

Research Article

Elastic Information Management for Air Pollution Monitoring in Large-Scale M2M Sensor Networks

Yajie Ma,¹ Yike Guo,² Dilshan Silva,² Orestis Tsinalis,² and Chao Wu²

¹ College of Information Science and Engineering, Wuhan University of Science and Technology, Wuhan 430081, China

² Department of Computing, Imperial College London, London SW7 2BW, UK

Correspondence should be addressed to Yajie Ma; mayajie@wust.edu.cn

Received 20 August 2013; Accepted 24 October 2013

Academic Editor: Jianhua He

Copyright © 2013 Yajie Ma et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

In large-scale machine-to-machine sensor networks, the applications such as urban air pollution monitoring require information management over widely distributed sensors under restricted power, processing, storage, and communication resources. The continual increases in size, data generating rates, and connectivity of sensor networks present significant scale and complexity challenges. Traditional schemes of information management are no longer applicable in such a scenario. Hence, an elastic resource allocation strategy is introduced which is a novel management technique based on elastic computing. With the discussion of the challenges of implementing real-time and high-performance information management in an elastic manner, an air pollution monitoring system, called EIMAP, was designed with a four-layer hierarchical structure. The core technique of EIMAP is the elastic resource provision scheduler, which models the constraint satisfaction problem by minimizing the use of resources for collecting information for a defined quality threshold. Simulation results show that the EIMAP system has high performance in resource provision and scalability. The experiment of pollution cloud dispersion tracking presents a case study of the system implementation.

1. Introduction

Recently, an increasing amount of research interest has been drawn towards data management in large-scale machine-to-machine (M2M) sensor networks [1–3], where a large number of high-throughput autonomous sensor nodes communicate directly with each other without human intervention and can be distributed over wide areas. M2M sensor networks have found their applications ranging from home monitoring to industrial sensing, including environment and habitat monitoring, traffic control, and health care. Such networks are usually characterised by a large number of sensors, wide coverage areas, a huge amount of data, complicated connectivity, and increasingly stringent response-time requirements. Their applications normally require data management over widely distributed sensors under restricted power, processing, storage, and communication resources. The continual increases in size, data rates, and connectivity of sensor networks present significant scale and complexity challenges. This is especially true when the computational resources available are limited. Thus, efficient support from sensor data

management for data acquisition, transmission, storage, and retrieval becomes critical [4].

1.1. Motivation. Current research on information management for sensor networks has increasingly focused on real-time sensor data collection and sharing of computational and storage resources for sensor data processing and management. Technologies that support the building of large-scale infrastructures, integrating heterogeneous sensors, data, and computational resources deployed over a wide area accelerate the integration of sensor networks with distributed computing, grid computing, and even the state-of-the-art cloud computing. Recent researches on large-scale M2M sensor networks bring about some instructive designs of information management frameworks. For example, our former work in [5] designed a grid-based sensor information management platform. It obtains a high resolution of pollution characteristics in urban environment by high-density distributed sensors. Research in [6] investigates a feedback-based model-driven push approach to support user queries.

It presents a two-tier sensor architecture which develops sensor proxies at higher tier that each proxy controls tens of sensors at lower layer. In order to support energy-efficient query processing, sensors only transmit deviations to proxies compared with model-predicted values, which makes the system depend highly on the model calculated from past observations. Methods of parallel processing of data in sensor networks are discussed in [3], where the authors investigated how much degree existing distributed database solutions and programming models (e.g., MapReduce) are suitable for large-scale sensor network data processing. Based on the analysis, a general architecture for different data processing applications is developed. A similar data processing approach is proposed in [7]. The idea is to use distributed databases to store sensory data and MapReduce programming model for large-scale sensory data parallel processing. An interesting and useful implementation in this approach is that it employs cloud-based storage and computing infrastructure, which is remarkable for the research and design in this paper.

However, as the amount of information monitored by an M2M sensor network increases, two key issues arise in this context that cannot be addressed by existing approaches.

1.1.1. Effectively Avoiding Information Overload. This is done by organizing information collection and processing to focus on analysing only information relevant to the user needs. This includes deciding what information each of the sensor units should be collecting, at what rates, how and where it is processed, summarized, and stored, what information should be exchanged between the sensors, and how all such information should flow within the network. Tradeoffs between whether only local information collected from one sensor or global information collected from all sensors arise when addressing these decisions. This kind of on-demand processing has to eliminate overprovisioning when used with utility pricing. It is also expected to be able to remove the need to overprovision in order to meet the demands of millions of users.

1.1.2. Efficiently Maximizing the Value of Collected Information under Resource and Real-Time Constraints. A finite number of sensor units exist, and each has finite processing capacity, memory, and storage size, communication bandwidth, and battery power available to it. Tradeoffs occur since allocating a group of sensors to explore a particular geographic region would mean fewer resources available for exploring other areas. Similarly, within one region, assigning more processing or memory capacity to explore the features of a particular event in detail would mean less capacity available to explore other events. However, for real-time applications, the underlying algorithms require that the services quality improvement be monotonic to the consumption of the resource needed. In this case, how to identify the critical information needed and intelligently allocate resources to get it are a key point that deserves further research.

Based on the consideration of these issues, a vehicle/person-mounted air pollution monitoring system EIMAP (the acronym for “Elastic Information Management for Air

Pollution”) is proposed in this paper. This system has a four-layer architecture which can contain thousands of sensors distributed over an entire urban area to monitor airborne pollutants including SO₂, NO, NO₂, benzene, and ozone. The data volume that needs to be processed varies from several bytes (individual readings per sensor per minute that are used to identify irregularities and anomalies in real time) to 8 GB (whole readings per sensor per day that are used to capture high-resolution urban air pollution distribution resulting from transportation down to the single building level). In order to provide flexible resource for such large volume-variant data flows, an elastic resource allocation mechanism was introduced to EIMAP system. The key difference between our system and existing approaches is that EIMAP is endowed with data-aware QoS-driven capability for sensor management. Whether a sensor is active or not does not only depend on the energy-efficient consideration, but also, and more importantly, rely on the environment that the sensor resides, and in how much degree it is required to provide resource to the task.

1.2. Research Contributions. Our design of EIMAP has led to the following main contributions.

(1) Introducing Elasticity to Large-Scale M2M Sensor Information Management. Elasticity captures a fundamental aspect of cloud computing: when limited resources are offered for potentially unlimited use, providers must manage them elastically by scaling up and down as needed [8]. Elastic information management (EIM) is a technique that is based on elastic computing (EC), which is a feature of cloud computing. In [9], EC is defined as the use of computer resources which vary dynamically to meet a variable workload. The mathematical definition of elasticity in economics is

$$E_{y,x} = \left| \frac{\partial \ln y}{\partial \ln x} \right| = \left| \frac{\partial y}{\partial x} \cdot \frac{x}{y} \right|, \quad (1)$$

where $E_{y,x}$ denotes the elasticity of y with respect to x ; $\partial y/\partial x$ is the derivative of y with respect to x . This formula reveals the ratio of the percent change in one variable to the percent change in another variable [10]. Based on this concept, we designed a four-layer architecture for M2M sensor network information management. In comparison with other architectures, the novelty is that a special layer, elastic management layer, is embedded which provides a scalable, resource-aware infrastructure for processing environmental streaming data produced by a range of heterogeneous mobile sensors.

(2) Developing a Scheduling Algorithm for Real-Time Resource Allocation. This algorithm overcomes the disadvantage of fixed resource provision strategy, which is not adaptive in changing environments. It also takes into account both of the resource provision and environmental feature detection by modeling it as a constraint satisfaction problem of minimizing the use of resources in a sensor network for collecting monitoring information at a defined quality threshold. The experimental results show that this algorithm performs well in elastic resource allocation.

1.3. Paper Layout. The remainder of this paper is organized as follows. In Section 2, we discuss the related work in the areas of information management in M2M sensor networks and resource allocation strategies for information management. Section 3 addresses the challenges by introducing EIMAP which comprises a four-layer hierarchical information management architecture. In Section 4, the design of the scheduler for elastic management is presented with the pseudocode for each part of the scheduling algorithm. Section 5 analyses the performance of the scheduling algorithm and simulates the EIMAP system in the WikiSensing sensor data management platform to evaluate the capability of the concurrent streaming management. Section 6 presents a case study of air pollution monitoring in East London. Section 7 concludes the paper with a summary of the research and a discussion of future work.

