UNIVERSITY OF SOUTHAMPTON

Coordinating Measurements for Participatory Sensing Applications

by

Alexandros Zenonos

A thesis submitted in partial fulfillment for the
degree of Doctor of Philosophy

in the
Faculty of Physical Sciences and Engineering
Electronics and Computer Science

February 2018

UNIVERSITY OF SOUTHAMPTON

ABSTRACT

FACULTY OF PHYSICAL SCIENCES AND ENGINEERING
ELECTRONICS AND COMPUTER SCIENCE

Doctor of Philosophy

by Alexandros Zenonos

It is estimated that there are more than 7 billion mobile phone devices active worldwide. This radical growth of mobile technology is starting to be exploited by experts for cheap large-scale data collection. In this work, we are interested in environmental data, such as radiation, noise and air pollution, which is crucial for public health. The traditional approach of collecting environmental data typically requires equipment that is expensive to obtain and maintain, as well as a number of environmental sciences experts to administer them. On the other hand, by exploiting the wide availability of mobile devices, fine grained sensor data can be collected in cities. This data can be used to create detailed maps providing insight to experts about the environmental phenomenon, which in turn will assist the authorities in decision making and urban planning. In more detail, we are interested in the concept of participatory sensing, where people contribute information from the mobile devices they carry with them. However, even though collecting data through people's mobile devices is effective and cheap, people are often self-interested actors that only have local information about the environment and pursue their own agenda. This means measurements may be taken in a suboptimal way. In particular, participants often do duplicate work, i.e., different people take a number of measurements at the same location and time, or they do not explore the whole map of interest, which leads to a partial or false picture of the environment.

To address these challenges, a coordination system is needed to guide or suggest when, where and who should take measurements. Specifically, the use of intelligent algorithms can solve this problem by coordinating and assisting humans to take more informative measurements as well as fill the gaps for areas that are not covered yet and avoid duplicate work. Moreover, since humans are often predictable in their daily routines the system can exploit this fact in order to make more informative suggestions to people. In particular, a key aim in this work is to ensure that people can get suggestions about taking measurements at times and locations that are least intrusive to their daily life. However, people might not provide the measurements suggested or worse provide false information for their own reasons.

Against this background, we provide a complete participatory sensing framework with algorithms for coordinating measurements for environmental monitoring. Our algorithms use local search, heuristics, clustering techniques and stochastic simulations to map participants to observations that need to be taken. In particular, our algorithms intelligently search through the space of possible solutions to find mappings that will maximise the total information learned about the environment in a given time period.

The main contributions of this thesis are three algorithms that solve the problem with different requirements. Specifically, the first algorithm, Local Greedy Search, LGS, deals with more deterministic scenarios, in terms of participants' mobility patterns and behaviour. The second algorithm, adaptive Best-Match, ABM, deals with uncertainty in participants' mobility patterns and behaviour, in terms of taking the suggested measurements. Finally, the third algorithm, Trust-based adaptive Best-Match, TABM, deals with coordinating participants in the presence of malicious users, who attempt to alter the overall picture of the environment by submitting false measurements.
We empirically evaluate our algorithms on real-world human mobility and air quality data. Our results show that our algorithms outperform the state of the art in terms of utility gain and accuracy, while being faster at runtime. This indicates that coordinating measurements has a significant benefit in participatory sensing applications in terms of understanding environmental phenomena.

Contents

List of Figures

List of Tables

Nomenclature

$\mathcal{A}_i$	A single agent
$\mathbf{A}$	The set of all agents
o	An observation/measurement
$\mathbf{O}$	Set of all observations
$\mathbf{O}_t$	Set of all observations until time t
E	Last timestep of an environmental campaign
$\mathbb{U}(\mathbf{O}_E)$	Total utility gained from a set of observations up to timestep E
$\mathcal{E}$	The environment to be monitored
$\mathcal{C}$	Cluster of agents
L	Set of all locations of the environment
l	A specific location
t	A specific timestep
T	Set of all timesteps
$loc(o_{i,t})$	Spatial coordinates of observation o
$c_i(l,t)$	Cost for agent i to take measurement at location l at time t
C_i	Cost of all measurements taken by agent i
M	Total number of agents in the campaign $\|A\|$
B_i	Budget of agent i
$r(\mathcal{A}_i)$	Reliability of agent i
s	Set of observations selected by the algorithm
S^*	Set of observations that maximise utility
X_l	Random variable for location l
x_l	Realisation of random variable X_l
$\mathbf{x}_*$	Input vector of unobserved locations
$m(\mathcal{A}_i)$	Maliciousness value for an agent i
$\mathcal{L}$	Discretised set of all location in the environment
$\mathcal{T}$	Discretised set of all timesteps in the environmental campaign
$K(\cdot,\cdot)$	Covariance matrix
$l_1, l_2, l_3, \sigma_n, \sigma_f$	GP hyperparameters

Acknowledgements

I would like to thank everyone who supported me throughout this research.

First of all, I would like to thank my supervisors, Prof. Nick Jennings and Dr. Sebastian Stein for their guidance, encouragement and support. Their help has been invaluable as they provided me with the tools, skills and motivation a researcher needs in order to succeed. I just hope that I did well enough to make them proud. I am also grateful to them for this PhD opportunity, as being part of the ORCHID project was a unique experience. I had the chance to attend group meetings, summer schools, conferences, and other special events in the UK and abroad. This helped me gain a lot of experience and develop a number of skills.

I would also like to thank all the members of the ORCHID project and Agents, Interaction and Complexity group (AIC) for the fruitful discussions we had.

I would like to thank Prof. Andreas Pitsillides for his mentorship throughout my academic years. Last but not least, I would like to thank my parents and my brother for their full support at all times as well as Eleni Erimaki for her patience and understanding.

Finally, this research was undertaken as part of the ORCHID project funded by EPSRC (EP/I011587/1).

To family and friends

Chapter 1

Introduction

Applications involving the placement of sensors for monitoring dynamic environmental phenomena, such as radiation, air and noise pollution, are receiving considerable attention (Brown et al., 2016; Venanzi et al., 2013; Seinfeld and Pandis, 2012; Stansfeld and Matheson, 2003). It is a subject that concerns many, from environmental organisations to policymakers and the general public. Noise pollution can cause heart conditions, loss of sleep and changes in brain chemistry (Chepesiuk, 2005). Poor air quality can have short-term effects on health, such as headaches, asthma, eye irritations and lack of concentration (Mabahwi et al., 2014; Seaton et al., 1995a). More importantly, however, air pollution is responsible for a range of heart-related diseases and leads to approximately 7 million deaths per year (Landrigan, 2017). This costs the global economy hundreds of billions of pounds in terms of lost labour income and trillions in welfare losses (World-Bank, 2016). Thus, understanding the phenomenon and predicting how it is going to change, in the long term as well as on a daily or even hourly basis, is crucial in allowing decision makers to take action. For example, in terms of urban planning, city councils can make decisions about where to build parks and plant trees to minimise the effect of high pollution areas in cities (Paoletti et al., 2011) or construct new roads so as to efficiently handle traffic based on air pollution measurements[1]. Furthermore, it can help doctors link environmental factors with symptoms, and thus improve patients' treatment (Burke et al., 2006).

In all of these cases, monitoring spatio-temporal phenomena used to require a significant amount of effort and a high cost to accomplish. In particular, traditional methods of environmental monitoring usually involved a number of expensive specialised static sensors (Jutzeler et al., 2014; Chong and Kumar, 2003). Such approaches also required a number of experts working for a significant number of hours in order to sample the environment and analyse the collected data (Brian et al., 2008).

[1] `http://planningguidance.communities.gov.uk/blog/guidance/air-quality/when-could-air-quality-be-relevant-to-a-planning-decision/`

(a) Dylos (b) Aeroqual (c) Airbeam

Figure 1.1: *Examples of portable devices that measure air quality in terms of atmospheric particulate matter (PM).*

Recent advances in technology now offer an alternative way to monitor environmental phenomena. In particular, there are an estimated 3 billion Internet users[2] and more than 7 billion mobile phones active worldwide according to GSMA Intelligence's real-time tracker[3]. Moreover, 80% of all adults who are online own a smartphone[4]. This proliferation of the Internet and mobile technology has given rise to a new data collection paradigm which is called *participatory sensing*. Participatory sensing is a term used to describe the contribution of sensory information by a group of people using mobile equipment (Burke et al., 2006). This approach, which will be the one explored in this thesis, is so named because people contribute sensory information using cheap sensor devices, so that the burden is divided between thousands of individuals (not necessarily experts). In other words, participatory sensing has made city-scale environmental campaigns feasible and cost-effective (D'Hondt et al., 2013). The sensors utilised can vary from the ones built in to smartphone mobile devices, such as microphone and global positioning systems (GPS), to external portable ones, for example, sensors that can be connected and controlled via a smartphone. For instance, 'Dylos'[5] , 'Aeroqual'[6] or 'AirBeam'[7] are low-cost mobile air quality sensors that are able to measure fine particulate matter in the atmosphere (shown in Figure 1.1).

Empowering citizens with sensors makes participatory sensing a promising paradigm for data collection especially in urban planning, public health and natural resource management (Burke et al., 2006). As we show below, participatory sensing is already being utilised in many existing applications as a data collection paradigm.

In more detail, existing participatory sensing projects have included asking people to observe plants and collect plant life data (Han et al., 2011), assessing the living quality of

[2] `http://www.internetworldstats.com/stats.htm`

[3] `http://www.independent.co.uk/life-style/gadgets-and-tech/news/there-are-officially-more-mobile-devices-than-people-in-the-world-9780518.html`

[4] `http://techcrunch.com/2015/01/12/80-of-all-online-adults-now-own-a-smartphone-less-than-10-use-wearables/`

[5] `http://www.dylosproducts.com/`

[6] `http://www.aeroqual.com/product/series-500-portable-air-pollution-monitor`

[7] `http://www.takingspace.org/aircasting/airbeam/`

people in cities (Shen et al., 2017), reporting bird sighting data (Wiggins, 2011), and taking geotagged photos of blooming flora for the purposes of water conservation (Reddy et al., 2010b). Moreover, participatory sensing has been used in projects that could potentially improve public health and assist in urban planning and management. For instance, it has been successfully used to monitor non-life-threatening radioactive environments by collecting more than 43 million measurements in five years, creating fine-grained spatio-temporal heatmaps of radiation (Brown et al., 2016). Also, NoiseTube[8] is a project that explores the use of the participatory sensing approach to measure and map urban noise pollution using smartphones (Stevens and D'Hondt, 2010) in cities. NoiseTube is further discussed in the next chapter, as it has been deployed in the real-world since 2008 and, due to its maturity, serves as a case study in this work.

In most participatory sensing systems, including the projects mentioned above, a number of experts (or task requesters/taskmasters) initiate a campaign (or a task) that ordinary people can contribute to in order to collect information. People take part in such missions for different reasons (Gao et al., 2015). Specifically, some people participate for monetary incentives (extrinsic incentives) (Jaimes et al., 2012). This can take the form of micro-payments or coupons (Albers et al., 2013). For example, a micro-payment scheme was used as an incentive to promote realtime participation in a university campus garbage monitoring campaign (Reddy et al., 2010a). Likewise, SenseUtil is a model where the consumer who needs data pays the producers who carry out sensing tasks and report the data. The price is determined based on the concept of demand and supply (Thepvilojanapong et al., 2013), where the price changes dynamically according to the sensing frequency, quantity of nearby sensing locations and user preferences.

Others volunteer for social reasons, for example, to gain public recognition or a high position on a leader-board. In some systems, volunteers compete against friends for points or badges (Anderson et al., 2013). Finally, some people volunteer because of their personal interest in a social cause, altruism, or as a hobby (intrinsic incentives) (Jennings et al., 2014). For example, an application for finding an endangered species of insects in the UK relies on the excitement of the visitors of a particular area on the South coast of England that the insect is believed to inhabit (Zilli et al., 2013).

In this work, we capture the need for incentivisation (either intrinsic, extrinsic or social interest) by assigning a cost for taking measurements to each participant (Chapter 4) or a budget (Chapter 5). While a cost can capture the energy cost or monetary compensation required to take measurements, it is difficult to quantify its value, as it has to be related to the information gained when a measurement is taken. A budget is another way to constrain the number of measurements that each user is able to take. This can represent the number of measurements the mobile device is able to take before battery life is depleted or the number of measurements a user is willing to take given a specific incentive, such as money or social interest. Crucially, in either case it cannot be assumed

[8] `http://www.noisetube.net/`

that participants will provide an unlimited number of measurements; they should not be seen as robotic entities that behave exactly as instructed all the time. Instead, it is more fitting to view them as self-interested agents that have their own personal goals and limited information about the environment. For instance, they might believe that taking a measurement in the city centre is more useful than elsewhere, since more people are potentially affected, or they might not take a measurement at all since they are too busy with their daily routine and might believe that the impact of a single measurement is low at their location.

Despite the need for incentivisation, participatory sensing delivers impressive results. However, all the aforementioned projects lack an equally important element. They do not provide a coordination system that can efficiently guide or suggest to participants when and where to take measurements in order to collect the most valuable information about the environment. Rather, they rely on people taking measurements whenever they want. In particular, citizens are self-interested actors making local decisions and pursuing their own agenda. This is a major problem, because some areas remain unexplored, which leads to a false or partial picture of the situation over the entire environment the campaign initiator is interested in. Also, people may provide redundant information by taking measurements at the same time and place. This information adds an unnecessary communication and processing burden, as it entails no new information, and consequently needs to be addressed by the interested parties. Furthermore, the very openness of this approach enables the contribution of corrupted data. In particular, people can act selfishly and exploit the system for their own benefit. Crucially, participatory sensing systems are prone attacks from *malicious* users (Mousa et al., 2015; Gadiraju et al., 2015). For example, users who fabricate higher pollution measurements to affect the decision of authorities and policymakers about the development of parks and roads. This issue is shown to be ubiquitous in systems that depend on people to perform specific tasks (Gadiraju et al., 2015). Consequently, monitoring human behaviour, in terms of false measurements contribution in participatory sensing settings is an essential need in order to create accurate maps of the environment.

Moreover, in some cases, such as when people carry GPS enabled phones with them or their phones have Internet connectivity, knowledge about how they tend to move at particular times might be available to the campaign initiator. This information could be available either by learning those patterns (Gonzalez et al., 2008; Baratchi et al., 2014a) or by asking users to explicitly submit their future travel plans[9]. Typically, people tend to be predictable in their daily routine (McInerney et al., 2013b). However, to date, only limited work in participatory sensing has looked into harnessing this knowledge (Chen et al., 2014). This is a shortcoming because such knowledge can indicate when and where sensors are available. In other words, a participatory sensing system could exploit the fact that participants are likely to be at specific locations at specific times in order to

[9] `http://www.tripomatic.com/`

prepare a plan for suggesting measurements that will lead to a better spatio-temporal map exploration.

Against this background, we are interested in monitoring environmental phenomena using the participatory sensing paradigm. In particular, we focus on *intelligently* collecting data (measurements/observations) with the assistance of people in order to maximise the information we learn about the environment over a period of time. This, in turn, will enable the development of a fine-grained pollution map, covering an area of interest. In more detail, the ultimate aim of this research is to provide a framework with algorithms for efficiently coordinating measurements in the participatory sensing setting for environmental monitoring. To this end, we provide the conceptual architecture and the core algorithms to build a system which will be able to be deployed on a server and should notify participants when and where to take a measurement either in real time or in advance. The coordination system however, should be generalisable not only in different environmental phenomena but also in different applications.

1.1 Research Challenges

Participatory sensing is a promising paradigm for various domains as discussed above. However, there are still a number of open challenges that need to be addressed in order to utilise its full potential. In particular, we focus on intelligently collecting data with the assistance of people in order to maximise the information we learn about the environment. In doing so, however, we will take into consideration a number of constraints that real-life applications exhibit, such as limited resources and willingness of users to participate. Furthermore, it is necessary to consider uncertainties in the environment, such as locations where no observations are taken or where the phenomenon is rapidly evolving over space and time. In addition, it is important to focus on participants' behaviour in terms of reliability and commitment with the environmental campaign. This is necessary because the aim of this work is to design algorithms that will not only perform well in small-scale deterministic scenarios, but rather be applicable in the real world. Specifically, this work considers the following key challenges:

1. **Eliminate redundancy and explore the interested area**
   In many participatory sensing applications for environmental monitoring an important challenge is when and where to take measurements in order to learn more about the area the campaign initiator is interested in. There have been a number of partial efforts to develop an orchestration platform to assist the initiator in understanding the impact of the campaign by receiving real-time feedback (these are further explained in Section 2.1.1). However, the main challenges are still open since current approaches only provide feedback to the campaign initiator and no intelligent algorithm is involved. Thus, even though it is possible to assess the

success of the campaign, there is no way to efficiently explore the area of interest. These challenges can be best highlighted through an example like the NoiseTube application. Observing Figure 1.2, which shows an example of aggregated noise data from Paris, it is clear that only some parts are covered and there is no information available about the rest of the map. Consequently, the campaign initiator only has a partial understanding of the noise pollution in the city of Paris.

This example is an instance of a ubiquitous challenge in participatory sensing domains as none of the aforementioned applications provide any sort of coordination in order to guide participants towards more informative measurements. Moreover, it is plausible that the use of cellular phones as sensory devices might result in the presence of a large number of densely located mobile sensors in an urban area. As a result, it is possible that the information collected by many of the phones is redundant (Thepvilojanapong et al., 2010). Such duplicate information may result in energy loss, in terms of battery life, communication and processing power. In this context, Thepvilojanapong et al. (2010) have shown that avoiding duplicate information in participatory sensing applications could save up to 77% of the energy consumed. The challenge here is to maximise the information collected, but at the same time reduce redundant measurements that do not add value to the total information gained. Consequently, given an increasing number of people and a number of possible spatio-temporal locations to take measurements, the possible combinations are exponentially increased, making the problem difficult to be solved.

2. **Quantify informativeness**
   As explained above, finding when and where to take measurements to gain more information about the environment is the main challenge of this thesis. This assumes that each measurement provides or entails a specific amount of information that would be gained whenever that measurement is taken. This is true but no practical to determnine in environmental monitoring as there is no trivial way of telling how much information each measurement would provide. To illustrate this, consider taking a measurement in a city where no other measurements were taken versus a measurement in a city where multiple measurements already exist. Intuitively, a measurement where no other measurements exist is more valuable, but it is not clear by how much. This depends on the distance in space and time of other nearby measurements of the phenomenon.

3. **Minimise cost / constraint on budget**
   Participatory sensing makes use of mobile devices that are limited in terms of battery capacity. Thus, making observations cannot be considered 'free'. For instance, when using a smartphone's GPS sensor, the battery is draining significantly faster than normal (Ma et al., 2013). In addition, there may be an extra

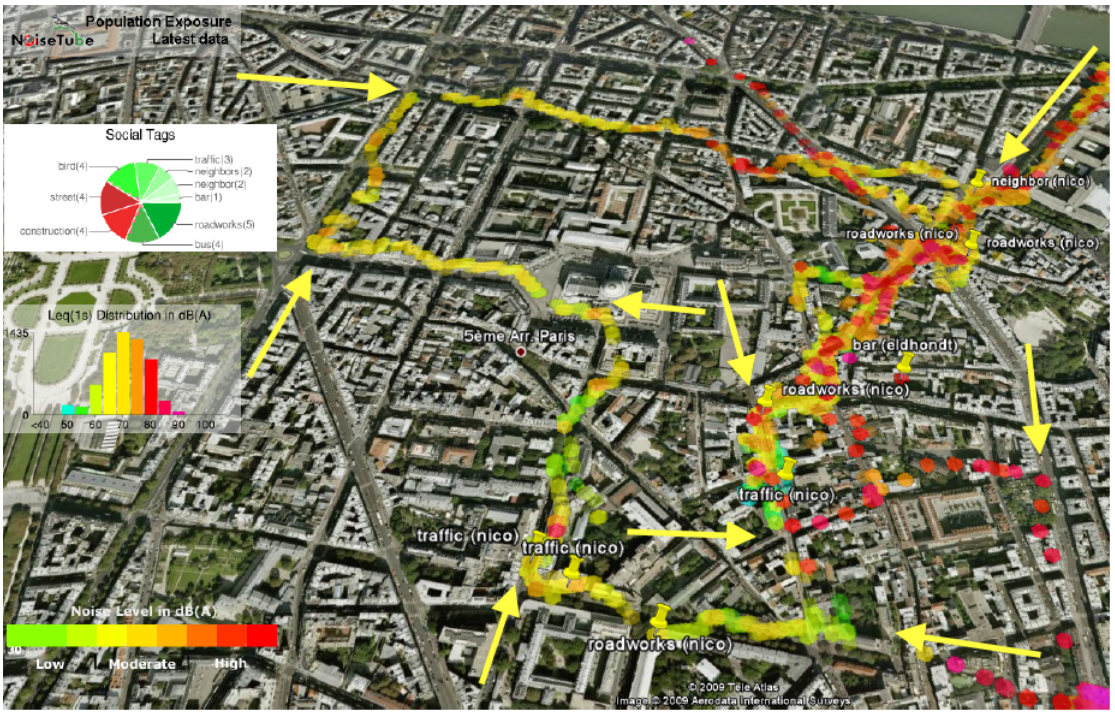


Figure 1.2: *Collective noise map for part of Paris, France adopted by (Stevens and D'Hondt, 2010). Red indicates high noise levels and green low levels. The arrows point to the roads where measurements were taken.*

cost of communicating those measurements in real time as the devices have a limited communication bandwidth. Each mobile phone device has a different battery life and people make different uses of it. Thus, it is difficult to quantify this cost, since taking a measurement might be more costly for one participant than another. Given the previous observation, i.e., avoiding unnecessary measurements can significantly improve energy efficiency, the challenge here is to effectively capture this limitation of resources and consider its effect when coordinating measurements.

Also, suggesting to people when and where to take a measurement can be intrusive (Shilton et al., 2008). Even if they have volunteered for a campaign, frequent suggestions can cause annoyance (Shilton et al., 2008). Consequently, each measurement can be associated with the cost of the burden caused to people by having to make a measurement when suggested. Another way to see this is as a budget or quota on the daily amount of measurements that each participant is able to take (Reddy et al., 2010b; Chon et al., 2013). If users were able to take measurements at all times, no algorithm would be necessary as having more information is always preferable. However, having this constraint requires a decision to be taken as to when and where each user should take a limited number of measurements.

4. **Deal with unreliability and uncertainty in human behaviour**
   Intelligent decisions in realistic scenarios about the place and time of taking measurements are clearly affected by a number of factors. Specifically, it depends on

the number of participants in the campaign and the percentage of those who are available at any given time. Furthermore, it depends on their willingness to contribute to the campaign when requested to do so. Also, even if they are willing to contribute, they might end up not taking the suggested measurement due to unforeseen circumstances, such as human error or device malfunction, or because they changed their mind about participating in the campaign. These reasons complicate the problem as uncertainty is introduced, which is related to both human behaviour and device reliability.

Current approaches do not directly deal with these issues. They partially attempt to tackle the human-related constraints by focusing on incentivisation of people as discussed above. Their rationale is to motivate and engage more people in the campaigns and get more and higher-quality information from each one of them. However, monetary incentives imply that a significant budget is associated with the environmental monitoring campaign, which is not favoured in this work, as participatory sensing is about large-scale low-cost deployment. Even if other incentives are used there is always uncertainty associated with human behaviour, which makes it impossible to completely alleviate these constraints (Reddy et al., 2010b). This inherent uncertainty makes the problem more difficult as any decisions taken in advance might be invalid when the actual campaign is running. On the other hand, planning in real time might not be optimal since decisions often need to be taken fast and information available might not be utilised.

5. **Deal with malicious users**

   Participatory sensing campaigns rely on the collective information provided by people. However, there are no standard mechanisms in place to guarantee the quality of their contribution. In fact, participatory sensing campaigns are vulnerable to misreporting. Specifically, in environmental monitoring settings, given that measurements are submitted solely by people, there might be a tendency by malicious users to falsify their contribution for their own selfish reasons (Dua et al., 2009). As a result, a false representation of the environment might be created, which in turn could affect urban planning decisions. In other settings, people might misreport or falsify data to gain monetary rewards (Gadiraju et al., 2015). Malicious users are potentially present in participatory sensing settings (Mousa et al., 2015). The challenge here is to identify malicious users or minimise the effect of their actions in order to achieve a truthful representation of the environment. Currently, research in this area is very limited, as shown in (Mousa et al., 2015). For instance, they attempted to use experts to rate users or monitor campaigns. However, this is not always feasible. For instance, experts cannot always rate people monitoring an environmental phenomenon, since the actual values of the phenomenon are not known a priori. Also, expert verification of the values, based on a collection of measurements at the same spatio-temporal location, is not always possible, since users have only a limited budget; an extra measurement could potentially limit the

ability of the system to achieve better coverage by taking measurements at more informative locations.

6. **Achieve scalability and high performance**
   Participatory sensing campaigns usually take place in large geographic areas, such as cities, states or countries for a sustained period of time and involve potentially hundreds or thousands of people. For example, NoiseTube is deployed in London, Brussels and Paris; MobileGas is deployed in Zürich and Citisense in San Diego, California. Consequently, designing a system that can deal with measurements taken in a very limited space, i.e., less than the area of a small city or a university campus, is not of great use in these applications. Thus, the challenge is to be able to decide when and where to take measurements given a large area over a time-horizon. In other words, our aim is to coordinate measurements for city-scale environmental monitoring campaigns. Finding the optimal solution, i.e., the best spatio-temporal locations where measurements should be taken in order to maximally increase the total information gained over the environment is difficult. In particular, it is likely to be very costly in terms of the computation time (Stranders, 2010). Thus, in order to make a valuable contribution in this kind of participatory sensing campaign, the time to compute these locations should be kept at a minimum. However, this may require a trade-off with performance. Finding a good compromise between these two is an important challenge that needs to be addressed in our research.

Having listed the main challenges faced in this work, there are a number of requirements that should be fulfilled in order to tackle these challenges. These requirements are presented in the following section.

1.2 Research Requirements

The challenges noted in Section 1.1 lead to the following key research requirements:

1. **Coordinating measurements**
   Participants should be notified to take a measurement only when it is necessary. Specifically, they should avoid taking duplicate measurements that cause energy loss or measurements that cost more than the benefit in general. This is to maximise the information learned about the environment, i.e., to achieve better spatio-temporal map exploration and therefore understand the phenomenon being monitored (addressing challenge 1).

2. **Representing the dynamics of environmental phenomena**
   Sensors often need to monitor highly dynamic and uncertain environments. The

environmental phenomenon being monitored should be modelled in a way that captures the spatio-temporal properties that are present in realistic scenarios. A good model of the phenomenon is required in order to predict unobserved locations at any given time. This, in turn, will provide an accurate picture of the phenomenon over the entire environment being monitored. Thus, it will facilitate better decisions about map exploration and redundancy elimination (addressing challenge 1). In particular, depending on how dynamic the phenomenon is, there might be a need to take more measurements at a specific location frequently. However, there might be cases where the phenomenon is not rapidly evolving, which means less measurements are required at the specific location. Moreover, many measurements at the same location and time might not be useful and they should rather be taken taken elsewhere. In addition, the model should be able to predict the future state of the world in order to enable the decision makers to take the best possible actions, concerning urban planning, at any given time.

3. **Calculating the informativeness of spatio-temporal locations**
   A metric to capture or calculate how informative measurements are at specific points in space and time is needed in order to be able to evaluate these locations and select those that are the most informative ones (addressing challenge 2).

4. **Incorporating constraints on measurements**
   Dealing with the challenges concerning the cost of taking measurements, the system should incorporate a cost function or limit the number of measurements people can take. Moreover, constraints related to human behaviour, such as availability of individuals at specific times or willingness to contribute, should also be taken into consideration. Thus, the algorithms should be able to function in highly uncertain environments (addressing challenges 3 and 4).

5. **Robustness to malicious users**
   The robustness of the system to malicious actions is crucial for providing an accurate and up-to-date representation of the environment to the parties interested in environmental monitoring. A key requirement is to provide the mechanisms to identify potentially malicious users and treat malicious measurements accordingly (addressing challenge 5).

6. **Scalability, performance and complexity**
   The system should be able to scale with the number of participants, i.e., hundreds or thousands, in campaigns for monitoring environments. The system should be efficient in terms of computational complexity in order to make the system applicable in real-life scenarios where time is crucial. At the same time, however, it should produce high-quality suggestions that will lead to an efficient spatio-temporal exploration and consequently optimal information maximisation by avoiding duplicated work (addressing challenge 6).

1.3 Research Contributions

The primary aim of this research is to provide the framework and algorithms in order to fulfil the requirements set out in the section above (Section 1.2). Concretely, the key contributions of this work are:

1. We propose the first participatory sensing coordination framework (Section 3.1). This framework captures the architecture of a system for participatory sensing applications. In particular, we focus on a framework for environmental monitoring applications that intelligently coordinates participatory sensing campaigns.

2. We are the first to formalise the problem of coordinating measurements for participatory sensing applications (Section 3.2). In particular, we focus on when, where and who should take an observation in order to maximise the information learned about the environmental phenomenon.

3. We develop the first algorithm (Local Greedy Search - LGS) to make decisions about who should take a measurement, when they should take it and where, so that more information about the area of interest is learned while balancing this with the cost of taking the measurements (Chapter 4).

4. We develop a novel stochastic coordination algorithm (adaptive Best-Match - ABM) that extends LGS by efficiently coordinating measurements in uncertain scenarios as well as handling hundreds of participants at every timestep. In particular, while LGS uses more accurate (deterministic) mobility patterns, ABM relies on more noisy probabilistic estimates as these are not always available. The algorithm consists of an offline component, which is responsible for simulating participatory sensing campaigns and choosing the best measurements to be taken, as well as an online component that adapts the measurements based on real-time information. The algorithm considers each individual's budget, incorporates probabilistic knowledge about human mobility patterns and deals with the uncertainty related to the willingness of people to take a measurement when notified by the system (Section 5).

5. We present the first coordination algorithm (Trust-based adaptive Best-Match - TABM) that coordinates measurements in participatory sensing in the presence of malicious users. While ABM assumes truthful measurements, TABM works in the presence of highly noisy or malicious measurements. Specifically, our algorithm swaps low-trust users with high-trust nearby users. At the same time, even when low-trust measurements are taken, they will not have a great impact on the predicted function over space and time (Section 5.7). However, this comes at a greater computational cost compared to ABM.

This work has led to the following publications, which form the basis for Chapter 4 and Chapter 5:

- Zenonos, Alexandros, Stein, Sebastian and Jennings, Nicholas R. (2015). Coordinating measurements for air pollution monitoring in participatory sensing settings. *In, 14th Int. Conference on Autonomous Agents and Multi-Agent Systems*, Istanbul, TR, 04 - 08 May 2015, 493-501.
- Zenonos, Alexandros, Stein, Sebastian and Jennings, Nicholas R. (2016). An Algorithm to Coordinate Measurements Using Stochastic Human Mobility Patterns in Large-Scale Participatory Sensing Settings.*In, Thirtieth AAAI Conference on Artificial Intelligence (AAAI-16)*, Phoenix, Arizona USA, 12-17 February 2016, AAAI Press, 3936-3942.
- Zenonos, Alexandros, Stein, Sebastian and Jennings, Nicholas R. (2017). Coordinating measurements in uncertain participatory sensing settings *Journal of Artificial Intelligence Research* [Accepted].
- Zenonos, Alexandros, Stein, Sebastian and Jennings, Nicholas R. (2017). A Trust-Based Coordination System for Participatory Sensing Applications *In 5th AAAI Conference on Human Computation and Crowdsourcing (HCOMP 2017)*, Quebec city, Canada, 24-26 October 2017 [Accepted].

In particular, the first publication presents the LGS algorithm. The second one utilises stochastic mobility patterns where people are fully reliable. The third one includes both the aforementioned approaches, while also relaxing the assumption of fully reliable people. Finally, the last publication deals with the presence of malicious users in participatory sensing campaigns.

1.4 Thesis Outline

The remainder of this report is organised as follows:

- In Chapter 2, we discuss related work and give a background on relevant techniques used in the literature. In particular, we provide the background on the techniques on how an environmental phenomenon can be modelled. Moreover, we divide related work into three parts (agent coordination, task allocation and sensor placement), which represent the key research areas that our research draws on.
- In Chapter 3, we present our vision of how a real participatory sensing system could be used in practice such that it utilises an intelligent coordination system. We also formally describe the problem of participatory sensing coordination and our representation of the environment.

- In Chapter 4, we describe our coordination algorithm (LGS) that utilises full knowledge of human mobility patterns. We also describe how we designed and performed our experiments and then compare our algorithm to our benchmarks in terms of performance.
- In Chapter 5, we describe the algorithm (ABM) we developed to tackle the extended version of our problem where there is uncertainty about human behaviour and mobility patterns We describe how we designed and performed our experiments and present our findings.
- In Chapter 6, we describe the algorithm (TABM) we developed to coordinate measurements in the presence of potentially malicious users in the campaign. We describe how we designed and performed our experiments and present our findings.
- In Chapter 7, we discuss the conclusions from our work and highlight potential future work to be done.

Chapter 2

Literature Review

In this chapter, we provide the background that this research draws on. We firstly present a number of participatory sensing applications (Section 2.1). Then, we describe how environmental phenomena are represented in related work (Section 2.2) and then how the informativeness of observations can be quantified (Section 2.3). Quantifying informativeness enables the evaluation of the spatio-temporal locations of the environment being monitored, and thus enables the most informative selection of measurements in space and time. Next, we present literature on coordination techniques. Specifically, our work draws on the intersection of three main research areas: agent coordination (Section 2.4), task allocation in the context of crowdsourcing (Section 2.5) and sensor placement in environmental monitoring (Section 2.6). In the sections below we present these research areas and show the relationship between our work and each one of those. The algorithms described in these sections could be applied in our setting, and thus used as benchmarks or provide the basis for developing novel algorithms.

2.1 Exemplar Participatory Sensing Applications

As mentioned in the first chapter, participatory sensing is a distributed data collection approach that involves citizens carrying sensors and taking measurements using mobile devices. In this section, we will give some examples of such applications, illustrating the usefulness and the potential of participatory sensing. We will also evaluate to what extent they meet our requirements, identify potential gaps in the literature and opportunities to improve the impact and usefulness of these campaigns. We focus mostly on environmental monitoring because of its importance in human health. For instance, air quality and noise monitoring are extremely important, as air and noise pollution are detrimental for human health (Passchier-Vermeer and Passchier, 2000; Seaton et al., 1995b). In particular, we discuss NoiseTube, which was already introduced in Chapter 1, GasMobile, Citisense, ExposureSense, HazeWatch, P-sense, CommonSense, OpenSense,

Safecast, The Next Big One and TrafficSense. NoiseTube is a research project started in 2008, deployed in the real world, and finished active development in 2014. However, due to its success there is a plan to reactivate it again in the near future[1]. Thus, it is a suitable example for identifying potential gaps in the literature. GasMobile, Citisense, ExposureSense, H-watch and P-sense, CommonSense and OpenSense are prominent applications of the participatory sensing paradigm in air pollution monitoring, which has severe impact on human health. Also, Safecast, The Next Big One and TrafficSense are important participatory sensing applications as they show the power of the crowd and mobile technology in other domains. Consequently, these examples constitute good candidates for understanding how participatory sensing is applied in practice and in environmental monitoring in particular, as well as identifying potential gaps in the literature. However, by no means this is an exhaustive enumeration of participatory sensing applications.

2.1.1 Noise Pollution

In this section we present the case study of Noisetube (Stevens and D'Hondt, 2010). NoiseTube is a project that tackles the noise pollution problem in several large cities in Europe. In particular, the deployment is focused on Brussels, Paris and London. It proposes a participative approach of monitoring noise pollution by involving the general public. Part of this project is the use of the NoiseTube app, a smartphone application which turns smartphones into noise sensors, enabling citizens to measure the sound exposure in their everyday environment. Each participant is able to share their geotagged measurement data in an attempt to create a collective map of noise pollution, which is available to NoiseTube community members. The main motivation for participation in this campaign is social interest. In other words, people contribute in order to understand their noise exposure, to build a collective map, to help local governments in tackling noise pollution by understanding noise statistics and to assist researchers by providing real data to analyse.

On the other hand, this project enables system designers to assess the potential of the participatory sensing approach in the context of environmental monitoring. In particular, developing a smartphone application, which is a widely adopted technology, can potentially reach thousands of people that could cover large cities. As a result, it can potentially provide a complete noise pollution map to the interested parties.

Maisonneuve et al. (2010) argue that although noise pollution is a major problem in cities around the world, current noise pollution monitoring approaches fail to assess the actual exposure experienced by citizens. In particular, static sensors are located away from streets and emission sources in order to reflect the average pollution over an area (Jutzeler et al., 2014). Consequently, they might underestimate the true exposure

[1] `http://citizen-observatory.be`

of people to air pollution. Thus, participatory sensing provides a low-cost solution for the citizens to measure their personal exposure and contribute to the community by taking measurements at the sources of the noise pollution. As a result, it can provide more accurate maps, as fine-grained measurements, both in space and time, that reflect the true exposure of people can be taken, which will enable authorities to take better action. This approach seems to work well, achieving the same accuracy as standard noise mapping techniques but at a significantly lower cost, as no expertise nor expensive sound level meter equipment is required (D'Hondt et al., 2013).

However, this project still faces some challenges. Some people find it impractical or intrusive to continuously use their mobile phone for noise monitoring (Stevens and D'Hondt, 2010). Also, the use of an app for taking noise measurements drains the phone's battery faster than normal (Stevens, 2012). Moreover, different devices have different accuracies, so there is a need for intelligent filtering of erroneous values and estimating accuracy. More incentives, financial or otherwise, should be used in order to encourage contribution and promotion of large-scale campaigns (Stevens and D'Hondt, 2010). The main challenge that we try to address in this work is the need of a system that is able to coordinate measurements taken by participants, which is required given the results obtained so far (Stevens and D'Hondt, 2010; Stevens, 2012; D'Hondt et al., 2013). In particular, their analysis has shown that people tend to take measurements when and where they want, following their own agenda, which results in a potentially suboptimal information collection. As shown in Figure 1.2 people often fail to explore the area of interest. At the same time, however, the intrusiveness of the mobile phone app, the energy consumption and the cost for taking measurements in general should be considered (requirement 4).

