A 2-Approximation Algorithm for Optimal Deployment of k Base Stations in WSNs

Hui Wu   Sabbir Mahmud

School of Computer Science and Engineering
The University of New South Wales
email{huiw, sabbirm}@cse.unsw.edu.au

Abstract. We study the problem of deploying k base stations in a wireless sensor network such that the maximum shortest hop distance from all the sensor nodes to their designated base stations is minimised. We propose a 2-approximation algorithm for this problem, and prove that a $(2 - \epsilon)$-approximation algorithm does not exist unless $P = NP$ holds. The time complexity of our 2-approximation algorithm is $O(n^2 \log n)$, where n is the number of sensor nodes of the wireless sensor network. In the special case where k is 1, we propose an $O(n^2)$ time algorithm that is guaranteed to find the optimal location of the base station. Furthermore, we show that our previous heuristic for balancing clusters of sensors can be modified to significantly improve the performance of our 2-approximation algorithm.

1 Introduction

A wireless sensor network (WSN) consists of a set of distributed autonomous sensor nodes communicating with each other via radio signals. All the sensor nodes of a WSN work cooperatively to monitor physical or environmental conditions such as temperature. The applications of WSNs range from environmental monitoring to industrial monitoring. In some applications such as border surveillance, bushfire detection and traffic control, several thousands of sensor nodes might be deployed over the monitored region. The diameter of the monitored region can be several kilometres.

In wireless sensor networks, most of the energy of a sensor node is consumed by communications. The communication range of a sensor node is determined by its transmit power. The higher the transmit power, the longer the communication range [1]. In WSNs, sensor nodes are typically battery-powered. In order to save power, the transmit power of a sensor node is kept low, leading to a short communication range. As a result, multi-hop communication between each sensor and its base station is needed in large scale WSNs. In multi-hop communications, a sensor node may spend most of its energy on relaying data packets. Hence, it is important to shorten the hop distance between each source sensor node and its base station. The hop distance can be dramatically reduced by deploying multiple base stations. In order to deploy multiple base stations, we need to partition all the sensor nodes into multiple disjoint clusters, and place one base station in each cluster. Each sensor node sends its data only to its designated base station. The location of the base station of each cluster is very important. If a base station is

deployed far from the data sources, many sensor nodes are required to relay data packets and the energy consumption of those sensor nodes will be significantly increased. Therefore, it is an important design issue to find the best location of a base station.

In this paper, we study the problem of deploying k base stations in a WSN such that the maximum shortest hop distance from all the sensor nodes to their designated base stations is minimised. This problem is important due to two major reasons. Firstly, in some time-critical applications of WSNs, the data collected by each sensor node must be delivered to the base station in a timely manner. The maximum shortest hop distance from all the sensor nodes to their designated base stations specifies the upper bound on the latencies of data gathering. As a result, a shorter maximum shortest hop distance leads to a lower maximum latency of data collection. Secondly, a longer maximum shortest hop distance leads to higher energy consumption of the data collection. This problem is similar to the classical k-center problem [16]. In the k-center problem, we have a set of cities and k warehouses. The objective is to select k cities as the locations of k warehouses such that the cost of any city to its nearest warehouse is minimised. If we restrict the locations of base stations to the locations of sensor nodes, the problem of optimally deploying k base stations such that the maximum shortest hop distance is minimised, is reduced to the k-center problem. Nevertheless, the optimal locations may not be the locations of sensor nodes. Consider the following example. There are 5 sensor nodes deployed uniformly on the circumference of a circle with a radius of R, where R is the maximum communication range of all the sensor nodes. We want to deploy a base station such that the maximum shortest hop distance from all the sensor nodes to the base station is minimised. If we restrict the candidate locations of the base station to the locations of sensor nodes, the maximum shortest hop distance is 2 in an optimal deployment of the base station. Clearly, if we place the base station at the center of the circle, the maximum shortest hop distance is only 1. Therefore, we cannot reduce this problem to the classical k-center problem.

We make the following major contributions in this paper.

1. We propose the first 2-approximation algorithm for the problem of deploying k base station such that the shortest hop distance of all the sensor nodes to their designated base stations is minimised, and prove that no $(2 - \epsilon)$-approximation algorithm for this problem exists unless $P = NP$ holds.
2. We present the first polynomial-time algorithm for deploying one base station in a cluster of sensor nodes such that the maximum shortest hop distance from all the sensor nodes to the base station is minimised.
3. We present a modified version of our previous heuristic for balancing clusters of sensors [14]. We have simulated our 2-approximation algorithm and the cluster balancing heuristic on 171 instances of different distributions. Our simulation results show that our cluster balancing heuristic performs significantly better that our 2-approximation algorithm.

This paper is organised as follows. Section 2 presents definitions and the network model. Section 3 proposes a polynomial-time algorithm for optimally deploying one base station in a cluster of sensor nodes. Section 4 proposes a 2-approximation algorithm. Section 5 describes a heuristic for balancing clusters of sensors. Section 6 dis-

cusses related work. Section 7 presents our simulation results, and Section 8 concludes this paper.

