Coordinating Measurements for Air Pollution Monitoring in Participatory Sensing Settings

Alexandros Zenonos, Sebastian Stein and Nicholas R. Jennings
Electronics and Computer Science
University of Southampton, UK
{az2g13,ss2,nrj}@ecs.soton.ac.uk

ABSTRACT

Environmental monitoring is important, as it allows authorities to understand the impact of potentially harmful environmental phenomena, such as air pollution, noise or temperature, on public health. To achieve this effectively, participatory sensing is a promising paradigm for large-scale data collection. In this approach, ordinary citizens (non-expert contributors) collect environmental data using low-cost mobile devices. However, these participants are generally self-interested agents having their own goals and making local decisions about where and when to take measurements, if any at all. This can lead to a highly inefficient outcome, where observations are either taken redundantly or do not provide sufficient information about key areas of interest. To address these challenges, a coordination system is necessary to guide and to coordinate participants. This paper proposes such a participatory sensing framework and presents a novel algorithm based on entropy and mutual information criteria, called Local Greedy Search (LGS), that takes into consideration knowledge about human mobility patterns and the inconvenience cost that is incurred by taking measurements. In particular, the algorithm uses a local search technique to map participants to measurements that need to be taken. We empirically evaluate our algorithm on real-world human mobility and air quality data and show that our coordination algorithm outperforms the state-of-the-art greedy and myopic algorithms. In particular, LGS gains 33.4% more information than the best benchmark in realistic city-scale scenarios with hundreds of agents.

Categories and Subject Descriptors

I.2.8 [**Problem Solving, Control Methods, and Search**]: Heuristic Methods; I.2.11 [**Distributed Artificial Intelligence**]: Coherence and Coordination; Intelligent Agents

General Terms

Algorithms, Experimentation, Measurement, Performance, Theory
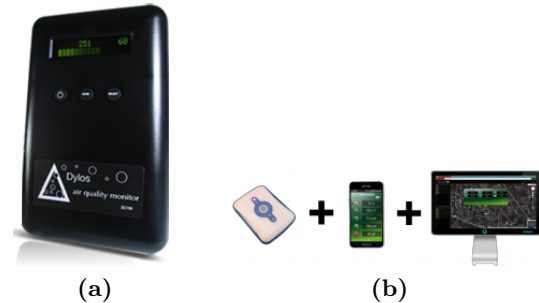
(a)　　　　　　(b)

Figure 1: Examples of portable devices that measure air quality in terms of atmospheric particulate matter (PM).

Keywords

Participatory Sensing; Collective Intelligence; Coordination; Gaussian Processes; Emissions; Air Quality; Teamwork; Human Mobility; Approximation Algorithms; Dynamic Environments

1. INTRODUCTION

Applications involving the placement of sensors for monitoring environmental phenomena, especially air pollution, are receiving considerable attention [5, 12], as it is a subject that concerns many, from environmental organisations to policy-makers and the general public. Air pollution is responsible for a range of heart-related diseases and leads to approximately 7 million annual deaths[1]. Given this, understanding the situation and predicting how it is going to change, in the long term as well as on a daily or even hourly basis, i.e., achieving *situational awareness*, 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 minimize the effect of high pollution areas in cities. They can also make planning decisions about new roads so as to handle traffic efficiently based on air pollution measurements. Furthermore, it can help doctors link environmental factors with symptoms and thus affect potential patients' treatment [1].

At present, the problem of monitoring air pollution is mainly tackled with networks of static stations. These are funded and operated by government authorities collecting measurements on a continuous basis, and they are controlled by a number of experts. These stations are very costly to

---

[1]http://www.who.int/mediacentre/news/releases/2014/air-pollution/en/

acquire and maintain, resulting in the collection of limited information [5].

However, another way to monitor environmental phenomena is to exploit participatory sensing, which is a promising paradigm for data collection [1]. Instead of making use of expensive equipment and employing a number of experts to work for hours to collect data, the burden is divided between a higher number of individuals (not necessarily experts), carrying cheap sensor devices, such as mobile phones and 'Senspod'[2], 'Sensordrone'[3] or 'Dylos'[4] as shown in Figure 1. This paradigm enables the public to gather and share local knowledge. The benefits of this approach are firstly that it is cheaper than the traditional approach of using static stations. Secondly, static sensors are located away from streets and emission sources in order to reflect the average pollution over an area [5]. Consequently, it might underestimate the true exposure of people to air pollution. Participatory sensing alleviates this by enabling people to directly make measurements at the sources of air pollution.