It has been argued that setting up participatory sensing campaigns is not a well explored field of study, and thus not much has been done on supporting those campaigns (Zaman et al., 2014). To this end, the team behind NoiseTube published a proof-of-concept architecture for orchestrating participatory sensing campaigns through feedback and analytics to the campaign initiator (Zaman et al., 2014; D'Hondt et al., 2014). In other words, when someone initiates a participatory sensing campaign, i.e., monitoring a geographical area for a period of time with the assistance of the general public, the system could monitor and support this campaign in real-time. The support is not about coordinating measurements per se but rather about providing important information to the campaign initiator (and the participants) about the progress of the campaign in terms of covered area. Specifically, they present an approach to support participatory campaigns by developing an orchestration framework[2] with focus on scalability, usability and data quality. The system is similar to Ohmage (Ramanathan et al., 2012), which is a general purpose participatory sensing platform to monitor crowdsourced data, and more specifically tailored for environmental monitoring. The idea is that a campaign is

[2]The 'orchestration' here is used to describe a system that consists of many components that if combined together can support participatory sensing campaigns in terms of providing feedback to the campaign initiator and the participants.

initiated and the system processes contributions, monitors the overall campaign progress, and provides feedback to participants to guide campaign creators towards a successful campaign. In the first step of this workflow, data is parsed and stored only when people are contributing to the specific campaign and their measurements fulfil the campaign initiator's constraints, such as geographical area and time interval. This framework keeps track of the progress by measuring the average contribution rate, which can also give an indication of the quality of the measurements. For example, if the progress is below a specific threshold the campaign initiator is informed and is able to extend the campaign duration in order to obtain better results. The framework was validated by simulating reruns of several noise monitoring campaigns that took place within the NoiseTube platform. The results show that this framework can potentially support participatory sensing campaigns but further analysis is needed to understand the extent that this is helpful.

We believe that this framework is a good step towards tackling the main challenge of supporting participatory campaigns but it only partially does so. It does not utilise data that can potentially be available to the campaign initiator, such as human mobility patterns and historical sensory information about the geographical area of interest in order to provide the full picture of the campaign. In addition, the main disadvantage of this system is that it does not utilise any coordination algorithm (requirement 1) to suggest when, where and who to take measurements in order to make participatory sensing applications more efficient. Even if the campaign initiator is provided with feedback about the progress of the campaign, there is currently no way to suggest to participants to take measurements in an efficient way to provide a better coverage of the area through time and thus maximise the information obtained. Analysing this information to make suggestions about which measurements to take requires algorithms that can intelligently take decisions. Such algorithms fall under the Artificial Intelligence domain, which was out of scope for the NoiseTube project.

2.1.2 Air Pollution

Another important area in which participatory sensing is making an impact is air pollution monitoring. Several participatory sensing platforms are utilised, each one with different specifications and constrains.

For example, GasMobile is a low-power and low-cost mobile sensing system for participatory air quality monitoring (Hasenfratz et al., 2012a). Instead of relying on the expensive static measurement stations operated by official authorities for highly reliable and accurate measurements, GasMobile relies on the participatory sensing paradigm. In particular, GasMobile is a system developed from the combination of a small-sized, low-cost ozone sensors and an off-the-shelf smartphone. This system, besides taking ozone measurements to calculate air quality, can also exploit nearby static measurement

Figure 2.1: *Overview of air pollution map based on GasMobile measurements. This is the average concentration value per region.*

stations to improve calibration and consequently the system's accuracy. This system was used in a two-month campaign in an urban area. Specifically, the system was attached to a single bicycle and took measurements from several rides all around the city. The sampling interval was pre-set to five seconds, collecting a total of 2815 spatially distributed data points. Data collected was aggregated based on the area selected by the user interested in the results. To produce the map they divided this area into rectangular regions and took the average ozone concentration of the observations in that region. Then, each region was classified into one of three categories: green, yellow or red depending on the average concentration value as shown in Figure 2.1. The system was evaluated at a prototyping stage but it has great potential as it shows that air pollution monitoring can be achieved in a cost-effective manner. The results also show participatory sensing can produce results of high accuracy as the mean error[3] for 2815 measurements was 2.74ppb, which is only slightly higher than in static settings (Hasenfratz et al., 2012b). However, we identify several gaps to be addressed. Specifically, besides taking measurements, participants will make use of their phone, and thus a 5 second measurement interval, which is used as a sampling rate, would drain the phone's battery fast; even

[3]The mean error is calculated in units of parts per billion (ppb) which is a standard concentration metric for measuring air quality.

though due to the fact that the phone is connected with a USB cable and not Bluetooth communication, battery consumption is reduced by a factor of two (Hasenfratz et al., 2012a). However, in many cases, a USB connection might not be preferred as it restricts the mobility of the smartphone. Also, the system was used on only one bicycle, while in a real application each participant should have their own sensor or at least the sensors will be deployed in a larger set of bicycles. Consequently, it would be infeasible for all cyclists to take measurements at all times. Concretely, a coordination system would be necessary to suggest when, where and who should take measurements in order to learn more information about the environment while at the same time saving energy (requirements 1 and 4). Moreover, assuming that a potential future direction of this project is to attach sensors to public bicycles, and participants will move from a public bicycle dock to another, it is possible that they will provide duplicate measurements (measurements taken at the same location at the same time). Depending on the dynamism of the phenomenon, duplicate measurements can potentially be used to verify measurements taken, but they provide no extra information about the environment. In this setting, instead of getting a large number of duplicate measurements, it would be more beneficial to coordinate measurements, i.e., postpone some measurements or suggest some users to take them earlier in order to gain more information about the environment. We argue that an intelligent algorithm could be exploited in order to suggest the points both in space and in time at which taking measurements would lead to a better coverage of the environment from the point of view of the campaign initiator, alleviating data redundancy. Also, although air pollution is highly location/time-dependent, GasMobile does not exploit this information. For instance, traffic junctions and industrial installations can have a considerable impact on the air pollution. In addition, some roads might be congested with traffic during specific times of the day, and thus air pollution could be higher at those times. An environmental model could capture this relation (requirement 2) and thus be able to predict what the ozone concentration values are at unobserved locations as well as what these values will be in the future.

Another important participatory sensing application that attracted media attention due to its significance and popularity is Citisense (Nikzad et al., 2012) whose purpose is to monitor air pollution in large regions, such as San Diego, California, US. Citisense consists of three components: a wearable pollution sensor, a mobile phone application and a web interface. Users carry the pollution sensor and the mobile phone with them throughout the day in order to learn their air pollution exposure. The web interface provides a more detailed reflection on the air pollution exposure as well as air pollution maps built with historical air pollution data collected from the users (satisfying requirement 2). The sensor is connected via Bluetooth to the mobile phone and it is claimed to take measurements for five days in a single charge. The mobile phone app (shown in Figure 2.3) is responsible for collecting readings from the sensor and presenting them to the user. Each reading is timestamped and geotagged by utilising the

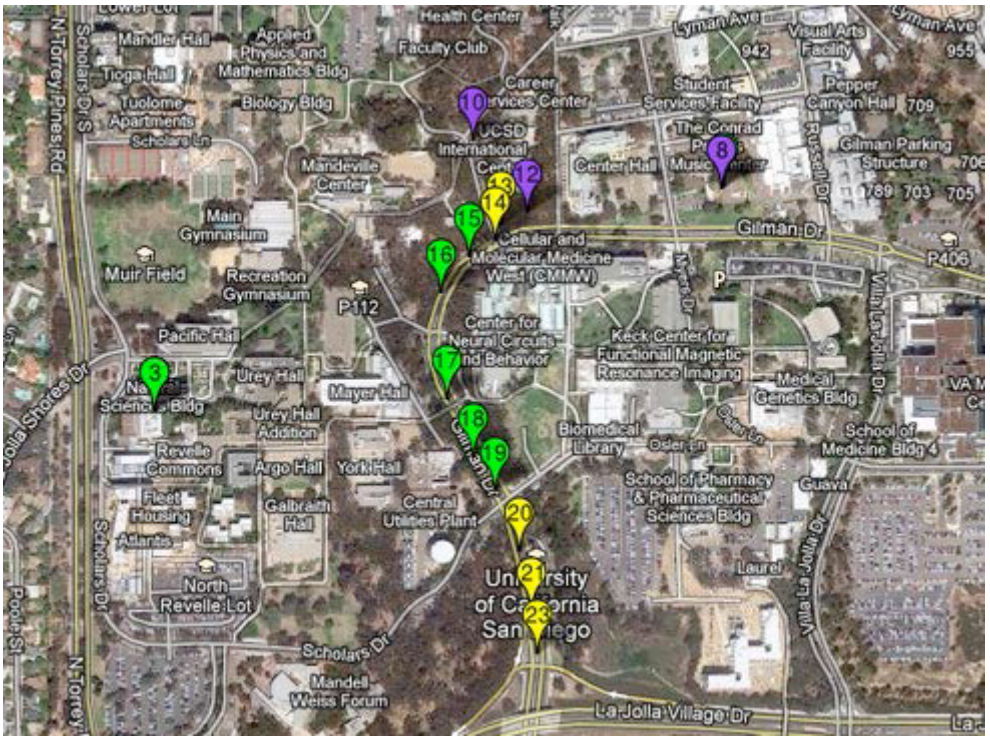


Figure 2.2: *Measurements taken for the Citisense project. Most areas remain unexplored.*

mobile phone's GPS and network-based localisation services. Citisense study was conducted in the field for one month, involving 16 participants. The results show that the users' exposure levels differ from the average measurements displayed by static sensors scattered in cities. In particular, the participatory sensing approach is able to identify pollution hot spots in the environment that develop due to busy roads, buildings and natural topology. Also, Citisense made an impact on the awareness of people. Specifically, participants better understood the properties of air pollution and in particular, they realised that being near busy streets or buses, air pollution is significantly higher than in other areas. However, as the authors admit, power management is an important challenge. Even though the sensor has a 5-day battery life, the mobile phone's battery is draining much faster, requiring users to charge their phone during the day. Clearly, this adds a significant burden to participants. Consequently, measurements were missed due to resource limitations and areas of interest remained unexplored. Given this, Citisense could strengthen, but not replace, current air quality monitoring techniques[4] due to these challenges that are also part of our requirements (requirement 4). It is evident that the lack of a coordination system limits the potential of this application.

Another participatory sensing application that attempts to monitor air pollution in large cities is ExposureSense (Predic et al., 2013). It exploits the increasing number of sensors

[4] `http://mobihealthnews.com/18828/citisense-aims-to-improve-air-quality-data-with-wearable-sensors/`

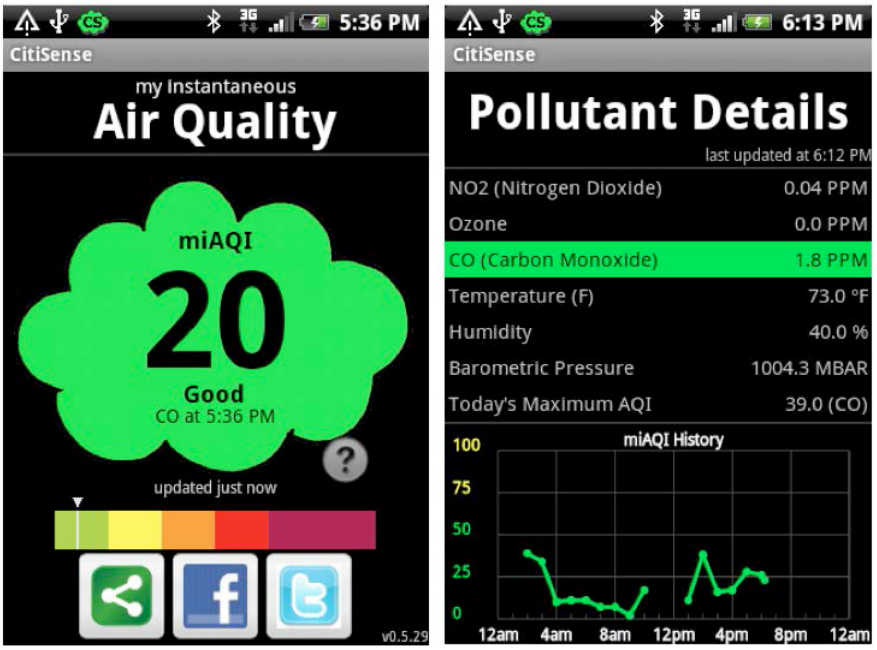


Figure 2.3: *Citisense mobile application interface*

that smartphones tend to have to convert them into powerful mobile sensor devices. ExposureSense has a different approach than other participatory sensing applications for air pollution. It attempts to correlate humans' daily activities and air quality monitoring in order to estimate the users' daily pollution exposure. To do so, the smartphone's accelerometer is used to infer the activities of users and an external mobile sensor is used for air quality monitoring. In particular, machine learning techniques are applied on accelerometer data to infer users' daily activities. In order to gather data from mobile devices they connect smartphones to air quality sensors via a USB cable. Data are also collected from external sensor networks, which are combined with data collected from the users and interpolation is performed. Data is spatio-temporally correlated in order to estimate people's daily pollutant exposure (satisfying requirement 2). Exposure intensity is scaled based on activity type, burned calories and movement speed. However, even though this application attempts to learn the users' daily activities using accelerometer measurements, it does not take into consideration the human mobility patterns of people. Thus, in our opinion, important information about users' daily activities is omitted. Research has shown that people are typically predictable in their daily routine (McInerney et al., 2013b), which can be used to enhance activity recognition inference. Specifically, having the knowledge about where someone will go can be associated with their intended activity. Human mobility patterns can also be exploited to understand people's habits which could be associated with their personal pollution exposure. For instance, if someone is jogging every afternoon in their neighbourhood, they might be exposed to more pollution than someone running in the park. In other words, the activity itself might not provide as much information as if we know the

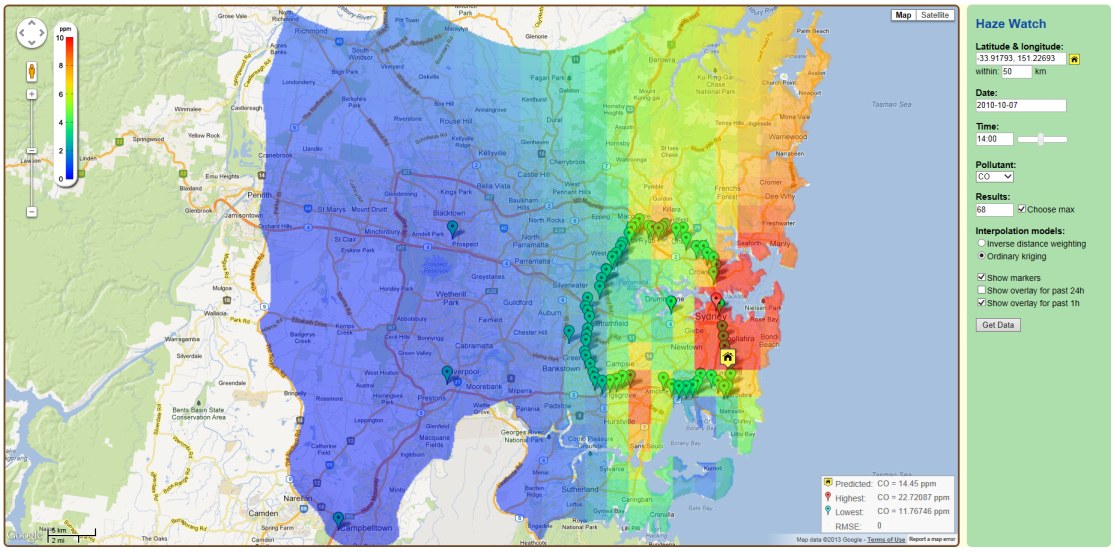


Figure 2.4: *Screenshot from Haze Watch web interface. It shows the average air quality per region.*

context of that activity. Therefore, utilising human mobility patterns in participatory sensing settings could provide insight.

Hazewatch (Sivaraman et al., 2013) is another low-cost participatory sensing system for urban air pollution monitoring in Sydney. Hazewatch uses several low-cost sensor units attached to vehicles to measure air pollution concentrations, and users' mobile phones to tag and upload data in real time. This project identifies the disadvantages of current approaches, i.e., using static sensors to monitor air pollution in cities, and aims to crowdsource fine-grained spatial measurements of air pollution in Sydney. Moreover, it aims to engage users in managing their pollution exposure via personalised tools. Specifically, HazeWatch, among others, suggests low pollution routes to users. However, the system has only been tested by a single vehicle going around Sydney as shown in Figure 2.4. This illustrates that some areas are not explored and inference is made based on just a few measurements (satisfying only requirement 2).

Another air pollution monitoring system following the participatory sensing paradigm is P-sense (Mendez et al., 2011). The ultimate goal of this project is to allow government officials, international organisations, communities and individuals to access pollution data to address their particular problems and needs. P-sense enables air pollution measurements at a finer granularity than what is currently achieved by static sensors in cities. It also enables users to assess their exposure to pollution according to the places visited during their daily activities. P-sense is easily extensible to allow the integration of existing data acquisition systems that could enrich the air quality dataset. P-sense consists of four main components: the sensing devices, the first-level integrators (i.e., the users), the data transport network, and the servers. The environmental data are collected by a number of sensors, such as gas, temperature, humidity, carbon monoxide,

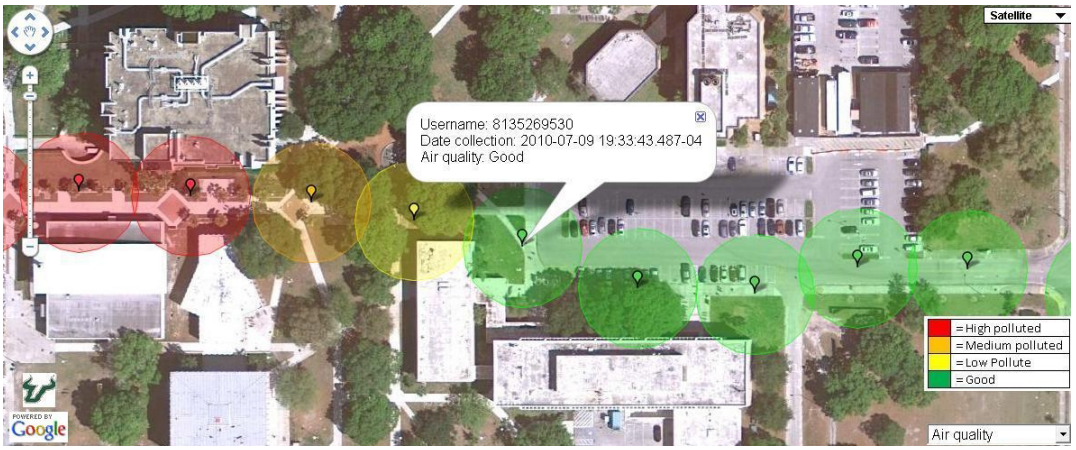


Figure 2.5: *Measurements in P-sense application*

carbon dioxide and air quality sensors integrated to mobile phones via Bluetooth. All environmental data acquired from those sensors are transmitted to first-level integrator devices, i.e., mobile phones. The phone is able to analyse data in real time, providing visual feedback to users. The first-level integrators transmit environmental data over the Internet (data transport network) to a dedicated server, where they are stored and processed. Users are able to connect to the server and get visual feedback for the data. However, there are several important research challenges to address before this system is deployed in the real-world, as highlighted in the work by Mendez and Labrador (2014). The most important are related to incentives, visualisation, privacy and security. Concretely, participatory sensing campaigns rely on people's contribution to succeed. This means that before a campaign starts it should consider how it will incentivise people to participate and contribute information. Generally, we cannot assume that people can take an unlimited number of measurements, but rather they have a budget or a cost for doing so. This cost should be minimised or stay within a specified budget while at the same time maximising the information learned about the environment (requirement 4). Moreover, in order to build up-to-date and fine-grained heat maps, both spatially as well as temporally, a coordination system is required that guides users when and where to take measurements (requirement 1). This is a crucial element missing from the proposed framework for this application. Privacy and security of information are not addressed in our work but rather reliability and robustness to malicious users, which are related, are of interest (requirement 5). In particular, a mechanism is needed to make the participatory sensing campaign robust to cases where malicious users want to contaminate the campaign by falsifying the values of their measurements.

CommonSense (Willett et al., 2010; Aoki et al., 2009) is a participatory sensing project that aims to design a mobile air quality monitoring system by conducting interviews with citizens, scientists and regulators in order to derive the principles and the framework for data collection and citizen participation in general. This approach can also help in identifying and capturing practical constraints that people face in participatory

sensing campaigns (requirement 4). Unlike the rest of the applications presented, they divide analysis into discrete mini-applications designed to promote and facilitate novice contributions. This approach allows the community members affected by poor air quality to engage in the process of locating pollution sources and exploring local variations in air quality. Based on the fieldwork, a set of personas was developed to characterise relevant stakeholders. Specifically, 'activists' are responsible for orchestrating actions and publicising environmental issues. 'Browsers' are interested in environmental quality but not directly involved in sensing. 'Data collectors' are novice community members who are likely to be affected by air pollution.

However, this approach requires significant effort in order to train users and it is not practical for large-scale participatory sensing campaigns (requirement 6). Data collected from the designated people are annotated to provide the context where measurements were taken. This is an additional burden to people's workload, and thus it is not favoured in this thesis. Importantly, however, this application utilises machine learning to make predictions for the unobserved locations. This is a key step and is also part of our requirements (requirement 2).

Besides relying on citizens to take measurements, CommonSense attempts to monitor air pollution by other means. In particular, in one study they run trials with air quality monitoring devices attached to the rooftops of street cleaners' vehicles in the city of San Francisco. These devices are associated with mobile phones that send data to CommonSense servers. In this way, a systematic coverage of a large city can be achieved as well as testing, refining and calibrating the system for future deployments. But, still, street cleaners have a small number of vehicles and their routine is limited to major roads, which results in unexplored areas, especially in residential zones. Moreover, since they pass from a point with a specific frequency, it might not be sufficient to provide temporal exploration of the area.

Overall, CommonSense attempts to monitor the environment with both street cleaners' vehicles and novice people contribution. However, even though there is a significant amount of fieldwork, which is essential in understanding the needs of citizens and elicits the requirements for a participatory sensing application in air pollution monitoring, no real-world trial with users using their phones was possible. This shows that there are challenges in running such campaigns that are related to the incentivisation of ordinary people. Moreover, since we want to minimise the burden caused to people by taking measurements, this framework is not suitable in our settings as it requires considerable human interaction to annotate measurements, and training of people to understand scientific language. At the same time, street cleaners' vehicles are not sufficient as they have predetermined paths of a constant temporal granularity which might not be sufficient. Also, sensors on street cleaners vehicle might be affected over time by other environmental phenomena such as humidity, high temperatures or dust.

OpenSense (Hasenfratz et al., 2014) is a project that aims to monitor air pollution in large cities, which was deployed in Zurich, Switzerland. More than 25 million measurements were collected in over a year from sensors attached to the top of public transport vehicles. Based on these data, land-use regression models (satisfying requirement 2) were built to create spatio-temporal pollution maps. One of the challenges that this approach aims to tackle is the lack of fine-grained spatio-temporal air quality data. Static sensors are expensive to acquire and to maintain, and thus only a few are placed in every city. The proposed system consists of 10 nodes installed on top of public transport vehicles that cover a large urban area on a regular schedule. The collected data are processed and predictions about the unobserved locations are made using the regression models (Mueller et al., 2016). Although this is a good approach for providing fine-grained spatio-temporal information about air pollution, measurements are only taken in roads where there are bus routes. Consequently, some areas of the city remain unexplored as in the application above. Also, as the authors point out since sensors are placed on top of buses they endure vibrations, heat, humidity and long operating times, which might lead to inaccurate measurements (Hasenfratz et al., 2014). Combining this approach with human participation could be more beneficial as people will be able to capture air quality in even more detail.

2.1.3 Other Environmental Monitoring Applications

Participatory sensing was utilised in other applications concerning environmental monitoring. In particular, Safecast is an open platform utilised to measure radiation in Fukushima, following the Fukushima Daichi Nuclear Power Plant disaster in 2011. The levels of radiation did not impose a direct threat to life, so people volunteered to assist authorities in measuring the radiation levels in the environment. Specifically, Safecast, was utilised to allow people to submit measurements taken using specialised equipment (Geiger tubes). A total of approximately 1000 devices were used globally, and environmental radiation data collection has seen an exponential growth between 2011-2016. This was a significant milestone in participatory sensing, as this was the first time it was successfully employed in the wild on such scale (Brown et al., 2016). However, even though there was a general target (monitor the Fukushima area), there was no system to suggest when and where volunteers should take measurements to efficiently monitor the environment. We believe that Safecast would greatly benefit from an intelligent coordination system, as duplicate measurements in the area will be reduced, while attention will be drawn in unexplored areas (requirement 1).

Participatory sensing was also utilised in more ambitious tasks. One promising application is called The Next Big One (Faulkner et al., 2011). This is a participatory sensing application for the early detection of earthquakes. These events are difficult to model and characterise a priori. Thus, this project utilises the accelerometer sensors available

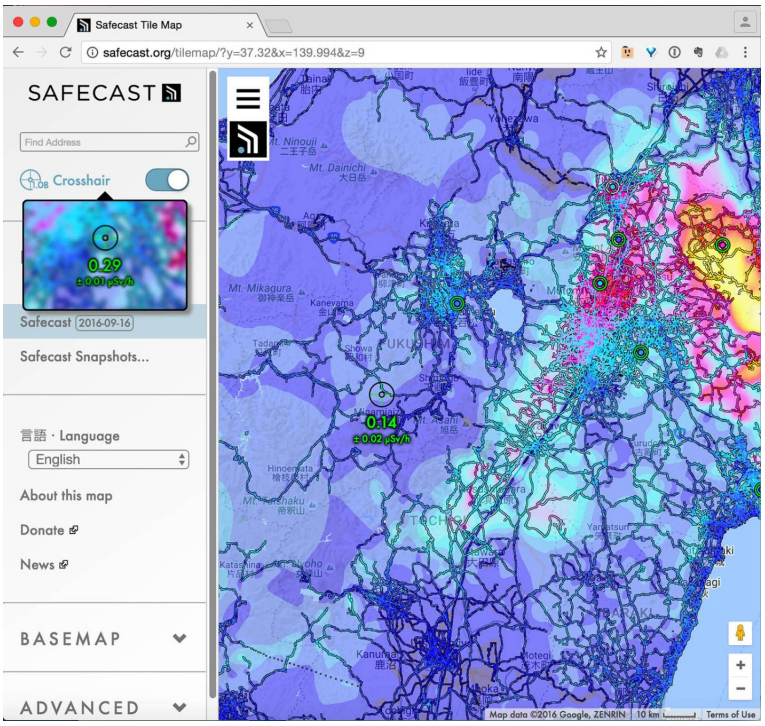


Figure 2.6: *Safecast online platform. It shows a contour map of radiation in Japan.*

on smartphones in a way to detect rare events and earthquakes. The focus of the study is to harness the power of the crowd, i.e., the wide availability of accelerometer sensors, for early earthquake detection. In shake table experiments, it is found that it is possible to distinguish seismic motion from accelerations due to normal daily use. However, for this application to be robust thousands of phones must be utilised. It is estimated that a million phones would produce 30 Terabytes of accelerometer data per day. We believe though, that intelligent coordination can alleviate the need for massive numbers of measurements by selecting only those that contribute the most in the campaign (requirement 1 and 4). In other words, coordination would enable taking measurements that maximise information about the environment but at the same time minimise the cost of taking those. Given a rough seismic model of the area of interest, an intelligent algorithm can suggest to people when to submit their accelerometer readings in order to contribute in the monitoring for seismic events in a way that maximises the information learned about the environment, while at the same time minimising the communication burden.

TrafficSense (Mohan et al., 2008) is a participatory sensing application for the monitoring of road and traffic conditions. In particular, this application relies on people carrying their smartphones with them while traveling and utilising their sensors like accelerometer, microphone, GSM radio, and/or GPS sensors to detect potholes, bumps, braking and honking. The effectiveness of the sensing functions was tested in the roads of Bangalore and it is shown that is it possible to monitor the roads using a variety

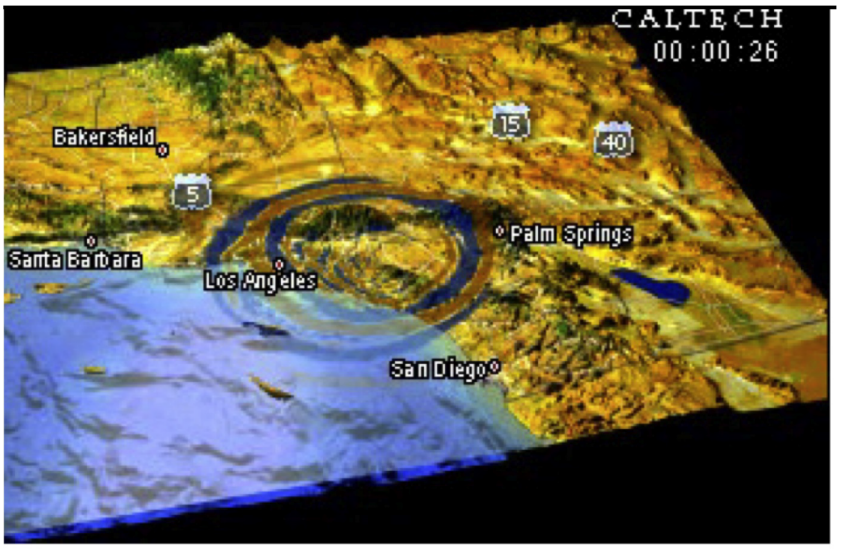


Figure 2.7: *Seismic information in graphical form*

of sensors built into the smartphones that users carry with them. In particular, the accelerometer was used for braking detection and to distinguish pedestrians from users stuck in traffic. Also, it is used to detect spikes that would suggest bumps in the roads. Audio was recorded using the phones' microphones in order to detect noisy and chaotic traffic conditions. Finally, GPS and GSM cell triangulation are used to localise users' positions.

While this is a promising way of monitoring the traffic, it is infeasible to utilise all these sensors of users' mobile devices throughout the day due to energy constraints (requirement 4).

2.1.4 Summary

Participatory sensing is a cheap data collection paradigm that is actively being researched. While delivering impressive results, there are still many open challenges. Even though it has been around for more than a decade it has not yet become the standard data collection technique. The most important challenges are the incentivisation of people to participate in those campaigns and contribute information on a daily basis as well as the energy management of the mobile devices. Crucially, people's measurements cannot be considered free, or at least they are limited. Moreover, none of the existing participatory sensing applications employs artificial intelligence techniques to improve the quality of the campaigns. In particular, there is a major gap for a participatory sensing framework that will consider incentivisation and energy constraints (requirement 4) to coordinate the measurements (requirement 1) that need to be taken in order to maximise the information learned about the environment. Concretely, it is crucial to develop intelligent algorithms that search for mappings from participants to

TABLE 2.1: *Table summarising the requirements of our work that each application fulfils*

Name	Req1	Req2	Req3	Req4	Req5	Req6
	Coordinating Measurements	Phenomenon Modelling	Quantifying Informativeness	Incorporating Constraints	Robustness	Scalability
NoiseTube						
GasMobile						
Citisense						
ExposureSense		✓				
HazeWatch		✓				
P-sense						
CommonSense				✓		
OpenSense		✓				
Safecast		✓				
The Next Big One						
TrafficSense						

spatio-temporal locations, given that users have a cost for taking those measurements or a daily budget. Furthermore, humans are typically predictable in their daily life, i.e., having their daily routine which is not exploited by any participatory sensing application. This information could be exploited to make relevant suggestions to users about taking measurements that are on or nearby their daily route. Table 2.1 summarises the requirements that each participatory sensing application meets. It is clear they fail to meet most of our requirements, which we believe are crucial to a successful participatory sensing campaign.

2.2 Environmental Phenomena Representation

A key challenge in monitoring environmental phenomena is to identify any spatio-temporal patterns in the observations that have been made. These patterns are used to make predictions (such as noise and air quality) about the locations where no observations have been made and about the future state of the world.

Regression is commonly used to accomplish this (Stranders et al., 2013; Tiwari et al., 2016; Schwager et al., 2017). This is a statistical process for estimating the relationship among variables and in particular understanding how the value of a variable will change by varying another variable. Two of the most common types used for environmental monitoring are Piecewise Linear Regression (Section 2.2.1) and Gaussian Processes (Section 2.2.2).

In our work, we use Gaussian Processes, as it is a non-linear non-parametric regression technique that can identify potentially complex spatio-temporal patterns in noisy observations. Also, Gaussian Processes have been used successfully in modelling spatio-temporal phenomena as shown in the work of Guestrin et al. (2005); Krause et al. (2006); Low et al. (2011a); Garg et al. (2012) and Ouyang et al. (2014). Specifically, they provide uncertainty estimations alongside the predictions, which can be used as a basis for utility functions.

In this section, we introduce Piecewise Linear Regression and Gaussian Processes in the context of modelling environmental phenomena, but we focus more on Gaussian Processes since it is our model of choice and we examine their properties used in our work. The model described is key for representing the environment as well as valuing information from measurements taken, which are part of the framework proposed in Section 3.1. Finally, we briefly discuss other approaches used for environmental monitoring.

2.2.1 Piecewise Linear Regression

Linear regression is commonly used in many practical applications because of its simplicity and computational performance. Environmental phenomena however, exhibit non-linear behaviour over space and time (Stranders et al., 2013). Thus, linear regression is not suitable for modelling the environment. However, Padhy et al. (2010) proposed the use of a variation of linear regression, called Piecewise Linear Regression, as an alternative that could be used in environmental monitoring. In particular, in order to model temperature and pressure, which have a non-linear relationship over time, they used Bayesian inference to decide whether each data point can be sufficiently explained by the current regression model or whether a new linear model is required.

Consequently, their environment is separated into a number of regions such that each region can be modelled by a linear regression. However, the parameters used in this approach increase with the number of linear regressions used. This makes it difficult to estimate them, which causes the model to be computationally expensive. Also, it is not certain where to start and stop in each linear regression, as these parameters need to be estimated again, which might result in an inaccurate model. Specifically, for each time step a set of linear regression models is required to model the environment. Besides the number of parameters of these models to be estimated, each model is only valid for a specific area, which also needs to be learned. Moreover, standard piecewise linear regression does not provide the confidence intervals over its estimates which are useful in order to measure the information gained at each spatio-temporal location.

2.2.2 Gaussian Processes Regression

Gaussian processes (GPs) (Rasmussen and Williams, 2006) are a class of nonparametric probabilistic models that are used in modelling spatio-temporal phenomena. For this kind of phenomena, the interest is not only on the value of the phenomenon (e.g. noise level or air pollution level) at the sensed location, but also at locations where no observations were taken (requirement 2). In such problems, regression techniques are used to perform these predictions. Although Piecewise Linear Regression can sometimes capture these relationships, as we have seen above, it is not flexible and it does not model the uncertainty of its predictions (Guestrin et al., 2005; Krause et al., 2006; Low et al., 2011a; Garg et al., 2012; Ouyang et al., 2014), which is useful to our utility function, as will be shown in Section 2.3. In contrast, Gaussian processes can capture more complex non-linear relationships and also provide a way to measure the uncertainty of those predictions through the notion of variance. Moreover, they are flexible in the sense that they can model different phenomena by using different covariance functions, as we will see below. These make Gaussian processes the most suitable tool for our work.

Given this, we provide an introduction to Gaussian processes, explain their properties and how they are of interest to our research. We start with some notation that will be used throughout this thesis.

Let $\mathbf{x}_*$ be the input vector (test data) and y_* its corresponding output value (prediction). In an environmental context, $\mathbf{x}_*$ would represent a single location on the map described by the spatial (x_1, x_2) and temporal coordinates (x_3). The output value (y_*) would be the prediction for the actual value (e.g. the air and noise pollution level) at that specific spatio-temporal location represented by $(\mathbf{x}_*)$. Also, let $y = f(x)$ be a process that denotes the relationship of a D-dimensional input vector $\mathbf{x} \in \mathbb{R}^D$ and an output variable $y \in \mathbb{R}$. When representing spatio-temporal phenomena, $D = 3$. In addition, let $\{(\mathbf{x}_i, y_i)|i = 1 \dots n\}$ denote a set of input-output pairs which represents past observations of the process f (training data). In terms of environmental phenomena, the training data would be the set of known spatio-temporal locations where measurements were taken in the past with the corresponding value of the measurement at that time. Finally, we denote the collection of n-dimensional output vectors y_i as $\mathbf{y}$. In other words, $\mathbf{y}$ is the output for n locations. Also, we denote the D-dimensional input matrix as X which is a collection of n $\mathbf{x}_i$ (i.e., n rows).

Gaussian processes (GPs) are defined as a collection of random variables, any finite number of which have a joint Gaussian distribution. In practice, a GP is completely specified by its mean function and covariance function (or kernel). A mean function $m(\mathbf{x})$ and a covariance function $k(\mathbf{x},\mathbf{x}')$ of a real process $f(\mathbf{x})$ are defined as follows:

$$\begin{aligned} m(\mathbf{x}) \quad &= \mathbb{E}[f(\mathbf{x})], \\ k(\mathbf{x},\mathbf{x}') &= \mathbb{E}[(f(\mathbf{x}) - m(\mathbf{x}))(f(\mathbf{x}') - m(\mathbf{x}'))] \end{aligned} \tag{2.1}$$

where $\mathbb{E}[X]$ is the expectation of a random variable X. Thus, we can write a Gaussian Process as follows:

$$f(\mathbf{x}) \sim \mathcal{GP}(m(\mathbf{x}), k(\mathbf{x},\mathbf{x}')) \tag{2.2}$$

The covariance function k plays a critical role in Gaussian processes. It determines the covariance between $f(\mathbf{x})$ and $f(\mathbf{x}')$. In other words, it specifies the relationship between two outputs with respect to their associated input. This enables GPs to identify the covariance between the outputs of training data, test data and the combination of both, which gives the predictive power of GPs as shown below. When $m(\mathbf{x})$ and $k(\mathbf{x}, \mathbf{x}')$ are known, they function as a prior over function f. However, when new observations are made, a GP can be updated to fit these data, increasing the prediction accuracy at the unobserved locations.

