The estimation technique enables estimating missing information instantaneously using values acquired from devices that can enter the system at any time. The confidence zone is instantaneously modified by ACAs and evolves constantly: its shape changes according to the information and the position of the sensors inside it. The modification of the confidence zone can be also influenced by the mobility of the sensors present in the environment: the estimation technique enables to calculate missing values even using sensors whose position is not fixed.

The same estimation technique has been evaluated on the dataset containing heterogeneous information. Figure 6.22 shows the results obtained from the proposed approach by comparing temperature, humidity, wind speed and solar radiation on a limited number of sensors. We obtained an average absolute error of 0.2 (m/s) using wind speed data, 3.77 (%) using humidity data, 0.82 (°C) using temperature data and 5.36 (W/m$^2$) using solar radiation data.

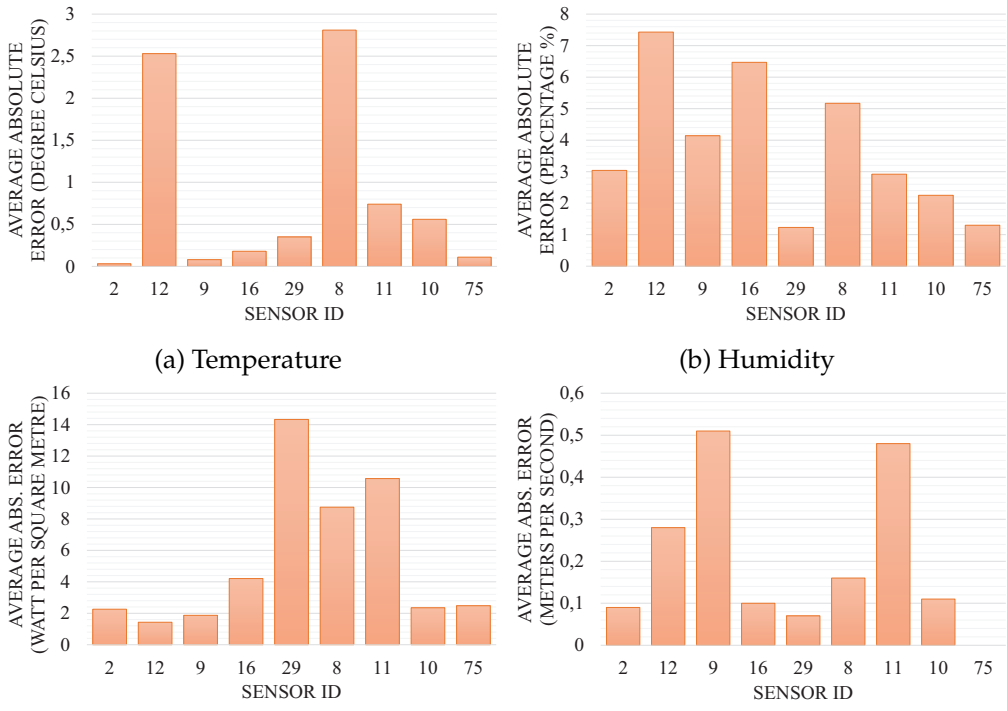
The proposed technique enables to estimate missing information through a mechanism of cooperation addressing the properties of openness and heterogeneity: the openness is addressed because some sensors that provide information that is not relevant to the calculation of estimates are excluded from the confidence zone. ACAs instantaneously adapt to the environmental context in which they are located to determine the set of agents with which to cooperate for estimating missing information. Heterogeneity is addressed because the same estimation technique has been applied using different types of information without any particular configuration.

Figure 6.23 shows the results obtained from endogenous estimation by the confidence zone and the techniques available in KNIME. We remark that the box plots have, especially for temperature

and humidity, a wide interquartile range due to the variability of the information perceived by the agents involved in the estimation process. The resolution process through which the results have been obtained is cooperative, therefore multiple agents that perceive different environmental dynamics are involved in the estimation process. For this reason, the variability is high. Despite this, the median error is lower than the other techniques. The results show that the cooperative estimation approach allows obtaining significative results through the integration of multiple information.

## 6.5 Exogenous Estimation using Heterogeneous Information

We used the $k$-fold cross-validation to evaluate the accuracy of the obtained results. This validation technique partitions the original sample in $k$ subsamples. Among the $k$ subsamples, the $k$-th one is retained as the validation data, for which estimated data is compared to the real information; the remaining $k-1$ subsamples are used as training data. During the training phase, the ACAs assemble the ACWs for the available data. During the test phase, the information from the $k$-th partition is estimated, thus simulating the unavailability of the sensor. The test phase is then repeated $k$ times,




(a) Temperature

(b) Humidity

(c) Solar Radiation

(d) Wind Speed

Figure 6.22: Average absolute error obtained by applying the cooperative estimation technique by the confidence zone on a dataset of heterogeneous information. The four graphs show the error obtained by applying the technique on four datasets containing respectively temperature (Figure 6.22a), humidity (Figure 6.22b), solar radiation (Figure 6.22c), wind speed (Figure 6.22d).

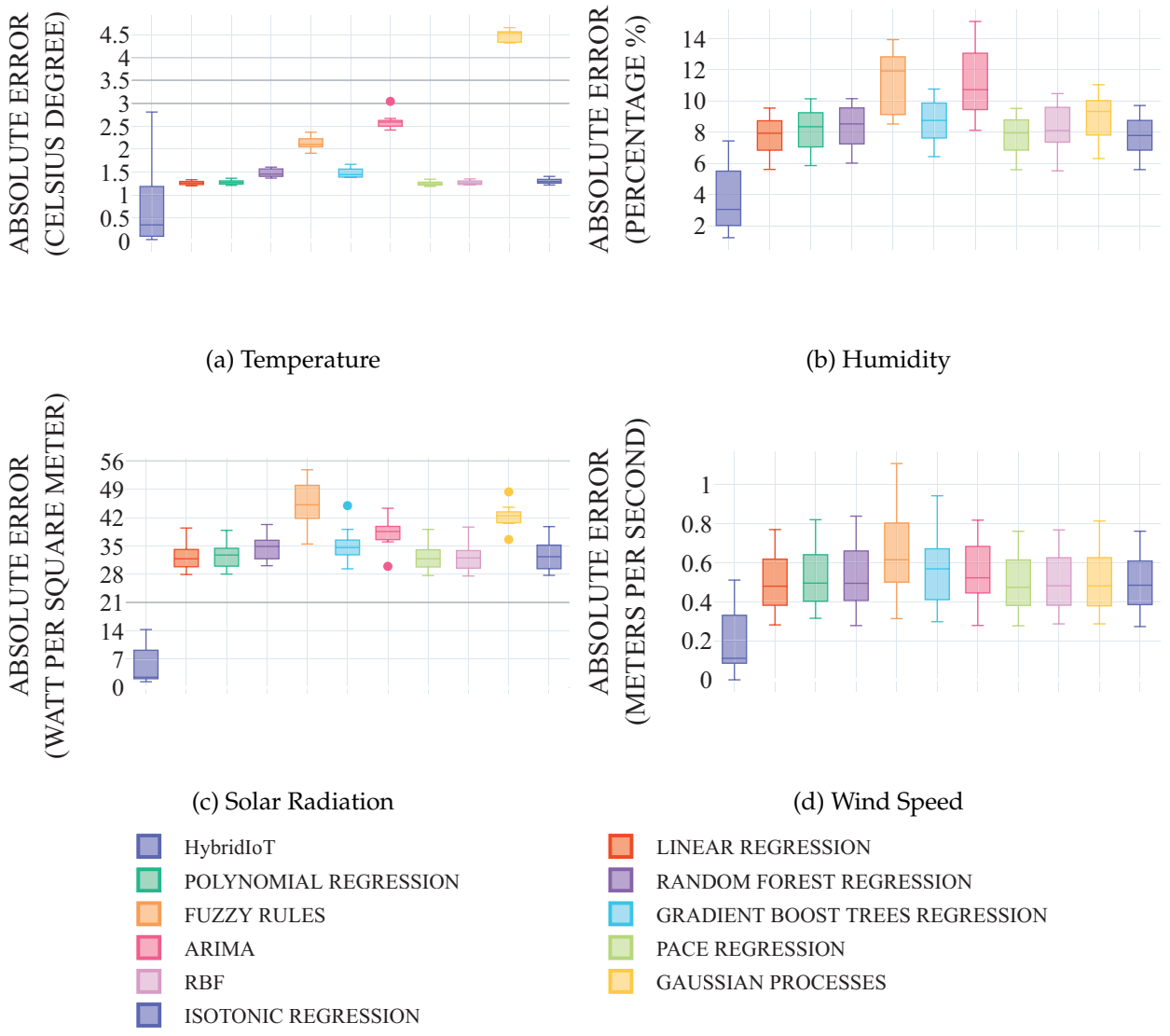



Figure 6.23: Results obtained by comparing the proposed endogenous technique by confidence zone and the estimation techniques available in KNIME.

with each of the $k$ subsamples used exactly once as the test data.

Figure 6.24 shows the pipeline used to evaluate the results obtained by HybridIoT. The first step of the validation consists in using the training partitions to define the ACWs to be used for estimating the missing data (step ❶). The $k$-th partition is being validated by estimating its data (step ❷) to simulate the unavailability of the sensor. During this step, the agent pursues an endogenous estimation. Then, the ACA determines the set of the other agents with which it can cooperate to provide an accurate estimate (step ❸). Finally, the ACA pursues an exogenous estimation by cooperating with the available agents (step ❹). The steps ❶ to ❹ are repeated for each

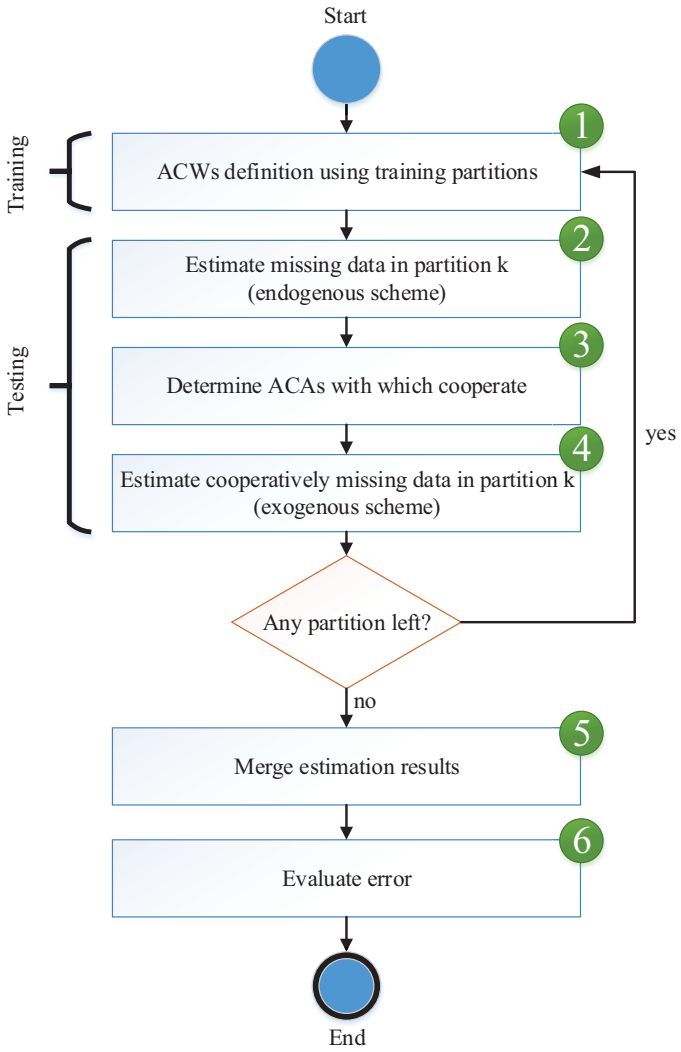



Figure 6.24: Main steps of the pipeline used to evaluate the proposed solution.

partition. The estimates provided for all the partitions are merged to compare the results to the real values (step ❺). The average error is calculated by comparing the estimated and the real values for each sensor of the dataset (step ❻). In the experiments, we used a $k$ value of 5, 10 and 15. For the dataset containing only homogeneous information, the values of $k$ chosen (5, 10, 15) yield to test partitions containing respectively 20%, 9% and 6% of data with respect to the size of the dataset. No significant differences in terms of error were observed using different values of $k$.

### 6.5.1 Analysis of the Results

As far as we know, nowadays no solution is available to estimate missing information by integrating heterogeneous information from different data sources. However, although the experiments were carried out on one type of information, the same technique can be applied for estimating missing

information of different types. Because we cannot compare the results obtained through the heterogeneous estimations to any specific technique, we show that the results obtained by the heterogeneous estimation are on a par with those obtained in the homogeneous case.

Exogenous estimation is addressed through cooperation between agents that perceive different types of information. In the experiments, we estimated temperature values using context windows containing solar irradiance, humidity and wind speed.

As described in the previous paragraph, an ACA that encountered a NCS evaluates different estimations for each weight obtained from a cooperative process with the other agents. The agent then evaluates a histogram of the estimation values to exclude those that are not relevant. In the experiments the number of bins has been set to 10: this value was obtained by using the Freedman-Diaconis rule [49], which allows calculating the size of the classes of a histogram. The number of bins was calculated as the median of the number of bins obtained by applying the Freedman-Diaconis rule for all the stations.

Figure 6.25 shows the average error for the regional temperature data set using heterogeneous ACWs. The results show that ACAs can provide accurate estimates through heterogeneous information without using any specific data fusion technique. As for endogenous estimation, the estimation of a missing value using the exogenous technique is almost instantaneous.