While delivering impressive results [3, 15], existing participatory campaigns lack an important element. They do not provide a coordination system that can efficiently guide or suggest to participants where and when to take measurements in order to fill coverage gaps throughout time in the area of interest, exploiting in this way the 'collective intelligence' of the crowd. 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 that the campaign initiator is interested in. Also, people may provide redundant information by taking measurements at the same time and place, which can waste participants' effort as well as communication and processing resources.

However, people participate in environmental campaigns either for their own personal/social interest, or for monetary incentives [4]. Crucially, it cannot be assumed that participants can provide an unlimited number of measurements for free as they cannot be seen as robotic entities that behave exactly as instructed. Instead, they act as self-interested agents having their own personal goals and limited information about the environment. Thus, in order to take measurements they typically incur inconvenience costs or even require financial compensation. In our work, we capture this by assuming a cost, different for each agent, for taking measurements. Moreover, humans are typically predictable in terms of their mobility habits [8]. So, even though knowledge about their daily routines could be available, no participatory sensing platform exploits it to suggest to individuals where to take measurements based on their future location.

Most current coordination mechanisms are computationally intensive and do not apply in participatory sensing as they focus on finding informative paths for autonomous agents and do not consider the mobility patterns of agents nor the cost of taking a measurement [13]. Some coordination approaches focus on decentralised approximations and deal with communication limitations [14], which is not a concern in this work because participants do not need to communicate with each other. Recent work [2], uses mobility patterns to coordinate agents in a different problem domain. The focus of that paper is assigning agents to tasks based

on their mobility patterns, so as to maximize the payoff of the tasks within a given time limit. However, no cost is associated with each agent to correspond to the inconvenience or the incentive needed to execute the task and tasks are completely independent from each other. Furthermore, once executed, they are no longer available, which is not the case when monitoring environmental phenomena.

In this paper, we address these shortcomings by proposing a complete participatory sensing framework which includes a novel algorithm that decides who should take measurements where and when, in order to achieve better situational awareness for a specified time period, while at the same time taking into consideration the cost of doing so. The algorithm is also able to incorporate knowledge available about human mobility patterns. Our algorithm uses local search and a greedy approach, extending them to trade off utility for speed, in order to make our approach feasible in city-scale scenarios. In more detail, the contributions of this paper are:

- We define the problem of coordinating measurements for participatory sensing applications and propose a framework for building participatory sensing applications for environmental monitoring.

- We develop a novel local search greedy algorithm that is the first to incorporate human mobility patterns 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.

- We empirically evaluate our algorithm on real human mobility pattern data and real air quality sensor data and show that it is 33.4% better in terms of total utility gained than the state-of-the-art greedy algorithm. We also demonstrate the generality of our approach by evaluating our algorithm on different environments by varying the dynamism of the observed phenomenon and showing that it is consistently better than the benchmarks in all scenarios.

The remainder of this paper is organised as follows: In Section 2 we introduce our framework. In Section 3 we formally define the problem. In Section 4 we introduce the technical details of the Gaussian process model we use. In Section 5 we detail our algorithm. In Section 6 we empirically evaluate it and present our experimental results, and finally we conclude in Section 7.

2. PARTICIPATORY SENSING FRAMEWORK

Our framework is designed to provide a platform for participatory sensing campaigns in order to efficiently monitor an environment while taking into consideration knowledge available about the participants. In particular our framework consists of four components: (a) the main component is the coordination algorithm, which is the main contribution of this work, (b) the human mobility patterns component, which is a system for making predictions about the mobility patterns of the participants, (c) the cost component, which captures the cost incurred by each participant for taking
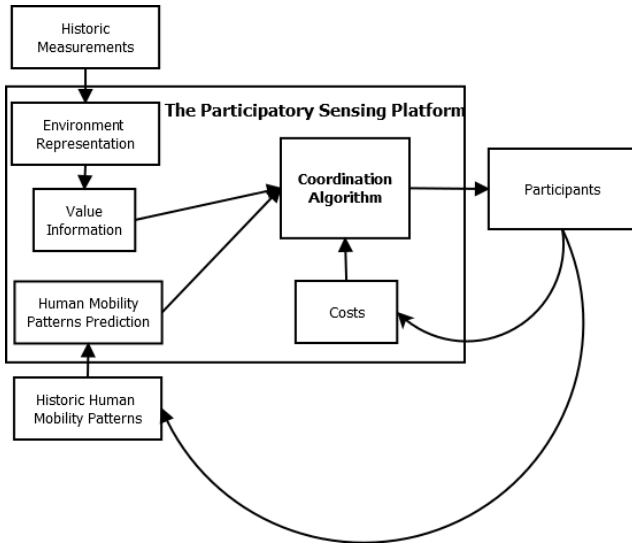
---

[2] http://v2.sensaris.com/products/

[3] http://sensorcon.com/products/
sensordrone-multisensor-tool

[4] http://www.dylosproducts.com/

Figure 2: A conceptual architecture of the coordination system

measurements and (d) the environmental phenomenon component, which supplies the system with information about the environment being monitored, given past measurements. Figure 2 shows the overall architecture of the participatory sensing framework and illustrates how each component interacts with each other.