In GPs a key assumption is that data can be represented as a sample from a multivariate Gaussian distribution. This is expressed as:

$$\begin{bmatrix} \mathbf{y} \\ y_* \end{bmatrix} \sim \mathcal{N}\left(0, \begin{bmatrix} K(X,X) & K(X,X_*) \\ K(X_*,X) & K(X_*,X_*) \end{bmatrix}\right) \tag{2.3}$$

where $K(\cdot,\cdot)$ are obtained by evaluating the covariance function k for all pairs of columns. X represents the input vector of training data and X_* the input vector of test data. For simplicity in notation we set $K(X,X) = K$, $K(X,X_*) = K_*^T$, $K(X_*,X) = K_*$ and $K(X_*,X_*) = K_{**}$. So, we calculate K:

$$K = \begin{bmatrix} k(x_1,x_1) & k(x_1,x_2) & \dots & k(x_1,x_n) \\ k(x_2,x_1) & k(x_2,x_2) & \dots & k(x_2,x_n) \\ \vdots & \vdots & \ddots & \vdots \\ k(x_n,x_1) & k(x_n,x_2) & \dots & k(x_n,x_n) \end{bmatrix} \tag{2.4}$$

K_* and K_{**} as follows:

$$\begin{aligned} K_* &= \begin{bmatrix} k(x_*,x_1) & k(x_*,x_2) & \dots & k(x_*,x_n) \end{bmatrix} \\ K_{**} &= k(x_*,x_*) \end{aligned} \tag{2.5}$$

Thus, K is the covariance of the training points and K_* for all the pairs of training and test points. For the purposes of environmental monitoring we are interested in the conditional probability $p(y_*|\mathbf{y})$. In other words, given a set of observations $\mathbf{y}$ how likely is a certain prediction for y_*. Using the properties of the Gaussian distribution we obtain:

$$y_*|\mathbf{y} \sim \mathcal{N}\left(K_* K^{-1}\mathbf{y}, K_{**} K^{-1} K_*^T\right) \tag{2.6}$$

Thus, the best estimate for y_* is the mean of the distribution and the uncertainty about the estimation is the variance as shown below:

$$\begin{aligned} \mu &= K_* K^{-1} \mathbf{y} \\ \Sigma &= K_{**} - K_* K^{-1} K_*^T \end{aligned} \tag{2.7}$$

An important property that we exploit in this work is that the covariance of the prediction outputs $\mathbf{y}_*$ does not depend on the actual value of the observations $\mathbf{y}$ made, but rather only on the input vectors X, which are the spatio-temporal locations of those observations. This will enable us to run simulations forward in time, as it is not necessary to know the actual value of the measurements to estimate the variance at a future timestep. We will come back to this property when dealing with the algorithms developed that exploit it.

As we have already mentioned, the covariance function is of critical importance. However, we have not yet examined how it is expressed mathematically. A popular choice of such function is the *square exponential*. It is defined as follows:

$$k(x, x') = exp\left(-\frac{1}{2}|x - x'|^2\right) \tag{2.8}$$

Intuitively, this means that two points have a high covariance value if they are close to each other (spatially and/or temporally) and a low one if they are far apart. In other words, the covariance of two nearby observations is higher, which means they are more correlated compared to others. In practice, however, functions are smoother. Thus, when a new observation is made, those far apart will have little effect. To achieve this, the squared exponential function is extended with a positive constant factor (l), called the characteristic length-scale, which controls the smoothness of the process. In other words, it determines how much of an effect observations that are far apart have on the new one. Moreover, it is rarely the case that sensor observations are completely noise free, thus an additive independent Gaussian distributed noise ε with variance σ_n^2 is assumed. Specifically, instead of observing $f(x)$, $f(x) + \varepsilon$ is observed. In order to incorporate this information into the covariance function an extra term $\sigma_n^2 \delta_{xx'}$ is added to it, where $\delta_{xx'}$ is the Kronecker delta. This is defined as follows:

$$\delta_{xx'} = \begin{cases} 1, & \text{if } x = x', \\ 0, & \text{if } x \neq x'. \end{cases} \tag{2.9}$$

The effect of adding this is to increase the variance of the output variable by adding this term into the diagonal of the matrix. In addition, another parameter called signal variance σ_f is introduced as a maximum bound on the covariance of the function. The full version of the squared exponential covariance function is obtained by extending

Equation 2.8 as follows:

$$k(x, x') = \sigma_f^2 exp\left(-\frac{1}{2l^2}|x - x'|^2\right) + \sigma_n^2 \delta_{xx'} \tag{2.10}$$

In order to see the result of varying those parameters graphically, we refer the reader to Figure 2.5 on Page 20 of Rasmussen and Williams (2006). Intuitively though, the larger the l, the smoother the output function will be. When the noise variance (σ_n) is low, sharp variations in the output function are made in order to better explain the data. The larger the signal variance (σ_f), the larger the error bars will be when making predictions further from the observed points. Clearly, these free parameters ($\theta = \{\sigma_f^2, l, \sigma_n^2\}$), called *hyperparameters*, affect the covariance function, and thus, if arbitrarily chosen, the model, which is based on the covariance function, will not be able to represent the phenomenon.

Another popular covariance function is Matérn, which is commonly used for spatial statistics and geostatics (Jutzeler et al., 2014; Ouyang et al., 2014). Matérn is defined as follows:

$$k(x, x') = \sigma_f^2(1 + \sqrt{3}r)\exp(-\sqrt{3}r) + \sigma_n^2 \delta_{x,x'} \tag{2.11}$$

where $r = \sqrt{(x - x')^T \mathbf{P}^{-1}(x - x')}$, $\mathbf{P} = \begin{bmatrix} l_1 & 0 & 0 \\ 0 & l_2 & 0 \\ 0 & 0 & l_3 \end{bmatrix}$

and $\theta = \{l_1, l_2, l_3, \sigma_f^2, \sigma_n^2\}$ are the hyperparameters that need to be learned. Specifically, l_1 is the length-scale that controls the smoothness of the regression function over the x-axis, l_2 controls the smoothness of the regression function over the y-axis and l_3 over time. Intuitively, (l_1, l_2, l_3) captures the dynamism of the phenomenon in both the spatial and the temporal dimension. Also, σ_f^2 is the signal variance that controls the uncertainty of predictions made further away from the observed points, and σ_n^2 is the noise variance that controls the percentage of the data variation that can be attributed to noise.

However, there is no standard kernel to be used in each application. Domain specific knowledge should be taken into consideration when a specific kernel is chosen. For example, an ideal kernel for air pollution would be a non-stationary[5] one that considers air dispersion and mathematically captures the dynamics of air pollution particles. However, often this is difficult to capture and, worse, such kernels require many parameters and are computationally expensive to compute.

Estimating θ is equivalent to finding a value for θ that results in a high $p(\theta|\mathbf{x}, \mathbf{y})$. In practice, it is achieved by maximising the log marginal likelihood $\log p(\theta|\mathbf{x}, \mathbf{y})$. This is

[5]Non-stationary covariance functions allow the model to adapt to functions whose smoothness varies with the inputs. A stationary kernel is one where covariance only depends on distances between points (Paciorek and Schervish, 2004).

given by:

$$\log p(\theta|\mathbf{x},\mathbf{y}) = -\frac{1}{2}\mathbf{y}^T K^{-1}\mathbf{y} - \frac{1}{2}\log|K| - \frac{n}{2}\log 2\pi \tag{2.12}$$

A practical implementation of Gaussian process regression (GPR) is shown in Algorithm 1, which is adopted from (Rasmussen and Williams, 2006). Following through Al-

Algorithm 1 Gaussian Process Regression

1: **input**: X (inputs), $\mathbf{y}$(target), K(covariance function), σ_n^2(noise level), $\mathbf{x}_*$(test input)
2: L:=cholesky(K+$\sigma_n^2 I$)
3: α:=$L^T\backslash$(L$\backslash\mathbf{y}$)
4: μ:=$K_*^T\alpha$
5: $\mathbf{V}$:=L$\backslash K_*$
6: $\mathbf{\Sigma}$:=$K_{**} - \mathbf{V}^T\mathbf{V}$
7: $\log p(\mathbf{y}|X)$:=$-\frac{1}{2}\mathbf{y}^T\alpha - \sum_i L_{ii} - \frac{n}{2}\log 2\pi$
8: **return**: μ(mean), $\mathbf{\Sigma}$(variance), $\log p(\mathbf{y}|X)$(log marginal likelihood)

gorithm 1, we observe that it is somewhat different from what we have already discussed above. The reason is the computational complexity of the algebraic manipulations. In particular, explicit inversion of the covariance matrix K is very expensive, especially for a large amount of data. Instead of explicit inversion, Cholesky decomposition is preferred, which has a complexity of $\frac{n^3}{6}$. Cholesky decomposition exploits the fact that the covariance matrix Σ is symmetric and positive definite. Hence, matrix Σ can be decomposed into a product of a lower triangular matrix L and its transpose L^T.

This matrix manipulation makes Gaussian processes efficient to use in practice. In terms of monitoring spatio-temporal phenomena, they have been used in several cases. In particular, Guestrin et al. (2005) and Krause et al. (2006, 2008) have used GPs to find informative locations to place sensors, based on the uncertainty given by the model, in order to monitor spatial phenomena. Other work has looked into using GPs to model environmental phenomena in order find informative paths for teams of mobile robots (Stranders, 2010). In our work, we use Gaussian processes to model our environment and quantify the informativeness of making observations as shown in Chapter 3.

Moreover, we briefly introduce the concept of the heterskodetastic GP (HGP) model, which we use in Section 5.7. Basically, a HGP is similar to a GP but it allows variable noise across the input. This varying noise feature, commonly referred to as heteroskedasticity (Venanzi et al., 2013), is relevant to our participatory sensing settings where data are typically provided by devices with individual noise levels (i.e. the different levels of accuracy). Typically, regression models are evaluated in terms of Root Mean Square Error (RMSE), defined below:

$$RMSE = \sqrt{\frac{1}{|\mathbf{x}_*|}\sum_{i=1}^{|\mathbf{x}_*|}(y_i - y_i^*)^2} \tag{2.13}$$

This is a metric that captures the differences between the predicted and observed values in the model and it is an indicator of the accuracy of the model. As indicated in (Venanzi et al., 2013), each measurement can be modelled with an independent noise parameter. This has an effect in the covariance function of Gaussian Processes. More specifically, the new K calculated in Equation 2.4 becomes $K + \Sigma$, where Σ is a diagonal matrix populated with individual measurement noises. In other words, $\Sigma = \text{diag}(\hat{\theta}_1, \ldots, \hat{\theta}_n)$, where $\hat{\theta}_i = \frac{1}{\sigma_i^2}$.

2.2.3 Other Environmental Representation Approaches

Besides Gaussian Processes, other techniques have been applied in the field of environmental monitoring. This section will briefly analyse this literature in order to get a broad view of researchers' approaches in this domain and explain why these approaches are not suitable for our settings.

Rahimi et al. (2005) propose a nested stratified random sampling method for environmental monitoring. This approach does not focus on modelling the environment, but rather on a strategy for collecting data. This removes the predictive capabilities compared to Gaussian processes, but it is a simple solution that works well for spatial monitoring. In particular, this technique iteratively increases the sampling resolution, so that it enables reconstruction of phenomena using a systematic method for balancing accuracy with sampling cost. Concretely, the surveillance area examined is divided into a number of strata. Then, environmental variables are sampled in each of the strata with a number of samples proportional to the area of the strata. Then, the variance is computed and if it exceeds a specific threshold value, the stratum is partitioned. The process ends when the variance of each leaf of the stratum tree[6] is below that threshold. This sampling technique, however, does not capture the potential rapid temporal variations of the environmental variables. It focuses on snapshots of the environment and thus it is not suitable for monitoring spatio-temporal phenomena.

Szczytowski et al. (2010) make use of Voronoi diagrams to remove unnecessary samples from oversampled regions and generate new sampling locations in undersampled locations until an accuracy threshold value is met. This approach follows a similar philosophy as the one above. Initially, each sensor node (SN) checks locally for the fulfilment of accuracy requirements. Then, using Voronoi diagrams, the closest neighbour of each SN is detected. Next, each SN checks its neighbour's measurement for an accuracy threshold violation. If a threshold violation exists, then a virtual node (VN) is placed at the Voronoi edge in order to separate the neighbours whose measurement variance violates the required accuracy. Now, the Voronoi diagram is rebuilt taking into consideration the VNs added. This process iterates until accuracy requirements are met

[6] A stratum tree is a hierarchical tree of strata created by continuously applying the stratification process.

by adding the VNs. This technique captures the temporal dynamics of the environment but with a high cost since the algorithm needs to be updated when a single change in the environment is detected. Thus, it is impractical to use it for monitoring spatio-temporal phenomena.

Contrary to the previous approaches, the Kalman filter (Deshpande et al., 2005a,b) is a technique that is able to capture the spatio-temporal dynamics of the phenomenon. To do so, Kalman filters take into consideration the domain knowledge or the physics of a phenomenon to create a statistical model of it. For example, Kalman filters are successfully used in tracking and navigation applications, such as tracking a moving vehicle through GPS or an aircraft through a radar (Bizup and Brown, 2003). More specifically, based on the noisy prediction of the aircraft's previous position, its velocity and elapsed time and the new noisy observation of its position, it is able to accurately identify its current state. Also, Kalman filters are efficient in terms of communication cost since only new information is needed to improve the estimation or the tracking accuracy. However, for the various phenomena in which we are interested, the dynamics of the environment are considered unknown, thus a Kalman filter is less suited for these purposes. In contrast, Gaussian processes are able to identify and model the complex dynamics of the environmental phenomenon by using a suitable covariance function and learning the hyperparameters of it from the data (requirement 2).

For some environmental phenomena, such as noise or air pollution, other heuristic techniques have been utilised to monitor the environment. In particular, the proximity model (Jerrett et al., 2005) has been used to estimate the exposure of people to areas of high pollution. This model considers the distance of the subjects to the pollution sources, like roads and highway arteries. For example, studies have found significant positive correlation between pollution concentration and decreasing distances to schools from major automotive routes (Wyler et al., 2000). In another study, the distance from a child's home to the nearest main road was measured to understand the dynamics and the effect of pollutants to human health (Jerrett et al., 2001). In these studies, people have been interviewed about the traffic conditions in their neighbourhood. While this method is straightforward in terms of analysing long-term exposure to pollution, it has many limitations. First, it does not capture the air pollution in areas other than the roads, home, workplace and schools where the studies run. Second, it does not consider the temporal aspect of the phenomenon, nor the geographical and topological details of each area. Third, humans are susceptible to biases (i.e., recall bias). Specifically, important information might be omitted or altered by people due to their personal beliefs. For example, people might overestimate the traffic if they are in hurry to go to their work or underestimate it if they are going on vacations.

Another commonly used technique for environmental monitoring is land-use regression (Jutzeler et al., 2014). This is a special type of linear or non-linear regression model where traffic, altitude, wind, the surrounding buildings, street type, canyons and

other topological data are taken into consideration. However, it is often difficult to get the topography for all the areas that we are interested in. Given this, we are interested in a more generic approach that is able to model any environmental phenomena in different cities.

2.2.4 Summary

In this section we presented the state-of-the-art in modelling environmental phenomena. In particular, we have introduced and analysed piecewise linear regression, Gaussian Processes and other statistical techniques. We have argued that Gaussian Processes (GPs) is a useful tool for modelling spatio-temporal phenomena. However, the modelling capabilities of GPs come at a significant computational cost. On the other hand, GPs can model a wide range of phenomena by choosing or devising an appropriate kernel. Thus, GPs is an attractive technique, which we use in order to keep the generality and flexibility of our work.

2.3 Valuing Information

An important requirement described in Section 1.2 (requirement 3) is to be able to capture the informativeness of making an observation at a specific location at a specific time. Valuing potential observations is a fundamental step required for understanding the capability of an observation in improving situational awareness. Also, in the previous section we mainly focused on how Gaussian processes are able to model spatio-temporal phenomena. However, the quality of the representation is dependent on the spatio-temporal locations of where the measurements are taken. Thus, in this section we present the state of the art in valuing the information gained from taking measurements.

Let us consider a utility function $u(\mathbf{x}_L, A)$ which depends on the state of the world, $\mathbf{x}_L$, and chosen observations $A \subseteq L$, where L is a finite set of all the possible locations where observations can be taken. One sensing quality function commonly used (Shewry and Wynn, 1987) is the entropy criterion, where $u(\mathbf{x}_L, A) = -\log_2 p(\mathbf{x}_A)$ and $\mathbf{x}_A$ is a vector with the realisations of the observations made at locations A, i.e., $\mathbf{x}_A \sim X_A$. In this case, this is the Shannon entropy, $H(X_A)$, for a random vector X_A. Intuitively, maximising this criterion means picking observations that are the most uncertain (have the highest entropy). Entropy is a local metric, which means it takes into account the reduction in uncertainty at the location where the observation is made rather than the entire environment. Intuitively, entropy measures the peakedness of the probability distribution of a random variable. The more peaked the distribution is, the more confidence exists about its value, and thus not much information will be learned by making this observation. However, placing sensors iteratively at the locations with maximal entropy

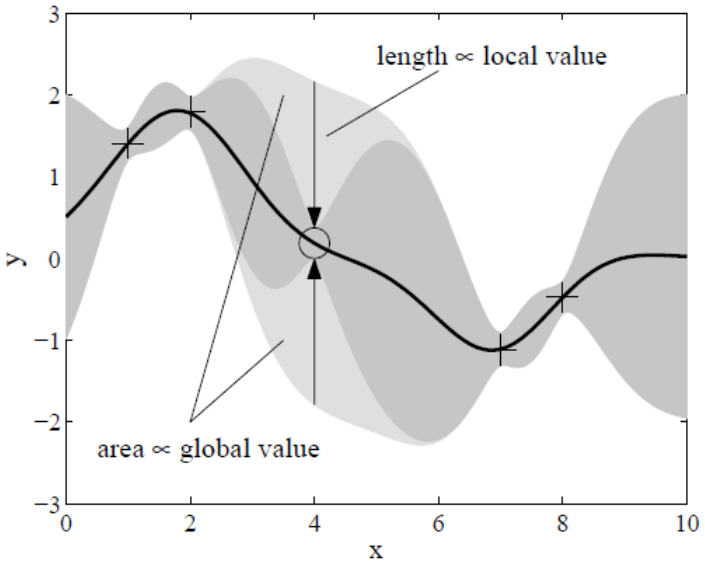


Figure 2.8: *GP illustrating the difference between global and local metrics when a new observation is made, adopted from (Stranders et al., 2013)*

results in placing a large proportion of those along the border of the environment where they are maximally uncertain about each other's measurements (Guestrin et al., 2005). Thus, the centre of the area of interest area remains unexplored.

Extending this criterion to cases where one is interested in reducing the uncertainty about a subset $B \subseteq L$ is expressed as follows:

$$u(\mathbf{x}_L, A) = -\log_2 p(\mathbf{x}_B) + \log_2 p(\mathbf{x}_B|\mathbf{x}_A) \tag{2.14}$$

This criterion is formally known as the D-optimality criterion (Chaloner and Verdinelli, 1995). It provides the *mutual information* between observation sets A and B and could be expressed as $I(X_B; X_A) = H(X_B) - H(X_B|X_A)$. Intuitively, it measures the uncertainty without making any observations minus the uncertainty after making the observations. This is a global metric, as it considers the reduction in uncertainty over the entire environment when making an observation. The difference between local and global metrics is shown in Figure 2.8. The figure illustrates the variance over the x-axis before taking any measurements in light shading, and the variance after a measurement is taken in dark shading. However, if one is interested in predicting the values of the unobserved locations, B could be set to be $L \backslash A$, meaning that the locations of interest are dependent on the observed locations A. This criterion is applicable in cases where the goal is to select locations that are maximally informative about the set of unobserved locations. In this slightly different setting, the goal is to maximise $H(X_{L\backslash A}) - H(X_{L\backslash A}|X_A)$ and is known as the Mutual Information criterion (Caselton and Zidek, 1984). This criterion can be represented as $MI(A) = I(X_A; X_{L\backslash A})$. This is shown to be a good metric for monitoring spatial phenomena (Guestrin et al., 2005). In particular, it is a better metric than entropy in terms of making observations more centrally by considering the effect

each sensor placement would have on the entire environment (Guestrin et al., 2005). However, this comes at a greater computational cost, since the effect of a set of new observations on the entire environment needs to be computed. However, due to the fact that we are interested not only on the spatial, but also the temporal, aspect of the phenomena, even if a measurement is taken at a timestep, we still consider that location as unobserved at the next one. As a result, the Mutual Information criterion is less applicable in our settings and the D-optimality Criterion is preferred in this thesis instead.

Another sensing quality function is the reduction of predictive variance (Krause, 2008). In other words, for a set B we choose $u(\mathbf{x}_L, A) = \sum_{s \in B} Var(X_s) - Var(X_s|\mathbf{x}_A)$ where $Var(X_s|\mathbf{x}_A) = E[(X_s - E[X_s|\mathbf{x}_A])^2|\mathbf{x}_A]$. This denotes the predictive variance of X_s after observing $X_A = \mathbf{x}_A$. This criterion is formally known as the A-optimality criterion (Chaloner and Verdinelli, 1995). However, Guestrin et al. (2005) have empirically shown that it is inferior to the aforementioned criteria in terms of finding informative locations. In particular, it is shown that the root mean error of the predictions at unobserved locations is higher when using this criterion for placing sensors rather than using the MI criterion.

Another approach is to consider a decision-theoretic metric. Specifically, instead of minimising uncertainty of a set of variables, the objective is to acquire observations that facilitate a decision-making process encoded by a parameter Θ. For example, let a parameter Θ encode the presence or absence of a high level of noise (in noise pollution monitoring settings). As in the cases above, a set of observations $A \subseteq L$ can be selected. When an observation is made, i.e. $X_A = \mathbf{x}_A$, a decision $d \in D$ is made. If the true state of the world is $\Theta = \theta$, then this decision results in a utility $g(d, \theta)$. If θ encodes the presence of high levels of noise the utility of choosing to take more measurements would be higher than not to.

In terms of Gaussian processes, the criteria above can easily be applicable since there is a direct relation between entropy and variance. In particular, the entropy of a random variable X_y conditioned on some set of variables X_A is a monotonic function of its variance and is expressed as follows:

$$H(X_y|X_A) = \frac{1}{2}\log(2\pi\sigma^2_{X_y|X_A}) = \frac{1}{2}\log(\sigma^2_{X_y|X_A}) + \frac{1}{2}(\log(2\pi) + 1) \tag{2.15}$$

2.4 Coordinating Agents

Our work is related to the agent coordination domain, as the overall purpose of our system is to coordinate the measurements taken by users acting as self-interested agents (requirement 1). Users typically have limited information about the environment and follow their own personal agendas. In the agent coordination literature, mobile agents,

such as autonomous ground vehicles (AGVs), autonomous unmanned aerial vehicles (UAVs) or unmanned underwater vehicles (UUVs) are often used to explore an environment or perform specific tasks in an area. Typically, coordination of teams of such agents is computationally intensive (not satisfying requirement 6) and the focus is on finding informative paths for a single autonomous agent (Marchant and Ramos, 2012; Binney et al., 2010).

In order to scale up, domain specific heuristics and clustering approaches are utilised to group spatially close sensing locations and thus reduce the search space (Singh et al., 2009; Stranders et al., 2010). However, coordination is typically shown only for a small team of agents (e.g., up to a dozen of autonomous robots (Singh et al., 2009; Ouyang et al., 2014; Low et al., 2011b; Stranders et al., 2010; Schwager et al., 2017; Tiwari et al., 2016; Reich and Sklar, 2006)). Thus, existing work does not scale to the settings we are interested in. Also, these techniques cannot easily be extended to consider probabilistic knowledge about the mobility patterns of participants or the willingness of the users to take a measurement, since the agents are robotic entities that do not have their own agendas, but rather follow computed paths on a graph.

In other related work, Stranders et al. (2009) deal with path finding for mobile sensors, considering both the spatial correlations of a phenomenon, as well as the temporal ones. They implement an adaptive receding horizon algorithm in a decentralised manner, which means that there is no central system that controls these sensors, but instead they autonomously decide what to do based on the information available to them by exchanging messages with other mobile sensors. The focus of their work is on decentralised approximations and dealing with reliability in the communication network between agents and permanent failure of agents' hardware, which is not a concern in this work. Rather, in our work an agent has a probability of being unavailable at a specific time, in which case users might ignore a notification to take a specific measurement at a given time, as well as a probabilistic model of their mobility patterns. Also, Stranders et al. (2009) assume that each agent has a specific radius within which it collects information and no underlying model exists, while in environmental phenomena the effect of a measurement can be captured by a probabilistic model that depends on the nature of the phenomenon. In other words, a probabilistic model can capture the development of the phenomenon in space and time, and thus is able to create heatmaps by interpolating between measurements as well as predicting into the future (see Section 3.3 for more details). Consequently, this enables us to be more accurate about the information gained when taking each measurement, as well as deciding when and where to take measurements, given the information collected by earlier measurements. Also, in many cases, the measurements taken are noisy, which can be captured by using a probabilistic model, creating accurate heatmaps about the phenomenon monitored. This can lead to better coordination of measurements in order to maximise the information about the environment. Furthermore, even though the algorithm uses a number of approximations

and heuristics, it is evaluated only on ten agents, which highlights the complexity of the solution.

Building on this, in other work Stranders et al. (2013) approach the problem of continuous multi-agent coordination by modelling the problem of space exploration by a team of mobile agents as a Markov Decision Process (MDP). They show that an approximation algorithm for solving MDPs can be used to continuously coordinate a small team of mobile agents (up to ten agents) for an infinite time horizon. However, it is not shown to work for larger teams of agents.

In other research, Partially Observable Markov Decision Process (POMDP) algorithms have been used in the context of agent coordination (Pineau et al., 2006; Hollinger and Singh, 2008). Firstly, POMDPs handle uncertainty in both action effect and state observability. Plans are expressed over information states instead of world states, since the world state is not observable. POMDPs form plans by optimising a value function, thus allowing the agent to numerically trade off between alternative ways to satisfy a goal, compare actions with different costs/rewards, as well as plan for multiple interacting goals. Also, instead of producing a sequence of actions, POMDPs produce a full policy for action selection. However, the state space grows exponentially with the number of variables that are considered in the selection problem. Also, the complexity of planning for POMDPs grows exponentially with the cardinality of the state space (Pineau et al., 2006). Thus, multiple agents, and multiple potential spatio-temporal locations where they can take measurements from, exponentially increase the state space of the problem. This makes the use of POMDPs infeasible for the settings we consider.

Drawing these together, even though the aforementioned algorithms solve problems that are related to coordinating measurements in participatory sensing settings, they are not applicable in our work mainly because they are not scalable to hundreds of participants. Also, in environmental monitoring of dynamic phenomena, the Markov property[7] might not hold. In particular, taking a measurement at a timestep might provide enough information such that no other measurement is required in the near future, given that the phenomenon is changing slowly over time. Therefore, it might be better for some people to wait many timesteps, given that they have a limited budget of measurements they can take in the future, before taking a measurement that would be of greater value in terms of providing more information about the phenomenon. Put differently, for a number of different measurements in space and time, we obtain different amounts of information about the phenomenon. This is not compatible with the Markov property, which requires that the future of the process depends only on the current state, i.e., measurements taken at a given timestep and not on the ones in the past. Consequently, the decision about when to take a measurement needs to be taken given the history of measurements taken so far in space and time and not just the last one.

[7]In general, the Markov property states that a reward at the time $t + 1$ is only dependent in the action a_t and state s_t.

Overall, in our work we follow the agent-based problem formulation applied in the multi-agent coordination problem, used in Stranders et al. (2013) for the coordination of measurements for environmental phenomena in the participatory sensing settings. This enables us to apply artificial intelligence techniques and exploit domain-specific knowledge to develop an efficient algorithm.

Having said that, there are two different approaches in coordinating agents, namely, offline and online approaches.

The purpose of offline algorithms is to pre-compute paths for a number of mobile sensors, such that the information collected is maximised, while at the same time placing bounds on their resources. These algorithms can run longer, since they are not expected to run in real time and potentially produce better results than online algorithms. However, when applied in highly uncertain environments they might not perform as well, since the environment in real time could be different from the one simulated (Singh et al., 2007; Meliou et al., 2007). In other words, they perform under the assumption that the characteristics of the environment are known beforehand and do not adjust their output in real time.

Online algorithms do not pre-compute the plan of the mobile sensors. Rather, the mobile sensors select observations on the go. That is, they are adaptive to their environment. This is also their main advantage, as mobile sensors can adapt to unknown environments. A group of algorithms, specifically the greedy ones (Resende and Ribeiro, 2010), can be reactive in terms of responding to the current state of environment at each time, in an effort to maximise the value of information at the given time. For example, such algorithms could instruct a mobile sensor to take a measurement when a threshold about the uncertainty of the value of the phenomenon (e.g. temperature or pollution level) is exceeded. This approach is used as a benchmark to our algorithms (see Chapters 4, 5). Overall, this category of algorithms produce an output very fast, as the output is utilised in real time. However, this comes with an accuracy tradeoff. In particular, in order to utilise domain knowledge, i.e., knowledge about the human mobility patterns, more computational expensive operations might be required. Thus, monitoring of the environment, following this paradigm, can be suboptimal.

Another category of online algorithms includes the receding horizon, or myopic algorithms (Stranders, 2010). This type of algorithm attempts to maximise the value of the information over a specific interval of time which is less than the length of the campaign and captures more than a single observation. However, when an unexpected event occurs, the algorithm recomputes the plan. Thus, the mobile sensors are able to adapt to their environment. However, if they are applied in a highly uncertain environment, frequent re-planning will be required, which will make the algorithm impractical to use.

In our work and more specifically in Chapter 5, we build on algorithms from both categories (online and offline) to both utilise knowledge known in advance as well as

adapt in real time when there is an uncertain and dynamic environment. Also, our work is benchmarked with algorithms explored in this literature and in particular the Patrol and the Myopic optimal algorithms. The patrol algorithm instructs an agent to take measurements at all the timesteps until a budget or energy is depleted. Myopic optimal takes finds the optimal set of measurements to be taken at any given timestep at a very high computational cost. In the next section, we introduce another relevant research area, where algorithms in these categories are also exploited.

2.5 Task Allocation

This work is also related to ongoing research in task allocation in the context of spatial crowdsourcing. Task allocation research is concerned with developing algorithms to mapping tasks, like taking a picture of specific products in different stores or reporting experiences in restaurants to people, given a number of requirements or constraints. Well-known commercial task allocation systems include: Gigwalk[8] and FieldAgent[9], which are crowdsourcing applications that allow businesses to recruit citizens to collect data and intelligence about their brand and the locations that matter most to their business. They essentially connect people who need information with people who can provide it. Citizens are compensated financially for their contribution, while businesses receive real-time feedback on their product and services. This literature is relevant to our work, as the purpose of our system is to allocate tasks to users, i.e., which measurements to take. However, there are substantial differences that are unique to environmental monitoring, which are discussed in the following sections.

2.5.1 Deterministic and Stochastic Human Mobility Patterns

Recent work by Chen et al. (2014) uses mobility patterns to effectively coordinate agents in crowdsourcing. The focus of their work is assigning agents to tasks based on their mobility patterns, so as to maximise the payoff of the tasks within a given time limit. However, no budget is associated with each user to correspond to the inconvenience or the incentive needed to execute the task. This is unrealistic in participatory sensing, because people cannot provide an unlimited number of measurements (requirement 4). Moreover, the tasks are assumed to be independent of each other and once they are executed, they are no longer available. This is not the case when monitoring environmental phenomena, where it is often important to revisit locations in order to keep track of the temporal variations of the phenomenon. Also, the reward gained when taking measurements of an environmental phenomenon is not easy to quantify. It cannot be captured by a fixed reward value as in task allocation. Rather, it should be calculated based on the

[8] `http://www.gigwalk.com/`
[9] `http://www.fieldagent.co.uk/`

model of the environment (which is examined in Section 3.3), since each measurement may be different in terms of the information it conveys. In other words, the utility in environmental monitoring is associated with what measurements are taken globally in space and time by the crowd. On the other hand, the reward for a particular task in crowdsourcing is usually independent of what tasks other people are executing, since each task has different characteristics, such as difficulty and type of task, which are not affected by other available tasks. Also, Chen et al. (2014) assume that humans have typically standard trajectories, which we also assume in this thesis.

Furthermore, even though simplified assumptions are made, such as the fact that users will always accept and perform the requested tasks, or that users have up to two alternative routes (Chen et al., 2015), the complexity of allocating people to tasks is still NP-hard. Thus, a range of offline greedy approaches, including a greedy construction heuristic and iterated local search, are utilised. Specifically, they first construct an initial solution as fast as possible by using a greedy heuristic and the quality of the initial solution is improved iteratively by employing an iterated local search (ILS), which is part of the stochastic local search (SLS) algorithm family (Hoos and Stützle, 2004). These algorithms are a number of high-performance local search algorithms that make use of randomised choices in generating or selecting candidate solutions for a given combinatorial problem instance. In particular, the algorithm performs four main actions: it swaps two agents with two task nodes if that improves the total remaining detour time for both agents. Next, it moves a task from an agent to another with the highest remaining detour time. Then, an unassigned task is chosen with the highest reward and the agent with the highest remaining detour time is selected to do it. Finally, an assigned task is replaced by an unassigned one with higher reward. All possible insertions are examined until the process exceeds a predefined number of iterations. That algorithm, however, is not applicable in our situation, as the problem we are addressing is different in the ways described above. However, similar to the work by Chen et al. (2014), we also build on heuristic approaches, and in particular, we propose a novel Stochastic Local Greedy Search (SLGS) algorithm (see Chapter 5).

2.5.2 Execution Uncertainty

As highlighted in (Ramchurn et al., 2009), users do not always successfully complete their allocated tasks, but they are subject to a probability of success that refers to the percentage of people successfully completing the tasks they are assigned to. However, most task allocation mechanisms do not take this into consideration. To address this, Ramchurn et al. (2009) take into consideration this limitation by developing trust-based mechanisms for robust and efficient task allocation in the presence of execution uncertainty. Even though this is an important concept, this trust mechanism is not

applicable in our settings where the main focus is coordination of agents for monitoring a spatio-temporal phenomenon. In particular, participants are assessed a priori in order to create a prior belief about their probability of success. This is infeasible to do in environmental monitoring since users have only a limited budget, an extra test measurement could potentially limit the ability of the system to achieve better coverage by taking measurements at more informative locations. Moreover, the reward of users when taking measurements, which is required in the aforementioned work, is not clearly defined, as people participate in participatory sensing campaigns for different reasons, as argued above, with no immediate reward value. In particular, there might not be an obvious reward as people might volunteer because they are interested in the cause of the participatory sensing campaign. Also, since the task in our settings is monitoring environmental phenomena, users cannot rate other users since the true value of the phenomenon is unknown and people have only local information about the environment.

On the other hand, in our work we assume that people will carry mobile equipment and receive a notification on their mobile phone regarding when to take a measurement. In that respect, related work has shown that only 83% of smartphone users engage with notifications on their device within five minutes of receiving them (Sahami Shirazi et al., 2014), which implies some desired measurements will be missed. In this work, we refer to execution uncertainty as the reliability of users.

2.5.3 Users' Trust

On top of execution uncertainty, people might not be trustworthy in terms of providing accurate measurements. In fact, the very openness of participatory sensing as a data collection approach enables the contribution of false data, i.e., the measurements' values are significantly different than the true value of the phenomenon monitored. In particular, people can act selfishly and exploit the system for their own benefit. Crucially, participatory sensing systems are prone to such *malicious* users' attacks (Mousa et al., 2015; Gadiraju et al., 2015). For example, a factory owner might falsify their readings to show normal air quality levels, while others may fabricate higher pollution measurements to affect the decision of authorities and policymakers about the development of parks and roads. There is a vast amount of literature on trust in multi-agent systems Keung and Griffiths (2009); Griffiths (2005), but in terms of participatory sensing this work is very limited.

One of the related work by Gadiraju et al. (2015) studies the prevalence of malicious users in crowdsourcing settings. In particular, they analysed the prevalence of malicious activity on crowdsourcing platforms and studied the behaviour exhibited by people on crowdsourced surveys. They did this by asking people to complete a survey using the CrowdFlower platform[10] about previously completed tasks. They collected answers

[10] `http://www.crowdflower.com/overview`

for a total of 34 questions, which were a mixture of open-ended, multiple choice and Likert scale questions from 1000 people. They analysed people's responses and their results show that approximately 25% of the users participating could be characterised as malicious. However, maliciousness depends on a number of factors and it is shown that different countries have different prevalence of malicious users. Mousa et al. (2015) highlight the issue of trust in participatory sensing settings and present how this problem is currently addressed. Specifically, one approach is to use Trusted Platform Modules (TPMs), which are hardware chips that reside on participants' devices and which ensure that measurements are taken by authentic and authorised sensor devices within the system. However, TPMs can control neither the software on a user's device nor the actual reading the user is taking. For example, a user can take a measurement in a controlled environment, where they can adjust pollution levels to the desired level in order to bypass the TPM mechanism.

Moreover, reputation systems have been proposed that require participants to rate each other or get rated by experts who compare their input against ground truth data (Jøsang and Ismail, 2002; Reddy et al., 2008). Other multi-agent approaches suggest formulating stereotypes (stereotype-trust and stereotype-reputation) about agents' behaviour given limited observability of their actions (Taylor et al., 2017). In other words, they assume that people with similar traits will behave similarly. This concept, however, is not easily applicable in participatory sensing settings, as it is difficult to assess whether a measurement provided is truthful or not. Moreover, people do not typically interact with each other about their readings, i.e. they are oblivious about the readings of other people, and thus subjective opinion about participants cannot be expressed.

Also, Reece et al. (2009); Bachrach et al. (2012); Irshad et al. (2017) provided methods to infer users' trust in crowdsourced classification and image labelling tasks. However, these classification methods are unsuitable for dealing with continuous spatio-temporal data, as in environmental monitoring applications, since dependencies over space and time need to be taken into consideration. In particular, the representation of the phenomenon must be derived as a continuous function accounting for the relationship among different measurements taken over space and time. Furthermore, in many cases ground truth data and experts might not be available.

