Discrete Mathematics, Algorithms and Applications
Vol. 1, No. 2 (2009) 175–191

A BETTER APPROXIMATION FOR MINIMUM AVERAGE ROUTING PATH CLUSTERING PROBLEM IN 2-D UNDERWATER SENSOR NETWORKS

WEI WANG

Department of Mathematics, Xi'an Jiaotong University
710049, P. R. China
wang_weiw@163.com

DONGHYUN KIM*, JAMES WILLSON†, BHAVANI THURAISINGHAM‡
and WEILI WU§

Department of Computer Science, University of Texas at Dallas
2601 North Floyd Road, Richardson, TX 75083, USA
**donghyunkim@student.utdallas.edu*
†jkw053000@utdallas.edu
‡bxt043000@utdallas.edu
§weiliwu@utdallas.edu

Accepted 3 April 2009

Previously, we proposed Minimum Average Routing Path Clustering Problem (MARP-CP) in multi-hop USNs. The goal of this problem is to find a clustering of a USN so that the average clustering-based routing path from a node to it nearest underwater sink is minimized. We relaxed MARPCP to a special case of Minimum Weight Dominating Set Problem (MWDSP), namely MWDSP-R. In addition, we showed the Performance Ratio (PR) of α-approximation algorithm for MWDSP-R is 3α for MARPCP. Based on this result, we showed the existence of a $(15 + \epsilon)$-approximation algorithm for MARPCP. In this paper, we first establish the NP-completeness of both MARPCP and MWDSP-R. Then, we propose a PTAS for MWDSP-R. By combining this result with our previous one, we have a $(3 + \epsilon)$-approximation algorithm for MARPCP.

Keywords: Polynomial time approximation scheme; underwater sensor networks; wireless network clustering.

Mathematics Subject Classification 2000: 68W25

1. Introduction

Recently, Underwater Sensor Networks (USNs) are being studied as a promising solution to ease the difficulties of monitoring the underwater world. USNs are also

‡This work is supported in part by the AFOSR A9550-08-1-0260.
§This work is supported in part by the IIS-0513669, CCF-0750992, and CCF-0621829.

investigated for a wide range of potential applications, both civilian and military [1]. Usually, a USN consists of a set of static sensor nodes and several mobile nodes such as Unmanned Underwater Vehicles (UUVs) and Autonomous Underwater Vehicle (AUVs). Depending on applications, the topology of a USN can be abstracted using a 2-D or 3-D graph. For example, for the ocean floor mineral exploration, a set of underwater sensor nodes is deployed on the ocean bottom and forms a 2-D wireless network. In the case of underwater intrusion detection, a group of sensor nodes are floating in the middle of water and form a 3-D wireless network. Since 2-D USNs are usually deployed on the floor of the deep ocean, the networks are far from the outside of water. Therefore, it has at least one super node called UnderWater Sink (UW-Sink). As like normal underwater sensor nodes, the UW-Sink has horizontal links to communicate with neighboring sensor nodes on the floor of the ocean. Furthermore, the node has long range vertical links to reach the out-of-water sinks on the surface of the sea, which are usually buoys or ships. The out-of-water sinks are connected with each other or onshore stations using conventional communication technologies such as radio or satellite signals. Therefore, we can think of the UW-Sinks as gateways or relay nodes between the in-water networks (the USNs) and the out-of-water networks.

For underwater communication, some USNs use light or radio signals. However, most of them are using acoustic channels since neither light nor radio signals can travel far inside water. The energy consumption of sending a message using an acoustic underwater channel increases exponentially proportional to the traveling distance of the message. For this reason, multi-hop communication is preferred in USNs. Generally, it is very expensive or almost impossible to replace the battery of deployed underwater sensor nodes. In addition, existing energy harvesting schemes for Terrestrial Sensor Networks (TSNs) do not provide enough energy to the nodes inside deep water. As a result, energy efficiency becomes a crucial issue of USNs.

In wireless networks having mobility such as USNs, clustering ensures basic level system performance, such as throughput and delay [2]. In addition, it makes the networks more energy-efficient by reducing collisions. In the case of Wireless Sensor Networks (WSNs) such as TSNs or USNs, each clusterhead can fuse messages from its members to decrease the total amount of messages traveling over the networks. It is well-known that nodes inside the same cluster are geographically close and thus aggregating messages from the nodes does not decrease the accuracy of the information of the original messages significantly. Note that if we merge the already-fused data, the precision of them will be lowered [3].

Once an aggregated message is prepared by a clusterhead, it can be sent to a UW-Sink in several ways. At any case, it is more desirable to make the expected travel distance and corresponding energy consumption from a clusterhead to its nearest UW-Sink shorter. In the clustering scheme for USNs in [4], each clusterhead communicates with a UW-Sink directly. To make such communication more energy-efficient, the authors select a set of clusterheads using a cost function, which is a function of Euclidean distance between nodes in a cluster and their clusterhead and between the clusterhead and its nearest UW-Sink. However, when underwater sensor nodes are deployed over the vast ocean, it is energy consuming for each clusterhead to communicate with a UW-Sink directly. This is because acoustic channels consume energy exponentially proportional to message transmission range and thus, in USNs, multi-hop transmission is preferred over directed transmission.

Alternatively, messages can move from a clusterhead to its nearest UW-Sink over the smallest hop path between them. For this purpose, we can use existing proactive or re-active routing schemes for wireless networks. However, it is known that, in mobile wireless networks such as USNs, the overhead of both schemes is quite huge [4]. A clustering-based routing scheme is a routing scheme over a clustered network in which any routing path contains only clusterheads [5]. This scheme may generate a routing path whose hop length is at most three times longer than that of a shortest hop path [6]. However, it is known to be energy-efficient since it can decrease both the amount of control messages and routing related information and prevent collision significantly. Furthermore, such a clustering-based routing path is resilient against node mobility since it will be changed only when any clusterhead on the path disappears. In these reasons, the clustering-based routing scheme is suitable to mobile WSNs such as USNs.