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 advance in order to inform them about the measurements they need to take.

The participants are in a feedback loop, where they provide the platform with their mobility patterns and cost value. Specifically, intelligent agents on participants' devices can monitor their behaviour and provide the platform with the mobility patterns. Also, each participant is associated with a cost of taking measurements per timestep, which can be given directly by participants or learned from their participation in previous campaigns with the assistance of the intelligent agents.

A mobility pattern prediction system infers their future mobility patterns for a specific time horizon producing a number of possible routines according to a probability distribution. Also, the cost of each participant at each time is associated with uncertainty since some participants might be more predictable than others in terms of their willingness to take specific measurements. The coordination algorithm suggests to participants where and when to take measurements, in order to efficiently monitor the environment for a time period. In this paper, we do not focus on a complete implementation of the aforementioned framework but rather focus on the algorithmic challenge of developing an efficient coordination algorithm that exploits knowledge about participants' mobility patterns as well as the cost for taking measurements. Specifically, due to the regularities that human mobility patterns exhibit [8], we assume that only a single routine is predicted for each agent for a specific period of time. Therefore, the full probabilistic treatment of

mobility patterns is left for future work. Also, for simplicity but without violating the underpinning idea of the framework we assume deterministic cost to each agent, leaving the uncertainty about costs for future work.

3. THE COORDINATION PROBLEM

This section formally introduces the problem that the coordination algorithm needs to solve. Let a participatory sensing campaign be a collection of observations[5] $\mathbf{O}$ constrained on geography, duration, context and users such that $\mathbf{O}$ is collected participatively by a set of agents $\mathbf{A}$. An environment is denoted by a tuple $\mathcal{E} = (\mathbf{A}, L, T)$, which consists of: a set of agents $\mathbf{A} = \{\mathcal{A}_1, \ldots, \mathcal{A}_M\}$, a set of spatial coordinates $L = \{l_1, \ldots, l_N\}$ where measurements can be taken, and a discrete set of temporal coordinates $T = \{1, 2, 3, \ldots E\}$ that specify when a measurement can be taken, where E is the end of the campaign.

In order to define the problem of coordinating agents to monitor an environment $\mathcal{E}$ we must firstly define the following:

Definition 1. A set of spatio-temporal observation coordinates $O = (L \cup \{\bot\}) \times T$.

An element $o \in O$ is called an observation. The observation made by $\mathcal{A}_i$ at time t is denoted by $o_{i,t}$. 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 $O_t \subseteq \mathbf{O}_t$. If no observations at a time step are made by an agent i, we record a null observation $o_{i,t} = (\bot, t)$.

Each location $l \in L$ and time $t \in T$ is associated with a random variable $X_{l,t}$ which describes the environmental phenomenon. We use $X_{l,t} = x_{l,t}$ to refer to the realisation of that variable for specific spatio-temporal coordinates. When an observation is made at some spatio-temporal coordinate, the realisation $x_{l,t}$ becomes known. All observations made are associated with a utility value as follows:

Definition 2. A utility function $u : 2^O \rightarrow \mathbb{R}^+$, assigns a utility value to a set of observations.

The real value assigned by the utility function is based on the entropy given by the Bayesian D-optimality criterion [6]. This is a criterion similar to mutual information described in the aforementioned work, but since we are interested in dynamic environments rather than static ones, we take into consideration all the locations at all timesteps instead of predicting the values only at unobserved locations. The D-optimality criterion, in the context of environmental monitoring, measures the reduction of entropy at all locations L within the environment by making a set of observations O_t. In other words, this criterion provides the mutual information between observation sets $\mathbf{O}_{t-1}$, which represents the observations made until the previous timestep, and O_t, which represents the observations made at the current timestep. Intuitively, it is proportional to the uncertainty without making any observations minus the uncertainty when making those observations. This will result in a number of values, one for each spatial location $l_1, \ldots, l_N$, representing the information gained at those coordinates by making the aforementioned observations. In particular, we use $X_{L,t|\mathbf{O}_{t-1}}$ to describe the belief at time t about the random variable over

---

[5]Observations and measurements are used interchangeably.