Other work (Venanzi et al., 2013) has shown that probabilistic trust-based models can be built to minimise the effect of the contribution of noisy measurements. Specifically, Venanzi et al. (2013) develop a method for aggregating crowdsourced spatial estimates where the reports consist of pairs of measurements and precisions. In other words, each user submits a pair of their measurements and the associated precision, which captures their confidence that their measurement is correct. Then, Heteroskedastic Gaussian Processes (HGPs) (Section 2.2.2) are used to model trust of crowdsourced spatial data. In particular, the trustworthiness of each user is a hyperparameter of the HGP. That hyperparameter (t) is used as an uncertainty scaling parameter which provides the model

with the ability to flexibly increase the noise around subsets of reports associated with untrustworthy users. Then, by training the model with the reports gathered from the crowd, they are able to estimate the underlying spatial function and also learn the individual user's trustworthiness. However, the system presented in there focuses neither on the time domain, nor on coordinating measurements taken. Rather it focuses on how to fuse data from a variety of untrustworthy sources. Also, they require the precision of users as an input, which might not be feasible in scenarios where users do not have specific knowledge of the quality of the sensor they are using. Finally, malicious users will not provide their true belief about their precision.

2.5.4 Summary

In this section we presented literature on task allocation for spatial crowdsourcing settings. In particular, we discussed the three main topics in this area. The main focus is developing efficient algorithms for allocating tasks to the most appropriate people, given a particular goal. However, people may not always perform their allocated task for their own reasons. Consequently, probabilistic information about human behaviour could be considered when devising task allocation algorithms. Another major concern is that people cannot always be trusted to provide truthful measurements. Specifically, people may have their own agendas which involve altering the real picture of the environment.

2.6 Sensor Placement

Finally, our work is related to the sensor placement problem in the context of environmental monitoring. Specifically, it can be viewed as the task of placing a number of sensors that equals the number of users, in a dynamic environment where specific constraints are associated with the sensors. For instance, the number of sensors to be placed in the environment is changing at every timestep (depending on whether a user has some budget left or not), and the location of the sensors is constantly changing as humans follow their daily routine. Importantly, each sensor is associated with uncertainty about their future location and whether they will actually be able to take a measurement when instructed to do so. Since the nature of this problem is combinatorial, finding the optimal solution is computationally infeasible. In a seminal paper, Krause et al. (2008) show that the sensor placement problem is NP-hard. They also prove that the sensor placement problem has a desirable property, submodularity, that allows a greedy algorithm to provide specific guarantees about the approximation ratio of the solution provided. In particular, building on the work of Nemhauser et al. (1978), they show that using a greedy algorithm, the solution is always at least $1 - \left[\frac{(K-1)}{K}\right]^K$ times the optimal value and has a limiting value (i.e., as $K \to \infty$) of $(1 - \frac{1}{e})$, where K is the number of sensors placed. In the context of monitoring spatial phenomena,

the same property is exploited to produce a polynomial-time approximation algorithm, which is within $(1 - \frac{1}{e})$ of the optimum (Guestrin et al., 2005; Golovin and Krause, 2011). Specifically, Guestrin et al. (2005) greedily deploy a fixed number of sensors in an environment such that a submodular function, and in particular, mutual information between the chosen locations and the locations which are not selected, is maximised. The algorithm at each iteration adds the sensor which results in the maximum increase in mutual information until the desired number of sensors is reached. This is representative of a large class of algorithms that greedily select the next measurement that maximises an entropy-based criterion until a given budget is exhausted. More recent approaches attempt to make more efficient algorithms in terms of computational complexity, namely to make greedy faster by trading off utility gained (Mirzasoleiman et al., 2015). Moreover, (Mirzasoleiman et al., 2016) have focused on developing distributed algorithms whose performance approximates the standard greedy one. We believe, however, that the greedy algorithm described in Guestrin et al. (2005) is the most relevant representative of this class as it is shown to perform well in similar settings and it will be used as benchmark to our approach.

2.7 Summary

At the start of the chapter we introduced a number of participatory sensing applications to highlight the need of an intelligent coordination system. In particular, none of the applications satisfy all the requirements set for this work and we demonstrated that there is a major opportunity to make an important contribution in this area. Specifically, we highlighted the lack of an intelligent coordination system to guide people when and where to take measurements in order to maximise the information collected about the environment over time (requirement 1). Some of the applications model the environment and predict environmental values at unobserved locations (satisfying requirement 2) but they do not attempt to quantify information gained by taking measurements (requirement 3). Moreover, most of them do not consider uncertain human behaviour and the presence of malicious users (requirements 4 and 5), nor do they focus on scaling up the campaigns (requirement 6).

Next, we introduced the techniques and concepts utilised in our work, including how to represent the environment and value information collected by taking measurements. We have showed that Gaussian processes are powerful in terms of representing spatio-temporal phenomena, while at the same time providing a measure of certainty about predictions, which can be used to value measurements.

Finally, we positioned our work at the intersection of three main research areas, agent coordination, task allocation and sensor placement, all of which are closely related to the problem dealt with in our work. We provided an introduction to these areas and have

shown how they are related to our problem. In particular, the benchmarks that we will use in our empirical work in Chapters 4 and 5 are drawn from these three categories.

Chapter 3

Problem Description and Model

As we have seen in Chapter 1, the aim of this research is to provide the framework and algorithms in order to satisfy a number of requirements specified in Section 1.2. In this chapter we introduce the overall architecture of the framework we propose (Section 3.1). This includes features that others are working on but that are required to present the big picture of our vision in this area. Next, we formally introduce the problem of coordinating measurements for participatory sensing applications (Section 3.2.1) and the extended versions of the problem with relaxed assumptions in terms of human mobility patterns and user reliability (Section 3.2.2) as well as malicious user behaviour (Section 3.2.3). Then, we describe our Gaussian Process model that is used to represent the environment (Section 3.3). Finally, we present an example based on the basic problem formulation, to illustrate the main characteristics of our problem (Section 3.4).

3.1 Overall Architecture

Our framework shows how our coordination algorithm fits into the broader context of participatory sensing campaigns for environmental monitoring. In particular, it describes how to efficiently monitor an environment by coordinating measurements, taking into consideration available knowledge about the participants. This is informed by the examples presented in Section 2.1 and it is designed to address the main challenges in participatory sensing in general (Section 1.1). In particular, it captures our vision for a participatory sensing framework that satisfies the requirements set in Chapter 1. Figure 3.1 shows the overall architecture of the framework and illustrates how the components interact with each other. In particular, our framework consists of five core components:

- The main component is the *coordination algorithm*, which is the main contribution of this work. This component decides when and where each participant should take

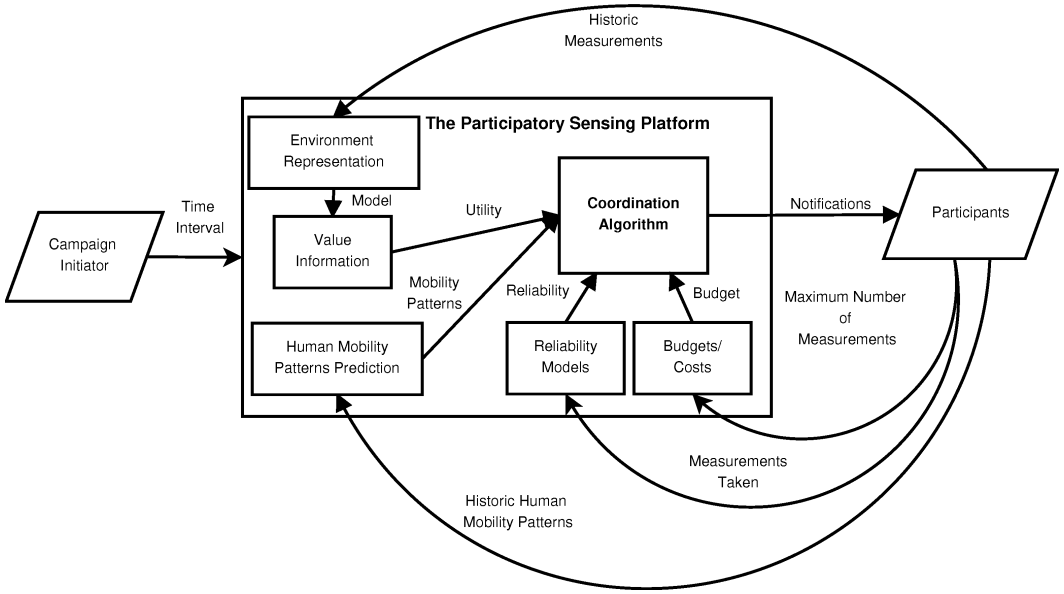


Figure 3.1: A conceptual architecture of an intelligent participatory sensing platform

a measurement to maximise information about the environment. More specifically, it produces a mapping from users to spatio-temporal locations.

- The *human mobility patterns prediction* component is a system for making predictions about the mobility patterns of the participants. This component provides probabilistic information about the future locations of the users, which is used to make decisions about when and where to take a measurement.

- The *budgets/costs* component captures the number of measurements an individual is willing to take, since users have a limited number of measurements they can take.

- The *reliability models* component captures the uncertainty related to individuals about whether they will actually take a measurement when asked to do so, as discussed by (Ramchurn et al., 2009). This model is built based on information collected from users based on their past behaviour in participatory sensing campaigns. This is similar to the notion of execution uncertainty described in the agent-based task allocation problem (Section 2.5.2).

- The components related to the environmental phenomenon supply the system with information about the environment being monitored. In particular, *historic measurements* from multiple users are fused together using a model that creates a representation of the environment (*environment representation* component). This, in turn, is used to value future measurements taken in terms of their information value (*value information* component) as in Section 2.3.

A participatory sensing campaign is initiated by a person, group or organisation interested in understanding an environmental phenomenon for a particular area (campaign initiator). The initiator is responsible for the recruitment of the participants for exploring a specific area and setting up the time interval of the campaign, i.e., the starting and ending date and time.

Anyone willing to take part in the campaign would own or be provided with a smartphone with Internet connectivity and specialised equipment, depending on the phenomenon and environment to be monitored. The participatory sensing platform is responsible for contacting the participants over the Internet in real time or in advance in order to suggest which measurements they need to take and at what time. This would take the form of notifications on their smartphone. It can be possible to agree a priori to a number of measurements in order not to be intrusive to the users' daily routine. The coordination algorithm is the most important component of the framework, which is based on the intersection of three research areas discussed in Sections 2.4, 2.5 and 2.6, and it is the main contribution of Chapters 4 and 5.

The participants are in a feedback loop, where they provide the platform with the measurements taken, as well as their mobility patterns and their budget and/or cost. In our work, we assume that participants have to explicitly take a measurement using their mobile device. This is reasonable, because although there are devices that are able to continuously take measurements this is usually associated with high energy cost. Also, this is required to ensure the quality of the measurements. For instance, measurements cannot be taken automatically from the device as it might be in users' pockets or bag, which might distort the actual measurements. Moreover, continuous measurements decrease the need for coordination and so we focus on settings where measurements are taken explicitly and maybe constrained by a budget. Concerning their mobility patterns, intelligent agents on participants' devices can monitor their behaviour and provide the platform with the mobility patterns as per the work of Sánchez-González et al. (2016). This might raise privacy concerns but this is a different research area which is currently active. Recent work has shown that it is possible to learn mobility patterns in a privacy preserving manner (Agadakos et al., 2017).

Also, each participant is associated with a budget (Chon et al., 2013), which, in our framework, can be given directly by participants or learned from their participation in previous campaigns with the assistance of the intelligent agents. The human mobility pattern prediction system infers their future mobility patterns for a specific time horizon producing a number of possible routines with associated probabilities (McInerney et al., 2013b; Baratchi et al., 2014b; Thomason et al., 2015). However, this human mobility pattern system is a separate active research area which is out of scope of this thesis.

The framework also considers that people are not guaranteed to take the measurement requested. As discussed in Section 2.5.2, 83% of users check their smartphone notifications within 5 minutes of receiving them. Thus, the intelligent agents on participants' phones can monitor the behaviour of the participants, as is commonly done in the crowdsourcing domain (DiPalantino et al., 2010), to provide a model for their reliability (shown in Figure 3.1) with respect to the system. Specifically, this model can be used to estimate the probability that a user will take a measurement when notified to do so.

Further, the environmental phenomenon can be modelled using a probabilistic technique as shown in Section 2.2. This could capture the spatio-temporal relationships of the phenomenon, interpolate over space in order to get the phenomenon's values in unobserved locations as well as predict the phenomenon's values into the future.

In this work, we do not focus on a complete implementation of the aforementioned framework, since each of the components is an active research area on its own. We rather focus on the algorithmic challenge of developing the main component, which is an efficient coordination algorithm that maps users to spatio-temporal locations in the environment. Our algorithm, however, is able to exploit probabilistic knowledge about participants' mobility patterns and consider budget and reliability constraints of each participant, as provided by the other components.

3.2 Problem Description

This section formally introduces the problem of coordinating measurements in participatory sensing for environmental monitoring. In particular, we focus on the problem that the *coordination algorithm*, shown in Figure 3.1, has to solve subject to budget constraints and the reliability of users.

3.2.1 Basic Problem Formulation

First of all, an environmental campaign is initiated to collect as much information about a particular phenomenon in an environment as possible. A campaign is a collection of observations $\mathbf{O}$ constrained on geography, duration, context and users such that $\mathbf{O}$ is collected participatively by a set of users $\mathbf{A}$. For the purposes of the problem definition, we use the agent abstraction instead of referring to humans, since we assume that they will act as mobile agents taking measurements when suggested.

An environment $\mathcal{E}$ is a continuous set of spatio-temporal locations (L, T) that the *campaign initiator* is interested in. This is defined by the spatial and temporal boundaries of the area and *time interval* of interest up to time E. A set of *participants* $\mathbf{A}$= $\{\mathcal{A}_1, \ldots, \mathcal{A}_M\}$ can take a set of discrete measurements (also called observations) within

the spatial boundaries of this environment within the time period of the campaign. Formally, the problem of monitoring an environment $\mathcal{E}$ in the multi-agent setting can be given by:

- A set of spatio-temporal observation coordinates $O = (L \cup \{\perp\}) \times T$. An element $o \in O$ is called an observation. The observation made by $\mathcal{A}_i$ at time t is denoted as $o_{i,t}$. The set of observations made by all agents at or before time t is denoted as $\mathbf{O}_t \subseteq O$. The set of observations made by all agents at time t is denoted as $O_t \subseteq \mathbf{O}_t$. We also denote by $loc(O_t) = \{l|(l,t) \in O_t\}$ the spatial coordinates of the observations made at time t. If loc is applied to a single observation, i.e., $loc(o_{i,t})$, it describes the spatial coordinate of the observation of a particular agent i at time t. If no observations at timestep t are made by an agent i, we denote this by the null observation $o_{i,t} = (\perp, t)$.
- A utility function $u : 2^O \rightarrow \mathbb{R}^+$ assigns a utility value to a set of observations. The value assigned by this function is based on the entropy, which is a way to measure information (given by the value information component in the framework in Section 3.1) and it is further discussed in Section 3.3. Here it is sufficient to say that the goal is to maximise the sum of utilities over the time period of the environmental campaign. However, each individual $\mathcal{A}_i$ has a specific *budget*, i.e., $B_i \in \mathbb{N}^+$, which is the maximum number of measurements that it can take within a day. Each agent can have a different budget. In this work, we assume that people only take measurements without deviating from their daily routine. As described in Section 3.1, people tend to contribute a limited amount of information in participatory sensing campaigns. Hence, we cannot assume that people can take an unlimited number of measurements but they rather have a budget and/or incur a cost when taking a measurement. We represent the budgets of all users with $B = \{B_1, B_2, \ldots, B_M\}$.
- A cost function $c_i : O \rightarrow \mathbb{R}^+$, assigns a cost, $c_i(l,t)$, to each agent i for asking them to make an observation at a location l and time t. We define the cost function for an agent i that makes no observation at time t to be $\forall t\, c_i(\perp, t) = 0$. The cost is an abstract representation of the incentive required for users to participate as well as the energy limitation of their devices and the annoyance caused by explicitly using their devices to take measurements and it is different for each participant.

We denote by $\mathbb{U}$ the total utility earned by all the agents at time t, which is given by:

$$\mathbb{U}(O_t) = \left(u(O_t) - \sum_{i=1}^{M} c_i(loc(o_{i,t}), t) \right) \tag{3.1}$$

where M is the total number of agents in the campaign. The utility earned so far up to time E from all the agents can be expressed as:

$$\mathbb{U}(\mathbf{O}_E) = \sum_{j=1}^{E} \mathbb{U}(O_j) = \sum_{j=1}^{E} \left[u(O_j) - \sum_{i=1}^{M} c_i(loc(o_{i,j}), j) \right] \tag{3.2}$$

The goal is to maximise the total $\mathbb{U}$ gained from all the agents throughout the campaign, i.e., by its end time E. Importantly, $u(O_j)$ depends on the measurements taken in previous time steps and thus we cannot assume independence. We also need to keep in mind that we know the cost functions and budgets of the agents a priori and we have to intervene at the right time to ask them to take an observation. Also, due to the regularities that human mobility patterns exhibit (McInerney et al., 2013b; Baratchi et al., 2014a), we assume that only a single location is predicted for each agent for a specific period of time (assumed only for the basic problem definition but relaxed in Chapter 5. In other words, each agent has a known set of locations ($l \in L$) that will be at specific timesteps ($t \in T$). This is realistic as most people have a routine that involves going to work five days a week and visit the same locations over time. Human mobility prediction systems can learn those patterns and provide us with users' future locations. However, this assumption might not always be true. In particular, in Section 3.2.2 we present an extended setting, where we assume that we have probabilistic information about each user's future location and that will form the basis for our algorithms in Chapters 5. Furthermore, people are typically reactive when they receive a notification on their smartphone (Sahami Shirazi et al., 2014). However, this might not always be the case, which is also captured in the formulation in the next section (Section 3.2.2) and considered in Chapter 5.

Finally, people participate in participatory sensing campaigns for a number of reasons, extrinsic or intrinsic incentives or social interest. However, people sometimes contribute false data. Specifically, there are cases where people may act selfishly or have their own agendas about altering the overall picture of the environment. This is discussed further in Section 3.2.3.

Given this nomenclature, the optimisation problem can be described as follows: We are looking for a decision $s : \mathbf{A} \rightarrow 2^O$, which determines which agents $\mathbf{A}$ should make which observations to reach an optimal solution to the problem. Formally, $S^* = arg\,max_s\, \mathbb{U}(\mathbf{O}_t)$. Note that a decision s gives us the set of observations $\mathbf{O}_t$, which is the union of all the observations taken from all the agents. In other words, different sets of observations have different total utility and the objective is to find the set such that the total utility over space and time is maximised ($\mathbb{U}(\mathbf{O}_t)$).

Consequently, a hypothetical solution to this problem should associate a number of agents participating in the campaign with a number of observations taken from each one of them. A solution could be $\{\mathcal{A}_1 \mapsto \{(\perp, 1), (l_1, 2), (l_2, 3), (\perp, 4)\}\} \cup \{\mathcal{A}_2 \mapsto$

$\{(\perp, 1), (l_5, 2), (\perp, 3), (l_7, 4)\}\}$ where $l_i \in L$. This is interpreted as the first agent needs to make an observation at l_1 at time step 2 and at l_2 at time step 3, whilst agent two should make an observation at l_5 at time step 2 and at l_7 on time step 4.

In addition, we introduce the following feasibility constraints on s:

1. An agent is said to make a null observation at each time step $(\perp, t)$ unless stated otherwise (l, t).
2. An agent can only make one observation at each temporal coordinate.
3. An agent can only make a total of B_i measurements per day.

3.2.2 Stochastic Extension

In this section, we modify and extend the formulation described above in order to introduce uncertainty to the problem and capture more realistic scenarios. Concretely, we assume that the mobility patterns of the participants are learned by a realistic system that is able to make predictions about future locations and provide a distribution over these locations regarding where each participant could be. Also, people could be unreliable in terms of providing a measurement when requested (similar to execution uncertainty discussed in Section 2.5.2). In order to capture this unreliability we define the following function:

A function $r : \mathbf{A} \rightarrow \{v \in \mathbb{R} | \, 0 \leq v \leq 1\}$ assigns a real number between zero and one to users representing their reliability (the reliability model component in Figure 3.1). This is the probability that they actually take a suggested measurement when requested to do so by the system. Each user has a personal reliability that is independent of other users. We represent the reliability for all users with $R = \{r(\mathcal{A}_1), r(\mathcal{A}_2), \ldots, r(\mathcal{A}_M)\}$. In our formulation, even if a user fails to take the measurement suggested, their budget is reduced, so as to avoid suggesting measurements to be taken by the same user if they are not willing to contribute. Intuitively this implies that the users will not be continuously notified to take measurements if they keep ignoring them.

Thus, the expected utility is defined as follows:

$$\mathbb{U}(\mathbf{O}_E) = \sum_{t=1}^{E} u(O_t) \tag{3.3}$$

where $u(O_t)$ is the utility gained from a set of observations made by participants at timestep t, given the effect of all the measurements taken before that. The coordination algorithm needs to decide when and where the citizens should make these observations to maximise this function, given a probability distribution over people's possible locations at each timestep and constraints of budget as well as user reliability.

Given this notation, the optimisation problem solved by this algorithm can be formulated as follows: map a set of participants to a set of measurements to maximise the expected utility over the period of the campaign, subject to individual budget constraints of participants. Formally, $S^* = arg\,max_s\, \mathbb{E}(\mathbb{U}(\mathbf{O}_E))$, where $s : \mathbf{A} \rightarrow 2^O$.

Importantly, the utility function remains the same as the goal for this extended problem remains the same; that is to maximise the total information about the environment over time. However, uncertainty about human mobility patterns and execution uncertainty makes solution to the previous problem infeasible for this one.

3.2.3 Coordination in the Presence of Malicious Users

As argued, users participating in the participatory sensing campaigns can be malicious. In our work, malicious users are those who try to mislead and disrupt the participatory sensing campaign by intentionally providing false, corrupted or fabricated measurements. This is also known as data poisoning (Mousa et al., 2015). In particular, in our settings malicious users can perform corruption attacks, which occur when the user deliberately provides corrupted or forged data.

In order to capture this behaviour, we define a maliciousness function $m : \mathbf{A} \rightarrow \{0, 1\}$ that assigns a binary number (zero or one) to users, which represents whether a user is malicious or not. This determines whether the measurement provided is the true value of the phenomenon being monitored or a noisy version of it. Each user has a personal maliciousness value that is independent of other users. We characterise all users, in terms of maliciousness with $\mathcal{M} = \{m(\mathcal{A}_1), m(\mathcal{A}_2), \ldots, m(\mathcal{A}_M)\}$.

However, the optimisation problem remains the same, and the expected utility is defined as follows:

$$\mathbb{U}(\mathbf{O}_E) = \sum_{t=1}^{E} u(O_t) \tag{3.4}$$

and the objective is $S^* = arg\,max_s\, \mathbb{E}(\mathbb{U}(\mathbf{O}_E))$, where $s : \mathbf{A} \rightarrow 2^O$.

Similarly to the problem above the utility function and the goal remains the same. However, the presence of malicious users will affect the solution of this problem as solutions to the previous problems can potentially underperform.

3.3 Modelling Environmental Phenomena

Given the introduction on Gaussian Processes in Section 2.2.2 and the definitions in Section 3.2, we now focus on probabilistically modelling the environmental phenomenon. This enables us to quantify the informativeness of measurements used in our utility

function (Equation 3.4). In order to model the environmental phenomenon, we first discretise the environment in a way such that a two-dimensional grid is created over space and the time is divided into hourly measurements (timesteps). Consequently, we say that locations $\mathcal{L} \subset L$ are the intersections of the grid and $\mathcal{T} \subset T$ are the timesteps. In our work, we convert longitude and latitude into UTM (Universal Transverse Mercator) format, i.e., meters, so as to be able to make calculations in the Euclidean space.

Each location $l \in \mathcal{L}$ and time $t \in \mathcal{T}$ is associated with a random variable $X_{l,t}$, that describes an environmental phenomenon, such as noise or air pollution. We use $X_{l,t} = x_{l,t}$ to refer to the realisation of a random variable at a particular spatio-temporal coordinate, which becomes known after an observation is made. In order to describe the phenomenon at time t over the set of locations ($\mathcal{L}$), given that some observations have been made in the past ($\mathbf{O}_{t-1}$), we use $X_{\mathcal{L},t|\mathbf{O}_{t-1}}$. Similarly, we denote by the random variable $X_{\mathcal{L},t|O_t}$, the environmental phenomenon over the set of locations $\mathcal{L}$ at time t given that a set of observations are made at time t (O_t). For simplicity in the notation, and unless stated otherwise, we use $X_y = X_{\mathcal{L},t|\mathbf{O}_{t-1}}$ and $X_A = X_{\mathcal{L},t|O_t}$. Similarly, the realisation of the measurements over the set of locations $\mathcal{L}$ given a set of observations is denoted by $X_A = x_A$. Given the nomenclature above, we can now model the phenomenon.

As explained in Section 2.2.2, the measurement of an environmental phenomenon can have a multivariate Gaussian joint distribution over all of their locations $\mathcal{L}$ and timesteps $\mathcal{T}$. The main advantages of GPs in environmental monitoring are that they can capture structural correlations of a spatio-temporal phenomenon, as well as providing a value of certainty on the predictions (i.e., predictive uncertainty). Crucially, it is sufficient to know the locations of the observations but not the actual value of the measurement, to get the variance over the environment.

Gaussian Processes provide the mathematics of the utility function we need to maximise, as shown in Section 2.2.2. Similar to the work by Guestrin et al. (2005), we want to maximise the sum of information obtained over time, which is captured by the entropy over the entire environment at a specific timestep minus the entropy that can be obtained by taking specific measurements in the next time step over the entire environment.

In other words, our utility function measures the reduction of entropy at all locations of the environment (global metric) by making a set of observations and it is proportional to the uncertainty without making any observations minus the uncertainty when observations are made. This is given by:

$$I(X_y; X_A) = H(X_y) - H(X_y|X_A) \tag{3.5}$$

In terms of Gaussian Processes, the conditional entropy of a random variable X_y given a set of variables X_A is expressed as follows:

$$\begin{aligned} H(X_y|X_A) &= \frac{1}{2}\log(2\pi e\sigma^2_{X_y|X_A}) \\ H(X_y|X_A) &= \frac{1}{2}\log(\sigma^2_{X_y|X_A}) + \frac{1}{2}(\log(2\pi)+1) \end{aligned} \tag{3.6}$$

Using a GP to model the environment, we develop an algorithm to exploit predictive uncertainty and the information metric designed.

3.4 Worked Example

In this section we present an example scenario illustrating our basic problem formulation given in Section 3.2. In order to present this example we have to introduce the following:

- A graph $G(L, E)$ that represents the layout of the agents' environment, where E are the edges that an agent can move on in order to reach spatial coordinates L. An example of such a graph is shown in Figure 3.2. In the real world, l_1, l_2, l_3, l_4 represent the locations where users take measurements from. The edges of the graph represent the potential movement or trajectory of the users.
- At each time step, an agent can move from one vertex to another via an edge.
- Agents have infinite budget.

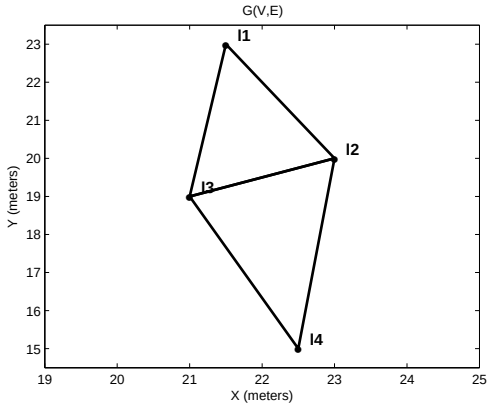


Figure 3.2: *Graph G*

For the purposes of this example we assume to have two agents ($\mathcal{A}_1$ and $\mathcal{A}_2$). The campaign starts at $t = 1$ and ends at $t = 2$.

At the first timestep, $\mathcal{A}_1$ is at location l_1 and $\mathcal{A}_2$ at location l_4 as shown in Figure 3.3. The set of observations made by both agents before the campaign begins is assumed to be empty ($O_0 = \emptyset$). At $t = 1$, O_1 will contain the observations taken by agents

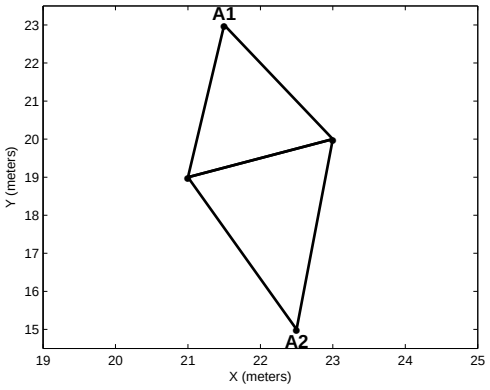


Figure 3.3: *Position of two agents on a graph at t=1*

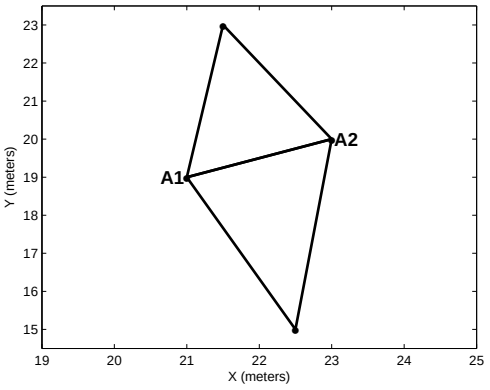


Figure 3.4: *Position of two agents on a graph at t=2*

at the first timestep or null ($\perp$) accordingly. Observations made at one location affect the entropy at others. For this example, we assume that the reduction in entropy at the locations is proportional to the distance of the locations to the location where the observation was made. We assume that by both making observations at their locations, i.e., $O_1 = \{(l_1, 1), (l_4, 1)\}$, we get the following:

$$I(X_{O_0}; X_{O_1}) = \begin{pmatrix} 10+0 \\ 5+5 \\ 5+5 \\ 0+10 \end{pmatrix} \tag{3.7}$$

where I is the mutual information between the two measurements, the first value in each row is the reduction in entropy in each location (l_1, l_2, l_3, l_4) caused by the observation taken by $\mathcal{A}_1$ and the second value from the observation taken by $\mathcal{A}_2$. In other words, we measure how much information was collected over the entire environment by taking those observations. For simplicity, and without loss of generality, we assume that we can just add the reduction of entropy caused by each agent. We can see that the reduction in entropy caused by $\mathcal{A}_1$ at l_1 is greater than elsewhere, where the reduction in entropy at l_4 is zero, which means taking an observation at l_1 does not affect l_4. Similarly, the reduction of entropy caused by $\mathcal{A}_2$ at l_4 is greater than elsewhere and zero at l_1. Next, we can calculate $u(O_1)$ as follows:

$$u(O_1) = 10 + 10 + 10 + 10 = 40 \tag{3.8}$$

Furthermore, we assume a constant cost value $c_1(l_1, 1) = 4$ for agent $\mathcal{A}_1$ making an observation at its location and infinity for elsewhere and $c_2(l_4, 1) = 6$ for agent $\mathcal{A}_2$. An initial simplifying assumption is that agents have their predetermined paths that they will follow, which are known to the system in advance, and nothing new learned on their way will affect this. This means that agents will never deviate from their route to make

an observation at a location other than their predetermined path and therefore the cost of moving between locations is zero. Having said that, $\mathbb{U}(O_1)$ can be calculated:

$$\mathbb{U}(O_1) = u(O_1) - \sum_{i=1}^{2} c_i(loc(o_{i,1}), 1) = 40 - (4 + 6) = 30 \tag{3.9}$$

At $t = 2$, both agents move to a different location as seen in Figure 3.4. $\mathcal{A}_1$ moves to l_3 and $\mathcal{A}_2$ to l_2. Both of them can again make an observation. However, observations were made just one timestep earlier at nearby locations. Thus, we expect that the reduction in entropy will not be as high. At this timestep, O_2 contains the locations of both agents and the corresponding time, i.e., $O_2 = \{(l_3, 2), (l_2, 2)\}$. Given that information is reduced at 1 for a measurement to their locations and only 0.5 elsewhere, $I(X_{O_1}; X_{O_2})$ is as follows:

$$I(X_{O_1}; X_{O_2}) = \begin{pmatrix} 0.5 + 0.5 \\ 0.5 + 1 \\ 1 + 0.5 \\ 0.5 + 0.5 \end{pmatrix} \tag{3.10}$$

As we can see, an observation made by $\mathcal{A}_1$ at l_3 results in greater reduction of entropy at that location but it is still just $\frac{1}{10}$ of what it would be achieved if no observation were taken before. Next, we can calculate $u(O_2)$ as follows:

$$u(O_2) = 1 + 1.5 + 1.5 + 1 = 5 \tag{3.11}$$

Given that the cost for taking a measurement remains the same for both agents, $\mathbb{U}(O_2)$ can be calculated:

$$\mathbb{U}(O_2) = u(O_2) - \sum_{i=1}^{2} c_i(loc(o_{i,2}), 2) = 5 - (4 + 6) = -5 \tag{3.12}$$

Consequently, $\mathbb{U}(\mathbf{O}_2) = 30 - 5 = 25$. In another case, if both agents take no measurements at the first timestep but both take one at the second, i.e., $O_1 = \{(\perp, 1), (\perp, 1)\}$ and $O_2 = \{(l_3, 2)(l_2, 2)\}$, the mutual information $I(X_{O_1}; X_{O_2})$ would be:

$$I(X_{O_1}; X_{O_2}) = \begin{pmatrix} 5 + 5 \\ 1 + 10 \\ 10 + 1 \\ 5 + 5 \end{pmatrix} \tag{3.13}$$

The observation made by $\mathcal{A}_1$ has a great impact on the reduction of entropy at l_3 and less on l_1, l_4. Concerning the reduction of entropy at l_2, we observe that this time it is not zero since the distance now from that location is not as large as it is from l_1 to l_4, which was the case in the previous situation. A similar case holds for $\mathcal{A}_2$. Intuitively this could be because both agents make observations at more central locations and thus, the observation of each one reduces the entropy at all locations of the environment. Thus,

$u(O_2)$ is calculated as follows:

$$u(O_2) = 42 \tag{3.14}$$

Consequently, $\mathbb{U}(O_2)$ can be calculated as follows:

$$\mathbb{U}(O_2) = u(O_2) - \sum_{i=1}^{2} c_i(loc(o_{i,2}), 2) = 42 - (4 + 6) = 32 \tag{3.15}$$

Clearly, the second case $\mathbb{U}(\mathbf{O}_2) = 32$ is much better since the utility gained is more than that gained in the first case. Having gone through this simple example, we can conclude that a coordination algorithm is needed to intelligently assign agents to sets of observations. In particular, both agents taking measurements at both timesteps results in less total utility rather than when the two agents take measurements only at the second timestep.

In the scenario described above, the system should have the following output: $\{\mathcal{A}_1 \mapsto \{(\perp, 1), (l_3, 2)\}\} \cup \{\mathcal{A}_2 \mapsto \{(\perp, 1), (l_2, 2)\}\}$. This would be the optimal solution to this problem among a total of 16 possible cases as seen in Table 3.1. The reason is that the total utility earned is higher when agents make observations at crucial positions that affect the entropy at more locations. When $\mathcal{A}_1$ is at l_3, there is a reduction in entropy at all locations. Similarly, when $\mathcal{A}_2$ is at l_2, there is also a reduction in entropy at all locations.

No.	Timestep		$\mathbb{U}(\mathbf{O}_2)$
	t=1	t=2	
1	$\perp, \perp$	$\perp, \perp$	0
2	l_1, l_4	l_3, l_2	25
3	$\boldsymbol{\perp, \perp}$	$\mathbf{l_3}$,$\mathbf{l_2}$	**32**
4	l_1, l_4	$\perp, \perp$	30
5	$\perp, l_4$	$\perp, \perp$	14
6	$l_1, \perp$	$\perp, \perp$	16
7	$\perp, l_4$	l_3, l_2	9
8	$l_1, \perp$	l_3, l_2	11
9	$\perp, l_4$	$\perp, l_2$	10.5
10	$l_1, \perp$	$\perp, l_2$	12.5
11	$\perp, l_4$	$l_3, \perp$	12.5
12	$l_1, \perp$	$l_3, \perp$	14
13	$\perp, \perp$	$\perp, l_2$	15
14	$\perp, \perp$	$l_3, \perp$	17
15	l_1, l_4	$l_3, \perp$	23.5
16	l_1, l_4	$\perp, l_2$	21.5

TABLE 3.1: *Different cases of agents making (or not) observations at each timestep.*

In the next chapters we use the model developed here as a basis to develop our algorithms. In particular, our algorithms minimise the entropy over the entire environment, which is proportional to the minimisation of the predictive uncertainty given by the GP model.

3.5 Summary

In this chapter we presented our proposal for a participatory sensing framework that relies on an intelligent coordination system to efficiently coordinate measurements in spatio-temporal settings (requirement 1). Specifically, we discussed the framework focusing on environmental monitoring applications and explaining how the framework could be utilised in such settings. Even though the central component of our framework is the coordination algorithm, it is able to accommodate all of the requirements set in Chapter 1.

Also, we formally introduced the problem of coordinating measurements in participatory sensing settings. This was described in three subsections that each one corresponds to the problem solved in each one of the subsequent chapters (Chapter 4 and 5 accordingly). We also presented how we modelled the environment in all of the settings we present in this thesis (requirements 2 and 3).

Finally, we demonstrated an example based on the basic problem formulation, to highlight the essence of the coordination problem, which is addressed in the following chapter (Chapter 4).

