Proceedings of the Fifth Conference on
Human Computation and Crowdsourcing
(HCOMP 2017)

# A Trust-Based Coordination System for Participatory Sensing Applications

**Alexandros Zenonos, Sebastian Stein**
University of Southampton, UK
{az2g13,ss2}@ecs.soton.ac.uk

**Nicholas R. Jennings**
Imperial College London, UK
n.jennings@imperial.ac.uk

## Abstract

Participatory sensing (PS) has gained significant attention as a crowdsourcing methodology that allows ordinary citizens (non-expert contributors) to collect data using low-cost mobile devices. In particular, it has been useful in the collection of environmental data. However, current PS applications suffer from two problems. First, they do not coordinate the measurements taken by their users, which is required to maximise system efficiency. Second, they are vulnerable to malicious behaviour. In this context, we propose a novel algorithm that simultaneously addresses both of these problems. Specifically, we use heteroskedastic Gaussian Processes to incorporate users' trustworthiness into a Bayesian spatio-temporal regression model. The model is trained with measurements taken by participants, thus it is able to estimate the value of the phenomenon at any spatio-temporal location of interest and also learn the level of trustworthiness of each user. Given this model, the coordination system is able to make informed decisions concerning when, where and who should take measurements over a period of time. We empirically evaluate our algorithm on a real-world human mobility and air quality dataset, where malicious behaviour is synthetically produced, and show that our algorithm outperforms the current state of the art by up to 60.4% in terms of RMSE while having a reasonable runtime.

Participatory sensing (PS) is a low-cost, but large-scale, data collection paradigm that relies on ordinary people (non-experts) collecting information using mobile devices they carry on them. It has been successfully used in monitoring environmental phenomena, such as radiation, air and noise pollution (Stevens and D'Hondt 2010; Brown et al. 2016). The collection of this information facilitates a better understanding of these phenomena and assists the authorities in taking actions related to better urban planning and preserving public health. For example, it may drive decisions about the best possible locations to build a new park, parking spots or the construction of roads.

However, people are typically willing to take only a limited number of measurements per day (which can be seen as a budget) and have only limited information about the environment to be monitored. Since they do not have complete knowledge of the state of the environment nor how it is going to change over time, they are not capable of making optimal decisions about when and where to take measurements to maximise the collected information. Thus, as argued in (Stevens and D'Hondt 2010; Zenonos, Stein, and Jennings 2015a; 2015b), a coordination system is necessary to guide people on when and where to take measurements, which is important as it maximises the information learned about the environment using minimum effort by the users.

A second key challenge in participatory sensing applications is that the very openness of this data collection 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 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. Recent work (Gadiraju et al. 2015), studies the prevalence of malicious users in crowdsourcing settings. Specifically, 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). 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, like 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, Rogers, and Jennings 2013), has shown that probabilistic trust-based models can be built to minimise the effect of the contribution of noisy measurements. Specifically, they 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 (HGP) 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 that work focuses neither on the time domain, nor on coordinating measurements taken. It rather 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. Moreover, malicious users will not provide their true belief about their precision.

At the same time, existing approaches address the coordination issue only partially. In particular, applications involving participatory sensing assume that people are generally trustworthy (Krause et al. 2008) and discuss optimal policies for the online integration of sensor information in participatory sensing applied to traffic monitoring data. They propose a greedy algorithm to optimise a spatio-temporal entropy-based criterion but they do not focus on the temporal dynamics of the phenomenon. Other work, (Chen et al. 2014; 2015) coordinates participants for task allocation. Specifically, the aim of their work is to assign tasks to humans based on their mobility patterns to maximise the payoff of tasks in a given time period. However, there is no limit of how many tasks people can do and the tasks are completely independent from each other. Once a task is executed, it is no longer available. On the other hand, in environmental monitoring, measurements are dependent on each other and since the phenomenon is dynamic, there is a need to revisit locations to take more measurements. More relevant work focuses on coordinating measurements for environmental monitoring (Zenonos, Stein, and Jennings 2015a; 2015b) but all of the cases above assume that no malicious users are present. In other words, they ignore the impact of malicious measurements in the areas of interest, which might lead to a false representation of the phenomenon.