In [6], we studied the problem of clustering multi-hop USNs requiring high data precision to best support the clustering-based routing on them. The goal of this problem is to find a clustering of a USN such that the expected clustering-based routing path of a message from a source to its nearest UW-Sink is minimized. We formulated this as Minimum Average Routing Path Clustering Problem (MARPCP). Then, we relaxed MARPCP to a restricted form of Minimum Weighted Dominating Set Problem (MWDSP-R). We also showed that any α-approximation algorithm for MWDSP-R is 3α-approximation algorithm for MARPCP, and the Performance Ratio (PR) of the $(5 + \epsilon)$-approximation algorithm for MWDSP in [7] is $(15 + \epsilon)$ for MARPCP.

In this paper, we propose a $(3 + \epsilon)$-approximation algorithm for MARPCP. We notice that when MARPCP is relaxed to MWDSP-R, the node weight in MWDSP-R has a special characteristic. Based on this observation, we design a Polynomial Time Approximation Scheme (PTAS) for MWDSP-R. Note that so far there has been no PTAS for MWDSP in any case. By combining this result with our previous one in [6], we show that our algorithm is a $(3 + \epsilon)$-approximation algorithm for MARPCP. Last, we show that both MARPCP and MWDSP-R are NP-complete.

The rest of this paper is organized as follows. In Sec. 2, we introduce our assumptions, notations, and definitions. Section 3 presents some preliminary information. Section 4 describes a PTAS for MWDSP-R and corresponding analysis. Finally, we make conclusions and state some future work in Sec. 5.

2. Notations, Definitions, and Assumptions

Now, we introduce some notations which will be used frequently in the rest of this paper. $G = (V, E)$ is a 2-D UDG representing a USN except UW-Sinks. For any two nodes $u, v \in V$, $Hopdist(v, u)$ is the hop distance between v and u over the shortest path between them, which is denoted by $Mindist(v, u)$. Similarly, the Euclidean distance between u and v is defined as $Eucdist(u, v)$. For a subset $S, T \subseteq V$, $Hopdist(S, T)$ is the minimum of $Hopdist(u, v)$ among every $u \in S$ and $v \in T$ pair. For a node $v \in V$, $N^r(v) = \{u | u \in V$ and $Hopdist(v, u) \leq r\}$. We will use $N(v)$ and $N^1(v)$ interchangeably. For a subset $S \subseteq V$, $N^r(S) = \{u | u \in V$ and $\exists w \in S, Hopdist(u, w) \leq r\}$. For a node $v \in V$, $N[v] = \{v\} \cup N(v)$. For a subset $S \subseteq V$, $N[S] = S \cup N(S)$ and G_S is a subgraph induced by S. $CH(v)$ is the clusterhead of v.

Now, we introduce some definitions and several assumptions of this paper. We also introduce more notations if needed. First, we assume every node in a USN is using the same maximum transmission range for energy-efficiency. Also, we assume the following data-centric routing model. Each clusterhead receives reports from each of its cluster members, fuses the messages into one message, and forwards the summary to its nearest UW-Sink. It is known that since all the members in the same cluster are geographically near and data made by them are quite similar, such local data fusion can decrease redundant messages greatly without sacrificing the precision of the data acquired from the USN. However, in this paper, we assume the data gained from the USN has to be precise and thus do not permit further merging of already-fused messages.

In addition, we assume following well-known clustering-based routing scheme which is also used in some previous literatures such as [5]. That is, casual cluster-based routing methods for multi-hop wireless networks use $Routepath(s, d)$, which is defined as below, instead of $Minpath(s, d)$. Suppose an edge weighted graph $G' = (V', E')$, in which each node $v \in V'$ is corresponding to a cluster $C_v \subseteq V$ and there exists an edge $(v, u) \in E'$ if C_v and C_u are two neighbors in G. For each on each $(v, u) \in E'$, $w(v, u) = Hopdist(CH(v), CH(u))$ is the weight of the edge. Then, the total number of hops a message travels from s to d by the clustering-based routing scheme is

$$Routedist(s, d) = Hopdist(s, CH(s)) + \sum_{p \in MWP(CH(s), CH(d))} w(p) + Hopdist(d, CH(d)),$$

where $MWP(CH(s), CH(d))$ is the path p with the minimum weight between the corresponding nodes of $CH(s)$ and $CH(d)$ in G' and $w(p)$ is the total weight of the path p. Also, $Routepath(s, d)$ is the path from s to d with length $Routedist(s, d)$ in G. It is very well-known that the communication cost in multi-hop wireless networks is highly related to the number of hops a message moves [8]. Therefore, in this paper, we use hop distance as a metric instead of Euclidean distance.

Now, we formally introduce the cost function of our problem. Suppose $\{s_1, \ldots, s_n\}$ and $\{U_1, \ldots, U_m\}$ are the set of sensor nodes and UW-Sinks, respectively. Also, let $CH_1, \ldots, CH_c$ be a set of clusterheads selected among n sensor nodes. It is reasonable to assume that each node has the same probability to generate a report message, and thus given each node has generated one report, the total number of messages that move over the network is

$$\sum_{1\leq j\leq c} \min_{1\leq k\leq m} Routedist(CH_j, U_k) + n - c$$
$$= n + \sum_{1\leq j\leq c} \min_{1\leq k\leq m} (Routedist(CH_j, U_k) - 1),$$

and the expected number of hops that a message will move from its source to the nearest UW-Sink is

$$\frac{1}{n}\left(n + \sum_{1\leq j\leq c} \min_{1\leq k\leq m} (Routedist(CH_j, U_k) - 1)\right).$$

Observe that the total number of messages generated by cluster members and sent to clusterheads is $n - c$. Once a set of messages sent by members arrives at a clusterhead, they will be fused into one summary message and forwarded to the nearest UW-Sink of the clusterhead. Since n is a fixed constant for any input graph, we can remove this in our cost function. Now, we formally define our problem using the cost function.

