Energy Efficient Sensor Activation for Water Distribution Networks Based on Compressive Sensing

Rong Du, *Student Member, IEEE,* Lazaros Gkatzikis, *Member, IEEE,*
Carlo Fischione, *Member, IEEE,* and Ming Xiao, *Senior Member, IEEE*

Abstract—The recent development of low cost wireless sensors enables novel Internet-of-Things (IoT) applications, such as the monitoring of water distribution networks. In such scenarios, the lifetime of the wireless sensor network (WSN) is a major concern, given that sensor node replacement is generally inconvenient and costly. In this paper, a compressive sensing based scheduling scheme is proposed that conserves energy by activating only a small subset of sensor nodes in each timeslot to sense and transmit. Compressive sensing introduces a cardinality constraint that makes the scheduling optimization problem particularly challenging. Taking advantage of the network topology imposed by the IoT water monitoring scenario, the scheduling problem is decomposed into simpler subproblems, and a dynamic-programming-based solution method is proposed. Based on the proposed method, a solution algorithm is derived, whose complexity and energy-wise performance are investigated. The complexity of the proposed algorithm is characterized and its performance is evaluated numerically via an IoT emulator of water distribution networks. The analytical and numerical results show that the proposed algorithm outperforms state-of-the-art approaches in terms of energy consumption, network lifetime, and robustness to sensor node failures. It is argued that the derived solution approach is general and it can be potentially applied to more IoT scenarios such as WSN scheduling in smart cities and intelligent transport systems.

Index Terms—Energy balancing, energy efficiency, water distribution networks, compressive sensing

I. INTRODUCTION

ENERGY-EFFICIENCY is crucial in wireless sensor networks (WSNs), where battery recharging or replacement is difficult or impossible. This is the typical case of water distribution networks. Compressive sensing (CS) provides an efficient way to reconstruct a signal from limited samples. In this article we investigate the potential of CS-based sensor activation schemes to extend network lifetime.

For the sake of reducing the risks of water pollution and pipeline leakages [1], [2], WSNs have been deployed in several cities to monitor water distribution networks [1], [3], [4]. This is a typical example of IoT. As the pipelines are located underground, it is inconvenient and costly to replace the batteries of the sensor nodes once the batteries are depleted. On the other hand, it is also important for WSNs to guarantee a good sensing performance [5], e.g., timely measurement and accurate estimation. Thus, a tradeoff between energy-efficiency and monitoring performance is naturally introduced.

The miniaturization and reduced cost of water sensors facilitate the deployment of a dense, spatially distributed, sensor network for the collection of hydraulic and water quality data [6]. Dense WSNs take advantage of redundant sensor nodes to minimize the duty cycle of each individual sensor node, to prolong network lifetime [7], and to enhance robustness to sensor failures. Since only a subset of sensor nodes has to be active in each time instance, scheduling the sleep periods of the sensor nodes has been shown to improve energy-efficiency [8]. Under this assumption, the sensor nodes can be uniformly deployed over the area of interest and then the activation schedule has to be decided. The resulting network of activated sensor nodes has to be connected and the monitoring performance of the system should not be compromised.

The measurements of dense sensor networks are highly correlated. Thus, we may prolong network lifetime significantly by introducing only a small loss in monitoring accuracy [9]. Consequently, in this paper, we build upon our prior work [10] to devise a CS based activation scheme that significantly reduces the number of sensor nodes to be activated with the objective of reducing the energy consumption and balancing the residual energy of the sensor nodes. In particular, energy balancing can achieve the objectives of maximum lifetime and robustness to sensor node failures. As pipelines are located underground and hence battery replacement is not easy, these objectives are of utmost importance in water distribution networks.

In this context, we consider a densely deployed WSN as the one depicted in Fig. 1. In each time instance, only a few sensor nodes are scheduled to sense and transmit data to the sink nodes in a multi-hop fashion. We pursue energy efficiency along with the following goals. First, the energy consumption of sensor nodes should be balanced. Second, sensing performance should not be compromised to achieve the goals of accurate monitoring and fast response. Third, the
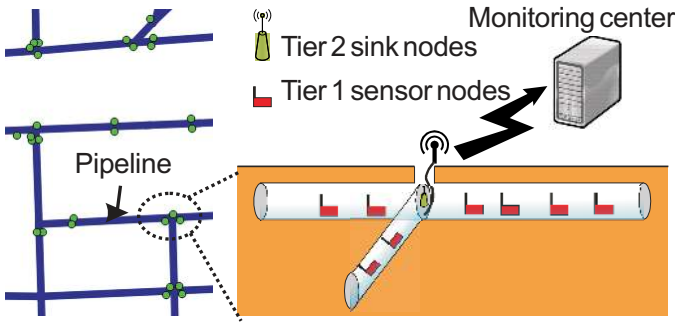
Fig. 1. WSN in the water distribution network system.

activated sensor nodes should be always connected, so that the sensed data reach the sink nodes. In summary, the key contributions of the paper are as follows:

- We propose the use of dense WSNs and compressive sensing for energy efficient monitoring of water distribution networks. We exploit the network structure to devise a CS-based sensor node activation and data gathering scheme that significantly extends network lifetime.
- We formulate a novel energy balancing optimization problem with connectivity and cardinality constraints that arises from the specific scenario under consideration.
- We propose a dynamic programming approach to solve the energy-balancing problem and derive a low-complexity solution algorithm. We characterize the complexity of the algorithm. Sufficient conditions for optimality are also derived.
- We evaluate the performance of the proposed algorithm by comparing it to a derived lifetime upper bound and state-of-the-art algorithms proposed in the literature. The results indicate that our algorithm outperforms the existing algorithms in the considered network structure, and approaches the upper bound performance if the network is dense enough.

The rest of the paper is organized as follows. We provide an overview of related works in Section II. The proposed model of a water monitoring system, and the optimization problem formulation of energy balancing under connectivity and cardinality constraints are presented in Section III. Section IV is devoted to the analysis of the optimization problem and the proposed algorithm of polynomial complexity. A more general case is discussed in Section V. Numerical evaluations are provided in Section VI. The conclusions of this work are presented in Section VII.

II. RELATED WORK AND PRELIMINARIES

A. WSNs for water monitoring

Wireless sensor nodes are energy-constrained devices, and hence their limited resources should be used efficiently. In this direction, several works consider the problems of sensor activation and transmission scheduling towards maximizing network lifetime. In several scenarios, it has been shown that this objective is equivalent to balancing the residual energy of the nodes. Indicatively, in seminal work [11], energy consumption is modeled as a function of the traffic flow routing decisions. In this setting, the problem of maximizing

network lifetime can be cast as a linear optimization problem and the authors proposed the flow augmentation algorithm to solve it efficiently. It is also shown that energy balancing is a good approximation to network lifetime maximization. In [12] the authors consider a different scenario that each sensor node may either transmit data to its one hop neighbors with unit energy consumption, or directly to the sink node through long range communication but at higher energy cost. In this context, it is shown that the problem of maximizing network lifetime is equivalent to the problems of flow maximization and energy-balancing. A similar topology is considered in [13], [14], where network lifetime is also pursued by balancing energy consumption in the network.

WSNs enable us to detect pollution and pipeline leakages in water distribution networks, by monitoring system parameters, such as water quality and pressure in pipelines. However, the fact that water pipelines are located underground and hence are not easily accessible introduces additional challenges, concerning energy efficiency and sensor placement.

Deriving the optimal sensor placement for water quality monitoring is a challenging task. Due to budget constraints, a limited number of sensor nodes are deployed in the most representative positions of the network [15]. Existing works cover a diverse set of objectives [16], [17], [18], such as minimizing the population exposed to the contaminant and the detection time. Due to the binary nature of placement decisions, such problems are generally solved by mixed-integer programming. In general, the problem is particularly difficult, and the solution is achieved by heuristic algorithms [19]. An alternative to deterministic placement of sensor nodes is the uniform deployment of a dense sensor network over the area of interest.

In dense sensor networks, redundant sensor nodes are deployed to account for the failure of individual sensor nodes. In this case, not all sensor nodes have to be active to monitor, and hence scheduling the sleep and activation periods of sensor nodes can provide significant energy benefits. For instance, in sensor coverage problems, such mechanisms are used to maximize network lifetime while guaranteeing that all target demands are covered [20]. Besides, scheduling the activation/sleeping periods has been also considered to maximize the lifetime of a query-based WSN in [21]. In summary, here we consider a different strategy space, namely the optimal activation of sensor nodes under compressive sensing, so that network connectivity and monitoring quality in each monitoring timeslot is guaranteed.

B. Compressive sensing for data gathering

Energy savings in WSNs can be realized by minimizing the amount of transmitted data, e.g., through compression of sensed data and minimization of transmissions [22]. Recently, the alternative of compressive sensing (CS) was proposed to enable reconstruction of signals from a limited number of samples. CS is widely studied in large scale WSNs for environmental monitoring and data gathering [23], such as for underwater sensor networks [24].

Consider a network of N sensor nodes that have to transmit their sensing data denoted by vector $\boldsymbol{d} = [d_1, d_2, \ldots, d_N]^T$
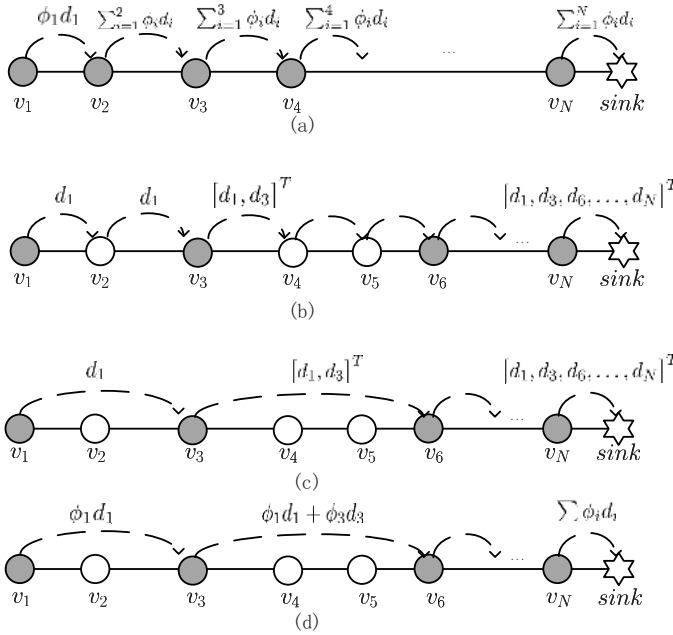
Fig. 2. Data gathering in (a) Compressive Data Gathering (CDG) [23]; (b) Compressed Sparse Function (CSF) [25]; (c) improved CSF; (d) Proposed scheme. d_i is the measurement of sensor node i, and ϕ_i is the projection vector of size $M \ll N$ used in sensor node i

to the sink node, where d_i is the data collected by sensor node i. A compressive data gathering (CDG) algorithm for data compression in the sensor node side and data recovery at the sink node side has been proposed in [23], such that each sensor node transmits only $M \ll N$ measurements to its next hop node, as depicted in Fig. 2 (a). As a result, the amount of transmitted data is greatly reduced and consequently significant energy is saved.

Inspired by the CDG algorithm, a compressed sparse function (CSF) algorithm for data gathering in WSNs has been proposed in [25]. It uses a discrete cosine transform (DCT) to derive a sparse representation of the measured quantity. The derived method satisfies the restricted isometry property [26] and hence guarantees that the original signal can be well recovered under certain conditions. Consequently, if the sink node collects M out of N data measurements of $\boldsymbol{d}$, it can use CS to estimate the remaining $N - M$ data measurements. The resulting data transmission schedule of CSF is depicted in Fig. 2 (b). The grey nodes correspond to the sensor nodes that need to upload their data measurements, and the white ones are simply relay data nodes.