2 Definitions and Network Model

A WSN consists of a set of n identical sensor nodes, each of which is located in a $2D$ plane. The location of each sensor node is known. All the sensor nodes have the same transmit power. Therefore, they have the same maximum communication distance R. We assume that there are no communication barriers between any two adjacent sensor nodes. Therefore, a sensor node v_i can directly communicate with a sensor node v_j if the Euclidean distance between v_i and v_j is not greater than R. There are k base stations to be deployed in a target WSN. As a result, all the sensor nodes need to be partitioned into k clusters with one base station in each cluster. A sensor node in each cluster sends its data to its designated base station only. If the Euclidean distance between a sensor node and its base station is greater than R, the data of the sensor node must be transmitted via other sensor nodes to the base station.

Definition 1. *The connectivity graph of a WSN is an undirected graph $G = <V, E>$, where $V = \{v_i : i = 1..n$ and v_i is a sensor node\}, and $E = \{(v_i, v_j) : if the Euclidean distance between v_i and v_j is not greater than R\}.*

Without loss of generality, we assume that the connectivity graph G of the target WSN is connected. A connectivity graph is a unit disk graph [17].

Definition 2. *Given two sensor nodes v_i and v_j, the shortest hop distance from v_i to v_j is the length of the shortest path from v_i to v_j in the connectivity graph.*

Let P be a set of n distinct points called sites, in a $2D$ plane. The Voronoi diagram [15] of P is the subdivision of the plane into n cells, one for each site. A point q lies in the cell of a site $p_i \in P$ iff the Euclidean distance between q and p_i is less than the Euclidean distance between q and p_j ($p_j \in P$ and $i \neq j$). The edges of the Voronoi diagram are all the points in the plane that are equidistant to the two nearest sites.

Definition 3. *A sensor node v_i is a neighbour of a sensor node v_j if the Voronoi cells of v_i and v_j share a Voronoi edge.*

Definition 4. *Let V be a set of n sensor nodes in a 2D plane and $C_i(i = 1, 2, \cdots, k)$ be k disjoint clusters of V. A cluster C_i is a neighbour of a cluster C_j if there are two sensor nodes $v_s \in C_i$ and $v_t \in C_j$ such that v_s is a neighbour of v_t.*

Definition 5. *Given a cluster C_i of sensor nodes and a sensor node $v_j \notin C_i$, the Euclidean distance from v_j to C_i, denoted $d(v_j, C_i)$, is $\min\{d(v_k, v_j) : v_k \in C_i$ and $d(v_k, v_j)$ is the Euclidean distance between v_k and $v_j\}$.*

Definition 6. *Given a WSN and a point p on a 2D plane, the unit sensor density of p is the number of sensor nodes that are one hop away from p. The maximum unit sensor density of the WSN is the largest unit sensor density of all the points on the 2D plane.*

Throughout this paper, we assume that the maximum unit sensor density is a constant. In WSNs, the maximum communication distance is typically short in order to reduce the energy consumption of data transmissions. Hence this assumption is reasonable.

3 Single Base Station Deployment Problem

Deploying a single base station in a set of sensor nodes is a building block of our heuristic for optimally deploying k base stations. This problem is described as follows. Given a set of sensor nodes and a base station, find the optimal location of the base station such that the maximum shortest hop distance from all the sensor nodes to the base station is minimised.

The key idea of our algorithm for this problem is to find the candidate locations of the base station such that one candidate location must be the optimal location of the base station. To find all possible candidate locations, we consider each pair of sensor nodes v_i and v_j. If the Euclidean distance between v_i and v_j is greater than $2R$, where R is the maximum communication distance of all the sensor nodes, we will ignore this pair. Otherwise, we find the candidate circles of v_i and v_j. A candidate circle of v_i and v_j is a circle that satisfies the following two constraints: 1) The radius of the circle is R. 2) v_i and v_j are on its circumference. The center of a candidate circle is a candidate location of the base station. Notice that for each pair of sensor nodes at most two candidate circles exist. If the Euclidean distance of a pair of sensor nodes is equal to $2R$, only one candidate circle of this pair exists. After finding all the candidate locations, our algorithm will search for the best candidate location of the base station. The best candidate location is the one that minimises the maximum shortest hop distance from all the sensor nodes to the base station placed at this candidate location. The algorithm is shown as follows.

Algorithm *SingleBS(V)*
Input *: A set $V = \{v_1, v_2, \cdots, v_m\}$ of m sensor nodes in a $2D$ plane and a base station.*
Output *: The optimal location of the base station such that the maximum shortest hop distance from all the sensor nodes to the base station at the optimal location is minimised.*

begin
  $C = \emptyset$;
  for *each pair of sensor nodes $(v_i, v_j)(v_i, v_j \in V)$* **do**
      if *the Euclidean distance between v_i and $v_j \leq 2R$* **then**
        Find the candidate circles C_1 and C_2 of v_i and v_j;
        Let c_1 and c_2 be the centers of C_1 and C_2;
        $C = C \cup \{c_1\} \cup \{c_2\}$;
  for *each candidate location $c_i \in C$* **do**
      Place the base station at c_i;
      Construct the connectivity graph $G(V \cup \{c_i\})$ of all the sensor nodes and the base station;
      Compute the maximum shortest hop distance MSHD(c_i) of all the sensor nodes in V
      to the base station located at c_i;
  Let c_j be the candidate location with the smallest maximum shortest hop distance;
  return $(c_j, MSHD(c_j))$;
end

Theorem 1. *The algorithm SingleBS(V) is guaranteed to find the optimal location of the base station.*

Proof. Assume that the optimal location is c_{opt}. Let $S = \{v_1, v_2, \cdots, v_r\}$ be the set of sensor nodes that are one hop away from the base station at the optimal location c_{opt}.