Against this background, we present a novel trust-based coordination system, which is able to effectively coordinate measurements in the presence of malicious user behaviour. In particular, extending the work of (Venanzi, Rogers, and Jennings 2013) and (Zenonos, Stein, and Jennings 2015b), we develop an algorithm that uses real-time information to coordinate measurements in order to maximise the information learnt about the environment over time taking into consideration potentially malicious users participating in the campaign. 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. We show that our method is more accurate than other standard GP coordination algorithms. In more detail, our contributions in this paper are:

- We present the first coordination algorithm in participatory sensing that performs effectively in the presence of malicious users.
- We empirically evaluate our algorithm on real human mobility and air quality sensor data and show that it significantly outperforms the state of the art in scenarios where a varying number of malicious users are present.

In the next section we formally define the coordination problem. Next, we present how we model the environment followed by our proposed algorithm. Then, we evaluate our algorithm in different scenarios against the state of the art. Finally, we conclude and suggest possible avenues for future work.

## The Coordination of Measurements Problem

This section formally introduces the problem of coordinating measurements in participatory sensing for environmental monitoring subject to budget constraints and unreliability of users concerning taking an action when requested to do so. Then we extend the problem to capture the presence of malicious users.

### Basic Coordination Problem

First of all, an environmental campaign is initiated to collect as much information about a particular phenomenon in an environment. An environment $\mathcal{E}$ is a continuous set of spatio-temporal locations ($L = \{l_1, l_2, \ldots, l_n\}, T = \{1, 2, \ldots, E\}$) 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 **A**= $\{\mathcal{A}_1, \ldots, \mathcal{A}_M\}$ can take a set of discrete measurements [1]($O$) within the spatial boundaries of this environment within the time period of the campaign ($O = L \times T$). The set of observations made before or at time $j$ is denoted as $\mathbf{O}_j \subseteq O$, while the set of observations made at time $j$ is denoted as $O_j \subseteq \mathbf{O}_j$.

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 used in (Guestrin, Krause, and Singh 2005). Intuitively,

[1] Observations and measurements are used interchangeably.

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 the individual is willing to take within a day. Chon et al., 2013 show that 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. We represent the budget of all users with $B = \{B_1, B_2, \ldots, B_M\}$.

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 that represents their reliability. 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 $R = \{r(\mathcal{A}_1), r(\mathcal{A}_2), \ldots, r(\mathcal{A}_M)\}$.

We denote by $\mathbb{U}$ the total utility earned by all the agents. Thus,

$$\mathbb{U}(\mathbf{O}_E) = \sum_{j=1}^{E} u(O_j) \tag{1}$$

where $u(O_j)$ considers the observations made by participants at the selected locations and timesteps. 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.

### Coordination in the Presence of Malicious Users

As argued, users participating in the participatory sensing campaigns can be malicious for their own reasons. 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 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 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 value that is independent of other users, which is not known in advance. We characterise all users, in terms of maliciousness with $\mathcal{M} = \{m(\mathcal{A}_1), m(\mathcal{A}_2), \ldots, m(\mathcal{A}_M)\}$.

## The Environment Model

In this section, we present how we modelled the environment. We first discretised the environment in a way such that a two-dimensional grid is created over space and the time is divided to hourly measurements (timesteps). Consequently, we say that locations $\mathcal{L} \subset L$ are the intersections of the grid and $\mathcal{T} \subset T$ each timestep. In our work, we convert longitude and latitude into UTM format, i.e., meters, so as to be able to make calculations in Euclidean space.

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

The measurements of an environmental phenomenon can have a multivariate Gaussian joint distribution over all of their locations $\mathcal{L}$ and timesteps $\mathcal{T}$. This is an effective way to capture the spatio-temporal relationship of different coordinates. In particular, we use Gaussian Processes (GPs) that can generalise the multivariate Gaussians to an infinite number of random variables and thus, generalise over the entire set of locations and timesteps (Rasmussen and Williams 2006) and which has been used in related work (Zenonos, Stein, and Jennings 2015a). The main advantages of GPs in environmental monitoring 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.

