Cloud-Assisted Data Fusion and Sensor Selection for Internet of Things

Farshid Hassani Bijarbooneh, Wei Du, Edith C.-H. Ngai, Xiaoming Fu, *Senior Member, IEEE*, and Jiangchuan Liu, *Senior Member, IEEE*

***Abstract*—The Internet of Things (IoT) is connecting people and smart devices on a scale that was once unimaginable. One major challenge for the IoT is to handle vast amount of sensing data generated from the smart devices that are resource-limited and subject to missing data due to link or node failures. By exploring cloud computing with the IoT, we present a cloud-based solution that takes into account the link quality and spatio-temporal correlation of data to minimize energy consumption by selecting sensors for sampling and relaying data. We propose a multiphase adaptive sensing algorithm with belief propagation (BP) protocol (ASBP), which can provide high data quality and reduce energy consumption by turning on only a small number of nodes in the network. We formulate the sensor selection problem and solve it using both constraint programming (CP) and greedy search. We then use our message passing algorithm (BP) for performing inference to reconstruct the missing sensing data. ASBP is evaluated based on the data collected from real sensors. The results show that while maintaining a satisfactory level of data quality and prediction accuracy, ASBP can provide load balancing among sensors successfully and preserves 80% more energy compared with the case where all sensor nodes are actively involved.**

***Index Terms*—Belief propagation, constraint optimization, Internet of Things (IoT), quantization, wireless sensor networks.**

I. Introduction

THE INTERNET has enabled an explosive growth of information sharing. With the advent of embedded and sensing technology, the number of smart devices including sensors, mobile phones, RF identifications (RFIDs), and smart grids has grown rapidly in recent years. Ericsson and Cisco predicted that 50 billion small embedded sensors and actuators will be connected to the Internet by 2020 [1] forming a new Internet paradigm called Internet of Things (IoT). IoT can support a wide range of applications in different domains, such as health care, smart cities, pollution monitoring, transportation and logistics, factory process optimization, home safety and security [2], [3].

In the past decade, many studies have contributed to the hardware, software, and protocol design of the smart devices, such as wireless sensor networks (WSNs) [4]–[6]. Machine-to-machine automation with wireless sensors is being widely deployed, but usually in islands of disparate systems. The evolution of IoT attempts to connect these existing systems to the cloud, which enables advanced data fusion, storage, and coordination capability for achieving higher data quality and energy efficiency. The upcoming challenge of IoT lies in handling volumes of data generated from enormous amount of devices, which is known as the big data problem.

The wireless sensors in many IoT applications are battery powered, resulting in extreme energy constraints on their operations, such as sampling, data processing and radio communications. To conserve energy and achieve longer network lifetime, the costs of sensor sampling, processing, and radio communications have to be minimized. It is often the case that sensor readings in the same spatial regions are highly correlated. Depending on the application, the sensor readings are temporally correlated as well. By leveraging the computation capability of the cloud, data fusion can be performed to increase the data quality by exploring the spatial and temporal correlation of data. The wireless sensors can be coordinated by the cloud to be ON and OFF according to the change in the environment. In this paper, we explore a seamless solution by integrating cloud and IoT to provide comprehensive data fusion and coordination of sensors to improve data quality and reduce energy consumption.

Belief propagation (BP) [7]–[9] is a technique for solving inference problems. In the IoT context, the belief of a sensor node is the data measurement of an event in the environment, and BP provides an iterative algorithm (also called the sum-product algorithm) to infer the measurements of the sensor nodes, especially in cases where the data are missing, because of packet losses or because there are no data available at some selectively disabled sensor nodes (mainly to conserve energy and reduce radio inference). In BP, each sensor node determines its belief by incorporating its local measurement with the beliefs of its neighbor sensor nodes (spatial cooperation), and its beliefs obtained in the past (temporal cooperation). In such inference problems, the assumption that the data are spatio-temporally correlated significantly improves the accuracy of data inference using BP in WSNs.

In monitoring applications for the IoT, the data are collected and put in an *environment matrix* (EM) [10], where the data readings for each sensor node are stored in one row of the matrix and each column index represents a timestamp for the interval at which the data were sampled. Hence, an EM is a matrix of size $N \times T$ where N is the number of sensor nodes

Manuscript received August 27, 2015; revised October 18, 2015; accepted November 02, 20015. Date of publication November 19, 2015; date of current version May 10, 2016.

F. H. Bijarbooneh and E. C.-H. Ngai are with the Department of Information Technology, Uppsala University, 751 05 Uppsala, Sweden (e-mail: Farshid.Hassani@it.uu.se; Edith.Ngai@it.uu.se).

W. Du is with CITI Lab, INSA-Lyon, 69621 Villeurbanne, France (e-mail: wei.du@insa-lyon.fr).

X. Fu is with the Institute of Computer Science, University of Göttingen, 37073 Göttingen, Germany (e-mail: fu@cs.uni-goettingen.de).

J. Liu is with the School of Computing Science, Simon Fraser University, Burnaby, BC V5A 1S6, Canada (e-mail: jcliu@cs.sfu.ca).

Digital Object Identifier 10.1109/JIOT.2015.2502182

and T the number of time intervals, and the time dimension T is expanding as more data are collected. BP performs the inference iteratively from the stream of data that are stored in EM based on the current and past data. Therefore, unlike the compressed sensing (CS) [11] approach, BP does not require a complete EM for the whole duration of the time interval to perform inference.

In this paper, we explore cloud-assisted adaptive sensing and data fusion to reduce energy consumption and improve data quality for the IoT. We propose an adaptive sensing BP protocol (ASBP), where the data are collected in several *rounds* (a round is a fixed time interval where the network repeats the same behavior) by active sensors (sensors that are collecting data in each round). We formulate and solve an optimization problem that selects the active sensors in each round, by maximizing the data utility while maintaining energy load balancing. We define *data utility* as the sum of the qualities of the path links from the selected active sensor nodes to the base station, subtracted by the sum of the correlations of the selected active sensors. If the selected active sensor nodes are located on a path with greater link quality, then the value of the data utility increases. Likewise, if the selected active sensor nodes result in a lower data correlation, then the data utility is increased. In each round of ASBP, the minimum number of selected active sensor nodes (which is a parameter of our sensor selection optimization problem) is adaptively tuned based on the performance of the BP inference (data prediction accuracy) throughout the previous round. In addition to BP, we also use data quantization to further compress the data and reduce the transmission costs.

In our active sensor selection formulation, we consider nonlinear multihop routing protocol constraints. To model the sensor selection problem effectively, we use both constraint programming (CP) [12] and heuristic-based greedy algorithm. CP is a powerful framework to model and solve combinatorial problems. A CP model consists of variables, variable domains, and constraints, as well as objective function (if required), in which the constraints express the relation between the variables. The core concept in CP is *constraint propagation*. Constraint propagation performs reasoning on a subset of variables, variable domains, and constraints to infer more restrictive variable domains, such that the restricted domains still contain all solutions to the problem. CP combines constraint propagation with search procedure to find a local or global optimum (using branch-and-bound search space exploration) to an optimization problem.

The contributions of this paper are as follows.

1) We present a novel data collection scheme (ASBP) that utilizes highly correlated spatio-temporal data in the network and uses BP to reconstruct the missing data due to packet losses and the sensor selection strategy.
2) We formulate the active sensor selection optimization problem, and propose two approaches, namely CP and a heuristic-based greedy algorithm to solve the problem. The CP approach solves the problem to optimality.
3) We conduct extensive simulation with a real deployment of a sensor network and the collected data to evaluate the impact of our proposed solution (for both CP and heuristic-based algorithm) on the overall energy consumption, data utility, and accuracy (error prediction of the missing data).

This paper is organized as follows. In Section II, we discuss the related work. In Section III, we give the system overview. In Section IV, we describe the formulation of our optimization problem on sensor selection, and we solve it using two approaches (CP and heuristic-based greedy algorithm). In Section VI, we conduct simulations to evaluate our solutions based on a real deployment of a WSN. Finally, we summarize and conclude this paper in Section VII.

II. Related Work

The information industry benefits greatly from the technological advancements brought by the IoT [13], [14]. The IoT creates a bridge between many available and recent technologies, such as WSNs, cloud computing, and information sensing [14]–[16]. In monitoring and data acquisition IoT-based systems, it is necessary to collect data effectively and efficiently [14], [15], [17], [18]. The IoT provides a platform for WSNs to connect to Internet and benefit from the power of cloud computing and data fusion. Therefore, it is necessary to study data collection schemes that can seamlessly integrate with the cloud and IoT systems. Data collection has been widely studied for stationary WSNs. Gnawali *et al.* [19] present the state-of-the-art routing protocol for a sensor network where the nodes are forwarding data directly to a sink. They consider stationary WSNs that have static routes from the wireless sensors to the sink. Madden *et al.* [20] introduced a distributed query processing paradigm called acquisitional query processing (ACQP) for sensor network data collection. The goal was to ensure a flexible tasking of motes via a relational query interface, while providing lifetime constraints, data prioritisation, event batching, and rate adaptation.