Our work is motivated by the CSF approach in [25]. However, instead of considering only the sensing cost, we investigate the joint problem of sensor node activation and transmission scheduling. In particular, by activating at least M sensor nodes, which we call the cardinality constraint and is defined later in Section III, we can still achieve a good estimation of the state of the entire network. Thus, we propose that only the activated sensor nodes should perform sensing and participate in data forwarding. The proposed approach results to a sensing and transmission scheduling scheme similar to the one depicted in Fig. 2 (c). However, since only the activated sensor nodes participate in data forwarding to the sink node, we face the additional constraint that the corresponding set

has to be connected so that the sensing data reach the sink node. Moreover, the idea of dual-level CS [27] is applied, as shown in Fig. 2 (d). That is, in the first level, the data of the activated sensor nodes are transmitted in a CDG way, and recovered by traditional CS; in the second level, the data of the inactive sensor nodes are estimated similar to CSF. Compared to CSF, which addresses the question of "how many sensor nodes is needed for sensing" and uses the rest of the sensor nodes as relay nodes, our scheme specifically decides which sensor nodes should be activated to sense and transmit, and improves energy efficiency since the rest sensor nodes (white nodes in Fig. 2) can be put in sleep mode. It also balances the energy consumption of the activated sensor nodes.

Besides CDG and CSF, several compressive sensing based data gathering algorithms have been proposed. In [28], a minimum energy compressed data aggregation (MECDA) method has been considered. An algorithm based on minimum spanning tree and shortest path forest has been proposed to find the routing with the smallest energy consumption from the sensor nodes to the sink node. However, as long as the network topology is fixed, the routing is unchanged. As a result, the energy consumption of some sensor nodes is larger than others, and these critical sensor nodes may expire earlier, resulting to a disconnected network. Consequently, the resulting network lifetime may be short. A hierarchical data aggregation scheme using CS has been proposed in [29].The network is divided into several multilevel clusters, to reducing the amount of transmitted data in each timeslot. However, in this paper, we reduce energy consumption by scheduling the activation/sleep periods of sensor nodes. In [30], a distributed scheme based on opportunistic routing called Compressive Data Collection (CDC) has been proposed. Although not all the sensor nodes are active in every timeslot, the number of active sensor nodes is not minimized due to the randomness in the opportunistic routing, which means that it is possible to consume more energy than needed. In [31], an energy-efficient delay-aware algorithm (EDAL) has been proposed. It aims at finding routes from a given set of source nodes to the sink node with the minimum total cost. It also considers energy balancing among nodes. However, in addition to energy balancing, we consider the monitoring quality, which is captured by the cardinality constraint. Also, the source nodes in this paper are not deterministic, and thus the energy of sensor nodes are better balanced.

To summarize, this paper aims at scheduling the activation of sensor nodes to prolong network lifetime with guaranteed monitoring quality by CS. A new CS-based data gathering scheme is proposed that relies on the solution of a non-trivial optimization problem with connectivity and cardinality constraints. In this direction, an easy-to-implement and low-complexity algorithm is proposed to solve the problem.

III. System Model and Problem Formulation

We consider a water distribution network that is monitored by two tiers of nodes. The first tier consists of battery-powered sensor nodes that are densely deployed in the pipelines. Their tasks are i) sensing, ii) simple data processing, and iii) data

relaying to a set of sink nodes. Since sink nodes can not be reached by every sensor node directly due to the harsh communication environment underground, the first tier sensor nodes form a multi-hop communication path up to a sink node. Given also that pipelines are located underground and the distance between pipelines is generally large in comparison to transmission range, it is natural to assume that each sensor node can only transmit data to nodes that are located in the same pipeline [32]. Such two tiers hierarchical architectures perform well in terms of scalability and address the challenges of underground communications. Thus, they have been extensively considered for water distribution monitoring applications [4], [33].

The second tier consists of sink nodes, which are deployed at the junctions of pipelines, and are powered by the grid. One sink node is deployed at each junction. They are responsible for i) network maintenance, ii) data collection, iii) data storing, and iv) data transmission to a remote monitoring center. As sink nodes are powered by the grid, they have enough power for data transmission, and their lifetime can be considered unlimited. Sink nodes are also equipped with transceivers that support long range communications such that the gathered data can eventually reach the remote monitoring center.

Given that communication among sensor nodes located in different pipelines is not possible, we consider the activation problem separately in each pipeline. Therefore, each pipeline network can be represented by a communication graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where vertex set $\mathcal{V}$ represents the nodes in the pipeline, namely N sensor nodes and one sink node at each end, and edge set $\mathcal{E}$ represents the links among nodes. Let s_l be the leftmost sink node, s_r the rightmost sink node, and $v_1, v_2, \ldots, v_N$ be the sensor nodes from left to right. Let r_i be the transmission range of node v_i, and $d(v_i, v_j)$ the distance between nodes v_i and v_j. Then, for any two nodes $v_i, v_j \in \mathcal{V}$, $\langle v_i, v_j \rangle \in \mathcal{E}$ if and only if $d(v_i, v_j) \leq r_i$.

Time is slotted and in each slot t, a sensor node is either activated to sense and transmit the sensed data, or is set to sleep mode to save energy. The activated sensor nodes transmit data in the CDG way, such that the volume of the transmitted data is the same for every sensor node. Coherently, and without loss of generality, we assume also that energy consumption, including sensing and transmission, of an activated sensor node is normalised to 1 energy unit per timeslot and 0 for the sleeping sensor nodes [34]. Each sensor node is characterized by an energy budget E_i.

Let binary variable $x_i(t)$ denote whether sensor node v_i is activated in timeslot t, $x_i(t)$ is 1 if sensor node v_i is active, and 0 otherwise. Then, vector $\boldsymbol{x}(t) = [x_1(t), \ldots, x_N(t)]^T$ is the corresponding activation schedule at t. Denote $\mathcal{V}_A(t)$ the set of activated sensor nodes along with the two sink nodes and $G(\mathcal{V}_A(t))$ the induced graph that contains only the nodes in $\mathcal{V}_A(t)$. The following connectivity constraint guarantees that the data measured by the activated sensor nodes can reach monitoring center.

Definition 1: **(Connectivity Constraint)** The activated nodes at timeslot t satisfy connectivity constraint if and only if the induced graph $G(\mathcal{V}_A(t))$ is connected.

This connectivity constraint also guarantees that the pipeline is uniformly monitored by sensor nodes, and hence the pipeline is well covered by the activated sensor nodes. Besides the connectivity constraint, the monitoring performance should also be ensured. As a scheme based on CSF [25] is used for data gathering and recovery, the monitoring performance requirement can be captured by the following cardinality constraint:

Definition 2: **(Cardinality Constraint)** The activated sensor nodes satisfy cardinality constraint if and only if for the number of the activated sensor nodes $M(t)$ holds: $M(t) = \sum x_i(t) \geq ck \log N \triangleq M_{\text{cs}}$, where c a positive constant and k is the sparsity of data [26], [35].

If the activated sensor nodes satisfy this cardinality constraint, the monitoring center can estimate the measurements of the remaining sensor nodes using CS and hence the desired monitoring accuracy is guaranteed. In summary, at each timeslot, an activation schedule is feasible if and only if both the connectivity constraint (Definition 1) and the cardinality constraint (Definition 2) are satisfied. Then, we can derive the following upper bound of network lifetime.

Proposition 1: Suppose that the energy consumption of each sensor node in each timeslot is 1 if the sensor node is activated and 0 if the sensor node is inactive. Suppose that at each timeslot, the activated sensor nodes satisfy both the connectivity constraint of Definition 1 and the cardinality constraint of Definition 2. Then, an upper bound of network lifetime is $\bar{T} = \sum_i E_i / M_{\text{cs}}$.

Proof: Denote the number of sensor nodes that must be activated in timeslot t by $m(t)$, which is determined by the number of required sensor nodes for connectivity $M_c(t)$ and by the CS cardinality requirement M_{cs}. We have that $m(t) \geq \max\{M_c(t), M_{\text{cs}}\}$ such that both the connectivity constraint and the cardinality constraint are satisfied. Relaxation $m(t) \geq M_{\text{cs}}$ ensures that at each timeslot, at least M_{cs} sensor nodes are activated. This leads to a reduction of the total energy of the network by M_{cs}. As the total energy of network is $\sum_i E_i$, an upper bound of the network lifetime is $\bar{T} = \sum_i E_i / M_{\text{cs}}$. ∎

The upper bound of network lifetime in Proposition 1 corresponds to a network instance where the total energy is perfectly balanced among a connected subset of sensor nodes of cardinality M_{cs} such that the cardinality constraint is met. Based on this observation, in each timeslot, we pursue to i) minimize the energy consumption and ii) balance the residual energy of the sensor nodes. The former, given that all sensor nodes are identical in terms of energy consumption (except the sink nodes), translates into activating the minimum number of sensor nodes that can guarantee both connectivity and sensing performance. However, given the number of sensor nodes to be activated, generally several feasible activation schedules exist. Accordingly, towards ii) we need to find a schedule that balances the residual energy of the sensor nodes.

Let $E_i(t)$ denote the residual energy of sensor node v_i at timeslot t. Then the normalized residual energy of it is $p_i(t) = E_i(t)/E_i$, where recall that E_i is the initial energy of sensor node v_i. Since a sensor node with residual energy less than 1 cannot be activated any more, we use $\mathcal{V}(t) = \{v_i \in \mathcal{V} | E_i(t) \geq 1\}$ to denote the set of sensor nodes

that have enough residual energy to participate in sensing and data forwarding. To balance the energy, among sensor nodes, the ones of maximum normalized residual energy have to be activated. Accordingly, we pose the following optimization problem:

$$\max_{\boldsymbol{x}} \sum_{i \in \mathcal{V}} x_i p_i \tag{1a}$$

$$\text{s.t.} \quad \sum_{i \in \mathcal{V}_A} x_i = \max\{M_{cs}, M_c\}, \tag{1b}$$

$$G(\mathcal{V}_A) \text{ is connected}, \tag{1c}$$

$$x_i \in \{0, 1\}, \ \forall i \in \mathcal{V}, \tag{1d}$$

where M_c is the minimum number of sensor nodes that must be activated to satisfy the connectivity constraint, and $M_{cs} = ck \log N$ is a known value imposed by CS [26]. In the optimization problem, we have discarded the time index t for notational simplicity. The decision variables of the problem are collected in the vector $\boldsymbol{x}$. Clearly, if the optimal solution is such that at time t we have $x_i(t) = 1$, then sensor node v_i is activated. Otherwise, if $x_i(t) = 0$, it is not activated. The objective of Problem (1) is to activate the minimum possible number of sensor nodes in each timeslot, and the sum of the normalized residual energy of these sensor nodes should be as large as possible. However, Problem (1) in general is NP-hard.

Proposition 2: Problem (1) is NP-hard.

Proof: Please refer to Appendix A. ∎

Even though Problem (1) is NP-hard in general, we will show that in our network model where special conditions hold (see Assumptions 1 and 2 as will be given in Section IV), it can be solved efficiently.

IV. OPTIMAL ACTIVATION FOR ENERGY BALANCING

In this section, we propose a solution approach to Problem (1) and we derive an efficient algorithm for activation of sensor nodes. We characterize the complexity of the derived algorithm, we prove its optimality under certain conditions and we describe how it could be translated into an applicable energy-efficient network protocol. As we develop the analysis, we compare the proposed algorithm to existing methods from the literature.