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:

$$\begin{aligned} m(\mathbf{x}) \quad &= \mathbb{E}[f(\mathbf{x})], \\ k(\mathbf{x},\mathbf{x}') &= \mathbb{E}[(f(\mathbf{x}) - m(\mathbf{x}))(f(\mathbf{x}') - m(\mathbf{x}'))] \end{aligned} \tag{2}$$

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

$$f(\mathbf{x}) \sim \mathcal{GP}(m(\mathbf{x}), k(\mathbf{x},\mathbf{x}')) \tag{3}$$

The kernel $k$ has 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. The kernel typically has free parameters (hyperparameters) which control the smoothness of the function, as well as its sensitivity to measurements and noise. 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. This is expressed as:

$$\begin{bmatrix} \mathbf{y} \\ y_* \end{bmatrix} \sim \mathcal{N}\left(0, \begin{bmatrix} K(X,X) & K(X,X_*) \\ K(X_*,X) & K(X_*,X_*) \end{bmatrix}\right) \tag{4}$$

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 $K(X_*,X_*) = K_{**}$.

Gaussian processes provide the mathematics of the utility function we need to maximise. Similar to Guestrin, Krause, and Singh (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:

$$I(X_y; X_A) = H(X_y) - H(X_y|X_A) \tag{5}$$

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:

$$\begin{aligned} H(X_y|X_A) &= \frac{1}{2}\log(2\pi e\sigma^2_{X_y|X_A}) \\ H(X_y|X_A) &= \frac{1}{2}\log(\sigma^2_{X_y|X_A}) + \frac{1}{2}(\log(2\pi)+1) \end{aligned} \tag{6}$$

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

Moreover, in order to deal with the potentially malicious users we implement a trust-based heterskodetastic GP (HGP). A key feature of the HGP model is that it allows variable noise across input. This varying noise feature, commonly referred to as heteroskedasticity, is relevant to our participatory sensing settings where data are typically provided by devices with individual noise levels (i.e. the different level of accuracy). As in (Venanzi, Rogers, and Jennings 2013), we model the trustworthiness of each user as a hyperparameter of the HGP that scales variance ($\sigma^2$). While in (Venanzi, Rogers, and Jennings 2013), however, precision is a user input, in our work, we take the variance of the measurements taken at each timestep, so that users have no explicit control of the accuracy of the measurements. This is realistic since measurements are taken by mobile sensors that users should have no direct control of. Also, unlike (Venanzi, Rogers, and Jennings 2013), we consider measurements taken both over space and time.

Extending standard GPs, the trust-based HGP model has a separate independent diagonal noise element described by $\Sigma = \text{diag}(\hat{\theta}_1, \ldots, \hat{\theta}_n)$, where $\hat{\theta}_i = \frac{t_i}{\sigma^2}$. In particular, each $t_i$ value is a hyperparameter of the covariance function/kernel, such that $t_i \in [0,1]$ and $\sigma^2$ is the average variance of the measurements. In other words, each user's measurement is associated with a trust value, 1 meaning the user's measurement is fully trustworthy while 0 is not. We use a trust-based uncertainty scaling technique based on adding extra uncertainty to individual data points, depending on how much such points are trustworthy. By doing so, the model is able to allow larger variance around untrustworthy points, whilst still modelling correlations in the locality of such points. This produces the effect of increasing the uncertainty in users' reports up to turning them into completely uninformative contributions when it is close to zero. Consequently, the hyperparameters to be learned are: $\hat{\boldsymbol{\Theta}} = \{l_1, l_2, l_3, \sigma_f^2, t_1, \ldots, t_n\}$. Estimating $\theta$ is equivalent to finding a value for $\hat{\boldsymbol{\Theta}}$ that results in a high $p(\mathbf{X}, \mathbf{y}|\hat{\boldsymbol{\Theta}})$. In practice, it is achieved by maximising the log marginal likelihood $\log p(\mathbf{X}, \mathbf{y}|\hat{\boldsymbol{\Theta}})$:

$$\Theta_{ML} = \arg\max_{\Theta} p(\mathbf{X}, \mathbf{y}|\hat{\boldsymbol{\Theta}}) \tag{7}$$

This is given by:

$$\log p(\mathbf{X}, \mathbf{y}|\hat{\boldsymbol{\Theta}}) = -\frac{1}{2}\mathbf{y}^T C^{-1}\mathbf{y} - \frac{1}{2}\log|C| - \frac{n}{2}\log 2\pi \tag{8}$$

where $C$ is the kernel $K$ (as in standard GP) with added noise $\Sigma$.