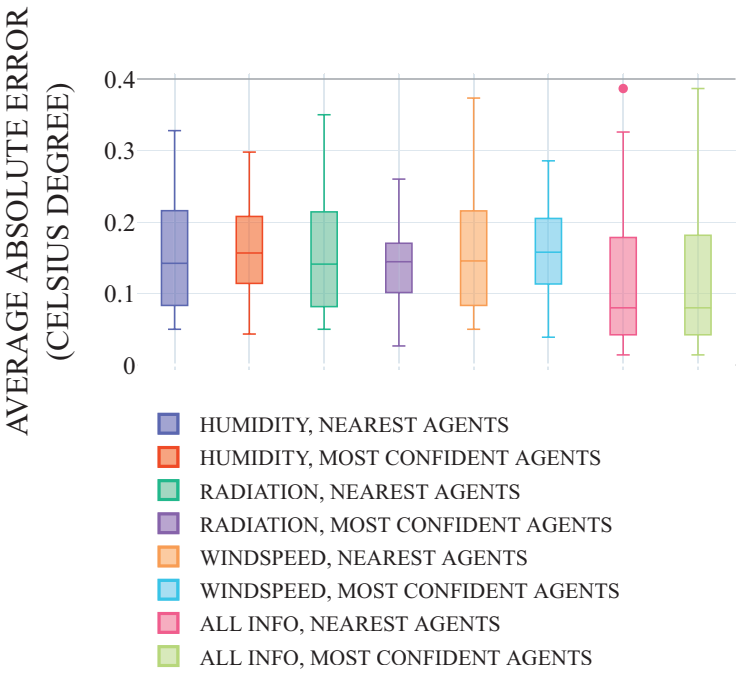



Figure 6.25: Results obtained from exogenous estimation. Each column represents the average error obtained by estimating temperature values using context windows containing the corresponding information. The two neighborhood criteria described were used.

Table 6.3 summarizes the average error and the standard deviation obtained by estimating

the temperature from heterogeneous contexts. The results show that ACAs can provide accurate estimates through heterogeneous information without using any specific data fusion technique. The error resulting from exogenous estimation is low despite the used ACWs contains values that are semantically different from the temperature values. The information used in this estimation scheme does not have significant variation compared to the temperature values. Despite the differences in the variations and information between ACWs, the results in Table 6.3 show that the proposed method can accurately estimate missing information through heterogeneous values.

Table 6.3: Average absolute error and standard deviation (in degree Celsius) obtained by exogenous estimation.

| | **Average Absolute Error** | |
|---|---|---|
| | **Most Confident agent strategy** | **Nearest agent strategy** |
| **Temp. from solar radiation** | 0.02716 | 0.03796 |
| **Temp. from humidity** | 0.02654 | 0.03679 |
| **Temp. from wind speed** | 0.02531 | 0.03914 |
| **Temp. from all info** | 0.07 | 0.05935 |
| | **Standard Deviation** | |
| | **Most Confident agent strategy** | **Nearest agent strategy** |
| **Temp. from solar radiation** | 1.639136 | 1.645494 |
| **Temp. from humidity** | 1.646173 | 1.647901 |
| **Temp. from wind speed** | 1.64716 | 1.651358 |
| **Temp. from all info** | 1.65435 | 1.62805 |

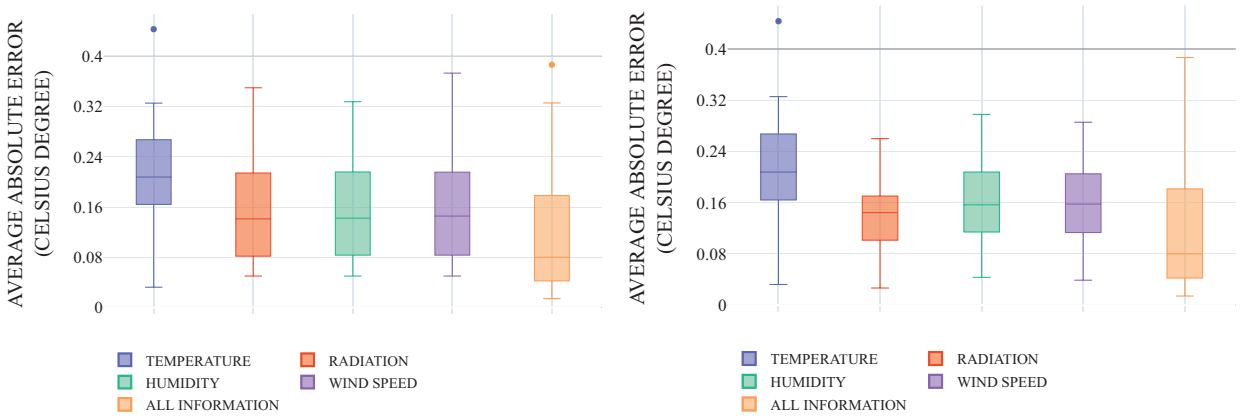
This section presented the exogenous estimation scheme carried out by HybridIoT on the heterogeneous dataset. In the following, we discuss HybridIoT with respect to the results obtained by both endogenous and exogenous estimation schemes. Moreover, we show that the results obtained by exogenous estimation are on a par with those obtained by the endogenous estimation.

## 6.6 Discussion

The obtained results assess the validity of HybridIoT for estimating missing heterogeneous information. With respect to the state of the art techniques, the proposed approach does not require any *a priori* configuration and the estimation of missing value is almost instantaneous.

To prove the effectiveness of the exogenous estimation, we show that the related results are on a par with those obtained by the endogenous estimation, using the same dataset. Figure 6.26 shows the box plots of error, in degree Celsius, obtained by comparing the real temperature values with the estimates calculated through both endogenous and exogenous estimations, this last using radiation, humidity and wind speed. Figure 6.26a shows the results obtained by using the most confident agents criterion, Figure 6.26b the results obtained by using the nearest agents criterion.

The sensors used for this experiment are those that acquired correctly all the available information (temperature, humidity, wind speed, solar radiation) within the considered temporal period. We used both the nearest agents and most confident criteria for the cooperative scheme: in the two cases, the results show that the exogenous estimation outperforms the endogenous one.




(a) Nearest neighbors criterion case

(b) Most confident neighbors criterion case

Figure 6.26: Box plot of absolute error (in degree Celsius) obtained by comparing the results of endogenous and exogenous estimations. Figure 6.26a shows the results obtained by using the most confident agents criterion, Figure 6.26b the results obtained by using the nearest agents criterion.

When estimating a missing value using different types of information that do not vary significantly in time, the weights generated by the estimation process can lead to a negligible variation of the last perceived value. Despite the two selection criteria, the estimation process may include ACWs containing environmental dynamics significantly different from the one containing the information to be estimated. Therefore, some estimates can be considered as outliers because they are significantly different from the real value.

The presence of outliers can be caused by the presence of ACWs whose distance from the one containing the information to estimate is significantly high. This implies that the ACWs contain information following different dynamics. For that, ACAs can estimate information that is not accurate.

Consider Table 6.4, which shows the distances between the ACWs of fixed size (11 and 17). We have chosen the size of the ACWs, for each sensor, of equal length (11 and 17 are divisors of 187). Each value represents the average of the distances between the ACWs observed by each sensor. Each distance value is calculated using Equation 5.8. Each distance value is reported according to the respective units of measurement (°C for temperature, % for humidity, etc.).

The distances between temperature and wind speed, for example, are significantly different. In particular, although this information is considerably subject to variations, the average distance between ACWs containing wind speed is less than the temperature information. This is due to the presence of ACWs that follow similar dynamics over time. Contrarily, the temperature observed in

Table 6.4: Average distance between ACWs of fixed size for each sensor ID

| ACW size | Information | Sensor ID | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | | Finale Emilia | Rolo | Marzaglia | Vignola | Modena urbana | San Pancrazio | Reggio nell'Emilia urbana | Parma urbana | Panocchia |
| 11 | Temperature | 1,10 | 1,16 | 1,14 | 1,15 | 1,14 | 1,12 | 1,10 | 1,12 | 1,15 |
| | Humidity | 4,17 | 4,17 | 5,23 | 5,62 | 7,24 | 5,73 | 6,70 | 6,96 | 6,02 |
| | Solar radiation | 19,32 | 24,64 | 21,29 | 19,66 | 21,62 | 21,72 | 17,75 | 19,38 | 19,94 |
| | Wind speed | 0,44 | 0,51 | 0,32 | 0,26 | 0,40 | 0,30 | 0,18 | 0,23 | 0,39 |
| 17 | Temperature | 0,98 | 0,94 | 0,99 | 1,00 | 1,00 | 0,98 | 0,95 | 0,97 | 1,01 |
| | Humidity | 3,68 | 3,89 | 4,70 | 5,21 | 6,63 | 5,22 | 6,06 | 6,26 | 5,39 |
| | Solar radiation | 18,09 | 23,38 | 20,05 | 18,85 | 20,45 | 20,24 | 16,35 | 18,01 | 18,58 |
| | Wind speed | 0,39 | 0,47 | 0,30 | 0,25 | 0,37 | 0,29 | 0,17 | 0,22 | 0,36 |

the dataset used has different variations over time. For that, we remark the presence of an outlier in the temperature box, in the results reported in Figure 6.26.

Applying exogenous estimation using all types of information available is effective for estimating missing information. Although applying exogenous estimation using one type of different information (with respect to the one that must be estimated) enables obtaining good results, the error obtained by using all types of available information is considerably low. This proves that the use of different types of ACWs enables ACAs to calculate accurate estimates by exploiting multiple types of information.

The ACAs can evaluate accurate estimates thanks to a cooperative behavior of ACAs. Agents evaluate estimates in a short amount of time using the information perceived by the available sensing devices. Because each agent pursues an autonomous computation and has a partial view of the environment, the technique can be employed in large scale urban contexts. Moreover, there is no need for any particular configuration depending on the information used or the sensing devices. The proposed approach enables addressing the property of openness thanks to the cooperative behavior between agents: sensors can enter or leave the system at any time without compromising its operation.

# Conclusion

# Conclusions and Future Perspectives

## General Conclusion

The problem addressed in this thesis can be summarized by this question:

*in an open, continuously evolving environment, how can we estimate information in parts of the environment not sufficiently covered by sensing devices?*

In the state of the art, this question has found multiple answers by applying standard techniques for estimating missing information. Nevertheless, the state of the art techniques do not allow to address properties such as openness, large scale and heterogeneity that the proposed solution does.

The thesis has been developed starting from a solution based on standard techniques including Voronoi tesselation and hierarchical clustering. This solution showed that it is possible to group, through a spatial criterion, sensors that perceive information whose evolutionary dynamics are similar and that can be used to estimate missing information in parts of the environment where physical sensors are not present.

Then we defined a technique to estimate missing information based on a multi-agent approach: through this approach, sensing devices are coupled with agents that perform autonomous computation in a limited part of the environment so that the technique can be deployed on large scale environments. In this solution, we have introduced the concept of confidence zones, which identify the local part where agents operate. Confidence zones are similar to the regions of Voronoi developed in the first technique; unlike the latter, the confidence zones are modified by the agents in such a way as to include only the sensors that contribute in a pertinent way to the estimation of missing values (that is, avoiding outliers). Agents continuously influence the shape of the confidence zones through their perceptions; this causes the confidence zones to change their shape at each estimation. We say that the confidence zones are permanently dynamic because they are also modified in case (*i*) agents whose position is not fixed and (*ii*) agents entering and leaving the system at any time. Also,

the confidence zones can overlap and are calculated autonomously by the agents. We compared the results of this proposal to those obtained by the method based on Voronoi and hierarchical clustering, but also to other standard techniques available in the KNIME software. What we have emphasized in this comparison is both the quality of the results and the capability of the proposed solution to calculate estimates addressing the properties of large scale and openness. The operation of sensing devices, therefore the operation of the coupled agents, does not affect the other agents. Consequently, the system can be deployed on a large scale without any particular configuration. Moreover, agents can enter or leave the system without any particular configuration.

The second estimation scheme responds to a different case:

*how can we estimate missing information in case sensing devices are not available in the proximity of an agent involved in a cooperative resolution process?*

In this case, we developed a simple learning mechanism related to agents. Agents observe the environment and assemble *Ambient Context Windows* (ACW) that describe the evolution of the environment in discrete time intervals. The agents that must estimate missing information reason on the ACWs related to the information perceived previously and deduce what is the numerical difference that the last perceived information has with respect to the information to estimate.

The last contribution goes beyond the estimation using sensors perceiving the same type of information:

*how can we estimate a certain type of information (for example temperature) if there are no agents nearby that provide information of the same type? Assuming that there are agents that perceive information of different types or different units, is it possible to design a mechanism of cooperation through which the use of heterogeneous information can contribute to the estimation of missing values?*

The proposed solution, based on a multi-agent approach, answer this question through the integration of heterogeneous ACWs assembled by agents involved in a cooperative resolution process. If an agent must estimate missing information and there are no agents that perceive information of the same type in the confidence zone, the agent cooperates with agents that are beyond the confidence zone. Each agent involved in the cooperative process compares its ACWs and the variations of its information to provide the agent that must estimate the information with an indication of how much the estimated information varies from the last perceived information.

Agents adapt instantly to the environmental context in which they are situated so as to choose the most pertinent estimation strategy:

- if there are no agents in their proximity, the estimates are calculated using the information in the available ACWs in the history of the agents;
- if there are agents who perceive homogeneous information within the confidence zone, the cooperation between agents allows estimating the missing information;

- if there are no agents in the confidence zone, a cooperative resolution process takes place between agents perceiving heterogeneous information.