A. Balancing residual energy in water distribution sensor networks

We propose to solve Problem (1) by the following two steps procedure. The first step consists in finding M_c in (1b), the minimum number of activated sensor nodes to satisfy the connectivity constraint, by finding the shortest path from sink node s_l to sink node s_r. If $M_c > M_{cs}$, where recall that M_{cs} is the minimum number imposed by CS (cardinality constraint), the number of required sensor nodes for activation, m, is set to M_c, otherwise it is set to M_{cs}. Given m, the second step consists in searching for the exact m connected sensor nodes of maximum sum of weights for activation.

Now we are in the position to develop our proposed solution algorithm to Problem (1). First let us clarify two useful assumptions that hold in WSNs deployed in water distribution pipelines:

Algorithm 1 Greedy-based search (GBS) algorithm

Input: The adjacency matrix of nodes $\boldsymbol{A}$.
Output: Minimum number of sensor nodes that ensures connectivity M_c.
1: Set $M_c \leftarrow 0$, $k \leftarrow 0$, $v_0 \leftarrow s_l$.
2: **while** $s_r \notin \mathcal{N}_-(v_k)$ **do**
3:     **if** $\mathcal{N}_-(v_k) \neq \emptyset$ **then**
4:         $k \leftarrow \max\{i : v_i \in \mathcal{N}_-(v_k)\}$, $M_c \leftarrow M_c + 1$
5:     **else**
6:         $M_c \leftarrow +\infty$ // No feasible solution
7:         **return** M_c
8:     **end if**
9: **end while**
10: **return** M_c

Assumption 1: All sensor nodes are characterized by the same communication range $r_i = r$.

Assumption 2: All sensor nodes are deployed in a line.

These assumptions are instrumental to set the analysis for the fundamental properties of the optimization solution. Once we have developed such an analysis, in Section V we extend our analysis to capture cases of unequal range and sensor nodes not strictly deployed on a straight line. A generic solution algorithm for Problem (1) is derived that can be applied in both cases. The proposed algorithm has to be executed in each timeslot for each pipeline. For notational simplicity, we focus on a specific pipeline and timeslot. We denote the set of neighbors of node v_i by $\mathcal{N}(v_i) = \{v_j | \langle v_i, v_j \rangle \in \mathcal{E}\}$, the upstream neighbor set (UNS) of v_i by $\mathcal{N}_+(v_i) = \{v_j | v_j \in \mathcal{N}(v_i) \wedge j < i\}$, and the downstream neighbor set (DNS) of v_i by $\mathcal{N}_-(v_i) = \{v_j | v_j \in \mathcal{N}(v_i) \wedge j > i\}$. The proposed algorithm consists of two subroutines. First, the minimum number of sensor nodes M_c that guarantees connectivity is derived, and then a dynamic programming algorithm determines the sensor nodes to be activated. The two subroutines are described in detail in the following subsections.

1) Calculating the minimum number of sensor nodes to be activated: The optimal number of sensor nodes for activation is determined by M_c, the minimum number of sensor nodes that guarantees connectivity. To calculate M_c, we propose the greedy-based search (GBS) algorithm, as shown in Algorithm 1, which is optimal if Assumptions 1 and 2 hold. In each iteration, the furthest node v_i that belongs to the DNS of the current node v_k is selected (Line 4). If in any iteration, the DNS of current node is empty, the network is disconnected and the GBS algorithm returns $+\infty$. This indicates that no feasible solution can be found, and the network has expired.

2) Finding the maximum weighted connected subset of sensor nodes: We cast this problem as an instance of dynamic programming [37]. Assume that in a given state, v_i has been selected to be activated and k additional sensor nodes have to be activated out of v_{i+1} to v_N such that the selected k sensor nodes and v_i are connected. Let $g(v_i, k)$ be the maximum total residual energy out of all the possible subsets of k activated

sensor nodes. Then,

$$g(v_i,k) = \begin{cases} \max\limits_{v_j \in \mathcal{N}'_-(v_i)} \{g(v_j, k-1)+p_j\} & \text{if } \mathcal{N}'_-(v_i) \neq \emptyset, \\ -\infty & \text{otherwise}, \end{cases}$$
(2)

where p_j is the normalized residual energy of sensor node v_j, and $\mathcal{N}'_-(v_i) = \mathcal{N}_-(v_i) \backslash \{s_r\}$ is the set of nodes in the DNS of v_i except for the sink node s_r.

For $k = 1$, the selected sensor node has to be a neighbor of the sink node s_r. Consequently, for any sensor node $v \in \mathcal{V}$ we set

$$g(v_i, 0) = \begin{cases} 0 & \text{if } s_r \in \mathcal{N}_-(v_i), \\ -\infty & \text{otherwise}. \end{cases}$$
(3)

Based on recursive function defined by (2) and (3), we devise the sensor node activation on edge (SAE) algorithm to solve Problem (1). The exact steps are described in Algorithm 2, where $\boldsymbol{A}_{(N+2)\times(N+2)}$ is the adjacency matrix of the network with $a_{ij} = 1$ if and only if v_i and v_j are connected. Vector $\boldsymbol{p} = [p_1, p_2, \ldots, p_N]^T$ captures the normalized residual energy of sensor nodes. Notice that the sink nodes v_l and v_r are powered by the grid, and hence always active.

The SAE algorithm calculates $g(v_i, 0)$ in (3) for the nodes which can directly communicate to the rightmost sink node s_r in lines 3 to 6. Then, the algorithm calculates $g(v_k, i)$ for $i = 1$ to m recursively according to (2) in lines 7 to 12, and finally $g(s_l, m)$ in line 13 and 14. The set of sensor nodes that leads to optimal $g(s_l, m)$ is formed in lines 15 to 21.

By making use of the SAE and GBS algorithms, the optimal solution to Problem (1) can be derived through the sensor node activation with cardinality constraint (SACC) algorithm, which is described in Algorithm 3. We prove optimality of SACC in the following result:

Theorem 1: Consider optimization problem (1), and let Assumptions 1 and 2 hold. Then, the SACC algorithm derives the optimal solution to Problem (1), namely it calculates one of the optimal activation schedules.

Proof: Please refer to Appendix B. ∎

Next, we analyze the complexity of the SACC algorithm.

Proposition 3: Let ρ be the density of the deployed sensor network, N the number of sensor nodes, and r the communication range of each sensor node. Then, time complexity of the SACC algorithm is $O(\max\{N^2, kNr\rho\log N\})$, where k is the sparsity of the data.

Proof: Please refer to Appendix C. ∎

Proposition 4: Let ρ be the density of the deployed sensor network, N the number of sensor nodes, and r the communication range of each sensor node. Then, the space complexity of the SACC algorithm is $O(N^2)$.

Proof: Please refer to Appendix D. ∎

Propositions 3 and 4 demonstrate that the proposed SACC algorithm is of low complexity. Thus, the sink nodes can apply it to determine which sensor nodes should be activated in each timeslot.

B. A network protocol for optimal sensor node activation

The SACC algorithm can be applied in a WSN for water monitoring based on the following phases: i) network con-

---

Algorithm 2 Sensor node activation on edge (SAE) algorithm

Input: $\boldsymbol{A}$, $\boldsymbol{p}$ and the number of sensor nodes m to be activated.
Output: The set of sensor nodes $\mathcal{V}_A$ to be activated.
1: Construct a matrix $\boldsymbol{G} = \{g_{ij}\}$ of size $(N+1) \times (m+1)$, where all its elements are set to $-\infty$ initially.
2: Construct a node matrix $\boldsymbol{H} = \{h_{ij}\}$ of size $(N+1) \times (m+1)$ where all its elements are set to -1 initially.
3: **for** $\forall v_k : s_r \in \mathcal{N}_-(v_k)$ **do**
4:    $g(v_k, 0) = 0$ and set $g_{k1} = g(v_k, 0)$
5:    $h_{k1} = N + 1$
6: **end for**
7: **for** $i = 2$ to m **do**
8:    **for** $k = N$ to m **do**
9:       $g_{ki} \leftarrow \max\limits_{j:v_k \in \mathcal{N}_-(v_j)} \{g_{j(i-1)} + p_k\}$
10:      $h_{ki} \leftarrow \arg\max\limits_{j:v_k \in \mathcal{N}_-(v_j)} \{g_{j(i-1)} + p_k\}$
11:    **end for**
12: **end for**
13: $g_{1(m+1)} \leftarrow \max\limits_{v_j \in \mathcal{N}_-(s_l)} \{g_{jm}\}$
14: $h_{1(m+1)} \leftarrow \arg\max\limits_{v_j \in \mathcal{N}_-(s_l)} \{g_{jm}\}$
15: $c \leftarrow 1$
16: Construct a set $\mathcal{S} = \emptyset$ whose elements are the subscripts of the sensor nodes that need to be activated.
17: **for** $k = m + 1$ to 2 **do**
18:    $\mathcal{S} = \mathcal{S} \cup \{h_{ck}\}$
19:    $k \leftarrow h_{ck}$
20: **end for**
21: $\mathcal{V}_A = \{v_i | i \in \mathcal{S}\}$
22: **return** $\mathcal{V}_A$

Algorithm 3 Sensor node activation with cardinality constraint (SACC) algorithm

Input: Adjacency matrix $\boldsymbol{A}$, and their normalized residual energy $\boldsymbol{p}$.
Output: A set of sensor nodes $\mathcal{V}_A$ that need to be activated.
1: $M_c \leftarrow \text{GBS}(A)$
2: **if** $M_c < +\infty$ **then**
3:    Calculate $M_{cs} = ck \log N$
4:    $\mathcal{V}_A \leftarrow \text{SAE}(\boldsymbol{A}, \boldsymbol{p}, \max\{M_c, M_{cs}\})$
5:    **return** $\mathcal{V}_A$
6: **else**
7:    **return** $\emptyset$
8: **end if**

---

figuration, ii) node activation, iii) data transmission, and iv) data recovery. Network configuration takes place only once, namely after the deployment of the sensors. Node activation and data transmission phases are executed sequentially at sink nodes and the activated sensor nodes respectively. Once the sink nodes have gathered all the data, they forward them to the monitoring center, where data recovery is performed.

1) Network Configuration: Once the sensor nodes have been deployed, the sensor nodes that belong to each pipeline are associated with the corresponding sink node. As a result,

each sensor node is assigned to a specific sink node, whereas each sink node may serve several pipelines. In this phase, each sensor node reports its residual energy and its neighbors to the corresponding sink node. Based on the collected information, the sink nodes construct the adjacency matrix.

2) Node Activation: Once the WSN has been established, the monitoring process of the water distribution network can be initiated. For every timeslot t, each sink node calculates through SACC algorithm which sensor nodes should be activated to sense and transmit data, whereas the remaining sensor nodes are put to sleep for energy saving. The sink nodes coordinate activation of sensor nodes by broadcasting the ID of the sensor nodes to be activated over a control channel. Thus, every sensor node becomes aware of the number of activated sensor nodes M in this slot. Then, each activated sensor node turns into sensing mode and only turns on again to receive and transmit data in the corresponding period according to the sequence in the list of activated nodes, as shown in Fig. 3. The rest of the sensor nodes switch to sleep mode until the beginning of the next timeslot. Notice that when the result of the SACC algorithm is an empty set $\emptyset$, Problem (1) has no feasible solution and the lifetime of the network has been reached.