## The Coordination Algorithm

In this section, we firstly give a high level overview of the proposed algorithm and then describe it in more details. Our algorithm extends the adaptive Best Match ('aBM') algorithm proposed in (Zenonos, Stein, and Jennings 2016) in order to consider the effect of potentially malicious users and the trust-based HGP model described in (Venanzi, Rogers, and Jennings 2013) by alleviating the requirement for manual user input of their estimated precision. Specifically, our algorithm estimates the users' trustworthiness in real time by applying the MLE technique at each timestep[2]. In particular, the $t$ value is estimated for all participants that took a measurement at a specific timestep. 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. Active participants are those who are selected to take a measurement by the coordination system in (Zenonos, Stein, and Jennings 2016).

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 has two major

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

additions compared to aBM (Zenonos, Stein, and Jennings 2016). 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. Thus, the contribution of less trusted users has a lower impact on estimating the state of the environment. The other component is called SWAP and it is responsible for swapping malicious or low-trust users with more trustworthy nearby users in real-time. As a result, people with lower trust values are not chosen to take more measurements.

## TaBM Algorithm

In more detail, the TaBM algorithm requires the number of timesteps of the PS campaign, the budget of each participant, its reliability and the hyperparameters of the model (line 1). Next, simulations run offline as in (Zenonos, Stein, and Jennings 2016) 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 also explained in more detail in (Zenonos, Stein, and Jennings 2016). 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). 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 sub-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).

## SWAP Function

The SWAP algorithm is responsible for removing malicious users from the set of selected users that are required to take measurement at any given timestep and substituting them with nearby high-trust ones. In more detail, this algorithm requires the details of the particular user currently examined, as explained in the algorithm above, the details of all other agents and the threshold calculated in the algorithm above (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 of the malicious one (line 5). Once, the nearest neighbour is found, it is checked that it satisfies certain properties (line 6). Specifically, the user should have some budget left and a trust value. 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 not measurement is found the user is not swapped but their measurement has a low impact on the overall prediction of the phenomenon.

**Algorithm 1** Trust-aware adaptive Best-Match (TaBM) Algorithm

1: **input**: $E$ (timesteps), $B$ (budget), $R$ (reliability) $\hat{\mathbf{\Theta}}$ (hyperparameters) $\mathbf{A}$ (agents)
2: $S_{1,\ldots,N}, C_{1,\ldots,N}$ $\leftarrow$SISCAS$(E, B, R)$ {Simulations running offline (Zenonos, Stein, and Jennings 2016)}
3: $t = zeros$
4: **for** $j = 1$ to $E$ **do**
5: $S_j^* \leftarrow$ MATCHING$(E, j, B, S_{1,\ldots,N}, C_{1,\ldots,N})$ {Online mapping of users to measurements (Zenonos, Stein, and Jennings 2016)}
6: $average\_trust = \frac{1}{|S^*|}\sum_{s=1}^{|S^*|} t_s$
7: $sem = \frac{std(t)}{|S^*|} \cdot 1.96$ {standard error mean for 95% confidence level}
8: $threshold = average\_trust - sem$
9: **for** $i = 1$ to $|S_j^*|$ **do**
10: **if** $t_i < threshold$ **then**
11: SWAP($S_i^*$,$\mathbf{A}$,$threshold$)
12: **else**
13: Take measurement
14: **end if**
15: **end for**
16: $\Theta_{ML} = \arg\max_{\Theta} p(S_j^*, \mathbf{y}|\hat{\mathbf{\Theta}})$ {$\mathbf{y}$ is the actual measurements taken by people in $S_j^*$}
17: **end for**