Definition 2.1 (Minimum Average Routing Path Clustering Problem [6]). Given a group of sensor nodes $\{s_1, \ldots, s_n\}$ and UW-Sinks $\{U_1, \ldots, U_m\}$ on the Euclidean plane, the goal of Minimum Average Routing Path Clustering Problem (MARPCP) is to find a set of clusterheads such that the expected number of hops for a message delivery using the clustering-based routing scheme is minimized. That is, the goal is to find a set of clusterheads $\{CH_1, \ldots, CH_c\}$ to minimize

$$\sum_{1\leq j\leq c} \min_{1\leq k\leq m} (Routedist(CH_j, U_k) - 1) \text{ such that } \forall\, i, \exists j, Hopdist(s_i, CH_j) \leq 1.$$

Observe that $Routedist(CH_j, U_k)$ cannot be computed before clusterheads are chosen. Therefore, finding a set of clusterheads of a given USN such that the objective function of MARPCP is minimized looks tricky. The proof of NP-completeness of MARPCP is presented in Appendix A.

Definition 2.2 (Dominating Set). Given a graph $G = (V, E)$, a Dominating Set (DS) is a subset of nodes $V' \subseteq V$ such that $\forall\, v \in V$, either $v \in V'$ or $\exists u \in V \backslash V'$, $(v, u) \in E$.

In the rest of this paper, $D(V)$ means a dominating set of V. In addition, for a subset $S \subseteq V$, $D_{opt}(S)$ is the smallest subset of $D(V)$ dominating S. Note that $D_{opt}(S) \subseteq S$ is not necessarily true. However, $D_{opt}(S) \subseteq N[S]$ has to be always true.

Definition 2.3 (Minimum Weighted Dominating Set Problem). Given an edge weighted graph $G = (V, E)$ with a weight function $w : V \to \mathbb{R}_0^+$, where $\mathbb{R}_0^+$ is the set of non-negative real numbers, Minimum Weighted Dominating Set Problem(MWDSP) is to find a DS D of G such that its total weight $\sum_{v \in D} w(v)$ is minimized.

Note that if the weights of all vertices are equal to 1, we call it Minimum Dominating Set Problem (MDSP). Computing a DS is one well-known way to cluster wireless networks. In this paper, to handle the high complexity of MARPCP, we relax MARPCP to MWDSP as follows. For each node $v \in V$, compute $\min_{1 \leq i \leq m}(Hopdist(v, U_i) - 1)$ and set this as a weight $w(v)$ on v. Through this process, we can induce a graph with non-negative weights from a graph G formed by the set of sensor nodes. Observe that when we relax a MARPCP instance to a MWDSP instance, the weight function on MWDSP has a special property. In this paper, we will use MWDSP-R to mention the MWDSP having this special property. It is clear to see that a feasible solution of MWDSP-R is also a feasible solution of MARPCP. This is because both of them are DSs for the same graph (when weights are ignored). The proof of NP-completeness of MWDSP-R is given in Appendix B. Here are some additional definitions which will be used later.

Definition 2.4 (Growth-Bounded Graph). A graph $G = (V, E)$ is f-growth-bounded if there exists a polynomial function $f(r)$ such that for all $v \in V$, $|MIS(N^r(v))| \leq f(r)$, where $f(r)$ only depends on a positive integer r and is independent from the structure of graph G.

It is known that both UDGs and UBGs are growth-bounded graphs [9].

Definition 2.5 (2-Separated Partition). Given a graph G, a collection of subsets $\{V_1, \ldots, V_k\}$ of V such that $\cup_{\forall i} S_i \subseteq V$ is a 2-Separated Partition (2-SP) of G, if $\forall\, i, j, i \neq j$, $Hopdist(V_i, V_j) \geq 2$.

3. Preliminaries

In WSNs such as TSNs or USNs, energy is a very scarce resource. It is well-known that clustering WSNs can improve the efficiency of the networks [2]. In detail, routing information is related to only clusterheads, whose number is relatively small, and maintained and handled by the clusterheads. Therefore, routing-related overhead is reduced. In WSNs, each clusterhead can be also used as a local data fusion point, which merges messages from its cluster members and forwards their summary to its nearest sink. This method can reduce the number of messages in the networks significantly and hence helps the WSNs to be more energy-efficient.

Usually, a clustering scheme can be incorporated with a routing scheme called a clustering-based routing scheme [5]. That is, messages move from the head of a cluster to the head of adjacent cluster following the shortest path between the clusterheads. When a message needs to move through multiple clusters, it repeats visiting the clusterhead of each cluster on its way to the destination. In this way, two nodes can exchange messages through their clusterheads. In this scheme, only clusterheads maintain routing-related information. In addition, routing paths consist of only clusterheads, whose number is significantly smaller than the size of the whole network. This kind of clustering-based routing scheme is highly beneficial to WSNs since the overhead caused by message routing can be greatly reduced. Furthermore, the routing scheme is resilient against the mobility of the networks since a path should be recovered only when a clusterhead on the path disappears.

In [4], a clustering scheme is proposed in the context of routing scheme to extend the lifetime of a USN. They assumed a direct communication model between clusterheads and UW-Sinks. To better cluster a given USN so that the energy-efficiency of the USN can be improved, they chose each clusterhead based on the Euclidean distance from each cluster member to the candidate node and the Euclidean distance from the candidate node to its nearest UW-Sink. Note that due to the energy-inefficiency issue of direct communication model in USNs, their scheme may not be adopted well to relatively large USNs which prefer multi-hop communications.

In [6], we studied the problem of clustering multi-hop USNs to improve the energy-efficiency of clustering-based routing on the USN. We proposed Minimum Average Routing Path Clustering Problem (MARPCP), in which hop distance is used as a metric instead of Euclidean distance. Next, we relaxed MARPCP to a special case of Minimum Weighted Dominating Set Problem (MWDSP-R). We showed that a $(5+\epsilon)$-approximation algorithm for MWDSP in [7] is a $(15+\epsilon)$-approximation algorithm for MARPCP.