Prediction-based energy-efficient approaches aim at predicting the data to minimize the number of transmissions. Chou *et al.* [21] proposed a distributed compression based on source coding, which highly relies on the correlation of the data, and it compresses the sensor readings with respect to the sensor past readings, and the reading measured by the other sensor nodes. They used adaptive prediction to track the correlation of the data, which is used to estimate the number of bits needed in source coding for data compression. Recent work in WSN addressed the use of compressive sensing [11]. The authors use compressive sensing to exploit the temporal stability, spatial correlation, and the low-rank structure of the EM. They propose an environmental space–time-improved compressive sensing (ESTI-CS) algorithm to improve the missing data estimation. Although compressive sensing achieved good accuracy on the estimation of the missing data, it does only consider implicit spatio-temporal correlation in the data. Furthermore, compressive sensing approaches rely on the construction of a data matrix and thus require the synchronization of the sensors on the data collection. However, in our work, we present a BP approach for the prediction of missing data, where the spatio-temporal correlation is explicitly enforced and the inference is performed online and iteratively as the data are

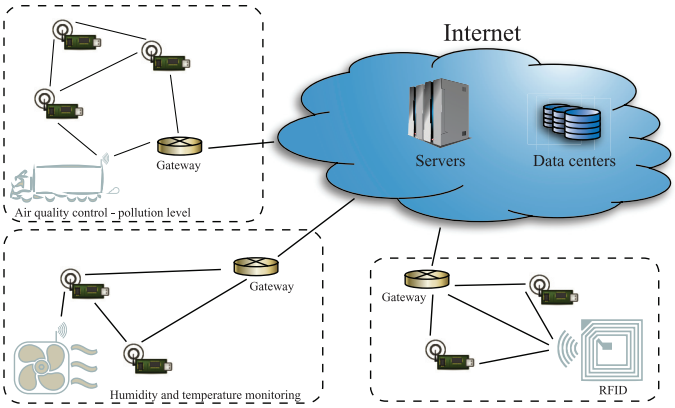


Fig. 1. Network architecture, where the nodes in an IoT application forward the data to the cloud. The servers perform node coordination to improve data quality and save energy, while the data centers stores the collected data as the data fusion and the data loss prediction is performed.

received at the base station. In addition to the above, to the best of our knowledge, there has been no work addressing a CP approach for energy-efficient sensor selection with dynamic routing, while considering the link quality and correlation of the data.

III. System Overview

A. Network Model

In our IoT application, stationary sensor nodes collect environmental data, such as temperature, humidity, light intensity, and noise level. Fig. 1 shows the network architecture of our data collection in IoT applications. We support heterogenous networks, where data can be collected from various devices. The network supports multihop routing and the gateways collect the data and forward the data to the cloud, where the data fusion is performed to further analyze the data, predict missing data, and store the data in the data centers. The computation power of the servers in the cloud is used to improve data quality and save energy of the sensor nodes using our ASBP protocol (to be discussed further in Section III-B). The sensor nodes periodically sample data, which is forwarded to the cloud using a multihop routing protocol (the ACQP system in TinyDB [20], or the collection tree protocol [19]). In this work, we use the real data collected at the Intel Berkeley Research Lab [22]. Fig. 2 shows the map of the Intel Berkeley Research Lab, and the location of the deployed sensor nodes, which are marked with hexagon shapes, and the sensor id. The link thickness between the sensor nodes represents the value of the link quality aggregated throughout the experiment.

The data are collected at the cloud using the gateways associated with different applications of IoT. The gateway only relays the data to the servers in the cloud, and it is at least aware of the routing tables of the sensor nodes. In this paper, we refer to the gateway and the base station as the same entity; however, the actual computations (the CP solver and greedy algorithm in Sections IV-A and IV-B) are performed on the cloud, and all coordinations are relayed by the gateway.

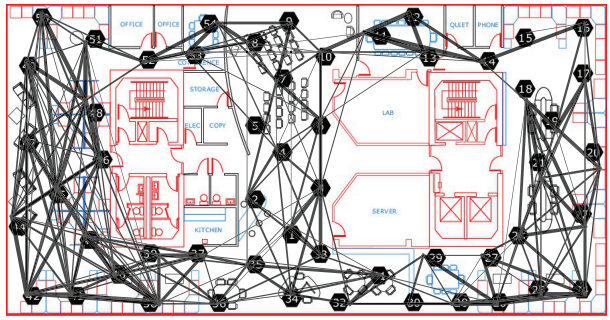


Fig. 2. Map of the Intel Berkeley Research Lab, with the hexagon-shaped nodes indicating the locations and the ids of the sensor nodes, which are deployed to monitor temperature, humidity, and light intensity. The value of the aggregated link quality is represented with the thickness of the link between the sensor nodes.

B. Protocol Design

In our setup, the sensor nodes collect and report the data periodically (typically every 30 s). Our protocol operates in several rounds (a *round* is a time interval where the network repeats the same behavior), and each round includes two phases. The first phase is used to collect the minimum required information, which is used in the second phase to improve energy-efficiency, energy load balancing, and the data quality. The two phases in each round are as follows.

Phase 1: Phase one begins as all sensor nodes become active, and starts collecting and forwarding a fixed number of quantized data to the base station (typically 20 sensor readings). Throughout this phase, the routing protocol estimates the link quality for the shortest routes between the sensor nodes and the base station. The base station then computes the correlation coefficient matrix from the sensor data, and also uses the routing tables to compute all the shortest paths from the sensor nodes to the base station. These data (link quality, correlation, and shortest routes) are then used as an input to solve our sensor selection optimization problem (further explained in Section IV) and select a subset of sensor nodes to be active during the second phase. The active sensor nodes are the only sensor nodes in the network that are participating in the data collection and relaying the data to the base station. The sensor selection problem is solved using either CP or a heuristic-based greedy algorithm to select a set of active sensor nodes, such that it maximizes the spatio-temporal correlation with the inactive sensor nodes, while considering link quality and the dynamic routing.

Phase 2: The base station broadcast, a message that informs a subset of the sensor nodes to become inactive (sleep mode with no radio activity) for a given period of time (typically 2 h). In this phase, the base station performs the BP algorithm [8], [9] to infer incrementally the missing data due to the inactive sensor nodes and packet losses (further explained in Section V). BP captures the high spatio-temporal correlation in the data using a graphical model, which is taken into account in modeling our sensor selection optimization problem. As the second phase is completed, the base station continues to use BP during the first phase of the next round. This allows us to compare the inference results during the first phase with the ground truth, and to

compute the error in prediction. This error is then used by our protocol to give feedback (on the minimum number of selected sensor nodes) to the sensor selection optimization problem of the next round. This allows a dynamic control over the accuracy of the data prediction in phase two. Throughout this paper, we say ASBP to refer to the protocol design above.

IV. Problem Formulation

We present our CP model for the sensor selection problem, followed by our heuristic-based greedy algorithm. The CP model finds a global optimum solution to the problem, whereas our heuristic-based algorithm finds a good quality local optimum solution. The CP model selects the routes for packet relays dynamically. Dynamic routing is essential for networks with multiple shortest paths to the base station, large varieties in the link quality, and energy-efficiency concerns. However, a heuristic-based greedy algorithm with good quality solutions is well suited for networks where the sensor selection problem cannot be solved in a centralized way, and data accuracy is of less concern.

A. CP Model With Dynamic Routing

As we mentioned in Section III-B, throughout phase one of ASBP, the data are collected in the EM, which is used to compute the correlation coefficient matrix. We also estimate the link quality from the packet reception rate during phase one. We then have the following constants in our sensor selection model.

1) Let S be the set of WSN sensor nodes, with $|S| = N$.
2) Let $L[s_1, s_2]$ be the link quality between neighbor sensor nodes s_1 and s_2, indicating the probability of receiving a packet sent from s_1 to s_2, with $s_1, s_2 \in S$. If s_1 is not the direct neighbor of s_2, then $L[s_1, s_2] = 0$.
3) Let $B[s]$ be the link quality between the base station and a direct neighbor sensor node s, and otherwise $B[s] = 0$.
4) Let $C[s_1, s_2]$ be the absolute value of the correlation of the data between sensor nodes s_1 and s_2, with $C[s_1, s_2] \in [0, 1]$.
5) Let $P[s]$ be the set of all shortest paths from the sensor s to the base station, where a path $p \in P[s]$ of length n is denoted by $p : \langle (s_1, s_2), (s_2, s_3), \cdots, (s_{n-1}, s_n) \rangle$ with $s_1 = s$ and s_n is directly linked to the base station.
6) Let $E[s]$ be the residual energy of the sensor s at the end of the first phase in ASBP protocol.