**Algorithm 2** SWAP Function

1: **input**: $\mathcal{A}$ (agent), $\mathbf{A}$ (agents), $threshold$ (trust value)
2: $evaluated = \emptyset$
3: **while** $|evaluated| < |\mathbf{A}|$ **do**
4: $\mathbf{A^*} \leftarrow remove(\mathbf{A}, evaluated)$
5: $\mathcal{A}^N \leftarrow nearestneighbour(\mathcal{A}, \mathbf{A^*})$
6: **if** $\mathcal{A}^N = \emptyset$ or $t_{\mathcal{A}^N} = \emptyset$ or $B_{\mathcal{A}^N} = 0$ **then**
7: Return
8: **end if**
9: Append $\mathcal{A}^N$ to $evaluated$
10: **if** $t_{\mathcal{A}^N} > threshold$ **then**
11: Substitute $\mathcal{A}$ with $\mathcal{A}^N$
12: **end if**
13: **end while**

# 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.

## Benchmarks

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

- **Greedy:** This algorithm is based on (Krause, Singh, and Guestrin 2008). 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 available by the number of timesteps left.
- **adaptive Best-Match:** This algorithm is presented in (Zenonos, Stein, and Jennings 2016). It consists of two main components (offline and online). The former is responsible for searching through the space of candidate solutions to produce a number of mappings of participants to spatio-temporal locations. It is adaptive in the sense that it considers the reliability of users to select which one within in a particular cluster should take a measurement, in order to maximise the expected utility while at the same time saving budget for future iterations. The online component handles the real-time situation, finding the best match between the simulation output from the offline algorithm and the real-time situation. This is similar to our algorithm but it does not have the trust capabilities.
- **Best-Match:** This is an algorithm presented in (Zenonos, Stein, and Jennings 2015b). The Best-Match algorithm works similarly to adaptive Best-Match. 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 other words, this algorithm does not take in consideration the reliability of users nor whether they are malicious.

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 0 for malicious and 1 for trustworthy users.

## Experimental Setup

To empirically evaluate our algorithm, we compare its performance against the algorithms described above in terms of Root mean Squared Error (RMSE) defined below:

$$RMSE = \sqrt{\frac{1}{|L|}\sum_{l=1}^{|L|}(y_l - y_l^*)^2} \tag{9}$$

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. In our settings it is interesting to use this metric to capture the effect/increase on RMSE when malicious measurements are taken over time. In this work, 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. We use an air quality dataset (Zheng, Liu, and Hsieh 2013), 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 and the mean values over the spatiotemporal locations of interest. 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.

Air quality exhibits spatial variations (PM2.5 is different depending on where you are in Beijing) as well as temporal variations (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, however, we use data from the Geolife trajectories dataset (Zheng et al. 2009; 2008; Zheng, Xie, and Ma 2010), which contains sequences of time-stamped locations of 182 people in Beijing over a period of 5 years (2007-2012), as reported by portable GPS devices. 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. This is used as the ground truth to compute the upper bound in our experiments. In particular, we experiment with up to 1000 users per timestep. 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 as in (Zenonos, Stein, and Jennings 2016).

Also, 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). Moreover, we cannot assume that participants will always be willing or able to take a measurement when requested, even if they initially agreed on contributing a number of measurements. 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. Furthermore, we vary maliciousness between $0.1 - 1$ for the experiments shown in Figure 1 and is fixed

to 0.25 for Figure 2 and Figure 3 as this is shown to be a typical prevalence of malicious users in the crowdsourcing domain (Gadiraju et al. 2015).

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, Singh, and Guestrin 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, 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.

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

## Results

Figure 1 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.

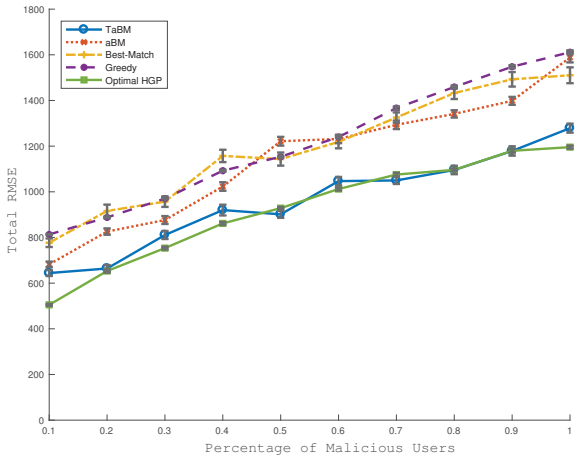


Figure 1: Total RMSE over space and time with a varying percentage of malicious users. The error bars indicate the 95% confidence interval.