2. Related Work

2.1. Information Management in M2M Sensor Networks. Information management for sensor networks has been drawn much research attention for decade [11–13]. In a large-scale sensor network for air pollution monitoring, although one would often be more interested in highly polluted areas in environmental monitoring, a quick response to a pollution event related to other areas or time snapshots would usually be highly desirable. Information management for such an application will focus more on defining the events, or the feature of interests, which involves the techniques of data representation, summarization, and organization. To do so, techniques of improving the performance of query under resource constraints have been developed in recent years. For example, several approaches have focused on adaptive sampling techniques which aim to restrict, in an intelligent way, the amount of information gathered within the network. A popular approach is based on using Kalman filters [14] which enable a group of sensors to respond to and track fast moving signals in a slowly changing background. For the QoS management, the Aurora system [11] developed by Brown University and the TinyDB [15, 16] system developed by MIT all provide QoS managing strategies to support reliable services.

Other systems have focused on efficient data summarisation to facilitate query propagation/processing and to improve distributed data storage within the networks. For instance, the BBQ system [17] maintains a correlation-aware probabilistic model in a base station to provide a robust interpretation of sensor readings. Data acquisition from sensors happens only when the model is not able to offer approximate answers to certain queries with acceptable confidences. StonesDB [18] applies a multiresolution scheme, generating two summary streams (wavelet-based summary and subsampled summary streams) from input data streams. In [19], the authors addressed the problem of the information-driven management criteria for sensor networks. They proposed a novel measurement for usefulness of information: uncertainty reduction rather than information

gain is used to justify the performance of the information-driven performance. Other techniques, such as synopsis-based approximate answers [20], histogram analysis [21], and wavelet-based data summaries [22, 23], can all be used to investigate how to guarantee high accuracy and speed of data summarisation.

2.2. Resource Allocation for Information Management. Resource allocation is a key technique for information management. Many attentions have been paid to this area in recent years. The research generally has two categories: one is to develop novel system infrastructures to meet different resource allocation demands; another is to design high performance algorithms to support fast resource allocation computation.

For the first category, [24] investigated the challenges of resource allocation for reconfigurable multisensor networks. The authors discussed the problems of resource allocation under environmental and technical constraints. A hierarchical model was proposed in this paper but not concrete resource allocation strategy was presented. For efficient resource allocation, hierarchical or layered system architectures can be studied in many papers. For example, in [25], the researchers designed a layered distributed system and extended the existing disjoint coalition formation protocol to solve the multi-sensor task allocation problem which aims to automatically decide the best sensors to specified task; in [26], a market-based 4-layer architecture was presented for adaptive task allocation which formulises the pricing mechanism to achieve a fair energy balance among sensor nodes while minimizing delay; [27] designed a two-tiered on-demand resource allocation strategy which is specially designed for VM-based computing environments.

For the second category, most of the algorithms treat the resource allocation problem as an optimisation. They usually handle tradeoffs between system performance and resource constraints. [28] discussed two tradeoffs in health monitoring sensor system. The authors addressed two optimisation problems in this paper, one of which is to obtain sustainable power supply, while another one is to achieve high quality of service. The solutions to two optimisation formulas were given as well. Groot et al. analysed an adaptive optimisation for the tradeoff between resource allocation and the reconfigurable resources within the multi-sensor network in [29]. The resource allocation algorithm aims to maximize the system utility by finding the optimal set of services. [30] considers sensor assignment problems in both static and dynamic sensing environments. Heuristic algorithms were studied to address the NP-hard optimisation problems. Other schemes for the category can also be seen, such as agent-based algorithm [31] and posterior-based decision making scheme [32].

3. Elastic Sensor Network Architecture

The key feature of the EIMAP architecture is using autonomous sensors, whether fixed or mobile, to provide coverage of a specific geographical area to collect real-time

pollution data on key aspects such as traffic conditions, vehicle emissions, ambient pollutant concentration, and human exposure. The focus of constructing an EIMAP system is related to the data management, computation management, information management, and knowledge discovery management associated with the sensors and the data they generate, and how they can be addressed in real-time within an open computing environment. To do so, in this section, we first analyse the challenges in implementing such a system, and then we propose a four-layer architecture to address these challenges.

3.1. Challenges in Implementing EIMAP System. Considering the resource characteristics of large-scale M2M sensor networks, the main issues and challenges related to constructing an elastic system are as follows.

3.1.1. Dynamic Interactivity via M2M Architecture. Within a mobile sensor network, the sensors themselves naturally form an M2M network and communicate with each other through it. In order to satisfy the real-time analysis requirements, the sensors themselves will have to store part of the information and communicate with each other within the M2M network. The measurements from the sensors, both mobile and static, will be filtered and processed using a set of specialized algorithmic processes, before being warehoused in a repository. The design and implementation of a suitable M2M sensor architecture will need to satisfy the real-time analysis requirements as well as decide the data storage/communication tradeoffs. The sensors in such a system will need to be equipped with sufficient computational capabilities to participate in the elastic environment and to feed data to the warehouse as well as perform analysis tasks and communicate with their peers.

3.1.2. Elastic Resource Allocation under Resource Constraints. In such a scenario, strategies of allocating or scheduling finite sensing resources for exploring surveillance regions in more detail have to be proposed. One also has to take into consideration the dynamic changes that occur in the sensed environments. We model this scheduling problem as a constraint satisfaction problem for selecting a particular resource allocation strategy for maximizing the value of information collected at any time step. Such resource allocation needs to take into account constraints on the resources, the decision-making time (e.g., the value of information may diminish if its transmission is delayed), and other problem-dependent constraints (e.g., a need to keep full coverage of a particular area or particular events using a minimum number of sensors). Hence, the strategies of allocating or scheduling have to be able to (a) define the resource and application constraints together with the associated solvers and (b) estimate the increased information (information gain) every time step for the different strategies through the selection of the appropriate measures in terms for its completeness, quality, and reliability.

3.2. EIMAP Hierarchical Architecture. Considering the challenges analysed above, we introduce the elastic computing

capability of EIMAP, which aims to provide a reliable, scalable infrastructure for elastic management of streams of environmental data produced by a range of heterogeneous mobile sensors. Therefore, a four-layer architecture was designed as shown in Figure 1. This architecture is also well suited to the dynamic, on-demand, pay-per-use nature of the emerging utility computing platforms.

3.2.1. Sensor Layer. This layer manages all the raw hardware level resources in the system, such as the environmental characteristics, different types of sensors, network connection, storage, sensing activities, and distributed raw data. Sensors within the environment are heterogeneous and may be mobile or static. Hence, the wireless connectivity can provide different access protocols to the IP backbone including WiFi (802.11.g), ZigBee (802.15.4), and WiMAX (802.16). The sensors have the capability to sample one or more pollutants or other environmental properties such as noise or temperature. This data will then be transported to the data store in the upper layer (which will be introduced in the following paragraphs). Since potentially the volume of sensory data is significant whereas, the processing resource is limited, the key of the sensing activity is efficiency—salient regions should be paid more attention to and, consequently, consume more processing resource. In this case, an attention-based sensing mechanism [33–36] is preferred, which can extract irregularities and anomalies from massive background noise. To do so, an intelligent control strategy supported by the elastic management from higher layer is necessary.

3.2.2. Elastic Management Layer. This is the core layer of the EIMAP architecture. The purpose of this layer is to provide an elastic resource provision infrastructure for the whole system. It contains resources that have been abstracted/encapsulated so that they can be exposed to the upper layer and the end users as integrated resources, for instance, repositories, resource catalogue services, resource scheduler, and specialized services such as sensor registry/activity management. The resource supply and its supply infrastructure can scale up and down dynamically based on application resource need, which is able to deliver software application environments with a resource usage-based computing model. The resource scheduling service, which is critical for the system performance, is the core service of this layer as well as the whole EIMAP architecture. It enables virtual organization management, resource management, and load balancing in order to guarantee an easy access to sensor data in heterogeneous physical sensors. We will discuss it in next section in detail.