In [10], the authors proposed a PTAS for the MDSP in growth-bounded graphs such as UDGs and UBG. Their main idea is as follows. Suppose we have a 2-separated collection $\{S_1, \ldots, S_k\}$ of V. Remember that for each i, $D_{opt}(S_i) \subseteq S_i$ is not necessarily true, but $D_{opt}(S_i) \subseteq N[S_i]$ has to be always true. Hence, $D_{opt}(S_1), D_{opt}(S_2), \ldots, D_{opt}(S_k)$ are disjoint. Combining this with the facts that $D_{opt}(V) \cap N[S_i]$ is a dominating set of S_i and $D_{opt}(S_i)$ is the set with minimum cardinality that dominates S_i, we have $|D_{opt}(S_i)| \leq |D_{opt}(V) \cap N[S_i]|$ and hence, $\sum_{i=1}^{k} |D_{opt}(S_i)| \leq |D_{opt}(V)|$. Observe that $\bigcup_{i=1}^{k} D_{opt}(S_i)$ is not necessarily a DS of G. Now, enlarge each S_i's to T_i's such that 1) $S_i \subseteq T_i$, 2) $|D_{opt}(T_i)| \leq (1+\epsilon)|D_{opt}(S_i)|$, and 3) $\bigcup_{i=1}^{k} D_{opt}(T_i)$ is a DS of $V(G)$. Then, we have

$$\left|\bigcup_{i=1}^{k} D_{opt}(T_i)\right| \leq \sum_{i=1}^{k} |D_{opt}(T_i)| \leq (1+\epsilon) \sum_{i=1}^{k} |D_{opt}(S_i)| \leq (1+\epsilon)|D_{opt}(V)|,$$

and $\bigcup_{i=1}^{k} D_{opt}(T_i)$ is a $(1+\epsilon)$-approximation of $D_{opt}(V)$.

To compute $\{S_1, \ldots, S_k\}$ and $\{T_1, \ldots, T_k\}$, their algorithm sets $i = 1$ and repeats 1) picking an arbitrary node v_i, 2) finding the minimum integer r_i satisfying

$|D_{opt}(N^{r_i+2}(v_i))| \leq (1+\epsilon)|D_{opt}(N^{r}(v_i))|$, 3) set $S_i = N^{r_i}(v_i)$ and $T_i = N^{r_i+2}(v_i)$, and 4) increase i by 1 as long as there is any vertex left in the remaining graph $G_{V \setminus \cup_{1\leq j \leq i} T_j}$.

The key point of the above method lies in the fact that given a constant $\epsilon > 0$ and a polynomial function $f(r)$, there exists a uniform constant $r(\epsilon, f) = O(\frac{1}{\epsilon} \ln \frac{1}{\epsilon})$ for all core nodes v_i which is dependent only on $f(r)$ and ϵ, such that $r_i \leq r(\epsilon, f)$ and $|D_{opt}(T_i)| \leq (1+\epsilon)|D_{opt}(S_i)|$. Thus, $|D_{opt}(T_i)|$ can be computed exactly locally by enumeration with time complexity $n^{O(f(r(\epsilon,f)))}$, since $|D_{opt}(T_i)| \leq |MIS(N(T_i))| = O(f(r(\epsilon, f)))$.

4. A PTAS for MWDSP in Growth-Bounded Graphs with Special Node Weights

In this section, we use the ides of designing PTAS for MDSP in [10] to have a PTAS for MWDSP in growth-bounded graphs with special node weights. The input of this algorithm is a UDG (or UBG) $G = (V, E)$ with positive integer weights on the nodes and a set of m UW-Sinks, $\{U_1, U_2, \ldots, U_m\}$. Remember that by the relaxation rule from MARPCP, the weight on each node is the hop distance between the node and its nearest UW-Sink. We notice that this algorithm cannot be directly generalized for the MWDSP, since there may not exist a uniform constant $r(\epsilon, f)$ due to the fact that the ratio of the maximum weight to a minimum weight of two nodes can be as large as $O(n)$. Hence to solve this kind of MWDSP in growth-bounded graphs, some new ideas are needed.

We first introduce our algorithm (Algorithm 1) and show that this is a PTAS for our problem. Since the structure of this algorithm is very similar with that in [10], introduced in detail in the previous section, we skip the description of this

Algorithm 1 PTAS-MDSP ($G = (V, E), \epsilon$)

$U \leftarrow V$ and $i \leftarrow 1$
while $U \neq \emptyset$ **do**
  Pick a *core node* $v_i \in U$ with the smallest weight
  $r_i \leftarrow 0$
  while $w(D_{opt}(N^{r_i+2}(v_i))) > (1+\epsilon)w(D_{opt}(N^{r_i}(v_i)))$ **do**
    /* each $D_{opt}(N^{r_i}(v_i))$ and $D_{opt}(N^{r_i+2}(v_i))$ are computed exhaustively */
    $r_i \leftarrow r_i + 1$
  end while
  $S_i \leftarrow N^{r_i}(v_i)$ and $T_i \leftarrow N^{r_i+2}(v_i)$
  $i \leftarrow i + 1$ and $U \leftarrow U \setminus T_i$
end while
Exhaustively compute $D_{opt}(T_i)$ for each i
Return $\cup_{\forall i} D_{opt}(T_i)$

algorithm and directly show that it is a PTAS for our problem. For this purpose, we will prove the following. In Lemma 4.3 we show that there is some constant r such that $\forall i, r_i \leq r$ in Algorithm 1. By showing this, we can guarantee the algorithm will stop. In Lemma 4.2, we will show that the output of Algorithm 1 is a feasible solution of MWDSP, which is a dominating set of G. In Lemma 4.4 and Theorem 4.5, we show that given a constant ϵ, Algorithm 1 is a polynomial time algorithm for MWDSP with performance ratio $(1+\epsilon)$, and thus a PTAS. Then, by Theorem 4.1 below, we can show that Algorithm 1 is a $(3+\epsilon)$-approximation algorithm for MARPCP.

Theorem 4.1. [6] *Any α-approximation algorithm for* MWDSP-R *is 3α-approximation algorithm for* MARPCP.