## Deployment in Urban Context

Today smart devices capable of sensing the surrounding environment are becoming more and more affordable; low prices lead to an increasing diffusion of these devices and consequently to a rise in their production. The diffusion of harmful gases for both the environment and citizens arises from the massive production of these devices [131]. Using already available devices such as smartphones or connected vehicles could limit the number of sensors to deploy in an urban context, lowering the demand for new devices and thus their production. The use of these available devices should not replace existing sensors; sensing infrastructure should be capable of integrating smartphones and connected vehicles with existing sensors to make the urban sensing participatory. This integration is advantageous for the sustainable development of cities: sensing infrastructures integrate a large number of devices capable of perceiving geo-localized information on a large-scale that allow for a local and precise description of the environmental dynamics.

Integrating heterogeneous information can be beneficial to experts for defining new services for citizens or improving existing ones to achieve the smartness objective of a city. Nevertheless, despite the limited number of devices capable of sensing the environment available on the market, we assume that the low cost and miniaturization can lead to the integration of sensors into devices such as smartphones or connected vehicles. However, the deployment of HybridIoT in a real context, considering data acquired from existing sensors and mobile devices such as smartphones, has no consequence on the nominal operation of the system:

- it operates regardless of the number of devices, these can enter or leave the system without any external intervention;
- delays in the transmission of information between devices do not affect the functioning of HybridIoT: agents evaluate estimates using the available information which is geo-localized and provided with time-stamps;
- mobility does not affect the functioning of HybridIoT: agents perform a local computation in the part of the environment where they are situated.

In the experiments conducted, the intermittence of sensors enables simulating both openness and mobility properties. Each ACA has a boolean variable which value is $1$ if the associated sensor is turned on, $0$ otherwise. By controlling this variable during the agent's operation is possible to simulate the intermittence of the associated sensor, not affecting the functioning of the entire system. In cross-validation, the value is set to $0$ when evaluating the estimates for a given test partition. In

using this variable, it is possible to simulate the malfunctioning of devices, therefore assessing that the system addresses the property of openness as agents cooperate only with available ACAs.

Currently, we plan to deploy the proposed system on the campus of the University of Toulouse III – Paul Sabatier and to use devices and smartphones for collecting information. The following factors are crucial for a reliable implementation of the system in a real context:

- developing an application for mobile devices. Different operating systems must be considered and therefore the effort could be considerable. The mobile application must ensure the privacy of users when gathering and treating information perceived by personal devices. Users should also agree to use the designed application and to take part in the experiments;
- implementing a communication protocol between devices. In a personal device, several applications can degrade the device performance. This can be overcome by using lightweight application protocols that improve smartphone performance in terms of bandwidth consumption, battery lifetime, and communication latency [36];
- managing communication delays due to the transmission of remote information: the communication on smartphones is bandwidth-limited and relatively expensive, especially when access to the network is achieved via cellular connection [43];
- verifying that the data acquired by the devices are correct and that there are no anomalies that could prevent the proper functioning of the system. In this way, having data been perceived from devices functioning properly, it is possible to validate the results obtained by the proposed technique;
- building a consistent database for evaluating the results obtained from the system deployed in a real context: this requires continuous monitoring of the environment from both *ad hoc* devices and smartphones. However, smartphones functioning depends on the will of its owner: the device is turned on/off according to non-controllable patterns, therefore it is not possible to exert strict control [43].

These factors are independent of the functioning of the system, have no direct consequences in its operation and parameters.

## Contribution

### Contribution to IoT

The IoT is an indispensable component in the development of sustainable and technological advances. The development of smart objects having sensing, communication and actuating capabilities have seen an accelerated growth. Such network-enabled smart objects have numerous applications in the areas such as environment monitoring [125].

Our contribution is an important advance in IoT technology: it is possible to couple IoT devices with cooperation mechanisms that allow them to operate no longer independently but jointly, thus contributing to the achievement of smartness and sustainable development goals.

The cooperation mechanism discussed in this thesis allows compensating the action of a sensor that is temporarily not available or malfunctioning. The advantage of cooperation in IoT devices is twofold: (*i*) it can be implemented as a resilience technique for devices that can encounter unpredictable malfunctions and (*ii*) it is possible to reduce the number of sensors to deploy in the urban environment, thus reducing the maintenance and installation costs necessary for implementing sensors networks.

### Contribution to AMAS

This thesis proved that the AMAS approach is relevant for the development of distributed systems in the urban environment able to manage its complexity, continuous and unpredictable evolution.

Although the state of the art is rich in innovative and heterogeneous techniques that can be integrated into the urban environments, these solutions often have limitations in functionality or ability to adapt to the environmental context. The AMAS approach enabled us to develop a system for the estimation of missing environmental information by addressing the properties of openness, large scale and heterogeneity through a mechanism of cooperation between agents. These properties have not been addressed simultaneously by the state of the art methods.

The most relevant contribution to AMAS concerns the openness property. Until now the property of openness has not been treated accurately. In the proposed approach, the intermittence of sensing devices enables simulating the openness property. Moreover, the openness property allows including multiple sensing devices in the estimation process. This thesis has shown, through a concrete application, that it is possible to define AMAS in which the boundaries of the system are not specified *a priori* and in which sensing devices (opportunely agentified) can easily enter or leave the system without any particular configuration.

Weyns et al. [146] state the following (about openness): "*Living in an environment, perceiving it, and being affected by it intrinsically imply openness. Software systems are no longer isolated but become permeable subsystems, whose boundaries permit reciprocal side effects. The reciprocal influence between system and environment is often extreme and complex, making it difficult to identify clear boundaries between the system and its environment.*" In the same article, the authors show interest in the challenge of openness as this "*enables a system to adjust itself dynamically to uncertainty in the environment, tasks, and availability of resources.*". This thesis has shown, through the design of HybridIoT, that it is possible to design an AMAS to addresses the property of openness, therefore integrating itself in environments with a high degree of uncertainty and continuously evolving.

## Perspectives

HybridIoT opens a wide range of opportunities in different applications.

An emerging challenge in *Intelligent Transport Systems* (ITSs) is the exploitation of data collected from a large amount of mobile and fixed devices. In large-scale environments, it is necessary to use technological means to identify patterns in mobility, how individuals interact between them and with the environment. The continuous development of cities must face today issues related to congestions and high rates of accidents. Through the integration of sensors, vehicles can collect a large quantity of *Floating Car Data* (FCD) which refers to the collection of vehicle position and kinematics data (e.g., speed, direction of travel). These data are used to get traffic information through ITS applications [71]. To monitor the environment, urban sensing approaches can be applied to a fleet of connected vehicles [10, 90] using them as mobile sensors. The proposed method uses a method based on HybridIoT to map the environment through FCD. At the time this thesis is written, a paper discussing a proposition of a solution to address this challenge is under review in an international journal.

Another application of HybridIoT takes place in the smart building context. Smart buildings, like smart cities, are truly complex systems [98]. Common applications of smart buildings involve energy management, modelling of occupant behaviour, and user comfort [130]. HybridIoT can be used to measure the correlations between sensors to provide experts with automatic information about the building state. Studying how sensors are correlated or uncorrelated allows to detect variations of behaviour in the building that might be indicative of an anomaly such as a malfunctioning air conditioning system, or a window that remained open during the night [70]. Because those anomalies can occur unexpectedly, the correlations must be processed in real-time (i.e. within fixed time constraints). At the time this thesis is written, a paper discussing a solution to address this challenge has been submitted to an international conference.

The work carried out in this thesis aimed at defining a technique for estimating missing information by integrating heterogeneous information. The perspectives in the context of explicability are the following:

- (short term) to deduce (or better, explain) the state of environmental resources (doors, windows, etc.) without using *a priori* semantic knowledge, based only on information perceived and eventually estimated by agents.
- (long term) use the information deduced by the agents about the state of environmental resources to recommend ”eco-friendly” behaviors and gestures to users, to optimize the consumption of environmental resources.

# Part IV

# Appendices

Appendix A

# Classical Methods for Estimating Missing Information

THE estimation of missing information has undergone constant evolution over the years due to the development of mathematical techniques and AI. The use of both mathematical tools and AI enables the design of effective systems for the estimation of missing information.

When designing a system for estimating missing information, it is important to choose the right estimation technique: each one has its advantages and disadvantages: neural networks are powerful tools with learning capabilities, they need reconfiguration and a variable number of parameters that depend on the application context; regression techniques are fast but have no learning capabilities. Moreover, for the design of such systems on a large scale, it is necessary to choose the technique based on several factors: the devices and information, the mobility of the sensors, the possibility of the system reconfiguration. There is no ”best” technique to use, no technique is better than the others. The system designer must eventually choose according to the needs that arise from the application context.

The estimation techniques are usually classified as univariate and multivariate: the first type manipulates datasets where only one variable is present, using only non-missing values in the available dimension; multivariate analysis uses multiple variables in the feature dimensions to estimate the missing values. **The solution proposed in this thesis is multivariate, therefore capable of using variables, in different dimensions, to estimate a missing value of a certain type.**

This appendix presents the main techniques for estimating missing information. These techniques are divided in two categories: mathematical and AI techniques. The rest of the chapter is organized as follows: section A.1 introduces the main mathematical tools for estimating missing information. Section A.2 presents the main AI-based tools for estimating missing information.

## A.1 Mathematical Techniques

### A.1.1 Mean Estimation

Mean estimation is the most simple approach to estimate missing data. Given a dataset of information, missing data are estimated by calculating the mean of the existing values. Estimating values at the center of the distribution reduces the variability of the data, therefore the standard deviation and the variance of the resulting dataset are attenuated [41].

Mean imputation is a fast and simple fix for the missing data. However, it will underestimate the variance and disturb the relations between variables: correlations and covariances are attenuated by mean estimation because the data is enriched with information that is uncorrelated with other variables in the dataset [41]. Therefore, the mean estimation technique should be used only as a rapid fix when a handful of values are missing, and it should be avoided in general [136].

Figure A.1 shows the plot of a sequence of values and the estimated sequence using the mean.

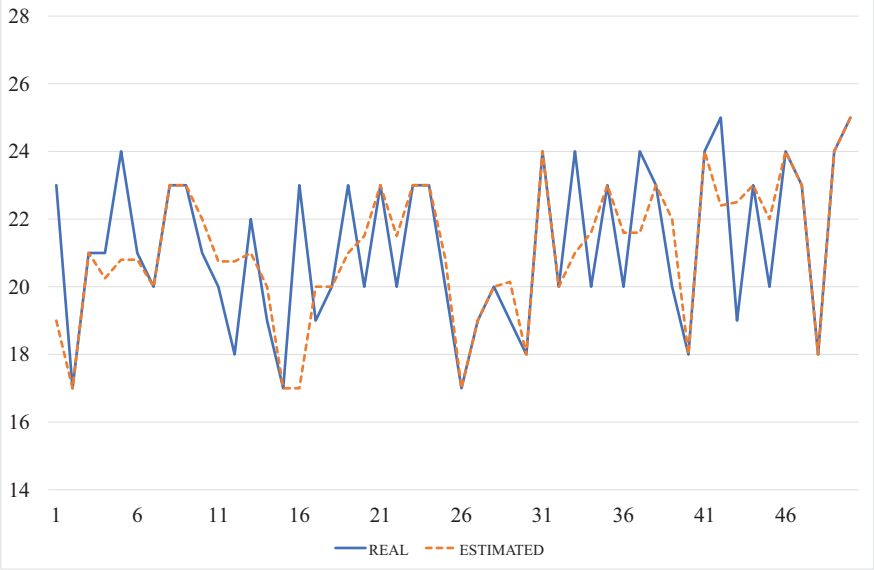



Figure A.1: A series of data and its estimate. The latter have been obtained by estimating the $50\%$ of points by a moving average with a window size of $10$ samples.

Calculating a missing value by averaging does not consider all the available information, but only the data close to the missing value to provide a more accurate estimate as close as possible to the real one. In Figure A.1, a window containing $10$ information is used to estimate the missing values using a limited number of samples. This technique is known as *moving average* and considers a subset of information to estimate a missing sample.

### A.1.2 Regression Estimation

Regression analysis is a mathematical tool used to describes the relationship between independent and dependent variables. Regression analysis allows to estimate missing data by using models constructed from the available information. With respect to the mean imputation, the regression imputation has the advantage of preserving the shape of data distribution.

Usually, performing regression analysis consists of two steps: the first step involves building a model from the observed data. Estimates for the incomplete or missing information are calculated under the fitted model [136]; the second step is to calculate estimates for the missing variables using the regression models. The estimates for the missing information are calculated under the fitted model. The variable with missing data is used as the dependent variable, whereas the best predictors are selected as independent variables.

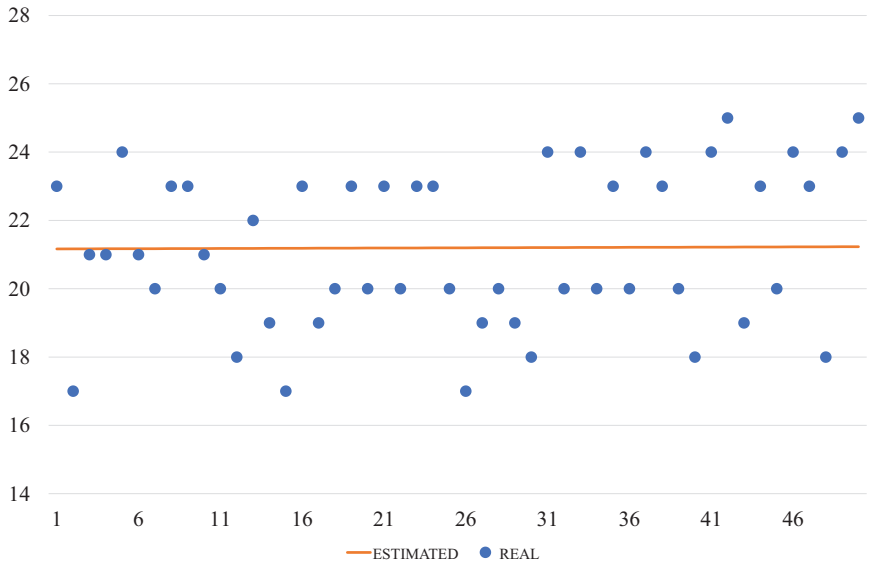



Figure A.2: The line describing the trend of the sample points have been obtained through a linear regression model.

