

# Approximation Algorithms for Capacitated Minimum Forest Problems in Wireless Sensor Networks with a Mobile Sink

Weifa Liang, *Senior Member, IEEE*, Pascal Schweitzer, and Zichuan Xu, *Student Member, IEEE*

**Abstract**—To deploy a wireless sensor network for the purpose of large-scale monitoring, in this paper, we propose a heterogeneous and hierarchical wireless sensor network architecture. The architecture consists of sensor nodes, gateway nodes, and mobile sinks. The sensors transmit their sensing data to the gateway nodes for temporary storage through multihop relays, while the mobile sinks travel along predetermined trajectories to collect data from nearby gateway nodes. Under this paradigm of data gathering, we formulate a novel constrained optimization problem, namely, the capacitated minimum forest (CMF) problem, for the decision version of which we first show NP-completeness. We then devise approximation algorithms and provide upper bounds for their approximation ratios. We finally evaluate the performance of the proposed algorithms through experimental simulation. In our experiments, the approximation ratio delivered by the proposed algorithms is always less than 2. In the case of arbitrary gateway capacities, this contrasts our theoretical results which show that the approximation ratio is at most linear in the number of gateways. Our experiments thus indicate that for realistic inputs, our worst case analysis of the approximation ratio is very conservative. The proposed algorithms are the first approximation algorithms for the CMF problem, and our techniques may be applicable to other constrained optimization problems beyond wireless sensor networks.

**Index Terms**—Wireless sensor networks, sink mobility, data gathering, constrained optimization problem, capacitated minimum forest problem



## 1 INTRODUCTION

THE application of wireless sensor networks has emerged as a promising solution to large-scale tracking and monitoring tasks [1]. Due to their low-data rate, low-energy consumption, and short-range communication, wireless sensor networks present the great opportunity to instrument and monitor the physical world at unprecedented scale and resolution. In conventional sensor networks, there is a single stationary sink (a base station), that has unlimited power supply serving as a gateway between sensors and users. This stationary sink data gathering paradigm however suffers from the following two major drawbacks. The first drawback arises from unbalanced energy consumption among the sensors. Since the sensors that are close to the sink have to relay data for other, more remote sensors, they usually bear disproportionate amounts of traffic, and thus deplete their energy much faster than others. The resulting unbalanced energy consumption shortens the network operational time and thereby affects data delivery reliability and other network performance aspects. The second drawback is the required network connectivity. It is necessary that a network consisting of a stationary sink and sensors is

connected; otherwise, the data generated by the sensors that lie in a component of the network not containing the sink cannot be transferred to the sink and is ultimately lost. In some deployments of sparse sensor networks, it is very difficult to ensure network connectivity due to the restriction of physical obstacles or other geographic constraints (e.g., water ponds, rocks, etc.). To balance the energy consumption among sensors and reduce the dependence of network connectivity, the concept of mobile sinks has been exploited. Recent studies show that the use of mobile sinks can significantly improve various performance aspects of a wireless sensor network including network lifetime, connectivity, data delivery reliability, throughput, and so on [3], [18], [20], [25], [29], [28], [34].

Deploying a wireless sensor network with mobile sinks for large-scale monitoring while minimizing its critical network resource consumptions is equivalent to solving *constrained optimization problems* with one or multiple constraints [15]. Previous research on these optimization problems mainly focuses on developing exact and heuristic algorithms [3], [17], [20], [21], [26], [27], [29]. However, since most of these problems are NP-hard, exact procedures are limited to solving small-size instances and not applicable to large-scale networks due to exponential running time. Although heuristics can sometimes yield adequate solutions, they do not provide any guarantee on how far the solutions are from being optimal. Thus, developing fast approximate solutions with guaranteed approximation ratios becomes an important research issue for large-scale wireless sensor networks.

Several approximation algorithms for various optimization problems in wireless sensor networks with fixed sinks,

- W. Liang and Z. Xu are with the Research School of Computer Science, The Australian National University, Canberra ACT 0200, Australia. E-mail: wliang@cs.anu.edu.au, edward.xu@anu.edu.au.
- P. Schweitzer is with the Institut Mittag-Leffler, Djursholm, SE-182 60, Sweden. E-mail: pascal.schweitzer@anu.edu.au.

Manuscript received 23 Oct. 2011; revised 1 May 2012; accepted 6 May 2012; published online 4 June 2012.

Recommended for acceptance by K. Roemer.

For information on obtaining reprints of this article, please send e-mail to: [tc@computer.org](mailto:tc@computer.org), and reference IEEECS Log Number TC-2011-10-0755. Digital Object Identifier no. 10.1109/TC.2012.124.

have been proposed in the past few years [14], [23], [30], [31]. In contrast to this, developing approximate solutions for optimization problems in sensor networks with mobile sinks is largely unexplored, and there are only very few results available [19], [25]. In this paper, we focus on developing approximation algorithms for an important constrained optimization problem in wireless sensor networks with a mobile sink, namely, the capacitated minimum forest (CMF) problem: given  $m$  gateway nodes with each gateway  $g$  having an integer capacity  $c(g) \geq 0$ , the problem is to find a minimum cost forest consisting of routing trees rooted at gateways and spanning sensor nodes, subject to the constraint that each tree rooted at a gateway  $g$  spans exactly  $c(g)$  sensor nodes.

The main contributions of this paper are as follows: We first consider the deployment of wireless sensor networks with a mobile sink for large-scale monitoring, by proposing a heterogeneous and hierarchical architecture, where the mobile sink travels along a predetermined trajectory for data collection. Under this paradigm of data gathering, we then formulate a novel, constrained optimization problem, namely, the CMF problem, for which we devise approximation algorithms with guaranteed approximation ratios. Finally, we evaluate the performance of the proposed algorithms through experimental simulation. In our experiments, the approximation ratio delivered by the proposed algorithms is always less than 2. In the case of arbitrary gateway capacities, this contrasts our theoretical results which show that the approximation ratio is at most linear in the number of gateways. The proposed algorithms are the first approximation algorithms for this fundamental problem, and our techniques may be applicable to other constrained optimization problems beyond wireless sensor networks.

The remainder of the paper is organized as follows: We first introduce the related work in Section 2, followed by introducing the system model and problem definition in Section 3. We then show that the CMF problem is NP-complete in Section 4. We then devise approximation algorithms for the CMF problem and analyze the approximation ratios of the proposed algorithms in Sections 5, 6, and 7. We finally conduct experiments through simulation to evaluate the performance of the proposed algorithms in Section 8, and we conclude our discussion in Section 9.

## 2 RELATED WORK

Extensive studies on optimizing critical network resources in wireless sensor networks with mobile sinks, such as maximizing the network lifetime and/or minimizing the number of mobile sinks employed, have been conducted in the past few years. For example, the studies in [3], [18], [19], [20], [25], [29], [31], and [35] focus on the network lifetime maximization, while other studies focus on minimizing the travel distance of mobile sinks [21]. Very few take both of the aspects into consideration [16], [17]. Most of these studies are based on homogeneous sensor networks that consist only of one type of sensor. Although the homogeneous architecture works very well for small to medium-size networks, it may not be appropriate for large-scale monitoring due to poor scalability, long data delivery delay,

and so on. With the increase of network size, the average length of routing paths from remote sensors to the mobile sink(s) (in terms of the number of hops) is longer, and the chance of link failures increases, leading to a much longer data delivery delay.

When deploying wireless sensor networks for large-scale monitoring to mitigate the drawbacks of homogeneous architectures, heterogeneous sensor networks with mobile sinks have been introduced and studied [7], [13], [26], [31], [32], [33]. In [13] and [26], it is assumed that the speed of each mobile sink is controllable. This implies that the amount of data collected by each mobile sink from a gateway is controllable through adjusting the speed of the mobile sink. In [31], Xing et al. consider the network lifetime maximization problem by devising an approximation algorithm for finding an optimal trajectory for the mobile sink, subject to the trajectory length constraint. The approximate solution obtained is based on a tree routing structure with the assumption that the forwarding load of each relay node is identical, independent of the number of descendants the relay node has. In contrast to this, by assuming that the mobile sink travels along a predetermined trajectory with constant speed, Gao et al. [7] develop a genetic algorithm for allocating sensor nodes to different gateway nodes to form a forest that minimizes the total energy consumption of sensors when routing sensing data. Based on assumptions similar to those in [7], Xu et al. [32] and [33] propose several heuristics for finding a forest of routing trees that minimizes the routing cost, by incorporating data correlation among sensors. However, there is no guaranteed bound on how far the solutions obtained are from being optimal.

Developing approximation algorithms with adequate bounds on the approximation ratios for the CMF problem is challenging due to the stringent constraint on gateway capacities. There are two classical optimization problems closely related to the CMF problem: the *capacitated minimum spanning tree (CMST) problem* [24]; and the *capacitated minimum Steiner tree problem (CMStT)* in wired networks, which are defined as follows. Given an edge-weighted graph, a root node, a set of *source nodes*, and a capacity  $c$ , the problem is to construct a minimum-cost spanning or Steiner tree rooted at the root node, spanning all source nodes, subject to the number of nodes in the subtree of each child of the root being no more than  $c$ . CMST is a network design problem that has been extensively studied in computer science and operation research over the past 40 years [2], [12]. At first sight, it may seem that CMF with uniform capacities can be reduced to CMST by adding a virtual "root node," connecting all gateway nodes to the root node, and then solving CMST. However, there are essential differences between CMF and CMST (or CMStT). In particular, the addition of a root node connected to all gateways in a CMF instance may change the cost of an optimal solution by more than a constant factor. Also, the number of children of the root in a CMST solution is not predetermined. The root can have an arbitrary number of children as long as the sum of source nodes in the subtree rooted at each child is no more than the specified capacity  $c$ . Contrarily to this, the number of gateway nodes in CMF is given beforehand.

Also, in a CMST solution different children of the root may have different capacities, whereas the CMF problem requires that the number of source nodes routed to a gateway must be equal to the capacity of the gateway. Thus it seems, that the state-of-the-art approximation technique for CMST or CMStT [12] cannot directly be applied to solve the CMF problem, and a different algorithm design and analysis technique is required.

### 3 PRELIMINARIES

#### 3.1 System Model

Suppose a stationary sensor network that consists of a large number of low-cost sensor nodes for sensing and a few powerful, large-storage gateway nodes has been deployed for the purpose of monitoring a region of interest. The sensing data generated by the sensors will be immediately relayed to their nearby gateways. The gateways are used to store the sensing data temporarily, perform data aggregation if needed, and eventually transmit the stored data to mobile sinks when the mobile sinks are within their transmission range. There is no energy constraint on the gateways, assuming that they can be recharged either by mobile sinks or renewable energy sources such as solar energy.