3.2.3. Data Analysis Layer. This layer (whether centralized or distributed) is concerned with information comprehension, including how to summarize the data and how to develop and use models representing the data to control the operation of the sensing activities, such as adjusting sampling rates of specific sensors or making decision of allocating more sensing resources to a particular geographic area to gain further information about it. Centralized and decentralized

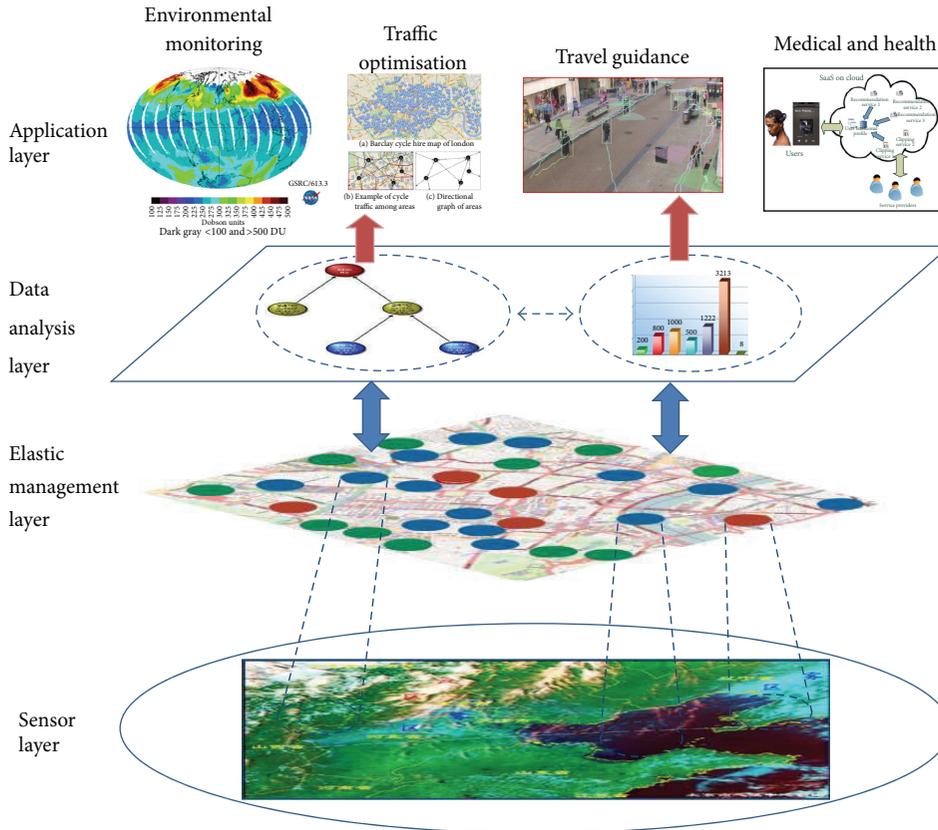


FIGURE 1: EIMAP hierarchical architecture.

data mining algorithms are developed in this layer to meet the needs of different data analysis tasks. The analysis results are delivered to the application layer according to different user requirements.

3.2.4. *Application Layer.* This layer retrieves information from the data analysis layer and uses this information as the input to different applications, not only for the air pollution monitoring. Because the lower layers are designed to be application-independent, the framework is universal for different applications, such as traffic optimisation, security surveillance, mental training, and city planning. Besides, a user-defined service module makes the system extensible so that the users can take advantage of new services that become available.

4. Scheduler for Elastic Sensing

Resource allocation is a key issue in EIMAP which affects not only the sensing activities regarding specific events but also the performance of the whole system including speed and accuracy of response, fairness of queries, and experience for users. Suppose such an application scenario: using sensors to track moving objects such as the pollution cloud (due to dispersion, the pollution cloud always moves and changes its shapes/size). A fixed resource provision strategy

is not preferred especially in a resource-restricted environment. Hence, elastic resource provision is a better choice to improve the system performance. In a sensor network, the available underlying resources are sensors, including the sensing behaviours, distributed computational capabilities, and communication links (connectivity, bandwidth, radio power, etc.) that sensors or sensor peers can provide. In consideration of the resource constraints in sensor networks, a resource awareness mechanism is essential to provide a strategy of allocating or scheduling finite sensing resources in exploring potential regions of interest and to take into consideration the dynamic changes that occur in the sensed environment.

A scheduler in the elastic management layer is designed for such an elastic sensing requirement, which aims to model a constraint satisfaction problem of selecting a particular resource allocation strategy for maximizing the value of information collected at any time step, or, minimizing the use of resources in a sensor network for collecting monitoring information at a defined quality threshold.

In order to model this constrained optimisation problem, we feature the surveillance area as follows.

- (1) The entire geographical area is divided into grid units and each grid has a predefined size to cover a reasonable region of the area according to the specific requirements of air monitoring.

TABLE 1: Scheduling algorithm description.

Step	Description
1	Generate a candidate set of nodes
2	Define objective function
3	Identify constraints
4	Find the solution of scheduling

- (2) There is a sensor in the centre of each grid which collects and maintains a series of sensor readings for historical or real-time query.

And according to the physical property of the resource provider, the resource constraints can be classified into two categories:

- (1) hardware resource constraints, including
 - (a) size of monitored area/number of grids,
 - (b) storage capability,
 - (c) surplus energy,
 - (d) communication distance,
 - (e) available bandwidth;
- (2) software resource constrains, including
 - (a) measuring accuracy requirement,
 - (b) pollutant diffusion model,
 - (c) sensory data attributes.

Suppose now we have identified the feature of interest in an area as an m -dimensional vector $A = (a_1, a_2, \dots, a_m)$ (A can be achieved by attention-based mechanisms and the computational detail is out of the scope of this paper). Suppose the available set of nodes in this area is V ($|V|$ is the number of nodes in V). Each node $i \in V$ has a reading $Y_i = (y_1, y_2, \dots, y_m)$ that describes the feature of the node or the grid where the node resides. $a_j \in A$ is an attribute corresponding to an element $y_j \in Y$. The resource constrained scheduling strategy can be described by the steps that are shown in Table 1.

4.1. Generate a Candidate Set of Nodes. The following GC algorithm shown in Algorithm 1 is used to find the candidate nodes for resource provision where the feature of interest A is likely to be detected. The algorithm returns a list of candidates by matching every reading in every node with a given feature.

The algorithm starts with a given number of iterations and a null set of candidate nodes C (line 1). For a given number of sampling times, the Euclidean distance between the mean value of the readings in each node and the given feature is calculated (lines 3 to 5) (the Euclidean distance between two readings P and Q can be calculated as Euclidean $(P, Q) = \sqrt{|p_1 - q_1|^2 + |p_2 - q_2|^2 + \dots + |p_m - q_m|^2}$). If the distance is no larger than a predefined threshold ϵ , then the node providing this reading satisfies the constraint of data similarity and has to be added into C as a candidate (line 6).

4.2. Define the Objective Function. In order to describe whether a candidate node is chosen to be a resource provider or not, we define a decision variable x_i :

$$x_i = \begin{cases} 1, & \text{if candidate node } i \text{ is chosen,} \\ 0, & \text{otherwise.} \end{cases} \quad (2)$$

The scheduler tries to find an optimal set of nodes from the candidate nodes given the resource constraints. In a sensor system, a vital resource constraint is the node energy. An energy-aware system will have better performance in system life time [37–39]. Hence, in our system, we select the surplus energy as the optimisation objective and the aim of the optimisation is to minimize the rate of the energy consumption, which can be formulated as

$$\min \sum_{i=1}^{|C|} w_i x_i, \quad (3)$$

$$w_i = \frac{\text{RE}(i)}{\text{SE}(i)},$$

where $\text{RE}(i)$ is the required energy for node i to execute the current task. $\text{SE}(i)$ is the surplus energy in node i . Then w_i ($w_i > 0$) is a weight to measure what percentage of the surplus energy of the sensor the current task will consume. For a sensor network, small value of w_i will bring better energy performance, which means the nodes with less energy consumption rate are chosen and the lifetime of the network is prolonged.