Consider the set of points shown in Figure A.2. A regression model is defined as *linear* if it is possible to define a line such that the distance between the points and the line is minimized. A linear regression model is defined as follows:

$$Y_i = \beta_0 + \beta_1 X_i + u_i \tag{A.1}$$

where:

- $i$ is the index of observations, $i = 1, \ldots, n$
- $Y_i$ is the dependent variable;
- $X_i$ is the independent variable (or regressor);

- $\beta_0 + \beta_1 X$ is the regression line;
- $\beta_0$ is the intercept of the regression line of the population;
- $\beta_1$ is the angular coefficient of the population regression line;
- $u_i$ is the statistical error.

Calculating a linear regression model through the equation (A.1) is to estimate missing values. However, if the data show significant variability, a linear model may be insufficient to produce accurate estimates for missing values. In this case, regression models can be extended using polynomial functions. Polynomial regression is an extension of the linear regression that uses polynomial-based model to describe the relations between samples. The polynomial regression results in a curve that fits the samples. Figure A.3 shows an example of curve obtained by a polynomial of degree $4$.

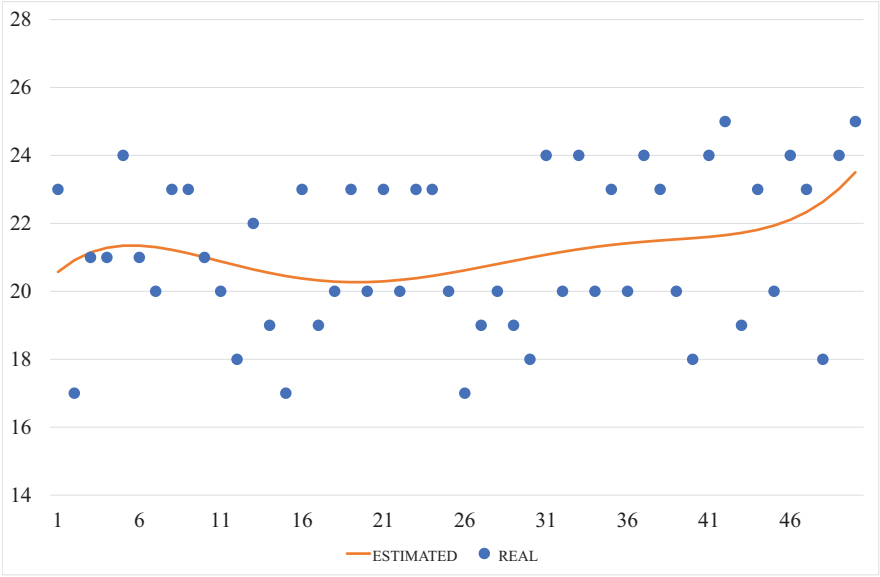



Figure A.3: A curve describing the trend of the sample points, obtained through a polynomial regression model.

The estimation of missing data depends strongly on the used regression model. Some regression imputation models begin by calculating estimates of the mean vector and covariance matrix of the data from the sub-matrix from data with no missing values [92]. However, estimated information can fit the regression model but vary considerably from the expected real value. Also, estimating information through regression affects the correlation. Consider for example the data shown in Figure A.2: if, on the one hand, the data that lay on the straight line have correlation $1$, combining them with the other data will increase the correlation.

### A.1.3 Discussion

The mean estimation and the regression analysis presented in the previous section are two of the simplest methods for estimating missing values. Their description, although simplistic and not exhaustive, introduces the reader to concrete solutions to solve the problem of estimating missing values. Nevertheless, it is beyond the objectives of this thesis to discuss exhaustively all the developments concerning, above all, regression analysis. However, it is necessary to underline that over the last few years several models based on regression analysis have been proposed that allow predicting accurately missing values [143, 89, 86].

The next section introduces some AI-based techniques that can be used for estimating missing information.

## A.2 Artificial Intelligence Techniques

### A.2.1 Neural Networks

McCulloch and Pitts described a model of the neural activity that could be carried out by machines. This model is based on the use of a simple thresholding function which influences a "fire" activity of the neuron [95]. Let $f(\cdot)$ be the function that represents a neuron's behavior (expressed as a mathematical function) and $\beta$ the "activity level" of the neuron; the activation function of a neuron can be formalized as follows:

$$f(\cdot) = \begin{cases} 1 & \text{if } \beta \geq 0 \\ 0 & \text{if } \beta < 0 \end{cases}$$

A multitude of neurons can be grouped to form a properly connected network. Neural networks are particularly useful for solving problems that cannot be expressed as a series of steps, such as recognizing patterns, classification, series prediction, and data mining [66]. Therefore, neural networks are usually employed in application domains where are required computational mechanisms that are as similar as possible to the activity of the human brain.

In a neural network, the input of a neuron is the result of the weighted sum of the output activities passed along from other neurons. Each neuron is connected to other neurons by "synapses" that can amplify or diminish the signal that is being forwarded [95].

In real applications, neural networks can be made up of hundreds of neurons, grouped by "levels" or "layers", each one containing a certain number of neurons connected to the neurons of the next level. Levels are generally grouped into three categories: input level, output level and hidden level. Each level contains a collection of neurons collected to the next level in the network and uses a mathematical function to transform the input values and pass the output to the next level.

The learning ability of neural networks was introduced in 1986 by Rumelhart et al. [117], who

introduced the *back-propagation* learning procedure. The idea behind the back-propagation is that the network automatically adjusts the weights of the connections between neurons so that the difference between the expected result and the output result of the network is minimized. Thanks to the back-propagation algorithm, a network composed of multiple levels can perform complex tasks that require learning capacities. Each layer in a network applies a given function of the output of the previous layer to perform a specific task. For example, a neural network for the recognition of human faces could have one level for edges recognition, one level for identifying the eyes, the next one for identifying the shape of the lips, etc. The idea of having multiple layers consists of dividing a complex task into several sub-problems, each of which is performed by a given layer of the network.

Different types of neural networks have been proposed in the last years, each one with unique features that make them effective in different application domains. The next section presents a particular architecture of neural networks known as Radial Basis Function neural Network (RBFN), which can be applied for the estimation of missing information.

### A.2.2 Radial Basis Function Networks

*Radial Basis Function Networks* (RBFNs) are a particular family of neural networks composed of two levels of neurons. The main feature of RBFNs is the neurons' activation function. The input of a neuron in the hidden level is the sum of the distance between the input pattern and the "center" of the basis function, usually a Gaussian function. The output of a RBFN is a scalar function of the input vector [154]. Figure A.4 shows the basic structure of a RBFN. The input vector is an $n$-dimensional vector that the network has to classify, and the relative weights are presented to the RBF neurons. The second is a hidden layer consisting of several RBF non-linear activation units, which functions are conventionally implemented as Gaussian functions [45]. The output scores given by the output nodes are computed by calculating a weighted sum of the activation values of the RBF neurons.

RBFN are effective for regression and function approximation [87]. In particular, RBFNs can be used to determine the underlying trend of an environmental series of information and accurately estimate missing values.

### A.2.3 Soft-Sensor

Soft-sensor or virtual sensor is a common name for a software where several measurements are processed together. The interaction of the signals can be used for calculating or to estimate new quantities that cannot be measured. A virtual sensor is a conceptual device whose output or inferred variable can be modeled in terms of other parameters that are relevant to the same process [57].

Soft-sensor is often used to overcome the problem of on-line estimation of process variables by integrating AI techniques such as Artificial Neural Networks (ANN). ANNs have been shown to

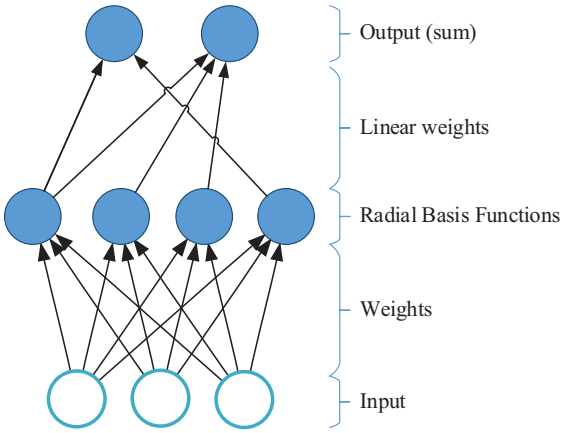



Figure A.4: The structure of a RBFN.

be an adequate choice because they can receive real-time readings of several process variables as well as feedback signals of downstream on-line analyzers for the target property. Once trained, this virtual sensor uses real time measurements of the selected process variables to infer the missing value [57].

### A.2.4 Fuzzy Rules

Fuzzy rules define hyper-rectangles that act as intervals in high dimensional spaces, used to define a degree of membership of input data. For each new pattern, a learning algorithm checks if the data is covered by existing hyper-rectangles, conversely a new fuzzy rule is created to define the class containing the given data. If a new data is incorrectly covered by an existing fuzzy rule, the fuzzy point's support-region is reduced so that the conflict is avoided. [127]. Then a *fuzzy classification* process computes a degree of matching for each sample and a corresponding input pattern by using the fuzzy rules. Different membership functions can be employed when dealing with numerical datasets [14].

Different prediction models based on fuzzy logic have been made for predicting environmental information [114, 111]. In these solutions each observation is analyzed subsequently and a set of *fuzzy rules* is inserted or modified accordingly.

### A.2.5 Random Forest Regression

Regression trees are a popular class of machine learning algorithms. They are robust against outliers and are flexible enough to fit interactions and non linear relations between data [136]. Regression trees consist of a series of data splitting rules, starting at the top of the tree. The construction of a regression tree involves two steps:

1. divide the set of values $X_1, X_2, ..., X_p$ into $J$ distinct and non-overlapping regions $R_1, R_2, ..., R_J$,

that could have any shape;

2. a new estimate is calculated for every observation that falls into the region $R_j$. The estimate is calculated as the mean of the observations in the $j$-th region $R_j$.

Considering every possible partition of the feature space in $J$ regions is computationally expensive. For this reason, a top-down, greedy approach is usually used, which splits each region into two separate regions, starting from the one containing all the observation [73]. The criterion used to separate a region is based on the minimization of RSS (Residual Sum of Squares), a measure of the difference between the data and an estimation model, defined as follow:

$$RSS = \sum_{j=1}^{J} \sum_{i \in R_j} (y_i - \hat{y}_{R_j})^2 \tag{A.2}$$

where $\hat{y}_{R_j}$ is the mean response for the training observations within the $j$-th box. The process continues until a stopping criterion is reached (for example, the algorithm stops until no region contains more than five samples). Once all the regions $\{R_1, ..., R_J\}$ have been created, the response for missing data is calculated using the mean of the training observations in the region to which that test observation belongs [73].

Figure A.5 shows an example of subdivision of a set of observations. At first, a tree is created by splitting the data according to their features. Then, the regions are created according to the branches of the regression tree.

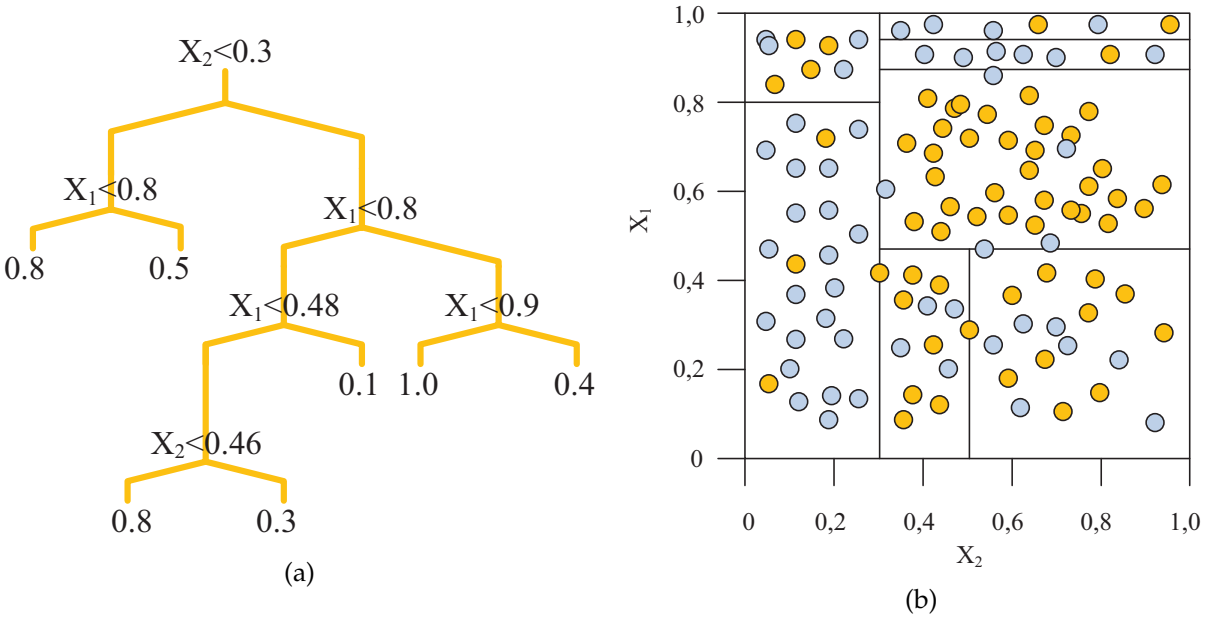



Figure A.5: A regression tree (Figure A.5a) and the corresponding regions (Figure A.5b).