Draw a circle C_{opt} with the radius R and the center c_{opt}. According to the definition of the maximum communication distance R, all the sensor nodes in S must be either in C_{opt} or on the circumference of C_{opt}. Next, we show that there is a candidate location c_k generated by our algorithm such that the set of sensor nodes that are one hop away from c_k is equal to S. Consider the following three possible cases.

1. There are two sensor nodes $v_i, v_j \in S$ such that v_i and v_j are on the circumference of C_{opt}. In this case, c_{opt} is one of our candidate locations.
2. Only one sensor node $v_i \in S$ is on the circumference of C_{opt}. Turn the circle C_{opt} clockwise around v_i until another sensor $v_j \in S$ is on the circumference of C_{opt}. Now all the sensor nodes in S are still in C_{opt}, and this case reduces to Case 1.
3. No sensor node is on the circumference of C_{opt}. Arbitrarily select a sensor node v_t, and move C_{opt} along the straight line $c_{opt}v_t$ until one sensor node in S is on the circumference of C_{opt}. Now all the sensors in S are still in C_{opt} or on the circumference of C_{opt}. Hence, this case reduces to Case 2.

Based on the above discussions, we can conclude that such a candidate location c_k exists. For each sensor node v_i, any path from v_i to c_k or c_{opt} must include a sensor node in S. Therefore, the shortest hop distance from v_i to c_k is equal to that from v_i to c_{opt}. As a result, c_k is also an optimal location of the base station.

Theorem 2. *Given a cluster of m sensor nodes, the time complexity of the algorithm SingleBS(V) is $O(m^2)$.*

Proof. For a cluster with m sensor nodes there are $m(m-1)/2$ pairs of sensor nodes. Therefore, it takes $O(m^2)$ time to find all the candidate locations. At most two candidate locations exist for each pair of sensor nodes. Under our assumption on the maximum unit sensor density, for each sensor node v_i the number of neighbouring sensor nodes which are two hops away from v_i is at most p^2, where p, the maximum unit sensor density, is a constant. Therefore, the total number of candidate locations is $O(m)$. The connectivity graph $G(V \cup \{c_i\})$ of all the sensor nodes and the base station at the candidate location c_i can be constructed as follows. Firstly, construct the connectivity graph $G(V)$ of all the sensor nodes, which takes $O(m^2)$ time. Secondly, add a new node for the base station at the candidate location c_i and the new edges between the base station and all the sensor nodes that can directly communicate with the base station to $G(V)$. After constructing the connectivity graph $G(V \cup \{c_i\})$, we can use breadth-first search to compute the total shortest hop distance of all the sensor nodes to the base station in $O(e)$ time, where e is the number of edges in $G(V \cup \{c_i\})$. Given the maximum unit sensor density p, $e \leq pm$ holds. Since p is a constant, it takes $O(m)$ time to compute the total shortest hop distance. As a result, the time complexity of our algorithm is $O(m^2)$.

4   A 2-Approximation Algorithm

In this section, we will propose a 2-approximation algorithm for the problem of deploying k base stations such that the maximum shortest hop distance from all the sensor nodes to their designated base stations is minimised. Furthermore, we will prove that a

2-approximation algorithm is the best unless $P = NP$ holds.

Let V be the set of all the sensor nodes in a WSN and C the set of all the candidate locations defined in Section 3. The connectivity graph of all the sensor nodes and all the candidate locations is a undirected graph $G = (V \cup C, E)$, where $E = \{(v_i, v_j):$ $v_i, v_j \in V \cup C$ and the Euclidean distance between v_i and v_j is at most $R\}$. In a connectivity graph G, the shortest hop distance between two sensor nodes v_i and v_j is the number of edges of the shortest path between v_i to v_j that does not include any candidate location. The shortest hop distance between a sensor nodes v_i and a candidate location v_j is the number of edges of the shortest path between v_i to v_j that does not include any candidate location except v_j. The shortest hop distance between two candidate locations v_i and v_j is the number of edges of the shortest path between v_i to v_j that does not include any candidate location except v_i and v_j.

An undirected graph is a complete graph if there is an edge between every pair of distinct nodes. A subgraph of an undirected graph is a clique if it is a complete subgraph.

Definition 7. *Given a natural number w and a connectivity graph G, $G(w) = (V \cup C, E(w), W(w))$, where $E(w) = \{(v_i, v_j):$ the shortest hop distance between v_i and v_j in G is at most $w\}$, and $W(w) = \{w_{ij} : (v_i, v_j) \in E(w)$ and w_{ij} is the shortest hop distance between v_i and v_j in $G\}$.*

Definition 8. *Given an undirected graph G, a dominating set of G is a subset V' of the nodes of G such that for every node v_j in G there exists an edge (v_i, v_j), where $v_i \in V'$. A dominating set with the smallest size is a minimum dominating set.*

Definition 9. *Given an undirected graph G, an independent set of G is a subset V' of the nodes of G such that for any two nodes $v_i, v_j \in V'$, the edge (v_i, v_j) does not exist in G. An independent set V' is a maximal independent set if for any node $v_i \notin V'$, $V' \cup \{v_i\}$ is not an independent set.*

Note that the maximal independent set is different from the maximum independent set [18].