Chapter 4

Coordinating Measurements in Deterministic Scenarios

In the previous chapter, we formalised the problem of coordinating measurements in the participatory sensing domain. In this chapter, we present our proposed solution for deterministic scenarios and describe how we designed and performed our experiments. Our main contribution is an algorithm that addresses the basic problem formulated in Section 3.2, which satisfies requirements 1, 2, 3 and partially 4, as it considers the cost to individuals for taking a measurement but not uncertainty in human behaviour. This algorithm is useful when more information about users is available and thus there are fewer candidate solutions to be evaluated. In particular, this algorithm coordinates people to take measurements in an efficient way. The algorithm is heuristic as it strives for good performance, in terms of execution runtime, instead of optimality. Finding an optimal solution is computationally intractable, especially when large geographic areas are monitored for a number of days. This is because this category of optimisation problems is known to be NP-hard (Krause, 2008). Thus, the main challenges addressed in this chapter are the mapping of participants to spatio-temporal locations in order to explore the area of interest and avoid redundant measurements (challenge 1), while at the same time considering the individual costs of participants for taking a measurement (challenge 3).

In the following sections we present our algorithm (Section 4.1) and evaluate it by comparing it to the state of the art (Section 4.2), in terms of accuracy, total utility gained and execution time.

4.1 Local Greedy Search Algorithm (LGS)

In this section we present the coordination algorithm developed to address the problem formalised in Section 3.2.

First of all, in this work, we take into consideration the known mobility patterns of the participants in order to efficiently plan ahead in terms of when and where to take measurements. In particular, for each timestep, all participants are assigned a binary value. This value indicates whether or not the specific individual should take a measurement at that specific time. Since participants' locations are assumed to be known, time alone is sufficient to identify their spatial location. Thus, a full policy can be represented by a binary matrix, where columns represents participants and rows the timesteps of a participatory sensing campaign. The null policy, which is the policy where nobody takes any measurements, is represented by the zero matrix. For example, a two timestep campaign with two participants can be represented as a 2×2 matrix. In particular, assuming that the two participants take measurements at every timestep the following matrix can be created: $\begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix}$.

The goal of the algorithm, in the context of the design discussed above, is to produce a binary matrix, such that the utility, $\mathbb{U}(\mathbf{O}_E)$, earned over the entire campaign is maximal. In other words, the algorithm seeks to produce a mapping between a set of agents $\mathbf{A}$ and observations that have to be taken at specific spatio-temporal locations in order to maximise the difference between the utility and cost functions.

The algorithm uses a local search technique to reach a local maximum. In general, local search is a metaheuristic method for solving optimisation problems as discussed in Section 2.5. The idea is to find the best state among a set of possible states according to an objective function. The way local search achieves finding the best state is by starting at a random state and then moving to neighbours of that state, that are defined in the context of each particular problem, until the best state is found. Given this brief introduction to the local search technique, we attempt to both intuitively and formally describe the algorithm developed. The pseudocode for the algorithm is shown in Algorithm 2.

The algorithm initially starts with no measurements at all. So, initially a zero matrix is created (line 2). At this point, the utility is known to be zero by definition. Next, the algorithm checks what the total utility would be by adding a single measurement to the matrix, i.e., setting that position in the matrix to 1 based on the utility function defined in Equation 3.4. In this way, a subset p of possible measurements, other than cases where agents are known to be unavailable[1], are checked one by one (lines 6-22). This enables

[1]It is possible in the dataset for some locations to be missing, since a real dataset was used. This is further discussed in Section 4.2.3

the algorithm to look ahead in time and check what the utility will be if a measurement is taken in the future. The fact that only a random subset is checked reduces the overall runtime of the algorithm. If the utility of any of the produced matrices (line 23) is greater than the utility of the zero matrix (line 25), then the matrix resulting in the highest utility is selected (line 28). If the utility of the zero matrix is higher, then the algorithm stops (line 26)[2]. In the same fashion, assuming that the zero matrix does not produce a higher utility than any of the newly produced matrices, the algorithm keeps the best configuration so far (line 28). It then attempts to add another measurement to the matrix (line 10), and again all possible positions of the subset, other than the one already selected, are evaluated (line 6). However, it is possible to evaluate another policy (line 8) by removing one measurement previously selected, and thus backtracking to a previous iteration. This will enable the algorithm to avoid bad local maxima.

The algorithm works greedily, in the sense that it starts by considering the null policy, and when a measurement is chosen it cannot be altered unless a very bad choice is made. For example, if a measurement at the last position of the matrix produces the best result in the first iteration, it is set to one and it cannot be changed back to zero in later iterations unless removing a single observation from a previous matrix results in better utility. This approach limits the number of policies that are evaluated and thus leads to a faster runtime. The procedure continues until no further increase in the utility can be gained. In this chapter we assume that the budget of each agent is set to infinity and we only deal with the cost of taking measurements.

We have $2^{(M \cdot E)}$ possible combinations since we have an independent option for whether or not to take a measurement by an agent at any timestep. However, since our algorithm is greedy, its runtime is polynomial in the number of agents and timesteps. Specifically, it will run for a maximum of $(M \cdot E)$ iterations and at each one of them compute $(M \cdot E) - l'$ policies, assuming that $(|p| = |z|)$, where M is the number of agents, E the duration of the campaign and l' is the number of observations already chosen. Initially, $l' = 0$. The total number of iterations can be expressed as follows:

$$\frac{M \cdot E(M \cdot E + 1)}{2} \tag{4.1}$$

[2]It might be the case that any single observation at any timestep is very costly compared to the utility gained.

Algorithm 2 Local Greedy Search Algorithm (LGS)

input: E (timesteps), A (agents)
Initialise $M = |A|$, $maxU' = 0$, $S^* \leftarrow null\ matrix(E, M)$, $obsList = \emptyset$
for $k = 1\ to\ (M \cdot E)$ **do**
    $z \leftarrow null\ positions\ of\ S^*$, $newobs = S^*$
    $p \subset z$, $sz \leftarrow |p|$ ▷ p is randomly chosen
    for $l = 1\ to\ sz + 1$ **do**
        if $l = sz + 1\ \&\ k > 2$ **then**
            Change $obsList(k-2)$ to 0 in $newobs$ matrix ▷ Backtrack feature
        else
            Change l^{th} zero bit to 1 in $newobs$ matrix
        end if
        Set $C = 0$ ▷ Initialise cost to zero
        for $i = 1\ to\ E$ **do**
            for $j = 1\ to\ M$ **do**
                if $newobs(i, j) = 1$ **then**
                    $C \leftarrow c_j(i) + C$ ▷ calculate the cost for each agent and add it to the total cost
                end if
            end for
            $\mathbb{U}(O_i) \leftarrow [u(O_i) - C]$ ▷ calculate the total utility given what observations are made
        end for
        $s_l \leftarrow \mathbb{U}(\mathbf{O}_E)$
    end for
    Keep the maximum $\mathbb{U}(\mathbf{O}_E)$ of s_l in $maxU$ variable
    Add/Remove observation from $obsList$
    if $maxU < maxU'$ **then**
        return: S^*
    else
        Set S^* to be the best configuration
    end if
    $maxU' \leftarrow maxU$
end for
return: S^*

In order to illustrate the behaviour of the algorithm an example is provided, which is explained in more detail in Section 3.4. In this example, we assume a two timestep campaign with two participants. In the context of the framework developed, it is represented by a 2×2 binary matrix. As shown in Figure 4.1, the algorithm starts by evaluating the null matrix and then adds the single best measurement that produces the highest total utility. In this example, $\mathcal{A}_1$ taking a measurement at the second timestep, $(t = 2)$, has the highest utility and it is selected. Next, the algorithm determines whether taking another measurement can increase the total utility earned ($\mathbb{U}(\mathbf{O}_E)$), but no replacement of the selected measurement can be performed. Specifically, in this example, when $k = 3$ it is assumed that the third instance produces the best utility. For example, the first policy has a 14.5 total utility, the second policy has 12.5 and the third 32. Finally, the algorithm checks whether any of the policies at $k = 4$ produce a higher total utility. Assuming none of those produces a higher utility than what is already produced, the algorithm terminates and returns the following: $\begin{bmatrix} 0 & 0 \\ 1 & 1 \end{bmatrix}$. This is equivalent to:

$$\{\mathcal{A}_1 \mapsto \{(\perp, 1), (l_3, 2)\}\} \cup \{\mathcal{A}_2 \mapsto \{(\perp, 1), (l_2, 2)\}\} \tag{4.2}$$

in terms of the formulation in Section 3.2.

4.2 Empirical Evaluation

In this section, we evaluate the algorithms developed using real human mobility patterns and air quality sensor data. In the first part, we introduce our benchmarks, describe our hypotheses and the experiments performed to empirically understand how different algorithms perform in our settings. Finally, we discuss our findings.

4.2.1 Benchmarks

The algorithm developed is benchmarked against the state-of-the-art algorithms which are described below:

- **Greedy**: This algorithm checks which measurements should be taken in order to maximise the utility at each timestep, i.e., maximise $\mathbb{U}(O_1), \mathbb{U}(O_2), \cdots, \mathbb{U}(O_E)$ sequentially. It does so in a greedy way, i.e., select the single observation among the number of agents that maximises the utility but only at a specific timestep, instead of looking ahead as in LGS, and then the next best observation until no further improvement can be achieved for that timestep. The final policy produced, S^*, is the concatenation of the outcome of each timestep. The Greedy algorithm is the simplest approximation algorithm and it is used in determining where to place sensors in static environments (see Section 2.6).

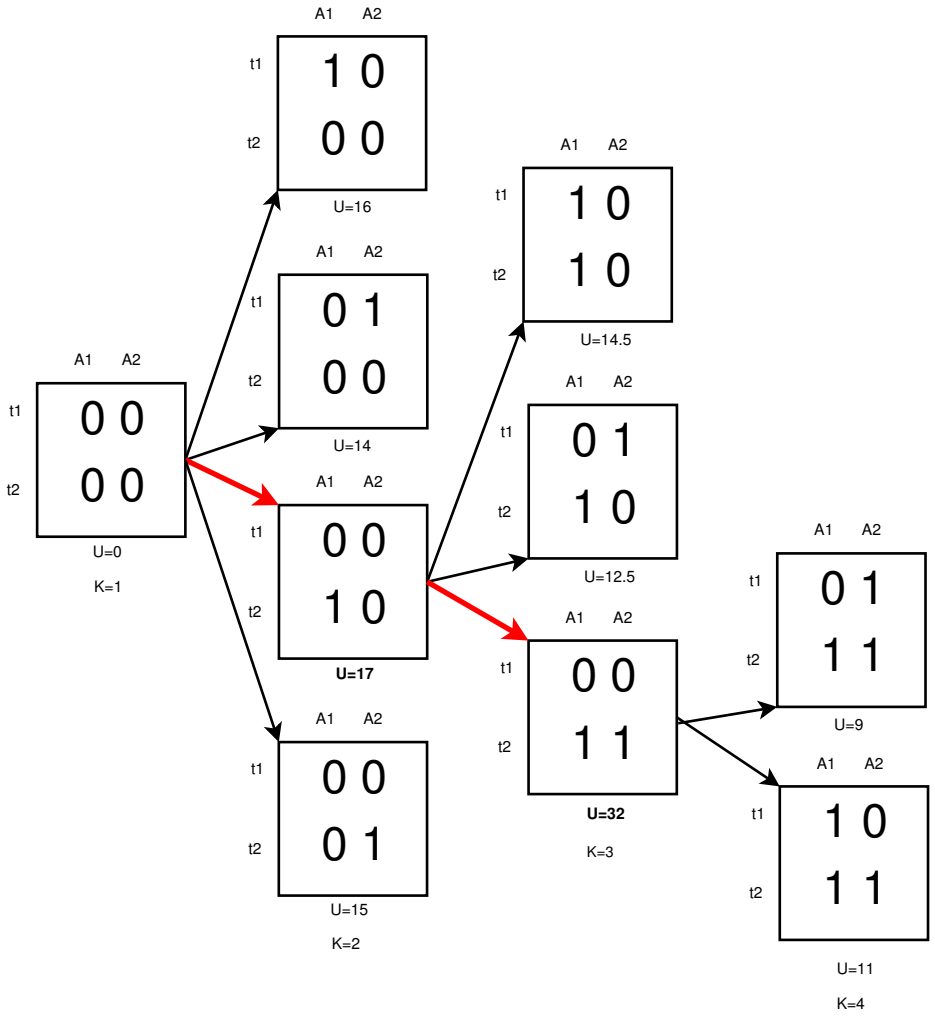


Figure 4.1: LGS *algorithm example.*

- **Patrol**: This algorithm assumes that measurements are taken at each timestep by all agents no matter the cost. It is an algorithm that replicates the behaviour of mobile sensors, i.e., patrolling an area in order to monitor environmental phenomena (see Section 2.4).

- **Myopic Optimal (MyopicOpt)**: This algorithm makes decisions myopically, i.e., considers only the current timestep, but it computes all the possible combinations of agents making an observation for a particular timestep. Thus, it finds the assignment of agents to observations that maximises utility $\mathbb{U}(O_t)$ for that timestep. Like Greedy, it produces a policy S^* that is the concatenation of the outcome of each timestep and it is used in the literature on coordinating agents (see Section 2.4).

- **Random**: This algorithm assumes that measurements are made randomly by agents throughout time. Specifically, it selects uniformly distributed measurements for each policy. It is an algorithm that creates a policy that could have potentially

been created by participants making local decisions, i.e., without coordination, in environmental monitoring campaigns.

- **Random100**: This algorithm runs 100 random policies and selects the best one of those.
- **Optimal (Brute Force)**: This algorithm produces the optimal policy S^* for coordinating measurements by evaluating all the possible combinations. This is only feasible to do in small-scale scenarios.

4.2.2 Experimental Hypotheses

Given the benchmarks above, we formulate the following experimental hypotheses:

- *Hypothesis* 1*:* The total utility earned by the LGS algorithm will consistently be higher than that of the Greedy, Patrol, Random100 and Random algorithms, irrespective of the number of agents participating.

  Outperforming Greedy is a result of the fact that LGS looks ahead in time, and thus is able to select measurements that should increase the total utility earned by the end of the campaign. Outperforming the rest is caused by the fact that the Patrol and Random algorithms ignore the costs of taking measurements and thus taking a measurement at every time-step or randomly results in a suboptimal behaviour.

- *Hypothesis* 2*:* The total utility earned by the LGS algorithm will be higher than Greedy, MyopicOpt, Patrol, Random100 and Random in most scenarios of varying dynamism.

  This is because LGS aims to increase the total utility by taking account of the dynamics of the environment. Even though Greedy and MyopicOpt are expected to perform better as the phenomenon becomes more dynamic, i.e., the phenomenon is almost independent at each timestep, LGS will still outperform them, because it is able to greedily add measurements in a similar way but at the same time look ahead and thus make decisions that lead to a higher utility over time.

- *Hypothesis* 3*:* The total accuracy of the LGS algorithm (measured in terms of RMSE[3], defined in Section 2.2.2) will be higher than Greedy, MyopicOpt, Patrol, Random100 and Random, irrespective of the number of agents participating.

  This is because the accuracy is correlated with the total utility gained. Better

[3]Whilst the utility captures the total information gained by a set of measurements, RMSE evaluates the overall accuracy in terms of the actual values of the phenomenon.

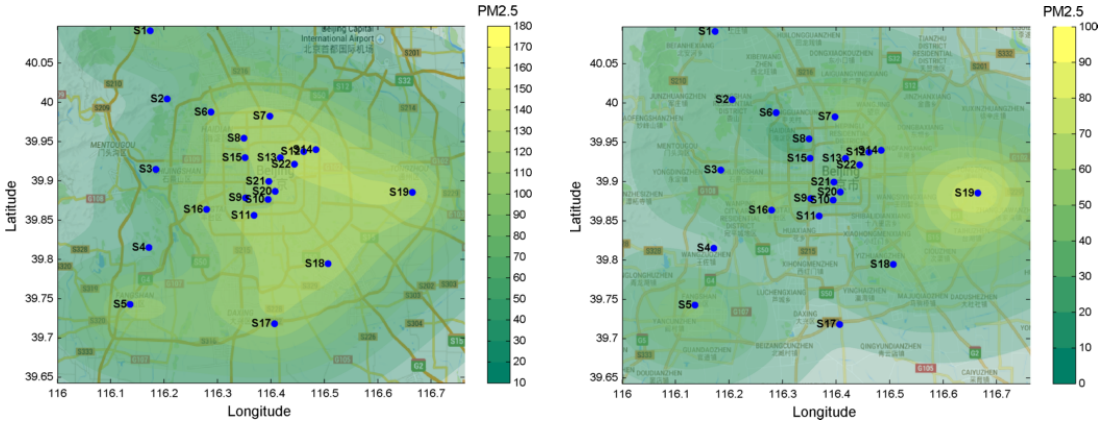
AQI Category	PM2.5 Level	Associated Health Impacts
0-50	Excellent	Little or no risk.
51-100	Moderate	Few hypersensitive individuals should reduce outdoor exercise.
101-150	Unhealthy for Sensitive Groups	Slight irritations may occur.
151-200	Unhealthy	Everyone may begin to experience health effects.
201-300	Very unhealthy	Healthy people will be noticeably affected.
300+	Hazardous	Healthy people will experience reduced endurance in activities.

TABLE 4.1: *Air Quality Index (AQI) for air pollution (`http://airnow.gov/index.cfm?action=aqibasics.aqi`)*

map exploration, i.e., collection of more information, will lead to better accuracy of the heatmap produced. Consequently, the LGS algorithm will perform better than the rest of the algorithms as argued in Hypothesis 1.

4.2.3 Experimental Setup

In order to empirically evaluate our algorithm, we compare its performance against the algorithms described above. In particular, we focus on air quality in terms of fine particulate matter (PM2.5) in Beijing, where the levels of air pollution are known to be high and thus it is of considerable interest to both the authorities and the people living there. Table 4.1 shows the air quality index for air quality. We use an air quality dataset (Zheng et al., 2013) which contains one year's (2013-2014) fine grained air quality data from static air quality monitoring stations in Beijing (see Appendix B). We use this data to train our GP model, and in particular learn the hyperparameters using the MLE technique. These include the dynamism of the phenomenon (l_3) and the smoothness over latitude and longitude (l_1, l_2). The sensors are scattered across Beijing and take measurements every hour.



(a) (b)

Figure 4.2: *Air quality measurement stations in Beijing overlaid by air quality measurements extrapolated by GP at different timesteps, demonstrating the spatio-temporal variations of air quality.*

Figure 4.2 shows the stations and the state of the environment represented by a GP for two different timesteps. As we can observe, air quality exhibits spatial variations, i.e., PM2.5 is different depending on where you are in Beijing, as well as temporal variations, i.e., it is different depending on the time of the day. Figure 4.3 shows the variance over the environment by taking measurements at the stations' locations. Also, the use of GPs helps with missing entries in the dataset as we are able to interpolate over the environment.

At the same time of using an air quality dataset, we are interested in the mobility patterns of people in the city. Ideally, these patterns are learned using a human mobility prediction system. In this chapter, however, we use deterministic mobility patterns. Specifically, we use data from the Geolife trajectories dataset (Zheng et al., 2009), which contains sequences of time-stamped locations of 182 people in Beijing (see Appendix A) over a period of 5 years (2007-2012). For our experiments, we extracted the patterns of 108 people over a two-month period, so as to get as much temporal overlap between collected patterns as possible. This is due to the fact that the dataset included empty entries for some users's locations over a period of time, or some of the users where not in Beijing for some or most of the time during this period. Due to the fact that we are dealing with real data, we filter out a large portion of those in order to bring them in a format that we can use for further experiments. Also, In order to test our system for more than 108 agents, we take patterns of different months from the same pool of agents' trajectories.

Participants are assumed to be equipped with the necessary equipment and they are able to take measurements when necessary if their spatial coordinates are available in the original dataset. Our system simulates human mobility patterns by getting the location of people every hour. However, as described in the problem description, taking a measurement involves a cost which is different for each agent. The cost, which is

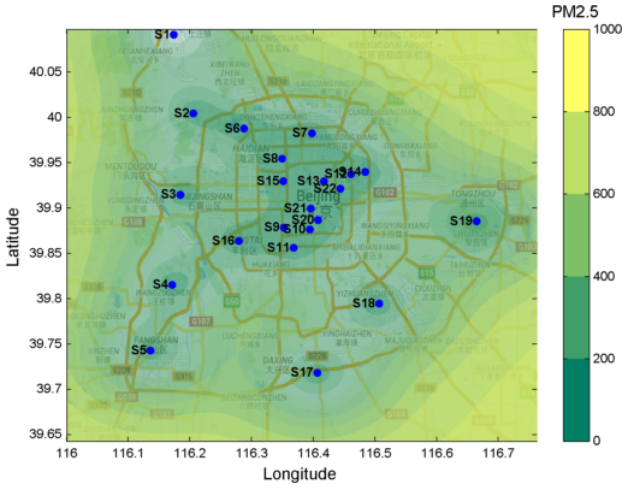


Figure 4.3: *Air quality measurement stations in Beijing overlaid by predicted uncertainty given by GP.*

proportional to the utility and is empirically learned, is randomly assigned to each agent. In addition, we assume there are peak and off-peak hours where measurements are more expensive or cheaper for all agents respectively. For instance, we speculate that during commute hours it is more inconvenient for people to take measurements as they are in a hurry. Thus, measurements to be taken during these hours (08:00 - 10:00 and 17:00 - 19:00) are more expensive by 30% than in other times. This way we capture the cost which would be associated with participants in a real deployment. However, this values could be change to what the campaign initiator wants or to a model that better captures the cost of people at different times of the day or the week. Also, each participant has a daily budget but in this work we assume it is infinite and focus on the cost of taking measurements.

The next section presents our findings from two different experiments. In the first experiment we simulate a varying number of agents in a 5-day campaign and compare the utility gained from LGS, Greedy, Random and Random100 to test hypothesis 1 (Section 4.2.4.1). The optimal algorithm and MyopicOpt are infeasible to run in city-scale scenarios, as they require more than a day's worth of computation. In fact, since the problem is combinatorial, the number of possible combinations for 250 agents for even a single timestep is not computable as it is $C_{\mathbf{A}}(|L|)$, where $|L| = 43 \times 43$ and $\mathbf{A} = 250$.

To make our system generally applicable, we experiment with a number of artificial environments by altering the hyperparameters (Section 4.2.4.2), and in particular l_3, which controls the dynamism of the environment. This change shows how our algorithm will potentially perform in other cities or for phenomena with other levels of dynamism. In the second experiment we simulate a varying degree of dynamism for a single day with 5 agents and compare LGS against all of the six benchmarks both in terms of utility gained as well as runtime (hypothesis 2). Experimenting with small-scale scenarios will enable us to compare our algorithm with the optimal one.

For both of these experiments we also empirically compare our algorithm's performance against the algorithms described above in terms of Root Mean Squared Error (RMSE) defined below:

$$RMSE = \sqrt{\frac{1}{|L|}\sum_{l=1}^{|L|}(y_l - y_l^*)^2} \tag{4.3}$$

where $|L|$ is the total number of locations of interest. This is a metric used typically to measure the accuracy of regression models and it captures the differences between the predicted and observed values (as discussed in Section 2.2.2). In our settings it is interesting to use this metric to capture the RMSE of the algorithms in practice (hypothesis 3) as it demonstrates the benefit of utilising a coordination algorithm in terms of the accuracy of the model.

Moreover, in order to obtain statistical significant in our results, we performed two-sided t-test significance testing at the 95% confidence interval.

4.2.4 Results and Analysis

In this section, we present and analyse our findings. In particular, we focus on the effect in the total utility and accuracy measured in RMSE when we vary the number of participants (Section 4.2.4.1). Also, we present the effect on utility and accuracy when varying the dynamism of the phenomenon as well as the overall execution time of the algorithms (Section 4.2.4.2).

4.2.4.1 Effect of the Number of Agents

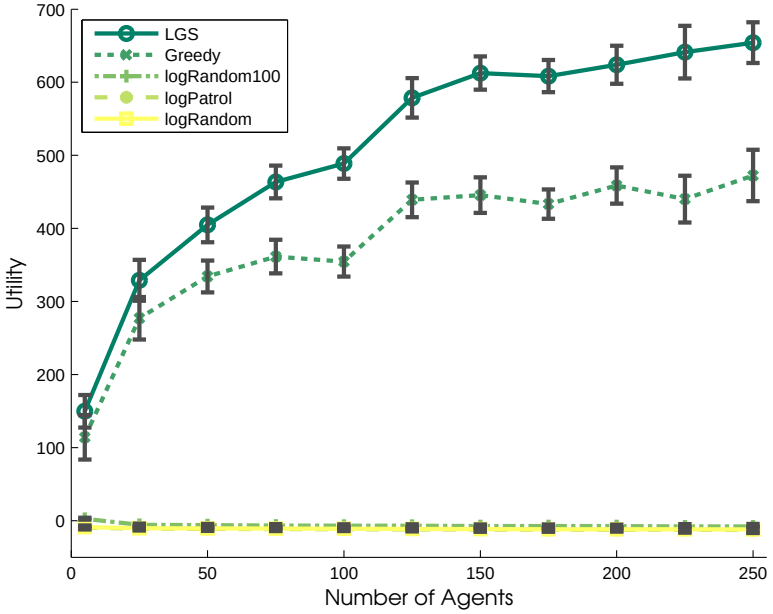


Figure 4.4: *Total utility gained for a 5-day participatory sensing campaign. The error bars indicate the 95% confidence interval.*

Figure 4.4 shows results of the performance of varying the number M of agents participating in the system. The dynamism in this experiment is fixed at $l_3 = 10.6$, which was found using the Maximum Likelihood Estimation (MLE) technique, which is a method of estimating the parameters of a statistical model[4]. In particular, the MLE technique was run a number of times with multiple initialisations. Then, the hyperparameters which resulted in the highest negative log marginal likelihood (see Equation 2.12 in Section 2.2.2) were selected. Estimating the hyperparameters of a Gaussian Process model is an active research topic, and number of approaches have been proposed (Rasmussen

[4]We use GPML v3.4 toolbox and in particular a nonlinear conjugate gradient method.

and Williams, 2006). However, in our work, we use the fastest one, which is commonly used in literature (Low et al., 2011a; Ouyang et al., 2014).

The results confirm our first hypothesis that LGS outperforms the rest of the algorithms. We can observe that LGS is 33.4% better on average than the Greedy algorithm. This is because LGS can look ahead in time, and thus make choices that will increase the total utility by the end of the participatory sensing campaign. Concretely, it is important to look ahead in time because of the temporal correlation of measurements. In other words, a future measurement at a specific location might be more valuable than taking a measurement at the same location at the present. This depends on the participants available at each timestep, as well as the cost of taking measurements. The Greedy algorithm lacks this ability, and thus it does not perform as well as LGS. The rest of the algorithms do not involve an intelligent element, and thus they act as a lower bound to Greedy and LGS. In particular, their performance is so bad that we show the logarithm of the absolute value of those because the cost of taking those measurements was more than the utility gained, resulting in large negative utility.

Figure 4.5 shows results of varying the number M of agents participating in the system. In this experiment we compare the algorithms in terms of the RMSE to benchmark the accuracy of the algorithms in terms of the actual air quality levels. The results show that LGS is significantly better than the other algorithms, confirming hypothesis 3. However, the Greedy algorithm is only marginally worse. This can potentially indicate that the covariance function used in our model (Gaussian Process) can be further improved. Specifically, in this thesis we used one of the most common covariance functions used in the literature (Matern kernel). However, modelling air pollution using Gaussian Processes is a separate research area (Guizilini and Ramos, 2015; Liu et al., 2016). This has a significant effect on the accuracy of air quality over the environment, which is why the benefit of LGS in terms of accuracy measured by RMSE is not that great.

4.2.4.2 Effect of the Dynamism of the Phenomenon

Figure 4.6 shows results of the performance of the algorithms when varying the time-scale (l_3), which controls the dynamism of the phenomenon. Originally, the time-scale was found to be ($l_3 = 10.6$) using the MLE technique. The smaller the time-scale, the more dynamic the phenomenon is. Consequently, as the time-scale approaches zero, each timestep is more independent from the other. Thus, MyopicOpt is similar to the optimal algorithm and Greedy performs near-optimally. Intuitively, the more dynamic the phenomenon is, the more information is gained by taking observations continuously (at every timestep). However, the dynamism of air pollution was at the scale of hours, which makes Greedy and MyopicOpt far from optimal. The results confirm our second hypothesis as LGS is better than the rest of the algorithms in all scenarios. However, as the time-scale approaches zero, LGS's performance tends to be similar to MyopicOpt

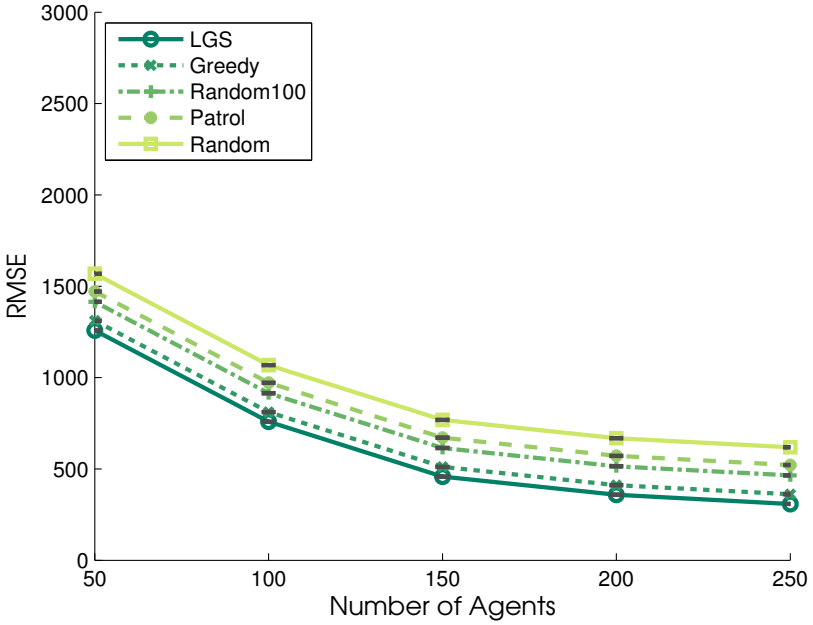


Figure 4.5: *Total RMSE for a 1-day participatory sensing campaign. The error bars indicate the 95% confidence interval.*

and Greedy. Also, the utility gained from LGS is near the optimal one. Figure 4.7 shows

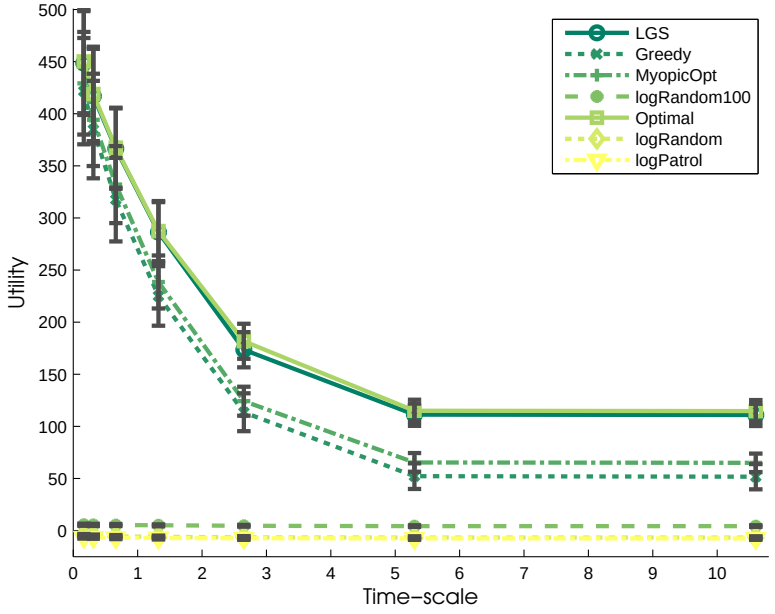


Figure 4.6: *Total utility gained for a 1-day participatory sensing campaign. The error bars indicate the 95% confidence interval.*

results of the time efficiency of the algorithms when varying the time-scale (l_3). As can be seen, LGS needs more time in dynamic environments as more measurements need to be taken and it is generally slower than Greedy, MyopicOpt, Patrol and Random algorithms. However, the optimal algorithm requires a lot more time. Specifically,

Figure 4.7 includes the natural logarithms of LGS, Greedy and the optimal algorithm's runtime.

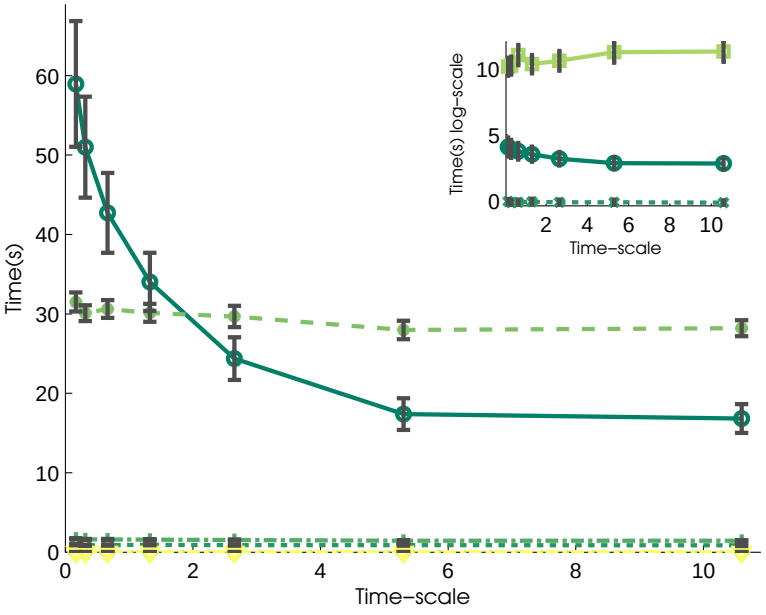


Figure 4.7: *Average runtime of the algorithms for a 1-day participatory sensing campaign. The error bars indicate the 95% confidence interval.*

4.3 Summary

In this chapter, we presented a novel coordination algorithm (satisfying requirement 1) that maximises the total utility gained over a period of time while at the same time minimising the cost incurred by taking measurements (satisfying part of requirement 4). In particular, we demonstrated how efficient the algorithm is compared to the state-of-the-art Greedy algorithm and Optimal approach. An empirical evaluation on real data showed that, (a) LGS is 33.4% better than the Greedy algorithm in terms of utility gained, (b) LGS is faster than the Optimal approach, (c) the dynamics of the environment affect the performance of the LGS algorithm and the total utility gained (the more dynamic, i.e., the lower the time-scale, the higher the utility and the computational time), but still LGS outperforms the benchmarks in all scenarios, and (d) LGS is better in terms of RMSE than the rest of the algorithms.

Chapter 5

Coordinating Measurements under Uncertainty and Presence of Malicious Users

In this chapter, we address the extended problem described in Section 3.2.2 as well as Section 3.2.3. Specifically, in the next section (Section 5.1) we present an algorithm that is able to coordinate measurements in the presence of uncertainty in people's routines as well as in reliability in terms of taking the measurements they are requested to. Following this work, Section 5.7.1 presents an algorithm that is able to coordinate measurements both in the presence of uncertainty and malicious users. However, it does so in expense of computational complexity.

5.1 Coordinating Measurements under Human Mobility Pattern and Task Execution Uncertainty

In this section, we tackle the extended problem described in Section 3.2.2 that introduces uncertainty in the willingness of people to take the measurements suggested, as well as in their mobility patterns. As discussed in Sections 2.4, 2.5 and 2.6, finding the optimal solution is computationally infeasible for realistic settings.

In this work, we focus on designing an efficient algorithm that outperforms the state of the art. In particular, contrary to the previous chapter, even though people are typically predictable in terms of their mobility patterns, only probabilistic knowledge might be available. Also, people might not always perform the action requested, which makes the problem more difficult by introducing this kind of uncertainty as discussed in Section 2.5.2. Thus, the main challenges addressed in this chapter are the probabilistic nature of human mobility patterns and human reliability (challenge 4), the budget

constraints (challenge 3), and the large number of participants (challenge 6). Given these challenges, our algorithm must be able to adapt under uncertainty and be scalable (requirement 4 and 6).

Our approach, the adaptive Best-Match algorithm or aBM (Algorithm 3), consists of two main components, the offline component, i.e., the Simulations for Scalable Searching (SiScaS) algorithm (Algorithms 4, 5 and 6), and the online component, i.e., the Matching algorithm (Algorithm 7). This approach enables the algorithm to run offline in order to find a good solution. However, since there is uncertainty related to human behaviour and mobility patterns, participants might not be in their predicted locations or might be unavailable to take measurements. Thus, an online component can adjust measurements in realtime.

In the next section (Section 5.2), we first intuitively explain how our algorithm works by providing a general overview and then provide the formal details. Next (Section 5.3), we describe the core components of the offline algorithm in more detail, while Section 5.4 presents the online component of the algorithm. In Section 5.7.3 we describe our experiments and show our results and in Section 5.6 we conclude by providing a summary for this chapter.

5.2 Adaptive Best-Match

Adaptive Best-Match or aBM (Algorithm 3), has an offline and online component. The offline algorithm is responsible for searching through the space of potential candidate solutions in order to produce a number of mappings of participants to spatio-temporal locations. Specifically, the algorithm makes small changes to the candidate solution (local search), in terms of when and where each user should take a measurement, and evaluates its performance by simulating the environmental campaign. The algorithm is explained in detail in Section 5.3. This algorithm, however, treats spatial clusters of people as a single entity, which speeds up the searching process. The Adapt algorithm (Algorithm 6), which is part of the offline component of aBM and is presented in Section 5.3.2, deals with finding people within a particular cluster who should take a measurement, in order to maximise the expected utility while at the same time saving budget for future iterations.

The next part of the algorithm (presented in Section 5.4) is responsible for acting in real time, matching the simulated output with the current situation. In particular, given the uncertainty in human mobility patterns, users are not guaranteed to be at the locations used in the simulations. Thus, an algorithm that handles the real-time situation is necessary. Our algorithm finds the best match between the simulation output from the offline algorithm to the real-time situation.

Our decision to exploit both offline and online components is due to the fact that the offline algorithm can find good solutions by making assumptions about the uncertainty related to mobility patterns and human behaviour. However, in real time, the actual location of the users can be observed as well as whether a user actually takes the suggested measurement or not. Thus, an online algorithm is required to adapt the measurements to be taken, which are produced by the offline component, in order to match the real-time situation and increase the total utility gained. For instance, if the offline component determined that a user who will be in a specific location should take a measurement, but in real time the user is not there, a nearby available user could potentially take the required measurement instead.