4.3. Identify Constraints. In air pollution monitoring, considering the pollutant diffusion model, we cannot let a sensor monitor an arbitrary size of area in order to guarantee the measurement accuracy. Furthermore, a single sensor is less possible to provide enough storage and computation capacity for the whole task. Therefore, the scheduler has to find a set of nodes with reasonable number of nodes for resource provision. To simplify the analysis, suppose all sensors have the same storage space to cache data; all the links have the same bandwidth; the communication distance is adequate for data transmitting from one grid to a neighbour grid. And we suppose that all the pollution data analysed in this paper are generated and diffused under the same model. Hence, the hardware/software resource constraints that need to be taken into account are reduced to the number of grids, surplus energy, measuring accuracy, and data attributes. According to the guidance of environmental data collection [40], the minimum number of nodes N in a sampling unit has to satisfy

$$N = \frac{t^2 s^2}{D^2}, \quad N \leq |C|, \quad (4)$$

where t is the critical value for 2-tailed t -test with a specified degree of freedom; s is the standard deviation of the samples; D is the absolute deviation. If the square of a monitoring unit is S , the maximum distance L between two sampling nodes is

$$L = \sqrt{\frac{S}{N}}. \quad (5)$$

```

(1) Given A; NS = NUM.SAMPLES; C = ∅;
(2) for (i = 1 to |V|) {
(3)   for (j = 1 to NS)
(4)      $\bar{Y}_{ij} = \frac{1}{NS} \sum_{j=1}^{NS} Y_{ij}$ 
(5)   if (Eulidean( $\bar{Y}_{ij}$ , A) <= ε)
(6)     C = C ∪ {i};
(7) }
(8) return C;

```

ALGORITHM 1: GC algorithm.

```

Input RE,  $w^{LB}$  and distance measurements between any node pair
Output A resource provider set P
(1)  $P = \Phi$ ;  $\delta = \frac{RE/SE_{max}}{w^{LB}}$ ; /* SN is null at beginning */
(2) for each  $i: i \in C$  parallel.do { /* Parallel process for each  $i$  */
(2.1)  $x_i = 0$ ; /* Node  $i$  is a candidate */;
(2.2) calculate  $w_i$ ;
(2.3) if ( $w_i \geq 1$ )  $x_i = -1$ ; /*  $i$  is no longer a candidate */
} /* end for */
(3) for each  $i: i \in C$  & ( $i$  is a candidate) parallel.do {
(3.1)  $p_i = \min\{1, (w^{LB}/w_i)^{w_i/w^{LB}}\}$ ;
(3.2) if  $p_i > \delta$   $x_i = 1$ ; /*  $i$  becomes a provider */
} /* end for */
(4) for each  $i: i \in C$  & ( $i$  is a candidate) parallel.do
(4.1) if ( $x_j = 0$  for all  $j$  with distance ( $i, j$ ) ≤ L)
(4.2)  $x_i = 1$ ; /*  $i$  becomes a provider */
(5)  $P = \{i \mid x_i = 1, i \in C\}$ ;

```

ALGORITHM 2: PAS procedure.

Therefore, the constraint optimisation can be formulated as a 0-1 integer linear programming (ILP) problem as shown in the following.

0-1 ILP for Resource Allocation

OPT1:

$$\min \sum_{i=1}^{|C|} w_i x_i \quad (6)$$

$$\text{s.t. } \sum_{j=1}^{|C|} a_{ij} x_j > 1, \quad \forall 1 \leq i \leq |C| \quad (7)$$

$$x_i \in \{0, 1\}, \quad \forall 1 \leq i \leq |C|, \quad (8)$$

where a_{ij} is a decision variable related to the geography distance between node i and node j :

$$a_{ij} = \begin{cases} 1, & \text{if distance}(i, j) \leq L, \\ 0, & \text{otherwise.} \end{cases} \quad (9)$$

In OPT1, the number of grids constraint is explicitly represented by inequality (7), the surplus energy constraint is formulized by w_i , the measuring accuracy is considered

by ε and t , and hence represented by L , and the sensory data attributes constraints are examined by D and s , and also represented by L .

Finding the optimal solution for ILP is NP-hard and may be solved in linear time as an LP-type problem with a constant number of variables [41, 42]. Approaches, such as enumeration, cutting plane, and branch and bound, are unacceptable for real-time scheduling in the scenario of air pollution monitoring because the time complexity of them exponentially increases with the number of variables. Hence, approximate solutions are compromised for such problems.

4.4. Find Out the Solution of Scheduling. Here we give a proximate algorithm for scheduling (PAS) to find the resource provider set P . In this algorithm, a parameter w^{LB} is used which is defined as follows

w^{LB} is a lower bound for all w_i , which is a predefined constant satisfying $0 < w^{LB} < 1$.

Algorithm 2 shows the pseudocode of the parallel procedure of PAS in each candidate. In the procedure, δ is a threshold where SE_{max} is the maximum surplus energy in the whole network which can be simply set as the initial energy value. Step 2.3 deletes all the nodes that have less surplus

energy than the required energy from the candidate set. Step 3.1 is the key processing where each node calculates the probability of becoming a provider. The probability function makes the nodes with weights comparatively closer to w^{LB} have higher probability to become providers (in the case that w_i is smaller than w^{LB} , node i will become a provider with $p_i = 1$). While Step 4 is a complementary process which guarantees that the set P satisfies the requirement of measurement accuracy, for any node, if there is no other provider within distance L , this node becomes a provider.

5. Performance Analysis

5.1. Scheduling Algorithm Performance Analysis

5.1.1. Complexity Analysis. The time complexity of FC algorithm is $O(|V|)$. In the PAS procedure, each of the steps 2, 3, and 4 has the time complexity $O(|C|)$. Therefore, the time complexity of the whole algorithm is $O(|V|)$.

For the message complexity, suppose the maximum degree of the sensor network topology is Δ . The algorithm only requires the message exchange in PAS step 4. Hence, the message complexity is $O(\Delta|C|) = O(\Delta|V|)$.

5.1.2. Size of Provider Set. In this experiment, we calculate the average size of the provider set P and the calculation time of PAS. We compare both of the values with the results calculated by ILP.

We use a topology generator to generate random topologies in an area with radius = 100. In consideration of the purpose of this experiment, we simply assume that all the nodes are candidates and the maximum distance L is given different values instead of calculated by formula (4) (the selection of candidates and the calculation of L will not affect the results in this experiment). For different topology parameter values, the random graph is generated and simulated until a predefined confidence interval for the population mean is reached and, then, simulation results are measured by simply taking the average of all cases. Here, we achieve a precision of 1% with the 90% confidence interval of the provider set. In the experiment, $w^{LB} = 0.01$ and $RE = 0.9$. Each node is randomly assigned a surplus energy value between 0 and 100. Then $\delta = (RE/SE_{max})/w^{LB} = 0.9$, which means if the value p_i in PAS step 3.1 is larger than 0.9, then node i becomes a provider. The total number of nodes ranges from 40 to 130. The experiment investigates the impact of different distance limitation L on the size of P . The results are shown in Figure 2.

In the figure, we can see that the size of provider set P generated by PAS approximately increases linearly with the total number of nodes. Larger L corresponds to smaller P because a single node can cover a larger geographical area. The size of P generated by PAS is about 1 to 2 times of that generated by ILP. As OPT1 matches the classic minimum independent set problem, according to [30], the size of any independent set in a unit-disk graph is at most $4opt + 1$ our algorithm gives a reasonable result.

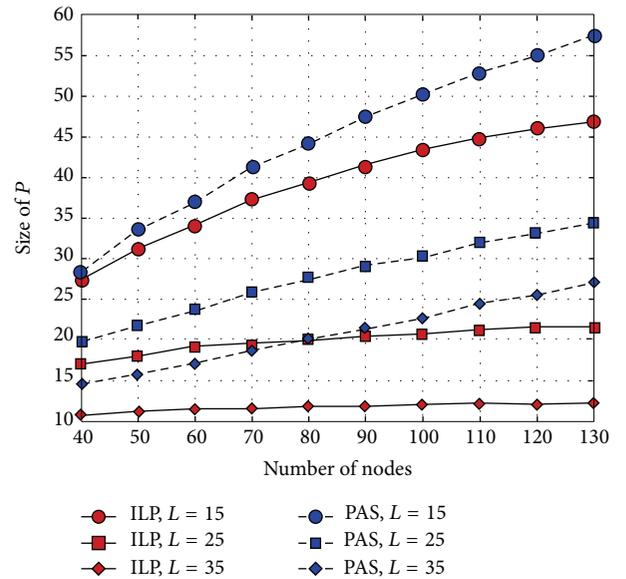


FIGURE 2: Size of provider set with different distance limitation.

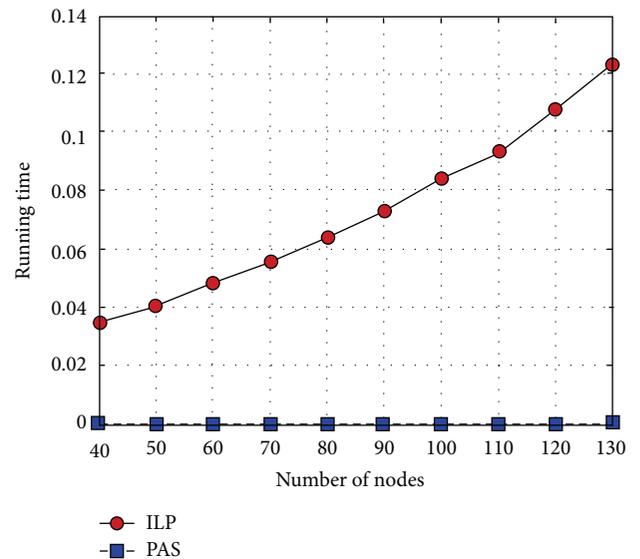


FIGURE 3: Comparison of running time ($L = 25$).