Let $x[s]$ be a Boolean variable with value 1 if the sensor node s is selected for the data collection, and 0 otherwise. Let $q[s]$ represent the maximum achievable path quality among all possible shortest paths from sensor s to the base station, in a solution to the sensor selection problem.

We require the maximization of the path quality

$$\texttt{maximize} \quad \sum_{s \in S} x[s] \cdot q[s]. \tag{1}$$

A second objective is to minimize the correlation of the data between the selected sensors. This objective implies that data from the inactive sensors are more likely to have a high correlation with the enabled sensors, hence improving the accuracy of the missing data construction

$$\texttt{minimize} \quad \sum_{\substack{s, s' \in S, \\ s \neq s'}} x[s] \cdot x[s'] \cdot C[s, s']. \tag{2}$$

We define the data utility $u[s]$ to be the weighted linear sum of the two objective terms in (1) and (2) for sensor s

$$\forall s \in S, \quad u[s] = \omega_1 \cdot x[s] \cdot q[s] - \omega_2 \cdot \sum_{\substack{s' \in S, \\ s' \neq s}} x[s] \cdot x[s'] \cdot C[s, s'] \tag{3}$$

where ω_1 and ω_2 are non-negative weight coefficients used to normalize and allow preference adjustment between the path quality and the aggregated correlation of the data for sensor s versus all the other sensors in the network.

The combined objective considering the data utility $u[s]$ and the residual energy $E[s]$ of the sensor nodes becomes

$$\texttt{maximize} \quad \sum_{s \in S} E[s]^{\alpha} \cdot u[s] \tag{4}$$

where α is a parameter to adjust the weight of the energy coefficient on the data utility (typically α is set to 0.5).

The *path quality* constraint enforces that the path quality $q[s]$ from a selected sensor to the base station must exceed a given threshold τ

$$\forall s \in S, \quad q[s] \geq x[s] \cdot \tau \tag{5}$$

where the threshold τ is adjusted according to the link quality to provide a consistent packet delivery on a path to the base station (typically $\tau \in [0.3, 0.7]$).

The *routing constraint* enforces that a path with higher quality is preferred in selecting the active sensors and all sensors on such a path must be active. For example, Fig. 3 shows two paths p_1 and p_2 from sensor node 1 to the base station B. We assume that the link quality between sensor nodes on a path to the base station is an independent random variable. Therefore, path quality is the joint probability of the link quality probabilities along a path to the base station

$$\begin{aligned} q_{p_1}[1] &= L[1,5] \cdot L[5,6] \cdot L[6,4] \cdot B[4] \\ &= 0.78 \cdot 0.88 \cdot 0.65 \cdot 0.91 \\ &= 0.406 \\ q_{p_2}[1] &= L[1,5] \cdot L[5,6] \cdot L[6,4] \cdot B[4] \\ &= 0.86 \cdot 0.75 \cdot 0.66 \cdot 0.91 \\ &= 0.387 \end{aligned}$$

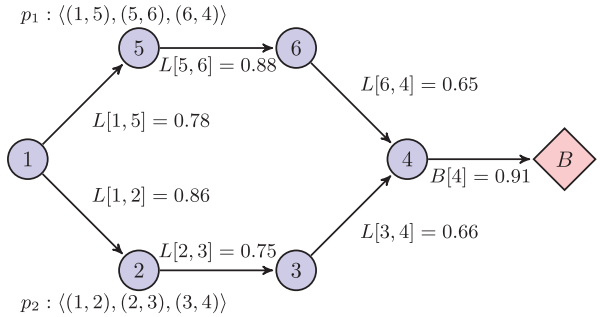


Fig. 3. Sensor node 1 sending data to the base station B.

where $q_{p_i}[s]$ denotes the path quality of the path p_i originated at sensor s. The path p_1 has a higher path quality ($q_{p_1}[1] > q_{p_2}[1]$). Hence, when maximizing the path quality (1), the path p_1 is preferred for routing the data, and to enforce that all sensors on the path must be selected, the path quality $q1$ is constructed as follows:

$$q[1] = \max\left(x[5]\cdot x[6]\cdot x[4]\cdot q_{p_1}[1], x[2]\cdot x[3]\cdot x[4]\cdot q_{p_2}[1]\right). \tag{6}$$

Assuming that sensor 1 is selected ($x[1] = 1$), the path quality constraint (5) requires that $q[1] \geq \tau > 0$, and according to (6), all sensors on either path p_1 or p_2 must be active ($x[5] = x[6] = x[4] = 1$ or $x[2] = x[3] = x[4] = 1$). Note that the routing constraint (6) must be enforced only if the origin sensor 1 is selected ($x[1] = 1$), and otherwise the value of the path link should not be included in the objective function (4). Therefore, the nonlinear term $x[s]\cdot q[s]$ is used in the construction of data utility (3). In general, the routing constraint becomes $\forall s \in S$

$$q[s] = \max_{p\in P[s]}\left(B[n_p]\cdot \prod_{(s',s'')\in p}\left(x[s'']\cdot L[s',s'']\right)\right) \tag{7}$$

where n_p is the last sensor on the path p, and s', s'' are two adjacent sensors on the path p. For example, in Fig. 3 for the paths p_1 and p_2, we have $n_{p_1} = n_{p_2} = 4$. The n-ary constraint $\max$ is essential in our CP implementation of the routing constraints (7).

The *active sensor* constraint enforces that the minimum number of active sensors is at least μ

$$\sum_{s\in S} x[s] \geq \mu \tag{8}$$

where μ provides a tradeoff between energy efficiency and data quality (BP inference error).

In summary, our CP model for the sensor selection problem is defined as follows.

Inputs:

1) L: link quality estimations;
2) B: base station link quality estimations;
3) C: correlation coefficient matrix;
4) P: shortest routes to the base station;
5) E: residual energy.

Outputs:

1) x: selected sensors with $x[s] = 1$ iff sensor s is selected for data collection and $x[s] = 0$ otherwise;
2) $u[s]$: data utility of sensor node s;
3) $q[s]$: path quality achieved in the routing of data from sensor node s to the base station.

Objective:

$$\texttt{maximize} \quad \sum_{s\in S} E[s]^{\alpha}\cdot u[s].$$

Such that

$$\forall s\in S,\quad u[s] = \omega_1\cdot x[s]\cdot q[s] - \omega_2\cdot \sum_{\substack{s'\in S,\\ s'\neq s}} x[s]\cdot x[s']\cdot C[s,s']$$

$$\forall s\in S \quad q[s] = \max_{p\in P[s]}\left(B[n_p]\cdot \prod_{(s',s'')\in p}\left(x[s'']\cdot L[s',s'']\right)\right)$$

$$\forall s\in S,\quad q[s] \geq x[s]\cdot\tau$$

$$\sum_{s\in S} x[s] \geq \mu.$$

This CP model is directly expressed and solved in our chosen CP solver without further transformation to the formulation. For our CP implementation of this model, we derive implied constraints from the routing constraints (7), to reduce the search effort needed to solve the problem. We observe that some sensor nodes are often shared along the shortest paths from the origin sensor node s to the base station. For example, in Fig. 3, sensor node 4 is shared by both paths p_1 and p_2. If sensor node 1 is selected, it implies that sensor 4 must be also selected regardless of which path is used in forwarding the data to the base station. We incorporate these implied constraints in our model to help improve the performance of the solver

$$\forall s\in S,\quad (x[s] = 1) \implies \left(\wedge_{s'\in P_{\cap}[s]} x[s']\right) = 1 \tag{9}$$

where $P_{\cap}[s]$ is the intersection set of all sensor nodes on the paths from s to the base station. The implication (9) states that if the sensor node s is selected ($x[s] = 1$), then the conjunction of all shared sensor nodes on the paths from s to the base station must be 1, enforcing that all the shared sensor nodes are part of the solution ($x[s'] = 1,\ s' \in P_{\cap}[s]$).

Our custom search procedure branches on the $x[s]$ decision variables. It selects a sensor with the largest mid-value in the domain of the data utility $u[s]$. The mid-value is often a better choice when the domain range is large, which is the case at the beginning of the search. The search procedure breaks ties by selecting the closest sensor to the base station with hop-count as the metric. We then set the value of $x[s]$ to 1 on the left branch and 0 on the right branch.