Figure 2 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.

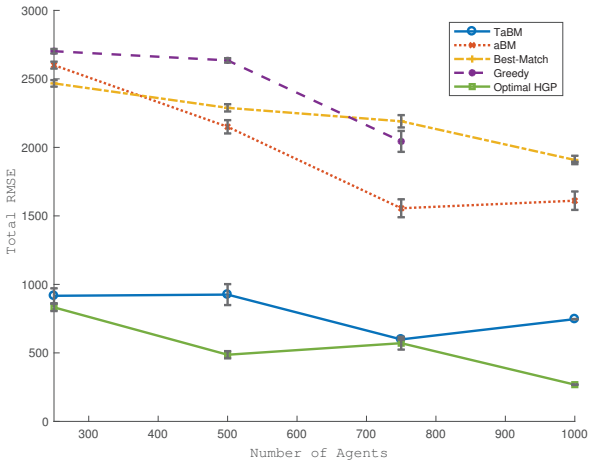


Figure 2: Total RMSE over space and time with a varying number of users. The error bars indicate the 95% confidence interval.

Figure 3 shows that aBM, Best-Match and the Optimal HGP has similar runtime. However, the TaBM algorithm is more computational expensive than these algorithms but with the significant trade-off in performance as discussed above. The Greedy algorithm has significantly higher computational cost compare 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. In particular, we were not able to calculate the total RMSE and runtime for 1000 participants as it was very computationally intense to do so.

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.

## Conclusions and Future Work

We introduced the problem of coordinating measurements in participatory sensing settings in the presence of malicious users. This is the first approach in dealing with malicious users in participatory sensing coordination algorithm. 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. 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, (b) Trust-based adaptive

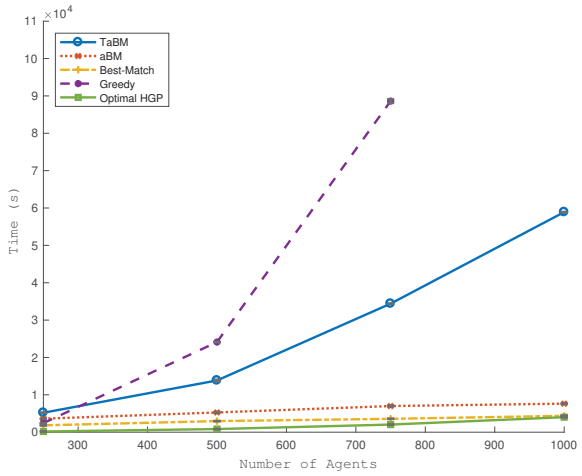


Figure 3: Average runtime for 24 timesteps and a varying number of users. The error bars indicate the 95% confidence interval.

Best-Match is significantly faster than the Greedy approach and comparable to the adaptive Best-Match and Best-Match.

There are a number of potential avenues for the future. In particular, 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.

## References


Bachrach, Y.; Graepel, T.; Kasneci, G.; Kosinski, M.; and Van Gael, J. 2012. Crowd iq: Aggregating opinions to boost performance. In *Proceedings of the 11th International Conference on Autonomous Agents and Multiagent Systems - Volume 1*, AAMAS '12, 535–542. Richland, SC: International Foundation for Autonomous Agents and Multiagent Systems.

Brown, A.; Franken, P.; Bonner, S.; Dolezal, N.; and Moross, J. 2016. Safecast: successful citizen-science for radiation measurement and communication after fukushima. *Journal of Radiological Protection* 36(2):S82.