5.1.3. Running Time. This experiment compares the calculation times of PAS and ILP with $L = 25$. The result is shown in Figure 3. From the figure we can see that the convergence time of our algorithm is much less than that of ILP, and our algorithm is network scale independent while the running time of ILP increases with the increasing total number of nodes. Hence, our algorithm has better performance in scalability.

5.1.4. Average Surplus Energy. This experiment calculates the average surplus energy (SE) of each node in provider set. PAS algorithm is an optimisation solution aiming to minimize the ratio of the energy consumption, in other word, maximize the surplus energy of the provider set with a given required

TABLE 2: Comparison of number of providers generated by PAS Steps 3.2 and 4.2.

Number of nodes	PAS step		Number of nodes	PAS step	
	3.2	4.2		3.2	4.2
40	7.0700	12.6667	90	16.3800	12.9133
50	9.1200	13.0700	100	17.9633	12.5867
60	11.0633	13.1833	110	19.9667	12.1800
70	13.0100	13.2200	120	22.4867	11.5933
80	14.3367	13.2267	130	23.8700	11.5333

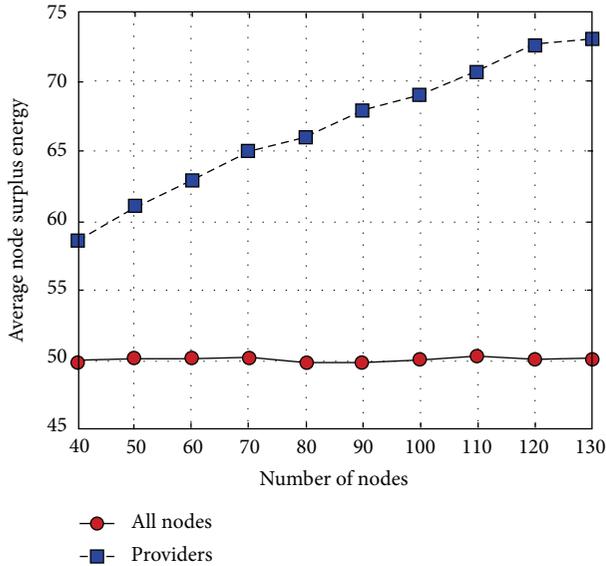


FIGURE 4: Comparison of surplus energy.

TABLE 3: Air pollution monitoring scheduling results.

	08:30	15:30	17:30
L (meters)	199.92	128.30	187.44
Number of candidates	17	3	9
Number of providers	6	2	4
AR	0.35	0.67	0.44

energy for a task. Therefore, we expect that the provider set generated by PAS has higher average SE in comparison with that of the whole network.

The result is shown in Figure 4. For the whole network, as the SE of each node is randomly assigned from 0 to 100, the average SE is about 50. For the providers, the curve in the figure presents two features. First, the average SE is much larger than 50 as we expected. Second, SE approximately linearly increases with the number of nodes. To explain this, let us check the providers generated by PAS. In PAS, a node has two chances to become a provider: in step 3.2 and step 4.2. Step 3.2 is a mandatory criterion for a node to become a provider if $p_i > \delta$ (i.e., this node has very low energy consumption rate or very high SE). And step 4.2 is a complementary processing to satisfy the distance constraint. So the more proportion of providers the selected by step 3.2,

the higher average SE is achieved. From Table 2 we can see that, in PAS the number of providers selected by step 4.2 is about a constant around 12, whereas the number of providers selected by step 3.2 increases with increasing total number of nodes. This statistics explains the result in Figure 4 well and this experiment proves that our system has high performance in energy consumption.

5.2. EIMAP System Performance Measurement. In this experiment, we use WikiSensing [43] and Siege benchmarking utility [44] to simulate our EIMAP system. WikiSensing is an online collaborative platform for sensor data management. It can simulate as many sensors as the system being tested requires, including sensor registration, data sampling, user query response and database management. We use WikiSensing to simulate the lower 2 layers of EIMAP: the sensor layer is simulated by generating 140 nodes records with specified location IDs. Each sensor has a sequence of readings stored in the database. The database is maintained on the IC cloud computing infrastructure [45]. Each node has the capability of receiving queries and sending response. The elastic management layer is realized by integrating our scheduling algorithm into the optimization module of WikiSensing. As the data analysis functions are not essential for this experiment, we can treat the data analysis layer as a layer that executes nothing but transmits the user queries from the interface between 3rd/4th layer to the interface between 2nd/3rd layer directly. And the application layer is simulated by the Siege benchmarking. It can simulate the users' behavior of accessing a web server with a configurable number of concurrent simulated users. The duration of the "siege" is measured in transactions, the sum of simulated users, and the number of times each simulated user repeats the process of accessing the server. With Siege benchmarking, it is possible for us to measure the performance of EIMAP to see how it will stand up to load on the internet. The simulation environment is illustrated as shown in Figure 5.

The experiment uses Siege to simulate concurrent users from 100 to 1000. The elapsed time of each test is 60 seconds. In WikiSensing we simulated 30 sensors and different aggregation ratio AR. Here we define AR as follows:

$$AR = \frac{\text{Number of providers}}{\text{Number of candidates}}. \quad (10)$$

The data stored in IC Cloud is air pollution data which will be described in detail in the next section. The performance evaluation calculates the average response time of the

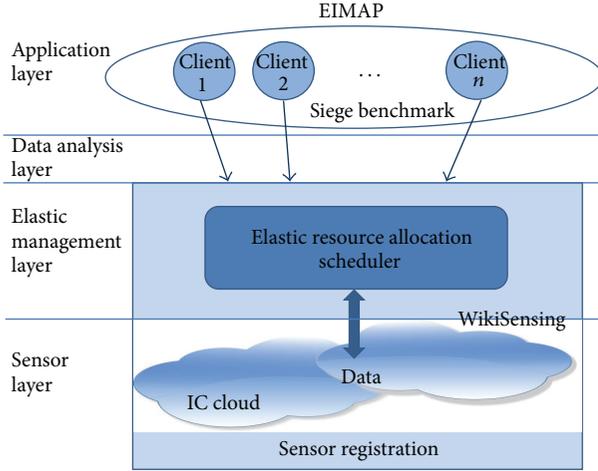


FIGURE 5: EIMAP system performance testing environment.

queries, which is the round trip time of sending a request and receiving a response. The results are shown in Figure 6.

In Figure 6, the response time presents linear increase as the number of concurrent users increases. $AR = 1$ means the system collects data from all the candidates (in most of existing approaches [2, 3, 6, 7] including our former research [5]; although the system architectures and resource providing schemes are different, they all can be categorised into the design with a scheduler that $AR = 1$), while $AR = 0.1$ means 1/10 of sensors are chosen to be providers and the system will only collect data from them. As AR increases, the response time increases (the response time of $AR = 0.1$ is much shorter than that of $AR = 1$), and hence, providing a better system performance for clients.