In this section, we present a high-level overview of our algorithm and then focus on each component and subcomponent of it.

In particular, the high level structure of our coordination algorithm (ABM) is shown in Algorithm 3. This algorithm shows that given the number of timesteps, the budget of the people and their reliability (line 1), a number of offline simulations (N) are made (line 2). For each simulation (N), a different mapping of users to spatio-temporal locations is produced (S), as defined in Section 3.2, which we represent with $S_{1,\ldots,N}$. Also, a number of spatio-temporal clusters are produced ($\mathcal{C}$), depending on the spatial locations of people, each of which is associated with the users that belong to it, their coordinates and the coordinates of the centroid of the cluster. Formally, $C = \{\mathcal{C}_{1,1}, \ldots, \mathcal{C}_{E,m}\}$, where the number of clusters is less than the number of agents $m \leq M$. Also, every $\mathcal{C}_{i,j}$ is associated with a number of users $A \subseteq \mathbf{A}$. In other words, it is a set of spatio-temporal clusters that include information about each participant's location that belongs in that cluster, their reliability and budget as well as the centroid in terms of coordinates of each cluster. Then, in real time (represented in lines 3 - 7), i.e., every timestep i, the MATCHING algorithm is called to find the best match between simulations and the real-time situation (line 4). Next, the selected users are notified to take the measurements required by the system (line 5). Finally, the environment is updated (line 6) with the information provided by the users.

Algorithm 3 Adaptive Best-Match (ABM) Algorithm

1: **input**: E (timesteps), B (budget), R (reliability)
2: $S_{1,\ldots,N}, C_{1,\ldots,N} \leftarrow$SiScaS$(E, B, R)$ ▷ Simulations running offline
3: **for** $i = 1$ to E **do**
4: $S_i^* \leftarrow$ Matching$(E, i, B, S_{1,\ldots,N}, C_{1,\ldots,N})$ ▷ Online mapping of users to measurements
5: Notification(S_i^*) ▷ Notify selected users to take measurement
6: $\mathcal{E} \leftarrow$ Update(S_i^*) ▷ The environment is updated with the new information obtained by the measurements taken by users at this timestep.
7: **end for**

5.3 Simulations for Scalable Searching (SiScaS)

The Simulations for Scalable Searching (SiScaS) algorithm is a critical component in our work, as it is responsible for a number of functions including calling the Stochastic Local Greedy Search (SLGS) algorithm, which is described in Algorithm 5.

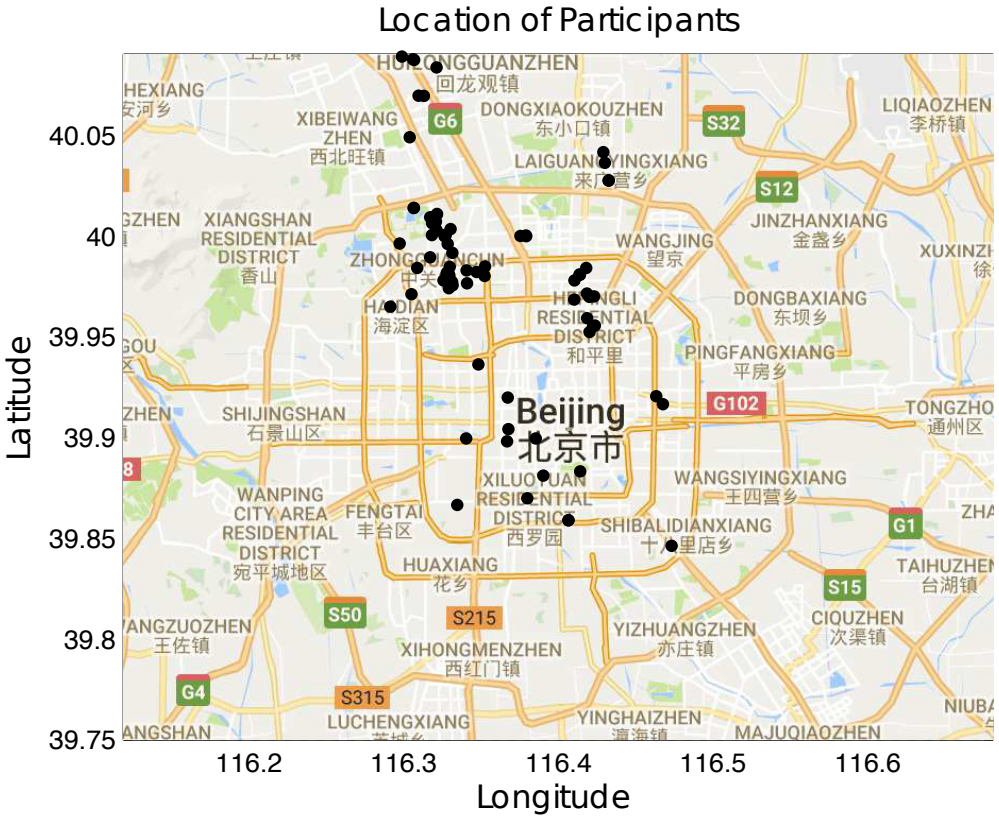
The SiScaS algorithm is shown in Algorithm 4. In particular, this algorithm is responsible for sampling from the human mobility patterns distributions, provided by the human mobility prediction component in Figure 3.1 (line 4), in order to get the possible locations for each of the participants. It also clusters people in spatially correlated groups for all the timesteps using a well-known clustering technique called DBSCAN (Ester et al., 1996)[1] (line 5). DBSCAN enables the grouping of people based on the distances between each other and is independent of the shape of the cluster. Also, DBSCAN, in contrast to other clustering techniques, does not require an explicit input of the number of clusters that should be formed. Rather, it requires the minimum number of points needed to form a cluster, as well as a distance threshold that prohibits points far apart from each other belonging to the same cluster. Consequently, people close to each other are said to belong to the same cluster, and thus can be treated as a single entity, which is crucial in scaling up the number of participants in the campaigns. In particular, each cluster could have a budget as the maximum of the people belonging to that cluster. This is feasible since, in our case, measurements taken at the same spatio-temporal location contribute the same information to the campaign. Since at each timestep people can be in different locations, the algorithm produces a different set of clusters for each timestep. For example, Figure 5.1 (a) shows an example of how a hundred people are scattered in an area, which is part of the real human mobility dataset we use for our experiments later on. Figure 5.1 (b) shows the same 100 people clustered in 47 spatial groups. On average there are 2 people per cluster in this occasion. However, isolated people are in their own cluster and people in more populated areas are grouped together.

Finally, Stochastic Local Greedy Search (SLGS) is called (line 6) and the human mobility patterns as well as the spatio-temporal clusters are passed to it. For each iteration of the algorithm, SLGS will produce a different mapping of participants to measurements since it will keep sampling from the mobility patterns and forming clusters for a number of times $Simulations = N$.

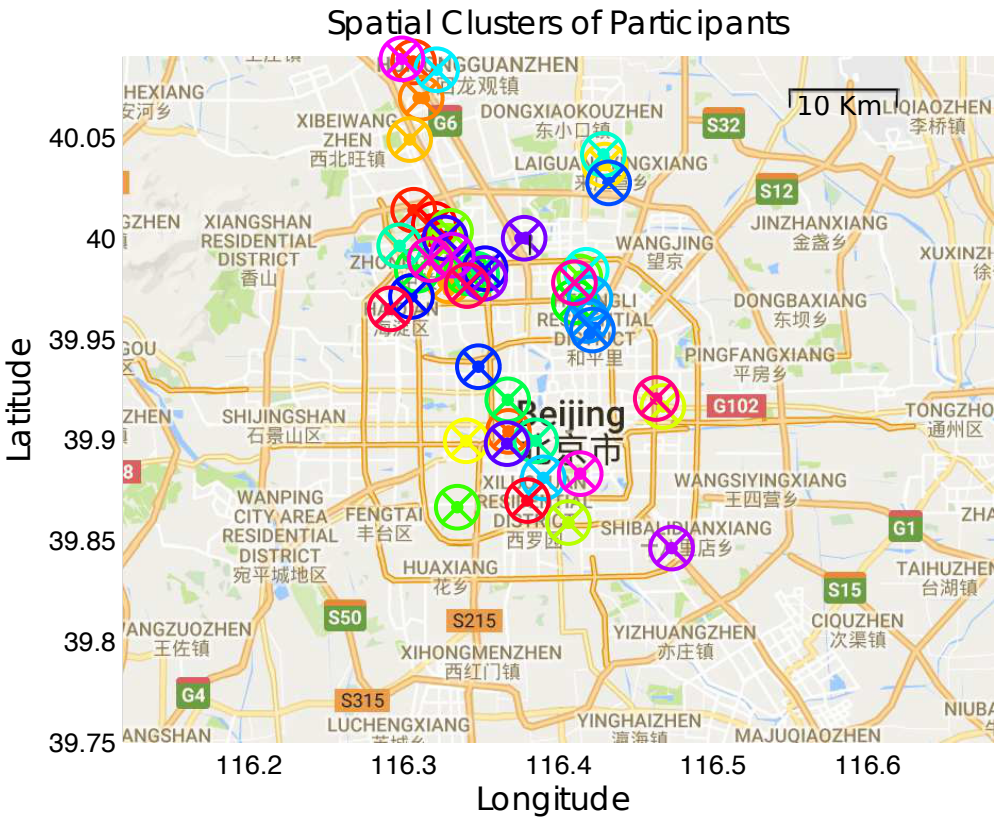
5.3.1 Stochastic Local Greedy Search (SLGS) Algorithm

The Stochastic Local Greedy Search (SLGS) algorithm is the core component of SiScaS (called on line 6 of Algorithm 4). The idea of SLGS is to stochastically evaluate a

[1] Other clustering algorithms such as K-means (MacQueen, 1967), Gaussian Mixture Model (McLachlan and Peel, 2000) or Hierarchical Clustering (Johnson, 1967) could be used here.



(a)



(b)

Figure 5.1: *Spatial locations of 100 participants in Beijing, showing the locations of individual users (a) and the locations of the means of the clusters created (b).*

number of policies, according to the utility function defined in Equation 3.5, and greedily proceed to a neighbouring policy by applying local changes in order to maximise that function. Thus, given a set of spatio-temporal clusters, the budget of people and a number of timesteps, SLGS finds a mapping between clusters and possible measurements, such that the information about the environment is maximised. SLGS is able to simulate how the information about the environment is changing over time by exploiting the property of Gaussian Processes that requires only the location of the measurement, and not the actual value of it, in order to provide the magnitude of uncertainty over the environment. In contrast to LGS, presented in Chapter 4, this algorithm is able to deal with probabilistic human mobility patterns and users' reliability. However, it lacks the backtracking feature of LGS, which makes this algorithm faster but potentially make suboptimal solutions more likely.

Algorithm 4 Simulations for Scalable Searching (SISCAS) Algorithm

input: E (timesteps), B (budget), R (Reliability)
$Simulations = N$ ▷ Number of simulations to run
for $s = 1$ to $Simulations$ **do**
    $A, l \leftarrow SAMPLEHMPs$ ▷ Sample from human mobility patterns distribution where $A \subset \mathbf{A}$ and $l \subset L$ are their corresponding locations
    $C_s \leftarrow DBSCAN(A, l, E)$
    $S_s \leftarrow SLGS(E, C_s, B, R)$
end for
return: $S_{1,\ldots,N}, C_{1,\ldots,N}$

Algorithm 5 Stochastic Local Greedy Search (SLGS)

input: E (timesteps), C (clusters), B (budget), R (Reliability)
$maxU' = 0$, $S^* \leftarrow null\ matrix(|C|)$
for $k = 1$ to $|C|$ **do**
    ▷ For each iteration k, an additional spatio-temporal cluster is taking a measurement.
    if $max_i(B_i) == 0$ **then**
        return: S^*
    end if
    $c \leftarrow RANDOMSAMPLE$ ▷ Take a random spatio-temporal sample from the set of clusters available where people have some budget left such that $c \subseteq C$
    $sz \leftarrow |c|$
    for $l = 1$ to sz **do**
        ▷ For each l, a different spatio-temporal cluster is taking a measurement.
        $O' \leftarrow O \cup o_l$ ▷ Where o_l is the extra observation to be taken
        $\mathbb{U}(\mathbf{O}_E) \leftarrow u(O'_t)$ ▷ Calculate the utility for every timestep t, where O' includes the spatio-temporal measurements selected so far, including a new measurement l.
        $s_l \leftarrow$ getMappings$(\mathbb{U}(\mathbf{O}_E), c)$ ▷ A function that associates the users in the spatio-temporal cluster with the utility of the measurement taken, i.e., $s_l : c \rightarrow \mathbb{U}(\mathbf{O}_E)$
    end for
    Keep maximum $\mathbb{U}(\mathbf{O}_E)$ of s_l in $maxU$ variable
    Set S_l to be the best configuration of all s_l
    $S^* \leftarrow$ Adapt(S_l, R, E) ▷ Get the subset of users that will be notified to get a measurement
    Reduce budget from users selected in S^*
    $\delta = (maxU - maxU')/maxU$
    if $\delta <$ threshold **then**
        return: S^*
    end if
    $maxU' \leftarrow maxU$
end for
return: S^*

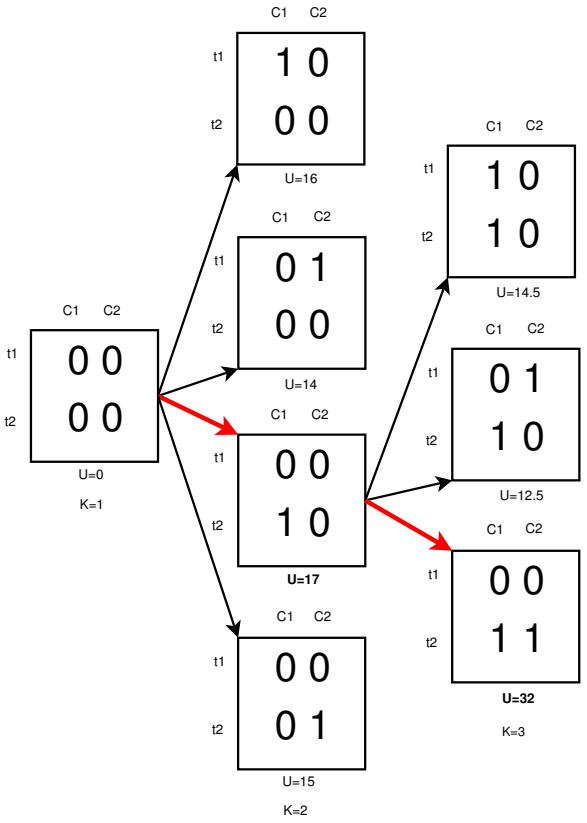


Figure 5.2: *Schematic representation of an* SLGS *Algorithm example*

A key feature of SLGS is that each cluster is treated as a single entity and can only take a single measurement at a time, which is assumed to be taken from its centroid. The reason for this is to avoid using individuals' locations to make our algorithm more efficient. However, the algorithm needs to decide who should actually take the measurement and reduce the budget of the participants accordingly.

To do so we use a greedy algorithm within each cluster (Section 5.3.2), choosing the users that provide the best expected utility, while taking into account their reliability. Intuitively, our approach requires the most reliable people to take the most important measurements. However, calculating the exact utility is intractable for a large number of users. This is because we would have to consider all the combinations of users in a cluster to get the expected utility. To overcome this problem, we calculate the probability that at least one of the selected users in the cluster will take the measurement. This is easy to calculate as it is one minus the product of probabilities of all users not taking a measurement when notified:

$$R^* = 1 - \prod_{j=1}^{W} (1 - r(A_j)) \tag{5.1}$$

where W is the number of people instructed to take a measurement within a cluster and $A_j \in \mathbf{A}$ the user to take the measurement.

Since the number of spatio-temporal clusters can be large (up to a maximum of the number of participants times the number of timesteps, i.e., $M \cdot E$), we sample again through space and time. That is, we select a random number of clusters $c \subseteq C$ for every timestep. Consequently, we are left with a smaller number of spatio-temporal clusters. We greedily select measurements that maximise the total information. However, in order to save computation time, we stop the process when the increase in information, by taking a specific measurement, is below a predefined threshold.

A simple example of how the SLGS algorithm works is presented on Figure 5.2. Here, for simplicity, we assume that there is a two timestep campaign (t_1 and t_2) with two clusters (C_1 and C_2). This is similar to the worked example in Section 3.4, but more than one agent is at the locations of $\mathcal{A}_1$ and $\mathcal{A}_2$, which is captured by the clusters C_1 and C_2 accordingly. Also, we assume that each of the clusters can only take one measurement during the campaign, i.e., have a budget of one. The algorithm starts by evaluating the null matrix, i.e., no measurements at all, and then adds the single best measurement that produces the highest total utility. However, at this stage the algorithm is unaware of who in particular will take the required measurements. A zero value means that no measurement is taken and a one means that a measurement is taken by users in the cluster at that timestep. The algorithm evaluates a number of candidate solutions at each iteration (k), selects (denoted with the bold arrow line) the one that produces the highest utility (U), calculated by Equation 3.4, and proceeds to the next iteration (k), where an additional measurement is added. For instance, when $k = 2$ the maximum utility is gained ($U = 17$) by cluster 1 (C_1) taking a measurement on timestep 2 (t_2). Similarly, when $k = 3$ the maximum utility is gained ($U = 32$) when an additional measurement is taken by cluster 2 (C_2) on timestep 2 (t_2).

Now, the SLGS algorithm, shown in Algorithm 5, is described in more detail. The algorithm accepts the locations of people spatially clustered per timestep (C), as well as the budget of each individual (B), their reliability (R) and the total number of timesteps (E) as shown in line 1. Given that there is sufficient budget left for at least one person in the cluster, it randomly selects a cluster per timestep (line 8). It then checks what the utility would be when adding a measurement for each cluster (lines 10 - 15). This is achieved by forwarding the campaign in time to check what the final utility would be (line 13). This enables the simulations to run fast since not every single position in the cluster is considered by the Gaussian Process. Next, the utility produced by the specific combination of measurements is stored as a mapping from users to spatio-temporal locations in s_l (line 14). Then, the algorithm finds the cluster that produced the highest marginal increase (δ) in utility, given the set of candidate solutions s_l, and selects it (line 16). Since the algorithm is greedy, this measurement can no longer be removed, and thus it is not considered in the following iterations. At this point Adapt is called (line 18) in order to select who, within the selected cluster, will actually take the measurement (see Section 5.3.2 for more details). At the same time, the budgets of

people in the cluster selected are adjusted accordingly, i.e., the budget of the individual particiapants selected within the cluster is reduced by one (line 19). The algorithm iterates until the marginal increase is below a percentage threshold or until everyone's budget is depleted (line 21 and line 5 respectively).

In order to speed up our algorithms, we reuse some of the results already calculated by partially evaluating policies in the SLGS algorithm. In particular, at each iteration of policy evaluation in time (line 13), i.e., when forwarding the campaign in time, we store the utility earned from that part of the policy. When this part of the policy appears again, we reuse the utility without the need to re-evaluate it.

5.3.2 Adapt Algorithm

Algorithm 6 is responsible for selecting the people within the cluster that should take the measurement, as explained above. It is called on line 18 of the SLGS algorithm (Algorithm 5).

Algorithm 6 Adapt Algorithm

1: **input**: S_l (users in space and time), $r_{1,\dots,M}$ (reliability), E (timesteps)
2: $maxU' = 0$, $S^* \leftarrow null$
3: **for** $f = 1$ to $|S_l|$ **do**
4: $sz = |S_l| - |S^*|$ ▷ People not yet selected, who have budget left
5: **for** $l = 1$ to sz **do**
6: $R_l^* = 1 - \prod_{j=1}^{|S^*|+1} (1 - r(A_j))$ ▷ Calculate the probability that at least one user takes a measurement according to Equation 5.1, where A_j are the people selected within the cluster including the new measurement o_l.
7: $u(\mathbf{O}_E) \leftarrow \sum_{t=1}^{E} R_l^* \cdot u(O_t)$ ▷ Calculate the utility for each timestep
8: $s_l \leftarrow$ getMappings($\mathbb{U}(\mathbf{O}_E), A$)▷ A function that associates the users with the utility of the measurement taken, i.e., $s_l : c \rightarrow \mathbb{U}(\mathbf{O}_E)$.
9: **end for**
10: Keep maximum $\mathbb{U}(\mathbf{O}_E)$ of s_l in $maxU$ variable
11: Set S^* to be the best configuration of all s_l
12: $\delta = (maxU - maxU')/maxU$
13: **if** $\delta <$ threshold **then**
14: **return**: S^*
15: **end if**
16: $maxU' \leftarrow maxU$ ▷ Update the highest utility
17: **end for**
18: **return**: S^*

This is a greedy algorithm that estimates the utility that at least one of the users that are requested to take the measurement in a particular cluster will actually take the measurement. In other words, it selects a subset of people within the cluster to take a measurement, saving measurements for future iterations. Figure 5.3 shows an

example of how this algorithm works. Specifically, in this example, the utility gained from the cluster taking measurement is $U = 17$ and the reliabilities of the agents are $r(\mathcal{A}_1) = 0.6, r(\mathcal{A}_2) = 0.9, r(\mathcal{A}_3) = 0.8, r(\mathcal{A}_4) = 0.7$. Initially, the measurement to be taken is by cluster 1 (C_1) on timestep 2 (t_2), which is the result of iteration $K = 2$ in Figure 5.2. The algorithm selects iteratively who in that cluster should take the measurement. For instance, in the first iteration, user 2 ($\mathcal{A}_2$) is selected (bold arrow line) since the total utility, calculated by Equation 5.1, is greater than any other choice, as it has the highest reliability. Similarly, in the next iteration, the probability of at least one user taking a measurement is calculated. The algorithm adds measurements greedily until a threshold is met. In this case, we assumed that all the measurements within the cluster have a utility of 17, but this is not typically the case in practice as people in the same group are in slightly different locations producing a different utility when taking a measurement.

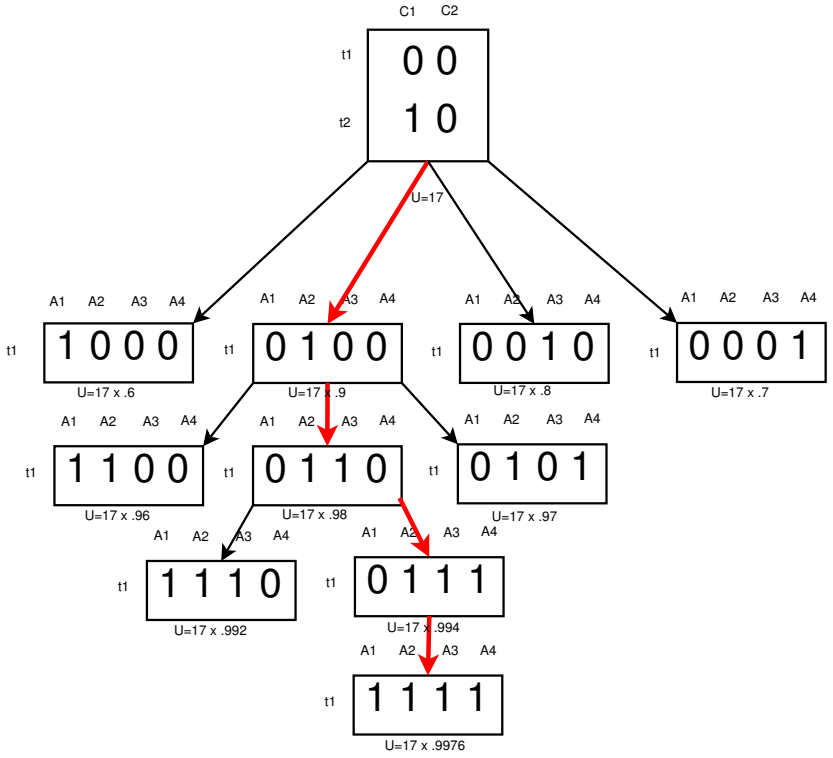


Figure 5.3: *Adapt Algorithm Example*

Now, Algorithm 6 is explained in more detail. This algorithm iterates through the number of users (line 3 - 17), who belong in the cluster selected in Algorithm 5. Then, it iteratively adds a user to the list of selected users (line 5 - 9), calculating the probability (reliability) that at least one of the selected users will actually take the measurement required (line 6). Next, the total utility is calculated (line 7) and this is stored as a mapping from users to spatio-temporal locations in s_l (line 8). Then, the algorithm finds the user that produced the highest marginal increase (δ) in utility, given the set of candidate solutions s_l, and selects it (line 10). Since the algorithm is greedy, this measurement can no longer be removed, and thus it is not considered in the following

iterations. The algorithm iterates until the marginal increase is below a percentage threshold (line 13).

5.4 The Matching Algorithm

SiScaS (presented in Section 5.3) produces a number of mappings (N) of participants to measurements depending on the samples taken from human mobility patterns, as well as the clusters that are formed.

Algorithm 7 Matching algorithm

1: **input**: E (timesteps), current (current timestep), B (budget), $S_{1,\dots,N}, C_{1,\dots,N}$
2: $A \times l \leftarrow GetHumanLocations$ ▷ Get GPS coordinates of users where $A \subset \mathbf{A}$ and $l \subset L$
3: $\widehat{C}^{current} \leftarrow DBSCAN(A, l, current)$ ▷ $\widehat{C}^{current}$ are the clusters formed at the current timestep in real time
4: **for** $s = 1$ to N **do**
5: Find nearest neighbour from $\widehat{C}^{current}$ to $C_s^{current}$ ▷ Find the best match between the cluster in real time ($\widehat{C}^{current}$) and a number of simulations ($C_s^{current}$) at a specific timestep ($current$)
6: $D_s \leftarrow$Calculate Euclidean distance of $\widehat{C}_{current}$ nearest neighbour
7: **end for**
8: $ind \leftarrow \arg\min_s D_s$ ▷ Get the index of the minimum distance.
9: $P \leftarrow C_{ind}^{current}$ ▷ Get people from the best simulation
10: $\widehat{P} \leftarrow \widehat{C}_{current}$ ▷ Get people in real time
11: Find $S^{'} = S_{ind}^{*} \cap \widehat{P}$ ▷ S_{ind}^{*} is the best match between clusters formed in simulations in advance and real-time clusters. $S^{'}$ is a subset of those mappings that includes only those people that are actually available in real time.
12: Get the people not taking measurements within selected cluster $S^{''} = \widehat{P} \backslash S^{'}$
13: Select $X = |S_{ind}^{*}| - |S^{'}|$ measurements
14: Append X random measurements to $S^{'}$ from $S^{''}$
15: $M \leftarrow$ Get people with budget left in ind simulation
16: $totalBudget \leftarrow$ Sum the budget left in ind simulation
17: $O = totalBudget/(E - current)$
18: Choose O random measurements from M
19: Append new measurements to $S^{'}$

However, in real time, participants can actually be in a different location or they may not be available at all. Also, people might not take the designated measurement even if they are actually at the desired position. The idea of the Matching algorithm is to decide who to notify in real time, given the output of SiScaS ($S_{1,\dots,N}$) and the state of the world at each timestep.

Concretely, the Matching algorithm (Algorithm 7) gets human locations (line 2) in real time and clusters them using the DBSCAN algorithm (line 3). Then, the algorithm finds the best match between the measurements that are most informative, as calculated

in advance, and the actual positions of participants in real time. Specifically, we find the nearest neighbours from the real-time clusters to the clusters produced in SiScaS (line 5) and then the Euclidean distance between them is calculated (line 7). The smaller the distance, the more similar the clusters are. Then, the simulation that best matches the current situation is found by selecting the smallest distance (D) from all the simulations (line 8). Given what measurements were selected in the simulations in advance, the corresponding people in the cluster are selected (line 9). Then, the people within the real-time cluster are selected (line 10). Since not everyone in the cluster should take a measurement, the algorithm finds whether there are actually users selected in simulations in that cluster (line 11). Given that not everyone in the cluster would have been in the simulation, we randomly select people from the cluster to match the number of users instructed to take the measurement (line 12 - 14). Next, the people whose budget has not been depleted in the best simulation are retrieved (line 15) and the total budget left is calculated (line 16). In order to evenly distribute the remaining budget, we divide the total budget by the timesteps left (line 17). Then, the algorithm randomly selects measurements to be taken by people whose budget was not depleted in the simulations (line 18). Finally, the randomly added measurements are appended to the previous ones (line 19).

5.5 Empirical Evaluation

In this section, we evaluate the algorithm developed using real human mobility patterns and air quality sensor data similar to Chapter 4. However, we highlight the differences that are specific to the extended problem (Section 3.2.2) that this chapter deals with. In the first part, we introduce our benchmarks and give a description of the experiments performed. Finally, we discuss our findings.

5.5.1 Benchmarks

The algorithm developed was benchmarked against the state-of-the-art algorithms which are introduced below:

- **Greedy:** This algorithm is based on the work by Krause et al. (2008) discussed in Section 2.6. It iterates through possible measurements available at each timestep, finding the one that produces the highest utility. It keeps adding measurements until a budget k is met. In our setting, k is derived from the total budget of people available at each timestep. In particular, we divide the total budget that is available by the number of timesteps left.
- **Best-Match:** The Best-Match algorithm works similarly to adaptive Best-Match presented in this Chapter and consequently an extension of the LGS algorithm

presented in Chapter 4. However, it is conservative in terms of the measurements taken. Specifically, when a cluster is selected in the simulations, all of the people belonging to that cluster are instructed to take a measurement. In real time, the people belonging to the cluster that matches the offline simulations are again instructed to take the measurement. In doing so, this algorithm does not take in consideration the reliability of users and may exhaust its budget more quickly than our approach. In practice, the Best-Match lacks the Adapt Algorithm (Algorithm 6).

- **Proximity-driven (Pull-Based):** This algorithm is often used in practice to let people execute tasks based on their spatial location, as described in Section 2.5. In environmental monitoring this can be interpreted as taking measurements when people are in an area of high uncertainty or when the measurement they take has a high utility. In other words, a measurement is taken if the utility gained exceeds a threshold. This approach is used by the state-of-the-art mobile crowdsourcing applications outlined in Section 2.5.

- **Random:** This algorithm is similar to the one in Section 5.7.3.1 and it randomly selects measurements to be taken by people until no budget is left. Specifically, at each timestep, a random set of participants is requested to take measurement.

- **Patrol:** The Patrol algorithm takes measurements at all timesteps similar to Section 5.7.3.1, until everyone's budget is depleted. This algorithm draws on the agent coordination literature (Section 2.4) and in particular on the work by Stranders et al. (2013), where agents continuously take measurements for environmental monitoring.

Also, since the optimal algorithm is computationally infeasible (shown in Chapter 4), we developed an upper bound to the algorithm that can be easily calculated. The upper bound is described below:

- **Upperbound:** We relax the assumption that people have a limited budget, we assume full knowledge of human mobility patterns and assume that people are reliable. Thus, all participants are assumed to take measurements at every timestep and the total utility can be trivially calculated.

5.5.2 Experimental Hypotheses

Given the benchmarks above, we formulate the following experimental hypotheses:

- *Hypothesis* 1*:* The total utility earned by the ABM algorithm will consistently be higher than that of the Greedy, Best-Match, Patrol, Random and Proximity-driven

algorithms, irrespective of the number of agents participating.

Outperforming Greedy is a result of the fact that ABM looks ahead in time and thus is able to select measurements that should increase the total utility earned by the end of the campaign. Outperforming Best-Match is due to the adaptive capabilities of ABM that is able to use agent's budget more effectively. The rest is outperformed by the fact that the Patrol and Random algorithms ignore the budget and thus taking a measurement at every time-step or randomly results in a suboptimal behaviour.

- *Hypothesis* 2*:* The total utility earned by the ABM algorithm will be higher than that of the Greedy, Best-Match, Patrol, Random and Proximity-driven algorithms in all scenarios of varying reliability.

  This is because ABM aims to increase the total utility by taking into account of the reliability of the people and probabilistically chooses the set of agents to maximise the total utility. The Greedy algorithm is indifferent to the reliability of people, and thus it performs worse than ABM and Best-Match. The ABM, even though it does not adapt, it is conservative in the sense that all the agents in the designated cluster are requested to take a measurement. As a result, the most important measurements are taken, but the budget is depleted earlier than using the ABM algorithm.

- *Hypothesis* 3*:* The total accuracy of the ABM algorithm (measured in terms of RMSE, defined in Section 2.2.2) will be higher than that of the Greedy, Best-Match, Patrol, Random and Proximity-driven algorithms, irrespective of the number of agents participating.

  This is because the accuracy is correlated with the total utility gained. Better map exploration, i.e., collection of more information, will lead to better accuracy of the heatmap produced. Consequently, the ABM algorithm will perform better than the rest of the algorithms as argued in Hypothesis 1.

- *Hypothesis* 4*:* The total utility earned by the ABM algorithm will be higher than that of the Greedy, Best-Match, Patrol, Random and Proximity-driven algorithms in all scenarios of varying dynamism.

  This is because ABM looks ahead in time as well as manages the budget so that measurements can be taken in future iterations of the algorithm compared to Best-Match. Also, as the settings become less dynamic the Greedy algorithm is able to perform better. This is because it treats individual timesteps as independent, and thus a good mapping of agents to specific locations in a particular timestep would have a high impact in the overall utility.

5.5.3 Experimental Setup

In order to empirically evaluate our algorithm, we compare its performance against the algorithms described above. In this section we extend the experimental setup described in Section 4.2.3.

In particular, we preprocess the dataset, and take the location of each user every ten minutes. We also take patterns of different weeks or months from the same pool of participants' trajectories in order to create thousands of different routes. This is used as the ground truth to compute the upper bound in our experiments. In order to make the system more realistic, we provide a probability distribution of the users' potential future locations. This is to simulate the behaviour of a real human mobility prediction system that is able to provide us with these probabilities over possible locations. In particular, in this work, we assume that the correct locations have a high probability of being assigned a higher probability than the rest of the locations. Specifically, we create the probability distribution of the locations such that 80% of the time the true location of the people will be allocated a higher probability than the alternative locations. At the same time, 20% of the time the correct location is assigned less probability than a random location from the user's mobility patterns. This is in line with evidence from the human mobility prediction literature (Song et al., 2010; McInerney et al., 2013a,b; Baratchi et al., 2014a). In particular, Song et al. (2010) claim that the predictability of human mobility patterns varies very little. Their results show that predictability peaks at 93%, and no users were observed whose predictability was under 80%. However, people have a limited budget of measurements they are willing to take per day. In our work, we assume that people have an average budget of two measurements per day, which is consistent with findings in real participatory sensing systems (Chon et al., 2013). We also experimented with different budgets and mobility patterns distributions and we got, broadly, the same results. Also, people may not take the measurement they are requested according the their reliability, as described in Section 3.2.

The next section presents the results of our experiments. Our experiments involve comparing the execution time of the algorithms and the performance in terms of accuracy and utility gained (Equation 3.4 in Section 3.2) in campaigns similar to Chapter 4. However, we additionally experiment with up to 1000 participants per timestep and different user reliabilities. We compare algorithms in terms of execution time, as the problem we address is NP-hard (Chapter 2), and thus no optimal solution is tractable.

Also, we experiment with up to a thousand of participants as the more people, the more complex the problem becomes in terms of finding the best solution. However, the more people participating in the campaign, the less the contribution of each one is, in terms of information they provide to the overall campaign. Also, as mentioned in Section 3.1, people are associated with uncertainty about whether they will actually

take a measurement when they are asked to do so. In order to examine the robustness of our algorithm, we vary the average reliability of the people between zero and one.

Moreover, in order to obtain statistical significant in our results, we performed two-sided t-test significance testing at the 95% confidence interval. Our experimental platform is the IRIDIS High Performance Computing Facility with 2.6 GHz Intel Sandybridge processors and 64GB RAM per node[2].

5.5.4 Results

Figure 5.4 shows results of the performance of the algorithms coordinating a thousand participants when varying the time-scale, which controls the dynamism of the phenomenon. Intuitively, the smaller the time-scale is, the more dynamic the phenomenon. Consequently, as the time-scale approaches zero, the phenomenon rapidly changes over time. In these environments, the adaptive Best-Match algorithm is better, in terms of total utility gained than the rest of the algorithms. The adaptive Best-Match algorithm saves measurements in the simulations by choosing who specifically should take measurements within the cluster, while at the same time maximising the total utility. This allows the algorithm to take extra measurements in real time, which increases the total utility and thereby leads to a higher performance than the Best-Match algorithm (hypothesis 4). Next is the greedy algorithm. This algorithm is able to choose individuals to take measurements that increase the total utility and that could potentially be in different

[2] `http://cmg.soton.ac.uk/iridis`

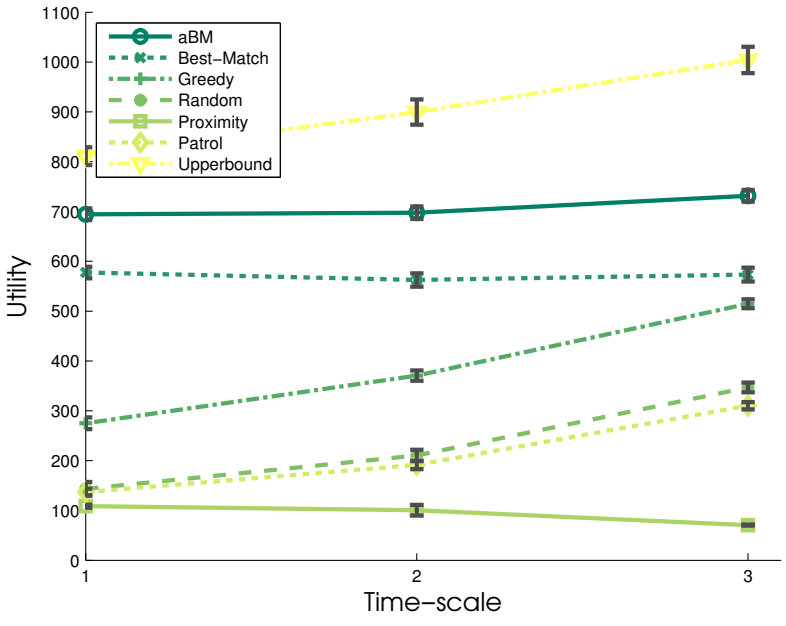


Figure 5.4: *Total utility gained for 24 timesteps when run 1000 participants. The error bars indicate the 95% confidence intervals.*

clusters. However, as it will be discussed in detail later in this Section, this comes at a great computational expense, as the algorithm needs to consider all the participants one by one until the k best observations are found at each timestep. Also, since it cannot look ahead in time, the algorithm struggles in highly dynamic environments. Specifically, it is possible that some future measurement is more informative if no measurement was taken at that location in the past. The adaptive Best-Match algorithm is designed to produce reasonable outcomes in dynamic environments and it is shown to outperform all the benchmarks in these environments. The proximity algorithm chooses measurements that are informative, since they are above a threshold, but it does not perform well. As we mentioned before, a future measurement might be more informative than the current one. Thus, taking a measurement, which is above the threshold at a timestep, might not be as informative as taking some other measurement in the future. If the threshold is very high, taking only that future measurement might not be as informative as taking a lot of measurements over time. Moreover, it is difficult to define which measurements are informative as the threshold needs to be determined empirically. Patrol is an algorithm that instructs all the users to take all the measurements whenever possible. This means measurements are taken as early as possible until budgets are depleted. This is not a good strategy as no budget is left later in the campaign. Even a random algorithm is better than patrol since only a random subset of people are taking measurements at each timestep. However, there is no intelligent component that determines how those measurements are taken, and thus uninformative measurements are taken.