Lemma 4.2. $\cup_{\forall i} D_{opt}(T_i)$ *is a dominating set for graph G.*

Proof. By Algorithm 1, $V(G) = \cup_{\forall i} T_i$. Thus, for each $v \in V$, there exists some T_i such that $v \in T_i$. Clearly, v is dominated by $D_{opt}(T_i)$. □

Lemma 4.3. *Let G be a growth-bounded graph with growth function f. For any any real number $\epsilon > 0$, there exists a constant $r(\epsilon, f)$ which depends only on ϵ and f, but is independent of the topology of G, such that $w(D_{opt}(N^{r+2}(v))) \leq (1+\epsilon)w(D_{opt}(N^r(v)))$ for each $v \in W$, where the set of core nodes W is choose as Algorithm 1.*

Proof. We prove the lemma by contradiction. If not, then there exists an $\epsilon_0 > 0$ and a node $v_0 \in W$ such that

$$w(D_{opt}(N^{r+2}(v_0))) > (1+\epsilon_0)w(D_{opt}(N^r(v_0))) \quad \text{for } r = 0, 1, 2, \ldots.$$

By the rule of selecting core nodes in Algorithm 1, v_0 is the node with the smallest weight, say k, in the remaining graph induced by U. Thus, the largest weight (the corresponding highest level) of nodes in $N^r(v_0)$ is at most $k+r$. Note $w(v_0) = k$, the weight of the neighbors of v_0 is at least k. We have $w(D_{opt}(N^0(v_0))) = k$ and $w(D_{opt}(N^1(v_0))) = k$; this is because the core node v_0 dominates v_0 and all its neighbors. Thus, if r is even, we have

$$\begin{aligned} w(D_{opt}(N^{r+2}(v_0))) &> (1+\epsilon_0)w(D_{opt}(N^r(v_0))) \\ &> \cdots > (1+\epsilon_0)^{1+r/2} w(D_{opt}(N^0(v_0))). \end{aligned}$$

If r is odd, we have

$$\begin{aligned} w(D_{opt}(N^{r+2}(v_0))) &> (1+\epsilon_0)w(D_{opt}(N^r(v_0))) \\ &> \cdots > (1+\epsilon_0)^{(1+r)/2} w(D_{opt}(N^1(v_0))). \end{aligned}$$

Furthermore, note that $MIS(N^{r+2}(v_0))$ is a dominating set of $N^{r+2}(v_0)$ and $D_{opt}(N^{r+2}(v_0))$ is the dominating set of $N^{r+2}(v_0)$ with minimum total weights. It follows that

$$w(D_{opt}(N^{r+2}(v_0))) \leq w(MIS(N^{r+2}(v_0)) \leq (k+r+2)|MIS(N^{r+2}(v_0))|$$
$$\leq (k+r+2)f(r+2).$$

If r is even, we obtain $(1+\epsilon_0)^{1+r/2}w(D_{opt}(N^0(v_0))) \leq (k+r+2)f(r+2)$, it follows that

$$(1+\epsilon_0)^{1+r/2} \leq \frac{k+r+2}{k}f(r+2) \leq (r+3)f(r+2).$$

Similarly, if r is odd, we obtain $(1+\epsilon_0)^{(1+r)/2} \leq (r+3)f(r+2)$. In either case, since the left hand side of the inequality is exponential in r while the right hand side is a polynomial in r, if r is sufficiently large, both inequalities cannot be true — a contradiction. □

Lemma 4.4. *The constant in Lemma* 4.3 *is bounded by* $r(\epsilon, f) = O(\frac{1}{\epsilon}\ln\frac{1}{\epsilon})$. *The time complexity of Algorithm* 1 *is* $n^{O((\frac{1}{\epsilon}\ln\frac{1}{\epsilon})^{k+1})}$, *where* k *is the degree of* f.

Proof. Consider the inequality $(1+\epsilon)^x \leq Cx^m$, where $\epsilon > 0, m \geq 1$ is fixed. Let $x_0 = m\frac{1}{\epsilon}\ln\frac{1}{\epsilon}$. We show there exists an ϵ_0, such that $(1+\epsilon)^{x_0} \leq Cx_0^m$ whenever $0 < \epsilon < \epsilon_0$. Taking logarithm on both sides and using the fact $\ln(1+\epsilon) \leq \epsilon$ for any $\epsilon \geq 0$, we obtain that $x_0\epsilon \leq \ln C + m\ln x_0$ implies $(1+\epsilon)^{x_0} \leq C{x_0}^m$. The former is equivalent to

$$m\ln\frac{1}{\epsilon} \leq \ln C + m\ln m + m\ln\frac{1}{\epsilon} + m\ln\ln\frac{1}{\epsilon}.$$

Clearly, there exists an ϵ_0 such that the above inequality holds whenever $0 < \epsilon < \epsilon_0$. This shows that when ϵ is sufficiently small, $r(\epsilon, f) \leq x_0 = m\frac{1}{\epsilon}\ln\frac{1}{\epsilon} = O(\frac{1}{\epsilon}\ln\frac{1}{\epsilon})$.

The time complexity of Algorithm 1 is dominated by enumerating the local optimal solution in T_i with the largest radius r_i. Note

$$w(v_i)|D_{opt}(T_i)| \leq w(D_{opt}(T_i)) \leq (w(v_i)+r_i+2)|MIS(T_i)|$$
$$\leq (w(v_i)+r_i+2)f(r_i+2).$$

It follows that $|D_{opt}(T_i)| \leq (r_i+3)f(r_i+2) = O((\frac{1}{\epsilon}\ln\frac{1}{\epsilon})^{k+1})$. □

Theorem 4.5. *Algorithm* 1 *is a* PTAS *for the* MWDSP-R *in growth-bounded graphs.*

Proof. Let $S_i = N^{r_i}(v_i)$ $(i = 1, 2, \ldots, k)$ be the set of nodes constructed in Algorithm 1. Clearly $\{S_1, S_2, \ldots, S_k\}$ is a 2-separated collection of V. Let D_{opt} be the optimal solution to MWDSP-R. Then $D_{opt} \cap N(S_i)$ dominates S_i. By the definition, $D_{opt}(S_i)$ is the set of nodes with minimum total weights that dominates S_i. Thus, we obtain $w(D_{opt}(S_i)) \leq w(D_{opt} \cap N(S_i))$. It follows from the fact that

$w(D_{opt}(T_i)) \le (1+\epsilon)w(D_{opt}(S_i))$ that

$$w\left(\bigcup_{i=1}^{k} D_{opt}(T_i)\right) \le \sum_{i=1}^{k} w(D_{opt}(T_i)) \le (1+\epsilon)\sum_{i=1}^{k} w(D_{opt}(S_i))$$
$$\le (1+\epsilon)w(D_{opt}(V)),$$