the set of all locations, given that the agents made certain observations in the past $\mathbf{O}_{t-1}$. We also denote by $X_{L,t|O_t}$ the belief at time t at all the locations given that a set of observations has actually been made at time t. For simplicity in the notation, and unless stated otherwise we use $X_y = X_{L,t|\mathbf{O}_{t-1}}$ and $X_A = X_{L,t|O_t}$. Therefore, realised measurements over all locations given a set of observations can be expressed as $X_A = x_A$. The D-optimality criterion is given by:

$$I_l(X_y; X_A) = H_l(X_y) - H_l(X_y|X_A) \qquad (1)$$

where $H_l(X_y)$ is a scalar representing the entropy at a specific location of the environment taking into consideration the set of observations made in the past, and $H_l(X_y|X_A)$ is a scalar representing the entropy at a specific location of the environment given that observations are made at time t. Finally, the utility for a set of observations made at time t, can be expressed as follows:

$$u(O_t) = \sum_{l=l_1}^{l_N} I_l(X_y; X_A) \qquad (2)$$

where $u(O_t)$ is the summation of the mutual information from all the locations. This results in a single utility value for all the locations. However, by making an observation users incur a cost, which can be defined in terms of our model as follows:

Definition 3. A cost function $c_i : O \to \mathbb{R}^+$ assigns a cost to each agent for making an observation at a particular location.

So, asking an agent i to take an observation at location l and time t incurs a cost $c_i(l, t)$. This could represent the inconvenience incurred by users when making the observation, the financial compensation that is offered or the distance users have to travel to make that observation. We also define the cost function for an agent i that makes no observation at time t to be $c_i(\perp, t) = 0$. We denote by $\mathbb{U}$ the total utility earned by all the agents at time t which is given by:

$$\mathbb{U}(O_t) = \left(u(O_t) - \sum_{i=1}^{M} c_i(l, t) \right) \qquad (3)$$

where M is the total number of agents in the campaign. Since users are never asked to deviate from their routes in our problem, we simplify the cost notation to $c_i(t)$. The utility earned so far up to time t from all the agents can be expressed as:

$$\mathbb{U}(\mathbf{O}_t) = \sum_{j=1}^{t} \mathbb{U}(O_j) = \sum_{j=1}^{t} \left[u(O_j) - \sum_{i=1}^{M} c_i(j) \right] \qquad (4)$$

Given this nomenclature, the optimisation problem can be described as follows: at a set of spatio-temporal coordinates $W \subseteq L \times T$, we ask a set of agents $\mathbf{A}$ to take observations that maximize the difference between the utility and the cost functions $\mathbb{U}(\mathbf{O}_E)$ throughout the campaign. Concretely, we are looking for a mapping $s : \mathbf{A} \times T \to (L \cup \{\perp\})$ that is a total function and determines which agents should make which observations to reach an optimal solution to the problem. Formally,

$$S^* = arg\,max_s\, \mathbb{U}(\mathbf{O}_E) \qquad (5)$$

Note that the mapping s gives us the set of observations $\mathbf{O}_E$, which is the union of all the observations taken from all the agents. Thus, a solution to this problem is a policy S^* that associates a number of agents participating in the campaign with a number of observations taken from each one of them. In order to apply this problem in practice a model of the environment is needed that our utility function will be based on. Concretely, a mathematical way of expressing the utility defined is needed. Gaussian processes provide a straightforward relationship between the D-optimality criterion and predictive variance, as described next.

4. MODELLING THE PHENOMENON

As shown in [6], we can assume that the measurements of an environmental phenomenon, i.e., air pollution measurements ($\mathbf{O}_E$), have a multivariate Gaussian joint distribution over all the locations and timesteps where a measurement is taken. It is an effective way of capturing the relationship of different spatio-temporal coordinates with respect to an environmental phenomenon. Since we are interested in the air pollution values over the entire environment (L) for a time period and not just on the locations where measurements are taken, we make use of a regression technique to do so. In particular, we use Gaussian processes (GPs), which are a class of nonparametric probabilistic models and can generalise the multivariate Gaussians to an infinite number of random variables and thus generalise over the entire set of locations (L) and timesteps (T)[10]. The main advantages of GPs are that they can capture structural correlations of a spatio-temporal phenomenon as well as provide 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. The GP can be fully specified by its mean function $m(\mathbf{x})$ and a covariance function (also referred to as kernel) $k(\mathbf{x}, \mathbf{x}')$. It can be viewed as a distribution over functions, where every random variable represents a value of a function f at a specific point. Formally,

$$f(\mathbf{x}) \sim \mathcal{GP}(m(\mathbf{x}), k(\mathbf{x}, \mathbf{x}')) \qquad (6)$$