6. Air Pollution Scenario

In this section, we introduce a case study for our algorithm by applying it to the air pollution scenario. The experiment, based on our former research [5], uses the air pollution data collected from 140 sensors (in a 100-metre rectangular grid) distributed in a $1\text{ km} \times 1.4\text{ km}$ area represented as red dots in the map of Figure 7(a). The map shows an urban area around the Tower Hamlets and Bromley areas in East London. There are some of the typical urban landmarks such as the main road extending from A6 to L10, the hospitals around C5 and K4, the schools in B7, C8, D6, F10, G2, H8, K8, and L3, the train stations at D7 and L5, and Gas Works between D2 and E1. 140 sensors collect data from 8:00 to 17:59 at a 1-minute interval to monitor the pollution volumes of NO, NO₂, SO₂, and Ozone. Then there are 600 data items for each node and totally 84000 data items for the whole network. Each data item is identified by a time stamp, a location, and a four-pollutant volume reading. The time-plot profiles of four pollutants over 10 hours are shown in Figure 7(b). Each profile is the overlap time plots of all the 140 sensors for one pollutant over 10 hours. For example, the upright figure shows the volume of NO from 08:00 to 17:59. At 8:30, 140 sensors generate three typical readings: over

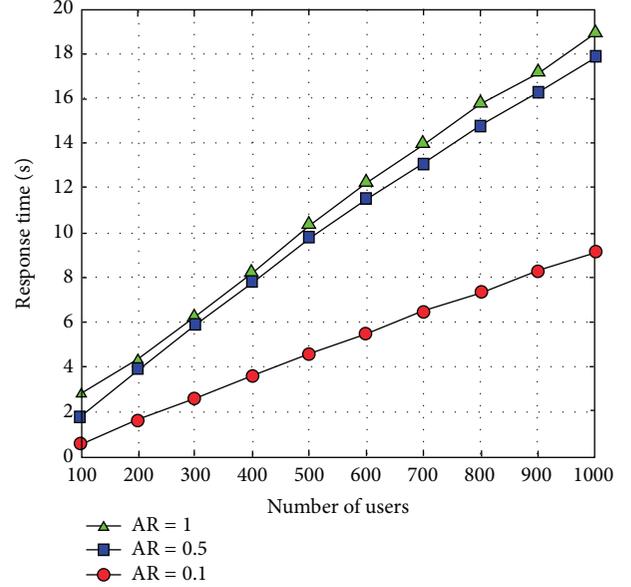


FIGURE 6: Average response time of EIMAP.

200 ppm, between 60 ppm and 80 ppm, and less than 20 ppm. However, this figure cannot tell us which sensor generates what readings.

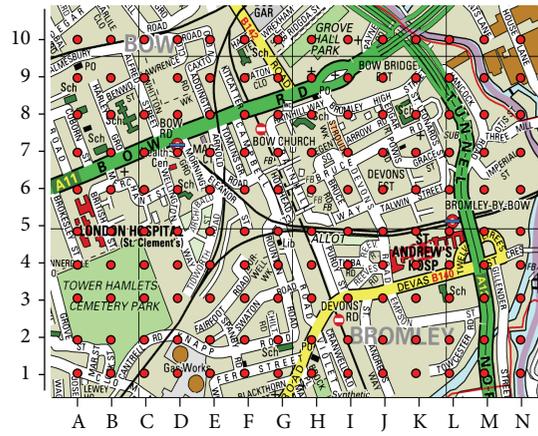
The case study will investigate the resource provision for tracking a given feature of interest. For this purpose, we specify the feature with high volume of NO + NO₂ + SO₂, which is defined as a vector $A = (170, 180, 150)$. And we pick up 3 time stamps 08:30, 15:30 and 17:30 for data analysis (according to Figure 7(b), around these 3 time stamps there exist fairly high level pollution volumes of NO, NO₂, and SO₂ in some of the locations that are distinct compared to other locations). As feature A is the saliency of the pollutants concentration which stands out against their neighbours/surroundings, according to air pollution dispersion characteristics (the concentration of traffic emissions on highway decayed 50% at 150 m location and further 30% at 400 m location [46]), we define $\epsilon = \|A\| \cdot 30\%$, which means a sampled value matches A if it falls into the intervals of $[\|A\| - \epsilon, \|A\| + \epsilon]$. And we delimit a sampling unit as an area that is covered by 25 grid units/nodes (about $500\text{ m} \times 500\text{ m}$). The maximum distance L is calculated as follows:

$$L = \sqrt{\frac{S}{N_{\max}}}, \quad (11)$$

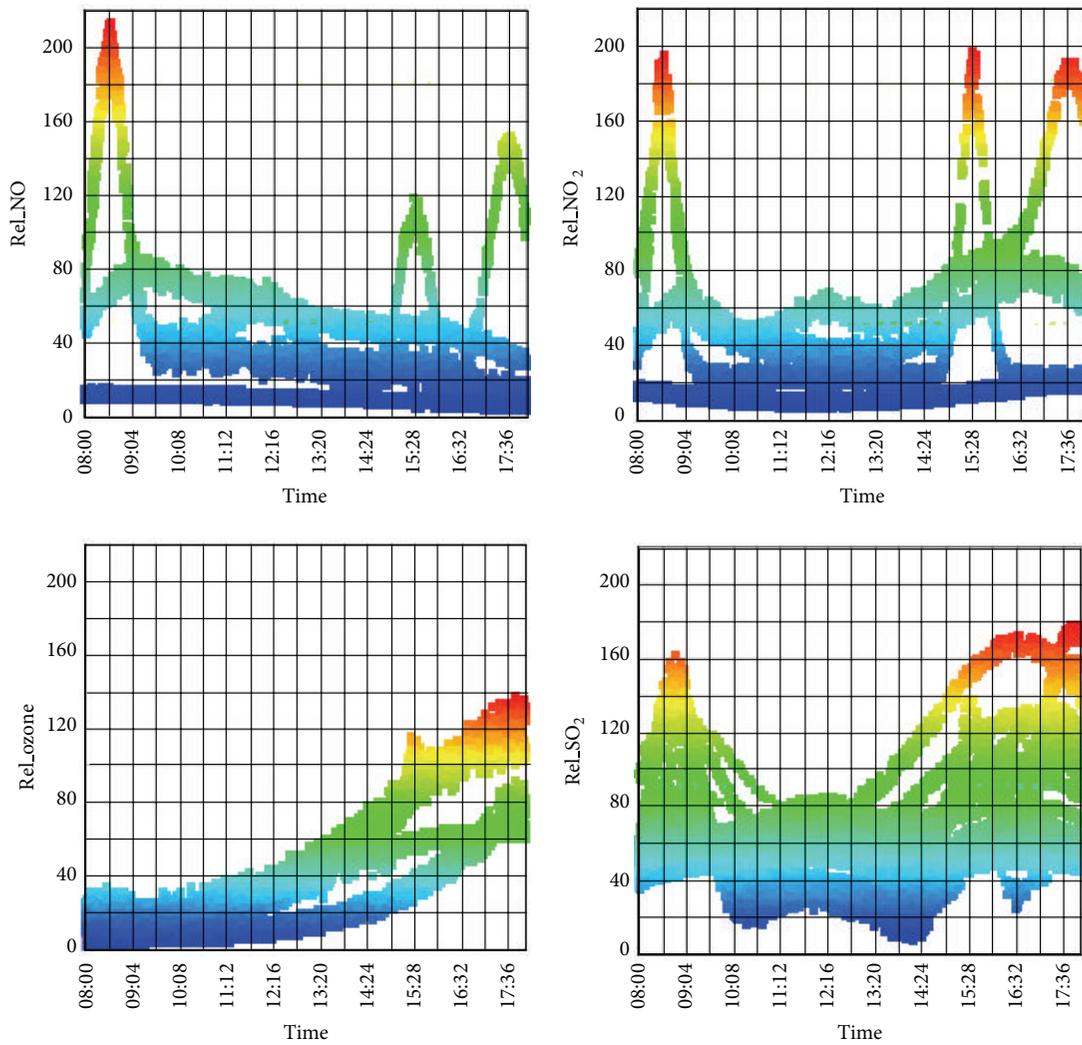
$$N_{\max} = \arg \max \{N_{\text{NO}}, N_{\text{NO}_2}, N_{\text{SO}_2}\},$$

which means we calculate N for each pollutant in each sampling unit and the maximum N is used to calculate L . Other parameters are given the same values as described in Section 5.

Table 3 summarises the results of executing the scheduling algorithm in this area. The values of L are different because the values of N are different according to formula (4). Figure 6 visualises the results of the feature tracking. Figure 8(a)(A)–(C) highlight the areas of interest monitored



(a) 140 sensors distributed in an area of East London



(b) Time plots profiles of four pollutants over 10 hours

FIGURE 7: Sensor distribution and data profiles in an area of East London.