Lemma 1. *Given a natural number w and a node v_i in $G(w)$, let S be a set of the adjacent nodes of v_i in $G(w)$. The subgraph formed by $S \cup \{v_i\}$ of $G(2w)$ is a clique.*

Proof. For any two adjacent nodes v_s and v_t of v_i in $G(w)$, we have $SHD(v_s, v_t) \leq SHD(v_i, v_s) + SHD(v_i, v_t)$, where $SHD(v_i, v_j)$ is the shortest hop distance between v_i and v_j. By the definition of $G(w)$, both $SHD(v_i, v_s) \leq w$ and $SHD(v_i, v_t) \leq w$ hold. Therefore, $SHD(v_s, v_t) \leq 2w$ holds. Hence, the edge (v_s, v_t) is in $G(2w)$ by the definition of $G(2w)$. In addition, the edges (v_i, v_s) and (v_i, v_t) are also in $G(2w)$. As a result, the subgraph formed by $S \cup \{v_i\}$ of $G(2w)$ is a clique.

It is easy to see that the optimal k base station deployment problem is equivalent to the problem of finding the smallest w such that there is a dominating set of k candidate locations in $G(w)$. Unfortunately, the minimum dominating set problem is NP-complete even for unit disk graphs [17]. We resort to finding a maximal independent set of k candidate locations in $G(w)$ for the smallest value of w, which can be done in binary search. The pseudo code of our 2-approximation algorithm is shown as follows.

Algorithm *2-approximation(V,k)*
Input *: A set $V = \{v_1, v_2, \cdots, v_n\}$ of sensor nodes and k base stations.*
Output *: k disjoint clusters of all the sensor nodes, k locations of base stations such that*
the maximum shortest hop distance from all the sensor nodes to their designated base stations is
at most twice the optimum of the maximum shortest hop distance, and the maximum shortest hop
distance from all the sensor nodes in the k clusters to their designated base stations.

begin
  Find the set $C = \{c_1, c_2, \cdots, c_m\}$ of all candidate locations of k base stations.
  $low = 1; high = n; S = \{\};$
  until *high=low+1* **do**
      $mid = (\lfloor high + low \rfloor)/2; A = \{\}; T = \{\}; s = 0;$
      construct $G(mid)$;
      while *there is a candidate location in $G(mid)$* **do**
          $s = s + 1;$
          Select a candidate location $c_i \in G(mid)$ with the maximum degree;
      /* c_i is the location of the base station for cluster $T(c_i)$ */
          $A = A \cup \{c_i\}; T(c_i) = \{\};$
          for *every edge (c_i, v_j) $G(mid)$* **do**
              if $v_j \in V$ **then**
                  $T(c_i) = T(c_i) \cup \{v_j\};$
              remove v_j and all the edges incident to it from $G(mid)$;
          remove c_i from $G(mid)$;
          $T = T \cup \{T(c_i)\};$
      if $s \leq k$ **then**
          $S = T; high = mid; MSHD = mid;$
      else
          $low = mid;$
  return A, S and $MSHD$;
end

Theorem 3. *The algorithm 2-approximation(V, k) constructs a set of k candidate locations for k base stations such that the maximum shortest hop distance from all the sensor nodes to their designated base stations is at most twice that found by an optimal algorithm.*

Proof. Assume that the maximum shortest hop distance computed by an optimal algorithm is w^*. We prove that the maximum shortest hop distance $MSHD$ computed by our algorithm is at most $2w^*$. Consider the case when our algorithm is executed on $G(2w^*)$. Let S_{opt} be the set of k candidate locations computed by an optimal algorithm. By Lemma 5.1, for every candidate location $c_i \in S_{opt}$, the subgraph formed by $\{c_i\} \cup A$ of $G(2w^*)$ is a clique, where A is a set of the adjacent nodes of c_i in $G(w)$. All these k cliques cover all the nodes in $G(2w^*)$. Therefore, the size of any maximal independent set of $G(2w^*)$ is at most $|S_{opt}|$. In other words, $G(2w^*)$ is the worst case when our algorithm finds a set of k candidate locations with a maximum shortest hop distance of $2w^*$.

Theorem 4. *The time complexity of the algorithm 2-approximation is $O(n^2 \log n)$, where n is the number of sensor nodes.*

Proof. There are $n(n-1)/2$ distinct pairs of sensor nodes. Therefore, it takes $O(n^2)$ time to find all the candidate locations. Clearly, the time complexity of one iteration of the **while** loop is dominated by constructing $G(mid)$. The number of nodes of $G(mid)$ is $n + |C|$. As we explained before, the total number of candidate locations is $O(n)$. constructing $G(mid)$ takes $O(n^2)$ time. The number of iterations of the **while** loop is $\log n$. Hence, the **while** loop takes $O(n^2 \log n)$ time. As a result, the time complexity of our algorithm is $O(n^2 \log n)$.