For simplicity in notation we denote the mean vector of some set of random variables X_A as μ_A. Suppose that a column vector x_A of realised measurements is available. The GP can then exploit these observations to make predictions for any location as well as provide their corresponding predictive uncertainty. The distribution of X_y, given these observations, is Gaussian with mean $\mu_{y|A}$ and variance $\Sigma_{y|A}$ which are given by:

$$\begin{aligned} \mu_{y|A} &= \mu_y + \Sigma_{yA}\Sigma_{AA}^{-1}(x_A - \mu_A) \\ \Sigma_{y|A} &= \Sigma_{yy} - \Sigma_{yA}\Sigma_{AA}^{-1}\Sigma_{Ay} \end{aligned} \qquad (7)$$

There is a lot of discussion around which kernel to use for each problem. Admittedly, air pollution could be modelled using a composite non-stationary covariance function taking into consideration the atmospheric diffusion and the change of smoothness of the function depending on the location. However, in order to preserve time efficiency a common choice of covariance function is Matérn [5, 9]:

$$k(\mathbf{x}, \mathbf{x}') = \sigma_f^2(1 + \sqrt{3}r)\exp(-\sqrt{3}r) + \sigma_n^2\delta_{\mathbf{x}, \mathbf{x}'} \qquad (8)$$

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

$\theta = \{l_1, l_2, l_3, \sigma_f^2, \sigma_n^2\}$ are the hyperparameters (parameters of the covariance function) 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 over y-axis, l_3 over time[6] . Intuitively, (l_1, l_2, l_3) capture 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 about predictions made further away from the observed points, σ_n^2 is the noise variance that controls the percentage of the data variation that can be attributed to noise, and $\delta_{\mathbf{x}, \mathbf{x}'}$ is the Kronecker delta defined as follows:

$$\delta_{xx'} = \begin{cases} 1, & \text{if } \mathbf{x} = \mathbf{x}', \\ 0, & \text{if } \mathbf{x} \neq \mathbf{x}' \end{cases} \tag{9}$$

This approach enables the GP to be updated both with the spatial aspect, as well as the temporal aspect of the phenomenon. In our work, hyperparameters are assumed to be initially unknown, but we exploit historic data provided from a number of static air pollution stations [16] to train the model. In particular, in order to learn the hyperparameters we use maximum likelihood estimation (MLE). That is finding the parameters θ that maximize the log marginal likelihood (ML) $\log p(x_A|\theta) =$

$$-\frac{1}{2}(x_A - \mu_A)^T \Sigma_{AA}^{-1}(x_A - \mu_A) - \frac{1}{2}\log|\Sigma_{AA}| - \frac{n}{2}\log 2\pi \tag{10}$$

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:

$$\begin{aligned} H(X_y|X_A) &= \frac{1}{2}\log(2\pi e\sigma_{X_y|X_A}^2) \\ H(X_y|X_A) &= \frac{1}{2}\log(\sigma_{X_y|X_A}^2) + \frac{1}{2}(\log(2\pi) + 1) \end{aligned} \tag{11}$$

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

5. GREEDY LOCAL SEARCH ALGORITHM

Finding an optimal solution to this particular problem is computationally infeasible, especially when large geographic areas are monitored for a number of days with hundreds of agents. This is because this category of optimisation problems is known to be NP-hard [6]. Thus, in this work, we concentrate on heuristic methods and in particular local search and greedy approach. In this section, we describe our algorithm in detail.

5.1 Algorithm Design

For each timestep all agents are assigned a binary value. This value indicates whether or not the specific individual should take a measurement at that specific time. In this paper, we assume knowledge of the mobility patterns of the participants in order to efficiently plan ahead in terms of where and when to take measurements. This is feasible as shown in [8, 7] by analysing the historic movements of participants. In our work, we can only ask someone to take a

measurement if we know their exact location at that point in time. So, each agent has either a single GPS location at any point or no information is available at all[7]. Also, given that 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, we are able to construct policies of people taking measurements at different locations and times in an offline manner. Thus, a policy can be represented by a binary matrix where each row represents a timestep (T) and each column an agent ($\mathcal{A}_i$). The null policy, which is the policy where nobody takes any measurements, is represented by the zero matrix.

The algorithm uses a local search technique to reach a local maximum [11]. The idea is to find the best state among a set of possible states according to the objective function defined. Generally, the way local search achieves finding the best state is by starting at a state, say the zero matrix, and move to another candidate solution by applying local changes, i.e., move to a neighbour of that state as defined in the context of each particular problem, say adding a bit in different positions of the matrix at a time, until an acceptance criterion is met. Given this brief introduction to the local search technique, we will, both formally (Algorithm 1) and intuitively, describe the algorithm developed.

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 4. In this way, a subset p of possible measurements, other than cases where agents are known to be unavailable, are checked one by one (lines 6). This enables 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 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)[8]. 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), and attempts to add another measurement to the matrix (line 10), and again all possible positions, but 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 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.