We assume that there are a number of mobile sinks that travel along predetermined trajectories (tours) to collect data from the gateways on the trajectories. We further assume that the speed at which a mobile sink traverses along its trajectory is fixed and does not vary from tour to tour. The data collected by the mobile sinks is finally uploaded to a mainframe computer for further processing. In other words, this heterogeneous and hierarchical wireless sensor network architecture consists of three-tiers: the top tier, consisting of a mobile sink (or a set of mobile sinks) used to collect data directly from the gateways; the bottom tier, consisting of many sensors that sense and transmit data to the gateways; and the middle tier, consisting of gateway nodes storing sensing data temporarily and transmitting the stored data to mobile sinks. The advantage of this heterogeneous and hierarchical wireless sensor network architecture is its capability to deliver a desired tradeoff between the energy consumption of sensors and the data delivery latency, making it appropriate for large-scale monitoring.

For simplicity, in the remainder of this paper, we assume that there is only one mobile sink. However, our discussion and the presented approximation algorithms can be adapted to networks with multiple mobile sinks.

#### 3.2 Motivations

Suppose a wireless sensor network has been deployed and a mobile sink traveling along a predetermined trajectory collects data from the gateways that are located near the trajectory. We assume that the trajectory of the mobile sink is fixed, so is its speed. This implies that the time spent by the mobile sink per tour  $\tau$  is constant. It is equal to the length of the trajectory divided by the speed of the mobile sink. As the mobile sink moves along the predetermined trajectory, a gateway can transmit its stored data to the sink whenever the sink is within its

transmission range. Since the duration of the sink within a gateway's vicinity is limited, the amount of data collected by the mobile sink from the gateway per tour is limited as well. Here we assume that the data transmission rate  $\lambda_g$  of each gateway  $g$  is fixed. Let  $t_g$  be the duration of the sink within the transmission range of gateway  $g$ . Then, the amount of data collected by the mobile sink from gateway  $g$  per tour is bounded by  $D_{\text{out}}(g) = \lambda_g \cdot t_g$ . Meanwhile, the amount of data received by a gateway  $g$  during a time interval of length  $\tau$  is  $D_{\text{in}}(g) = \tau \cdot \lambda_s \cdot n_g$ , where  $\lambda_s$  is the data generation rate of sensors and  $n_g$  is the number of sensors routing their data to gateway  $g$ . If the volume of the data stored at gateway  $g$  is larger than the amount the mobile sink can collect per tour, i.e., if  $D_{\text{in}}(g) > D_{\text{out}}(g)$ , then data loss is ultimately unavoidable. The data loss may compromise the quality of monitoring of the network, since not all sensing data is collected by the mobile sink. To maintain high-quality monitoring using the sensor network, the problem we face is to decide that which portion of the sensing data should be transmitted to the mobile sink. A naive approach to deal with this issue is to pick a portion of the data stored in gateway  $g$  at random and transmit only this portion of data when the mobile sink passes by. However, this can result in very poor data quality. Indeed, since usually the sensors in wireless sensor networks are randomly and densely deployed, the data generated by the sensors is spatially and temporally correlated. A smarter way to ensure high-quality monitoring is to make use of data correlation by choosing a subset of sensors and routing their sensing data to gateways such that all the data stored at the gateways will be collected by the mobile sink at its next tour. The collected data (from these chosen sensors) in the time duration  $\tau$  will be used to approximate the sensing data of all sensors in the network. The problem of choosing such a subset of sensors has been extensively studied in the past several years, using different data quality metrics such as the mean of the squared prediction error. As argued by Golovin et al. [10], in practical applications the objective function (i.e., the mean of the squared prediction error) is submodular and thus, as a consequence of a theoretical analysis by Nemhauser et al. [22], a simple greedy strategy yields a solution that is a  $(1 - 1/e) \approx 0.632$ -approximation of the optimum.

In this paper, we assume that the subset of sensors has been identified, and we focus on assigning the chosen sensors to different gateways such that the total routing cost is minimized, subject to following gateway capacity constraints. Denote by  $c(g)$  the *capacity* of a gateway  $g$ , which is the number of sensors that can continuously transmit their data to  $g$  within the duration of  $\tau$ . The value of  $c(g)$  is determined by the transmission rate  $\lambda_g$  of gateway  $g$  and the time duration  $t_g$  of the mobile sink within the transmission range of gateway  $g$ , i.e.,  $c(g) = \lfloor D_{\text{out}}(g)/\tau \rfloor = \lfloor t_g \cdot \lambda_g \rfloor$ . Here we assume without loss of generality that a single reading (i.e., one unit of sensing data) is generated per time unit at every sensor.

To route the sensing data from the chosen sensors to the gateways, we will adopt routing tree structures. Assuming that there are  $m$  gateway nodes, a collection of  $m$  routing

trees needs to be found. Each tree contains exactly one gateway node, which is its root, and together the trees span the gateways and the set of chosen sensors. To minimize the cost of routing the sensing data generated by the chosen sensors to the gateway nodes, extra relay nodes that are not among the chosen sensors may be employed. This enables the communication of chosen sensors and with gateways that are not within the transmission range and also allows us to minimize the number of relay nodes that are needed. We will use the Euclidean distance between two nodes (chosen sensor nodes or gateway nodes) to approximate the number of relay nodes needed. This approximation has been justified in densely deployed sensor networks (see [31]).

### 3.3 Problem Statement

Suppose  $G = (V \cup W, E)$  is a subnetwork, of a wireless sensor network whose sensors are densely, randomly deployed in a region of the plane, where  $V$  is the set of chosen sensor nodes with  $n = |V|$  and  $W = \{g_1, \dots, g_m\}$  is the set of gateway nodes with  $m = |W|$ . In practical applications  $n$  will be orders of magnitude larger than  $m$ . There is a link in  $E$  between any two sensor nodes and between every gateway and every sensor node. The weight (or the cost)  $d(u, v)$  associated with the link between  $u$  and  $v$  is their Euclidean distance. Each gateway node  $g \in W$  has an integer capacity  $c(g)$  and  $n = \sum_{i=1}^m c(g_i) = |V|$ . The CMF problem in  $G$  is to find a minimum cost forest consisting of routing trees rooted at the gateway nodes, subject to the number of chosen sensors in the tree rooted at  $g_i \in W$  being equal to its capacity  $c(g_i)$  for all  $i$  with  $1 \leq i \leq m$ . We refer to the case where all gateways have identical capacities as the case of *uniform capacities*; otherwise, we say the gateways have *arbitrary capacities*.

Given an optimization problem, denote by *Approx* and *OPT* the costs of the approximate solution delivered by an approximation algorithm  $\mathcal{A}$ , and the optimal solution to the problem respectively, we say that the approximation ratio of algorithm  $\mathcal{A}$  is  $\xi$  if  $\frac{\text{Approx}}{\text{OPT}} \leq \xi$  when the problem is a minimization problem, or  $\frac{\text{Approx}}{\text{OPT}} \geq \xi$  when the problem is a maximization problem. Throughout the paper, for any subgraph  $H$  of  $G$ , we will denote by  $c(H)$  the cost of  $H$ , which is the cost sum of all edges in  $H$ . Thus, the task is to find a suitable forest  $F$  which minimizes  $c(F)$ .

## 4 NP-COMPLETENESS OF CMF

In this section, we show that the CMF problem in the Euclidean plane is NP-complete. For this, we adopt an approach taken by Papadimitriou [24] to show NP-completeness of the CMST problem. Following Papadimitriou, we will assume that the distance between two sensors is taken as the integral part of their distance in the Euclidean plane.

**Theorem 1.** *The decision version of the CMF problem in the Euclidean plane (with integer part distances) is NP-complete.*

**Proof.** We first show that the problem is in NP. Since we assume that the distance between two sensor nodes is the integral part of their Euclidean distance, the cost of a given routing forest can be computed in polynomial

time. Thus, given the optimal routing structure, the cost of the optimal solution can be verified in polynomial time and the problem is in NP.  $\square$

We now show that the problem is NP-hard by a reduction from a special CMST problem in the Euclidean plane.

In general, the CMST problem is defined as follows: Given a set of source nodes in the plane and a distinguished root node, the problem is to construct a spanning tree rooted at the root node, such that the sum of the costs of the edges in the tree is minimized, subject to the requirement that each component (tree) of the forest obtained by deleting the root node contains at most  $c$  vertices.

In the decision version, an integer  $K$  is given and the problem asks for the existence of a solution of cost at most  $K$ . Papadimitriou [24] shows that this decision version of the CMST problem in the Euclidean plane is NP-hard via a reduction of the exact set cover problem. In fact his proof shows the NP-hardness on a special type of instance. We will use two properties of this special type of instance, which are as follows:

1. In any solution of cost at most  $K$  the components of the forest obtained by deleting the root node have exactly  $c$  nodes (as opposed to having at most  $c$  nodes). Consequently, there is an integer  $m$  such that the input consists of  $mc + 1$  vertices, including the root.
2. The root has coordinates  $(x, 0)$  for some  $x \in \mathbb{Z}$ , no other node has the same first coordinate as the root, and for every nonroot node  $v$  the integer part of the distances from  $v$  to position  $(x, i)$  are equal for all  $i \in \{0, \dots, m - 1\}$ .

We now reduce an instance of the CMST problem of this special type to an instance of CMF. Suppose the instance of CMST is given by a set of nodes  $V$  that excludes the root, a root node  $s$ , and a capacity  $c$ . By the first property, there must be an integer  $m$  such that  $mc = |V|$ . We design a CMF instance that consists of the set of sensor nodes  $V$  and  $m$  gateways located at positions  $(x, i)$  for  $i \in \{0, \dots, m - 1\}$ . The capacity of each gateway is set to  $c$ .

We finish the proof by showing that the CMF instance with uniform capacity  $c$  has a solution of cost  $K$  if and only if the original instance of the CMST problem has a solution of cost  $K$ . If  $F$  is a solution to the CMF problem, then we obtain a solution to the CMST problem by moving all gateways to position  $(x, 0)$ . Since each component of the forests  $F$  contains exactly one gateway, this turns the graph into a tree. By Property 2, the cost of this tree is the same as the cost of  $F$ . Conversely, let  $T$  be a solution to the CMST problem. We remove the root  $s$  and replace it with  $m$  gateways in positions  $(x, i)$  for  $i \in \{0, \dots, m - 1\}$ . Every vertex that used to be adjacent to the root is assigned to one of the gateways and we add an edge between each such vertex and its assigned gateway. We obtain a forest that is a valid solution to the CMF problem whose cost is equal to the cost of  $T$ .  $\square$

We remark that taking the integer parts of actual Euclidean distances is not only applied in theory, but is also common in practice when evaluating algorithms (see [6], [11]).

## 5 APPROXIMATION ALGORITHMS FOR CMF WITH THE MST COST ASSUMPTION

In the previous section, we have shown that the CMF problem is NP-complete. We now develop approximation algorithms for this problem. We assume that the set of chosen sensors  $V$  is given, and we treat the wireless sensor network as an undirected graph  $G = (V \cup W, E)$  in the Euclidean plane, where  $W$  is the set of gateway nodes. The edge set  $E = (V \times V) \cup (V \times W)$  contains the edges between all sensors nodes and gateways. The weight  $d(u, v)$  associated with every edge  $(u, v)$  is equal to the Euclidean distance between the two endpoints  $u$  and  $v$ . We are also given a capacity function  $c : W \rightarrow \mathbb{N}$  that assigns every gateway  $g \in W$  a capacity  $c(g)$  such that  $\sum_{g \in W} c(g) = |V| = n$ .