3) Data Transmission: The SACC algorithm determines the sensor nodes to be activated, whereas the routing decision is to transmit the sensing report to the closest sink node hop by hop in a CDG manner [23]. Meanwhile, the deactivated sensor nodes do not participate in relaying. Once the sink node receives the data from the activated sensor nodes, it updates its estimation of the residual energy of each sensor node and calculates the schedule for the subsequent timeslot.

Next, we analyze the energy savings of the proposed SACC algorithm over CDG and CSF [25] in each timeslot. Suppose the energy consumption of the sensor node that transmits one message is 1, then we have that the total energy consumption in transmission of our approach is at most m^2, as we pick m sensor nodes at each timeslot and each of them transmits at most m messages. For the CSF, the average total energy consumption in a timeslot is $0.5(m + 1)N$, whereas for CDG is mN. It follows that the SACC algorithm and its implementation consumes less energy than CSF and CDG as long as $m < 0.5N$, which generally holds as $m \ll N$. To sum up, the energy consumption caused by the SACC algorithm is reduced and more balanced compared to CSF and CDG.

4) Data Recovery: For each timeslot, the data of the inactive sensor nodes have to be recovered at the monitoring center. The data from each sink node are recovered individually, therefore we present the process of data recovery for a single pipeline. Consider a scenario of N sensor nodes where the activated sensor nodes in a timeslot are $v_{a_1}, v_{a_2}, \ldots, v_{a_m}$. Since the data are transmitted in the CDG way, the monitoring center first estimates the data of these m activated sensor nodes by CS. Let the estimation be $\boldsymbol{d_r} = [d_{a_1}, d_{a_2}, \ldots, d_{a_m}]^T$. Then,
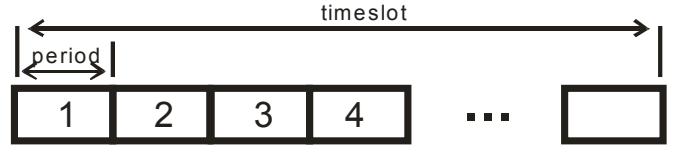
1. Sink node broadcast the activation result to all the sensor nodes
2. The sensor nodes in the activation list start sensing
3. Sensor node v_{a_1} transmits data to sensor node v_{a_2}
4. Sensor node v_{a_2} transmits data to sensor node v_{a_3}
...
m+2. Sensor node v_{a_m} transmits data to the sink

Fig. 3.   Sensor node activation and data transmission in a timeslot

according to CSF, we have

$$\boldsymbol{d_r} = \begin{bmatrix} p_1(a_1) & p_2(a_1) & \ldots & p_N(a_1) \\ p_1(a_2) & p_2(a_2) & \ldots & p_N(a_2) \\ \vdots & \vdots & \ddots & \vdots \\ p_1(a_m) & p_2(a_m) & \ldots & p_N(a_m) \end{bmatrix} \begin{bmatrix} c_1 \\ c_2 \\ \vdots \\ c_N \end{bmatrix} \triangleq \boldsymbol{P}_{\mathcal{A}} \boldsymbol{c},$$

(4)

where $\boldsymbol{P}_{\mathcal{A}}$ is the corresponding submatrix of $\boldsymbol{P} = \{p_i(j)\}$, which is the Type-IV DCT function [25], $p_i(j) = \sqrt{2/N} \cos[\pi(i - 0.5)(j - 0.5)/N]$, and $\boldsymbol{c}$ is the sparse representation of d under the basis $\boldsymbol{P}$, i.e., $d_i = \begin{bmatrix} p_1(i) & p_2(i) & \ldots & p_N(i) \end{bmatrix} \boldsymbol{c}$. Then, the monitoring center uses CS again to estimate the coefficient $\boldsymbol{c}$ by solving the following problem

$$\hat{\boldsymbol{c}} = \arg\min \quad \|\boldsymbol{c}\|_{l_1} \quad \text{s.t.} \quad \|\boldsymbol{d_r} - P_{\mathcal{A}}\boldsymbol{c}\|_{l_2} \leq \epsilon. \quad (5)$$

Given estimation of the coefficient $\hat{\boldsymbol{c}}$, the monitoring center can estimate the data of the inactivated sensor nodes by $\hat{d}_i = \begin{bmatrix} p_1(i) & p_2(i) & \ldots & p_N(i) \end{bmatrix} \hat{\boldsymbol{c}}, \quad \forall i \notin \{a_1, \ldots, a_m\}$. As we will show in the numerical evaluation section, this scheme enables the monitoring center to accurately estimate the state of the water distribution network based only on the measurements of the activated sensor nodes.

V. ANALYSIS OF ENERGY BALANCING OPTIMALITY IN GENERAL SCENARIOS

In the analysis of the previous section, we assumed that the transmission ranges of the sensor nodes are identical and that sensor nodes are located on a perfect line. In this section, we extend the analysis to the case of unequal transmission ranges. After that, we demonstrate that the solution of the proposed algorithm is optimal even in scenarios where sensor nodes are not located on a perfect line if certain conditions hold.

A. Optimality of the SACC algorithm in the case of unequal transmission ranges

In this section, we assume that each sensor node v_i is characterized by a different transmission range r_i. In this case, the definition of neighborhood needs to be updated accordingly, i.e., $v_j \in \mathcal{N}(v_i)$ if and only if $d(v_i, v_j) \leq r_i$ and v_i and v_j are in the same pipeline. Next, we derive a sufficient condition for the optimality of SACC in this generalized context.
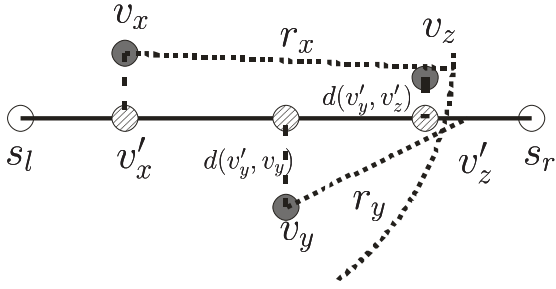
Fig. 4. Illustration of the case that sensor nodes are not strictly in a line

Fig. 5. Performance of compressive sensing with different M_{cs}/N-ratio

Theorem 2: Consider optimization problem (1). Suppose the sensor nodes are deployed in a line, and let the transmission ranges of the sensor nodes satisfy the following condition,

$$\forall x, \forall v_y \in \mathcal{N}_-(v_x), \forall v_z \in \{v_i | v_i \in \mathcal{N}_-(v_x) \wedge i > y\}, \quad (6)$$
$$v_z \in \mathcal{N}_-(v_y).$$

Then, the SACC algorithm calculates the optimal solution of Problem (1).

Proof: Please refer to Appendix E. ∎

On the other hand, if Condition (6) is not satisfied, then optimality of SACC is not guaranteed.

B. Optimality of SACC algorithm in the case that sensor nodes are not deployed on a line

Generally, sensor nodes may not be deployed on a perfect line. Nevertheless, condition (6) can be generalized to capture such cases. We draw a line l from s_l to s_r and project all the sensor nodes onto l, such that v_i' is the projection of v_i. Then, a sufficient condition for optimality of the SACC algorithm is:

$$\forall v_x \in \mathcal{V}, \forall v_y \in \mathcal{N}_-(v_x),$$
$$\forall v_z \in \{v_i | v_i \in \mathcal{N}_-(v_x) \wedge d(s_l, v_y') < d(s_l, v_z')\}, \quad (7)$$
$$d(v_y', v_z')^2 + d(v_y', v_y)^2 \leq r_y^2,$$

where r_y is the transmission range of sensor node v_y, v_x', v_y', v_z' are the projections of v_x, v_y, v_z on the line that connects s_l and s_r respectively.

An illustration of the required construction is provided in Fig. 4. We deploy virtual nodes on the line that connects the sink nodes s_l and s_r. Each virtual node is the projection of the sensor node on the line, and the virtual node's transmission range is also the projection of the transmission range of the corresponding sensor node. If Condition (7) holds, then Condition (6) also holds for the virtual nodes, and hence the SACC algorithm is optimal for Problem (1). Thus, Condition (7) can be considered as the generalization of Condition (6) for the case that sensor nodes are not strictly in a line.

VI. NUMERICAL EVALUATION

In this section, we evaluate the performance of our SACC algorithm in comparison to solutions from the literature. First, we discuss the performance of compressive sensing in the water distribution scenario.
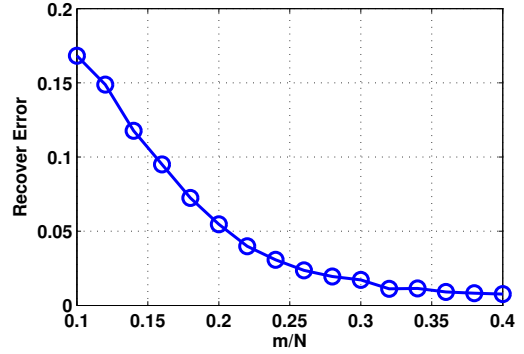
A. Compressive sensing

We use EPANET[1], a well known software that simulates water distribution piping systems, to generate a water distribution network and simulate the water pressure measurements of $N = 50$ sensor nodes, which are deployed along a pipeline in the network. We randomly select m sensor nodes to be activated, and collect their data in the CDG way. The actual data d of these m sensor nodes are recovered by CS using OMP algorithm [38]. Then we estimate the data of the inactivated sensor nodes based on the idea of CSF [25]. That is, we generate $P_{\mathcal{A}}$ according to the identifier of the activated sensor nodes and (4). Then, we calculate the coefficient $\hat{c}$ by using OMP to solve (5). The estimation on original data, $\hat{d}$, is calculated by $\hat{d}_i = [p_1(i), \ldots, p_N(i)]\hat{c}$. The resulting estimation error is defined as $err = \|\hat{d} - d\|_2 / \|d\|_2$. The performance of CS for different values of m is depicted in Fig. 5. Our results indicate that an error of less than 5% can be achieved by activating only 20% ($M/N = 0.2$) of the available sensor nodes.

B. Evaluation of the proposed algorithms

Without loss of generality, since the network problem can be decomposed to independent problems per pipeline, we focus on a single pipeline. The length of the pipeline, L, is normalized to length unit 1. Two sink nodes are deployed at the endpoints of the pipeline. In the initialization of each experiment, N sensor nodes are uniformly i.i.d. distributed over the pipeline within $(0, 1)$, and all sensor nodes characterized by the same transmit range r.

First, we compare the network lifetime performance of the proposed SACC algorithm to that of the CDG [23], CSF [25], CDC [30] and MECDA [28] algorithms. We consider a scenario of $N = 100$ sensor nodes uniform randomly deployed in a pipeline. The initial energy budget of the sensor nodes are independently set to be $b_i \sim \mathcal{N}(100, 1)$. The minimum number of active sensor nodes is set to be $M_{cs} = 20$. The ratio of nodes' transmission range to pipeline length, i.e. r/L, which we call 'normalized transmission range', ranges from 0.10 to 0.20. Due to the randomness of system parameters, each simulation is repeated for 100 different initial states.