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

Chen, C.; Cheng, S.-F.; Lau, H. C.; and Misra, A. 2015. Towards city-scale mobile crowdsourcing: Task recommendations under trajectory uncertainties. In *Proceedings of the 24th International Conference on Artificial Intelligence*, IJCAI'15, 1113–1119. AAAI Press.

Chon, Y.; Lane, N. D.; Kim, Y.; Zhao, F.; and Cha, H. 2013. 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*.

Gadiraju, U.; Kawase, R.; Dietze, S.; and Demartini, G. 2015. 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, 1631–1640. New York, NY, USA: ACM.

Guestrin, C.; Krause, A.; and Singh, A. P. 2005. Near-optimal sensor placements in gaussian processes. In *Proceedings of the 22Nd International Conference on Machine Learning*, ICML '05, 265–272. New York, NY, USA: ACM.

Irshad, H.; Oh, E.-Y.; Schmolze, D.; Quintana, L. M.; Collins, L.; Tamimi, R. M.; and Beck, A. H. 2017. Crowdsourcing scoring of immunohistochemistry images: Evaluating performance of the crowd and an automated computational method. *Scientific Reports* 7.

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

Krause, A.; Horvitz, E.; Kansal, A.; and Zhao, F. 2008. Toward community sensing. In *2008 International Conference on Information Processing in Sensor Networks (ipsn 2008)*, 481–492.

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

Mousa, H.; Ben, S.; Hasan, O.; Younes, O.; Hadhoud, M.; and Brunie, L. 2015. Trust management and reputation systems in mobile participatory sensing applications : A survey. *Computer Networks* 90:49–73.

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

Reddy, S.; Shilton, K.; Burke, J.; Estrin, D.; Hansen, M.; and Srivastava, M. 2008. Evaluating Participation and Performance in Participatory Sensing. *Proceedings of the International Workshop on Urban Community and Social Applications of Networked Sensing Systems UrbanSense08* 4–8.

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

Sahami Shirazi, A.; Henze, N.; Dingler, T.; Pielot, M.; Weber, D.; and Schmidt, A. 2014. Large-scale assessment of mobile notifications. *Proceedings of the 32nd annual ACM conference on Human factors in computing systems - CHI '14* 3055–3064.

Stevens, M., and D'Hondt, E. 2010. Crowdsourcing of Pollution Data using Smartphones. In Vukovic, M.; Kumara, S.; and Greenshpan, O., eds., *Workshop on Ubiquitous Crowdsourcing, held at Ubicomp '10 (September 26-29, 2010, Copenhagen, Denmark)*.

Venanzi, M.; Rogers, A.; and Jennings, N. R. 2013. Crowdsourcing spatial phenomena using trust-based heteroskedastic gaussian processes. In *First Conference on Human Com-*

*putation and Crowdsourcing (HCOMP)*, 182–189. AAAI Press.

Zenonos, A.; Stein, S.; and Jennings, N. R. 2015a. Coordinating measurements for air pollution monitoring in participatory sensing settings. *14th Int. Conference on Autonomous Agents and Multi-Agent Systems*.

Zenonos, A.; Stein, S.; and Jennings, N. R. 2015b. 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)*, 3936–3942.

Zenonos, A.; Stein, S.; and Jennings, N. 2016. Coordinating measurements in uncertain participatory sensing settings. Technical report, ECS Department, University of Southampton.

Zheng, Y.; Li, Q.; Chen, Y.; Xie, X.; and Ma, W.-Y. 2008. Understanding mobility based on gps data. In *Proceedings of the 10th International Conference on Ubiquitous Computing*, UbiComp '08, 312–321. New York, NY, USA: ACM.

Zheng, Y.; Zhang, L.; Xie, X.; and Ma, W.-Y. 2009. Mining interesting locations and travel sequences from gps trajectories. In *Proceedings of the 18th International Conference on World Wide Web*, WWW '09, 791–800. New York, NY, USA: ACM.

Zheng, Y.; Liu, F.; and Hsieh, H.-P. 2013. 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, 1436–1444. New York, NY, USA: ACM.

Zheng, Y.; Xie, X.; and Ma, W.-Y. 2010. Geolife: A collaborative social networking service among user, location and trajectory. *IEEE Data Eng. Bull.* 33(2):32–39.