Throughout this section we assume that for  $G$  there is a minimum spanning tree (MST)  $T$  of the graph induced by the set of chosen sensors  $V$  such that the cost  $c(T)$  of  $T$  is no greater than the cost  $OPT$  of the optimal solution to the problem (CMF), i.e.,  $c(T) \leq OPT$ . We refer to this assumption as the *MST cost assumption*. Although in practice this assumption is reasonable in many densely deployed networks, we will show in Section 6 how to extend the approximation algorithm to work on arbitrary networks not necessarily fulfilling the MST cost assumption.

In the following, we first provide an overview of our approximation algorithms for CMF. We then devise an approximation algorithm for CMF and analyze its approximation ratio. In the analysis of the approximation ratio, we distinguish two cases: the case where all gateways have the same capacity and the case where the capacities may be arbitrary.

### 5.1 An Overview of our Approximation Algorithms

Before providing details, we first describe the strategy behind the approximation algorithm for graphs with the MST cost assumption. Following the MST cost assumption, there is an MST  $T$  in the complete graph induced by the chosen sensors with  $c(T) \leq OPT$ . A traveling salesman tour (TSP tour)  $L$  based on  $T$  can be found by visiting the nodes in  $T$  in a preorder tree walk. The cost  $c(L)$  of  $L$  is no more than  $2c(T)$ , due to the triangle inequality (see [5]).

To route the sensing data from the chosen sensors to the  $m$  gateway nodes, the TSP tour  $L$ , consisting of all chosen sensors, is partitioned into  $m$  segments such that the number of sensors contained in each segment equals the capacity of a corresponding gateway node, i.e., if  $\mathcal{Q} = \{Q_1, Q_2, \dots, Q_m\}$  is the segment partition of  $L$ , then for all  $1 \leq i \leq m$  we have  $|Q_i| = c(g_i)$ . A feasible routing structure is then obtained by routing all sensing data from the sensors in  $Q_i$  to gateway  $g_i$  for all  $i \in \{1, \dots, m\}$ . For each such  $i$ , a possible tree consists of the edges in the segment  $Q_i$  together with an edge joining an arbitrary sensor in  $Q_i$  with  $g_i$ . In the following, we propose algorithms that improve over this strategy by choosing a favorable perfect matching between the segments in  $\mathcal{Q}$  and the gateways in  $W$ .

### 5.2 CMF with Uniform Capacities

For instances with uniform capacity, let  $c \in \mathbb{N}$  be the capacity of every gateway. Algorithm 1 first checks whether the input is feasible, i.e., whether  $|V| \neq m \cdot c$ , if not, reject

infeasible inputs immediately. For feasible inputs the algorithm computes an MST  $T$  of the complete graph induced by the set of sensors  $V$ . A TSP tour  $L$  based on  $T$  is then found such that the cost  $c(L)$  of  $L$  is no more than  $2c(T)$ . A partition  $\mathcal{Q}$  of the tour  $L$  into segments is computed such that all segments contain exactly  $c$  sensors. The algorithm then constructs a weighted, complete bipartite graph  $G_{\mathcal{Q}} = (W, \mathcal{Q}, E', w)$ , where  $W$  is the set of gateways and  $\mathcal{Q} = \{Q_1, Q_2, \dots, Q_m\}$  is the segment partition of  $L$ . The weight associated with the edge  $w(g_i, g_j)$  is  $d(Q_i, g_j) = \min\{d(v, g_j) \mid v \in Q_i\}$ , which is the minimum distance from  $g_j$  to a sensor in  $Q_i$ . The algorithm finally computes a minimum weight maximum matching  $M_{\mathcal{Q}}$  of  $G_{\mathcal{Q}}$ , i.e., a matching that minimizes the weighted sum of all matched edges. The solution delivered by the algorithm is the forest consisting of trees  $T_1, \dots, T_m$ , where  $T_j$  is an MST on the set  $Q_i \cup \{g_j\}$ , and gateway  $g_j$  is matched to  $Q_i$  in  $M_{\mathcal{Q}}$ . Details are given by Algorithm 1.

#### Algorithm 1. CMF with uniform capacities

**Input:**  $G = (V \cup W, E)$  and  $c \in \mathbb{N}$ : A finite graph  $G$  in the Euclidean plane consisting of a set of sensor nodes  $V$  and a set of gateways  $W = \{g_1, \dots, g_m\}$ , and a capacity  $c \in \mathbb{N}$ .

**Output:** A forest of routing trees on  $V \cup W$  for which every component contains exactly one gateway and  $c$  sensor nodes.

- 1: **if**  $|V| \neq \sum_{i=1}^m c(g_i)$  **then** declare the input as infeasible and **exit**;
- 2: Compute an MST  $T$  of the graph induced by the nodes in  $V$ ;
- 3: Compute a TSP tour  $L$  by performing a preorder tree walk in  $T$ ;
- 4: Partition  $L$  into segments  $Q_1, \dots, Q_m$  each containing  $c$  nodes;
- 5: Construct the weighted, complete bipartite graph  $G_{\mathcal{Q}} = (W, \mathcal{Q}, E', w)$  with  $\mathcal{Q} = \{Q_1, \dots, Q_m\}$ , where for  $g_i \in V$  and  $Q_j \in \mathcal{Q}$  the weight of the edge  $(Q_i, g_j)$  is  $d(Q_i, g_j) = \min\{d(v, g_j) \mid v \in Q_i\}$ ;
- 6: Compute a minimum weight perfect matching  $M_{\mathcal{Q}}$  of  $G_{\mathcal{Q}}$ ;
- 7: **return** the forest  $\mathcal{F}$  consisting of trees  $T_1, \dots, T_m$ , where tree  $T_j$  is an MST of the complete graph induced by the nodes in the set  $Q_i \cup \{g_j\}$  and  $g_j$  is the gateway matched to  $Q_i$  in  $M_{\mathcal{Q}}$ ,  $1 \leq i, j \leq m$ .

If  $m$  is the number of gateway nodes and  $n$  is the number of chosen sensors, then the running time of Algorithm 1 is  $O(m^3 + mn + n \log n)$ , since the graph  $G_{\mathcal{Q}}$  can be computed in  $O(mn)$ , the perfect matching  $M_{\mathcal{Q}}$  in  $G_{\mathcal{Q}}$  can be computed in  $O(m^3)$ , and all other tasks can be performed in time  $O(n \log n)$ .

We now investigate the quality of the solution delivered by the algorithm. To analyze the costs of solutions delivered by Algorithm 1 on instances for which all gateway nodes have identical capacities, we will need the following lemma.

**Lemma 1.** *Let  $\mathcal{Q} = \{Q_1, Q_2, \dots, Q_m\}$  be a partition of the sensors in  $V$  such that  $Q_i$  contains exactly  $c(g_i)$  sensors. Let  $\mathcal{P} = \{P_1, P_2, \dots, P_m\}$  be a node partition in an optimal solution of CMF such that  $P_i$  is the set of sensors assigned to*

$g_i$ . If for all  $i \in \{1, \dots, m\}$  the set  $P_i \cap Q_i$  is nonempty, then  $\sum_{i=1}^m d(Q_i, g_i) \leq OPT$ .

**Proof.** For all  $i \in \{1, \dots, m\}$ , let  $q_i \in P_i \cap Q_i$ . Since in the optimal solution each gateway  $g_i$  is connected to  $q_i$  and the cost of this connection is at least  $d(q_i, g_i)$ , we have  $OPT \geq \sum_{i=1}^m d(q_i, g_i) \geq \sum_{i=1}^m d(Q_i, g_i)$ .  $\square$

**Theorem 2.** For input instances of the CMF problem that satisfy the MST cost assumption and for which all gateways have the same capacity, Algorithm 1 computes a solution of cost at most  $3OPT$ , where  $OPT$  is the cost of the optimal solution.

**Proof.** We first show that there is a perfect matching  $M$  in the bipartite graph  $G_Q$  that has a cost of at most  $OPT$ . Since  $M_Q$  is a maximum matching of minimum weight this implies  $c(M_Q) \leq c(M) \leq OPT$ .

Let  $\{P_1, \dots, P_m\}$  be the node partition of  $V$  corresponding to an optimal solution, such that  $P_i$  is the set of sensor nodes assigned to gateway  $g_i$ . By Lemma 1, it suffices to construct a perfect matching  $M$  such that for each  $i$  there is a node  $p_i \in Q_{\pi(i)} \cap P_i$ , where  $\pi(i) = j$  if  $M$  matches  $g_i$  to  $Q_j$ . To this end, we construct a bipartite multigraph  $G' = (W, \mathcal{Q}, E')$ , where  $W$  is the set of gateway nodes and  $\mathcal{Q}$  is the set of segments. For every  $p \in P_i$  that is contained in  $Q_j$  there is an edge in  $E'$  between a gateway node  $g_i \in W$  and a segment  $Q_j \in \mathcal{Q}$ . Note that the degree of each gateway node  $g_i$  in  $G'$  is exactly  $c$  since  $P_i$  has size  $c$ . Similarly, the degree of each segment  $Q_j$  is  $c$ , since  $Q_j$  has size  $c$ . We use a consequence of the marriage theorem which says that every regular bipartite graph has a perfect matching. Thus, there is a perfect matching  $M$  in  $G'$ . Let  $\pi(i) = j$  if  $M$  matches gateway  $g_i$  with segment  $Q_j$ . The matching  $M$  is also a matching in  $G(W, \mathcal{Q}, E', w)$  and by construction it has the property that for each  $i$  there is a node  $p_i \in Q_{\pi(i)} \cap P_i$ .