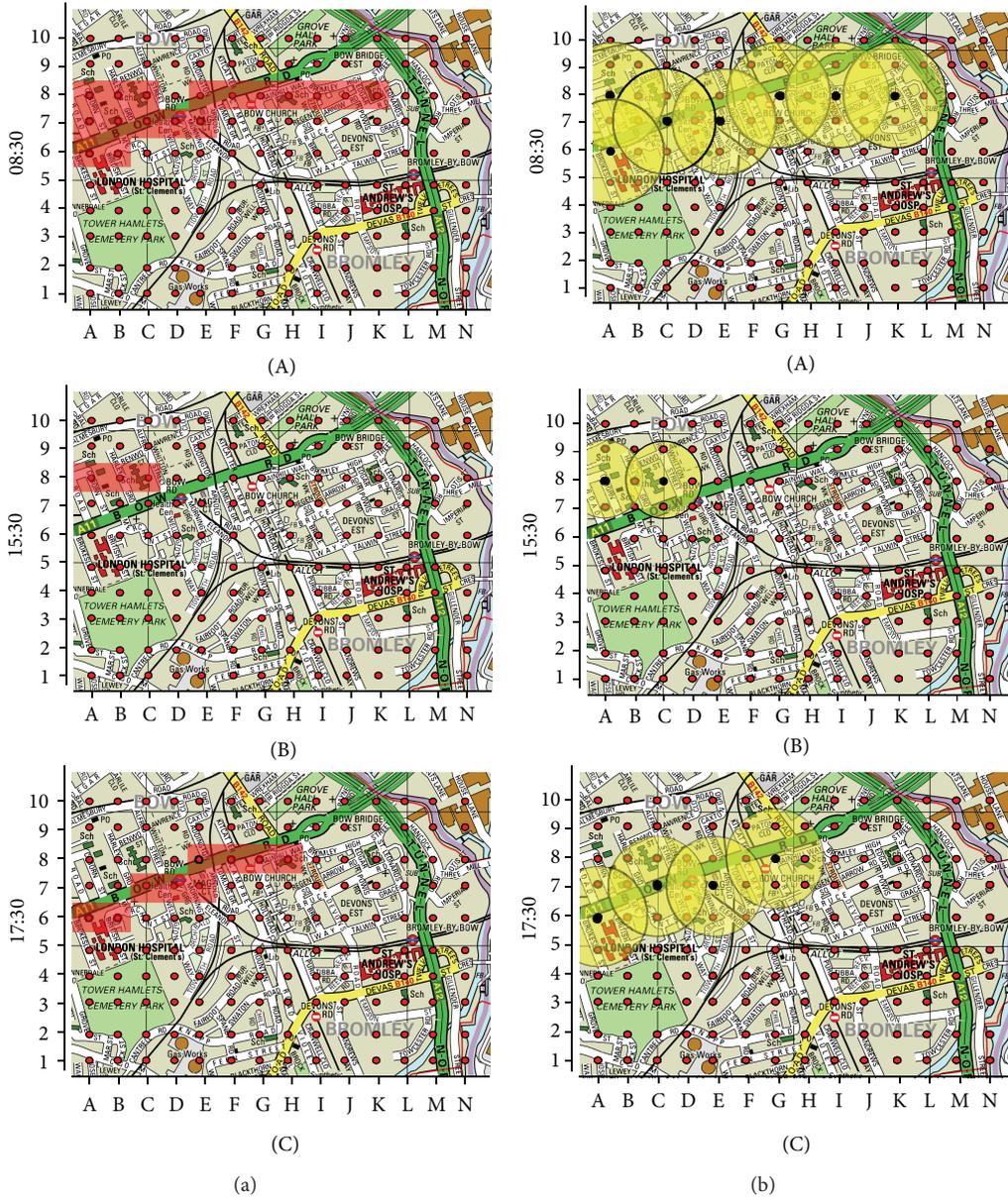


FIGURE 8: Visualisation of feature tracking.

by all the candidates at different time stamps. For example, in the morning the feature is located at the main road and schools (A8, B7, H8, and K8). At 15:30, the feature is only found at two schools and at 17:30 the feature only covers the main road. This characteristic matches the traffic property during a weekday (people going to school and work in the morning rush hours by vehicles makes both the main road and school areas have high pollution, while people off school at about 15:30 and off work at about 17:30, respectively, make the pollution distribution different). These three figures show the resource provision without scheduling scheme, where all the nodes that match the feature are active. Figure 8(b)(A)–(C) illustrate the resource providers chosen by our scheduling algorithm. Each provider is represented by a black node and the yellow circle is the corresponding sensor coverage

area with radius L . From the figures we can see that the areas of interest are all picked up and monitored by fewer sensors. Hence with our scheduling algorithm, the resource is elastically provided, whereas the feature tracking still performs well.

7. Conclusion

In this paper, we discussed the main challenges in information management and real-time resource allocation when applying large-scale M2M sensor networks to air pollution monitoring. An elastic information management architecture was proposed to address those challenges by using pervasive roadside and vehicle/person-mounted sensors by

combining and extending state-of-the-art elastic computing and data management techniques. The experiments results on the elastic resource allocation scheme, the entire system performance, and the air pollution monitoring case study show that our design provides higher performance in energy efficiency and system response speed, as well as an effective saliency detection and coverage in the scenario of pollutant distribution with less sensors.

Direct further work on the algorithm improvement includes continued research on other resource constraints not taken into account in this paper, such as the storage capability and available bandwidth. Our long-term work will focus on the development of the management platform to allow demonstration and further analysis of other applications. The integration with sensor hardware is also a key step which can provide the collection of the application data from the real world to support the real-time data analysis.

Acknowledgments

This work was jointly supported by National Science Foundation of China, Grant no. 61104215, and Engineering and Physical Science Research Council (EPSRC), Grant no. EP/H042512/1. This work is also partly supported by project “Digital City Exchange,” Grant no. EP/I038837/1, funded by Research Council UK.

References

- [1] S. Wylie, J. Heide, B. Avcı, D. D. Vaccaro, O. Ghica, and G. Trajcevski, “Distributed data management for large-scale wireless sensor networks simulations,” in *Proceedings of the 15th International Conference on Extending Database Technology*, pp. 626–629, Berlin, Germany, March 2012.
- [2] M. Balazinska, A. Deshpande, M. J. Franklin et al., “Data management in the worldwide sensor web,” *IEEE Pervasive Computing*, vol. 6, no. 2, pp. 30–40, 2007.
- [3] C. Jardak, J. Riihijärvi, F. Oldewurtel, and P. Mähönen, “Parallel processing of data from very large-scale wireless sensor networks,” in *Proceedings of the 19th ACM International Symposium on High Performance Distributed Computing (HPDC '10)*, pp. 787–794, Chicago, Ill, USA, June 2010.
- [4] M. Welsh, “Where do we go from here? The big problems in sensor networks,” in *Proceedings of the Wireless Sensing Solutions Conference*, Chicago, Ill, USAA, Keynote Talk, September 2005.
- [5] Y. Ma, M. Richards, M. Ghanem, Y. Guo, and J. Hassard, “Air pollution monitoring and mining based on sensor Grid in London,” *Sensors*, vol. 8, no. 6, pp. 3601–3623, 2008.
- [6] M. Li, D. Ganesan, and P. Shenoy, “PRESTO: feedback-driven data management in sensor networks,” *IEEE/ACM Transactions on Networking*, vol. 17, no. 4, pp. 1256–1269, 2009.
- [7] B. Tang and Y. Wang, “Design of large-scale sensory data processing system based on cloud computing,” *Research Journal of Applied Sciences, Engineering and Technology*, vol. 4, no. 8, pp. 1004–1009, 2012.
- [8] S. Dustdar, Y. Guo, B. Satzger, and H.-L. Truong, “Principles of elastic processes,” *IEEE Internet Computing*, vol. 15, no. 5, pp. 66–71, 2011.
- [9] R. Buyya, J. Broberg, and A. Goscinski, *Cloud Computing: Principles and Paradigms*, John Wiley and Sons, 2011.
- [10] E. Dowling, *Introduction to Mathematical Economics*, McGraw-Hill, 3rd edition, 2000.
- [11] D. Carney, U. Cetintemel, M. Cherniack et al., “Monitoring streams: a new class of data management applications,” in *Proceedings of the 28th International Conference on Very Large Data Bases (VLDB '02)*, pp. 215–226, Hong Kong, 2002.
- [12] S.-H. Baek, E.-C. Choi, J.-D. Huh, and K.-R. Park, “Sensor information management mechanism for context-aware service in ubiquitous home,” *IEEE Transactions on Consumer Electronics*, vol. 53, no. 4, pp. 1393–1400, 2007.
- [13] T. Kawakami, B. L. N. Ly, S. Takeuchi, Y. Teranishi, K. Harumoto, and S. Nishio, “Distributed sensor information management architecture based on semantic analysis of sensing data,” in *Proceedings of the International Symposium on Applications and the Internet (SAINT '08)*, pp. 353–356, Turku, Finland, August 2008.
- [14] W. E. L. Grimson, C. Stauffer, R. Romano, and L. Lee, “Using adaptive tracking to classify and monitor activities in a site,” in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 22–29, Los Alamitos, Calif, USA, June 1998.
- [15] S. R. Madden, M. J. Franklin, J. M. Hellerstein, and W. Hong, “TinyDB: an acquisitional query processing system for sensor networks,” *ACM Transactions on Database Systems*, vol. 30, no. 1, pp. 122–173, 2005.
- [16] S. Madden, M. J. Franklin, J. M. Hellerstein, and W. Hong, “The design of an acquisitional query processor for sensor networks,” in *Proceedings of the ACM SIGMOD International Conference on Management of Data*, pp. 491–502, San Diego, Calif, USA, June 2003.
- [17] A. Deshpande, C. Guestrin, S. R. Madden, J. M. Hellerstein, and W. Hong, “Model-driven data acquisition in sensor networks,” in *Proceedings of the 30th International Conference on Very Large Data Bases (VLDB '04)*, pp. 588–599, Toronto, Canada, 2004.
- [18] Y. Diao, D. Ganesan, G. Mathur, and P. Shenoy, “Rethinking data management for storage-centric sensor networks,” in *Proceedings of the 3rd Biennial Conference on Innovative Data Systems Research (CIDR '07)*, pp. 22–32, Asilomar, Calif, USA, January 2007.
- [19] E. H. Aoki, A. Bagchi, P. Mandal, and Y. Boers, “A theoretical look at information-driven sensor management criteria,” in *Proceedings of the 14th International Conference on Information Fusion (Fusion '11)*, Chicago, Ill, USA, July 2011.
- [20] S. Acharya, P. B. Gibbons, V. Poosala, and S. Ramaswamy, “The Aqua approximate query answering system,” in *Proceedings of the ACM SIGMOD International Conference on Management of Data*, pp. 574–576, Philadelphia, Pa, USA, June 1999.
- [21] Y. E. Ioannidis and V. Poosala, “Histogram-based approximation of set-valued query answers,” in *Proceedings of the 25th International Conference on Very Large Data Bases (VLDB' 99)*, pp. 174–185, Edinburgh, Scotland, September 1999.
- [22] K. Chakrabarti, M. Garofalakis, R. Rastogi, and K. Shim, “Approximate query processing using wavelets,” in *Proceedings of the 26th International Conference on Very Large Databases (VLDB' 00)*, pp. 111–122, Cairo, Egypt, September 2000.
- [23] D. Ganesan, B. Greenstein, D. Perelyubskiy, D. Estrin, and J. Heidemann, “An evaluation of multi-resolution storage for sensor network,” *SIGCOMM Computer Communication Review*, vol. 34, no. 1, pp. 125–130, 2004.
- [24] T. H. De Groot, R. F. Tigrek, O. A. Krasnov, A. Huizing, and A. Yarovoy, “Resource allocation challenges for reconfigurable