B. Heuristic-Based Greedy Algorithm

Instead of using CP to solve the sensor selection problem optimally as described above, we also designed a heuristic-based algorithm built upon a simple greedy search strategy. The intuition behind is that we should remove a sensor if: 1) the data from the sensor are strongly correlated with the others, meaning that we can predict fairly accurately the reading from that sensor; 2) the sensor is already overused, meaning that the sensor has a low energy; and 3) the sensor has a poor connection to the base station, meaning that the data transmission from that sensor has a high risk to fail. Thus, we do a greedy selection by taking all three aspects into consideration and remove sensors one by one until we are left with the required number of sensors. While simple, the heuristic algorithm may only find a local optimum to the sensor selection problem, which might be far from the global optimum.

Our heuristic algorithm returns a set $idSelected$ of sensor nodes to be selected during the phase two of each round in ASBP protocol. The algorithm takes the constant set of sensor nodes S, link quality L, base station link quality B, correlation

Algorithm 1. The heuristic-based greedy algorithm with dynamic routing

input: S, L, B, C, E, μ, τ
output: $idSelected$
1 $idSelected \leftarrow S$
2 $q \leftarrow \texttt{BestShortestPath}(L, B)$
3 $idNonReachable \leftarrow \{s \in S | q[s] < \tau\}$
4 $idSelected \leftarrow idSelected - idNonReachable$
5 $\texttt{SetZero}(L, idNonReachable)$
6 $\texttt{SetZero}(B, idNonReachable)$
7 $\texttt{SetZero}(C, idNonReachable)$
8 $idFeasible \leftarrow idSelected$
9 **while** $idFeasible \neq \emptyset \ \wedge |idSelected| > \mu$ **do**
10 $L' \leftarrow L$
11 $B' \leftarrow B$
12 $\{idMin\} \leftarrow \min \left(\underset{s \in idFeasible}{\arg\min} \ (E[s]^{\alpha} \cdot u[s]) \right)$
13 $\texttt{SetZero}(L', \{idMin\})$
14 $\texttt{SetZero}(B', \{idMin\})$
15 $q \leftarrow \texttt{BestShortestPath}(L', B')$
16 $idNonReachable \leftarrow idSelected \cap \{s \in S | q[s] < \tau\}$
17 $idPotential \leftarrow idSelected - idNonReachable$
18 **if** $|idPotential| < \mu$ **then**
19 $idFeasible \leftarrow idFeasible - \{idMin\}$
20 **continue**
21 $idSelected \leftarrow idPotential$
22 $idFeasible \leftarrow idPotential$
23 $\texttt{SetZero}(L, idNonReachable)$
24 $\texttt{SetZero}(B, idNonReachable)$
25 $\texttt{SetZero}(C, idNonReachable)$
26 **return** $idSelected$

C, and initial energy E as an input, in addition to the parameters μ and τ representing the minimum threshold on the number of selected sensors and the link quality, respectively. Our heuristic-based algorithm is listed in Algorithm 1. In our algorithm, the identifier of a variable is written with *italic* font, and the identifier of a function is written with `typewriter` font. Here, the variables are imperative programming variables as opposed to the CP decision variables of Section IV-A.

The heuristic algorithm creates a set of selected sensors $idSelected$ (line 1), and initialize it with all the possible sensor ids. The function `BestShortestPath` (line 2) takes the link quality matrix L and base station link quality array B as an input, and returns an array q of path quality values for the shortest path from each sensor node to the base station. The implementation of `BestShortestPath` is trivial, as it uses Dijkstra's algorithm [23] to compute the shortest paths, while respecting the *path quality* constraint (5).

The heuristic algorithm maintains a set $idNonReachable$ of sensor nodes that are not able to reach the base station due to the violation of the *path quality* constraints (5) (line 3). Before entering the main loop of the algorithm, any sensor nodes in the set $idNonReachable$ are removed from the set of selected sensor nodes (line 4), and the values of link quality, base station link quality, and correlation for those sensor nodes in $idNonReachable$ are set to 0 from the corresponding data using the function `SetZero` (lines 5–7). The function $\texttt{SetZero}(A, Ids)$ takes an $n \times n$ matrix A, and a set of indices Ids, and for each index i in Ids sets the value of every possible pair of (i, j) $1 \leq j \leq n$ in A to zero ($A(i, j) = 0 \wedge A(j, i) = 0, \ 1 \leq j \leq n$), and if A is a one dimensional array, then it only sets $A(i) = 0$. In other words, `SetZero` reflects the unreachability of the sensor nodes in $idNonReachable$ into the network data structures (link quality and correlation).

The main loop of the algorithm (line 9) iteratively selects a sensor node that contributes the least value to the objective (4) (equivalent to a sensor node with the lowest data utility weighted by the initial energy), and performs a lookahead move (lines 10–20) to detect if removing this sensor node violates any of the constraints. The set $idFeasible$ of feasible sensor nodes is initialized with the set of selected sensor nodes $idSelected$ (line 8). The set $idFeasible$ is used to keep the track of the sensor nodes that are potentially removable from the set of selected sensor nodes $idSelected$. A lookahead move is performed, by first creating copies L' and B' from L and B, respectively (lines 10–11). We then select a least contributing sensor $\{idMin\}$ that minimizes the value of the objective function (lines 12). To perform the lookahead move, the link quality data for the sensor $\{idMin\}$ is set to zero (lines 13–14), and then the path quality q is updated (line 15) to discover the nonreachable sensor nodes $idNonReachable$ (line 16).

If removing the nonreachable sensor nodes in the set $idNonReachable$ from the set of selected sensor nodes $idSelected$ (line 17) that causes the violation of the *active sensor* constraint (line 18), then the sensor node $\{idMin\}$ is removed from the set $idFeasible$ of feasible sensor nodes (line 19), and we skip to the next iteration (line 20). If the lookahead move does not violate the *active sensor* constraint, then we replace the set of selected sensors $idSelected$ and the set of feasible sensor nodes $idFeasible$ with the potential sensor set $idPotential$, and we set the values of link quality and correlation for the nonreachable sensor nodes $idNonReachable$ to zero using the function `SetZero` (lines 21–25). The algorithm ends if there are no more feasible sensor nodes ($idFeasible = \emptyset$) or the *active sensor* constraint is violated.

V. Bayesian Inference and Data Quantization

This section describes how to use BP to infer the missing data because of the inactive sensor nodes and the data transmission losses of the active sensor nodes throughout the second phase of our ASBP protocol.

A. Introduction to BP

BP is a classic algorithm for performing inference on graphical models [8], [9]. In general, it assumes that some observations are made and the task is to infer the underlying events behind these observations. Denote y_i the observation at node i and x_i the underlying event, $i = 1, \ldots, N$. For the application of IoT, y_i is the reading of sensor i about some phenomenon that is being monitored, such as the temperature, and x_i is

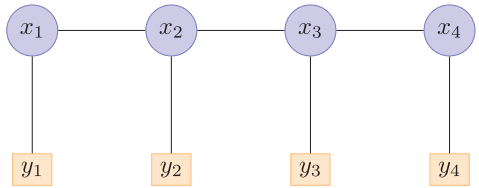


Fig. 4. Example of a graphical model.

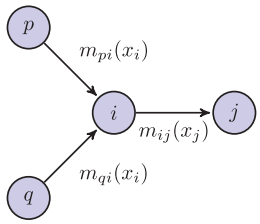


Fig. 5. Graphical depiction of message passing from nodes p and q to the node i in BP The updated message $m_{ij}(x_j)$ is then sent to the node j.

the true reading of the phenomenon. Clearly, there are some statistical dependencies between y_i and x_i, encoded in a so-called evidence function $\phi_i(x_i, y_i)$. Very often, we consider the observation y_i to be fixed and write $\phi_i(x_i)$ as a short-hand of $\phi_i(x_i, y_i)$. Furthermore, there are also statistical dependencies between the several underlying events x_i, encoded in a so-called potential function $\phi_{ij}(x_i, x_j)$. In IoT, the potential function captures spatial correlations between the readings at nearby sensors.

Given the above notation, the inference of the x_i can be formulated as the maximization of the following belief function:

$$b(\{x_i\}_{i=1}^N) = \prod_{ij} \phi_{ij}(x_i, x_j) \prod_i \phi_i(x_i).$$

A graphical depiction of this model is shown in Fig. 4. The rectangles are the observation nodes y_i and the circles represent the underlying events x_i. The potential functions are associated with the links between x_i and the evidence functions are associated with the links between y_i and x_i.