The forest  $\mathcal{F}$  returned by Algorithm 1 consists of MSTs  $T_j$  on the sets  $Q_i \cup \{g_j\}$  where gateway  $g_j$  is matched to  $Q_i$  in  $M_Q$ . Consider the set of edges  $E_{M_Q}$  that contains the edges in the TSP tour  $L$  and each edge  $(q_i, g_j)$  with  $q_i \in Q_i$  that satisfies  $d(q_i, g_j) = d(Q_i, g_j)$  and for which  $(Q_i, g_j)$  is in  $M_Q$ . Note that this edge set  $E_{M_Q}$  in particular contains spanning trees for all sets  $Q_i \cup \{g_j\}$  whenever gateway  $g_j$  is matched to  $Q_i$  in  $M_Q$ . The total cost of all spanning trees in the returned solution is thus bounded by the cost of  $E_{M_Q}$ , i.e.,  $\sum_{i=1}^m c(T_i) \leq c(E_{M_Q})$ . The cost of the solution of Algorithm 1 is thus no more than  $\sum_{i=1}^m c(T_i) \leq c(E_{M_Q}) = c(M_Q) + c(L) \leq c(M) + 2OPT \leq 3OPT$ .  $\square$

### 5.3 CMF with Arbitrary Capacities

We now modify Algorithm 1 slightly so that it is applicable to instances that do not necessarily have uniform capacities. Instead of only one capacity  $c \in \mathbb{N}$ , the new algorithm has as input a capacity function  $c : W \rightarrow \mathbb{N}$  assigning a capacity to each gateway. This modified algorithm first checks whether the input is feasible, i.e., whether  $|V| \neq \sum_{i=1}^m c(g_i)$ , where  $\{g_1, \dots, g_m\}$  is the set of gateways, rejecting infeasible inputs immediately. When breaking the TSP tour into segments, the size of each segment has to match the capacity of a gateway. More specifically, a partition  $\mathcal{Q} = \{Q_1, Q_2, \dots, Q_m\}$  of the tour  $L$  into  $m$  segments is computed

such that for all  $1 \leq i \leq m$  we have  $|Q_i| = c(g_i)$ . For  $i, j \in \{1, \dots, m\}$  there is an edge between  $Q_i \in \mathcal{Q}$  and  $g_j \in W$  if  $c(g_j) = |Q_i|$ . As before, the weight of such an edge is  $d(Q_i, g_j) = \min\{d(v, g_j) \mid v \in Q_i\}$ . Details of Algorithm 2 are as follows:

**Algorithm 2.** CMF with an arbitrary capacities

**Input:**  $G = (V \cup W, E)$  and  $c : W \rightarrow \mathbb{N}$ : A finite graph  $G$  in the Euclidean plane consisting of a set of sensor nodes  $V$  and a set of gateway nodes  $W = \{g_1, \dots, g_m\}$ , and a capacity function  $c$  assigning to every gateway a natural number.

**Output:** A forest of routing trees on the set  $V \cup W$  for which every component contains exactly one gateway  $g_i$  and  $c(g_i)$  sensor nodes.

- 1: if  $|V| \neq \sum_{i=1}^m c(g_i)$  then declare the input as infeasible and **exit**;
- 2: Compute an MST  $T$  of the graph induced by the nodes in  $V$ ;
- 3: Compute a TSP tour  $L$  by performing a preorder tree walk in  $T$ ;
- 4: Partition  $L$  into segments  $Q_1, \dots, Q_m$  by breaking  $L$  into parts such that  $|Q_i| = c(g_i)$ ;
- 5: Construct the weighted bipartite graph  $G_Q = (W, \mathcal{Q}, E', w)$  with  $\mathcal{Q} = \{Q_1, \dots, Q_m\}$ , that has an edge of weight  $d(Q_i, g_j) = \min\{d(v, g_j) \mid v \in Q_i\}$  between  $g_i \in W$  and  $Q_j \in \mathcal{Q}$  if  $c(g_i) = |Q_j|$ ;
- 6: Compute a minimum weight perfect matching  $M_Q$  in  $G_Q$ ;
- 7: **return** the forest  $\mathcal{F}$  consisting of trees  $T_1, \dots, T_m$ , where for all  $j$  with  $1 \leq j \leq m$ , tree  $T_j$  is an MST of the complete graph that is induced by the nodes in the set  $Q_i \cup \{g_j\}$ , where  $g_j$  is the gateway that is matched to  $Q_i$  in  $M_Q$ .

By similar arguments as for Algorithm 1, it can be shown that the running time of Algorithm 2 is  $O(m^3 + mn + n \log n)$ . We now analyze its approximation ratio, which depends on the number of gateways  $m$ .

**Theorem 3.** For instances of the CMF problem with arbitrary capacities that satisfy the MST cost assumption, Algorithm 2 computes a solution of cost at most  $(m+2)OPT$ , where  $m$  is the number of gateways and  $OPT$  is the cost of the optimal solution.

**Proof.** As in the proof of Theorem 2, the cost of the forest  $\mathcal{F}$  returned by Algorithm 2 can be bounded by the cost of the set of edges  $E_{M_Q}$  that contains the edges in the TSP tour  $L$  and an edge of the form  $(q_i, g_j)$  with  $q_i \in Q_i$  that satisfies  $d(q_i, g_j) = d(Q_i, g_j)$  for every edge  $(Q_i, g_j)$  in  $M_Q$ . Again  $c(E_{M_Q}) \leq c(M_Q) + c(L)$ , and since  $c(L) \leq 2OPT$ , it thus suffices to bound the cost of  $c(M_Q)$ .

Let  $\mathcal{P} = \{P_1, P_2, \dots, P_m\}$  be a node partition in an optimal solution of CMF such that  $P_j$  is the set of sensors assigned to  $g_j$ . For every  $j \in \{1, \dots, m\}$  let  $v_j$  be some sensor in  $P_j$ . This implies  $\sum_{j=1}^m d(v_j, g_j) \leq OPT$ . Also for every  $j \in \{1, \dots, m\}$  let  $w_j$  be a sensor in  $Q_i$ , where  $Q_i$  and  $g_j$  are matched by  $M_Q$ , such that  $d(w_j, g_j) = d(Q_i, g_j)$ . By the triangle inequality  $w(Q_i, g_j) = d(w_j, g_j) \leq d(w_j, v_j) + d(v_j, g_j)$ . Note that  $d(w_j, v_j) \leq OPT$  by the MST cost

assumption. We can thus bound the cost of  $M_Q$  as follows:

$$\begin{aligned} c(M_Q) &= \sum_{(Q_i, g_j) \in M_Q} w(Q_i, g_j) = \sum_{j=1}^m d(w_j, g_j) \\ &\leq \sum_{j=1}^m d(w_j, v_j) + \sum_{j=1}^m d(v_j, g_j) \\ &\leq m \cdot OPT + OPT = (m+1) \cdot OPT. \end{aligned}$$

Thus, in total, the cost of the approximate solution computed by Algorithm 2 is at most  $c(M_Q) + c(L) = (m+1)OPT + 2OPT = (m+3) \cdot OPT$ .  $\square$

## 6 APPROXIMATION ALGORITHMS FOR CMF WITHOUT THE MST COST ASSUMPTION

To obtain the results of the previous section, we assumed that the cost  $c(T)$  of an MST  $T$  of the graph induced by the chosen sensors satisfies  $c(T) \leq OPT$ . In this section, we show how to extend our results to instances that do not necessarily fulfill the MST cost assumption. We make use of Goemans and Williamson's [9] approximation technique for constrained forest problems, and present an algorithm that combines this technique with the techniques from the previous section. We then analyze its approximation ratios on CMF instances with uniform and arbitrary capacities.

### 6.1 Goemans and Williamson's Approximation Technique

We here recall the approximation technique of Goemans and Williamson [9] which is used to solve the following type of integer program (IP).

Given an edge-weighted graph  $G = (V, E)$ , a function  $f: 2^V \rightarrow \{0, 1\}$ , and a nonnegative cost function  $c: E \rightarrow \mathbb{R}^+$ , consider the following IP:

$$\begin{aligned} &\text{Minimize} && \sum_{e \in E} c(e)x_e \\ &\text{subject to:} && \\ \text{(IP)} &&& x(\delta(S)) \geq f(S), \quad \emptyset \neq S \subseteq V \\ &&& x_e \in \{0, 1\}, \quad e \in E, \end{aligned} \quad (1)$$

where for any set  $S \subseteq V$  the expression  $\delta(S)$  denotes the set of edges with exactly one endpoint in  $S$  and  $x(\mathcal{F}) = \sum_{e \in \mathcal{F}} x_e$  for every set of edges  $\mathcal{F}$ . The IP can be interpreted as a special type of covering problem, in which the task is to find a minimum-cost set of edges that intersects all cut sets  $\delta(S)$  of sets  $S$  with  $f(S) = 1$ . Every minimal solution to such an IP corresponds to a forest, therefore the type of problem is called a *constrained forest problem* [9].

The technique of Goemans and Williamson provides an approximation algorithm to any constrained forest problem. Their algorithm takes as input an undirected graph  $G = (V, E)$ , edge costs  $c(e) > 0$  for all  $e \in E$ , and a *proper function*  $f$ , where a function  $f$  is a proper if it is symmetric, meets the disjointness property, and satisfies  $f(V) = 0$ :

- A function  $f: 2^V \rightarrow \{0, 1\}$  is *symmetric* if  $f(S) = f(V \setminus S)$  for all  $S \subseteq V$ .

- A function  $f: 2^V \rightarrow \{0, 1\}$  has the *disjointness property* if  $f(S_1) = f(S_2) = 0$  implies  $f(S_1 \cup S_2) = 0$  for all disjoint subsets  $S_1, S_2 \subseteq V$ .

The algorithm outputs a forest, which is a feasible solution, and whose cost approximates the optimal cost within a ratio of at most  $2 - 2/k$ , where  $k = |\{v \in V \mid f(\{v\}) = 1\}|$ .

On a high level, the algorithm maintains an initially empty forest  $\mathcal{F}$  and repeatedly adds an edge to the forest merging two components (trees). During this process the edge to be added is chosen as the one that minimizes a certain continuously updated measure. The running time of the algorithm is  $O(\min\{|V|^2 \log |V|, |V||E|\alpha(|E|, |V|)\})$ , where  $\alpha$  is the inverse Ackermann function (see [9]).

### 6.2 An Approximation Algorithm for CMF

We now develop an algorithm for general CMF instances. The general strategy for solving this problem is essentially to use the technique by Goemans and Williamson to decompose the problem into subproblems, such that we have a bound on the MST of each subproblem, and then apply Algorithm 2 described in the previous section. However, instead of computing a perfect matching  $M_Q$  in the call of Algorithm 2 for each component, we compute one perfect matching for the entire graph. This allows us to bound all of the edges in the matching by the global optimum.

Let a graph  $G = (V \cup W, E)$  together with a capacity function  $c: W \rightarrow \mathbb{N}$  be an input instance of the CMF problem. To apply Goemans and Williamson's algorithm, we define  $f: 2^{V \cup W} \rightarrow \{0, 1\}$  as follows: For each  $S \subseteq V \cup W$ , we set  $f(S) = 1$  if  $|S| \neq \sum_{g \in S \cap W} (1 + c(g))$  and  $f(S) = 0$  otherwise. Note that  $f(\{g\}) = 1$  for every  $g \in W$  and  $f(\{q\}) = 0$  for every  $q \in V$ . Thus,  $m = |W| = |\{v \in V \cup W \mid f(\{v\}) = 1\}|$ .

**Lemma 2.** For a feasible input (i.e., an input  $G = (V \cup W, E)$  with  $|V| = \sum_{i=1}^m c(g_i)$ ), the function  $f$  is proper.

**Proof.** Since the input is feasible, we have  $|V \cup W| = \sum_{i=1}^m (1 + c(g_i))$  and thus  $f(V \cup W) = 0$ .

**Symmetry:** It suffices to show that  $f(S) = 0$  implies  $f((V \cup W) \setminus S) = 0$ . Suppose  $S \subseteq V \cup W$  is a set with  $f(S) = 0$ . Then,

$$\begin{aligned} |(V \cup W) \setminus S| &= |V \cup W| - |S| \\ &= \sum_{g \in W} (1 + c(g)) - \sum_{g \in S \cap W} (1 + c(g)) \\ &= \sum_{g \in ((V \cup W) \setminus S) \cap W} (1 + c(g)). \end{aligned}$$

Thus,  $f((V \cup W) \setminus S) = 0$ .