Theorem 5. *It is an NP-complete problem to find a $(2 - \epsilon)$-approximation algorithm for the k base station deployment problem.*

Proof. We use a reduction from the minimum dominating set problem to the k base station deployment problem. Consider a unit disk graph $G' = (V', E')$. The corresponding k base station problem is as follows. There are a set $V = V' = \{v_1, v_2, \cdots, v_n\}$ of n sensor nodes and a set $C = \{c_1, c_2, \ldots, c_n\}$ of n candidate locations. The set E of edges of the connectivity graph $G = (C \cup V, E)$ is constructed as follows.

1. For every edge $(v_i, v_j) \in E'$, both $(v_i, v_j) \in E$ and $(c_i, v_j) \in E$ hold.
2. For every $v_i \in V$ $(c_i, v_i) \in E$ holds.

The weighted graph $G(1)$ is constructed as follows.

1. All the nodes of $G(1)$ are the same as in G
2. All the edges of $G(1)$ are the same as in G.
3. Every edge weight is 1.

The weighted graph $G(2)$ is constructed as follows.

1. $G(2)$ is a complete graph.
2. For every edge (v_i, v_j) in $G(2)$, if it is in G, it's edge weight is 1. Otherwise, its edge weight is 2.

In the above k base station deployment problem, the smallest maximum shortest hop distance from all the sensor nodes to their designated base stations in an optimal solution is either 1 or 2. Clearly, there is a set of k candidate locations leading to a maximum shortest hop distance of 1 iff there is a minimum dominating set with size k in G'. Suppose that there exists an $(2 - \epsilon)$-approximation algorithm. By applying the $(2 - \epsilon)$-approximation algorithm to this k-base station deployment problem, we can get the optimal solution. As a result, we can solve the minimum dominating set problem in polynomial time, which contradicts the fact that the minimum dominating set problem in unit disk graphs is NP-complete [17].

5 A Heuristic for Balancing Clusters

The k clusters constructed by our 2 approximation algorithm may not be balanced. In this section, we presents a modified version of our previous heuristic for balancing clusters of sensor proposed in [14]. Our modified heuristic starts with the k clusters constructed by our 2-approximation algorithm, and repeatedly moves a sensor node from

a cluster with a larger maximum shortest hop distance to a neighbouring cluster with a smaller maximum shortest hop distance until clusters become balanced. In each iteration, a modifiable cluster C_i with the smallest maximum shortest hop distance among all the clusters in C is selected. A cluster C_i is modifiable if there exist a neighbouring cluster C_j with $\text{MSHD}(C_j) > \text{MSHD}(C_i)$ and a sensor node $v_k \in C_j$ such that $\text{MSHD}(C_j - \{v_k\}) \leq \text{MSHD}(C_j)$ and $\text{MSHD}(C_i \cup \{v_k\}) \leq \text{MSHD}(C_j)$ hold, where $MSHD(C_s)$ is the maximum shortest hop distance from all the sensor nodes to the base station at the optimal location of the cluster $C(s)$. In other words, moving v_k from C_j to C_i may reduce the maximum shortest hop distance of both clusters. If such a modifiable cluster does not exist, all the clusters are balanced and the algorithm terminates. If such a modifiable cluster C_i exists, the algorithm will select the neighbouring cluster C_j with the largest maximum shortest hop distance among all the neighbouring clusters of C_i and find the set Q of sensor nodes in C_j which are the neighbouring sensor nodes of C_i. Then it keeps moving a sensor node in Q with the smallest Euclidean distance to C_i from C_j to C_i until no sensor node in Q can be moved from C_j to C_i. A sensor node $v_k \in Q$ is moved from C_j to C_i only if v_k satisfies the following constraints:

1. $\text{MSHD}(C_i \cup \{v_k\}) \leq \text{MSHD}(C_j)$.
2. $\text{MSHD}(C_j - \{v_k\}) \leq \text{MSHD}(C_j)$.

The first constraint ensures that after v_k is moved from C_j to C_i, the maximum shortest hop distance of C_i does not exceed the maximum hop distance of C_j. The second constraint guarantees that after moving v_k from C_j to C_i, the maximum shortest hop distance of C_j will not increase. The reason why we need the second constraint is that if a sensor node $v_s \in Q$ is on the shortest paths of other sensor nodes in C_j to the base station, moving v_s from C_j to C_i may increase the maximum shortest hop distance of C_j. The modified heuristic is shown in pseudo code as follows.

Algorithm *Balancing(C, L)*
Input : *A set $C = \{C_1, C_2, \cdots, C_k\}$ of k disjoint clusters and a set $L = \{c_1, c_2, \cdots, c_k\}$ of the optimal locations of k base stations, where $c_i(i = 1, 2, \cdots, k)$ is the optimal location of the base station of the cluster C_i.*
Output : *A set of k disjoint clusters with smaller maximum total shortest hop distance and the optimal location of the base station of each cluster.*

begin
  / A is the set of non-modifiable clusters at this moment. */*
  / B is the set of modifiable clusters at this moment. */*
  $A = \{\}; B = C; // C = A \cup B$ holds all the time.
  for *each cluster $C_i \in C$* **do**
      $modifiable(C_i) = true$;
  while *$B \neq \emptyset$* **do**
      Select a cluster C_i with the minimum MSHD(C_i) and $modifiable(C_i) = true$ from B;
      $S = \{C_s : C_s \in C$ and C_s is a neighbouring cluster of $C_i\}$.
      Find $C_j \in S$ with the maximum MSHD(C_j);
      $Q = \{v_s : v_s \in C_j$ and v_s is a neighbouring sensor node of $C_i\}$;
      $NodeMoved(C_i) = 0$;
      while *$Q \neq \emptyset$* **do**
         Select a sensor node $v_s \in Q$ with the smallest Euclidean distance to C_i;