We have $2^{(M \cdot E)}$ possible combinations since we have an independent option for whether or not to take a measure-

---

[6] Note that l_1, l_2, l_3 may also be used in the context of problem description denoting locations $1, 2, 3$ of L.

[7] When no information is available for an agent at a specific timestep, the value is automatically set to 0 which means no measurement should be taken.

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

Algorithm 1 Local Greedy Search Algorithm (LGS)

1: **input**: E (timesteps), A (agents)
2: Initialise $M = |A|$, $maxU' = 0$, $S^* \leftarrow$ null matrix(E, M), $obsList = null$
3: **for** $k = 1\, to\, (M \cdot E)$ **do**
4:    $z \leftarrow$ null positions of S^*, $newobs = S^*$
5:    $p \subset z$, $sz \leftarrow |p|$
6:    **for** $l = 1\, to\, sz + 1$ **do**
7:      **if** $l = sz + 1$ **then**
8:        Change $obsList(k - 2)$ to 0 in $newobs$ matrix
9:      **else**
10:        Change l^{th} zero bit to 1 in $newobs$ matrix
11:      **end if**
12:      Set $C = 0$
13:      **for** $i = 1\, to\, E$ **do**
14:        **for** $j = 1\, to\, M$ **do**
15:          **if** $newobs(i, j) = 1$ **then**
16:            $C \leftarrow c_j(i) + C$ {calculate the cost for each agent and add it to the total cost}
17:          **end if**
18:        **end for**
19:        $\mathbb{U}(O_i) \leftarrow [u(O_i) - C]$ {calculate the total utility given what observations are made}
20:      **end for**
21:      $s_l \leftarrow \mathbb{U}(\mathbf{O}_E)$
22:    **end for**
23:    Keep the maximum $\mathbb{U}(\mathbf{O}_E)$ of s_l in $maxU$ variable
24:    Add/Remove observation from $obsList$
25:    **if** $maxU < maxU'$ **then**
26:      **return**: S^*
27:    **else**
28:      Set S^* to be the best configuration
29:    **end if**
30:    $maxU' \leftarrow maxU$
31: **end for**
32: **return**: S^*

---

ment 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:

$$\frac{M \cdot E(M \cdot E + 1)}{2} \tag{12}$$

6. EMPIRICAL EVALUATION

In this section, we will evaluate the algorithm developed in the previous section in different participatory sensing scenarios. In the first part of the empirical evaluation we present the benchmarks our algorithm is contrasted with. Next, we state our hypotheses and describe the experiments performed. Finally, we present and analyze our findings.

6.1 Benchmarks
The algorithm developed is benchmarked against six algorithms which are described below:

- **Greedy**: This algorithm checks which measurements should be taken in order to maximize the utility at each timestep, i.e., maximize $\mathbb{U}(O_1), \mathbb{U}(O_2), \cdots, \mathbb{U}(O_E)$ sequentially. It does so in a greedy way, i.e., select the single observation among the number of agents that maximizes 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 [6].

- **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 behavior of mobile sensors, i.e., patrolling an area in order to monitor environmental phenomena as in the work of [14].

- **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, finding the optimal assignment of agents to observations that maximize 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.

- **Random**: This algorithm assumes that measurements are made randomly by agents throughout time. 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 S^* 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 of those. This is only feasible to do in small-scale scenarios.

6.2 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:* LGS will be faster than the optimal.

This is due to the fact that the optimal policy is found by evaluating all possible coordination policies, which is exponential in terms of the number of the participants and the timesteps of the campaign.

- *Hypothesis 3:* The total utility earned by the LGS algorithm will be higher than Greedy, MyopicOpt, Patrol, Random100 and Random in all scenarios of varying dynamism.

This is because LGS aims at increasing the total utility by computing policies irrespective of the phenomenon. 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.

6.3 Experimental Setup

In order to empirically evaluate our algorithm, we compare its performance against the six 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. Table 1 shows the air quality index for air quality in Beijing. We use an air quality dataset [16] which contains one year's (2013-2014) fine grained air quality data from static air quality monitoring stations in Beijing. We use this data to train our GP model, and in particular learn the hyperparameters. These include the dynamism of the phenomenon (l_3) and smoothness over latitude and longitude (l_1, l_2). The sensors are scattered in Beijing and take measurements every hour. Figure 3 shows the stations and the state of the environment represented by a GP for a particular timestep. 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.