In particular, adaptive Best-Match is 23.93% better than the Best-Match algorithm for 1000 agents and 94.27% better than Greedy. It is consistent for different participants and the results are significant to a 95% confidence level in a two-tailed t-test significance test.

Figure 5.5 shows the results of the performance of the algorithms in terms of utility gained when we vary the number of participants (M) in the campaign. The dynamism in this experiment is fixed at 1, to show the performance of the algorithms in a highly dynamic phenomenon. We can observe that adaptive Best-Match is 12.74% better than the Best-Match algorithm and 3.3 times better than the Greedy algorithm for 250 participants. It is 20.31% better than Best-Match and 2.8 times than Greedy with 500 participants. It is 21.43% better than Best-Match and 2.6 times than Greedy for 750 participants. Finally, it is 23.91% better than Best-Match and 2.5 times than Greedy for 1000 participants. The results are significant to a 95% confidence level in a two-tailed t-tests significance test. Overall, we can observe that adaptive Best-Match algorithm is significantly better in most scenarios and at least as good as the Best-Match up to 150 users (hypothesis 1). Crucially, the upperbound is on average only 13.14% better than adaptive Best-Match, which highlights the good performance of our algorithm. Also, Patrol, Random and Proximity algorithms have similar performance.

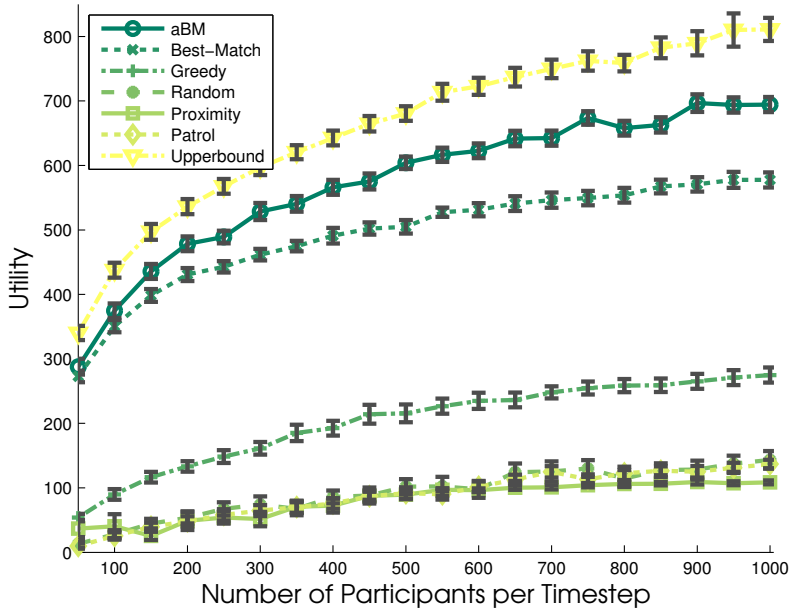


Figure 5.5: *Total utility for 24 timesteps and a varying number of participants at a constant time-scale of 1. The error bars indicate the 95% confidence interval.*

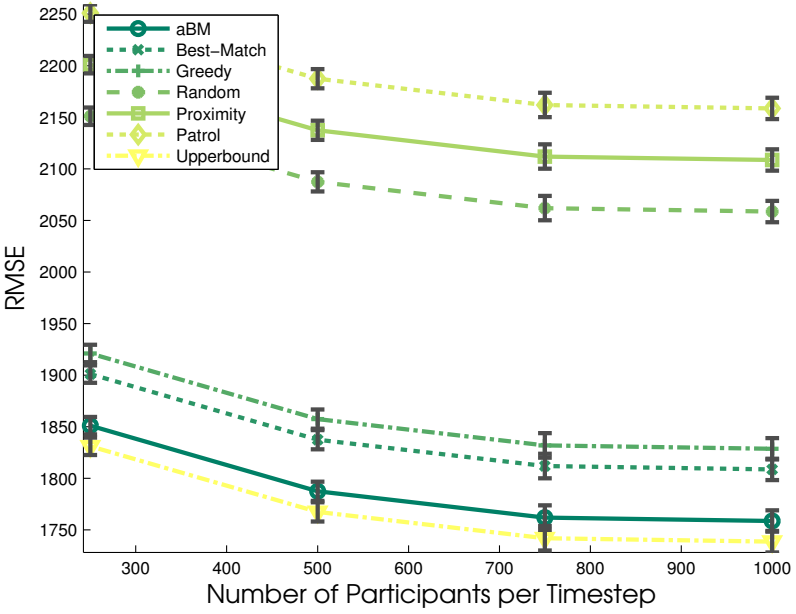


Figure 5.6: *Total RMSE gained for a 1-day participatory sensing campaign. The error bars indicate the 95% confidence interval.*

Figure 5.6 shows the results of the performance of the algorithms in terms of the RMSE when varying the number of participants in the campaign. Our results confirm the intuition that the more measurements the more accurate the heatmap produced would be. However, there is significant difference between each algorithm. Specifically, ABM is the most accurate and close to the upperbound. Our results are in line with the findings we presented about the utility when varying the number of participants as the

utility gained over the environment is related to how much the environment has been explored over time, which ultimately leads to a more accurate map of the environment (hypothesis 3).

Figure 5.7 shows the performance of the algorithms in terms of the total runtime when varying the number of participants per timestep. The results show that the adaptive Best-Match algorithm is faster than the Greedy and Proximity-driven algorithms and it is comparable to Best-Match. Specifically, it is not significantly different up to 250 agents, but it is 40.5% slower for 1000 agents and 30.02% on average. It is evident that the runtime of adaptive Best-Match and Best-Match algorithms grows linearly with the number of agents. The Proximity-driven and Greedy algorithms require much more time, because as the number of users increases, the number of possible measurements that could be taken is greatly increased. In fact, the Greedy algorithm is about 50 times slower than the aBM algorithm for 1000 agents. Depending on the number of measurements to be taken at each timestep (k), the Greedy algorithm attempts to find the best measurements by iteratively adding the next single best measurement to the list of measurements to be taken. Similarly, the proximity-driven algorithms evaluates all the possible measurements to decide whether or not that measurement would lead to a higher gain than the pre-defined threshold.

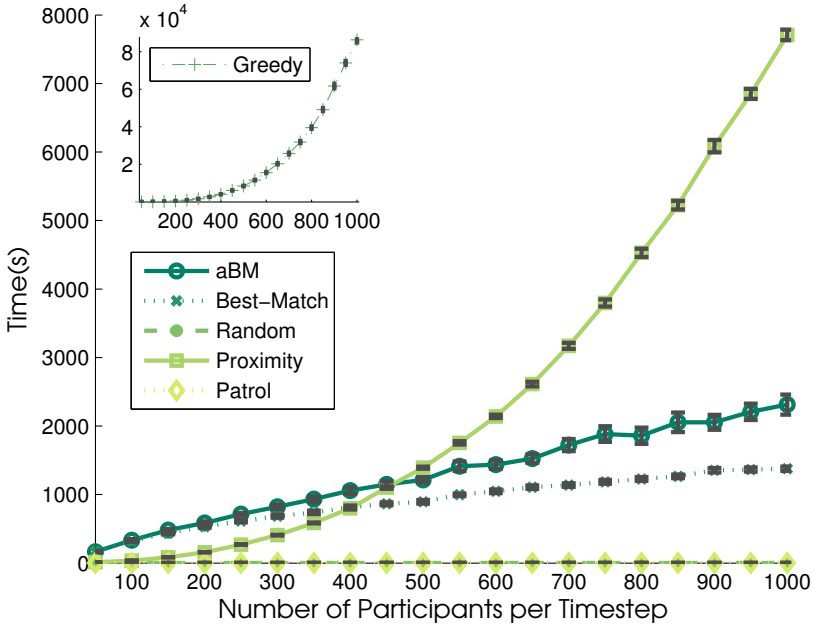


Figure 5.7: *Average runtime for 24 timesteps and a varying number of participants. The error bars indicate the 95% confidence interval (wrong image here).*

Figure 5.8 shows the performance of the algorithms in terms of utility when we vary the average reliability of the users. The dynamism is fixed at 1, i.e., a highly dynamic phenomenon and the agent number to 500, which is a representative number such that all algorithms work efficiently, given the runtime of the algorithms in Figure 5.7. We

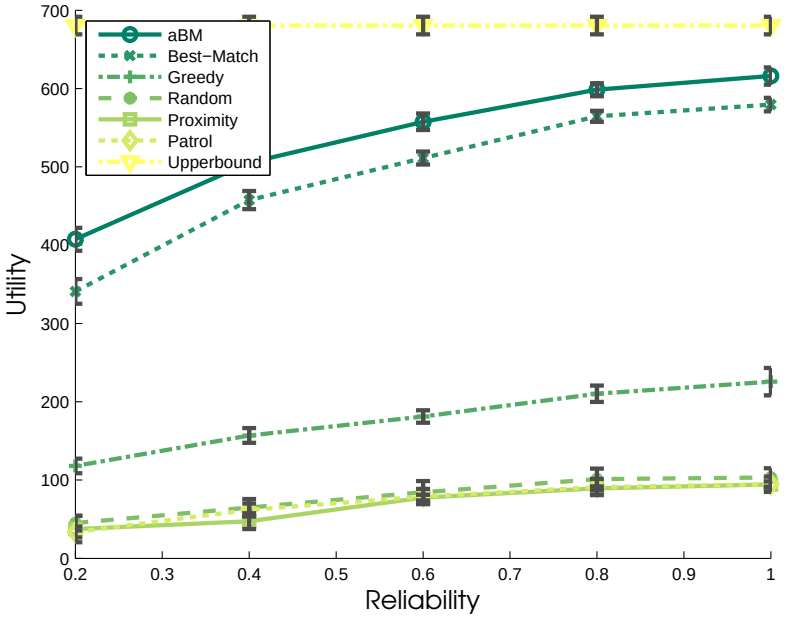


Figure 5.8: *Total utility for 24 timesteps for 500 agents with a varying reliability at a constant time-scale of 1. The error bars indicate the 95% confidence interval.*

observe that adaptive Best-Match is 19.55% better than the Best-Match when reliability is 0.2 and 6.3% when reliability is 1 and 10.27% on average. This is because the offline component of the adaptive Best-Match is able to select more people when reliability is low by choosing the most important measurements in the simulations to be taken by people with the highest reliability (hypothesis 2). Also, in real time, the online component of the algorithm selects a number of available participants who still have some budget left to take measurements randomly, evenly distributed across the time domain. On the other hand, Best-Match selects all the participants in the best cluster. This makes sure that the most important measurements are taken, but at the cost of using the budget of some people that could potentially have taken other measurements at a different location and time.

5.6 Summary

In this Chapter we presented an algorithm that maximises the total utility gained over a period of time while at the same time considering people's budget (requirements 1 and 4). We focused on solving the extended problem presented in Section 3.2.2 that introduces uncertainty in human mobility patterns and behaviour, in terms of task execution uncertainty (challenge 4). At the same time, our algorithm scales up to a thousand participants per timestep (requirement 6). In particular, we demonstrated how efficient the adaptive Best-Match algorithm is compared to the state-of-the-art Best-Match and Greedy algorithms. An empirical evaluation on real data showed that, (a) adaptive Best-Match is significantly better than the Best-Match and Greedy algorithms

in terms of total utility gained, (b) adaptive Best-Match is significantly faster than the Greedy approach and comparable to Best-Match, (c) dynamic environments affect the performance of the adaptive Best-Match algorithm and the total utility gained, but it still outperforms the benchmarks in all scenarios, (d) adaptive Best-Match is significantly better than the Best-Match and Greedy algorithms in all scenarios with different degrees of user reliability, and (e) adaptive Best-Match outperforms the rest of the algorithms in terms of accuracy measured in terms of RMSE.

5.7 Coordinating Measurements under Uncertainty in the Presence of Malicious Users

In this section, we consider the effect of the presence of potentially malicious users as described in Sections 2.5.3 and 3.2.3. Thus, the main challenge addressed in this chapter is coordination of measurements, in order to maximise the information about the environment (challenge 1), in the presence of malicious users in the participatory sensing campaign (challenge 5), given the budget constraints (challenge 3). Specifically, in addition to the task execution uncertainty (reliability) of users (Section 5.1), our algorithm must be able to deal with malicious users (requirement 5). Consequently, we use a different model that is able to capture individual user maliciousness and the ABM algorithm is extended to swap potentially malicious users with non-malicious users in real time. We firstly give a high level overview of the proposed algorithm and then describe it in more detail.

5.7.1 Trust-based adaptive Best-Match

Our algorithm extends the trust-based Heterskedastic Gaussian Process (HGP) model described in Section 2.2.2 by alleviating the requirement for manual user input of the estimated precision. Specifically, our algorithm estimates the users' trustworthiness in real time by applying the MLE technique at each timestep[3]. In particular, the t value, which is a scaling hyperparameter of the noise in Heterskedastic Gaussian Process model, is estimated for all participants that took a measurement at a specific timestep as described in Section 2.5.3. This value can only be learned after a user has already taken a measurement and it is updated each time a user takes a measurement. At the same time, trust values affect the mean prediction for specific areas. In particular, the contribution of less trusted users has a lower impact on the predicted function over space and time. By applying the MLE technique at each timestep, we incrementally learn the trustworthiness of all users actively participating. Specifically, active participants are associated with a trustworthy value when they take measurement and thus over time

[3]We use GPML v4 toolbox and in particular a nonlinear conjugate gradient method.

more users are associated with a trustworthy value. Active participants are those who are selected to take a measurement by the coordination system in Section 5.1. This becomes more clear in the Algorithm 8.

Therefore, at each timestep, when selecting users to take measurements, some of these may already be associated with trust values (if they have previously taken measurements). This enables us to compare trust levels of individuals who we have information about. Then, if the trustworthiness of a user that is about to take a measurement is significantly lower than the rest, we swap that user with the closest one that still has budget left and whose trustworthiness is not significantly different than the rest of the users. This ensures that malicious users will be swapped out.

Overall, our Trust-based adaptive Best Match (TABM) algorithm (Algorithm 8) has two major additions compared to ABM (Section 5.1) in order to effectively deal with malicious users. The first one is the application of the MLE technique per timestep in order to learn the t values of participants' taken measurements, which in turn is used as a hyperparameter in the trust-based HGP model. This enables the system to get a value for each participant's measurement that characterises its accuracy. Also, the contribution of less trusted users has a lower impact on estimating the state of the environment. The other component is called SWAP and is responsible for swapping malicious or low-trust users with more trustworthy nearby users in real time. Given the t values learned above, the system is able to filter out potentially malicious users by replacing them with higher-value users. As a result, people with lower trust values are not chosen to take more measurements.

In more detail, the TABM algorithm requires the number of timesteps of the participatory sensing campaign, the budget of each participant, their reliability and the hyperparameters of the model (line 1). Next, simulations run offline, as in Section 5.1 and a spatio-temporal mapping between participants and locations is produced (line 2). Then, the trust-related hyperparameters are initialised (line 3) followed by the online component of the algorithm (lines 4 - 17). In particular, for each timestep, the MATCHING algorithm utilises information provided by the offline simulations to select participants to take measurements (line 5). This algorithm is explained in more detail in Algorithm 7 in the previous Chapter (Section 5.4). At this point, given the set of users to take a measurement at a specified timestep, the algorithm calculates the average trust of the users, if it exists (line 6). Next, the standard error of the mean is calculated (line 7). Given these values, a trust threshold is calculated (line 8), that is the lowest value of trust a user's measurement can have in order not to be swapped. In other words, all the participants taking a measurement should not have a trust value less than the threshold as this implies they are significantly more likely to be malicious. In order to evaluate participants, the algorithm iterates through the participants which were selected to take a measurement at each timestep (lines 9 - 15). If someone's trust value is below the *threshold* (line 10), then the SWAP function is called (line 11), which is further discussed

in the following section. Otherwise, the participant takes the measurement as originally intended (line 13). Finally, given the measurements taken, the new trust values for the participants are estimated (line 16).

5.7.2 SWAP Algorithm

The SWAP algorithm is responsible for removing malicious users from the set of selected users that are required to take measurements at any given timestep and for substituting them with nearby high-trust ones. Intuitively, the algorithm searches through the participants for the closest one, in terms of Euclidean distance, who at the same time has taken high-trust measurements and has some budget left. This user will take the place of the potentially malicious one.

In more detail, this algorithm requires the details of the particular user currently examined, the details of all other agents and the threshold calculated in Algorithm 8 (line 1). Next, an empty set named *evaluated* is created to keep track of the users examined (line 2). While the size of that set is less than the total number of agents the algorithm searches for a suitable user to substitute the malicious one (line 3). The set of candidate users is created by removing any already evaluated users from the set of all participants (line 4). In order to find a suitable substitution, the algorithm looks for the nearest neighbours to the malicious one (line 5). Once the nearest neighbour is found, it is checked whether it satisfies certain properties (line 6). Specifically, the user should have some budget left and a trust value. In our work, we assume we do not have a default trust value but rather we assume it is unknown. Given that these are satisfied, the algorithm checks whether the new user's trust is above the threshold (line 10). Then, the substitution is made (line 11) by removing the malicious user from the set of selected users and adding the new one. If no substitute is found, the user is not swapped but their measurement has a low impact on the overall prediction of the phenomenon. This is due to the fact that the trust value is a scaling parameter as discussed in Section 2.5.3. Thus, a low-trust value means that the variance of the kernel of the model is not greatly affected. Consequently, the measurement provided by that user entails less information about the phenomenon.

Algorithm 8 Trust-aware adaptive Best-Match (TABM) Algorithm

input: E (timesteps), B (budget), R (reliability) $\hat{\mathbf{\Theta}}$ (hyperparameters) $\mathbf{A}$ (agents)
$S_{1,\dots,N}, C_{1,\dots,N} \leftarrow$SISCAS$(E, B, R)$ ▷ Simulations running offline 4
$t = zeros$
for $j = 1$ to E **do**
    $S_j^* \leftarrow$ MATCHING$(E, j, B, S_{1,\dots,N}, C_{1,\dots,N})$ ▷ Online mapping of users to measurements 7
    $average\_trust = \frac{1}{|S^*|} \sum_{s=1}^{|S^*|} t_s$
    $sem = \frac{std(t)}{|S^*|} \cdot 1.96$ ▷ standard error mean for 95% confidence level
    $threshold = average\_trust - sem$
    for $i = 1$ to $|S_j^*|$ **do**
        if $t_i < threshold$ **then**
            SWAP(S_i^*,$\mathbf{A}$,$threshold$)
        else
            Take measurement
        end if
    end for
    $\Theta_{ML} = \arg\max_{\Theta} p(S_j^*, \mathbf{y}|\hat{\mathbf{\Theta}})$ ▷ $\mathbf{y}$ is the actual measurements taken by people in S_j^*
end for

Algorithm 9 SWAP Algorithm

input: $\mathcal{A}$ (agent), $\mathbf{A}$ (agents), $threshold$ (trust value)
$evaluated = \emptyset$
while $|evaluated| < |\mathbf{A}|$ **do**
    $\mathbf{A^*} \leftarrow remove(\mathbf{A}, evaluated)$
    $\mathcal{A}^N \leftarrow nearestneighbour(\mathcal{A}, \mathbf{A^*})$
    if $\mathcal{A}^N = \emptyset$ or $t_{\mathcal{A}^N} = \emptyset$ or $B_{\mathcal{A}^N} = 0$ **then**
        Return
    end if
    Append $\mathcal{A}^N$ to $evaluated$
    if $t_{\mathcal{A}^N} > threshold$ **then**
        Substitute $\mathcal{A}$ with $\mathcal{A}^N$
    end if
end while

5.7.3 Empirical Evaluation

In this section, we evaluate the algorithm developed using real human mobility patterns and air quality sensor data. In the first part, we introduce our benchmarks and give a description of the experiments performed. Finally, we discuss our findings.

5.7.3.1 Benchmarks

The algorithm developed was benchmarked against the state-of-the-art algorithms, which are also described in the previous chapters. In particular, we compare our algorithm against the Greedy, aBM and Best-Match algorithms and omit the rest since we have already shown their poor performance in the previous chapters. The benchmarks are presented below:

- **Greedy:** This algorithm is the same as the one described in Section 5.1 (Section 5.5.1).
- **adaptive Best-Match (aBM):** This algorithm is presented in Section 5.1 and in particular it is described in detail in Section 5.2. This is an adaptive coordination algorithm shown to perform well in uncertain environments.
- **Best-Match:** This is an algorithm presented in Section 5.1 and in particular in Section 5.5.1.

Also, since the optimal algorithm is computationally infeasible we developed an upper bound to the algorithm that can be easily calculated. The upper bound is described below:

- **Upperbound with Optimal HGP:** We relax the assumption that people have a limited budget, we assume full knowledge of human mobility patterns and assume that people are reliable. Thus, all participants are assumed to take measurements at every timestep and the total utility can be trivially calculated. We use a HGP model with trust values of 0 for malicious and 1 for trustworthy users.

5.7.3.2 Experimental Hypotheses

Given the benchmarks above, we formulate the following experimental hypotheses:

- *Hypothesis* 1*:* The total RMSE of the TABM algorithm will consistently be lower than that of the adaptive Best-Match, Best-Match and Greedy algorithms, irrespective of the number of agents participating.

Outperforming Greedy is a result of the fact that TABM looks ahead in time, and thus is able to select measurements that should increase the total utility earned by the end of the campaign, and thus result in better RMSE. Outperforming ABM and Best-Match is due to the combination of trust and adaptive capabilities of TABM that is able to swap potentially malicious users who have an impact on the RMSE.

- *Hypothesis 2:* The total RMSE of the TABM algorithm will be lower than that of the adaptive Best-Match, Best-Match and Greedy algorithms in all scenarios of varying maliciousness.

  This is because TABM aims to increase the total utility by taking account of the presence of potentially malicious users and minimises their impact to the campaign as well as swaping them with more trustworthy participants. The rest of the coordination algorithms do not consider the presence of malicious users, and thus do not perform as well.

5.7.3.3 Experimental Setup

To empirically evaluate our algorithm, we compare its performance against the algorithms described above.

As in Chapter 4 and Section 5.1, we focus on air quality in terms of fine particulate matter (PM2.5) in Beijing, and we use the same air quality dataset (Zheng et al., 2013) and mobility patterns (Zheng et al., 2009, 2008, 2010).

Also, as in Section 5.1, we assume that people have an average budget of two measurements per day and an average reliability of 83%, which is consistent with findings in real participatory sensing systems (Chon et al., 2013; Sahami Shirazi et al., 2014). Furthermore, we vary maliciousness between $0.1 - 1$ for the experiments whose results are shown in Figure 5.9 and it is fixed to 0.25 for experiments whose results are presented in Figure 5.10 and Figure 5.11, as this is shown to be a typical prevalence of malicious users in the crowdsourcing domain (Gadiraju et al., 2015). In our settings, we model malicious users as agents whose measurements are significantly different from the ground truth.

The next section presents the results of our experiments. Our experiments involve comparing the execution time of the algorithms and the performance in terms of RMSE with different numbers of participants (up to 1000 per timestep) and different degrees of maliciousness. We compare algorithms in terms of execution time, as the problem we address is NP-hard (Krause et al., 2008), and thus no optimal solution is tractable but at the same time a solution should be given in a reasonable amount of time.

At the same time, the RMSE measures the accuracy of the air quality heatmap created by taking measurements over time. Also, the more people participate, the more complex the problem becomes in terms of finding the best solution. Furthermore, people are associated with uncertainty about whether they will actually take a measurement when they are asked to do so.

Since measurements can be malicious, and thus deviate from their real values, it is crucial to compare the algorithms in terms of the accuracy of the resulted heatmap. While we are still interested in maximising utility, it is not directly worth comparing the algorithms since even though an algorithm's utility might be high, the overall accuracy might be small due to the malicious contributions. This is because the utility is proportional to the variance of the Gaussian Process model, which is affected by the location and the time of the measurements but is independent of the actual values of the phenomenon.

Finally, in order to obtain statistical significance in our results, we performed two-sided t-test significance testing with $\alpha = 0.05$ significance level.

5.7.3.4 Results

Figure 5.9 shows that the TaBM algorithm outperforms the benchmarks with respect to the RMSE. Crucially, at the same time, it is not significantly different from the optimal approach. Also, we observe that the more malicious users exist in the system, the more the RMSE increases for all the algorithms as expected. In more detail, TaBM is up to 18.11% (14.8% on average) more accurate than the second algorithm of the Figure (aBM). TaBM is better because it is able to capture measurements that are potentially malicious, i.e., higher or lower than the true value of the phenomenon, in the HGP model. The users taking such measurements are swapped with higher-trust neighbours, and thus malicious measurements are reduced. However, there is a linear increase in RMSE for all algorithms since the more malicious users there are, the more measurements deviate from the true values of the phenomenon. When malicious users are more numerous than the trustworthy ones, TaBM is still better as it is able to reduce the impact of extreme measurements using parameter t that captures individual trust level. Thus, it continues to be better than the rest even though the overall RMSE significantly increases.

Figure 5.10 shows that the TaBM algorithm outperforms the benchmarks by up to 60.4% with respect to the RMSE for 250 users. Also, it is consistently better for all number of users in the participatory sensing campaign. What is mostly evident from our results, is that a trust-based heteroskedastic GP approach with SWAP capabilities significantly improves the accuracy of the coordination algorithm. However, there is no improvement in performance when more agents participate in the campaign compared to other algorithms.

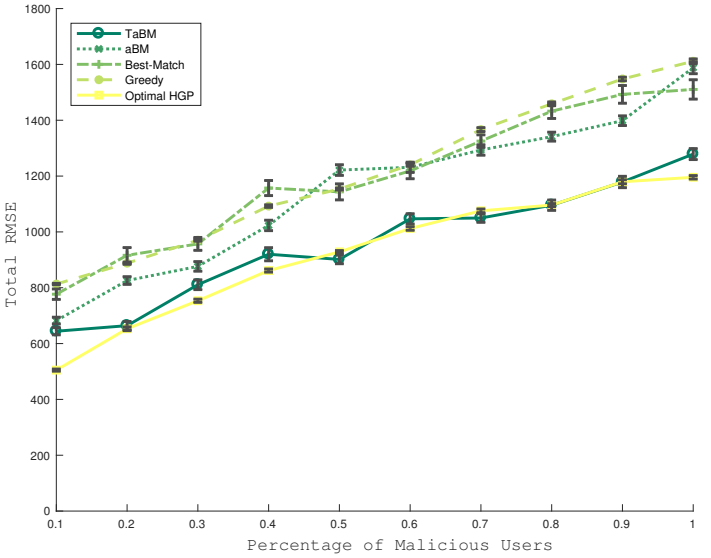


Figure 5.9: *Total RMSE over space and time with a varying percentage of malicious users. The error bars indicate the 95% confidence interval.*

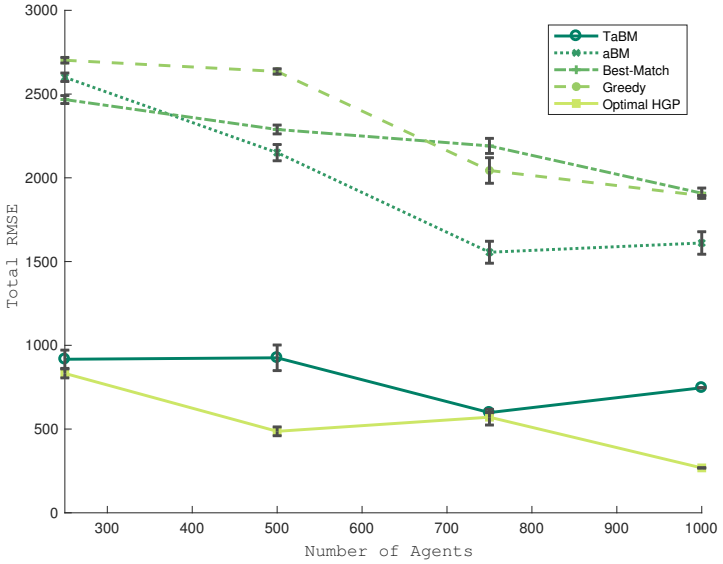


Figure 5.10: *Total RMSE over space and time with a varying number of users. The error bars indicate the 95% confidence interval.*

Figure 5.11 shows that ABM, Best-Match and the Optimal HGP have similar runtime. However, the TABM algorithm is more computationally expensive than these algorithms but with the significant trade-off in performance as discussed above. In particular the algorithm's bottleneck is the kernel inversion, which is required in the MLE technique for the estimation of the trust value. That is to say, TABM should be used only in cases where malicious users are present. Also, the Greedy algorithm has a significantly higher computational cost compared to the rest of the algorithms, as the algorithm needs to consider all of the participants one by one until the k best observations are found at each timestep.

Overall, the TABM algorithm makes more accurate predictions in terms of RMSE in

all scenarios. Specifically, it overcomes the issue of malicious measurements over time by correctly learning to place a low degree of trustworthiness on potentially malicious users and then swap low-trust users with high-trust nearby users. This effectively allows important spatio-temporal measurements to be taken as accurately as possible. Finally, the results show that our method is more accurate and considerably more informative in estimating air quality levels on a prominent air quality dataset.

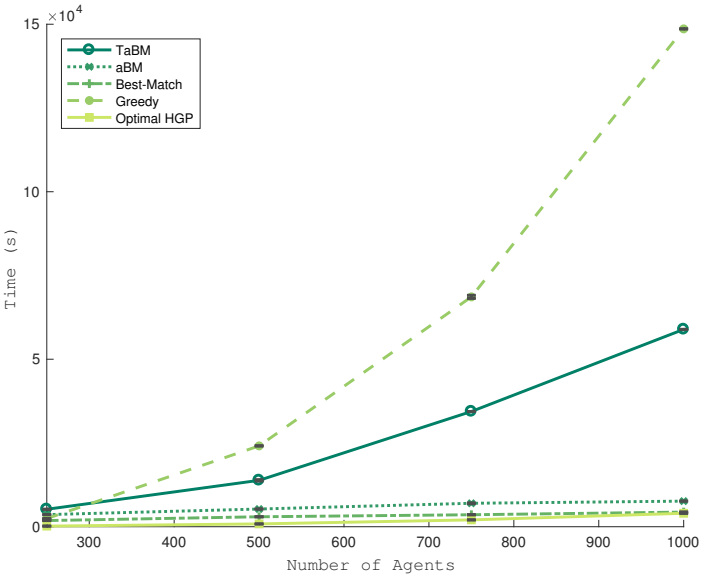


Figure 5.11: *Average runtime for 24 timesteps and a varying number of users. The error bars indicate the 95% confidence interval.*

5.7.4 Summary

In this chapter, we developed a novel coordination algorithm that maximises the total utility gained over a period of time (requirement 1), constrained on the number of measurements each user is willing to take (requirement 4), under the presence of malicious users and evaluated it in terms of RMSE and execution time (requirement 5). As in Section 5.1, we still considered human reliability (requirement 4) and we experimented with up to 1000 participants per timestep (requirement 6).

We demonstrated how efficient the Trust-based adaptive Best-Match (TaBM) algorithm is compared to the state-of-the-art algorithms. An empirical evaluation on real data showed that, (a) Trust-based adaptive Best-Match is significantly better than the adaptive Best-Match, Best-Match and Greedy algorithms in terms of total RMSE, and (b) Trust-based adaptive Best-Match is significantly faster than the Greedy approach.

Chapter 6

Conclusions and Future Work

This chapter summarises our work and suggests new ideas and avenues for the future.

6.1 Conclusions

In this thesis we have studied the challenges of monitoring environmental phenomena, and in particular air pollution, in participatory sensing settings. Specifically, air pollution causes millions of deaths per year and billions of pounds are lost in labour income. The state of the art in monitoring air quality is placing air quality static sensors in cities to capture the average pollution of the area. As argued, this approach is expensive to utilise as the equipment is typically expensive and requires experts to operate and maintain it. An alternative low-cost approach is participatory sensing.

In this thesis, we firstly examined participatory sensing applications (Section 2.1) focusing on environmental monitoring to identify gaps and potential areas for contribution. It is obvious that there is a major gap in coordinating participatory sensing campaigns. In particular, none of the existing applications guide users when and where to take measurements such that more information is learned about the environment (challenge 1) subject to a user budget or a cost (challenge 3), either in terms of battery consumption or inconvenience incurred to users by taking measurements. Moreover, we observed that, currently, no system exploits the spatio-temporal characteristics of the phenomena and the potentially known human mobility patterns of the participants to improve participatory sensing campaigns.

To address these challenges we proposed a novel participatory sensing framework (Section 3.1) that can utilise historic information about the environment to build an environmental model and human mobility patterns. This framework is the first to coordinate participants towards more informative measurements, given that they have a limited

budget or a cost for doing so. In that direction, we developed intelligent algorithms to coordinate measurements in these settings.

Our initial algorithm (Chapter 4), called LGS, is able to outperform the state-of-the-art Greedy algorithm and Optimal approach. An empirical evaluation on real data showed that, (a) LGS is 33.4% better than the Greedy algorithm, (b) the dynamism of the environment affects the performance of the LGS algorithm and the total utility gained, but it still outperforms the benchmarks in all scenarios and, (c) LGS outperforms the rest of the algorithms in terms of RMSE. This work was published at the 14th International Conference on Autonomous Agents and Multi-agent Systems (AAMAS 2015). However, our approach made assumptions about human mobility patterns that might not always be true in real life. More specifically, even though people are typically predictable in their daily routine, there is a chance of not following it or even have multiple possible routines in particular days. In particular, real human mobility prediction systems provide probabilistic information about where each person will be in the future, which is not exploited in that work. Most importantly, however, this approach was shown to scale up-to 250 participants in our experiments, which is a limitation for participatory sensing environmental campaigns.

In order to address the shortcomings of our previous work we developed advanced stochastic algorithms to deal with the human uncertainty and scalability issues (Chapter 5). In particular, we demonstrated how efficient the adaptive Best-Match algorithm is compared to the state-of-the-art Best-Match and Greedy algorithms. An empirical evaluation on real data showed that, (a) adaptive Best-Match is significantly better than the Best-Match and Greedy algorithms in terms of total utility gained, (b) adaptive Best-Match is significantly faster than the Greedy approach and comparable to the Best-Match one, (c) dynamic environments affect the performance of the adaptive Best-Match algorithm and the total utility gained, but it still outperforms the benchmarks in all scenarios, (d) adaptive Best-Match is significantly better than Best-Match and Greedy algorithm in all scenarios with different degrees of user reliability, and (e) adaptive Best-Match is better than the rest of the algorithms in terms of RMSE. This work was accepted in the Journal of Artificial Intelligence Research (JAIR) and the thirtieth AAAI conference on Artificial Intelligence (AAAI-16). However, this algorithm does not deal with the actual measurement accuracy of participants' readings. Participatory sensing is vulnerable to malicious users taking advantage of the system, falsifying the overall picture of the environment to satisfy their own agendas.

In order to deal with the malicious users' challenges we developed a coordination algorithm that is able to coordinate measurements in participatory sensing settings in the presence of malicious users. In particular, we developed a novel algorithm that maximises the total utility gained over a period of time constrained on the number of measurements each user is willing to take and evaluated in terms of RMSE and execution time (Section 5.7). We demonstrated how efficient the Trust-based adaptive Best-Match

(TABM) algorithm is compared to the state-of-the-art algorithms. An empirical evaluation on real data showed that, (a) Trust-based adaptive Best-Match is significantly better than the adaptive Best-Match, Best-Match and Greedy algorithms in terms of total RMSE, and (b) Trust-based adaptive Best-Match is significantly faster than the Greedy approach and comparable to the adaptive Best-Match and Best-Match. This work was published at the fifth AAAI conference on Human Computation and Crowdsourcing (HCOMP-17).

This work constitutes a significant advancement in the area of artificial intelligence as our algorithms can be used in other applications beyond environmental monitoring. In particular, this work focuses on an entropy-based criterion as a utility function, which is the difference in the information between two timesteps. However, a new utility function for other applications can be devised. For example, in a crowdsourcing classification system, users could be asked to verify objects or events (e.g., traffic jams, vandalism or littering), which are classified from a machine vision algorithm, by physically visiting those locations. The utility in this scenario could capture how valuable human input is. For instance, verifying a rare event of vandalism at a specific location could be more important than verifying a traffic jam in a usually busy area. Our algorithm could be used to decide which users to ask to increase the overall system's efficiency.

In summary, we have demonstrated a framework and algorithms that satisfy many of the requirements set out in the introduction. Specifically, we addressed requirements 1, 2, 3, 4, 5, 6. However, requirement 6 was not fully satisfied as we only showed coordination up to 1000 users per timestep, while expectations were to scale up to hundred of thousands. Also, we did not provide theoretical guarantees of our results but rather our algorithms were evaluated empirically.

6.2 Future Work

There are a number of potential avenues for the future.