### A.2.6 Gradient Boost Decision Trees

Boosting assumes the availability of a base or weak learning algorithm which, given labeled training examples, produces a base or weak classifier. The goal of boosting is to improve the performance of the weak learning algorithm while treating it as a "black box" which can be called repeatedly, like a subroutine, but whose inner learning algorithms cannot be observed or manipulated [121].

A boosting algorithm takes as input a set of training examples $(x_1, y_1), ..., (x_m, y_m)$ where each $x_i$ is an instance from $X$, and each $y_i$ is the associated label or class. In the simplest case there are only two classes, $-1$ and $+1$.

A boosting algorithm learns from the data by calling multiple time the base learning algorithm. However, if the base learner is simply called repeatedly with the same set of training data, we cannot expect anything interesting to happen; instead, we expect the same, or nearly the same, base classifier to be produced over and over again, so that little is gained over running the base learner just once. Therefore, to improve the base learning, the boosting algorithm must in some way manipulate the data that it feeds to it [121].

The key idea behind boosting is to choose training sets for the base learner in such a fashion as to force it to infer something new about the data each time it is called. This can be accomplished by choosing training sets on which we can reasonably expect poor performance using the base learning algorithm. If this can be accomplished, then we can expect the base learner to output a new base classifier which is significantly different from its predecessors. This is because, although we think of the base learner as a weak and mediocre learning algorithm, we nevertheless expect it to output classifiers that make nontrivial predictions [121].

Boosting algorithms thus train different models in a sequential manner which result in a decision tree, each one modified sequentially to best fit the input data. Figure A.6 shows an illustration presenting the idea behind the boosting algorithm.

Gradient boosting is a particular version of the boosting techniques that uses the gradient of the loss function to calculate the weights to be given as input to each weak learner when building each new model. Gradient boosting is particularly suitable for regression and classification problems: in this context, we want to predict values so to minimize a loss or error function. By using gradient descent and updating our predictions based on a learning rate, we can find the values where error is minimum.

The gradient boosting technique allows to avoid misclassification and obtain accurate results. The gradient boosting allows to improve the machine learning algorithm by running it sequentially on the observations and modifying the weights in an appropriate way to improve the performance. In other words, at each trial the predictor learns from the mistake of the previous predictors.

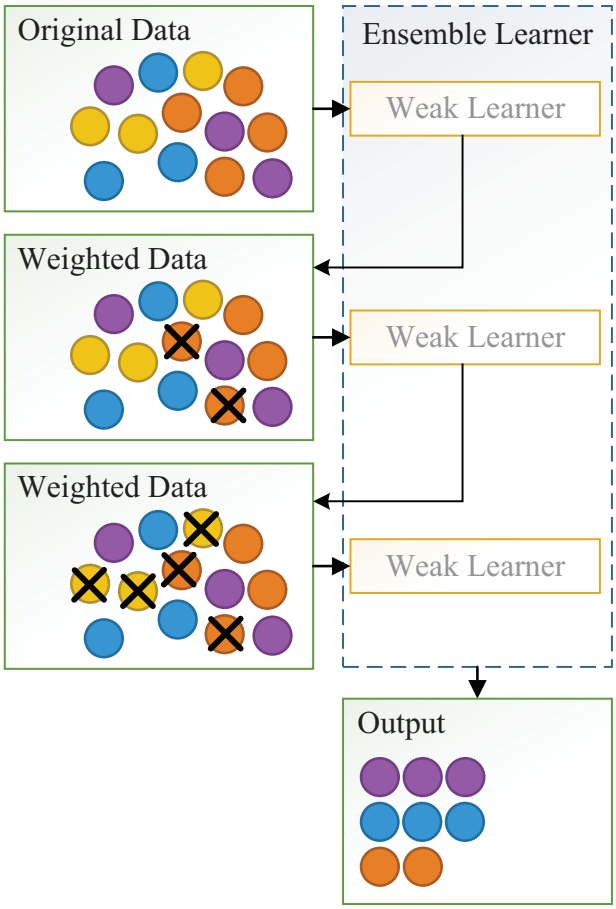



Figure A.6: The boosting technique trains many weak models in a gradual and sequential manner, which result in a strong learner, capable of providing precise classification results.

Appendix B

# The ADELFE Methodology

TODAY several agent-oriented methodologies exist. Some of the most recognized methodologies are ASPECS [30], PASSI [29], TROPOS [19], GAIA [153], Prometheus [148] and so on. However, these methodologies are not suited to handle complexity, dynamics, openness, or software adaptation. To respect these properties, specifics to the development of AMAS, the ADELFE methodology has been proposed. ADELFE is the french acronym for "*Atelier de Développement de Logiciel à Fonctionnalité Emergente*" which can be translated by *Toolkit for Designing Software with Emergent Functionalities*.

ADELFE is based on the *Rational Unified Process* (RUP) and includes five *Work Definitions* (WD) [17]:

- **WD1** - Preliminary requirements;
- **WD2** - Final requirements;
- **WD3** - Analysis;
- **WD4** - Design;
- **WD5** - Implementation.

## B.1 WD1 - Preliminary Requirements

This phase represents a description of specifications between customers, users and designers. The result of this phase is a document containing a precise description of the problem without using any particular modeling language. In this phase no specific modelling language is used. Four roles are involved at this stage [17]:

- Final User: he is responsible for defining both functional and non-functional requirements list, used to define the system and its environment;
- Client: his main role is to validate product documents drawn up by experts approving the requirements;
- Software Analyst: his role consists in giving a definition for the main concepts used to describe the system and its environment;
- Business Analyst: his role consists in defining the business concept and the relationships between them. He also describes formally what are the business activities, what are the products provided or required and who are the responsible persons of these activities.

The Preliminary Requirements Phase generates five work products (text document including textual description and/or diagrams) which are listed in Table B.1.

Table B.1: The five kind of work products from WD1 phase [17].

| **Name** | **Description** | **Work Product Kind** |
|---|---|---|
| Users Requirements Set | Textual description of the functional and non functional requirements | Free Text |
| Consensual Requirement Set | Textual description composed of consensual requirements | Free Text |
| Business Model | A document composed of: 1) a diagram modelling the domain-specific data structure; 2) a diagram showing the workflow of activities performed by thebusiness actors | Composite (Structural and Behavioural) |
| Glossary | A glossary of terms | Free Text |
| Constraints Set | Textual description composed of the limits and constraints of the system | Free Text |

## B.2 WD2 - Final requirements

In this work definition, the preliminary requirements are transformed to use cases. Also, this work definition characterize the environment of the system, by identifying the entities that interact with the system and the constraints on these interactions. The Business Analyst gives a detailed description of the system environment: he is responsible of use cases identification drawing diagrams that represent the interactions between actors and the system. After identifying the use cases, the client is responsible for validate product documents and approving the use cases. Then, MAS specialist verify the pursues three activities that consist in (1) the characterization of the system environment according to Russel and Norvig definition [118], (2) the identification of the possible ”bad” interactions between the actors and the system, (3) the analysis of the previous results to justify the MAS use [17].

The verification of the adequacy of MAS is crucial for the ADELFE methodology: because not all the applications require the MAS approach for their realization, it is necessary to answer

the question *"a traditional (Object-Oriented) approach sufficient to solve the problem or has the problem some characteristics which implies MAS approach for the solving?"*. Figure B.1 shows the flow of tasks necessary to verify the MAS adequacy.

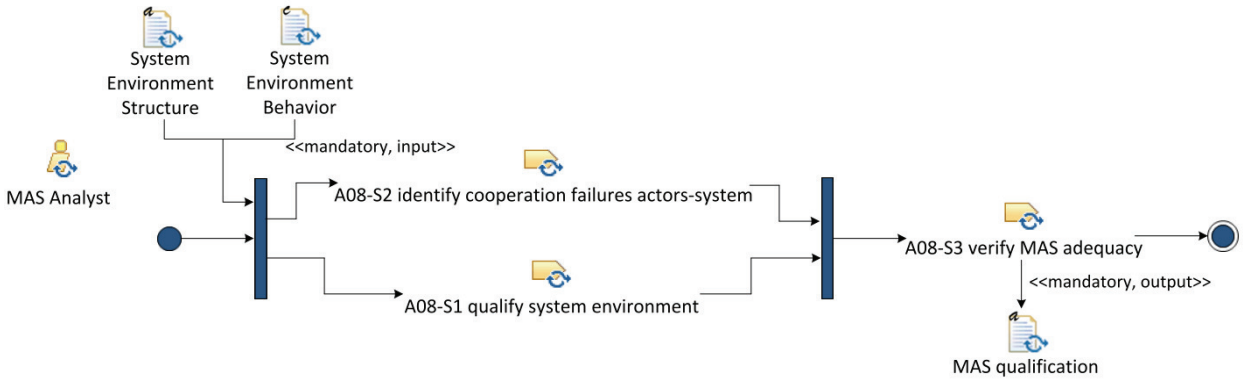



Figure B.1: Flow of tasks necessary to verify the MAS adequacy according to the ADELFE methodology.

Table B.2 reports the five types of work products resulting from phase WD2.

Table B.2: The four kind of work products from WD2 phase [17].

| Name | Description | Work Product Kind |
|---|---|---|
| System Environment Structure | A textual description describing the actors which interact with the system and the possible constraints. Moreover, this document contains a brief text of actors description. | Free Text |
| System Environment Behavior | A document composed of: 1) a use case diagram representing actors and the functionalities assigned to them; 2) a structured text description of the actors; 3) diagrams representing the interactions between the actors and the system | Composite (Structural and Behavioural and Free text) |
| MAS qualification | A text document composed by the description of the environment according the Russel and Norvig definition, the description of "bad" interaction between actors and system and the justification of an implementation that using a MAS is needed | Free Text |
| UI Prototype | This document is composed by the GUIs description through which the user(s) interact with the system and the links between the GUIs. | Free Text |

## B.3 WD3 - Analysis

The Analysis phase aims at identifying the system structure and justifying the AMAS adequacy. This phase enables to analyse the domain characteristics, determine the agents and validate an AMAS approach at the global and local level. Two roles are involved in the Analysis Phase [17]:

- MAS Analyst: his role consists in identifying the entities of the system and identifying their interactions. Also, his role consists in identifying the agents, their autonomy, goals and negotiation abilities.

- AMAS Analyst: his role consists in dealing with the AMAS principles; this concerns the identification of (cooperative) agents, how they determine the cooperation failures that can occur between entities and then to define the agents regarding the results of previous steps.

The entities of an AMAS are differentiated into two categories [17]:

- **Active entity**: an active entity is given behavioural autonomy, allowing it to change state without necessarily interacting with another entity. An active entity can interact with other entities, possibly through communication mediums;
- **Passive entity**: passive entities are associated to resources or data. Passive entities have no behaviors, they can only be perceived and eventually modified by active entities. A state transition can only be the result of an interaction with another system component.

The two main activities of this phase consist in veryfing the adequacy of AMAS at global and local level [17]:

- Global level: In this activity, the adequacy at the global level is studied to answer the question ”is an AMAS required to implement the system?”. This is done throw several simple questions related the global level. The ADELFE method provides a tool based on a questionnaire to assess the adequacy of the AMAS approach to our problem (Figure B.2). This adequacy is studied both at the system and local level (the level of the different parts composing the system) [12].

  At global level, the following criteria are used to verify the adequecy of AMAS [13]:

  1. Is the global task incompletely specified? Is an algorithm a priori unknown?
  2. Is the correlated activity of several entities needed to solve the problem?
  3. Is the solution generally obtained by repetitive tests? Are different attempts required before finding a solution?
  4. Can the system environment evolve? Is it dynamic?
  5. Is the system functionally or physically distributed? Are several physically distributed components needed to solve the global task? Or is a conceptual distribution needed?
  6. Does a great number of components needed?
  7. Is the studied system non-linear?
  8. Finally, is the system evolutionary or open? Can new components appear or disappear dynamically?

- Local level: In this activity, the AMAS adequacy is studied at the local level to determine the agents that need to be implemented as AMAS. And at the component level, three more criteria are used [13]:

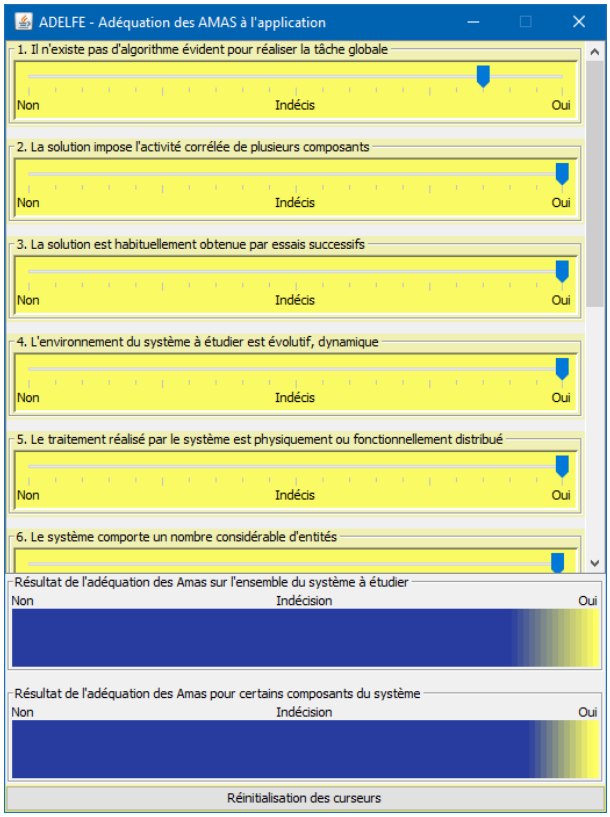



Figure B.2: ADELFE's tool used to assess the pertinence of the AMAS approach for the HybridIoT system.