Ideally, at the same time the human mobility patterns are learned using a human mobility prediction system. In this work, h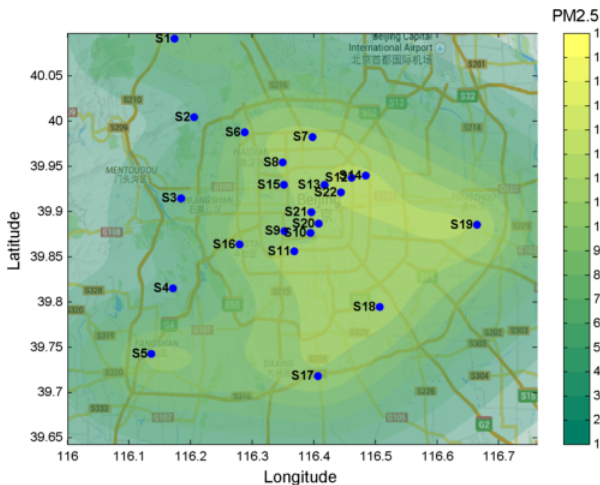owever, we use data from Geolife trajectories dataset [17] which contains sequences of time-stamped locations of 182 humans in Beijing over a period of 5 years (2007-2012). For our experiments, we extracted the patterns of 108 humans over a period so as to get as much overlap between collected patterns as possible. In order to test our system for more than 108 agents, we take patterns of different months from the same pool of agents' trajectories. Humans are assumed to be equipped with the necessary equipment and they are able to take measurements when necessary if their spatial coordinates are available. 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 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. This way we capture the cost which would be associated with participants in a real deployment.

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, MyopicOpt, Random and Random100. The optimal algorithm is infeasible to run in city-scale scenarios.

To make our system generally applicable, we experiment with a number of artificial environments by altering the hyperparameters, and in particular l_3, which controls the dynamism of the environment. This change shows how our algorithm will perform potentially in other cities or for phenomena with other levels of dynamism. In the second experiment we simulate a varying number 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. Experimenting with small-scale scenarios will enable us to compare our algorithm with the optimal one.

6.4 Evaluation

6.4.1 Effect of the number of agents

Figure 4 shows results of the performance of varying the number M of agents participating in the system. The dy-
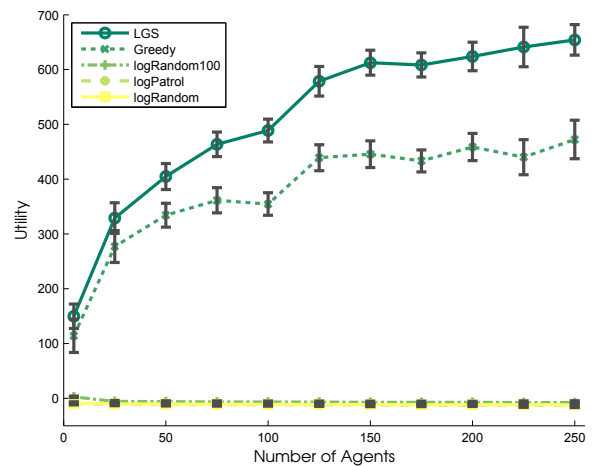

Figure 3: Air quality measurement stations in Beijing overlaid by air quality measurements predicted by GP.

Figure 4: Total utility gained for a 5-day participatory sensing campaign. The error bars indicate the 95% confidence interval.

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

namism in this experiment is fixed at $l_3 = 10.6$, which was found using the MLE technique. The results confirm our first hypothesis that LGS will outperform 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.

6.4.2 Effect of the dynamism of the phenomenon

Figure 5 shows results of the performance of the algorithms when varying the time-scale (l_3) which controls the dynamism of the phenomenon. Originally, 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 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 consecutively (in every timestep). However, the dynamism of air pollution was at the scale of tens of hours, which make Greedy and far from optimal. The results confirm our third hypothesis as LGS is better than the rest of the algorithms in all scenarios. However, as time-scale approaches zero, LGS's performance tends to be similar to MyopicOpt and Greedy. Also, the utility gained from LGS is near the optimal one. Figure 6 shows results of the time efficiency of the algorithms when varying time-scale (l_3). LGS needs more time in dynamic environments as more measurements need to be taken and is generally slower than Greedy, MyopicOpt, Patrol and Random algorithms. However, the optimal
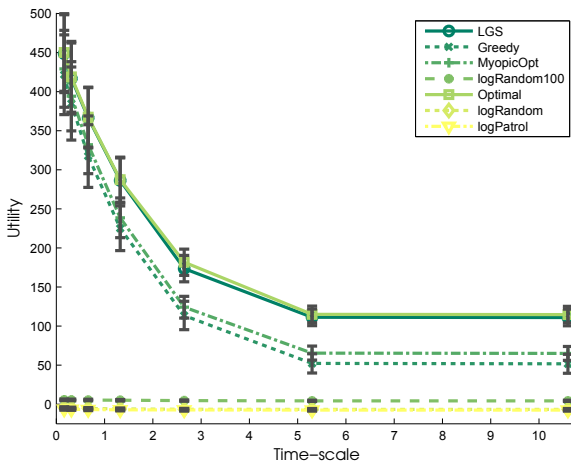
algorithm requires a lot more time, confirming our second hypothesis. Specifically, Figure 6 includes the natural logarithms of LGS, Greedy and the optimal algorithm's runtime.
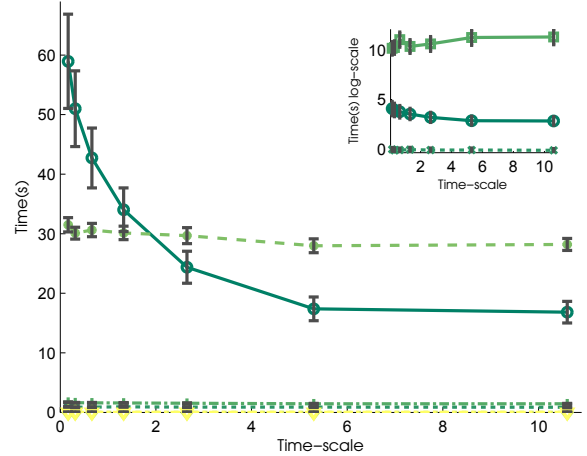