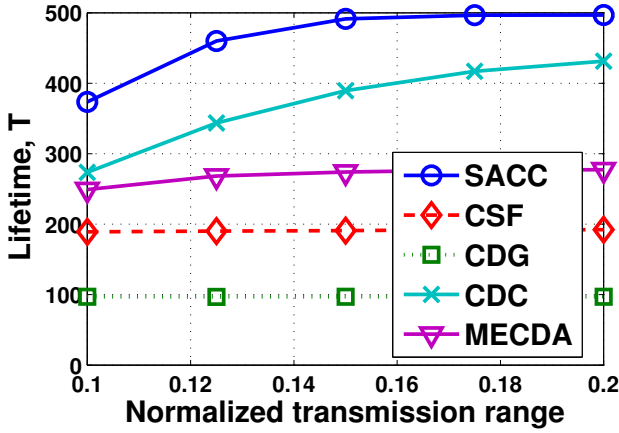[1]http://www.epa.gov/nrmrl/wswrd/dw/epanet.html

Fig. 6. Comparison of the network average lifetime achieved by SACC, CDG, CSF, CDC and MECDA with different transmission ranges

In Fig. 6, the horizontal axis represents the normalized transmission range, and the vertical axis represents the network lifetime. It shows that the network lifetime achieved by the SACC algorithm is longer than the network lifetime achieved by the CDG, CSF, CDC and MECDA algorithms. As the transmission range of sensor nodes increases, network connectivity improves. Thus, the network lifetime achieved by SACC, CDC and MECDA increases, which is not the case for CDG and CSF. The reason is that, in the latter two approach, all sensor nodes need to be active for all the time in these two algorithms. The lifetime achieved by the proposed SACC algorithm is about 5 times that of the CDG and 2.5 times of the CSF. This demonstrates the benefits of putting sensor nodes in sleep mode. On the other hand, the MECDA algorithm aims at finding the routing with the smallest energy consumption in each timeslot. Thus, the sensor nodes on the smallest energy consumption path will always be active until one of the sensor nodes expires, and hence the residual energy of the sensor nodes becomes extremely unbalanced. It is very likely that the network becomes disconnected as some critical sensor nodes expire even though the rest of the sensor nodes have abundant residual energy. Consequently, the network lifetime achieved by the MECDA algorithm is smaller than the lifetime achieved by the SACC algorithm. The CDC algorithm is based on opportunistic routing. The activation of the sensor nodes is more or less random, and thus the energy can be consider balanced. However, it can not guarantee that the number of active sensor nodes is minimum in each timeslot. Thus, more sensor nodes than needed may be activated, and hence energy consumption per timeslot is increased. Consequently, it is outperformed by SACC algorithm in terms of network lifetime. The lifetime achieved by the CDC algorithm is about 80% of that of the proposed SACC algorithm.

Since CDG and CSF do not take advantage of the deactivation of nodes, we benchmark the performance of the SACC algorithm with another two greedy algorithms that support deactivation of sensor nodes. We will describe these two algorithms first. The first one is called greedy-based search with random activation (GBS+R) algorithm and works

as follows: The GBS algorithm is first used to select M_c sensor nodes that ensure connectivity. The subscript of the chosen sensor nodes is decided according to Line 4 in the GBS algorithm. If $M_c \geq M_{\mathrm{cs}}$, these M_c sensor nodes are the ones to activate. Otherwise, the rest of $M_{\mathrm{cs}} - M_c$ sensor nodes are chosen randomly out of the remaining sensor nodes. The second algorithm that we use for comparison is called greedy-based search with maximum residual energy (GBS+M) algorithm, which is similar to GBS+R algorithm in the sense that the GBS algorithm is used to find n sensor nodes that guarantee connectivity. If $M_{\mathrm{cs}} > M_c$, we select the additional $M_{\mathrm{cs}} - M_c$ sensor nodes of highest normalized residual energy among the remaining $N - M_c$ sensor nodes.

We compare the SACC algorithm, the GBS+R algorithm, and the GBS+M algorithm in Fig. 7. The maximum lifetime of the network is calculated as $T_{\max} = \sum E_i / M_{\mathrm{cs}}$. It can be seen that the average network lifetime using the SACC algorithm is longer than the GBS+M algorithm and the GBS+R algorithm. When r/L is small, the average lifetime of the network is very limited. In this case, once a small number of vital sensor nodes expires, the network becomes disconnected. For example, the worst case in Fig. 7 (a) is that the lifetime is only 100 timeslots. It indicates that in the topology of this test, there exists a sensor node without which the network is definitely disconnected. As a result, that sensor node needs to be activated in every timeslot. Once it expires, the network expires.

However, as r/L increases from 0.08 to 0.20, the average network lifetime increases. It almost achieves the network maximum lifetime when the ratio is large enough, for example, $r/L = 0.16$ in Fig. 7 (a) and $r/L = 0.1$ in Fig. 7 (b). One reason is that the expected number of neighboring sensor nodes increases as r/L increases. In return, the probability that there exists a vital sensor node in the network or that the network is disconnected initially becomes smaller. Consequently, the variance of the performance also becomes smaller as r/L increases. Another reason is that, when r is small, the required number of sensor nodes for connectivity may be larger than the required number for compressive sensing. In this case the derived upper bound is loose, which leads to a gap between the lifetime in simulation and the theoretical maximum lifetime, as shown in Fig. 7 (a)-(c) when $r/L = 0.08$.

If the ratio M_{cs}/N is fixed, the network maximum lifetime is the same. However, under the same r/L, the network with more sensor nodes has larger average lifetime. Let us take Fig. 7 (b) and (e) as example. The maximum lifetime for these two networks is 500 timeslots. However, with the same normalized transmission range ($r/L = 0.12$), the average lifetime of the case (e) is 496.65 whereas it is 461.33 for the case (b). The reason is that even though the maximum lifetime dost not change, the density of the network increases as we deploy more sensor nodes.

Energy balancing may also achieve robustness against sensor failures. A failure of sensor node in our simulation means that the sensor node becomes non-functional any more due to a reason other than depleted battery. Such reasons can be a failure of the communication unit, a failure in the circuit, etc. We evaluate the performance of the algorithms under the
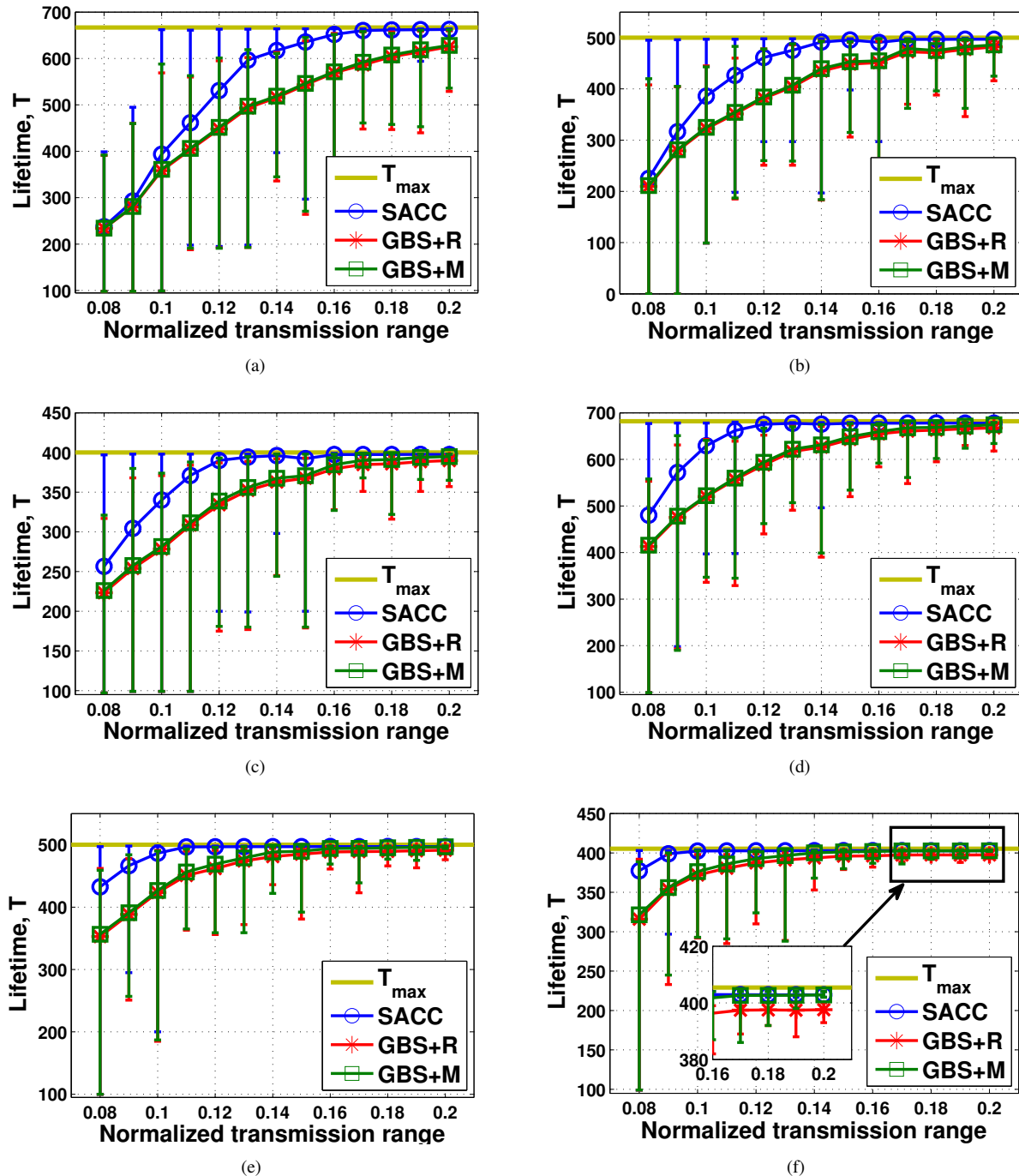
Fig. 7. Comparison of the network's average lifetime, maximum lifetime and minimum lifetime for the three algorithms under different transmission range with (a) $N = 100, M_{cs} = 15$; (b) $N = 100, M_{cs} = 20$; (c) $N = 100, M_{cs} = 25$; (d) $N = 150, M_{cs} = 22$; (e) $N = 150, M_{cs} = 30$; (f) $N = 150, M_{cs} = 37$

case of sensor failures. In each timeslot, every sensor node has a probability p_e to fail. As the sensor node can not work any more once it fails, we consider that its residual energy drops to 0 in just one timeslot. For the sake of fairness, the selection of sensor nodes to fail is uniformly random, but the same for the SACC, GBS+R and GBS+M algorithms in each test. In Fig. 8 we depict the results, where, on average, the network lifetime of the SACC algorithm is still better than the lifetime of GBS+R and GBS+M. This shows the robustness of the SACC algorithm against sensor node failures.

In summary, the SACC algorithm outperforms the CDG, CSF, GBS+R and GBS+M algorithms in terms of energy consumption and network lifetime.

VII. CONCLUSIONS

In this paper, we considered the problem of energy-efficient wireless sensor nodes activation in a typical Internet of Things scenario, namely monitoring a water distribution network. The uniform deployment of redundant sensor nodes leads to a dense sensor network and is a common strategy to enhance robustness to sensor node failures and to increase network lifetime, as long as only a small subset of sensor nodes is activated in each timeslot. In this context, we demonstrated how compressive sensing can be used to estimate the state
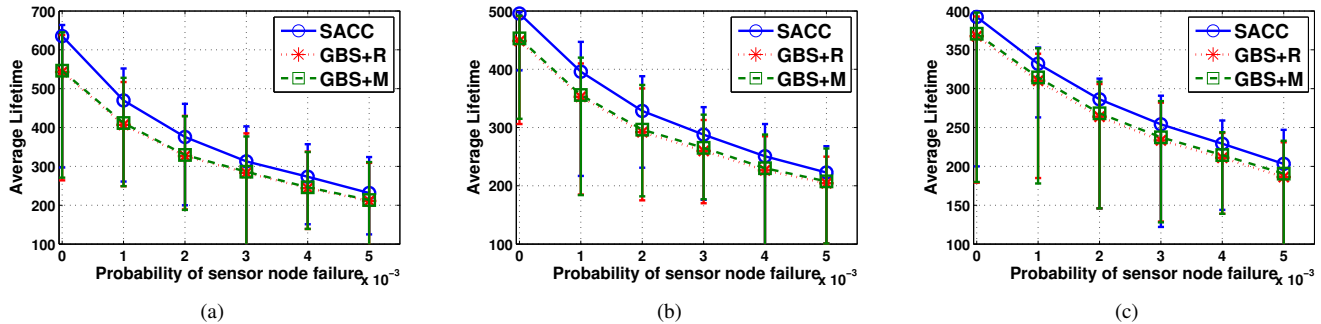
Fig. 8. Comparison of the network's average lifetime, maximum lifetime and minimum lifetime for the three algorithms under probability of sensor node failure with (a) $N = 100, M_{\text{cs}} = 15, r/L = 0.15$; (b) $N = 100, M_{\text{cs}} = 20, r/L = 0.15$; (c) $N = 100, M_{\text{cs}} = 25, r/L = 0.15$

of the whole network out of the measurements of a limited number of activated sensor nodes, and we tested it by an IoT emulator of water distribution networks.