1. Does a component have only a limited rationality?
2. Is a component "big" or not? Is it able to do many actions, to reason a lot? Does it need significant abilities to perform its own task?
3. Can the behaviour of a component evolve? Does it need to adapt to the changes of its environment?

Table B.3 reports the four work products generated from the Analysis phase.

## B.4 WD4 - Design

In this phase are defined the interaction protocols between agents as well as their behaviors (nominal and cooperative). This activity results in a complete characterization of the MAS. Three roles are involved in the Design Phase [17]:

- Architectural Designer: his role consists in defining a precise architecture of the systems in terms of modules

Table B.3: The four kind of work products from WD3 phase [17].

| Name | Description | Work Product Kind |
|---|---|---|
| System Analysis | A document composed of: 1) a textual description of the entities described as active or passive; 2) diagrams depicting the interactions between entities. | Composite (Free Text and Behavioural) |
| Global AMAS Adequacy Synthesis | This document stores the answers to the questions regarding the global level about an implementation using an AMAS. | Structured Text |
| Agent Extraction | This document supplements the System Analysis document with: 1) the definition of the goal, the study of autonomy and the negotiation abilities for each active entity; 2) the list of the cooperation failure interactions between entities or between entity and its environment; 3) the definition of the cooperative agent and the AMAS system diagram which represents them. | Composite (Free Text and Behavioural) |
| Local AMAS Adequacy Synthesis | This document completes the Global AMAS adequacy synthesis with the answers to the questions regarding the local level about an implementation using an AMAS. | Structured Text |

- MAS Designer: his role consists in defining how the entities and the agents interact together or with their own environment.
- AMAS Designer: his role consists in defining the cooperative behavior of entities, defining skills, aptitudes, an interaction language, a world representation, a criticality and the characteristics of the agents. Also, his goal consists in testing the behavior of the agents.

Table B.4 reports the six work products generated from the Design phase.

## B.5 WD5 - Implementation

This phase aims at providing the designed system. Two roles are involved in the Implementation Phase [17]:

- AMAS Framework Developer: his goal consists in describing the system architecture and implementing everything that is not an agent.
- AMAS Developer: his goal consists in implementing the agents' behavior, their nominal and cooperative behaviors.

Table B.5 reports the six work products generated from the Design phase.

Table B.4: The six kind of work products from WD4 phase [17].

| Name | Description | Work Product Kind |
|---|---|---|
| Module Organization | This document depicts the organization and the dependencies of the key elements of the software. | Structural |
| Communication Acts | This document is composed of the specific textual description of the entity interactions and the agent interactions and the precise diagrams depicting this. | Composite (Free and Structural and Behavioural) |
| MAS Environment | This document contains the description of the entities behaviour. It is illustrated with inner state related to their current role. | Composite (Free and Structural and Behavioural) |
| MAS Architecture | This document is composed of the agent nominal behaviour description, illustrated with inner state related to their current role and depicted by structural diagram of agent. Skills, aptitudes, an interaction language, a world representation and a criticality define an cooperative agent behaviour. Moreover, it contains the physical characteristics of the agent and its structural rules. | Composite (Free and Structural and Behavioural) |
| Cooperative MAS Architecture | This document contains the elements of a cooperative agent behaviour enabling to anticipate or detect and repair the non cooperative situations. A cooperative agent behaviour is composed of skills, aptitudes, an interaction language, a world representation and a criticality. | Composite (Free and Structural and Behavioural) |
| Software Architecture | This document is composed of the fast prototyping of the agent behaviour and the refinement of the Software architecture (entities), Software architecture (nominal) and Software architecture (cooperative) document. | Composite (Free and Structural and Behavioural) |

Table B.5: The two kind of work products from WD5 phase [17].

| Name | Description | Work Product Kind |
|---|---|---|
| Framework Code | This document is composed of: 1) a textual description of the architecture of the system; 2) the implementation of all what is not agent. | Composite (Structured Text and Free Text) |
| AMAS Code | This document is composed of the implementation of the cooperative agent behaviour (nominal behaviour and cooperative behaviour). | Composite (Structured Text and Free Text) |

# Glossary

- **MCS** – Mobile Crowd Sensing
- **IoT** – Internet Of Things
- **ICT** – Information and Communication Technology
- **GPS** – Global Positioning System
- **HVAC** – Heating, Ventilation and Air-Conditioning
- **AI** – Artificial Intelligence
- **AmI** – Ambient Intelligence
- **NCS** – Non Cooperative Situation
- **MAS** – Multi-Agent System
- **ADELFE** – Atelier de Développement de Logiciels à Fonctionnalité Emergente (toolkit to develop software with emergent functionality)
- **AMAS** – Adaptive Multi-Agent System
- **ROI** – Region Of Interest
- **ACA** Ambient Context Agent
- **RSA** Real Sensor Agent
- **ACW** Ambient Context Window

# Own Bibliography


[1] D. A. Guastella, V. Camps, and M.-P. Gleizes. Estimating Missing Environmental Information by Contextual Data Cooperation. In M. Baldoni, M. Dastani, B. Liao, Y. Sakurai, and R. Zalila Wenkstern, editors, *PRIMA 2019: Principles and Practice of Multi-Agent Systems*, pages 523–531. Springer, 2019. doi: 10.1007/978-3-030-33792-6_37.

[2] D. A. Guastella, V. Camps, and M.-P. Gleizes. Multi-agent Systems for Estimating Missing Information in Smart Cities. In *Proceedings of the 11th International Conference on Agents and Artificial Intelligence - Volume 2: ICAART*, pages 214–223. SciTePress, 2019. doi: 10.5220/0007381902140223.

[3] D. A. Guastella, V. Camps, and M.-P. Gleizes. A Cooperative Multi-Agent System for Crowd Sensing Based Estimation in Smart Cities. *IEEE Access*, 8:183051–183070, 2020. doi: 10.1109/ACCESS.2020.3028967.

[4] D. A. Guastella and C. Valenti. Estimating Missing Information by Cluster Analysis and Normalized Convolution. In *2018 IEEE 4th International Forum on Research and Technology for Society and Industry (RTSI)*, pages 1–6, 2018. doi: 10.1109/RTSI.2018.8548454.

# Bibliography


[5] P. K. Agarwal, J. Gurjar, A. K. Agarwal, and R. Birla. Application of artificial intelligence for development of intelligent transport system in smart cities. *International Journal of Transportation Engineering and Traffic System*, 1(2):20–30, 2015.

[6] E. Al Nuaimi, H. Al Neyadi, N. Mohamed, and J. Al-Jaroodi. Applications of big data to smart cities. *Journal of Internet Services and Applications*, 6(1):25, 2015.

[7] G. Albanese, M. Cipolla, and C. Valenti. Genetic Normalized Convolution. In *International Conference on Image Analysis and Processing*, volume 6978 of *Lecture Notes in Computer Science*, pages 670–679. Springer, 2011.

[8] A. Aliberti, F. M. Ugliotti, L. Bottaccioli, G. Cirrincione, A. Osello, E. Macii, E. Patti, and A. Acquaviva. Indoor air-temperature forecast for energy-efficient management in smart buildings. In *2018 IEEE International Conference on Environment and Electrical Engineering and 2018 IEEE Industrial and Commercial Power Systems Europe (EEEIC / I CPS Europe)*, pages 1–6, 2018.

[9] Z. Allam and Z. A. Dhunny. On big data, artificial intelligence and smart cities. *Cities*, 89:80 – 91, 2019.

[10] A. Anjomshoaa, S. Mora, P. Schmitt, and C. Ratti. Challenges of drive-by IoT sensing for smart cities: City scanner case study. In *Proceedings of the 2018 ACM International Joint Conference and 2018 International Symposium on Pervasive and Ubiquitous Computing and Wearable Computers - UbiComp '18*, pages 1112–1120. ACM, 2018.

[11] B. Baccarne, P. Mechant, and D. Schuurman. Empowered cities? an analysis of the structure and generated value of the smart city ghent. In R. P. Dameri and C. Rosenthal-Sabroux, editors, *Smart City: How to Create Public and Economic Value with High Technology in Urban Space*, pages 157–182. Springer, 2014.

[12] C. Bernon, V. Camps, M.-P. Gleizes, and G. Picard. Tools for self-organizing applications engineering. In G. Di Marzo Serugendo, A. Karageorgos, Omer F. Rana, and F. Zambonelli, editors, *Engineering Self-Organising Systems*, pages 283–298. Springer, 2004.

[13] C. Bernon, V. Camps, M.-P. Gleizes, and G. Picard. Engineering adaptive multi-agent systems. In *Agent-Oriented Methodologies*, pages 172–202. IGI Global, 2005.

[14] M. R. Berthold. Mixed fuzzy rule formation. *International Journal of Approximate Reasoning (Elsevier)*, 32(2):67–84, 2003.

[15] M. R. Berthold, N. Cebron, F. Dill, T. R. Gabriel, T. Kötter, T. Meinl, P. Ohl, K. Thiel, and B. Wiswedel. KNIME - the konstanz information miner: Version 2.0 and beyond. *ACM SIGKDD Explorations Newsletter*, 11(1):26–31, 2009. doi: 10.1145/1656274.1656280.

[16] M. R. Berthold and K.-P. Huber. Missing values and learning of fuzzy rules. *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, 06(2):171–178, 1998. doi: 10.1142/S021848859800015X.

[17] N. Bonjean, W. Mefteh, M. P. Gleizes, C. Maurel, and F. Migeon. ADELFE 2.0. In M. Cossentino, V. Hilaire, A. Molesini, and V. Seidita, editors, *Handbook on Agent-Oriented Design Processes*, pages 19–63. Springer, 2014.

[18] L. Braubach, A. Pokahr, D. Moldt, and W. Lamersdorf. Goal Representation for BDI Agent Systems. In R.H. Bordini, M. Dastani, J. Dix, and A. El Fallah Seghrouchni, editors, *Programming Multi-Agent Systems. ProMAS 2004*, pages 44–65. Springer, 2005.

[19] P. Bresciani, A. Perini, P. Giorgini, F. Giunchiglia, and J. Mylopoulos. Tropos: An agent-oriented software development methodology. *Autonomous Agents and Multi-Agent Systems*, 8(3):203–236, 2004.

[20] B. Bruegge and A. H. Dutoit. *Object-Oriented Software Engineering Using UML, Patterns, and Java*. Prentice Hall, 3rd edition, 2010.

[21] D. Capera. *Systèmes multi-agents adaptatifs pour la résolution de problèmes: Application à la conception de mécanismes*. Thèse de doctorat, Université Paul Sabatier, Toulouse, France, juin 2005.

[22] D. Capera, J.-P. George, M.-P. Gleizes, and P. Glize. The AMAS theory for complex problem solving based on self-organizing cooperative agents. In *WET ICE 2003. Proceedings. Twelfth IEEE International Workshops on Enabling Technologies: Infrastructure for Collaborative Enterprises, 2003.*, pages 383–388, 2003. ISSN: 1080-1383.

[23] E. Cavalcante, N. Cacho, F. Lopes, T. Batista, and F. Oquendo. Thinking smart cities as systems-of-systems: A perspective study. In *Proceedings of the 2nd International Workshop on Smart*, SmartCities '16, New York, NY, USA, 2016. ACM.

[24] X. Chen, J.-H. Jung, and A. Gelb. Finite Fourier Frame Approximation Using the Inverse Polynomial Reconstruction Method. *Journal of Scientific Computing*, pages 1–21, 2018.

[25] Giovanni Ciatto, M. I. Schumacher, A. Omicini, and D. Calvaresi. Agent-based explanations in AI: Towards an abstract framework. In D. Calvaresi, A. Najjar, M. Winikoff, and K. Främling, editors, *Explainable, Transparent Autonomous Agents and Multi-Agent Systems*, volume 12175, pages 3–20. Springer, 2020.

[26] F. Cicirelli, G. Fortino, A. Giordano, A. Guerrieri, G. Spezzano, and A. Vinci. On the design of smart homes: A framework for activity recognition in home environment. *Journal of Medical Systems*, 40(9):200, 2016.

[27] A. Cocchia. Smart and digital city: A systematic literature review. In R. P. Dameri and C. Rosenthal-Sabroux, editors, *Smart City: How to Create Public and Economic Value with High Technology in Urban Space*, pages 13–43. Springer, 2014.

[28] E. Cosgrave, T. Tryfonas, and T. Crick. The smart city from a public value perspective. In *ICT for Sustainability 2014 (ICT4S-14)*, pages 369–377. Atlantis Press, 2014.

[29] M. Cossentino. From requirements to code with PASSI methodology. In *Agent-Oriented Methodologies*, pages 79–106. IGI Global, 2012.

[30] M. Cossentino, N. Gaud, V. Hilaire, S. Galland, and A. Koukam. ASPECS: an agent-oriented software process for engineering complex systems: How to design agent societies under a holonic perspective. *Autonomous Agents and Multi-Agent Systems*, 20(2):260–304, 2010.