**Disjointness:** Suppose  $S_1, S_2 \subseteq V \cup W$  are disjoint and  $f(S_1) = f(S_2) = 0$ . This implies

$$\begin{aligned} |S_1 \cup S_2| &= |S_1| + |S_2| = \sum_{g \in S_1 \cap W} (1 + c(g)) \\ &\quad + \sum_{g \in S_2 \cap W} (1 + c(g)) \\ &= \sum_{g \in (S_1 \cup S_2) \cap W} (1 + c(g)). \end{aligned}$$

Thus  $f(S_1 \cup S_2) = 0$ .  $\square$

Details of the approximation algorithm are given by Algorithm 3.

**Algorithm 3.** *CMF in general networks*

**Input:**  $G = (V \cup W, E)$  and  $c : W \rightarrow \mathbb{N}$ : A finite graph  $G$  in the Euclidean plane consisting of a set of sensor nodes  $V$  and a set of gateway nodes  $W = \{g_1, \dots, g_m\}$ , and a capacity function  $c$  assigning to every gateway a natural number.

**Output:** A forest consisting of routing trees on the set  $V \cup W$  for which every component contains exactly one gateway  $g_i$  and  $c(g_i)$  sensor nodes.

- 1: **if**  $|V| \neq \sum_{i=1}^m c(g_i)$  **then** declare the input as infeasible and **exit**;
- 2: Let  $\mathcal{F}$  be the forest obtained by applying Goemans and Williamson's algorithm to  $G$  with function  $f$ ;
- 3: For each component (each tree)  $T$  of  $\mathcal{F}$  compute a tour  $L_T$  as preorder tree walk of  $T$ ;
- 4: For each  $T$  set  $L'_T$  to be the tour obtained from  $L_T$  by omitting all gateways;
- 5: Partition  $V$  into segments  $Q_1, \dots, Q_m$  by breaking all tours  $L'_T$  into parts such that the sizes of the parts of  $T$  correspond to the capacities of the gateways in  $T$ ;
- 6: Form the weighted bipartite graph  $G_Q = (W, \mathcal{Q}, E', w)$  with  $\mathcal{Q} = \{Q_1, \dots, Q_m\}$ , that has an edge of weight  $d(g, Q_i)$  between  $g_i \in W$  and  $Q_j \in \mathcal{Q}$  if  $c(g_i) = |Q_j|$ ;
- 7: Compute a minimum weight perfect matching  $M_Q$  of  $G_Q$ ;
- 8: **return** the forest consisting of trees  $T_1, \dots, T_m$ , where, for all  $j$  with  $1 \leq j \leq m$ , the tree  $T_j$  is an MST of the complete graph that is induced by the set  $Q_i \cup \{g_j\}$ , and  $g_j$  is the gateway that is matched to  $Q_i$  by  $M_Q$ .

The running time of Algorithm 3 differs from the running time of Algorithm 2 only by the time required for a call to the Goemans and Williamson algorithm and is thus  $O(n^2 \log n + mn + m^3)$ . In our analysis of the approximation ratio of Algorithm 3, we first deal with CMF with uniform capacities first. We then consider CMF with arbitrary capacities.

### 6.3 CMF with Uniform Capacities

**Theorem 4.** *On CMF instances with  $m$  gateway nodes of equal capacity, Algorithm 3 has an approximation ratio of  $(5 - 4/m)$ .*

**Proof.** Let a graph  $G = (V \cup W, E)$  together with a capacity function  $c$  be an input instance of the CMF problem. We first argue that any optimal solution to the CMF problem is a solution to the **IP** using function  $f$  as defined earlier. For this it suffices to observe that every tree  $T'$  whose vertex set  $V(T')$  consists of a gateway  $g$  and  $c(g)$  sensor nodes fulfills  $f(V(T')) = 0$ . This implies that in particular every tree in a solution to the CMF problem has this property. Thus, on instances that are not immediately declared as infeasible, Algorithm 3 first computes a forest  $\mathcal{F}$  with cost at most  $2 - 2/m$  times the cost of an optimal solution to the CMF instance, where  $m = |\{g \mid f(\{g\}) = 1, g \in V \cup W\}| = |W|$  is the number of gateways. The algorithm then computes a perfect matching  $M_Q$  and TSP tours  $L'_T$  obtained from the TSP

tours  $L_T$  on the components (trees) of  $\mathcal{F}$ . Analogous to the proofs of theorems 2 and 3, the cost of the output of Algorithm 3 is bounded by  $c(M_Q) + \sum_{T \in \mathcal{F}} c(L'_T) \leq c(M_Q) + \sum_{T \in \mathcal{F}} c(L_T)$ . Furthermore, with the same arguments as used in the proof of Theorem 2, we can conclude that there is a perfect matching  $M$  of cost at most  $OPT$  in the bipartite graph  $M_Q$  implying  $c(M_Q) \leq c(M) \leq OPT$ . Thus, the cost of the approximate solution is at most

$$\begin{aligned} c(M_Q) + \sum_{T \in \mathcal{F}} c(L(T)) &\leq OPT + \sum_{T \in \mathcal{F}} (2 \cdot c(T)) \\ &= OPT + 2 \cdot \sum_{T \in \mathcal{F}} c(T) \leq OPT + 2 \cdot (2 - 2/m) \cdot OPT \\ &= (5 - 4/m) \cdot OPT. \end{aligned}$$

□

### 6.4 CMF with Arbitrary Capacities

Analogously to our analysis of Algorithm 2, we now analyze the approximation ratio of Algorithm 3 on CMF instances in general. The strategy for this case is similar to the one for the uniform capacity case and we have the following theorem.

**Theorem 5.** *On CMF instances with  $m$  gateway nodes of arbitrary capacity, Algorithm 3 has an approximation ratio of  $(m + 2)(2 - 2/m)$ .*

**Proof.** By the same arguments as used in the previous proof, the forest  $\mathcal{F}$  has cost at most  $(2 - 2/m) \cdot OPT$  and the cost of the solution delivered by Algorithm 3 is bounded by  $c(M_Q) + \sum_{T \in \mathcal{F}} c(L'_T) \leq c(M_Q) + \sum_{T \in \mathcal{F}} c(L_T)$ . The cost of  $c(M_Q)$  is bounded as follows: By the construction of  $G_Q$ , there exists a perfect matching  $M$  which, for  $j \in \{1, \dots, m\}$ , matches gateway  $g_j$  with a segment  $Q_i$  such that  $g_j$  and the nodes of  $Q_i$  are contained in the same tree  $T$  of  $\mathcal{F}$ . This implies that  $w(Q_i, g_j) \leq c(T) \leq c(\mathcal{F}) \leq (2 - 2/m)OPT$ . In total, the cost of the solution is bounded as follows:

$$\begin{aligned} c(M_Q) + \sum_{T \in \mathcal{F}} c(L(T)) &\leq \sum_{(Q_i, g_j) \in M_Q} w(Q_i, g_j) + \sum_{T \in \mathcal{F}} (2 \cdot c(T)) \\ &\leq m \cdot c(\mathcal{F}) + 2 \cdot \sum_{T \in \mathcal{F}} c(T) \\ &\leq m(2 - 2/m) + 2 \cdot (2 - 2/m) \cdot OPT \\ &= (m + 2)(2 - 2/m) \cdot OPT. \end{aligned}$$

□

## 7 CMF WITH COMPARABLE CAPACITIES

In this section, we consider a special type of CMF instance which has not necessarily uniform capacities, but has capacities that are small multiples of some integer. The motivation of this type of instance comes from the fact that in practice, one might have for example two types of gateways, where the capacity of the one type is twice as



large as the capacity of the other type. To describe our instances more formally, let  $C_{\max} = \max\{c(g_1), \dots, c(g_m)\}$  be the maximum capacity among all gateways and  $c_{\gcd} = \gcd\{c(g_1), c(g_2), \dots, c(g_m)\}$  the greatest common divisor of all capacities. We provide an algorithm with an approximation ratio that depends on the ratio  $r = C_{\max}/c_{\gcd}$ . For brevity, we say that these instances have comparable capacities. In our example above, the ratio  $r$  is 2. For convenience, we define  $l_i = c(g_i)/c_{\gcd}$  for each gateway. Note that  $1 \leq l_i \leq r$ .

To solve instances with comparable capacities, we devise a reduction to CMF with uniform capacities. To reduce the original instance to a uniform capacity instance, we replace every gateway  $g_i$  by  $l_i$  “virtual gateways” of capacity  $c_{\gcd}$  colocated at the original position of gateway  $g_i$ . We use Algorithm 3 to find a solution of the modified instance. Any solution to the uniform capacity instance can be transformed into solution to the original instance with at most the same cost as follows: First, for each gateway  $g_i$  we assign all sensor nodes that have previously been assigned to one of the virtual replacement gateways of  $g_i$ . We then compute an MST for the set of sensors assigned to  $g_i$ . Details of Algorithm 4 are as follows:

**Algorithm 4.** CMF with comparable capacities

**Input:**  $G = (V \cup W, E)$  and  $c : W \rightarrow \mathbb{N}$ : A finite graph  $G$  in the Euclidean plane consisting of a set of sensor nodes  $V$  and a set of gateway nodes  $W$ , and a capacity function  $c$  assigning to each gateway node a natural number.

**Output:** A forest consisting of routing trees on the set  $V \cup W$  for which every component contains exactly one gateway  $g_i$  and  $c(g_i)$  sensor nodes.

- 1: **if**  $|V| \neq \sum_{i=1}^m c(g_i)$  **then** declare the input as infeasible and **exit**;
- 2: Compute  $c_{\gcd} = \gcd(c(g_1), c(g_2), \dots, c(g_m))$ ;
- 3: Construct a new graph  $G_{eq} = (V \cup W', E')$ , where  $W' = \{g_i^k \mid g_i \in W, k \in \{1, \dots, l_i\}\}$ ;
- 4: Solve the CMF with equal capacity  $c_{\gcd}$  in  $G_{eq}$  with Algorithm 3. Let  $\mathcal{F}_{eq}$  be the solution, where  $\mathcal{F}_{eq} = \{T_j^k \mid T_j^k \text{ is a tree rooted at a virtual gateway node } g_j^k \in W' \text{ for all } k \in \{1, \dots, l_j\}\}$ ;
- 5: **return** the forest consisting of the trees  $T_1, \dots, T_m$ . Here, for all  $j \in \{1, \dots, m\}$ , tree  $T_j$  is an MST on  $\bigcup_{k=1}^{l_j} V(T_j^k) \cup \{g_j\}$ , where  $V(T_j^k)$  is the set of sensor nodes in  $T_j^k$ .

The running time of Algorithm 4 is  $O(n^2 \log n + rmn + (rm)^3)$  since every gateway is replaced by at most  $r$  virtual gateways before Algorithm 3 is called. Note that this is polynomial in the input, since the capacity constraints force  $rm \leq n$ .