We derived also an energy-balancing scheduling algorithm that activates the minimum number of sensor nodes, and specifically the ones of maximum normalized residual energy, to ensure both increased network lifetime and robustness to node failures. Our complexity analysis showed that the proposed algorithm can be executed at sink nodes. Our evaluation results demonstrated that the proposed algorithm outperforms existing state-of-the-art algorithms, namely CDG [23], CSF [25], CDC [30], MECDA [28]. In addition, the lifetime of our algorithm is close to a derived upper bound, if the network is dense enough.

An interesting topic of future work is to characterize analytically the connection between energy-balancing and maximum lifetime in the specific network topology. It is also interesting to test the algorithms in other applications, such as monitoring of railways and tunnels.

APPENDIX A
THE PROOF OF PROPOSITION 2

Proof: Without loss of generality, suppose that the available energy of both sink nodes s_l and s_r are large enough, i.e., $\sum_{i=1}^{N} p_i$. Let $d_{\min}$ the length in hops of the shortest path from s_l to s_r, which can be easily calculated for a given communication graph $\mathcal{G}$. Then, the minimum number of sensor nodes that satisfies constraint (1c) is $d_{\min} - 1$. Thus, Problem (1) is a K-cardinality-constrained maximum weight connected subgraph problem (MWCS) on vertex weight version of $\mathcal{G}$, where the weight of each vertex v_i is p_i, the weight of sink nodes are $\sum_{i=1}^{N} p_i$, and the required cardinality of selected subgraph is $K = \max\{M_{\text{cs}}, d_{\min} - 1\} + 2$. In K-cardinality-constrained MWCS, a connected subgraph of K vertices of maximum weight has to be found. Given that the weight of each sink node is larger than the remaining sensor nodes, the solution to the cardinality-constrained MWCS instance of Problem (1) always includes the two sink nodes. K-cardinality constrained MWCS is NP-hard even for binary weights and bipartite or planar graphs, as it has been proven in [36]. This concludes the proof. ∎

APPENDIX B
THE PROOF OF THEOREM 1

Proof: The proof consists of two parts. In the first part, we will show that when Assumptions 1 and 2 hold, the GBS algorithm finds the minimum number of sensor nodes that guarantees connectivity. In the second part, we will show that the result of the recursive function (2) is the same to the optimal solution calculated by the "exhaustive tree search algorithm", which is described as follows:

We enumerate all possible connected subsets of communication graph $\mathcal{G}$. For this purpose, we construct a tree of height larger than $m + 2$, where s_l is the root and the children of each node are its neighbors, i.e., the nodes within its transmission range. Let the weight of each vertex of such a tree be the normalized residual energy of the corresponding sensor node. The search targets the maximum weight path from the root to the sink node s_r with each sensor node appearing at most once in each path. This concludes the specification of the "exhaustive tree search algorithm". Such an algorithm is inefficient because of the enumeration of all possible paths, however we need it for the proof that our proposed solution algorithm to Problem (1).

To prove the first part, suppose Algorithm 1 returns the minimum required number of the activated sensor nodes to be M and the corresponding sensor nodes to be $v_{a_1}, v_{a_2}, \ldots, v_{a_M}$ from the left to the right. It is easy to see that these sensor nodes, together with two sink nodes, are connected.

According to line 2 in Algorithm 1, we have $s_r \notin \mathcal{N}_-(v_{a_{M-1}})$, i.e., the sink node s_r is not directly in range of sensor node $v_{a_{M-1}}$.

Next, we prove that there exists no connected subgraph $\mathcal{V}_A$ with less than M sensor nodes by contradiction. Assume that we can select $M - 1$ sensor nodes, i.e. $v_{b_1}, v_{b_2}, \ldots, v_{b_{M-1}}$ from the left to the right, such that $s_l, v_{b_1}, v_{b_2}, \ldots, v_{b_{M-1}}, s_r$ are connected. According to line 4 of Algorithm 1, $a_1 = \max\{i | v_i \in \mathcal{N}_-(s_1)\}$. Consequently, $b_1 \leq a_1$.

Similarly, we have $a_2 = \max\{i | v_i \in \mathcal{N}_-(v_{a_1})\}$ according to line 4 in the algorithm. As v_{b_2} is connected to v_{b_1} directly, $b_1 < b_2 \leq \max\{i | v_i \in \mathcal{N}_-(v_{b_1})\}$. Since $b_1 \leq a_1$, and Assumption 1 and 2 hold, we have $\max\{i | v_i \in \mathcal{N}_-(v_{b_1})\} \leq \max\{i | v_i \in \mathcal{N}_-(v_{a_1})\} = a_2$. Consequently, we have $b_1 < b_2 \leq a_2$. Based on this, we can sequentially show that $b_2 < b_3 \leq a_3$, $b_3 < b_4 \leq a_4$, ..., $b_{M-2} < b_{M-1} \leq a_{M-1}$.

However, as $s_l, v_{b_1}, v_{b_2}, \ldots, v_{b_{M-1}}, s_r$ are connected, we have $s_r \in \mathcal{N}_-(v_{b_{M-1}})$. The latter along with $s_r \notin \mathcal{N}_-(v_{a_{M-1}})$ implies that $b_{M-1} > a_{M-1}$, which contradicts to $b_{M-2} < b_{M-1} \leq a_{M-1}$. Hence, no connected subset of $M-1$ sensor nodes exists.

For the second part, let the optimal solution of the exhaustive tree search be $v_{a_1}, v_{a_2}, \ldots, v_{a_M}$. Then, s_l is a neighbor of v_{a_1}, v_{a_M} is a neighbor of s_r and for any $1 \leq k \leq M-1$, v_{a_k} is a neighbor of $v_{a_{k+1}}$. We show next that we can reorder $a_1, \ldots, a_M$ into the set of activated nodes $b_1 < \ldots < b_M$ suggested by SAE (recursion (2)), and we will still have s_l to be a neighbor of v_{b_1}, s_r to be a neighbor of v_{b_M} and for any $1 \leq k \leq M-1$, v_{b_k} to be the neighbor of $v_{b_{k+1}}$. For consistency, we will use v_{a_0} to represent the sink node s_l, and $v_{a_{M+1}}$ to represent the sink node s_r. Suppose $a_1, a_2, \ldots, a_M$ is not ordered, then there must exist $1 \leq j < k \leq M$ such that

$$a_j > a_k, \tag{8a}$$
$$v_{a_{j+1}} \in \mathcal{N}_+(a_j), \tag{8b}$$
$$v_{a_k} \in \mathcal{N}_+(a_{k-1}). \tag{8c}$$

Then, according to (8a), there exists a $k \leq l \leq M$ such that

$$a_l < a_j < a_{l+1} \text{ or } a_{l+1} < a_j < a_l. \tag{9}$$

Since sensor nodes are deployed in a line and have the same transmission range, we have

$$\forall x, \forall v_y \in \mathcal{N}_-(v_x), \forall v_z \in \mathcal{N}_+(v_y) - \{v_x\} \quad v_z \in \mathcal{N}(v_x), \tag{10a}$$
$$\forall x, \forall v_y \in \mathcal{N}_+(v_x), \forall v_z \in \mathcal{N}_-(v_y) - \{v_x\} \quad v_z \in \mathcal{N}(v_x). \tag{10b}$$

From (9), (10a) and (10b), we have $v_{a_j} \in \mathcal{N}(v_{a_l})$ and $v_{a_j} \in \mathcal{N}(v_{a_{l+1}})$. Since $v_{a_j} \in \mathcal{N}_-(v_{a_{j-1}})$, (8b) and (10a), we have $v_{a_{j+1}} \in \mathcal{N}(v_{a_{j-1}})$. Then, we have that the nodes $s_l, \ldots, v_{a_{j-1}}, v_{a_{j+1}}, v_{a_{j+2}}, \ldots, v_{a_k}, v_{a_{k+1}}, \ldots, v_{a_l}, v_{a_j}, v_{a_{l+1}}, \ldots, s_r$ are sequentially connected.

We say that such an operation is a reordering of (j, k). Then, if $a_{j+1} > a_k$, we continue to reorder $(j+1, k)$. After at most $k - j$ re-orderings, we will have that the nodes from a_j to a_k are ordered. In consequence, we can reorder $v_{a_1}, \ldots, v_{a_M}$ in finite steps to be $v_{b_1}, \ldots, v_{b_M}$, while ensuring $\sum_{i=1}^M p_{a_i} = \sum_{i=1}^M p_{b_i}$. This concludes our proof. ∎

APPENDIX C
THE PROOF OF PROPOSITION 3

Proof: As ρ denotes the density of nodes and r the communication radius, we have that the expected number of nodes that belong to DNS and UNS of a node are $r\rho$.

The complexity of the GBS algorithm is at most N, as the while-loop from Line 2 to 8 checks all the nodes once. The expected value of M_c in line 11 of the SACC algorithm is $\lceil N/(r\rho) \rceil$. Consequently, $m = \max\{\lceil N/(r\rho) \rceil, ck \log N\}$.

For the SAE algorithm, the complexity of lines 3 to 6 is $r\rho$. The complexity of lines 7 to 12 is $mNr\rho$, where m is the value determined by the GBS Algorithm. Then, the complexity of the SAE algorithm is $O(mNr\rho)$.

From the above, it follows that the time complexity of the SACC algorithm is $O(\max\{N^2, kNr\rho \log N\})$. ∎

APPENDIX D
THE PROOF OF PROPOSITION 4

Proof: The space complexity of SAE algorithm is due to the construction of matrices $\boldsymbol{G}$ and $\boldsymbol{H}$ or the input of adjacency matrix $\boldsymbol{A}$. The size of $\boldsymbol{G}$ and $\boldsymbol{H}$ are $N \times m$, where $m = \max\{\lceil N/(r\rho) \rceil, ck \log N\}$, whereas the size of $\boldsymbol{A}$ is $N^2 > N \times m$. On the other hand, the space complexity of GBS algorithm is mainly determined by the size of adjacency matrix $\boldsymbol{A}$. In total, the space complexity of the SACC algorithm is $O(N^2)$. ∎

APPENDIX E
THE PROOF OF THEOREM 2

To prove this theorem, we first introduce the definition of 'sorted sequence' and 'sequentially connected' set.

Definition 3: **(Sorted sequence)** A sequence $\{v_{a_1}, v_{a_2}, \ldots, v_{a_m}\}$ is called sorted sequence iff $a_i < a_{i+1}, \forall 1 \leq i \leq m-1$.