[31] R. P. Dameri. Searching for smart city definition: a comprehensive proposal. *International Journal of Computers & Technology*, 11(5):2544–2551, 2013.

[32] R. P. Dameri. *The Conceptual Idea of Smart City: University, Industry, and Government Vision*, pages 23–43. Springer, 2017.

[33] R. P. Dameri. Smart city definition, goals and performance. In *Smart City Implementation: Creating Economic and Public Value in Innovative Urban Systems*, pages 1–22. Springer, 2017.

[34] R. P. Dameri and C. Rosenthal-Sabroux. Smart city and value creation. In R. P. Dameri and C. Rosenthal-Sabroux, editors, *Smart City: How to Create Public and Economic Value with High Technology in Urban Space*, pages 1–12. Springer, 2014.

[35] D. Davies and D. Bouldin. A Cluster Separation Measure. *IEEE Trans. Pattern Anal. Mach. Intell.*, 1(2):224–227, 1979.

[36] N. De Caro, W. Colitti, K. Steenhaut, G. Mangino, and G. Reali. Comparison of two lightweight protocols for smartphone-based sensing. In *2013 IEEE 20th Symposium on Communications and Vehicular Technology in the Benelux (SCVT)*, pages 1–6, 2013. ISSN: 2373-0854.

[37] R. Der and G. Martius. *Self-Organization in Nature and Machines*, pages 9–21. Springer, 2012.

[38] G. Di Marzo Serugendo, M.-P. Gleizes, and A. Karageorgos. Self-organising systems. In G. Di Marzo Serugendo, M.-P. Gleizes, and A. Karageorgos, editors, *Self-organising Software*, pages 7–32. Springer, 2011.

[39] Dimensions. Dimensions.AI. Available at `https://app.dimensions.ai/discover/publication` (2020/06/22).

[40] J. Dutta, C. Chowdhury, S. Roy, A. I. Middya, and F. Gazi. Towards smart city: Sensing air quality in city based on opportunistic crowd-sensing. In *Proceedings of the 18th International Conference on Distributed Computing and Networking*, ICDCN '17, New York, NY, USA, 2017. ACM. doi: 10.1145/3007748.3018286.

[41] C. K. Enders. *Applied Missing Data Analysis*. Guildford Press, 2010.

[42] A. K. Ettinger, I. Chuine, B. I. Cook, J. S. Dukes, A. M. Ellison, M. R. Johnston, A. M. Panetta, C. R. Rollinson, Y. Vitasse, and E. M. Wolkovich. How do climate change experiments alter plot-scale climate? *Ecology Letters*, 22(4):748–763, 2019.

[43] A. Faggiani, E. Gregori, L. Lenzini, V. Luconi, and A. Vecchio. Smartphone-based crowdsourcing for network monitoring: Opportunities, challenges, and a case study. *IEEE Communications Magazine*, 52(1):106–113, 2014.

[44] X. Fang, S. Misra, G. Xue, and D. Yang. Smart grid — the new and improved power grid: A survey. *IEEE Communications Surveys Tutorials*, 14(4):944–980, 2012.

[45] H. Faris, I. Aljarah, and S. Mirjalili. Evolving radial basis function networks using moth–flame optimizer. In P. Samui, S. Sekhar, and V. E. Balas, editors, *Handbook of Neural Computation*, pages 537 – 550. Academic Press, 2017.

[46] J. Ferber. *Multi-agent systems: an introduction to distributed artificial intelligence*, volume 1. Addison-Wesley, 1999.

[47] G. Fortino, W. Russo, C. Savaglio, W. Shen, and M. Zhou. Agent-oriented cooperative smart objects: From IoT system design to implementation. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 48(11):1939–1956, 2018.

[48] A. Foucquier, S. Robert, F. Suard, L. Stéphan, and A. Jay. State of the art in building modelling and energy performances prediction: A review. *Renewable and Sustainable Energy Reviews*, 23:272–288, 2013.

[49] D. Freedman and P. Diaconis. On the histogram as a density estimator:l2 theory. *Zeitschrift für Wahrscheinlichkeitstheorie und Verwandte Gebiete (Springer)*, 57(4):453–476, 1981. doi: 10.1007/BF01025868.

[50] J. H. Friedman. Greedy function approximation: A gradient boosting machine. *The Annals of Statistics*, 29(5):1189–1232, 2001. doi: 10.1214/aos/1013203451.

[51] G. Gan, C. Ma, and J. Wu. *Data Clustering: Theory, Algorithms and Applications*. Statistics and Applied Probability. SIAM, 2007.

[52] A. Garro, M. Mühlhäuser, A. Tundis, S. Mariani, A. Omicini, and G. Vizzari. Intelligent agents and environment. In *Encyclopedia of Bioinformatics and Computational Biology*, pages 309–314. Elsevier, 2019.

[53] J.-P. Georgé, M.-P. Gleizes, and V. Camps. Cooperation. In G. Di Marzo Serugendo, M.-P. Gleizes, and A. Karageorgos, editors, *Self-organising Software: From Natural to Artificial Adaptation*, pages 193–226. Springer, 2011.

[54] J.-P. Georgé, M.-P. Gleizes, P. Glize, and C. Régis. Real-time Simulation for Flood Forecast: an Adaptive Multi-Agent System STAFF. In *Symposium on Adaptive Agents and Multi-Agent Systems (AISB 2003), University of Wales, Aberystwyth, 07/04/03-11/04/03*, pages 109–114. Society for the Study of Artificial Intelligence and the Simulation of Behaviour, 2003.

[55] M.-P. Gleizes, J. Boes, B. Lartigue, and F. Thiébolt. neOCampus: A demonstrator of connected, innovative, intelligent and sustainable campus. In G. De Pietro, L. Gallo, R. J. Howlett, and L. C. Jain, editors, *Intelligent Interactive Multimedia Systems and Services 2017*, volume 76, pages 482–491. Springer, 2018.

[56] M.-P. Gleizes, V. Camps, and P. Glize. A Theory of Emergent Computation based on Cooperative Self-organization for Adaptive Artificial Systems. In *Fourth European Congress of Systems Science, Valencia Spain, 20/09/99-24/09/99*, 1999.

[57] J.C.B. Gonzaga, L.A.C Meleiro, C. Kiang, and R. Maciel Filho. ANN-based soft-sensor for real-time process monitoring and control of an industrial polymerization process. *Computers & Chemical Engineering*, 33(1):43–49, 2009.

[58] D.G. Green, S. Sadedin, and T.G. Leishman. Self-organization. In *Encyclopedia of Ecology*, pages 3195–3203. Elsevier, 2008.

[59] L. Greengard and J.-Y. Lee. Accelerating the Nonuniform Fast Fourier Transform. *SIAM Review*, 46(3):443–454, 2004.

[60] J. Guerrero-Ibáñez, S. Zeadally, and J. Contreras-Castillo. Sensor technologies for intelligent transportation systems. *Sensors*, 18(4):1212, 2018.

[61] J. D. Hamilton. *Time Series Analysis*. Princeton University Press, 1994.

[62] A. Hamzi, M. Koudil, J.-P. Jamont, and M. Occello. Multi-agent architecture for the design of WSN applications. *Wireless Sensor Network*, 05(2):14–25, 2013.

[63] G. Hanrahan. *Artificial Neural Networks in Biological and Environmental Analysis*. Analytical Chemistry. CRC Press, 1 edition, 2017.

[64] D. Hasenfratz, O. Saukh, C. Walser, C. Hueglin, M. Fierz, T. Arn, J. Beutel, and L. Thiele. Deriving high-resolution urban air pollution maps using mobile sensor nodes. *Pervasive and Mobile Computing*, 16:268–285, 2015.

[65] I. A. T. Hashem, V. Chang, N. B. Anuar, K. Adewole, I. Yaqoob, A. Gani, E. Ahmed, and H. Chiroma. The role of big data in smart city. *International Journal of Information Management*, 36(5):748–758, 2016.

[66] J. Heaton. *Introduction to Neural Networks with Java*. Heaton Research, Inc., 2 edition, 2008.

[67] T. K. Ho. Random decision forests. In *Proceedings of 3rd International Conference on Document Analysis and Recognition*, volume 1, pages 278–282 vol.1, 1995. doi: 10.1109/ICDAR.1995.598994.

[68] A. Hodges. *Alan Turing: The Enigma*. Princeton University Press, 2014.

[69] A. R. Honarvar and A. Sami. Towards sustainable smart city by particulate matter prediction using urban big data, excluding expensive air pollution infrastructures. *Big Data Research*, 17:56 – 65, 2019. doi: 10.1016/j.bdr.2018.05.006.

[70] M. Houssin, S. Combettes, M.-P. Gleizes, and B. Lartigue. SANDMAN: a Self-Adapted System for Anomaly Detection in Smart Buildings Data Streams. In *(to appear in) Proceedings of the 18th Adaptive Computing (and Agents) for Enhanced Collaboration (ACEC) at WETICE 2020*, 2020.

[71] W. Huber, M. Lädke, and R. Ogger. Extended floating-car data for the acquisition of traffic information. In *Proceedings Of 6th World Congress On Intelligent Transport Systems (ITS), Held Toronto, Canada, November 8-12, 1999*, 1999.

[72] Information Commissioner's Office (ICO). Guide to the environmental information regulations. Available at `https://ico.org.uk/for-organisations/guide-to-the-environmental-information-regulations/` (2020-10-7), 2017.

[73] G. James, D. Witten, T. Hastie, and R. Tibshirani. *An Introduction to Statistical Learning*, volume 103 of *Springer Texts in Statistics*. Springer, 2013.

[74] V. Julian and V. Botti. Multi-agent systems. *Applied Sciences*, 9(7):1402, 2019.

[75] J. Junttila. Structural breaks, ARIMA model and finnish inflation forecasts. *International Journal of Forecasting*, 17(2):203 – 230, 2001. doi: 10.1016/S0169-2070(00)00080-7.

[76] J.O. Kephart and D.M. Chess. The vision of autonomic computing. *Computer*, 36(1):41–50, 2003.

[77] H. Knutsson and C.-F. Westin. Normalized and differential convolution. In *Computer Vision and Pattern Recognition*, pages 515–523. IEEE, 1993.

[78] A. Kobren, N. Monath, A. Krishnamurthy, and A. McCallum. A hierarchical algorithm for extreme clustering. In *International Conference on Knowledge Discovery and Data Mining*, pages 255–264. ACM, 2017.

[79] X. Kong, X. Liu, B. Jedari, M. Li, L. Wan, and F. Xia. Mobile crowdsourcing in smart cities: Technologies, applications, and future challenges. *IEEE Internet of Things Journal*, 6(5):8095–8113, 2019.

[80] M. Kuhn and K. Johnson. *Applied Predictive Modeling*. Springer, 2013. doi: 10.1007/978-1-4614-6849-3.

[81] K. Kumar, M. Parida, and V.K. Katiyar. Short term traffic flow prediction for a non urban highway using artificial neural network. *Procedia - Social and Behavioral Sciences*, 104:755 – 764, 2013. 2nd Conference of Transportation Research Group of India (2nd CTRG).

[82] P. Kumar, Y. Lin, G. Bai, A. Paverd, J. S. Dong, and A. Martin. Smart grid metering networks: A survey on security, privacy and open research issues. *IEEE Communications Surveys Tutorials*, 21(3):2886–2927, 2019.

[83] G. C. Lazaroiu and M. Roscia. Definition methodology for the smart cities model. *Energy (Elsevier)*, 47(1):326 – 332, 2012.

[84] E. L. Lehmann and G. Casella. *Theory of Point Estimation*. Springer Texts in Statistics. Springer, 1998.

[85] S. Lemouzy. *Systèmes interactifs auto-adaptatifs par systèmes multi-agents auto-organisateurs : application à la personnalisation de l'accès à l'information*. Phd thesis, Université Toulouse III - Paul Sabatier, 2011.

[86] T. Liu, H. Wei, and K. Zhang. Wind power prediction with missing data using gaussian process regression and multiple imputation. *Applied Soft Computing*, 71:905 – 916, 2018.

[87] J. Luengo, S. García, and F. Herrera. A study on the use of imputation methods for experimentation with radial basis function network classifiers handling missing attribute values: The good synergy between rbfns and eventcovering method. *Neural Networks*, 23(3):406 – 418, 2010.

[88] Y. Ma, S. Liu, G. Xue, and D. Gong. Soft sensor with deep learning for functional region detection in urban environments. *Sensors*, 20(12):3348, 2020.

[89] S. Maldonado, A. González, and S. Crone. Automatic time series analysis for electric load forecasting via support vector regression. *Applied Soft Computing*, 83:105616, 2019.

[90] G. Marcillaud, V. Camps, S. Combettes, M.-P. Gleizes, and E. Kaddoum. Management of intelligent vehicles: Comparison and analysis. In A. P. Rocha, L. Steels, and H. J. van den Herik, editors, *Proceedings of the 12th International Conference on Agents and Artificial Intelligence, ICAART 2020, Volume 1, Valletta, Malta, February 22-24, 2020*, pages 258–265. SCITEPRESS, 2020.

[91] F. Marvasti. *Nonuniform Sampling. Theory and Practice*. Information Technology: Transmission, Processing and Storage. Kluwer Academic/Plenum Publishers, 2001.

[92] T. Marwala. *Computational Intelligence for Missing Data Imputation, Estimation, and Management: Knowledge Optimization Techniques*. IGI Global, 2009.

[93] F. Mateo, J. J. Carrasco, A. Sellami, M. Millán-Giraldo, M. Domínguez, and E. Soria-Olivas. Machine learning methods to forecast temperature in buildings. *Expert Systems with Applications*, 40(4):1061 – 1068, 2013.