To analyze the approximation ratio of the algorithm, we employ the concept of the Steiner ratio.

**Definition 1.** The Steiner ratio  $\rho$  is defined to be  $\rho = \inf\{|MST(V)|/|MST(U)| \mid U \subseteq V \text{ finite sets of points in the Euclidean plane}\}$ , where for a set of points in the plane  $S$ ,  $|MST(S)|$  is the cost of an MST connecting all points in  $S$ .

The Steiner ratio is known to lie in the interval  $[\rho_0, \sqrt{3}/2] \approx [0.82416, 0.86602]$  (see [4]) and has been conjectured to be  $\sqrt{3}/2$  by Gilbert and Pollak [8].

The following theorem solves the special case of the CMF with comparable capacities.

**Theorem 6.** Algorithm 4 solves the CMF problem with an approximation ratio of  $5/\rho \cdot r < 6.07 \cdot r$ , where  $\rho$  is the Steiner ratio and  $r = C_{\max}/c_{\gcd}$  is the ratio of the largest capacity to the greatest common divisor of all capacities.

**Proof.** Given an instance  $I$  of the CMF problem, denote by  $c_{\gcd} = \gcd\{c(g_1), c(g_2), \dots, c(g_m)\}$  the greatest common divisor of all capacities. Algorithm 4 first transforms the instance  $I$  into an instance  $I'$  of CMF with uniform capacities and virtual gateways. This instance is solved by Algorithm 3. Using the obtained solution of instance  $I'$ , a solution for  $I$  is computed that has a cost which is no larger than the solution to instance  $I'$ . Algorithm 3 is a 5-approximation on instances with uniform capacity, by Theorem 4. To finish the proof, it therefore suffices to show that the cost  $OPT'$  of an optimal solution of the modified instance  $I'$  is at most by a factor of  $r/\rho$  larger than the cost  $OPT$  of the optimal solution to the original instance  $I$ .

Consider an optimal solution of the original instance and suppose for all  $i \in \{1, \dots, m\}$ , set  $V_i$  is the set of sensors assigned to a gateway  $g_i$  via a routing tree  $T_i$ . By the definition of the reduction, each gateway  $g_i$  is replaced by at most  $r$  gateways  $g_i^1, \dots, g_i^{l_i}$ . Consider a solution to the modified instance obtained by arbitrarily partitioning the nodes in set  $V_i$  into  $l_i$  subsets  $V_i^1, \dots, V_i^{l_i}$  of equal size and assigning each node in  $V_i^k$  to the virtual gateway  $g_i^k$ . Let  $T_i^k$  be an MST of the complete graph induced by the node set  $V_i^k \cup \{g_i^k\}$ . By the definition of the Steiner ratio,  $c(T_i^k) \leq (1/\rho) \cdot c(T_i)$ . Thus,

$$\begin{aligned} OPT' &\leq \sum_{i=1}^m \sum_{k=1}^{l_i} c(T_i^k) \leq \sum_{i=1}^m \sum_{k=1}^{l_i} \frac{1}{\rho} \cdot c(T_i) \\ &= \frac{1}{\rho} \sum_{i=1}^m l_i \cdot c(T_i) \leq \frac{1}{\rho} \sum_{i=1}^m r \cdot c(T_i) \quad \text{since } l_i \leq r, \\ &\leq \frac{1}{\rho} \cdot r \sum_{i=1}^m c(T_i) = \frac{1}{\rho} \cdot r \cdot OPT. \end{aligned}$$

Thus, the solution provided by Algorithm 4 is by at most a factor of  $(1/\rho) \cdot r \cdot (5 - 4/m') < (5/\rho) \cdot r$  larger than the optimum, where  $m'$  is the number of virtual gateways in instance  $I'$ .  $\square$

## 8 PERFORMANCE EVALUATION

In this section, we evaluate the performance of the proposed approximation algorithms for the CMF problem through experimental simulations. We in particular investigate the impact of parameters: the number of chosen sensors  $|V|$  and gateways  $m$  on the performance of the algorithms.

We consider a wireless sensor network consisting of 600 to 1,000 sensors randomly deployed in a  $200 \times 200$  square meters region. The transmission range of each sensor is 15 meters. The  $m$  gateway nodes are also randomly deployed in the monitoring region. The value in each figure is the mean of the results delivered by each mentioned algorithm applied to 20 different topologies. To improve

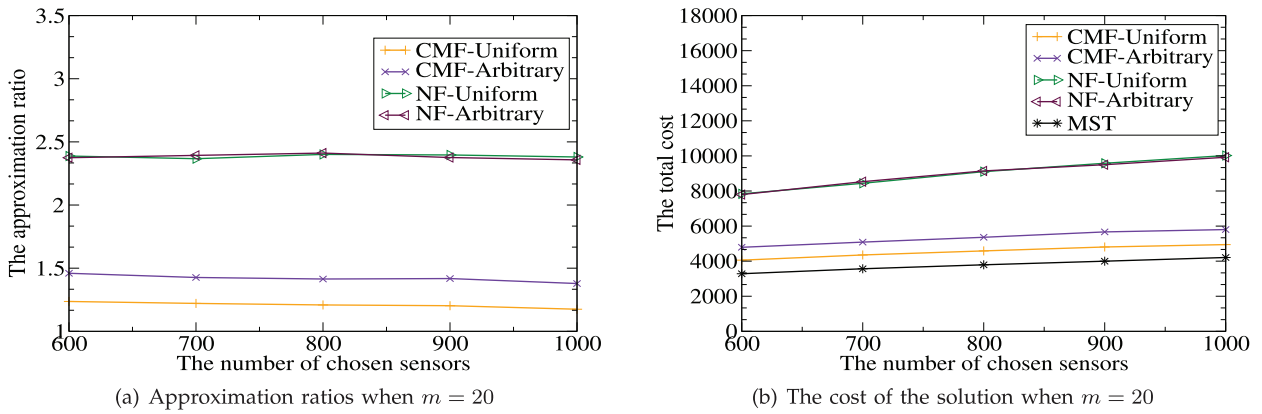


Fig. 1. The performance of Algorithm 1 CMF-Uniform, Algorithm 2 CMF-Arbitrary, Algorithm NF-Uniform, Algorithm NF-Arbitrary, and Algorithm MST, by varying the number of chosen sensors while the number of gateways is fixed at 20.

readability, in the figures, we refer to Algorithms 1 and 2 as CMF-Uniform and CMF-Arbitrary, respectively. We refer to Algorithm 3 as GW-Uniform and GW-Arbitrary, depending whether the algorithm is applied to instances in which all gateways have uniform capacities. We refer to Algorithm 4 as GW-Comparable. To evaluate the proposed algorithms, we use the following greedy heuristic for the CMF problem as the benchmark. We refer to this heuristic as Nearest-First.

Algorithm Nearest-First iteratively constructs the forest of routing trees. Initially, the forest consists of  $m$  trees with each containing only a gateway node as the root. In each iteration, the algorithm adds a sensor node  $v$  to a tree  $T$  in which the number of sensor nodes is less than the capacity of its gateway. An edge  $e$  between the new sensor node and a closet node in  $T$  is also added to  $T$ . The sensor node  $v$  that is added is chosen so that the length of  $e$  is minimal. This procedure is repeated until the capacity constraints of all gateways in the forest are met. We refer to Algorithm Nearest-First as NF-Uniform and NF-Arbitrary when it is applied to instances with uniform and arbitrary capacity constraints, respectively.

In accordance with the NP-completeness of the CMF problem, it is unlikely that there is an efficient exact algorithm to find an optimal solution to the problem. To analyze the approximation ratio of the solution delivered by Algorithm 1 or 2 to the optimal cost  $OPT$ , following our assumption that the cost of an MST of the graph induced by all chosen sensors is no greater than the  $OPT$  of the CMF problem, we use the cost of the MST as an estimation of  $OPT$  for Algorithms 1 and 2. To calculate the approximation ratio of Algorithms 3 or 4 to the optimal cost  $OPT$  without the MST cost assumption, we use a lower bound on  $OPT$ —the cost of a minimum spanning forest  $c(MSF)$  of graph  $G(V \cup W, E)$  as an estimation of  $OPT$ . Within the minimum spanning forest, each tree is a tree rooted at a gateway node, and there are not any capacity constraints on gateway nodes. It must be emphasized that this estimate is very conservative, and the optimal cost  $OPT$  can be much higher than this estimated cost  $c(MSF)$ . Consequently, the approximation ratios of our algorithms are lower than the ratios we reported, which are calculated based on the lower bound of  $OPT$ . We refer to algorithms

for finding MSTs and minimum spanning forests as Algorithm MST and Algorithm MSF, respectively.

### 8.1 Impact of the Number of Chosen Sensors on the Performance of Proposed Approximation Algorithms

We first investigate the performance of approximation Algorithms 1, 2, and 3 with uniform and arbitrary capacity constraints, and compare them with the performance of Algorithm NF-Uniform and Algorithm NF-Arbitrary. For this purpose, we vary the number of chosen sensors from 600 to 1,000 with an increment of 100 while the number of gateways  $m$  is fixed at 20. More specifically, we evaluate the performance of Algorithm 1 CMF-Uniform, Algorithm 2 CMF-Arbitrary, Algorithm 3 GW-Uniform with uniform gateway capacities, Algorithm 3 GW-Arbitrary with arbitrary gateway capacities, Algorithm NF-Uniform with uniform gateway capacities, and Algorithm NF-Arbitrary with arbitrary gateway capacities.

Fig. 1a depicts the approximation ratios of Algorithm 1 CMF-Uniform and Algorithm 2 CMF-Arbitrary and compares them with the approximation ratios of algorithms NF-Uniform and NF-Arbitrary. The figure shows that both Algorithms 1 and 2 outperform algorithms NF-Uniform and NF-Arbitrary significantly. The approximation ratios obtained by our approximation algorithms are no more than 1.5. Fig. 1b illustrates the actual cost of solutions delivered by Algorithms 1, 2, NF-Uniform, NF-Arbitrary, and MST. We conclude that the cost of the heuristic solution is significantly higher than the cost of the solutions provided by our approximation algorithms.