Hence, $\bigcup_{i=1}^{k} D_{opt}(T_i)$ is a $(1+\epsilon)$-approximation of $D_{opt}(V)$. □

5. Conclusion

In this paper, we study the problem of clustering multi-hop USNs to improve the energy-efficiency of the clustering-based routing scheme on the USNs. In [6], we formally defined this problem as MARPCP. We first relaxed MARPCP to a special MWDSP (MWDSP-R), and showed the PR of existing α-approximation algorithms for MWDSP-R is 3α for MARPCP. Using this result, we proved that the $(5+\epsilon)$-approximation algorithm for MWDSP in [7] is a $(15+\epsilon)$-approximation algorithm for MARPCP. In this paper, we propose a PTAS for MARPCP. By combining this result with our previous one, we have a $(3+\epsilon)$-approximation algorithm for MARPCP. As a future work, we are considering an energy model for this problem so that the algorithm can be energy-aware. In addition, we are also interested in a generalized MARPCP, in which each clusterhead dominate nodes within d-hops.

Appendix A

Suppose we have a boolean expression $C = C_1 \bigwedge C_2 \bigwedge \cdots C_m$, where $C_i = x_i \bigvee y_i \bigvee z_i$, where $x_i, y_i, z_i \in \{u_1, \bar{u}_1, u_2, \bar{u}_2, \ldots, u_n, \bar{u}_n\}$. Then, the 3-SATisfiability (3-SAT) problem is to determine if there exists a truth assignment for each variable such that all clauses become true (satisfied).

PLANAR 3-SAT is a special kind of 3-SAT in which the graph induced by a PLANAR 3-SAT instance by the rules introduced below is a planar graph. That is, for each clause, we construct a triangle whose edge node represents a literal in the clauses. For each variable, we construct a line whose edge nodes represent a variable and its negation, respectively. We establish an edge between a literal in a clause and a variable (or its negation) if the variable is used as a literal in the clause. It is known that PLANAR 3-SAT is NP-complete [11].

Now, we show that PLANAR 3-SAT is still NP-complete although we enforce each variable to be used in at most three clauses. For simplicity, we will mention this problem as 3-PLANAR 3-SAT. Take a planar embedding of $G(C)$ for some instance of PLANAR 3- SAT. Let (u, c_1),$(u, c_2), \ldots, (u, c_k)$ be the edges adjacent to variable-vertex u in the graph $G(C)$, which are arranged in clockwise order according to the planar embedding. Now introduce new variables $w_1, w_2, \ldots, w_k$ and clauses $\{w_1 \bigvee \bar{w}_2\}, \{w_2 \bigvee \bar{w}_3\}, \ldots, \{w_{k-1} \bigvee \bar{w}_k\}, \{w_k \bigvee \bar{w}_1\}$; Replace literals $v, \bar{v}$ with in C_i by $w_i, \bar{w}_i$, respectively, for $i = 1, 2, \ldots, k$. It is easy to see that the modified formula C' is satisfiable if and only if C is satisfiable. Moreover, $G(C')$ is a planar graph.

In this way, the occurrences of variables in any clauses can be reduced to at most three.

It is trivial to show MARPCP is in the complexity class NP. To show MARPCP is complete in NP, we will construct a UDG $G(C)$ from a 3-PLANAR 3-SAT instance C in polynomial time and show that C is satisfiable if and only if $G(C)$ has an optimal solution of certain value. Next, we give the details of our construction.

A.1. *Construction of a graph representing a literal*

For each literal, say x_i, construct a graph as in Fig. 1(a). For $x_i = u_i$, x_i is true if and only if the corresponding node is chosen as a dominator (if $x_i = \bar{u}_i$, the reverse holds). Note that the size of any feasible solution of MARPCP in this graph is at least three. That is, we need nodes a, c, and g or a, e, and g.

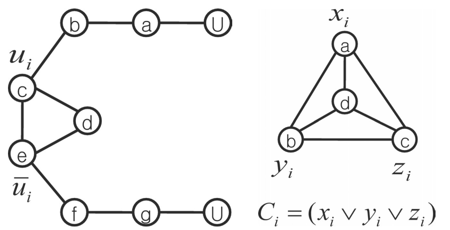
A.2. *Construction of a graph representing a clause and three corresponding literals*

For each clause $C_i = x_i \bigvee y_i \bigvee z_i$, we construct a graph $G(C_i)$ with $3[(2+3l)+9]+4$ nodes associated with it, which is constructed as follows: First construct a graph with four nodes–a complete graph K_4, the three vertices corresponding to the three literals x_i, y_i and z_i, respectively, and the the center vertex corresponding to the clause C_i (See Fig. 1(b)). Finally, connect each literal with its corresponding vertex using a path of length $2+3l$, where l is a non-negative integer (See Fig. 2). We obtain the graph $G(C_i, x_i, y_i, z_i)$ corresponding to C_i (See Figs. 3(a),(b)). In practice, to draw such a graph l should be large enough and the path may not be a straight path, but can be curved.

A key of our proof is the following lemma:

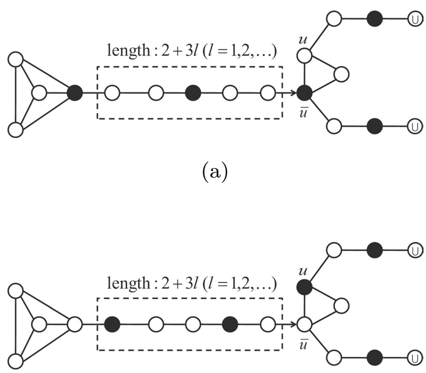
Lemma A.1. *We can find a satisfiable assignment of C_a if and only if we can find an optimal solution of* MARPCP *in $G(C_a, x_a, y_a, z_a)$.*

Proof. Suppose we have a satisfiable assignment for $C_a = (x_a \vee y_a \vee z_a) = (u_i, u_j, u_k)$. Now, we choose the set of clusterheads as follows. First, we pick all



(a) (b)

Fig. 1. Basic components.