BP performs inference by passing messages between nodes in the graph. The message from i to j is defined as

$$m_{ij}(x_j) = \sum_{x_i} \phi_i(x_i)\phi_{ij}(x_i, x_j) \prod_{k \in N(i), k \neq j} m_{ki}(x_i)$$

where $N(i)$ denotes the neighbors of node i. The message essentially integrates all messages from the neighbors of i, except j, as well as the local evidence seen at i. Intuitively, such a message models how likely it is at node i that node j will be in the state of x_j when node i is in the state x_i. Thus, BP performs message passing between nodes until reached convergence, and the inference is done by maximizing the belief at each node, which is to gather all incoming messages and the local belief, i.e.,

$$b_i(x_i) = \phi_i(x_i) \prod_{j \in N(i)} m_{ji}(x_i).$$

The message passing process in BP is illustrated in Fig. 5.

BP is well established in both theory and practice. For example, while it is known that BP is only guaranteed to converge on tree graphs, loopy BP has been shown to work well in most cases for graphs with loops [24]. In addition, there are two general BP variations which are sum-product and max-product BP, respectively [25]. The latter is adopted in this paper because of its efficiency.

B. BP for Inference on IoT

In using BP for inferring the missing data in IoT, we need to construct a graph to model the correlations between sensor readings. There are two types of correlations in sensor network.

1) Spatial correlation: Data from different sensors may be correlated with each other. Note that we do not assume that strong correlations always exist between data from nearby sensors. Instead, we compute the correlation coefficients between each pair of sensor nodes from the observed data. We claim spatial correlations only when we see large correlation coefficients, regardless of the spatial distance between two sensors.
2) Temporal correlation: Data from the same sensor may be correlated over time. Here, we simply assume that the sensor reading at time t is strongly correlated with that at time $t-1$.

Thus, we built our graph as illustrated in Fig. 6 where x_i^t denotes the true reading of sensor i at time t. The link between x_i^t and x_i^{t-1} represents the temporal correlations, with a temporal potential function defined as

$$\phi_i^t(x_i^t, x_i^{t-1}) = \exp\left(-\frac{(x_i^t - x_i^{t-1})^2}{\sigma_i^2}\right).$$

Similarly, the link between x_i^t and x_j^t represents the spatial correlations, with a spatial potential function associated and defined as

$$\phi_{ij}^s(x_i^t, x_i^{t-1}) = \exp\left(-\frac{(x_i^t - x_j^t)^2}{\sigma_{ij}^2}\right).$$

Note that the noisy sensor reading y_i^t is omitted from the graph for the purpose of simplification, and the evidence function associated with the link between x_i^t and y_i^t is defined as

$$\phi_i^e(x_i^t, y_i^t) = \exp\left(-\frac{(x_i^t - y_i^t)^2}{\sigma_i^2}\right).$$

y_i^t can be missing for two reasons: either sensor i is in the sleep mode or the packet failed to reach the base station. When it is missing, we turn the evidence function into a constant, i.e., $\phi_i^e(x_i^t, y_i^t) = 1$, for all possible values of x_i^t. Such a constant evidence function essentially treats everything as equally as possible. Intuitively, BP handles missing sensor readings by reasoning from the past data and the sensor nodes with correlated data. Note that σ_i and σ_{ij} are parameters that can be learned from some training data [26].

In comparison with approaches such as the *CS*-based approach in [11], BP based on the graph in Fig. 6 is advantageous for several reasons.

1) BP captures the spatial and temporal correlations between sensors explicitly via a graphical model which is updated over time. For example, we can disconnect the sensor

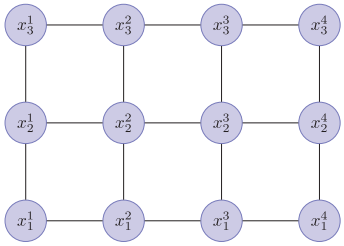


Fig. 6. Graphical model built for WSNs.

nodes when the correlation coefficients drop below some threshold.

2) BP allows the incremental inference that infers the missing data at time t from the available data at just time t and $t-1$. In contrast, the CS-based approach in [11] takes as an input data matrix with missing entries, and thus can only perform inference in a batch mode for a time interval.

We will demonstrate these advantages and the inference accuracy in Section VI.

C. Data Quantization

Quantization is a classic technique in signal processing that has been widely used for data compression [27]. Quantization of network data saves storage as it encodes the data into fewer bits. It requires fewer number of transmissions and smaller packet size. In many applications, a quantized measure is informative enough to represent aspects of the network. For example, many heating, ventilation, and air conditioning (HVAC) sensors only react if temperature or humidity falls within certain thresholds. In summary, quantized measures are less fine-grained and lossy; however, there are many advantages in using a quantized measure.

1) A quantized measure is informative enough for describing the correlation between the data.
2) A quantized measure can be encoded into a few bits, saving storage and transmission costs.
3) A quantized measure is coarse and thus cheaper to obtain. It is also stable and highly adjustable to match the needs of the network application.

Let the metric to be quantized take on values in the range $[r_{\min}, r_{\max}]$, and values outside this interval are mapped either to $r_{\min}$ or $r_{\max}$. The quantization is done by partitioning the interval into R bins using $R-1$ thresholds, denoted by $\tau = \{\tau_1, \ldots, \tau_{R-1}\}$. Each bin is represented by a value within the range of the bin, e.g., the centroid point of the bin's range. Let the value b_i represent the ith bin. A look-up table is used to map the metric value to b_i according to the bin threshold

$$Q(x) = b_i, \quad \text{if } \tau_{i-1} < x \leq \tau_i,\ i = 1, \ldots, R. \tag{10}$$

where $\tau_0 = r_{\min}$ and $\tau_R = r_{\max}$. The bin index values $\{b_1, \ldots, b_R\}$ are stored in a codebook, and a metric value can then be represented by a bin index that is encoded into few bits. For example, Fig. 7 shows six data $x_1, \ldots, x_n$ quantized into four bins with 2-bit binary indices $b_1 = (00)_2 = 0, \ldots, b_4 = (11)_2 = 3$ according to (10).

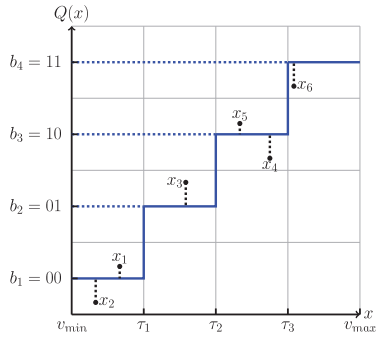


Fig. 7. 2-bit uniform quantization on the data $x_1, \ldots, x_n$ that partitions the interval $\{r_{\min}, r_{\max}\}$ into four equal bins using $\tau = \{\tau_1, \tau_2, \tau_3\}$. Each bin is represented by the centroid point of the bin, which is stored in a codebook. A metric value is then mapped into a bin index, encoded into 2 bits.

The length of each partition $\tau_i - \tau_{i-1}$ is either uniform with $\tau_i - \tau_{i-1} = \frac{v_{\max} - v_{\min}}{R}$ or nonuniform. In general, the thresholds τ are chosen according to the requirements of the application, adaptively adjusted, or learned from a set of training data. For example, consider an indoor temperature monitoring, where the temperature varies at most between 0° and 50°. Given 0.2° temperature accuracy requirement of the application, the minimum number of quantization level is $(v_{\max} - v_{\min})/0.2 = 50/0.2 = 250$, which implies that at least 8-bit quantization resolution ($2^8 = 256$ bins) is necessary to satisfy the requirement of the application.

As we mentioned in this section, the data quantization is lossy with the error defined as

$$\varepsilon(x) = x - code(b_i), \quad \text{for } Q(x) = b_i$$

where $code$ maps the code b_i to the metric value of the data x (typically, the centroid point of the bin). The error is upper bounded by the bin length, given by

$$\varepsilon(x) < \tau_i - \tau_{i-1}.$$

The quantization error is inversely proportional to R, whereby a smaller R leads to a larger $\varepsilon(x)$. When R is as large as $v_{\max} - v_{\min}$, the quantization becomes equivalent to the rounding of the real value, which is almost lossless.

VI. Experiments

A. Experimental Setup

We experiment with the real data collected from 54 sensor nodes deployed in the Intel Berkeley Research Laboratory [22]. The data are collected by a base station, and includes temperature, humidity, light intensity, and voltage values once every 30 s, throughout a time span of 36 days. The data set also includes aggregated connectivity data, representing the link quality between any two sensor nodes, and between sensor nodes and the base station. In our simulations of the ASBP protocol, we selected a time interval of 10 h, consisting of 5 rounds of 2 h, such that at least 30% of data are transmitted successfully to the base station.