Fig. 2a depicts the approximation ratios of Algorithm 3 GW-Uniform and Algorithm 3 GW-Arbitrary and compares them with those of Algorithm NF-Uniform and Algorithm NF-Arbitrary. It can be seen from Fig. 2a that Algorithm 3 outperforms Algorithm Nearest-First significantly. The figure also shows that, in our experiments, the approximation ratio of the proposed approximation algorithm is no more than 1.65 for all different network sizes and network topologies. This is consistent with our theoretical analysis that the approximation ratio is independent of the number of chosen sensors. We emphasize again, that the approximation ratio obtained are only an upper bound on the actual approximation ratio, since we do

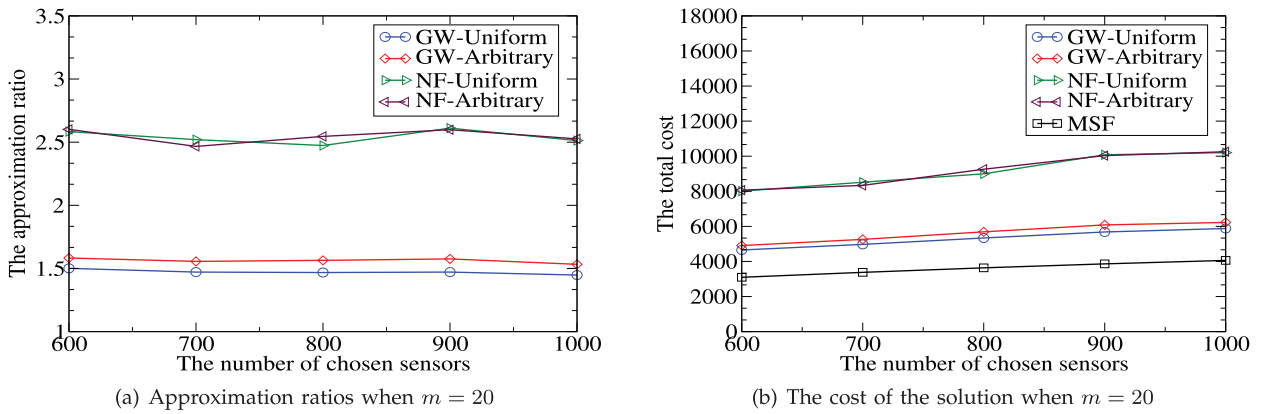


Fig. 2. The performance of Algorithm 3 and Algorithm  $MSF$  against Algorithm  $Nearest-First$  with uniform and arbitrary capacity constraints, by varying the number of chosen sensors from 600 to 1,000 while the number of gateways is fixed at 20.

not know  $OPT$  and can only compare the solutions with the lower bound  $c(MSF)$ . Fig. 2b depicts the cost of the solutions delivered by the mentioned algorithms.

The approximation ratios obtained by the heuristic algorithm are no smaller than 2.45 and 2.5 in Figs. 1a and 2a, respectively. We, thus, conclude that the multiplicative gap between the approximation ratios of our approximation algorithms and the heuristic is significant. Note that the multiplicative gap will increase if a larger lower bound for  $OPT$  is used, in particular, if the real value of  $OPT$  itself is used.

## 8.2 Impact of the Number of Gateways on the Performance of the Proposed Approximation Algorithms

We then study the impact of the number of gateways on the performance of the proposed algorithms with uniform and arbitrary capacity constraints. We vary the number of gateways from 20 to 40 with an increment of 5 while the number of chosen sensors is fixed at 1,000.

Fig. 3a depicts the approximation ratios of Algorithm 1  $CMF-Uniform$ , Algorithm 2  $CMF-Arbitrary$ , Algorithm  $NF-Uniform$ , and Algorithm  $NF-Arbitrary$ . It can be seen that the approximation ratio of Algorithm 1  $CMF-Uniform$  is independent of the number of gateways. This is consistent with our analytical result. For the arbitrary capacity case, it can be seen from Fig. 3a that the larger

the number of gateways, the higher the approximation ratio of Algorithm 2  $CMF-Arbitrary$ . However, the approximation ratio of Algorithm 2  $CMF-Arbitrary$  is no more than 1.65, this contrasts our analytical result which says that the approximation ratio is linear in the number of gateways. As expected this indicates that on realistic inputs the proposed algorithm performs better than what our worst case analysis shows. Fig. 3a clearly demonstrates that both Algorithm 1  $CMF-Uniform$  and Algorithm 2  $CMF-Arbitrary$  outperform Algorithm  $Nearest-First$ , and the approximation ratios of the approximation algorithms are at most a fraction of the approximation ratios of the heuristic. Fig. 3b depicts the costs of the solutions of the mentioned algorithms.

Fig. 4a plots the curves of approximation ratios of Algorithm  $GW-Uniform$  and Algorithm  $GW-Arbitrary$ . The figure shows that as the number of gateways increases, the approximation ratio increases as well. However, the approximation ratio in our experiments is at most 1.83. This contrasts the conservative analytical result that the approximation ratio of Algorithm 3  $GW-Arbitrary$  is proportional to the number of gateways, which means that the proposed algorithm is efficient in practice. Fig. 4b depicts the cost of the solutions delivered by the mentioned algorithms for uniform and arbitrary capacity cases.

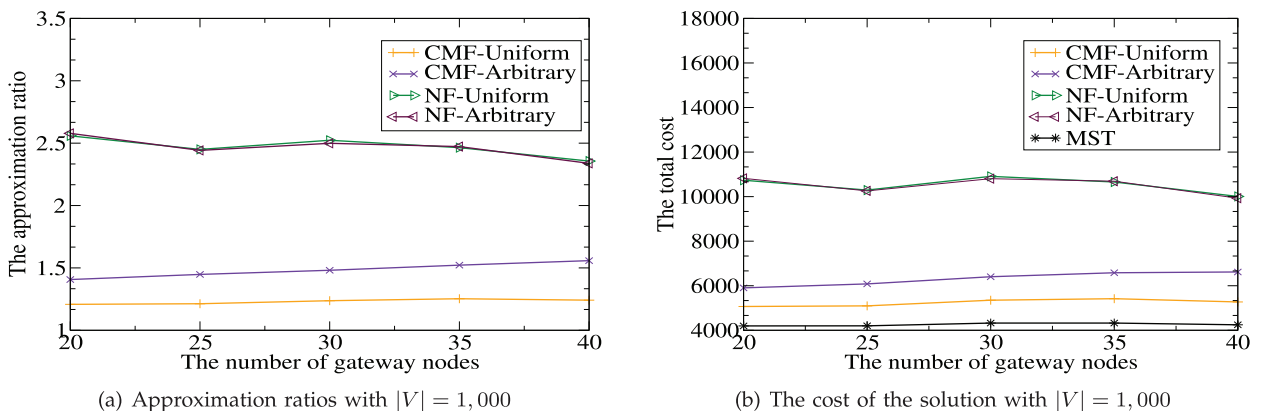


Fig. 3. The performance of Algorithm 1  $CMF-Uniform$ , Algorithm 2  $CMF-Arbitrary$ , Algorithm  $MST$ , and Algorithm  $Nearest-First$  by varying the number of gateways from 20 to 40 while the number of chosen sensors is fixed 1,000.

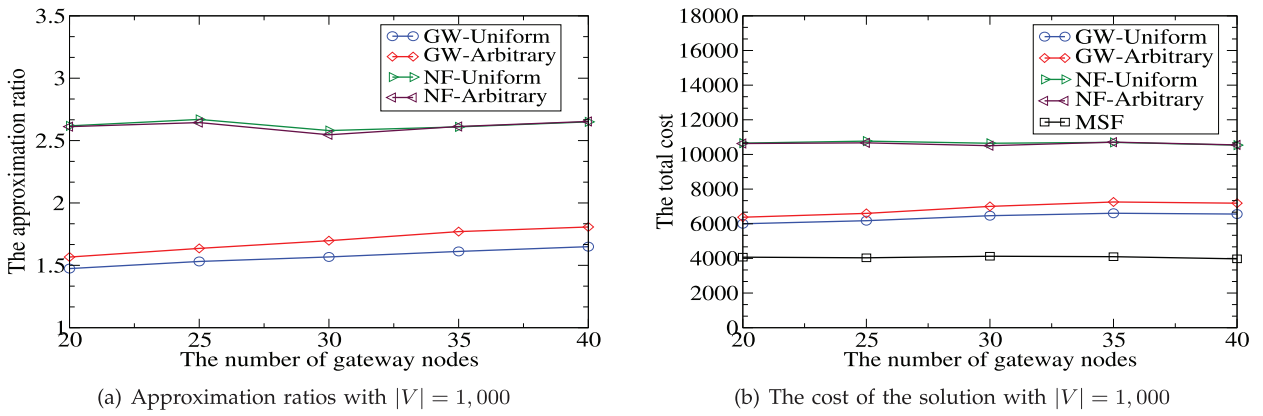


Fig. 4. The performance of Algorithm 3, Algorithm  $MSF$ , and Algorithm  $Nearest-First$  with uniform and arbitrary capacities by varying the number of gateways from 20 to 40 while the number of chosen sensors is fixed at 1,000.

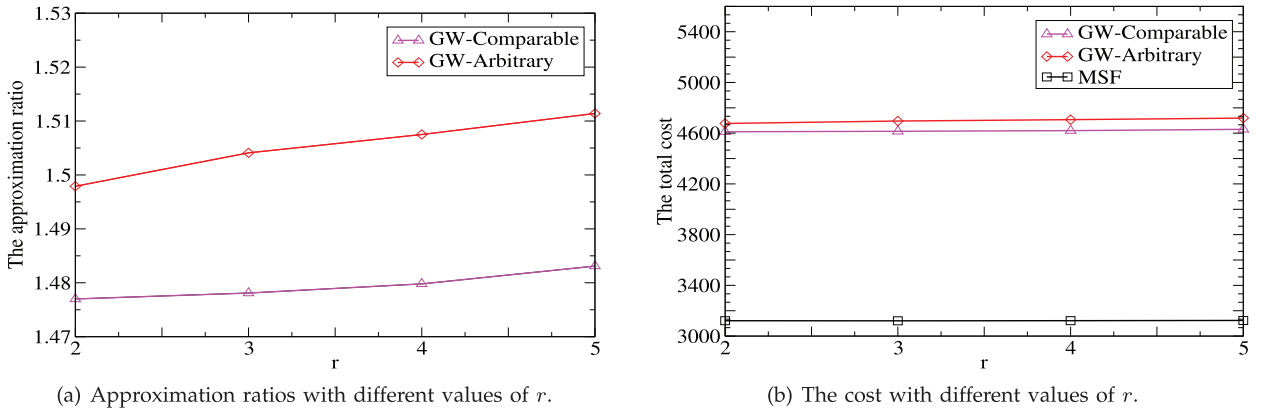


Fig. 5. The performance of Algorithm 4  $GW-Comparable$  and Algorithm 3  $GW-Arbitrary$  when  $|V| = 600$ ,  $m = 20$ , and  $c_{gcd} = 20$ .

### 8.3 Performance Evaluation of the Proposed Algorithm with Comparable Capacities

We finally evaluate the performance of Algorithm 4  $GW-Comparable$  against that of Algorithm 3  $GW-Arbitrary$  for a special case of CMF with comparable capacities. We vary the ratio  $r = C_{max}/c_{gcd}$  from 2 to 5 while fixing the number of chosen sensors to be  $|V| = 600$  and the number of gateways  $m = 20$ , assuming that  $c_{gcd} = 20$ . Fig. 5 depicts the performance curves of the two mentioned algorithms, from which it can be seen that Algorithm 4  $GW-Comparable$  outperforms Algorithm 3  $GW-Arbitrary$  in all cases. With the growth of the ratio  $r$ , the approximation ratio increases as well. This is consistent with our analytical result. However, Fig. 5a also clearly indicates that the actual approximation ratio in our experiments is between 1.475 and 1.515, which is significantly lower than the theoretical bound.