(a)

(b)

Fig. 2. Connecting a clause and a variable.

nodes adjacent to UW-Sinks as clusterheads. Next, we pick $Corner(G(x_a))$, which is a node in $G(x_a)$ and is connected to the path toward the complete graph K_4 representing C_a, if true is assigned to x_a. We do the same thing for $Corner(G(y_a))$ and $Corner(G(z_a))$. Then, starting from each clusterhead closest to $Corner(G(x_a))$, we select every 3rd nodes in the path to the K_4. If $Corner(G(x_a))$ is selected, we will have l nodes on the path. Otherwise, we will have $l+1$ nodes on the path. At last, if $Corner(G(x_a))$ is selected, we select $Corner(G(C_a), u_i)$, which is a node on the K_4 and connected to the path toward $G(x_a)$. Let the set of clusterheads selected so far be CL. Note that for each assignment on u_i, u_j, u_k does not change the cost of CL. However, if false is assigned to all of u_i, u_j, and u_k, then we must pick another node inside K_4 and this will be an extra cost. Therefore, we will have an optimal solution of MARPCP in $G(C_a, x_a, y_a, z_a)$ only if we have a satisfiable assignment on C_a. Proving the reverse direction is similar. Therefore, this theorem holds true. □

A.3. *Connecting all components together*

Now let C be an instance of PLANAR 3-SAT with each variable appearing in at most three clauses. We construct an instance of MARPCP, that is, a UDG $G(C)$ on the Euclidean space consisting of sensor nodes and UW sinks such that a satisfiable assignment for C implies an optimal solution T of MARPCP on $G(C)$ and vice versa. First we show how to construct graph $G(C)$ from a planar embedding of $G(C)$.

(1) Remove the edges $(u_1, u_2), (u_2, u_3), \ldots, (u_{n-1}, u_n), (u_n, u_1)$.
(2) Replace each clause-vertex $C_i = x_i \bigvee y_i \bigvee z_i$ in $G(C)$ with a clique K_4 representing the clause C_i (as done previously), replace each vertex-variable with a graph shown in Fig. 1(b). Replace each edge connecting clause-vertex and vertex-variable with a path of length $2 + 3l$.

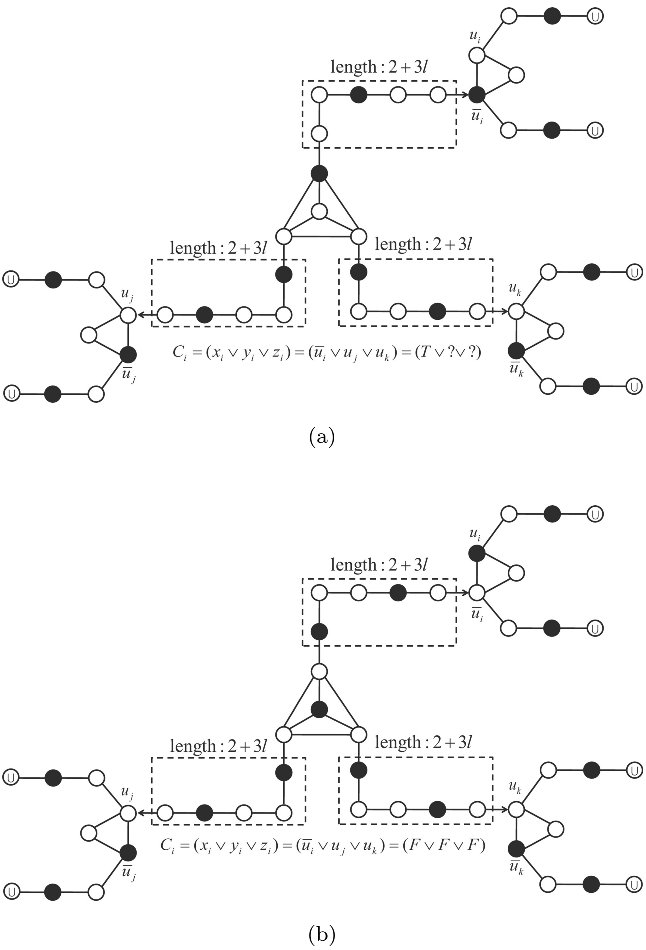


Fig. 3. Connecting each clause and related variables.

(3) Note that since each variable-vertex is connected to at most three clause-vertices, we can always draw it in the Euclidean plane as in Fig. 4.
(4) After the above operations, we can obtain the UDG $G(C)$.

Theorem A.2. *We have a satisfiable assignment of* C *if and only if we have an optimal solution of* MARPCP *in* $G(C)$.

Proof. The proof of this theorem is similar to that of Lemma A.1 and thus we skip this. □

Theorem A.3. MARPCP *is* NP*-complete in UDGs.*

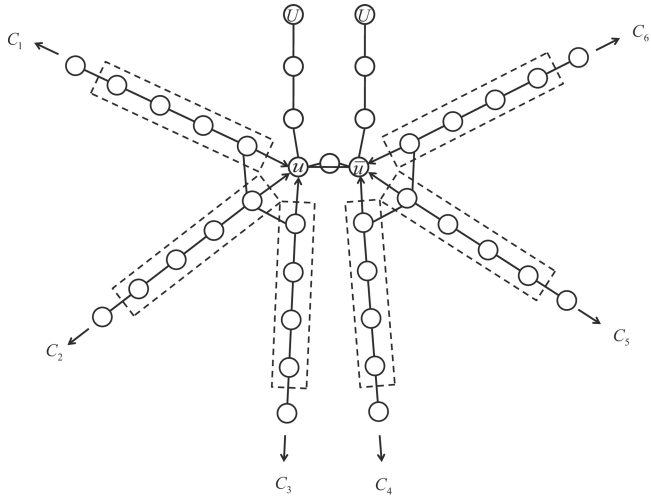


Fig. 4. Connecting multiple clauses (at most 3) to one variable.