- multi-sensor networks,” in *Proceedings of the 8th European Radar Conference (EuRAD '11)*, pp. 142–145, Manchester, UK, October 2011.
- [25] D. Pizzocaro, A. Preece, F. Chen, T. L. Porta, and A. Bar-Noy, “Demo: a distributed architecture for heterogeneous multi sensor-task allocation,” in *Proceedings of the 7th IEEE International Conference on Distributed Computing in Sensor Systems (DCOSS '11)*, Barcelona, Spain, June 2011.
- [26] N. Edalat, W. Xiao, C.-K. Tham, E. Keikha, and L.-L. Ong, “A price-based adaptive task allocation for wireless sensor network,” in *Proceedings of the 6th International Conference on Mobile Adhoc and Sensor Systems (MASS '09)*, pp. 888–893, October 2009.
- [27] Y. Song, Y. Sun, and W. Shi, “A two-tiered on-demand resource allocation mechanism for VM-based data centers,” *IEEE Transactions on Services Computing*, vol. 6, no. 1, pp. 116–129, 2013.
- [28] Y. He, W. Zhu, and L. Guan, “Optimal resource allocation for pervasive health monitoring systems with body sensor networks,” *IEEE Transactions on Mobile Computing*, vol. 10, no. 11, pp. 1558–1575, 2011.
- [29] T. H. de Groot, O. A. Krasnov, and A. Yarovoy, “Adaptive optimization algorithms for utility-driven resource allocation in reconfigurable multi-sensor networks,” in *Proceedings of the 9th European Radar Conference*, pp. 330–333, Amsterdam, The Netherlands, November 2012.
- [30] M. P. Johnson, H. Rowaihy, D. Pizzocaro et al., “Sensor-mission assignment in constrained environments,” *IEEE Transactions on Parallel and Distributed Systems*, vol. 21, no. 11, pp. 1692–1705, 2010.
- [31] M. Şensoy, T. Le, W. W. Vasconcelos, T. J. Norman, and A. D. Preece, “Resource determination and allocation in sensor networks: a hybrid approach,” *Computer Journal*, vol. 54, no. 3, pp. 356–372, 2011.
- [32] R. Mattikalli, R. Fresnedo, P. Frank, S. Locke, and Z. Thunemann, “Optimal sensor selection and placement for perimeter defense,” in *Proceedings of the 3rd IEEE International Conference on Automation Science and Engineering (IEEE CASE '07)*, pp. 911–918, Washington, DC, USA, September 2007.
- [33] N. D. B. Bruce and J. K. Tsotsos, “Saliency, attention and visual search: an information theoretic approach,” *Journal of Vision*, vol. 9, no. 3, article 5, pp. 1–24, 2009.
- [34] H. Liu and I. Heynderickx, “Visual attention in objective image quality assessment: based on eye-tracking data,” *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 21, no. 7, pp. 971–982, 2011.
- [35] O. Le Meur, A. Ninassi, P. Le Callet, and D. Barba, “Overt visual attention for free-viewing and quality assessment tasks: impact of the regions of interest on a video quality metric,” *Signal Processing*, vol. 25, no. 7, pp. 547–558, 2010.
- [36] X.-P. Hu, L. Dempere-Marco, and E. R. Davies, “Bayesian feature evaluation for visual saliency estimation,” *Pattern Recognition*, vol. 41, no. 11, pp. 3302–3312, 2008.
- [37] M. Younis, M. Youssef, and K. Arisha, “Energy-aware management for cluster-based sensor networks,” *Computer Networks*, vol. 43, no. 5, pp. 649–668, 2003.
- [38] T. P. Huynh, Y. K. Tan, and K. J. Tseng, “Energy-aware wireless sensor network with ambient intelligence for smart LED lighting system control,” in *Proceedings of the 37th Annual Conference of the IEEE Industrial Electronics Society (IECON '11)*, pp. 2923–2928, Victoria, Australia, November 2011.
- [39] I. Papadimitriou and L. Georgiadis, “Energy-aware routing to maximize lifetime in wireless sensor networks with mobile sink,” *Journal of Communications Software and Systems*, vol. 2, no. 2, pp. 141–151, 2006.
- [40] The United States Environmental Protection Agency document, “Guidance on Choosing A Sampling Design for Environmental Data Collection,” EPA QA/G-5S, EPA/240/R-02/005, 2002.
- [41] J. Matoušek, M. Sharir, and E. Welzl, “A subexponential bound for linear programming,” *Algorithmica*, vol. 16, no. 4-5, pp. 498–516, 1996.
- [42] K. Aardal and F. Eisenbrand, “Integer programming, lattices, and results in fixed dimension,” *Handbooks in Operations Research and Management Science*, vol. 12, pp. 171–243, 2005.
- [43] D. Silva, M. Ghanem, and Y. Guo, “WikiSensing: an online collaborative approach for sensor data management,” *Sensors*, vol. 12, no. 10, pp. 13295–13332, 2012.
- [44] “Siege Benchmark,” <http://www.joedog.org/siege-home/>.
- [45] Y.-K. Guo and L. Guo, “IC cloud: enabling compositional cloud,” *International Journal of Automation and Computing*, vol. 8, no. 3, pp. 269–279, 2011.
- [46] T. Reponen, S. A. Grinshpun, S. Trakumas et al., “Concentration gradient patterns of aerosol particles near interstate highways in the Greater Cincinnati airshed,” *Journal of Environmental Monitoring*, vol. 5, no. 4, pp. 557–562, 2003.



Hindawi

Submit your manuscripts at
<http://www.hindawi.com>