- Improve scalability: The current system is able to deal with hundreds or up to a thousand of users per timestep. However, we believe there is potential to improve even more, scaling up to many thousands and even hundred of thousands. At the moment, the Gaussian process is a major bottleneck in the overall system. In particular, calculating the covariance function has a $O(n^3)$ computational complexity. We believe that there is room for more efficient use of Gaussian processes in the context of environmental monitoring, and more research is required to develop Gaussian process models that are more scalable.

- Improve environmental representation: In this work we make use of common practices in the field of Gaussian processes like using Matern and/or Squared Exponential covariance function, and learning the hyperparameters by maximising the log likelihood. However, there are composite covariance functions that are complicated, in terms of the number of hyperparameters needed to optimise as well as computational expensive to compute, but may be most suited in spatio-temporal environmental applications. Also, different covariance functions could be used for different phenomena or different applications. Gaussian processes is an actively researched topic and future work could devise a kernel that works better for predicting air quality over space and time in a specific area. This could dramatically improve the accuracy and applicability of the system in real-world settings.

- Provide theoretical and more empirical analysis: The current algorithms are evaluated empirically on a composite of real datasets. Since, there was in fact a single composite dataset it would be interesting to explore the performance of our algorithms in other datasets. We expect no real difference since different settings would require only different hyperparameters of our Gaussian Process model. However, ideally a theoretical analysis of the algorithms developed could be provided as well as performance guarantees, if possible, by examining the effect of different properties that could be relevant in this field like submodularity, locality and temporality. This will also allow the development of algorithms that are more scalable and potentially have better performance (requirement 6).

- The trust model could be expanded. It can be given a Bayesian treatment in order to take into consideration knowledge about users' behaviour and efficiently update this over time. Also, more types of attack could be considered. In particular, sophisticated attacks like 'on-off', where the user alternates between normal and malicious behaviour or collusion attacks, where more than one malicious user collaborates to cause more damage than each one acting alone.

- Finally, a real user study could be carried out to demonstrate the effectiveness of participatory sensing utilising our coordination algorithm. We firmly believe that a successful trial will promote the participatory sensing paradigm even further and move towards a widespread adoption of it.

Appendix A

Mobility Patterns Data

Table A.1: Human Mobility Patterns Dataset

Entry	UserId	Latitude	Longitude	Time
1	1	39.9847	116.3184	3.9744e+04
2	1	39.9847	116.3184	3.9744e+04
3	1	39.9847	116.3184	3.9744e+04
...	...	...	...	...
180	182	26.1622	119.9438	3.9919e+04
181	182	26.1615	119.9432	3.9919e+04
182	182	26.1619	119.9432	3.9919e+04

The full dataset can be found here: `https://www.microsoft.com/en-us/download/details.aspx?id=52367&from=https%3A%2F%2Fresearch.microsoft.com%2Fen-us%2Fdownloads%2Fb16d359d-d164-469e-9fd4-daa38f2b2e13%2F`

Appendix B

Air Quality Data

Table B.1: Air Quality Dataset

Entry	StationId	Latitude	Longitude	Time	Value
1	1001	40.0907	116.1736	4.1570e+04	74
2	1002	40.0040	116.2053	4.1570e+04	75
3	1003	39.9847	116.3184	4.1570e+04	85
...	...	...	...	...	...
20	1020	39.8865	116.4074	4.1570e+04	89
21	1021	39.8991	116.3954	4.1570e+04	147
22	1022	39.9210	116.4434	4.1570e+04	139

The full dataset can be found here: `https://www.microsoft.com/en-us/research/publication/u-air-when-urban-air-quality-inference-meets-big-data/`

References

Ioannis Agadakos, Jason Polakis, and Georgios Portokalidis. Techu: Open and privacy-preserving crowdsourced gps for the masses. In *Proceedings of the 15th Annual International Conference on Mobile Systems, Applications, and Services*, MobiSys '17, pages 475–487, New York, NY, USA, 2017. ACM. ISBN 978-1-4503-4928-4. URL: http://doi.acm.org/10.1145/3081333.3081345.

Andreas Albers, Ioannis Krontiris, Noboru Sonehara, and Isao Echizen. Coupons as monetary incentives in participatory sensing. In Christos Douligeris, Nineta Polemi, Athanasios Karantjias, and Winfried Lamersdorf, editors, *Collaborative, Trusted and Privacy-Aware e/m-Services*, volume 399 of *IFIP Advances in Information and Communication Technology*, pages 226–237. Springer Berlin Heidelberg, 2013. ISBN 978-3-642-37436-4.

Ashton Anderson, Daniel Huttenlocher, Jon Kleinberg, and Jure Leskovec. Steering user behavior with badges. In *Proceedings of the 22Nd International Conference on World Wide Web*, WWW '13, pages 95–106, Republic and Canton of Geneva, Switzerland, 2013. International World Wide Web Conferences Steering Committee. ISBN 978-1-4503-2035-1.

Paul M. Aoki, Honicky, Alan Mainwaring, Chris Myers, Eric Paulos, Sushmita Subramanian, and Allison Woodruff. A vehicle for research: Using street sweepers to explore the landscape of environmental community action. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '09, pages 375–384, New York, NY, USA, 2009. ACM. ISBN 978-1-60558-246-7.

Yoram Bachrach, Thore Graepel, Gjergji Kasneci, Michal Kosinski, and Jurgen Van Gael. Crowd iq: Aggregating opinions to boost performance. In *Proceedings of the 11th International Conference on Autonomous Agents and Multiagent Systems - Volume 1*, AAMAS '12, pages 535–542, Richland, SC, 2012. International Foundation for Autonomous Agents and Multiagent Systems. ISBN 0-9817381-1-7, 978-0-9817381-1-6.

Mitra Baratchi, Nirvana Meratnia, Paul J. M. Havinga, Andrew K. Skidmore, and Bert A. K. G. Toxopeus. A hierarchical hidden semi-markov model for modeling mobility data. In *Proceedings of the 2014 ACM International Joint Conference on Pervasive*

and Ubiquitous Computing, UbiComp '14, pages 401–412, New York, NY, USA, 2014a. ACM. ISBN 978-1-4503-2968-2.

Mitra Baratchi, Nirvana Meratnia, Paul J. M. Havinga, Andrew K. Skidmore, and Bert A. K. G. Toxopeus. A hierarchical hidden semi-markov model for modeling mobility data. In *Proceedings of the 2014 ACM International Joint Conference on Pervasive and Ubiquitous Computing*, UbiComp '14, pages 401–412, New York, NY, USA, 2014b. ACM. ISBN 978-1-4503-2968-2.

Jonathan Binney, Andreas Krause, and Gaurav S Sukhatme. Informative path planning for an autonomous underwater vehicle. In *Robotics and automation (icra), 2010 IEEE international conference on*, pages 4791–4796. IEEE, 2010.

D.F. Bizup and D.E. Brown. The over-extended kalman filter - don't use it! In *Information Fusion, 2003. Proceedings of the Sixth International Conference of*, volume 1, pages 40–46, July 2003.

Boman Brian, Wilson Chris, and Ontermaa Esa. Water quality monitoring programs for environmental assessment of citrus groves. 2008.

Azby Brown, Pieter Franken, Sean Bonner, Nick Dolezal, and Joe Moross. Safecast: successful citizen-science for radiation measurement and communication after fukushima. *Journal of Radiological Protection*, 36(2):S82, 2016.

J. Burke, D. Estrin, M. Hansen, A. Parker, N. Ramanathan, S. Reddy, and M. B. Srivastava. Participatory sensing. In *Workshop on World-Sensor-Web: Mobile Device Centric Sensor Networks and Applications*, pages 117–134, 2006.

W.F. Caselton and J.V. Zidek. Optimal monitoring network designs. *Statistics & Probability Letters*, 2(4):223 – 227, 1984. ISSN 0167-7152.

Kathryn Chaloner and Isabella Verdinelli. Bayesian experimental design: A review. *Statistical Science*, 10(3):273–304, August 1995.

Cen Chen, Shih-Fen Cheng, Aldy Gunawan, Archan Misra, Koustuv Dasgupta, and Deepthi Chander. Traccs: Trajectory-aware coordinated urban crowd-sourcing. In *Second AAAI Conference on Human Computation & Crowdsourcing (HCOMP)*, pages 30–40, 2014.

Cen Chen, Shih-Fen Cheng, Hoong Chuin Lau, and Archan Misra. Towards city-scale mobile crowdsourcing: Task recommendations under trajectory uncertainties. In *Proceedings of the 24th International Conference on Artificial Intelligence*, IJCAI'15, pages 1113–1119. AAAI Press, 2015. ISBN 978-1-57735-738-4.

Ron Chepesiuk. Decibel hell: The effects of living in a noisy world. *Environmental health perspectives*, pages A35–A41, 2005.

Yohan Chon, Nicholas D Lane, Yunjong Kim, Feng Zhao, and Hojung Cha. A large-scale study of mobile crowdsourcing with smartphones for urban sensing applications. *Proc. of ACM International Joint Conference on Pervasive and Ubiquitous Computing (UbiComp'13), Zurich, Switzerland*, page 1?10, 2013.

C.-Y. Chong and S.P. Kumar. Sensor networks: Evolution, opportunities, and challenges. *Proceedings of the IEEE*, 91(8):1247–1256, 2003. cited By 1910.

Amol Deshpande, Carlos Guestrin, and Samuel R. Madden. Resource-aware wireless sensor-actuator networks. *IEEE Data Engineering*, 28:2005, 2005a.

Amol Deshpande, Carlos Guestrin, Samuel R. Madden, Joseph M. Hellerstein, and Wei Hong. Model-based approximate querying in sensor networks, 2005b.

Ellie D'Hondt, Matthias Stevens, and An Jacobs. Participatory noise mapping works! an evaluation of participatory sensing as an alternative to standard techniques for environmental monitoring. *Pervasive and Mobile Computing*, 9(5):681–694, 2013.

Ellie D'Hondt, Jesse Zaman, Eline Philips, Elisa Gonzalez Boix, and Wolfgang De Meuter. Orchestration support for participatory sensing campaigns. In *Proceedings of the 2014 ACM International Joint Conference on Pervasive and Ubiquitous Computing*, UbiComp '14, pages 727–738, New York, NY, USA, 2014. ACM. ISBN 978-1-4503-2968-2.

Dominic DiPalantino, Thomas Karagiannis, and Milan Vojnovic. Individual and collective user behavior in crowdsourcing services. Number MSR-TR-2010-59. Microsoft Research, May 2010.

Akshay Dua, Nirupama Bulusu, Wu-Chang Feng, and Wen Hu. Towards trustworthy participatory sensing. In *Proceedings of the 4th USENIX Conference on Hot Topics in Security*, HotSec'09, pages 8–8, Berkeley, CA, USA, 2009. USENIX Association.

Martin Ester, Hans-Peter Kriegel, Jörg Sander, and Xiaowei Xu. A density-based algorithm for discovering clusters a density-based algorithm for discovering clusters in large spatial databases with noise. In *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining*, KDD'96, pages 226–231. AAAI Press, 1996.

M. Faulkner, M. Olson, R. Chandy, J. Krause, K.M. Chandy, and A. Krause. The next big one: Detecting earthquakes and other rare events from community-based sensors. In *Information Processing in Sensor Networks (IPSN), 2011 10th International Conference on*, pages 13–24, April 2011.

Ujwal Gadiraju, Ricardo Kawase, Stefan Dietze, and Gianluca Demartini. Understanding malicious behavior in crowdsourcing platforms: The case of online surveys. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing*

Systems, CHI '15, pages 1631–1640, New York, NY, USA, 2015. ACM. ISBN 978-1-4503-3145-6.

H. Gao, C. H. Liu, W. Wang, J. Zhao, Z. Song, X. Su, J. Crowcroft, and K. K. Leung. A survey of incentive mechanisms for participatory sensing. *IEEE Communications Surveys Tutorials*, 17(2):918–943, Secondquarter 2015. ISSN 1553-877X.

Sahil Garg, Amarjeet Singh, and Fabio Ramos. Efficient space-time modeling for informative sensing. In *Proceedings of the Sixth International Workshop on Knowledge Discovery from Sensor Data*, SensorKDD '12, pages 52–60, New York, NY, USA, 2012. ACM. ISBN 978-1-4503-1554-8.

Daniel Golovin and Andreas Krause. Adaptive submodularity: Theory and applications in active learning and stochastic optimization. *J. Artif. Int. Res.*, 42(1):427–486, September 2011. ISSN 1076-9757.

Marta C. Gonzalez, Cesar A. Hidalgo, and Albert-Laszlo Barabasi. Understanding individual human mobility patterns. *Nature*, 453(7196):779–782, June 2008.

Nathan Griffiths. Task delegation using experience-based multi-dimensional trust. In *Proceedings of the Fourth International Joint Conference on Autonomous Agents and Multiagent Systems*, AAMAS '05, pages 489–496, New York, NY, USA, 2005. ACM. ISBN 1-59593-093-0. URL: http://doi.acm.org/10.1145/1082473.1082548.

Carlos Guestrin, Andreas Krause, and Ajit Paul Singh. Near-optimal sensor placements in gaussian processes. In *Proceedings of the 22Nd International Conference on Machine Learning*, ICML '05, pages 265–272, New York, NY, USA, 2005. ACM. ISBN 1-59593-180-5.

Vitor Guizilini and Fabio Ramos. A nonparametric online model for air quality prediction. In *Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence*, AAAI'15, pages 651–657. AAAI Press, 2015. ISBN 0-262-51129-0.

Kyungsik Han, E.A. Graham, D. Vassallo, and D. Estrin. Enhancing motivation in a mobile participatory sensing project through gaming. In *Privacy, security, risk and trust (passat), 2011 ieee third international conference on social computing (socialcom)*, pages 1443–1448, Oct 2011.

David Hasenfratz, Olga Saukh, Silvan Sturzenegger, and Lothar Thiele. Participatory air pollution monitoring using smartphones. In *Proc. 1st International Workshop on Mobile Sensing: From Smartphones and Wearables to Big Data*, 2012a.

David Hasenfratz, Olga Saukh, and Lothar Thiele. On-the-fly calibration of low-cost gas sensors. *Wireless Sensor Networks*, pages 228–244, 2012b.

David Hasenfratz, Olga Saukh, Christoph Walser, Christoph Hueglin, Martin Fierz, and Lothar Thiele. Pushing the spatio-temporal resolution limit of urban air pollution

maps. In *Proceedings of the 12th International Conference on Pervasive Computing and Communications (PerCom 2014)*, pages 69–77, Budapest, Hungary, March 2014.

Geoffrey Hollinger and Sanjiv Singh. Proofs and experiments in scalable, near-optimal search by multiple robots. *Proceedings of Robotics: Science and Systems IV, Zurich, Switzerland*, 1, 2008.

Holger Hoos and Thomas Stützle. *Stochastic Local Search: Foundations & Applications*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 2004. ISBN 1558608729.

Humayun Irshad, Eun-Yeong Oh, Daniel Schmolze, Liza M Quintana, Laura Collins, Rulla M Tamimi, and Andrew H Beck. Crowdsourcing scoring of immunohistochemistry images: Evaluating performance of the crowd and an automated computational method. *Scientific Reports*, 7, 2017.

L.G. Jaimes, I. Vergara-Laurens, and M.A. Labrador. A location-based incentive mechanism for participatory sensing systems with budget constraints. In *Pervasive Computing and Communications (PerCom), 2012 IEEE International Conference on*, pages 103–108, March 2012.

N. R. Jennings, L. Moreau, D. Nicholson, S. Ramchurn, S. Roberts, T. Rodden, and A. Rogers. Human-agent collectives. *Commun. ACM*, 57(12):80–88, November 2014. ISSN 0001-0782.

M Jerrett, M Sears, C Giovis, R Burnett, P Kanaroglou, S Elliott, S Cakmak, P Gossilin, Y Bedard, J Maclachlan, et al. Intraurban air pollution exposure and asthma prevalence in hamilton, canada. *International Society of Exposure Analysis, Charleston, SC, USA*, 2001.

Michael Jerrett, Altaf Arain, Pavlos Kanaroglou, Bernardo Beckerman, Dimitri Potoglou, Talar Sahsuvaroglu, Jason Morrison, and Chris Giovis. A review and evaluation of intraurban air pollution exposure models. *Journal of Exposure Science and Environmental Epidemiology*, 15(2):185–204, 2005.

Stephen C. Johnson. Hierarchical clustering schemes. *Psychometrika*, 32(3):241–254, 1967. ISSN 1860-0980.

Audun Jøsang and Roslan Ismail. The beta reputation system. *15th Bled Electronic Commerce Conference*, pages 2502–2511, 2002.

Arnaud Jutzeler, Jason Jingshi Li, and Boi Faltings. A region-based model for estimating urban air pollution. In *Proceedings of the Twenty-Eighth AAAI Conference on Artificial Intelligence, July 27 -31, 2014, Québec City, Québec, Canada.*, pages 424–430, 2014.

Sarah N Lim Choi Keung and Nathan Griffiths. Building a trust-based social agent network. In *Proceedings of the 12th International Workshop on Trust in Agent Societies*, pages 68–79, 2009.

Andreas Krause. *Optimizing Sensing: Theory and Applications*. PhD thesis, Carnegie Mellon University, December 2008.

Andreas Krause, Carlos Guestrin, Anupam Gupta, and Jon Kleinberg. Near-optimal sensor placements: Maximizing information while minimizing communication cost. In *Proceedings of the 5th International Conference on Information Processing in Sensor Networks*, IPSN '06, pages 2–10, New York, NY, USA, 2006. ACM. ISBN 1-59593-334-4.

Andreas Krause, Ajit Singh, and Carlos Guestrin. Near-optimal sensor placements in gaussian processes: Theory, efficient algorithms and empirical studies. *J. Mach. Learn. Res.*, 9:235–284, June 2008. ISSN 1532-4435.

Philip J Landrigan. Air pollution and health. *The Lancet Public Health*, 2(1):e4 – e5, 2017. ISSN 2468-2667.

X. Liu, T. Xi, and E. Ngai. Data modelling with gaussian process in sensor networks for urban environmental monitoring. In *2016 IEEE 24th International Symposium on Modeling, Analysis and Simulation of Computer and Telecommunication Systems (MASCOTS)*, pages 457–462, Sept 2016.

Kian Hsiang Low, John M. Dolan, and Pradeep Khosla. Active markov information-theoretic path planning for robotic environmental sensing. In *The 10th International Conference on Autonomous Agents and Multiagent Systems - Volume 2*, AAMAS '11, pages 753–760, Richland, SC, 2011a. ISBN 0-9826571-6-1, 978-0-9826571-6-4.

Kian Hsiang Low, John M Dolan, and Pradeep Khosla. Active markov information-theoretic path planning for robotic environmental sensing. In *The 10th International Conference on Autonomous Agents and Multiagent Systems-Volume 2*, pages 753–760. International Foundation for Autonomous Agents and Multiagent Systems, 2011b.

Xiao Ma, Peng Huang, Xinxin Jin, Pei Wang, Soyeon Park, Dongcai Shen, Yuanyuan Zhou, Lawrence K Saul, and Geoffrey M Voelker. edoctor: Automatically diagnosing abnormal battery drain issues on smartphones. In *NSDI*, volume 13, pages 57–70, 2013.

Nurul Ashikin Bte Mabahwi, Oliver Ling Hoon Leh, and Dasimah Omar. Human health and wellbeing: Human health effect of air pollution. *Proceedings of Social and Behavioral Sciences*, 153:221 – 229, 2014. ISSN 1877-0428. International Conference on Quality of Life, The Pacific Sutera Hotel, Sutera Harbour, Kota Kinabalu, Sabah, Malaysia, 4-5 January 2014.

J. MacQueen. Some methods for classification and analysis of multivariate observations. In *Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability, Volume 1: Statistics*, pages 281–297, Berkeley, Calif., 1967. University of California Press.

Nicolas Maisonneuve, Matthias Stevens, and Bartek Ochab. Participatory noise pollution monitoring using mobile phones. *Info. Pol.*, 15(1,2):51–71, April 2010. ISSN 1570-1255.

Roman Marchant and Fabio Ramos. Bayesian optimisation for intelligent environmental monitoring. In *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 2242–2249. IEEE, 2012.

J. McInerney, A. Rogers, and N. R. Jennings. Learning periodic human behaviour models from sparse data for crowdsourcing aid delivery in developing countries. In *Conference on Uncertainty in Artificial Intelligence (UAI)*, pages 401–410, July 2013a.

James McInerney, Sebastian Stein, Alex Rogers, and Nicholas R. Jennings. Breaking the habit: Measuring and predicting departures from routine in individual human mobility. *Pervasive Mob. Comput.*, 9(6):808–822, December 2013b. ISSN 1574-1192.

Geoffrey J. McLachlan and David Peel. *Finite mixture models*. Wiley series in probability and statistics. J. Wiley & Sons, New York, 2000. ISBN 0-471-00626-2.

Alexandra Meliou, Andreas Krause, Carlos Guestrin, and Joseph M. Hellerstein. Nonmyopic informative path planning in spatio-temporal models. Technical Report UCB/EECS-2007-44, EECS Department, University of California, Berkeley, Apr 2007.

D. Mendez, A.J. Perez, M.A. Labrador, and J.J. Marron. P-sense: A participatory sensing system for air pollution monitoring and control. In *Pervasive Computing and Communications Workshops (PERCOM Workshops), 2011 IEEE International Conference on*, pages 344–347, March 2011.

Diego Mendez and Miguel A Labrador. A general framework for participatory sensing systems. *Journal of Networks*, 9(11):2995, 2014.

Baharan Mirzasoleiman, Ashwinkumar Badanidiyuru, Amin Karbasi, Jan Vondrák, and Andreas Krause. Lazier than lazy greedy. In *Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence*, AAAI'15, pages 1812–1818. AAAI Press, 2015. ISBN 0-262-51129-0.

Baharan Mirzasoleiman, Amin Karbasi, Rik Sarkar, and Andreas Krause. Distributed submodular maximization. *J. Mach. Learn. Res.*, 17(1):8330–8373, January 2016. ISSN 1532-4435.

Prashanth Mohan, Venkata N. Padmanabhan, and Ran Ramjee. Trafficsense: Rich monitoring of road and traffic conditions using mobile smartphones. In *In Microsoft Technical Report*, 2008.

Hayam Mousa, Sonia Ben, Omar Hasan, Osama Younes, Mohiy Hadhoud, and Lionel Brunie. Trust management and reputation systems in mobile participatory sensing applications : A survey. *Computer Networks*, 90:49–73, 2015. ISSN 1389-1286.

M.D. Mueller, David Hasenfratz, Olga Saukh, Martin Fierz, and Christoph Hueglin. Statistical modelling of particle number concentration in zurich at high spatio-temporal resolution utilizing data from a mobile sensor network. *Atmospheric Environment*, 126:171 – 181, 2016. ISSN 1352-2310.

G.L. Nemhauser, L.A. Wolsey, and M.L. Fisher. An analysis of approximations for maximizing submodular set functions. *Mathematical Programming*, 14(1):265–294, 1978. ISSN 0025-5610.

Nima Nikzad, Nakul Verma, Celal Ziftci, Elizabeth Bales, Nichole Quick, Piero Zappi, Kevin Patrick, Sanjoy Dasgupta, Ingolf Krueger, Tajana Šimunić Rosing, and William G. Griswold. Citisense: Improving geospatial environmental assessment of air quality using a wireless personal exposure monitoring system. In *Proceedings of the Conference on Wireless Health*, WH '12, pages 11:1–11:8, New York, NY, USA, 2012. ACM. ISBN 978-1-4503-1760-3.

Ruofei Ouyang, Kian Hsiang Low, Jie Chen, and Patrick Jaillet. Multi-robot active sensing of non-stationary gaussian process-based environmental phenomena. In *Proceedings of the 2014 International Conference on Autonomous Agents and Multi-agent Systems*, AAMAS '14, pages 573–580, Richland, SC, 2014. International Foundation for Autonomous Agents and Multiagent Systems. ISBN 978-1-4503-2738-1.

Christopher J. Paciorek and Mark J. Schervish. Nonstationary covariance functions for gaussian process regression. *Advances in neural information processing systems*, 16: 273–280, 2004.

Paritosh Padhy, Rajdeep K. Dash, Kirk Martinez, and Nicholas R. Jennings. A utility-based adaptive sensing and multihop communication protocol for wireless sensor networks. *ACM Trans. Sen. Netw.*, 6(3):27:1–27:39, June 2010. ISSN 1550-4859.

Elena Paoletti, Tommaso Bardelli, Gianluca Giovannini, and Leonella Pecchioli. Air quality impact of an urban park over time. *Procedia Environmental Sciences*, 4:10 – 16, 2011. ISSN 1878-0296.

Willy Passchier-Vermeer and Wim F. Passchier. Noise exposure and public health. *Environmental Health Perspectives*, 108:pp. 123–131, 2000. ISSN 00916765.

Joelle Pineau, Geoffrey Gordon, and Sebastian Thrun. Anytime point-based approximations for large pomdps. *Journal of Artificial Intelligence Research*, 27:2006, 2006.

B. Predic, Zhixian Yan, J. Eberle, D. Stojanovic, and K. Aberer. Exposuresense: Integrating daily activities with air quality using mobile participatory sensing. In *Pervasive Computing and Communications Workshops (PERCOM Workshops), 2013 IEEE International Conference on*, pages 303–305, March 2013.

M. Rahimi, M. Hansen, W.J. Kaiser, G. Sukhatme, and D. Estrin. Adaptive sampling for environmental field estimation using robotic sensors. In *Intelligent Robots and Systems, 2005. (IROS 2005). 2005 IEEE/RSJ International Conference on*, pages 3692–3698, Aug 2005.

N Ramanathan, F Alquaddoomi, H Falaki, D George, C Hsieh, J Jenkins, C Ketcham, B Longstaff, J Ooms, J Selsky, H Tangmunarunkit, and D Estrin. ohmage: An open mobile system for activity and experience sampling. In *Pervasive Computing Technologies for Healthcare (PervasiveHealth), 2012 6th International Conference on*, pages 203–204, May 2012.

Sarvapali D. Ramchurn, Claudio Mezzetti, Andrea Giovannucci, Juan A. Rodriguez-Aguilar, Rajdeep K. Dash, and Nicholas R. Jennings. Trust-based mechanisms for robust and efficient task allocation in the presence of execution uncertainty. *J. Artif. Int. Res.*, 35(1):119–159, June 2009. ISSN 1076-9757.

Carl Edward Rasmussen and Christopher K. I. Williams. *Gaussian Processes for Machine Learning*. The MIT Press, 2006. ISBN 026218253X.

Sasank Reddy, Deborah Estrin, Mark Hansen, and Mani Srivastava. Examining micropayments for participatory sensing data collections. In *Proceedings of the 12th ACM International Conference on Ubiquitous Computing*, Ubicomp '10, pages 33–36, New York, NY, USA, 2010a. ACM. ISBN 978-1-60558-843-8.

Sasank Reddy, Deborah Estrin, and Mani Srivastava. Recruitment framework for participatory sensing data collections. In *Proceedings of the 8th International Conference on Pervasive Computing*, Pervasive'10, pages 138–155, Berlin, Heidelberg, 2010b. Springer-Verlag. ISBN 3-642-12653-7, 978-3-642-12653-6.

Sasank Reddy, Katie Shilton, Jeff Burke, Deborah Estrin, Mark Hansen, and Mani Srivastava. Evaluating Participation and Performance in Participatory Sensing. *Proceedings of the International Workshop on Urban Community and Social Applications of Networked Sensing Systems UrbanSense08*, pages 4–8, 2008. ISSN 15361233.

S. Reece, S. Roberts, C. Claxton, and D. Nicholson. Multi-sensor fault recovery in the presence of known and unknown fault types. In *2009 12th International Conference on Information Fusion*, pages 1695–1703, July 2009.

Joshua Reich and Elizabeth Sklar. Robot-sensor networks for search and rescue. In *IEEE International Workshop on Safety, Security and Rescue Robotics*, volume 22, 2006.

Mauricio G.C. Resende and Celso C. Ribeiro. Greedy randomized adaptive search procedures: Advances, hybridizations, and applications. In Michel Gendreau and Jean-Yves Potvin, editors, *Handbook of Metaheuristics*, volume 146 of *International Series in Operations Research & Management Science*, pages 283–319. Springer US, 2010. ISBN 978-1-4419-1663-1.

Alireza Sahami Shirazi, Niels Henze, Tilman Dingler, Martin Pielot, Dominik Weber, and Albrecht Schmidt. Large-scale assessment of mobile notifications. *Proceedings of the 32nd Annual ACM Conference on Human Factors in Computing Systems - CHI '14*, pages 3055–3064, 2014.

Juan Sánchez-González, Jordi Pérez-Romero, Ramon Agustí, and Oriol Sallent. *On Learning Mobility Patterns in Cellular Networks*, pages 686–696. Springer International Publishing, Cham, 2016. ISBN 978-3-319-44944-9.

Mac Schwager, Philip Dames, Daniela Rus, and Vijay Kumar. *A Multi-robot Control Policy for Information Gathering in the Presence of Unknown Hazards*, pages 455–472. Springer International Publishing, Cham, 2017. ISBN 978-3-319-29363-9.

A. Seaton, D. Godden, W. MacNee, and K. Donaldson. Particulate air pollution and acute health effects. *The Lancet*, 345(8943):176 – 178, 1995a. ISSN 0140-6736.

A. Seaton, D. Godden, W. MacNee, and K. Donaldson. Particulate air pollution and acute health effects. *The Lancet*, 345(8943):176 – 178, 1995b. ISSN 0140-6736.

John H Seinfeld and Spyros N Pandis. *Atmospheric chemistry and physics: from air pollution to climate change*. John Wiley & Sons, 2012.

Yang Ting Shen, Yi Shiang Shiu, Wei Kuang Liu, and Pei Wen Lu. *The Participatory Sensing Platform Driven by UGC for the Evaluation of Living Quality in the City*, pages 516–527. Springer International Publishing, Cham, 2017. ISBN 978-3-319-58524-6.

M. C. Shewry and H. P. Wynn. Maximum entropy sampling. *Journal of Applied Statistics*, 14(2):165–170, 1987.

K. Shilton, N. Ramanathan, S. Reddy, V. Samanta, J. Burke, D. Estrin, M. Hansen, and M. Srivastava. Participatory design of sensing networks: Strengths and challenges. In *Proceedings of the Tenth Anniversary Conference on Participatory Design 2008*, PDC '08, pages 282–285, Indianapolis, IN, USA, 2008. Indiana University. ISBN 978-0-9818561-0-0.

Amarjeet Singh, Andreas Krause, Carlos Guestrin, William Kaiser, and Maxim Batalin. Efficient planning of informative paths for multiple robots. In *Proceedings of the 20th International Joint Conference on Artifical Intelligence*, IJCAI'07, pages 2204–2211, San Francisco, CA, USA, 2007. Morgan Kaufmann Publishers Inc.

Amarjeet Singh, Andreas Krause, Carlos Guestrin, and William J. Kaiser. Efficient informative sensing using multiple robots. *J. Artif. Int. Res.*, 34(1):707–755, April 2009. ISSN 1076-9757.

V. Sivaraman, J. Carrapetta, Ke Hu, and B.G. Luxan. Hazewatch: A participatory sensor system for monitoring air pollution in sydney. In *Local Computer Networks*

Workshops (LCN Workshops), 2013 IEEE 38th Conference on, pages 56–64, Oct 2013.

Chaoming Song, Zehui Qu, Nicholas Blumm, and Albert-László Barabási. Limits of predictability in human mobility. *Science*, 327(5968):1018–1021, 2010. ISSN 0036-8075.

Stephen A Stansfeld and Mark P Matheson. Noise pollution: non-auditory effects on health. *British Medical Bulletin*, 68(1):243–257, 2003.

Matthias Stevens. *Community memories for sustainable societies: The case of environmental noise.* PhD thesis, Vrije Universiteit Brussel, 2012.

Matthias Stevens and Ellie D'Hondt. Crowdsourcing of Pollution Data using Smartphones. In Maja Vukovic, Soundar Kumara, and Ohad Greenshpan, editors, *Workshop on Ubiquitous Crowdsourcing, held at Ubicomp '10 (September 26-29, 2010, Copenhagen, Denmark)*, September 2010.

R. Stranders, A. Farinelli, A. Rogers, and N. R. Jennings. Decentralised coordination of continuously valued control parameters using the max-sum algorithm. In *Proceedings of The 8th International Conference on Autonomous Agents and Multiagent Systems - Volume 1*, AAMAS '09, pages 601–608, Richland, SC, 2009. International Foundation for Autonomous Agents and Multiagent Systems. ISBN 978-0-9817381-6-1.

R. Stranders, E. Munoz De Cote, A. Rogers, and N. R. Jennings. Near-optimal continuous patrolling with teams of mobile information gathering agents. *Artif. Intell.*, 195: 63–105, February 2013. ISSN 0004-3702.

Ruben Stranders. *Decentralised Coordination of Information Gathering Agents.* PhD thesis, University of Southampton, November 2010.

Ruben Stranders, Francesco Maria Delle Fave, Alex Rogers, and Nick Jennings. A decentralised coordination algorithm for mobile sensors. In *Twenty-Fourth AAAI Conference on Artificial Intelligence*, pages 874–880, 2010. Event Dates: 11 - 15 July, 2010.

P. Szczytowski, A. Khelil, and N. Suri. Asample: Adaptive spatial sampling in wireless sensor networks. In *Sensor Networks, Ubiquitous, and Trustworthy Computing (SUTC), 2010 IEEE International Conference on*, pages 35–42, June 2010.

Phillip Taylor, Nathan Griffiths, Lina Barakat, and Simon Miles. Stereotype reputation with limited observability. In *Proceedings of the 19th International Workshop on Trust in Agent Societies (Trust@ AAMAS 2017)*, 2017.

Niwat Thepvilojanapong, Shin'ichi Konomi, Yoshito Tobe, Yoshikatsu Ohta, Masayuki Iwai, and Kaoru Sezaki. Opportunistic collaboration in participatory sensing environments. In *Proceedings of the Fifth ACM International Workshop on Mobility in*

the Evolving Internet Architecture, MobiArch '10, pages 39–44, New York, NY, USA, 2010. ACM. ISBN 978-1-4503-0143-5.

Niwat Thepvilojanapong, Tomoya Tsujimori, Hao Wang, Yoshikatsu Ohta, Yunlong Zhao, and Yoshito Tobe. Impact of incentive mechanism in participatory sensing environment. In *SMART 2013, The Second International Conference on Smart Systems, Devices and Technologies*, pages 87–92, 2013.

Alasdair Thomason, Nathan Griffiths, and Matthew Leeke. *Extracting Meaningful User Locations from Temporally Annotated Geospatial Data*, pages 84–90. Springer International Publishing, Cham, 2015. ISBN 978-3-319-19743-2.

Kshitij Tiwari, Valentin Honoré, Sungmoon Jeong, Nak Young Chong, and Marc Peter Deisenroth. Resource-constrained decentralized active sensing for multi-robot systems using distributed gaussian processes. *The 16th Conference on Control, Automation and Systems*, pages 13–18, 2016.

Matteo Venanzi, Alex Rogers, and N. R. Jennings. Crowdsourcing spatial phenomena using trust-based heteroskedastic gaussian processes. In *First Conference on Human Computation and Crowdsourcing (HCOMP)*, pages 182–189. AAAI Press, 2013.

Andrea Wiggins. ebirding: Technology adoption and the transformation of leisure into science. In *Proceedings of the 2011 iConference*, iConference '11, pages 798–799, New York, NY, USA, 2011. ACM. ISBN 978-1-4503-0121-3.

Wesley Willett, Paul M. Aoki, Neil Kumar, Sushmita Subramanian, and Allison Woodruff. Common sense community: Scaffolding mobile sensing and analysis for novice users. In *Proceedings of the 8th International Conference on Pervasive Computing*, Pervasive'10, pages 301–318, Berlin, Heidelberg, 2010. Springer-Verlag. ISBN 3-642-12653-7, 978-3-642-12653-6.

World-Bank. *The cost of air pollution : strengthening the economic case for action.* World Bank Group, Washington, D.C., 2016.

Catherine Wyler, Charlotte Braun-Fahrländer, Nino Künzli, Christian Schindler, Ursula Ackermann-Liebrich, André P Perruchoud, Philippe Leuenberger, Brunello Wüthrich, et al. Exposure to motor vehicle traffic and allergic sensitization. *Epidemiology*, 11 (4):450–456, 2000.

Jesse Zaman, Ellie D'Hondt, Elisa Gonzalez Boix, Eline Philips, Kennedy Kambona, and Wolfgang De Meuter. Citizen-friendly participatory campaign support. In *Pervasive Computing and Communications Workshops (PERCOM Workshops), 2014 IEEE International Conference on*, pages 232–235, March 2014.

Yu Zheng, Quannan Li, Yukun Chen, Xing Xie, and Wei-Ying Ma. Understanding mobility based on gps data. In *Proceedings of the 10th International Conference on*

Ubiquitous Computing, UbiComp '08, pages 312–321, New York, NY, USA, 2008. ACM. ISBN 978-1-60558-136-1.

Yu Zheng, Furui Liu, and Hsun-Ping Hsieh. U-air: When urban air quality inference meets big data. In *Proceedings of the 19th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '13, pages 1436–1444, New York, NY, USA, 2013. ACM. ISBN 978-1-4503-2174-7.

Yu Zheng, Xing Xie, and Wei-Ying Ma. Geolife: A collaborative social networking service among user, location and trajectory. *IEEE Data Eng. Bull.*, 33(2):32–39, 2010.

Yu Zheng, Lizhu Zhang, Xing Xie, and Wei-Ying Ma. Mining interesting locations and travel sequences from gps trajectories. In *Proceedings of the 18th International Conference on World Wide Web*, WWW '09, pages 791–800, New York, NY, USA, 2009. ACM. ISBN 978-1-60558-487-4.

Davide Zilli, Oliver Parson, Geoff V Merrett, and Alex Rogers. A hidden markov model-based acoustic cicada detector for crowdsourced smartphone biodiversity monitoring. In *Proceedings of the Twenty-Third International Joint Conference on Artificial Intelligence*, IJCAI'13, pages 2945–2951. AAAI Press, 2013. ISBN 978-1-57735-633-2.