Definition 4: **(Sequentially connected)** A sequence $\{v_{a_1}, v_{a_2}, \ldots, v_{a_m}\}$ is sequentially connected iff $v_{a_{i+1}} \in \mathcal{N}(v_{a_i}), \forall 1 \leq i \leq m-1$.

Also, we need the following intermediate result:

Lemma 1: Consider a network $\mathcal{G}$ that satisfies Assumption 2 and Condition (6). Given a sequentially connected sequence of nodes $\mathcal{S} = \{v_{b_1}, v_{b_2}, \ldots, v_{b_m}\}(m \geq 2)$, if a new node v_x is added into $\mathcal{S}$ between the leftmost node and the rightmost node in $\mathcal{S}$, we can then reorder the sequence of the subscript set $\{b_1, \ldots, b_m, x\}$ to be $\{b'_1, b'_2, \ldots, b'_{m+1}\}$ such that it is sequentially connected. Additionally, if the original sequence $\mathcal{S}$ is sorted, i.e., $b_1 < b_2 < \ldots < b_m$, the new sequence is also sorted and sequentially connected.

Proof: As the new node v_x is located between the leftmost node and the rightmost node in $\mathcal{S}$, there must be two nodes $v_{b_{i-1}}$ and v_{b_i} such that $b_{i-1} < x < b_i$. As $d(v_{b_{i-1}}, x) < d(v_{b_{i-1}}, v_{b_i}) \leq r_{b_{i-1}}$, we have that $v_x \in \mathcal{N}_-(v_{b_{i-1}})$. Together with $v_{b_i} \in \mathcal{N}_-(v_{b_{i-1}})$ and $b_i > x$, we have that $v_{b_i} \in \mathcal{N}_-(v_x)$ according to Condition (6). As a result, the new sequence, $\{v_{b_1}, v_{b_2}, \ldots, v_{b_{i-1}}, v_x, v_{b_i}, \ldots, v_{b_m}\}$, is connected.

Furthermore, if $\{v_{b_1}, v_{b_2}, \ldots, v_{b_{i-1}}, v_{b_i}, \ldots, v_{b_m}\}$ is sorted, since $b_{i-1} < x < b_i$, we have that the new sequence $\{v_{b_1}, v_{b_2}, \ldots, v_{b_{i-1}}, v_x, v_{b_i}, \ldots, v_{b_m}\}$ is also sorted. ∎

Based on Lemma 1, Theorem 2 is proved as follows:

Proof: Notice that Assumption 1 is a special case of Condition (6). Thus, the proof of Theorem 2 is similar to the proof for Theorem 1. We need to prove the two parts as for Theorem 1.

To prove the first part, we suppose that the GBS algrotihm returns the minimum required number of activated sensor nodes to be M and the corresponding sensor nodes to be $v_{a_1}, v_{a_2}, \ldots, v_{a_M}$ from the left to the right. It is easy to see that these sensor nodes, together with two sink nodes, are connected. According to line 2 of the GBS algorithm, we have also that $s_r \notin \mathcal{N}_-(v_{a_{M-1}})$. That is, the sink node s_r is not directly in range of sensor node $v_{a_{M-1}}$.

Then, we need to prove that there exists no connected subgraph $\mathcal{V}_A$ with less than M sensor nodes. We will prove this by contradiction. Assume that we can select $M-1$ sensor nodes, i.e. $v_{b_1}, v_{b_2}, \ldots, v_{b_{M-1}}$ from left to right, such that $s_l, v_{b_1}, v_{b_2}, \ldots, v_{b_{M-1}}, s_r$ are connected. According to line 4, $a_1 = \max\{i | v_i \in \mathcal{N}_-(s_1)\}$. Consequently, $0 \le b_1 \le a_1$.

Similarly, we have $a_2 = \max\{i | v_i \in \mathcal{N}_-(v_{a_1})\}$ according to line 4 in the GBS algorithm. As v_{b_2} is connected to v_{b_1} directly, $b_1 < b_2 \le \max\{i | v_i \in \mathcal{N}_-(v_{b_1})\}$. Since $b_1 \le a_1$ and the sensor nodes are deployed in a line and (6) holds, we have $\max\{i | v_i \in \mathcal{N}_-(v_{b_1})\} \le \max\{i | v_i \in \mathcal{N}_-(v_{a_1})\} = a_2$. Then, we have $b_1 < b_2 \le a_2$. Based on this, we can sequentially show that $b_2 < b_3 \le a_3$, $b_3 < b_4 \le a_4$, ..., $b_{M-2} < b_{M-1} \le a_{M-1}$. On the other side, as $s_l, v_{b_1}, v_{b_2}, \ldots, v_{b_{M-1}}, s_r$ are connected, we have $s_r \in \mathcal{N}_-(v_{b_{M-1}})$. Together with (6), we have $s_r \in \mathcal{N}_-(v_{a_{M-1}})$, which contradicts to $s_r \notin \mathcal{N}_-(v_{a_{M-1}})$ and completes the proof of the first part.

In the second part, we compare the result of our algorithm to that of the exhaustive tree search algorithm. Suppose the exhaustive tree search algorithm finds an optimal sequence $\mathcal{N} = \{s_l, v_{a_1}, v_{a_2}, \ldots, v_{a_m}, s_r\}$. Then, we have $v_{a_1} \in \mathcal{N}(s_l)$, $v_{a_{k+1}} \in \mathcal{N}(v_{a_k}), \forall 1 \le k \le m-1$ and $s_r \in \mathcal{N}(v_{a_m})$, which is defined as sequentially connected. We will show that by reordering $a_1, \ldots, a_m$ to $b_1 < \ldots < b_m$, we still have $v_{b_1} \in \mathcal{N}(s_l)$, $v_{b_{k+1}} \in \mathcal{N}(v_{b_k}), \forall 1 \le k \le m-1$ and $s_r \in \mathcal{N}(v_{b_m})$. As the new sequence $\{s_l, v_{b_1}, \ldots, v_{b_m}, s_r\}$ is sorted, it can be achieved by the recursion function (2).

For consistency, we use also v_0 and v_{a_0} to represent the sink node s_l, and use v_{N+1}, $v_{a_{m+1}}$ to represent the sink node s_r. Suppose $a_1, a_2, \ldots, a_m$ is not sorted. Then there must exist $1 \le j < k \le m$ such that

$$a_k < a_j \,, \tag{11a}$$

$$1 \le a_l < a_j, \quad \forall l < j \,, \tag{11b}$$

$$a_k < a_l < a_j, \quad \forall j < l < k \,, \tag{11c}$$

$$a_l < a_{l+1}, \quad \forall k \le l \le m \,, \tag{11d}$$

where (11a) represents that node v_{a_j} is on the right side of v_{a_k}, (11b) represents that all the sensor nodes whose subscript is small than a_j are located between the sink node v_{a_0} and the sensor node v_{a_j}, (11c) represents that all the sensor nodes whose subscript is between a_j and a_k are located between the sensor node v_{a_k} and v_{a_j}, and (11d) represents that the sequence $\{v_{a_k}, \ldots, v_{a_{m+1}}\}$ is sorted.

As v_{a_k} is located between the sink node v_{a_0} and the sensor node v_{a_j}, there must exist $0 \le i < j$ such that

$$a_i < a_k < a_{i+1} \,, \tag{12a}$$

$$a_k < a_l, \quad \forall i < l < j \,. \tag{12b}$$

Then we have $v_{a_k} \in \mathcal{N}_-(v_{a_i})$ as $d(v_{a_i}, v_{a_k}) < d(v_{a_i}, v_{a_{i+1}}) < r_{a_i}$. Also, according to (12a) and (12b), we have that all the sensor nodes from $v_{a_{i+1}}$ to $v_{a_{k-1}}$ are located between v_{a_k} and $v_{a_{m+1}}$. From Lemma 1 we know that we can add sensor nodes $v_{a_{i+1}}, \ldots, v_{a_{k-1}}$ into the sequence $\{v_{a_k}, v_{a_{k+1}}, \ldots, v_{a_{m+1}}\}$, such that the resulting sequence $\mathcal{S}_k^{m+1}$, which consists of $v_{a_{i+1}}, \ldots, v_{a_{k-1}}$, $v_{a_k}, v_{a_{k+1}}, \ldots, v_{a_{m+1}}$, are sorted and sequentially connected.

Then the new sequence $\mathcal{S}' = \{v_{a_0}, \ldots, v_{a_i}, \mathcal{S}_k^{m+1}\}$ is still sequentially connected with more nodes sorted, i.e., at least the last $m - i + 2$ nodes in $\mathcal{S}'$ are sorted whereas only $m - k + 2$ nodes are sorted in the original sequence $\mathcal{S}$. Consequently, through such reordering operations, we will finally find a new sequence, whose elements are the same with $\mathcal{S}$, and the new sequence is sorted and sequentially connected. $\{s_l, v_{b_1}, v_{b_2}, \ldots, v_{b_m}, s_r\}$ is such a sequence, and it can be achieved from the recursive function (2) as it is sorted and sequentially connected. This concludes the proof. $\blacksquare$

REFERENCES

[1] J. Skadsen, R. Janke, W. Grayman, W. Samuels, M. Tenbroek, B. Steglitz, and S. Bahl, "Distribution system on-line monitoring for detecting contamination and water quality changes," *Journal of American Water Works Association* Vol. 100, No. 7, pp. 81–94, 2008.

[2] A. Whittle, M. Allen, A. Preis, and M. Iqbal, "Sensor networks for monitoring and control of water distribution systems," in *Proc. 6th International Conference on Structural Health Monitoring of Intelligent Infrastructure*, 2013, pp. 9–11.

[3] A. J. Whittle, L. Girod, A. Preis, M. Allen, H. B. Lim, M. Iqbal, S. Srirangarajan, C. Fu, K. J. Wong, and D. Goldsmith, "Waterwise@ SG: A testbed for continuous monitoring of the water distribution system in singapore," *Water Distribution System Analysis*, 2010, pp. 1362–1378.

[4] I. Stoianov, L. Nachman, S. Madden, and T. Tokmouline, "Pipenet: A wireless sensor network for pipeline monitoring," in *Proc. of IEEE International Symposium on Information Processing in Sensor Networks (IPSN)*, 2007, pp. 264–273.

[5] A. Speranzon, C. Fischione, K. H. Johansson, and A. Sangiovanni-Vincentelli, "A distributed minimum variance estimator for sensor networks," *IEEE Journal on Selected Areas in Communications*, vol. 26, no. 4, pp. 609–621, 2008.

[6] I. F. Akyildiz and E. P. Stuntebeck, "Wireless underground sensor networks: Research challenges," *Ad Hoc Networks*, vol. 4, no. 6, pp. 669 – 686, 2006.

[7] A. Krause and C. Guestrin, "Submodularity and its applications in optimized information gathering," *ACM Trans. on Intelligent Systems and Technology (TIST)*, vol. 2, no. 4, p. 32, 2011.

[8] S. Guha, P. Basu, C.-K. Chau, and R. Gibbens, "Green wave sleep scheduling: Optimizing latency and throughput in duty cycling wireless networks," *IEEE Journal on Selected Areas in Communications*, vol. 29, no. 8, pp. 1595–1604, 2011.

[9] G. Wei, Y. Ling, B. Guo, B. Xiao, and A. V. Vasilakos, "Prediction-based data aggregation in wireless sensor networks: Combining grey model and kalman filter," *Computer Communications*, vol. 34, no. 6, pp. 793–802, 2011.