We apply a uniform quantization on the temperature data between 10th and 90th percentile into 256 bins, where each bin

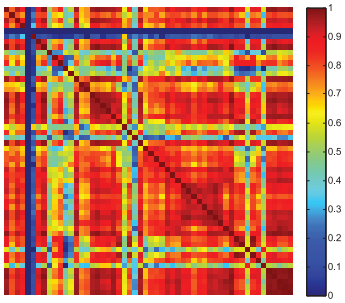


Fig. 8. Correlation coefficient matrix represented with jet color map, with absolute values varying in range $[0, 1]$.

is represented with an 8-bit value in the codebook. The values outside the interval are mapped to the minimum and maximum of the interval accordingly.

Fig. 8 demonstrates the absolute values for the correlation coefficient matrix using the jet color map. We expect that since the data are highly correlated, the uniform quantized data are also highly correlated. Our experiments show that 8-bit quantization resolution introduces at most 15% error in the data correlation. However, it does not affect the BP prediction results, as the data are already quantized when received at the base station.

In our energy consumption evaluations, we consider 14-mA transmission cost, as reported for the Mica2Dot mote [28], used in the Intel Berkeley Research Laboratory deployment.

In our simulations, each round is 2 h, where phase one of a round ends if at least 20 data readings are collected at the base station from all the sensor nodes. The weights ω_1 and ω_2 in data utility (3) are chosen to normalize the path quality and correlation. We expect that at least μ sensor nodes are selected; hence, the path quality is scaled by the minimum number μ (8) of sensor nodes ($\omega_1 = \mu$), because the sum of the correlation is at least μ we set $\omega_2 = 1$. The threshold τ of (5) is set to 0.7. The base station then solves the sensor selection optimization problem and initiates the second phase of the ASBP protocol.

B. Results and Analysis

We evaluate the performance of our ASBP in terms of data utility, energy efficiency, and data prediction accuracy. We compare the data prediction error of the results of our CP model, heuristic-based algorithm, and a random sensor selection. On the inference accuracy, we compare with the CS-based approach in [11] which we consider as the state-of-the-art. Our simulation of the ASBP protocol is implemented in C++, and the CP model is implemented using the CP solver *Gecode* [29] (revision 4.2.1), and runs under Mac OS X 10.9.2 64 bit on an Intel Core i5 2.6 GHz with 3 MB L2 cache and 8 GB RAM.

Fig. 9(a) and (b) compares the total data utility and energy consumption achieved in one round by the ASBP protocol using CP, our heuristic-based algorithm, and random sensor selection, with a minimum of 30% and 70% for the base station link quality, respectively. For each result, we vary the parameter μ in (8) to control the total number of selected sensor nodes for data collection. The increase in the minimum base station link quality to 70% affects the routing of the data in the multihop data collection. It increases the size of the data collection to five hops, which requires the sensor nodes closer to the base station to relay also the data for the nodes further away. Hence, the path quality $q[s]$ is decreased, and the total data utility is reduced.

In our results, the CP sensor selection achieves the optimum data utility, and the greedy heuristic-based algorithm manages to find a satisfactory local optimum. The results show that the general traditional random approach does perform very poorly compared to the global optimum. The results for the random sensor selection are computed by taking the mean of the data utility and energy consumption for ten random sensor selections. In all cases, the solution of the sensor selection problem for CP and the heuristic-based algorithm were found in less than 1 min. We observe that the data utility increases up to 25 selected nodes and then decreases. This is because of the trade-off between the path quality and the correlation. As the number of selected sensors increases, the sum of the data correlation between a selected sensor node and all the other sensor nodes becomes a larger factor in the data utility term (3) compare to the path quality term; hence, the data utility decreases. We conclude that an efficient sensor selection strategy should select 25 sensor nodes to maintain a balance between the path quality and the data correlation.

The heuristic-based strategy in Fig. 9(b) fails to find a solution for more than 30 selected sensor nodes, because our requirement for reaching the base station is limited to at least 70% link quality, and without backtracking, the greedy algorithm fails at maintaining a route to the base station for all selected sensor nodes.

The total energy consumption (in terms of the number of transmission for data collection and node coordination) for the data transmissions with both settings 30% and 70% on the minimum base station link quality is shown in Fig. 9(c). The minimum base station link quality is denoted in the legend of the plot. We observe that at the same threshold on the base station link quality, the energy consumption is almost independent of the sensor selection strategy. However, the energy consumption is almost doubled as the base station link quality threshold is increased to 70%, which is due to the additional multihop relay of the data required to reach the base station.

Fig. 10(a) shows the BP results with the CP model, heuristic-based algorithm, and random sensor selection strategies, upon varying the minimum number μ of selected sensor nodes. We first compute the mean square error (MSE) of the predicted data versus the ground truth for each sensor node in the temporal domain. The result is an array of 54 MSE values on the sensor node predicted data. We then plot the mean of the MSE error in Fig. 10(a). The results for the random sensor selection are computed by taking the average of ten runs. The standard deviation (SD) of CP and the heuristic-based algorithm is at most 12%. The CP model with $\mu = 10$ has an average error of about 5%, which indicates that in the temporal domain, in average the prediction of the BP deviates 5% from the ground truth. At the same data point, the SD is about 12%, and increasing the number of selected sensor nodes μ always drops the value of SD. As we expected, the best sensor selection (by CP) achieves the minimum error, whereas the random sensor selection does not

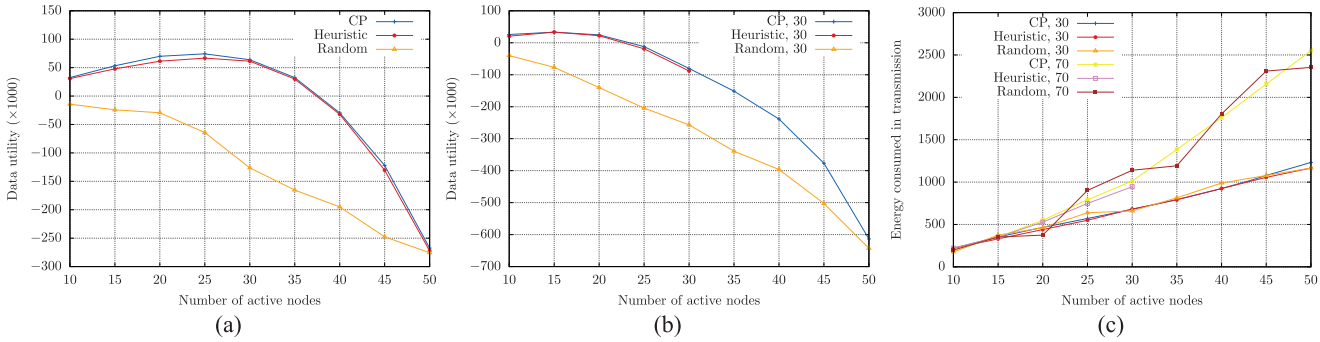


Fig. 9. Data utility and energy consumption for data transmission obtained by simulating the ASBP protocol in one round and solving the sensor selection problem with the CP model, our heuristic-based algorithm, and random sensor selection. Minimum thresholds of: (a) 30% and (b) 70% were used for the base station link quality, upon varying the minimum number μ of selected sensor nodes. (c) Energy consumption.

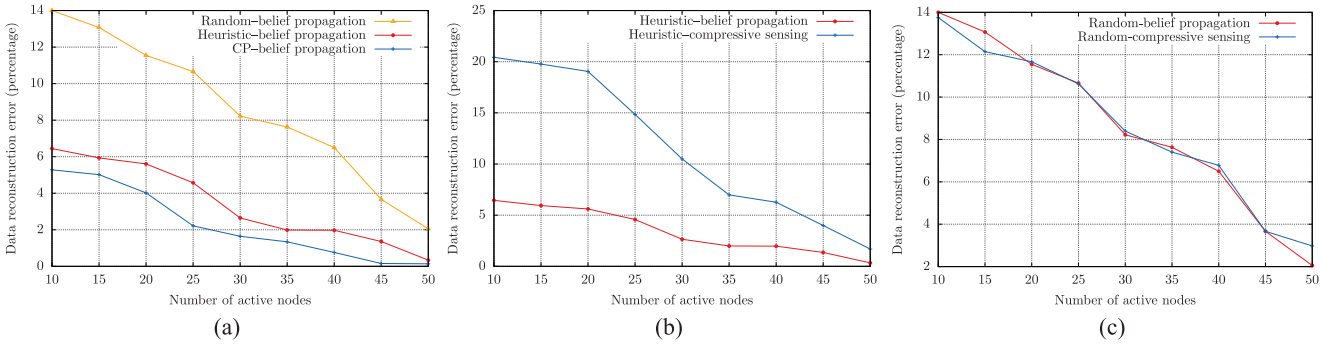


Fig. 10. Prediction MSE of our BP-based approach and the CS-based approach in [11] using the CP model, the heuristic-based algorithm, and the random sensor selection strategies, upon varying the minimum number μ of active nodes. (a) Prediction error of BP with CP, Heuristic, and random node selection. (b) Prediction error of BP versus CS using our heuristic-based node selection algorithm. (c) Prediction error of BP versus CS using random node selection.

consider the correlation of the data, and as a result has a higher prediction error.