## 9 CONCLUSION

In this paper, we have studied the deployment of wireless sensor networks with mobile sinks for large-scale monitoring. We proposed a heterogeneous and hierarchical architecture that consists of sensors, gateways, and mobile sinks, where the mobile sinks travel along predetermined trajectories to collect data from the gateways, while the sensors send their sensing data to gateways for temporary storage through multiple-hop relays. Under this new paradigm of data gathering, we first formulated a novel

constrained optimization problem, namely the CMF problem and showed its NP-completeness. We then devised approximation algorithms for instances where all gateways have uniform capacities and arbitrary capacities. Their approximation ratios are bounded by a constant and by  $(m+2)(2-2/m)$  respectively, where  $m$  is the number of gateway nodes. We also showed a reduction from a special case of CMF, where the capacities are small multiples of some common number, to the case of CMF with uniform capacities, obtaining an improved approximation ratio. We finally evaluated the performance of the proposed algorithms through experimental simulation. The experimental results demonstrate that the proposed algorithms are very efficient, and on realistic inputs the actual approximation ratios are less than 2. This clearly contrasts the conservative theoretical results for the case of arbitrary capacities, which show that the approximation ratio is linear in the number of gateways.

## REFERENCES

- [1] I.F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, "Wireless Sensor Networks: A Survey," *Computer Networks*, vol. 38, no. 4, pp. 393-422, 2002.
- [2] K. Altinkemer and B. Gavish, "Heuristics with Constant Error Guarantees for the Design of Tree Networks," *Management Science*, vol. 34, no. 3, pp. 331-341, 1988.
- [3] S. Basagni, A. Carosi, E. Melachrinoudis, C. Petrioli, and Z.M. Wang, "Controlled Sink Mobility for Prolonging Wireless Sensor Networks Lifetime," *Wireless Networks*, vol. 14, no. 6, pp. 831-858, 2008.

- [4] F.R.K. Chung and R.L. Graham, "A New Bound for Euclidean Steiner Minimal Trees," *Annals of the New York Academy of Sciences*, vol. 440, no. 1, pp. 328-346, 1985.
- [5] T.H. Cormen, C.E. Leiserson, R.L. Rivest, and C. Stein, *Introduction to Algorithms*, third ed. MIT Press, 2009.
- [6] M.C. de Souza, C. Duhamel, and C.C. Ribeiro, *A GRASP Heuristic for the Capacitated Minimum Spanning Tree Problem Using a Memory-Based Local Search Strategy*, pp. 627-657, Kluwer Academic Publishers, 2004.
- [7] S. Gao, H. Zhang, and S.K. Das, "Efficient Data Collection in Wireless Sensor Networks with Path-Constrained Mobile Sinks," *IEEE Trans. Mobile Computing*, vol. 10, no. 4, pp. 592-608, Apr. 2011.
- [8] E.N. Gilbert and H.O. Pollak, "Steiner Minimal Trees," *SIAM J. Applied Math.*, vol. 16, pp. 1-29, 1968.
- [9] M.X. Goemans and D.P. Williamson, "A General Approximation Technique for Constrained Forest Problems," *SIAM J. Computing*, vol. 24, no. 2, pp. 296-317, 1995.
- [10] D. Golovin, M. Faulkner, and A. Krause, "Online Distributed Sensor Selection," *Proc. ACM/IEEE Ninth Int'l Conf. Information Processing in Sensor Networks (IPSN)*, pp. 220-231, 2010.
- [11] L. Gouveia, "A Comparison of Directed Formulations for the Capacitated Minimal Spanning Tree Problem," *Telecomm. Systems*, vol. 1, pp. 51-76, 1993.
- [12] R. Jothi and B. Raghavachari, "Approximation Algorithms for the Capacitated Minimum Spanning Tree Problem and its Variants in Network Design," *ACM Trans. Algorithms*, vol. 1, no. 2, pp. 265-282, 2005.
- [13] A. Kansal, A.A. Somasundara, D.D. Jea, M.B. Srivastava, and D. Estrin, "Intelligent Fluid Infrastructure for Embedded Networks," *Proc. Second Int'l Conf. Mobile Systems, Applications, and Services (MobiSys)*, pp. 111-124, 2004.
- [14] W. Liang, "Approximate Minimum-Energy Multicasting in Wireless Ad Hoc Networks," *IEEE Trans. Mobile Computing*, vol. 5, no. 4, pp. 377-387, Apr. 2006.
- [15] W. Liang, "Constrained Resource Optimization in Wireless Sensor Networks with Mobile Sinks," *Proc IEEE Int'l Conf. Computing, Networking and Comm.*, pp. 559-603, 2012.
- [16] W. Liang and J. Luo, "Network Lifetime Maximization in Sensor Networks with Multiple Mobile Sinks," *Local Computer Networks*, pp. 354-361, 2011.
- [17] W. Liang, J. Luo, and X. Xu, "Prolonging Network Lifetime via a Controlled Mobile Sink in Wireless Sensor Networks," *Proc. IEEE Global Telecomm. Conf. (GLOBECOM)*, pp. 1-6, 2010.
- [18] J. Luo and J.-P. Hubaux, "Joint Mobility and Routing for Lifetime Elongation in Wireless Sensor Networks," *Proc. IEEE INFOCOM*, pp. 1735-1746, 2005.
- [19] J. Luo and J.-P. Hubaux, "Joint Sink Mobility and Routing to Maximize the Lifetime of Wireless Sensor Networks: The Case of Constrained Mobility," *IEEE/ACM Trans. Networking*, vol. 18, no. 3, pp. 871-884, June 2010.
- [20] J. Luo, J. Panchard, M. Piorkowski, M. Grossglauser, and J.-P. Hubaux, "MobiRoute: Routing Towards a Mobile Sink for Improving Lifetime in Sensor Networks," *Proc. IEEE Second Int'l Conf. Distributed Computing in Sensor Systems (DCOSS)*, pp. 480-497, 2006.
- [21] M. Ma and Y. Yang, "SenCar: An Energy-Efficient Data Gathering Mechanism for Large-Scale Multihop Sensor Networks," *IEEE Trans. Parallel and Distributed Systems*, vol. 18, no. 10, pp. 1476-1488, Oct. 2007.
- [22] G.L. Nemhauser, L.A. Wolsey, and M.L. Fisher, "An Analysis of Approximations for Maximizing Submodular Set Functions-I," *Math. Programming*, vol. 14, pp. 265-294, 1978.
- [23] T. Nieberg, J. Hurink, and W. Kern, "Approximation Schemes for Wireless Networks," *ACM Trans. Algorithms*, vol. 4, no. 4, pp. 49:1-49:17, 2008.
- [24] C.H. Papadimitriou, "The Complexity of the Capacitated Tree Problem," *Networks*, vol. 8, pp. 217-230, 1978.
- [25] Y. Shi and Y.T. Hou, "Theoretical Results on Base Station Movement Problem for Sensor Network," *Proc. IEEE INFOCOM*, pp. 376-384, 2008.
- [26] A.A. Somasundara, A. Kansal, D. Jea, D. Estrin, and M.B. Srivastava, "Controllably Mobile Infrastructure for Low Energy Embedded Networks," *IEEE Trans. Mobile Computing*, vol. 5, no. 8, pp. 958-973, Aug. 2006.
- [27] R. Sugihara and R.K. Gupta, "Optimizing Energy-Latency Trade-Off in Sensor Networks with Controlled Mobility," *Proc. IEEE INFOCOM*, pp. 2566-2570, 2009.
- [28] Q. Wang, X. Wang, and X. Lin, "Mobility Increases the Connectivity of K-Hop Clustered Wireless Networks," *Proc. ACM MOBICOM*, pp. 121-132, 2009.
- [29] Z.M. Wang, S. Basagni, E. Melachrinoudis, and C. Petrioli, "Exploiting Sink Mobility for Maximizing Sensor Networks Lifetime," *Proc. 38th Ann. Hawaii Int'l Conf. System Sciences (HICSS)*, 2005.
- [30] Y. Wu, Z. Mao, S. Fahmy, and N.B. Shroff, "Constructing Maximum-Lifetime Data-Gathering Forests in Sensor Networks," *IEEE/ACM Trans. Networking*, vol. 18, no. 5, pp. 1571-1584, Oct. 2010.
- [31] G. Xing, T. Wang, W. Jia, and M. Li, "Rendezvous Design Algorithms for Wireless Sensor Networks with a Mobile Base Station," *Proc. ACM MobiHoc*, pp. 231-240, 2008.
- [32] X. Xu and W. Liang, "Data Quality Maximization in Sensor Networks with a Mobile Sink," *Proc. IEEE Int'l Conf. Distributed Computing in Sensor Systems (DCOSS)*, pp. 1-8, 2011.
- [33] X. Xu, W. Liang, and T. Wark, "Monitoring Quality Optimization in Wireless Sensor Networks with a Mobile Sink," *Proc. 14th Int'l Conf. Modeling, Analysis and Simulation of Wireless and Mobile Systems*, pp. 77-84, 2011.
- [34] X. Xu, J. Luo, and Q. Zhang, "Delay Tolerant Event Collection in Sensor Networks with Mobile Sink," *Proc. IEEE INFOCOM*, pp. 2471-2479, 2010.
- [35] Y. Yun and Y. Xia, "Maximizing the Lifetime of Wireless Sensor Networks with Mobile Sink in Delay-Tolerant Applications," *IEEE Trans. Mobile Computing*, vol. 9, no. 9, pp. 1308-1318, Sept. 2010.



**Weifa Liang** (M'99-SM'01) received the BSc degree from Wuhan University, China, in 1984, the ME degree from the University of Science and Technology of China in 1989, and the PhD degree from the Australian National University in 1998, all in computer science. He is currently an associate professor in the Research School of Computer Science at the Australian National University. His research interests include design and analysis of energy-efficient routing protocols

for wireless ad hoc and sensor networks, information processing in wireless sensor networks, routing protocol design for WDM optical networks, design and analysis of parallel and distributed algorithms, combinatorial optimization, and graph theory. He is a senior member of the IEEE.



**Pascal Schweitzer** received the diploma degree in mathematics at the Johann Wolfgang Goethe University of Frankfurt, Germany, in 2005 and the PhD degree from the Max-Planck Institute for Informatics at Saarland University in Germany in 2009. He is currently a laureate of the European Post-Doc Institute and a post-doctoral fellow at the Australian National University supported by the national research fund of Luxembourg. His main research interests

include combinatorial and algorithmic aspects of graph theory and geometric group theory.



**Zichuan Xu** received the BSc and ME degrees from Dalian University of Technology, China, both in computer science, in 2008 and 2011, respectively. He is currently working toward the PhD degree in the Research School of Computer Science at the Australian National University. His research interests include wireless sensor networks, routing protocol design for wireless networks, algorithmic game theory, and optimization problems. He is a student member of the IEEE.

► For more information on this or any other computing topic, please visit our Digital Library at [www.computer.org/publications/dlib](http://www.computer.org/publications/dlib).