[10] R. Du, L. Gkatzikis, C. Fischione, and M. Xiao, "Energy efficient monitoring of water distribution networks via compressive sensing," in *Proc. of IEEE International Conference on Communications (ICC)*, 2015, pp. 8309–8314.

[11] J.-H. Chang and L. Tassiulas, "Maximum lifetime routing in wireless sensor networks," *IEEE/ACM Trans. on Networking (TON)*, vol. 12, no. 4, pp. 609–619, 2004.

[12] A. Jarry, P. Leone, O. Powell, and J. Rolim, "An optimal data propagation algorithm for maximizing the lifespan of sensor networks," in *Distributed Computing in Sensor Systems*, Springer, 2006, pp. 405–421.

[13] F. Ren, J. Zhang, T. He, C. Lin, and S. K. Ren, "EBRP: energy-balanced routing protocol for data gathering in wireless sensor networks," *IEEE Trans. on Parallel and Distributed Systems*, vol. 22, no. 12, pp. 2108–2125, 2011.

[14] C. Cassandras, T. Wang, and S. Pourazarm, "Optimal routing and energy allocation for lifetime maximization of wireless sensor networks with nonideal batteries," *IEEE Trans. on Control of Network Systems*, vol. 1, no. 1, pp. 86–98, March 2014.

[15] A. Bertoldi, G. Athanasiou, and C. Fischione, "Optimal sensor placement for bacteria detection in water distribution networks," in *Proc. of IEEE International Symposium on Communications, Control and Signal Processing (ISCCSP)*, 2014, pp. 248–253.

[16] J.-P. Watson, H. J. Greenberg, and W. E. Hart, "A multiple-objective analysis of sensor placement optimization in water networks," in *Proc. of ASCE World Water Environmental Resources Congress*, 2004, pp. 456–465.

[17] W. E. Hart and R. Murray, "Review of sensor placement strategies for contamination warning systems in drinking water distribution systems," *Journal of Water Resources Planning and Management*, vol. 136, no. 6, pp. 611–619, 2010.

[18] J. W. Berry, L. Fleischer, W. E. Hart, C. A. Phillips, and J.-P. Watson, "Sensor placement in municipal water networks," *Journal of Water Resources Planning and Management*, vol. 131, no. 3, pp. 237–243, 2005.

[19] L. Liu, Y. Song, H. Zhang, H. Ma, and A. V. Vasilakos, "Physarum optimization: A biology-inspired algorithm for the steiner tree problem in networks," *IEEE Trans. on Computers*, vol. 64, no. 3, pp. 819–832, 2015.

[20] L.-H. Yen, C.-M. Lin, and V. Leung, "Distributed lifetime-maximized target coverage game," *ACM Trans. on Sensor Networks (TOSN)*, vol. 9, no. 4, p. 46, 2013.

[21] G. Degirmenci, J. P. Kharoufeh, and O. A. Prokopyev, "Maximizing the lifetime of query-based wireless sensor networks," *ACM Trans. on Sensor Networks (TOSN)*, vol. 10, no. 4, pp. 56:1–56:24, Jun. 2014.

[22] M. A. Razzaque, C. Bleakley, and S. Dobson, "Compression in wireless sensor networks: A survey and comparative evaluation," *ACM Trans. on Sensor Networks (TOSN)*, vol. 10, no. 1, p. 5, 2013.

[23] C. Luo, F. Wu, J. Sun, and C. W. Chen, "Compressive data gathering for large-scale wireless sensor networks," in *Proc. of ACM International Conference on Mobile computing and networking*, 2009, pp. 145–156.

[24] F. Fazel, M. Fazel, and M. Stojanovic, "Random access compressed sensing for energy-efficient underwater sensor networks," *IEEE Journal on Selected Areas in Communications*, vol. 29, no. 8, pp. 1660–1670, September 2011.

[25] L. Xu, X. Qi, Y. Wang, and T. Moscibroda, "Efficient data gathering using compressed sparse functions," in *Proc. of IEEE International Conference on Computer Communications*, 2013, pp. 310–314.

[26] E. J. Candès, J. Romberg, and T. Tao, "Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information," *IEEE Trans. on Information Theory*, vol. 52, no. 2, pp. 489–509, 2006.

[27] L. Xiang, J. Luo, C. Deng, A. V. Vasilakos, and W. Lin, "Deca: Recovering fields of physical quantities from incomplete sensory data," in *Proc. of IEEE Communications Society Conference on Sensor, Mesh and Ad Hoc Communications and Networks (SECON)*, 2012, pp. 182–190.

[28] L. Xiang, J. Luo, and A. Vasilakos, "Compressed data aggregation for energy efficient wireless sensor networks," in *Proc. of IEEE conference on Sensor, mesh and ad hoc communications and networks (SECON)*, 2011, pp. 46–54.

[29] X. Xu, R. Ansari, A. Khokhar, and A. V. Vasilakos, "Hierarchical data aggregation using compressive sensing (HDACS) in wsns," *ACM Trans. on Sensor Networks (TOSN)*, vol. 11, no. 3, p. 45, 2015.

[30] X.-Y. Liu, Y. Zhu, L. Kong, C. Liu, Y. Gu, A. Vasilakos, and M.-Y. Wu, "CDC: Compressive data collection for wireless sensor networks," *IEEE Trans. on Parallel and Distributed Systems*, vol. PP, no. 99, pp. 1–11, 2014.

[31] Y. Yao, Q. Cao, and A. Vasilakos, "EDAL: An energy-efficient, delay-aware, and lifetime-balancing data collection protocol for heterogeneous wireless sensor networks," *IEEE/ACM Trans. on Networking*, vol. 23, no. 3, pp. 810–823, 2015.

[32] S. Xiong, L. Yu, H. Shen, C. Wang, and W. Lu, "Efficient algorithms for sensor deployment and routing in sensor networks for network-structured environment monitoring," in *Proc. of IEEE International Conference on Computer Communications*, 2012, pp. 1008–1016.

[33] Z. Sun, P. Wang, M. C. Vuran, M. A. Al-Rodhaan, A. M. Al-Dhelaan, and I. F. Akyildiz, "Mise-pipe: Magnetic induction-based wireless sensor networks for underground pipeline monitoring," *Ad Hoc Networks*, vol. 9, no. 3, pp. 218–227, 2011.

[34] J. Yang, X. Wu, and J. Wu, "Optimal Scheduling of Collaborative Sensing in Energy Harvesting Sensor Networks," *IEEE Journal on Selected Areas in Communications*, vol. 33, no. 3, pp. 512–523, 2015.

[35] Y. C. Eldar and G. Kutyniok, *Compressed sensing: theory and applications*, Cambridge University Press, 2012.

[36] D. S. Hochbaum and A. Pathria, "Node-optimal connected k-subgraphs," *UC Berkeley*, 1994.

[37] D. P. Bertsekas, *Dynamic programming and optimal control*, Athena Scientific Belmont, MA, 1995, vol. 1, no. 2.

[38] J. A. Tropp and A. C. Gilbert, "Signal recovery from random measurements via orthogonal matching pursuit," *IEEE Trans. on Information Theory*, vol. 53, no. 12, pp. 4655–4666, 2007.

Rong Du (S'12) received Bachelor and Master degrees in Automatic Control from Shanghai Jiao Tong University in 2011 and 2014, respectively. He is currently working toward the Ph.D. degree with the School of Electrical Engineering and ACCESS Linnaeus Center, KTH Royal Institute of Technology, Sweden. His current research interests include wireless sensor network for smart city, and wireless energy transfer.

Lazaros Gkatzikis (S'09, M'13) is a research staff member at the Huawei France Research Center, Paris. He obtained his Ph.D degree from the department of computer engineering and communications, University of Thessaly. He has also received a PhD scholarship from the NSRF-Heraclitus II program. In fall 2011 he was a research intern at the Technicolor Paris research lab. He was a post-doctoral researcher at University of Thessaly, Volos, Greece (2013) and at KTH royal institute of technology, Stockholm, Sweden (2014). His research interests include network optimization, game theory and performance analysis.

Carlo Fischione (M'05) is currently a tenured Associate Professor at KTH Royal Institute of Technology, Electrical Engineering and ACCESS Linnaeus Center, Stockholm, Sweden. He received the Ph.D. degree in Electrical and Information Engineering (3/3 years) in May 2005 from University of L'Aquila, Italy, and the Laurea degree in Electronic Engineering (Laurea, Summa cum Laude, 5/5 years) in April 2001 from the same University. He has held research positions at Massachusetts Institute of Technology, Cambridge, MA (2015, Visiting Professor), Harvard University Cambridge, MA (Associate, 2015), University of California at Berkeley, CA (2004-2005, Visiting Scholar, and 2007-2008, Research Associate) and Royal Institute of Technology, Stockholm, Sweden (2005-2007, Research Associate). His research interests include optimization with applications to wireless sensor networks, networked control systems, wireless networks, security and privacy. He has co-authored over 100 publications, including a book, book chapters, international journals and conferences, and four international patents. He received or co-received a number of awards, including the best paper award from the IEEE Transactions on Industrial Informatics (2007), the best paper awards at the IEEE International Conference on Mobile Ad-hoc and Sensor System 05 and 09 (IEEE MASS 2005 and IEEE MASS 2009), the Best Paper Award of the IEEE Sweden VT-COM-IT Chapter (2014), the Best Business Idea awards from VentureCup East Sweden (2010) and from Stockholm Innovation and Growth (STING) Life Science in Sweden (2014), the "Ferdinando Filauro" award from University of L'Aquila, Italy (2003), the "Higher Education" award from Abruzzo Region Government, Italy (2004), the Junior Research award from Swedish Research Council (2007), the Silver Ear of Wheat award in history from the Municipality of Tornimparte, Italy (2012). He is Associated Editor of Elsevier Automatica, has chaired or served as a technical member of program committees of several international conferences and is serving as referee for technical journals. Meanwhile, he also has offered his advice as a consultant to numerous technology companies such as Berkeley Wireless Sensor Network Lab, Ericsson Research, Synopsys, and United Technology Research Center. He is co-funder and CTO of the sensor networks start-up company MIND (ancient and modern musical instruments networked). He is Member of IEEE (the Institute of Electrical and Electronic Engineers), and Ordinary Member of DASP (the academy of history Deputazione Abruzzese di Storia Patria).
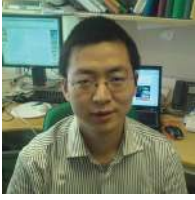
Ming Xiao (S'02-M'07-SM'12) received Bachelor and Master degrees in Engineering from the University of Electronic Science and Technology of China, ChengDu in 1997 and 2002, respectively. He received Ph.D degree from Chalmers University of technology, Sweden in November 2007. From 1997 to 1999, he worked as a network and software engineer in ChinaTelecom. From 2000 to 2002, he also held a position in the SiChuan communications administration. From November 2007 to now, he has been in Communication Theory, school of electrical engineering, Royal Institute of Technology, Sweden, where he is currently an Associate Professor in Communications Theory. He received the best paper Awards in "IC-WCSP" (International Conference on Wireless Communications and Signal Processing) in 2010 and "IEEE ICCCN" (International Conference on Computer Communication Networks) in 2011. Dr. Xiao received "Chinese Government Award for Outstanding Self-Financed Students Studying Aborad" in March, 2007. He got "Hans Werthen Grant" from royal Swedish academy of engineering science (IVA) in March 2006. He received "Ericsson Research Funding" from Ericsson in 2010. Since 2012, he has been an Associate Editor for IEEE Transactions on Communications, IEEE Communications Letters (Senior Editor Since Jan. 2015) and IEEE Wireless Communications Letters.