[94] S. Mellouli, L. Luna-Reyes, and J. Zhang. Smart government, citizen participation and open data. *Information Polity (IOS press)*, 19:1–4, 2014.

[95] Z. Michalewicz and D. B. Fogel. *How to Solve It: Modern Heuristics*. Springer.

[96] T. Munakata. *Fundamentals of the New Artificial Intelligence: Neural, Evolutionary, Fuzzy and More*. Texts in Computer Science. Springer, 2 edition, 2008.

[97] M. Negnevitsky. *Artificial Intelligence: A Guide to Intelligent Systems*. Addison-Wesley, 1st edition, 2001.

[98] J. Nigon, N. Verstaevel, J. Boes, F. Migeon, and M.-P. Gleizes. Smart is a matter of context. In *International and Interdisciplinary Conference on Modeling and Using Context*, pages 189–202. Springer, 2017.

[99] V. Noël and F. Zambonelli. Following the problem organisation: A design strategy for engineering emergence. In D. Camacho, L. Braubach, S. Venticinque, and C. Badica, editors, *Intelligent Distributed Computing VIII*, pages 311–317. Springer, 2015.

[100] Tomoaki O. A smart city based on ambient intelligence. *IEICE Transactions on Communications*, E100.B(9):1547–1553, 2017.

[101] A. Olaru, A. M. Florea, and A. El Fallah Seghrouchni. A context-aware multi-agent system as a middleware for ambient intelligence. *Mobile Networks and Applications*, 18(3):429–443, 2013.

[102] OpenStreetMap contributors. Planet dump retrieved from https://planet.osm.org. Available at `https://www.openstreetmap.org` (2020-10-7), 2017.

[103] S.-V. Oprea and A. Bâra. Machine learning algorithms for short-term load forecast in residential buildings using smart meters, sensors and big data solutions. *IEEE Access*, 7:177874–177889, 2019.

[104] Y. Pan, D. Wu, and D. L. Olson. Online to offline (o2o) service recommendation method based on multi-dimensional similarity measurement. *Decision Support Systems*, 103:1–8, 2017.

[105] C. Perera, A. Zaslavsky, P. Christen, and D. Georgakopoulos. Sensing as a service model for smart cities supported by internet of things. *Transactions on Emerging Telecommunications Technologies*, 25(1):81–93, 2014. 10.1002/ett.2704.

[106] T. Pham and L. van Vliet. Normalized Averaging Using Adaptive Applicability Functions with Applications in Image Reconstruction from Sparsely and Randomly Sampled Data. In *Image Analysis*, pages 485–492. Springer, 2003.

[107] T. Pham, L. van Vliet, and K Schutte. Robust Fusion of Irregularly Sampled Data Using Adaptive Normalized Convolution. *Journal on Advances in Signal Processing*, pages 1–12, 2006.

[108] F. Piette, C. Caval, C. Dinont, A. El Fallah-Seghrouchni, and P. Tailliert. A Multi-agent Solution for the Deployment of Distributed Applications in Ambient Systems. In *Engineering Multi-Agent Systems*, pages 156–175. Springer, 2016.

[109] S. Pirttikangas, Y. Tobe, and N. Thepvilojanapong. Smart environments for occupancy sensing and services. In H. Nakashima, H. Aghajan, and J. C. Augusto, editors, *Handbook of Ambient Intelligence and Smart Environments*, pages 825–849. Springer, 2010.

[110] I. Pisa, I. Santín, J. Vicario, A. Morell, and R. Vilanova. ANN-based soft sensor to predict effluent violations in wastewater treatment plants. *Sensors*, 19(6):1280, 2019.

[111] J. Plouffe. The fuzzy logic method for simpler forecasting. *International Journal of Engineering Business Management*, 3, 08 2011.

[112] Singh R. P., Javaid M., Haleem A., and Suman R. Internet of things (IoT) applications to fight against COVID-19 pandemic. *Diabetes & Metabolic Syndrome: Clinical Research & Reviews*, 14(4):521 – 524, 2020.

[113] M. Rafsanjani, Z. Varzaneh, and N. Chukanlo. A Survey Of Hierarchical Clustering Algorithms. *Journal of Mathematics and Computer Science*, 5(3):229–240, 2012.

[114] S. Revathi and N. Sivakumaran. Fuzzy based temperature control of greenhouse. *IFAC-PapersOnLine*, 49(1):549 – 554, 2016. 4th IFAC Conference on Advances in Control and Optimization of Dynamical Systems ACODS 2016.

[115] S. Rougemaille, J.-P. Arcangeli, M.-P. Gleizes, and F. Migeon. ADELFE design, AMAS-ML in action. In A. Artikis, G. Picard, and L. Vercouter, editors, *Engineering Societies in the Agents World IX: 9th International Workshop, ESAW 2008, Saint-Etienne, France, September 24-26, 2008, Revised Selected Papers*, pages 105–120. Springer, 2009.

[116] D. B. Rubin. Inference and missing data. *Biometrika*, 63(3):581–592, 1976.

[117] D. E. Rumelhart, G. E. Hinton, and R. J. Williams. Learning representations by back-propagating errors. *Nature*, 323(6088):533–536, 1986.

[118] S. Russell and P. Norvig. *Artificial Intelligence: A Modern Approach*. Prentice Hall, 3 edition, 2010.

[119] L. Sabatucci and M. Cossentino. Supporting dynamic workflows with automatic extraction of goals from BPMN. *ACM Transactions on Autonomous and Adaptive Systems*, 14(2):1–38, 2019.

[120] R. Sánchez-Corcuera, A. Nuñez Marcos, J. Sesma-Solance, A. Bilbao-Jayo, R. Mulero, U. Zulaika, G. Azkune, and A. Almeida. Smart cities survey: Technologies, application domains and challenges for the cities of the future. *International Journal of Distributed Sensor Networks*, 15(6), 2019.

[121] R. E. Schapire and Y. Freund. *Boosting: Foundations and Algorithms*. Adaptive Computation and Machine Learning. The MIT Press, 2012.

[122] V. Seal, A. Raha, S. Maity, S. Kumar Mitra, A. Mukherjee, and M. K. Naskar. A Simple Flood Forecasting Scheme Using Wireless Sensor Networks. *International Journal of Ad hoc, Sensor & Ubiquitous Computing (AIRCC)*, 3(1), feb 2012.

[123] Z. Shan, Y. Xia, P. Hou, and J. He. Fusing incomplete multisensor heterogeneous data to estimate urban traffic. *IEEE MultiMedia*, 23(3):56–63, 2016-07.

[124] A. Sharif, J. P. Li, and M. A. Saleem. Internet of things enabled vehicular and ad hoc networks for smart city traffic monitoring and controlling: a review. *International Journal of Advanced Networking and Applications*, 10(3):3833–3842, 2018.

[125] A. Sinha, P. Kumar, N. P. Rana, R. Islam, and Y. K. Dwivedi. Impact of internet of things (IoT) in disaster management: a task-technology fit perspective. *Annals of Operations Research*, 283(1):759–794, 2019.

[126] A. Solanas, C. Patsakis, M. Conti, I. S. Vlachos, V. Ramos, F. Falcone, O. Postolache, P. A. Perez-martinez, R. D. Pietro, D. N. Perrea, and A. Martinez-Balleste. Smart health: A context-aware health paradigm within smart cities. *IEEE Communications Magazine*, 52(8):74–81, 2014.

[127] V. Soler, J. Cerquides, J. Sabria, J. Roig, and M. Prim. A method to classify data by fuzzy rule extraction from imbalanced datasets. In *Proceedings of the 2006 conference on Artificial Intelligence Research and Development*, pages 55–62. IOS Press, 2006.

[128] S. K. Sood, R. Sandhu, K. Singla, and V. Chang. IoT, big data and HPC based smart flood management framework. *Sustainable Computing: Informatics and Systems*, 20:102 – 117, 2018. doi: 10.1016/j.suscom.2017.12.001.

[129] B. Spencer, O. Alfandi, and F. Al-Obeidat. A refinement of lasso regression applied to temperature forecasting. *Procedia Computer Science*, 130:728–735, 2018.

[130] B. L. R. Stojkoska and K. V. Trivodaliev. A review of internet of things for smart home: Challenges and solutions. *Journal of Cleaner Production*, 140:1454 – 1464, 2017.

[131] The Shift Project. Lean ICT: Towards Digital Sobriety. Available at `https://theshiftproject.org/wp-content/uploads/2019/03/Lean-ICT-Report_The-Shift-Project_2019.pdf` (2020/10/09).

[132] R. Tibshirani, G. Walther, and T. Hastie. Estimating the number of clusters in a data set via the gap statistic. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 63(2):411–423, 2001.

[133] D. Tomaras, V. Kalogeraki, N. Zvgouras, N. Panagiotou, and D. Gunopulos. Evaluating the health state of urban areas using multi-source heterogeneous data. In *2018 IEEE 19th International Symposium on "A World of Wireless, Mobile and Multimedia Networks"*, pages 14–22, 2018. doi: 10.1109/WoWMoM.2018.8449761.

[134] A. M. Townsend. *Smart Cities: Big Data, Civic Hackers, and the Quest for a New Utopia*. W. W. Norton & Company, 2013.

[135] A Turing. Lecture to the london mathematical society on 20 february 1947. published inam turing's ace report of 1946 and other papers', 1947.

[136] S. van Buuren. *Flexible Imputation of Missing Data*. Chapman and Hall/CRC, 2012.

[137] N. Verstaevel, J. Boes, J. Nigon, D. d'Amico, and M.-P. Gleizes. Lifelong machine learning with adaptive multi-agent systems. In *9th Conference on Agents and Artificial Intelligence*, pages 275–286. SciTePress, 2017.

[138] J. Viterbo, L. Mazuel, Y. Charif, M. Endler, N. Sabouret, K. Breitman, A. El Fallah-Seghrouchni, and J. Briot. Ambient intelligence: Management of distributed and heterogeneous context knowledge. *CRC Studies in Informatics Series. Chapman & Hall*, pages 1–44, 2008.

[139] B. Walczak and D.L. Massart. The radial basis functions - partial least squares approach as a flexible non-linear regression technique. *Analytica Chimica Acta*, 331(3):177–185, 1996. doi: 10.1016/0003-2670(96)00202-4.

[140] J.-B. Waldner. *Nanocomputers and swarm intelligence*. Wiley, 2013.

[141] K. I.-K. Wang, W. H. Abdulla, and Z. Salcic. Ambient intelligence platform using multi-agent system and mobile ubiquitous hardware. *Pervasive and Mobile Computing*, 5(5):558–573, 2009.

[142] Y. Wang and I. H. Witten. Modeling for optimal probability prediction. In *Proceedings of the Nineteenth International Conference on Machine Learning*, pages 650–657. Morgan Kaufmann, 2002.

[143] Y. Wang, Z. Xie, Q. Hu, and S. Xiong. Correlation aware multi-step ahead wind speed forecasting with heteroscedastic multi-kernel learning. *Energy Conversion and Management*, 163:384 – 406, 2018.

[144] M. Weiser and J. S. Brown. Designing calm technology. *Powergrid Journal*, 1, 1996.

[145] G. Weiss. *Multiagent Systems: A Modern Approach to Distributed Artificial Intelligence*. The MIT Press, 2000.

[146] D. Weyns and F. Michel. Agent environments for multi-agent systems – a research roadmap. In D. Weyns and F. Michel, editors, *Agent Environments for Multi-Agent Systems IV*, pages 3–21. Springer, 2015.

[147] D. Weyns, A. Omicini, and J. Odell. Environment as a first class abstraction in multiagent systems. *Autonomous Agents and Multi-Agent Systems*, 14(1):5–30, 2006.

[148] M. Winikoff and L. Padgham. The prometheus methodology. In F. Bergenti, M.-P. Gleizes, and F. Zambonelli, editors, *Methodologies and Software Engineering for Agent Systems*, volume 11, pages 217–234. Kluwer Academic Publishers, 2004.

[149] I. H. Witten, F. Eibe, and M. A. Hall. *Data Mining: Practical Machine Learning Tools and Techniques*. Elsevier, 2011. doi: 10.1016/C2009-0-19715-5.

[150] M. Wooldridge. *An Introduction to MultiAgent Systems*. Wiley, 2 edition, 2009.

[151] W. B. Wu, M. Woodroofe, and G. Mentz. Isotonic regression: Another look at the changepoint problem. *Biometrika*, 88(3):793–804, 10 2001. doi: 10.1093/biomet/88.3.793.

[152] L. Yu, N. Wang, and X. Meng. Real-time forest fire detection with wireless sensor networks. In *Proceedings. 2005 International Conference on Wireless Communications, Networking and Mobile Computing, 2005.*, volume 2, pages 1214–1217, 2005.

[153] F. Zambonelli, N. R. Jennings, and M. Wooldridge. Developing multiagent systems: The gaia methodology. *ACM Transactions on Software Engineering and Methodology*, 12(3):317–370, 2003.

[154] L. Zhang and P.N. Suganthan. A survey of randomized algorithms for training neural networks. *Information Sciences*, 364-365:146–155, 2016.

[155] Y. Zhang and A. Haghani. A gradient boosting method to improve travel time prediction. *Transportation Research Part C: Emerging Technologies*, 58:308 – 324, 2015. Big Data in Transportation and Traffic Engineering.

[156] J. Y. Zhu, C. Sun, and V. O. K. Li. Granger-causality-based air quality estimation with spatio-temporal (s-t) heterogeneous big data. In *2015 IEEE Conference on Computer Communications Workshops*, pages 612–617, 2015. doi: 10.1109/INFCOMW.2015.7179453.