    if *MSHD($C_j - \{v_s\}$)≤MSHD(C_j) && MSHD($C_i \cup \{v_s\}$) ≤MSHD(C_j)* **then**
        $C_i = C_i \cup \{v_s\}; C_j = C_j - \{v_s\};$
        Find the new optimal locations of the base stations of C_i and C_j;
        Recalculate MSHD(C_i) and MSHD(C_j);
        NodeMoved(C_i) = 1; $Q = Q - \{v_s\}$;
  if *NodeMoved(C_i)> 0* **then**
    for *each neighbouring cluster C_j of C_i* **do**
      if $modifiable(C_j) == false$ **then**
        $modifiable(C_j) = true; A = A - \{C_j\}; B = B \cup \{C_j\};$
  else
    $modifiable(C_i) = false; A = A \cup \{C_i\}; B = B - \{C_i\};$
end

6 Related Work

The problem of deploying multiple base stations in a large scale sensor network has been studied in a number of papers. Various objective functions have been used. [8] proposes a heuristic for deploying multiple mobile base stations to maximise the lifetime of the sensor network. The total lifetime of the network is divided into equal period of time known as rounds and all mobile base stations change their locations at the beginning of every round. An ILP (Integer Linear Programming) formulation is proposed to find the locations of base stations such that the maximum energy spent by each node in a round is minimised. [2] proposes a heuristic for maximising the life time of a WSN. The heuristic consists of a LP (Linear Programming) formulation for positioning multiple base stations in a sensor network and an ILP formulation for routing traffic flow from all of the sensors to these multiple sink nodes. Since the ILP problem is NP-complete, ILP-based approaches are not applicable to large scale WSNs.

[4] studies the problem of placing a base station in a WSN such that the smallest lifetime of any sensor nodes is maximised. It proposes an approximation algorithm that can guarantee $(1 - \epsilon)$-optimal network lifetime performance for base station placement problem with any desired error bound $\epsilon > 0$. The proposed $(1 - \epsilon)$-optimal approximation algorithm is based on several novel techniques. The first technique is to discretize cost parameter with performance guarantee. Subsequently, the continuous search space can be broken up into a finite number of subareas. The second technique is to exploit the cost property of each subarea and represent it by a novel notion called fictitious cost point, each with guaranteed cost bounds.

[13] proposes three energy efficient low-complexity algorithms to determine the locations of the base stations on the boundary of a WSN. They are a top-Kmax algorithm, an algorithm for maximising the minimum residual energy, and an algorithm for minimising the residual energy difference. The paper shows that the proposed base stations placement algorithms provide increased network lifetimes and amount of data delivered during the network lifetime compared to single base station scenario as well as multiple static base stations scenario.

[12] proposes an algorithm and a heuristic for placing k base stations in an optimal way such that the average Euclidean distance between the sensor nodes and their base stations is minimised. The algorithm assumes that each base station knows the locations

of all the sensor nodes, and the heuristic assumes that each base station only knows the locations of its neighbouring sensor nodes and other base stations. However, average Euclidean distance is not a good optimisation objective function. There are two key reasons. Firstly, a sensor node with a shorter Euclidean distance to its base station may have a longer hop distance to its base station. Secondly, it is possible that no sensor node can communicate with the base station at the location that minimises the average Euclidean distance of all the sensor nodes to the base station. Consider a WSN with a ring topology, i.e., all the sensor nodes are located on a ring. If the radius of the ring is greater than the maximum communication distance of the sensor nodes, no sensor nodes can communicate with the base station at the center of the ring.

[11] studies the problem of placing k base stations in an optimal way such that the total latency of all the sensor nodes to their gateways is minimised. The authors proposed two heuristics for the problem using genetic algorithms. However, their genetic algorithm has no worst-case performance guarantee.

[14] proposes a heuristic for deploying multiple base stations such that the maximum energy consumption of any cluster is minimised. The heuristic uses a greedy approach to create k initial clusters, and then repeatedly moves a sensor node from a bigger cluster to a smaller cluster until all clusters become balanced. However, the approximation ratio of the heuristic is unknown.

7   Simulation Results

In order to evaluate the performance of our 2-approximation algorithm and our heuristic, we used QualNet Network Simulator [19] to generate 171 different network instances with three different distributions, namely, grid, uniform and random distributions. We used three different numbers of base stations, i.e., 2, 4 and 6. For each of the uniform and random distributions, we varied the number of sensor nodes from 100 to 600 with an increment of 25 nodes, and generated 21 network instances with the 3 different numbers of base stations, leading to a total of 63 network instances. Altogether, we generated $63 + 63 = 126$ network instances for the uniform and random distributions. For the network instances in the grid distribution, the numbers of sensor nodes are the square of integers ranging from 10 to 24, i.e., we generated a total of $15 * 3 = 45$ instances for the grid distribution.