Proof. It is trivial to see that MARPCP is in the complexity class NP since given a certificate M, which is a set of nodes, we can verify the cost of M in MARPCP within polynomial time. Furthermore by Theorem A.2, we have a reduction from a NP-complete problem to MARPCP. Therefore, this theorem holds true. □

Appendix B

Theorem B.1. MWDSP-R *is* NP*-complete.*

Proof. It is trivial to see that MWDSP-R is in NP since given a MWDSP-R instance $(G = (V, E), k)$, a certificate $C \subseteq V$ can be verified by checking $\sum_{v \in C} w(v) = k$, where $w(v)$ is the weight on each node v. Now, we show that MWDSP-R is NP-complete by reducing Minimum Dominating Set Problem (MDSP) to MWDSP-R in polynomial time. For simplicity, we assume that there is only one UW-Sink u, which is a sufficient condition to show that this theorem is true for arbitrary number of UW-Sinks.

Suppose G is a UDG induced by a set of sensor nodes V. Let T be the shortest path tree of a UDG induced by $V \cup \{u\}$ rooted at u with height l. Suppose G_i is the subgraph of G induced by all nodes in the level i of T. Now, construct a graph Γ as follows: First, make a copy of graph G, denoted by $G' \leftarrow G$. At last, add a new node u which is adjacent to all the nodes in G_l and G'_l. Note that in Euclidean space, by properly selecting the location of u, it is always possible to construct T so that we can have the u which only adjacent to nodes in G_l and G'_l. Now, we claim that if

$$\bigcup_{1 \leq i \leq l} (D \bigcap V(G_i)) \bigcup (D \bigcap V(G'_i)) \bigcup U$$

is an optimal solution to MWDSP-R, then D is a minimum DS of G, where $U = \{u\}$ or $\emptyset$.

Suppose we have an optimal solution OPT of MWDSP-R. Let $D_i = OPT \bigcap V(G_i), D'_i = OPT \bigcap V(G'_i)$, and $U = OPT \bigcap \{u\}$. Then by the symmetry of the graph Γ (w.r.t. u),

$$w(OPT) = (l+1)|U| + \sum_{i=1}^{l} i|D_i| + \sum_{i=1}^{l} (2l+2-i)|D'_i|, \text{ and}$$

$$w(OPT) = (l+1)|U| + \sum_{i=1}^{l} i|D'_i| + \sum_{i=1}^{l} (2l+2-i)|D_i|.$$

Thus,

$$w(OPT) = (l+1)(|U| + \sum_{i=1}^{l} |D_i| + \sum_{i=1}^{l} |D'_i|),$$

Similarly, if D_1 be any DS of Γ, $D_{(1,i)} = D_1 \bigcap V(G_i)$, and $D'_{(1,i)} = D_1 \bigcap V'(G_i)$, then

$$w(D_1) = (l+1)(|U| + \sum_{i=1}^{l} |D_{(1,i)}| + \sum_{i=1}^{l} |D'_{(1,i)}|).$$

By the minimality of $w(OPT)$, we have

$$|U| + \sum_{i=1}^{l} |D_i| + \sum_{i=1}^{l} |D'_i| \leq |U| + \sum_{i=1}^{l} |D_{(1,i)}| + \sum_{i=1}^{l} |D'_{(1,i)}|.$$

That is, $U \bigcup D \bigcup D'$ is a minimum DS of Γ, where $D = \bigcup_{i=1}^{l} D_i$, $D' = \bigcup_{i=1}^{l} D'_i$.

Next, we show that D is a minimum DS of graph G by contradiction. Suppose D is not minimum and there is another DS C of G with $|C| < |D|$. Without loss of generality, we can assume $D_i = D'_i$, since otherwise we can replace $D_1, D_2, \ldots, D_l$ with $D'_1, D'_2, \ldots, D'_l$ (or conversely). Let $C_i = C \bigcap V(G_i)$ and $C'_i = C \bigcap V(G'_i)$. Then $\{u\} \bigcup C \bigcup C'$ is a DS for Γ. However, we have

$$|\{u\} \bigcup C \bigcup C'| = 2|C| + 1 < 2|D| \leq |U \bigcup D \bigcup D'|,$$

which contradicts the minimality of $U \bigcup D \bigcup D'$. This shows D is a minimum DS of graph G. Since computing a minimum DS is an NP-complete problem, it follows that MWDSP-R is also complete. □

References

[1] I. F. Akyildiz, D. Pompili and T. Melodia, Underwater acoustic sensor networks: research challenges, *J. Ad Hoc Netw.* **3** (2005) 257–279.

[2] J. Y. Yu and P. H. J. Chong, A survey of clustering schemes for mobile ad hoc networks, *IEEE Commun. Surv. Tutorials* **7** (2005) 32–48.

[3] R. Govindan, J. M. Hellerstein, W. Hong, S. Madden, M. Franklin and S. Shenker, The Sensor Network as a Database, USC Technical Report, No. 02-771, September 2002.

[4] P. Wang, C. Li and J. Zheng, Distributed minimum-cost clustering protocol for underwater sensor networks (UWSNs), *IEEE Int. Conf. Commun. (ICC 2007)*, June, (2007), pp. 3510–3515.

[5] E. M. Belding-Royer, Multi-level hierarchies for scalable ad hoc routing, *Wirel. Netw.* **9** (2003) 461–478.

[6] D. Kim, W. Wang, L. Ding, J. Lim, H. Oh and W. Wu, Minimum average routing path clustering problem in multi-hop 2-D underwater sensor networks, to appear in *Optim. Lett.*

[7] D. Dai and C. Yu, A $(5 + \epsilon)$-approximation algorithm for minimum weighted dominating set in unit disk graph, *Theor. Comput. Sci.* **410** (2008) 756–765.

[8] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam and E. Cayirci, Wireless sensor networks: a survey, *Comput. Netw.* **38** (2002) 393–422.

[9] F. Kuhn, T. Moscibroda, T. Nieberg and R. Wattenhofer, Fast deterministic distributed maximal independent set computation on growth-bounded graphs, in *Proc. of the 19th Int. Symposium on Distributed Computing (DISC)*, 2005.

[10] T. Nieberg and J. Hurink, A PTAS for the minimum dominating set problem in unit disk graphs, in *WAOA 2005*, eds. T. Erlebach and G. Persiano, LNCS 3879, (Berlin Heidelberg Springer-Verlag, 2006), pp. 296–306.

[11] D. Lichtenstein, Planar formulae and their uses, *SIAM J. Comput.* **11** (1982) 329–343.