Figure 6: **Average runtime for a 1-day participatory sensing campaign. The error bars indicate the 95% confidence interval.**

7. CONCLUSIONS

In this paper, we introduced the problem of coordinating measurements in participatory sensing settings and developed a novel algorithm that maximizes the total utility gained over a period of time while at the same time minimizing the cost incurred by taking measurements. In particular, we demonstrate how efficient the algorithm is compared to the state-of-the-art Greedy algorithm and Brute force approach. An empirical evaluation on real data showed that (a) LGS is 33.4% better than the Greedy algorithm (b) LGS is faster than the Brute Force approach (c) dynamic environment affects the performance of LGS algorithm and the total utility gained, but still outperforms the benchmarks in all scenarios. Our future work in this area is to extend the algorithm to deal with more agents in campaigns running for a longer period of time. Also, we need to deal with the problem of having uncertainty over the human mobility patterns and inconvenience cost. Furthermore, we would like to investigate the performance of other algorithms including variations of our algorithm. Finally, conducting a real-world trial would be an interesting avenue to follow.

8. ACKNOWLEDGMENTS

Figure 5: **Total utility gained for a 1-day participatory sensing campaign. The error bars indicate the 95% confidence interval.**

REFERENCES

[1] 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.

[2] C. Chen, S.-F. Cheng, A. Gunawan, A. Misra, K. Dasgupta, and D. Chander. Traccs: Trajectory-aware coordinated urban crowd-sourcing. In *Second AAAI Conference on Human Computation & Crowdsourcing (HCOMP)*, pages 30–40, 2014.

[3] E. D'Hondt, M. Stevens, and A. 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.

[4] L. Jaimes, I. Vergara-Laurens, and M. 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.

[5] A. Jutzeler, J. J. Li, and B. 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.

[6] A. Krause, A. Singh, and C. Guestrin. Near-optimal sensor placements in gaussian processes: Theory, efficient algorithms and empirical studies. *J. Mach. Learn. Res.*, 9:235–284, June 2008.

[7] 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 2013.

[8] J. McInerney, S. Stein, A. Rogers, and N. R. Jennings. Breaking the habit: Measuring and predicting departures from routine in individual human mobility. *Pervasive Mob. Comput.*, 9(6):808–822, Dec. 2013.

[9] R. Ouyang, K. H. Low, J. Chen, and P. 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.

[10] C. E. Rasmussen and C. K. I. Williams. *Gaussian Processes for Machine Learning*. The MIT Press, 2006.

[11] S. J. Russell and P. Norvig. *Artificial Intelligence: A Modern Approach*. Pearson Education, 2003.

[12] J. H. Seinfeld and S. N. Pandis. *Atmospheric chemistry and physics: from air pollution to climate change*. John Wiley & Sons, 2012.

[13] A. Singh, A. Krause, and W. J. Kaiser. Nonmyopic adaptive informative path planning for multiple robots. In *IJCAI*, pages 1843–1850, 2009.

[14] 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, Feb. 2013.

[15] M. Whitney and H. Richter Lipford. Participatory sensing for community building. In *CHI '11 Extended Abstracts on Human Factors in Computing Systems*, CHI EA '11, pages 1321–1326, New York, NY, USA, 2011. ACM.

[16] Y. Zheng, F. Liu, and H.-P. 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.

[17] Y. Zheng, L. Zhang, X. Xie, and W.-Y. 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.