The hardware platform we used for our simulations is Intel Core 2 Duo processor with a clock frequency of 3 GHz and 4 GB RAM. The maximum communication range of all the sensor nodes is 50 meters. We implemented our 2-approximation algorithm and heuristic in C++, and executed them to compute the maximum shortest hop distance for each of the 171 network instances.

The simulation results are shown in Figures 1, 2 and 3 for the uniform, random and grid distributions, respectively. In each figure, the horizontal axis represents the number of sensor nodes, and the vertical axis denotes the maximum shortest hop distance. Figures (a), (b) and (c) of each figure show the simulations results for 2 base stations, 4 base stations, and 6 base stations, respectively.

We also calculated the relative improvement of our heuristic over our 2-approximation algorithm based on our simulation results. For all the 171 network instances, the aver-
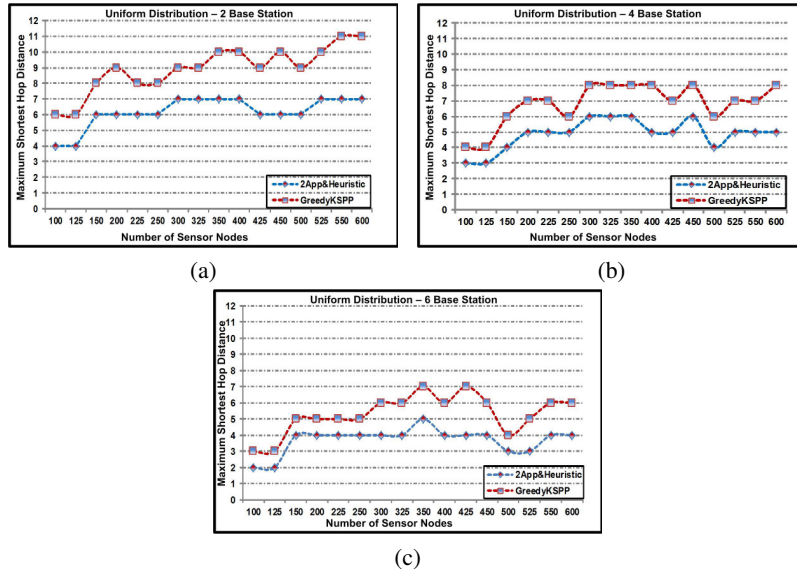
Fig. 1: Simulation results for the network instances in uniform distribution

age improvement, the smallest improvement, and the maximum improvement of our heuristic over our 2-approximation algorithm are 35%, 20%, and 50%, respectively. Since our 2-approximation algorithm is guaranteed to compute a maximum shortest hop distance that is at most twice the optimal value, our simulation results imply that the performance of our heuristic is close to that of an optimal algorithm.

8 Conclusion

In this paper, we propose a best possible 2-approximation algorithm for deploying k base stations such that the maximum shortest hop distance from all the sensor nodes to their designated base stations is minimised. The time complexity of our approximation algorithm is $O(n^2 \log n)$, where n is the number of sensor nodes of the WSN. In the spacial case where k is equal to 1, we propose an optimal algorithm for this problem. We also show that the performance of our 2-approximation algorithm can be significantly improved by a modified version of our previous heuristic for balancing clusters of sensors.

References

1. Q. Gao, K. J. Blowa, D. J. Holdinga, I. W. Marshallb, and X. H. Penga. Radio Range Adjustment for Energy Efficient Wireless Sensor Networks. Ad Hoc Networks, Volume 4, Issue 1, 2006, pp. 75-82.
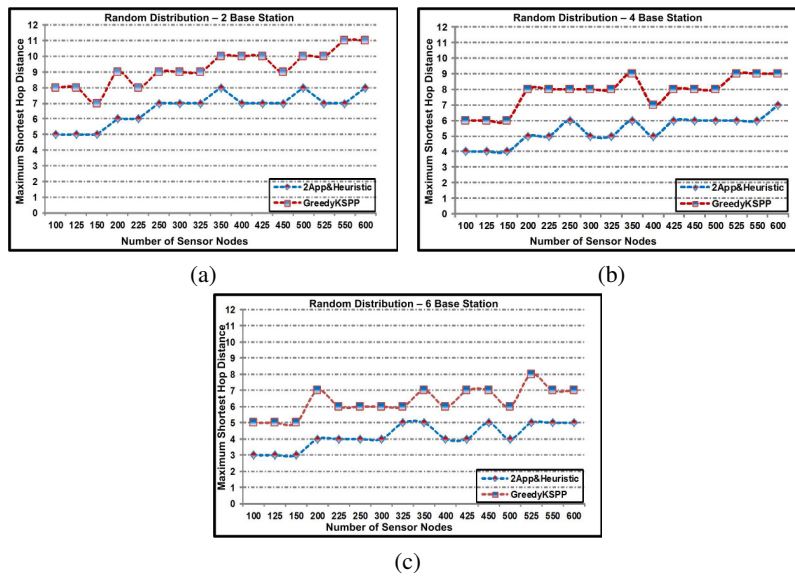
Fig. 2: Simulation results for the network instances in random distribution

2. Kim, H, Seok, Y, Choi, N, Choi, Y and Kwon, T. Optimal Multisink Positioning and Energy-efficient Routing in Wireless Sensor Networks. Lecture Notes in Computer Science (LNCS), Volume 3391, pp. 264-274.