The results compared with the energy consumption in Fig. 9(c) show that we can save up to 80% energy by selecting only 10 sensor nodes to be active for the data collection in each round, while maintaining at most the satisfactory average error of 5% with an SD of 12% in the prediction accuracy. In our approach, depending on the application and the required accuracy, we can adjust the selected number of sensor nodes as a tradeoff between the energy consumption and data quality (accuracy of the BP).

On the inference accuracy, we compared our BP-based approach with the CS-based approach in [11]. In particular, [11] modeled the estimation of the lost data as a problem of matrix completion, where an EM matrix is constructed by recording the data reading of a particular sensor at a particular time. The EM matrix is incomplete because some data are lost during transmission and some sensors are inactive, i.e., not selected, during some time periods. By applying the matrix completion techniques developed in CS, the missing data in the EM matrix can also be estimated. While interesting, a drawback of the matrix completion formulation in [11] is that in order to construct the EM matrix, data must be collected in different sensors regularly and in a synchronized way, so that the data in the time dimension are consistent. In contrast, our BP-based approach makes no such assumption and allows the sensors to collect data at irregular frequencies or even randomly. This is possible due to the explicit modeling of the data correlations in time and in space in the potential functions [9].

Fig. 10(b) and (c) shows the comparisons between our BP-based approach and the CS-based approach in [11] using the heuristic-based and random node selection, respectively. It can be seen that on the heuristic-based node selection, BP is strictly better than CS. For example, BP achieves 16% lower prediction error compared to CS when $\mu = 10$. On the random node selection, the two perform similarly. Note that the results on random node selection are the average of ten runs. Such results reveal the advantage of BP that the spatio-temorpal correlations are explicitly encoded in the graph structure and in the potential functions, which leads to the better accuracy in Fig. 10(b). On the other hand, in Fig. 10(c), BP builds the graph and learns the potential functions on randomly selected nodes without considering the correlations, whereas CS assumes the random sampling of the data which hold here. Even in such scenarios, BP still achieves a similar performance as CS.

VII. Conclusion

By exploring cloud computing with the IoT, we present a cloud-based solution that takes into account the link quality and spatio-temporal correlation of data to minimize energy consumption by selecting sensors for sampling and relaying data. We have presented a novel cloud-based ASBP protocol with energy-efficient data collection for the IoT applications.

ASBP solves an optimisation problem to select an optimal set of active sensor nodes that maximizes the data utility and achieves energy load balancing. In our protocol, BP iteratively infers the values of the missing data from the stream of active sensor readings. We have also compared our BP prediction results with the widely used compressive sensing technique [11], and show that our BP algorithm significantly outperforms compressive sensing. We formulate and solve the active sensor selection optimization problem using CP, and compare it with our heuristic-based greedy algorithm.

We have evaluated the performance of our ASBP protocol by extensive simulations using real data collected at the Intel Berkeley Research Lab sensor deployment and their link quality estimates. The simulation results show that our ASBP protocol can greatly improve energy-efficiency up to 80%, with the optimal CP active sensor selection, while maintaining in average 5% error in the BP data inference.

As future work, we plan to extend our ASBP protocol to a fully distributed implementation for real deployment, and compare versus our current optimal results. We are also interested to integrate adaptive sampling rate into our current results, as well as investigating multisink scenarios.

REFERENCES

[1] Cisco. (2011). "The Internet of Things," [Online]. Available: http://share.cisco.com/internet-of-things.html

[2] L. Atzori, A. Iera, and G. Morabito, "The Internet of Things: A survey," *Comput. Netw.*, vol. 54, no. 15, pp. 2787–2805, Oct. 2010.

[3] O. Vermesan *et al.*, "Internet of Things strategic research roadmap," in *Internet of Things-Global Technological and Societal Trends*. Delft, The Netherlands: River Pub., 2011, pp. 9–52.

[4] J. Amaro, F. J. T. E. Ferreira, R. Cortesao, N. Vinagre, and R. Bras, "Low cost wireless sensor network for in-field operation monitoring of induction motors," in *Proc. IEEE Int. Conf. Ind. Technol. (ICIT)*, Mar. 2010, pp. 1044–1049.

[5] S. Madden, M. J. Franklin, J. M. Hellerstein, and W. Hong, "Tag: A tiny aggregation service for ad-hoc sensor networks," *SIGOPS Oper. Syst. Rev.*, vol. 36, no. SI, pp. 131–146, Dec. 2002.

[6] S. Madden, R. Szewczyk, M. Franklin, and D. Culler, "Supporting aggregate queries over ad-hoc wireless sensor networks," in *Proc. 4th IEEE Workshop Mobile Comput. Syst. Appl.*, 2002, pp. 49–58.

[7] J. Pearl, *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. San Mateo, CA, USA: Morgan Kaufmann, 1988.

[8] J. S. Yedidia, W. T. Freeman, and Y. Weiss, "Understanding belief propagation and its generalizations," in *Exploring Artificial Intelligence in the New Millennium*, G. Lakemeyer and B. Nebel, Eds. San Mateo, CA, USA: Morgan Kaufmann, 2003, pp. 239–269.

[9] F. V. Jensen, *Introduction to Bayesian Networks*, 1st ed. Berlin, Germany: Springer-Verlag, 1996.

[10] L. Kong, D. Jiang, and M.-Y. Wu, "Optimizing the spatio-temporal distribution of cyber-physical systems for environment abstraction," in *Proc. IEEE 30th Int. Conf. Distrib. Comput. Syst. (ICDCS)*, Jun. 2010, pp. 179–188.

[11] L. Kong, M. Xia, X.-Y. Liu, M.-Y. Wu, and X. Liu, "Data loss and reconstruction in sensor networks," in *Proc. IEEE INFOCOM*, 2013, pp. 1654–1662.

[12] F. Rossi, P. van Beek, and T. Walsh, Eds., *Handbook of Constraint Programming*. Amsterdam, The Netherlands: Elsevier, 2006.

[13] L. D. Xu, "Enterprise systems: State-of-the-art and future trends," *IEEE Trans. Ind. Informat.*, vol. 7, no. 4, pp. 630–640, Nov. 2011.

[14] J. Zheng, D. Simplot-Ryl, C. Bisdikian, and H. Mouftah, "The Internet of Things," *IEEE Commun. Mag.*, vol. 49, no. 11, pp. 30–31, Nov. 2011.

[15] L. Palopoli, R. Passerone, and T. Rizano, "Scalable offline optimization of industrial wireless sensor networks," *IEEE Trans. Ind. Informat.*, vol. 7, no. 2, pp. 328–339, May 2011.

[16] J. Haupt, W. Bajwa, M. Rabbat, and R. Nowak, "Compressed sensing for networked data," *IEEE Signal Process. Mag.*, vol. 25, no. 2, pp. 92–101, Mar. 2008.

[17] A. Ulusoy, O. Gurbuz, and A. Onat, "Wireless model-based predictive networked control system over cooperative wireless network," *IEEE Trans. Ind. Informat.*, vol. 7, no. 1, pp. 41–51, Feb. 2011.

[18] M. Jongerden, A. Mereacre, H. Bohnenkamp, B. Haverkort, and J. Katoen, "Computing optimal schedules of battery usage in embedded systems," *IEEE Trans. Ind. Informat.*, vol. 6, no. 3, pp. 276–286, Aug. 2010.

[19] O. Gnawali, R. Fonseca, K. Jamieson, D. Moss, and P. Levis, "Collection tree protocol," in *Proc. 7th ACM Conf. Embedded Netw. Sensor Syst.*, 2009, pp. 1–14.

[20] S. R. Madden, M. J. Franklin, J. M. Hellerstein, and W. Hong, "Tinydb: An acquisitional query processing system for sensor networks," *ACM Trans. Database Syst.*, vol. 30, no. 1, pp. 122–173, Mar. 2005.

[21] J. Chou, D. Petrovic, and K. Ramachandran, "A distributed and adaptive signal processing approach to reducing energy consumption in sensor networks," in *Proc. 22nd Annu. Joint Conf. IEEE Comput. Commun. IEEE Soc. (INFOCOM'03)*, 2003, vol. 2, pp. 1054–1062.

[22] S. Madden. (2014). "Intel Lab data, 2004," [Online]. Available from http://www.select.cs.cmu.edu/data/labapp3/index.html

[23] T. Cormen, C. Leiserson, R. Rivest, and C. Stein, *Introduction to Algorithms*, 3rd ed. Cambridge, MA, USA: MIT Press, 2009.

[24] A. T. Ihler, J. W. Fischer III, and A. S. Willsky, "Loopy belief propagation: Convergence and effects of message errors," *J. Mach. Learn. Res.*, vol. 6, pp. 905–936, Dec. 2005.

[25] Y. Weiss and W. T. Freeman, "On the optimality of solutions of the max-product belief-propagation algorithm in arbitrary graphs," *IEEE Trans. Inf. Theory*, vol. 47, no. 2, pp. 736–744, Sep. 2006.

[26] J. Su, H. Zhang, C. X. Ling, and S. Matwin, "Discriminative parameter learning for Bayesian networks," in *Proc. 25th Int. Conf. Mach. Learn. (ICML'08)*, 2008, pp. 1016–1023.

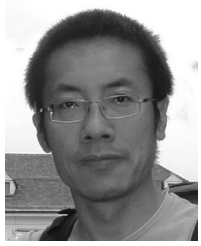
[27] A. Gersho and R. M. Gray, *Vector Quantization and Signal Compression*. Norwell, MA, USA: Kluwer, 1991.

[28] G. Anastasi, A. Falchi, A. Passarella, M. Conti, and E. Gregori, "Performance measurements of motes sensor networks," in *Proc. 7th ACM Int. Symp. Model. Anal. Simul. Wireless Mobile Syst. (MSWiM'04)*, 2004, pp. 174–181.

[29] Gecode Team. (2006). "Gecode: A generic constraint development environment," [Online]. Available: http://www.gecode.org/

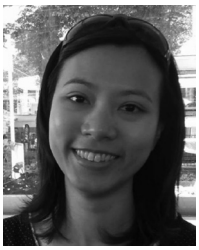
Farshid Hassani Bijarbooneh received the Bachelor's degree in applied mathematics from the Iran University of Science and Technology, Tehran, Iran, in 1999, and the Master's degree in computer science and Ph.D. degree in constraint programming for wireless sensor networks from Uppsala University, Uppsala, Sweden, in 2009 and 2015, respectively.

He was with the Mobility and Astra Research Groups, Uppsala University. He is currently a Postdoctoral Researcher with the SyMLab Research Group, Hong Kong University of Science and Technology, Clear Water Bay, Hong Kong. He has visited and collaborated with researchers from the Insight Centre for Data Analytics, Cork University, Cork, Ireland; INRIA Paris-Rocquencourt, Paris, France; Computer Networks (NET) Research Group, Göttingen University, Göttingen, Germany; and the Research Unit in Networking (RUN), University of Liège, Liège, Belgium. His research interests include optimization and constraint programming, cloud computing, Internet of Things, and sensor networks.



Wei Du received the B.S. degree in computer science from Tianjin University, Tianjin, China, in 1997, and the Ph.D. degree in computer science from the Chinese Academy of Sciences, Beijing, China, in 2002.

He is a Postdoctoral Researcher with the CITI Laboratory, INSALyon, Villeurbanne, France. Since 2012, he has been a Postdoctoral Researcher with INRIA, Rocquencourt, France; Hamburg University, Hamburg, Germany; the University of Liège, Liège, Belgium; the University of Innsbruck, Innsbruck, Austria; and the University of Göttingen, Göttingen, Germany. His research interests include applications of machine learning on computer networking.



Edith C.-H. Ngai received the Ph.D. degree from the Chinese University of Hong Kong, Shatin, Hong Kong, in 2007.

She is currently an Associate Professor with the Department of Information Technology, Uppsala University, Uppsala, Sweden. She was a Postdoctoral Researcher with Imperial College London, London, U.K., from 2007 to 2008. Since 2015, she has been a Visiting Professor with Ericsson Research Sweden. Her research interests include wireless sensor and mobile networks, Internet of Things, network security and privacy, smart city, and e-health applications.

Dr. Ngai is a member of the ACM. She has served as a TPC Member in leading networking conferences, including IEEE ICDCS, IEEE Infocom, IEEE ICC, IEEE Globecom, IEEE/ACM IWQoS, IEEE CloudCom, etc. She was a TPC Co-Chair of the Swedish National Computer Networking Workshop (SNCNW'12) and QShine'14. She is a Program Chair of ACM womENcourage 2015, a TPC Co-Chair of IEEE SmartCity 2015 and IEEE ISSNIP 2015. She has served as a Guest Editor for a special issue of the IEEE INTERNET OF THINGS JOURNAL, the IEEE TRANSACTIONS ON INDUSTRIAL INFORMATICS, Springer *Mobile Networks and Applications (MONET)*, and the *EURASIP Journal on Wireless Communications and Networking*. She is a VINNMER Fellow (2009) awarded by VINNOVA, Sweden. Her coauthored papers have received the Best Paper Runner-up Awards of IEEE IWQoS 2010 and ACM/IEEE IPSN 2013.

Xiaoming Fu (M'02–SM'09) received the Ph.D. degree in computer science from Tsinghua University, Beijing, China, in 2000.

He was a Research Staff with the Technical University Berlin, Berlin, Germany, until joining the University of Göttingen, Göttingen, Germany, in 2002, where he has been a Professor of computer science and the Head of the Computer Networks Group since 2007. His research interests include network architectures, protocols, and applications.

Dr. Fu is a Distinguished Lecturer of the IEEE Communications Society. He is currently an Editorial Board Member of the IEEE COMMUNICATIONS MAGAZINE, the IEEE TRANSACTIONS ON NETWORK AND SERVICE MANAGEMENT, Elsevier *Computer Networks*, and *Computer Communications*, and has served on the Organization or Program Committees of leading conferences such as INFOCOM, ICNP, ICDCS, MOBICOM, MOBIHOC, CoNEXT, ANCS, ICN, and COSN. He has served as a Secretary (2008–2010) and a Vice Chair (2010–2012) of the IEEE Communications Society Technical Committee on Computer Communications (TCCC), and Chair (2011–2013) of the Internet Technical Committee (ITC) of the IEEE Communications Society and the Internet Society. He has been involved in EU FP6 ENABLE, VIDIOS, Daidalos-II, and MING-T projects and is the Coordinator of the FP7 GreenICN, MobileCloud, and CleanSky projects. He was the recipient of the ACM ICN 2014 Best Paper Award, the IEEE LANMAN 2013 Best Paper Award, and the 2005 University of Göttingen Foundation Award for Exceptional Publications by Young Scholars.

Jiangchuan Liu (S'01–M'03–SM'08) received the B.Eng. degree in computer science (*cum laude*) from Tsinghua University, Beijing, China, in 1999, and the Ph.D. degree in computer science from the Hong Kong University of Science and Technology, Clear Water Bay, Hong Kong, in 2003.

He is a University Professor with the School of Computing Science, Simon Fraser University, Burnaby, BC, Canada, and an NSERC E. W. R. Steacie Memorial Fellow. He is an EMC-Endowed Visiting Chair Professor with Tsinghua University (2013–2016). From 2003 to 2004, he was an Assistant Professor with the Chinese University of Hong Kong. His research interests include multimedia systems and networks, cloud computing, social networking, online gaming, big data computing, wireless sensor networks, and peer-to-peer networks.

Dr. Liu has served on the Editorial Boards of the IEEE TRANSACTIONS ON BIG DATA, the IEEE TRANSACTIONS ON MULTIMEDIA, IEEE COMMUNICATIONS SURVEYS AND TUTORIALS, IEEE ACCESS, the IEEE INTERNET OF THINGS JOURNAL, *Computer Communications*, and Wiley's *Wireless Communications and Mobile Computing*. He is the Steering Committee Chair of the IEEE/ACM IWQoS from 2015 to 2017. He was the corecipient of the inaugural Test of Time Paper Award of the IEEE INFOCOM (2015), ACM TOMCCAP Nicolas D. Georganas Best Paper Award (2013), ACM Multimedia Best Paper Award (2012), the IEEE Globecom Best Paper Award (2011), and the IEEE Communications Society Best Paper Award on Multimedia Communications (2009).