3. Hou, Y. T., Yi Shi, Sherali, H. D. Midkiff, S. F. On Energy Provisioning and Relay Node Placement for Wireless Sensor Networks. IEEE Transactions on Wireless Communications, Volume 4, Issue 5, Sept. 2005, pp. 2579-2590.

4. Hou, Y. T., Yi Shi. Optimal Base Station Placement in Wireless Sensor Networks. ACM Transactions on Sensor Networks, Volume 5, Issue 4, November 2009, pp. 1-24.

5. Liu, J, Reich, J, and Zhao, F. Collaborative In-network Processing for Target Tracking. EURASIP Journal on Applied Signal Processing, 2003, 378–391, 2003, Hindawi Publishing Corp.

6. Gaurav Gupta and Mohamed Younis. Performance Evaluation of Load-Balanced Clustering of Wireless Sensor Networks. In Proceedings of the 10th International Conference on Telecommunication, 2003.

7. Alon Efrat, Sariel Har-Peled, Joseph S. B. Mitchell. Approximation Algorithms for Two Optimal Location Problems in Sensor Networks. In Proceedings of the 3rd International Conference on Broadband Communications, Networks and Systems, 2005.

8. Gandham, S, R, Dawande, M, Prakash, R, and Venkatesan, S. Energy Efficient Schemes for Wireless Sensor Networks with Multiple Mobile Base Stations. In Proceedings of IEEE Global Telecommunications Conference, 2003, vol. 1, pp. 377-381.

9. Hu, W, Chou, C, T, Jha, S and Bulusu, N. Deploying Long-lived and Cost Effective Hybrid Sensor Networks. 1st Workshop on Broadband Advanced Sensor Networks.

10. Qiu, L, Chandra, R, Jain, K and Mahdian, M. Optimising the Placement of Integration Points in Multi-hop Wireless Sensor Networks. In Proceedings of International Conference on Network Protocols (ICNP), 2004.
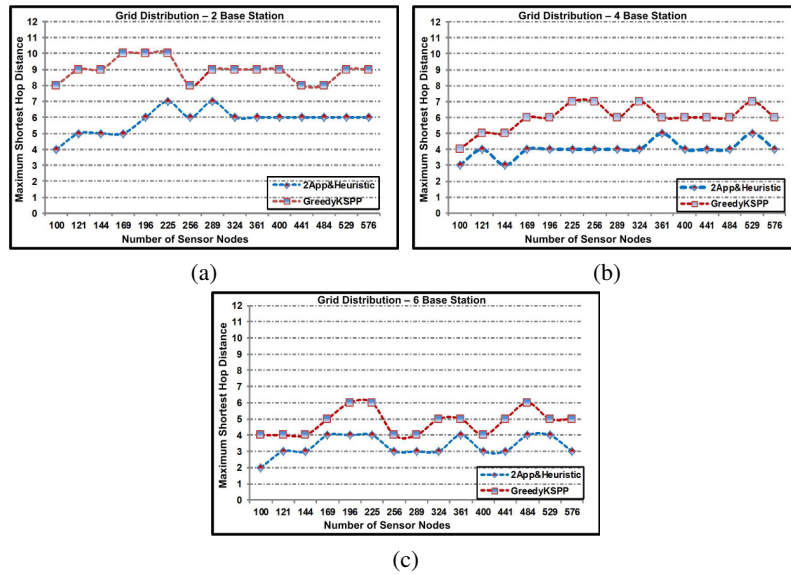
Fig. 3: Simulation results for the network instances in gird distribution

11. Yousef, W and Younis, M. Intelligent Gateways Placement for Reduced Data Latency in Wireless Sensor Networks. IEEE International Conference on Communications (ICC), 2007, pp. 3805-3810.

12. Vincze, Z, Vida, R and Vidacs, A. Deploying Multiple Sinks in Multi-hop Wireless Sensor Networks. IEEE International Conference on Pervasive Services, 2007, pp. 55-63.

13. A. P. Azad and A. Chockalingam. Mobile Base Stations Placement and Energy Aware Routing in Wireless Sensor Networks. In Proceedings of the 2006 IEEE Wireless Communications and Networking Conference, Las Vegas, Nevada, USA, 2006, pp. 264-269.

14. Sabbir Mahmud, Hui Wu and Jingling Xue. Efficient Energy Balancing Aware Multiple Base Station Deployment for WSNs. In Proceedings of the 8th European Conference on Wireless Sensor Networks, Bonn, Germany, 2011, pp. 179-194.

15. J. R. Sack and J. Urrutia, Handbook of Computational Geometry, Elsevier Science, Netherlands, 2000.

16. Dordit S. Hochbaum and Favid B. shmoys. A Best Possible Heuristic for the k-center Problem. Mathematics of Operations Research, Volume 16, Issue 2, May 1985, pp. 180-184.

17. Brent N. Clark, Charles J. Colbourn, David S. Johnson. Unit Disks Graph. Discrete Mathematic. Volume 86, Issue 1-3, Dec. 14, 1990.

18. Michael R. Garey and David S. Johnson. Computers and Intractability: A Guide to the Theory of NP-completeness. W. H. Freeman and Co. New York, USA, 1979.

19. http://www.scalable-networks.com.