A Distributed Cost Effective Cluster Algorithm

J.Jacqulin Margret., M.E, R.Devipriya.,(M.E),
Dept Of CSE Muthayammal Engineering College,Rasipuram
jacqulin01@gmail.com,devipriyaatrs@gmail.com

Abstract **- Our node clustering has been considered in underwater sensor networks (UWSNs) to improve energy efficiency and prolong network lifetime. Network survivability is a great concern in cluster-based UWSNs. In this paper, we propose a dependable clustering protocol to provide a survivable cluster hierarchy against cluster-head failures in such networks. The proposed clustering protocol attempts to select a primary cluster head and a backup cluster head during clustering so that the cluster members associated with the failed cluster head can quickly switch over to the backup cluster head in the event of a cluster-head failure. Meanwhile, it attempts to select a set of clusters with minimum total cost so that network lifetime can be prolonged to ensure long-term underwater environmental monitoring. Simulation results show that the protocol can effectively enhance network survivability and improve network capacity in the event of cluster-head failures.**

Keywords: **Underwater Sensor Networks,Network lifetime,Residual energy**

I.INTRODUCTION

Energy efficiency is a critical issue in underwater sensor networks (UWSNs) [1-3]. To ensure long-term and continuous underwater environmental monitoring, an underwater sensor node must make efficient use of its limited energy capacity to prolong network lifetime. To improve energy efficiency, node clustering has been widely considered in WSNs [4]. With clustering, each sensor node only needs to send data to its associated cluster head at a short distance while only the cluster heads need to relay the locally aggregated data to a data sink at a long distance, which can significantly reduce the energy consumption of each sensor node and thus prolong the lifetime of the whole network. Moreover, node clustering leads to a hierarchical network architecture, which enables scalable medium access control, robust routing, and coordinate-free localization [5]. In the harsh underwater environment, however, an underwater sensor node is vulnerable to failures or physical damages, which may affect normal network operation. In particular, a single cluster-head failure can result in the loss of connectivity of all affected cluster members and thus disrupt the operation of the whole cluster. For this reason, network survivability becomes a great concern in cluster based UWSNs.

To ensure normal network operation, it is important to provide an effective survivability mechanism to recover the connectivity of all affected cluster members in the event of a cluster-head failure. Traditionally, reclustering can be used to address cluster-head failures. However, such a recovery mechanism is usually time insensitive. In the event of a cluster-head failure, the sensor nodes within the failed cluster will become inactive until the next re-clustering is performed, which would make the sensing coverage incomplete and thus affect normal network operation during the inactive period. To shorten this inactive period, frequent re-clustering is required, which would result in significant control overhead.

In this paper, we propose a dependable clustering protocol to provide a survivable cluster hierarchy against cluster-head failures in UWSNs. For this purpose, the proposed clustering protocol attempts to select a primary cluster head and a backup cluster head for each cluster member during clustering so that the constructed cluster hierarchy can tolerate cluster-head failures. In the event of a cluster-head failure, the cluster members within the failed cluster can quickly switch over to the backup cluster head and thus recover their connectivity to the data sink without waiting for the next re-clustering to be performed. Meanwhile, the protocol attempts to select a set of clusters with minimum total cost so that network lifetime can be prolonged to ensure long-term underwater environmental monitoring. Simulation experiments are conducted to evaluate its performance in terms of network capacity in the event of cluster-head failures.

The rest of this paper is organized as follow. In Section II, we describe the network architecture and review related work. In section III, we present the proposed clustering protocol. In section V, we show simulation results to evaluate its performance. In Section VI, we conclude this paper.

II. NETWORK ARCHITECTURE AND RELATED WORK

A. Network Architecture

A UWSN typically consists of several underwater sinks located at the centers of different monitored areas, a number of ocean bottom sensor nodes surrounding each sink, and a surface station providing a link to an on-shore control center. A sink has sufficient power supply and is capable of handling multiple parallel communications with the sensor nodes. All sensor nodes are homogenous and quasi-stationary. Each of them can adjust its transmission range with transmission power control. Unlike terrestrial sensor networks, a UWSN has some unique characteristics, such as highly limited bandwidth, long propagation delay, harsh geographical environment, and relatively small network scale [1].

B. Energy Model

We use the same energy model as used in [7], which was proposed for underwater acoustic networks. According to the model, to achieve a power level *P*0 at a receiver at a distance *d*, the transmitter power *Etx* (*d*) must be

$$E_{tx}(d)=p_0.d^2 . (10^{(\alpha(f)/10)})^d$$

where α (*f*) , measured in dB/m, is a medium absorption coefficient depending on the frequency range of interest under given water temperature and water salinity.

$$\alpha(f)=0.11((10^{-3}.f^2)/(1+f^2))+44(10^{-3}.f^2)/(4100+f^2))+ 2.7*10^{-7}f^2+3*10^{-6}$$

where *f* is the carrier frequency for transmission measured in *kHz*.

C. Related Work

Node clustering has been widely studied in the context of terrestrial WSNs [5-8] and UWSNs [4]. However, the network survivability problem in the event of cluster-head failures has not been well addressed, in particular, for underwater sensor networks. A traditional solution to this problem is using periodical re-clustering [5-6]. However, reclustering would reduce the timeliness of the sensed data as well as the coverage of the whole network because the sensor nodes in a failed cluster can only be recovered until the next re-clustering is performed. In [10], a robust energyefficient distributed clustering (REED) is proposed for terrestrial WSNs. REED is a HEED-based protocol, which aims to construct a robust clustered architecture by selecting *k* independent sets of cluster heads. However, it is usually effective only for terrestrial WSNs, not for UWSNs, because its effectiveness is based on certain conditions on node density, which are usually satisfied in high-density terrestrial WSNs. In UWSNs, sensor nodes are typically dispersed sparsely. In [1], a fault-tolerant clustering mechanism is proposed to dynamically perform re-clustering locally once most cluster heads reach a consensus about the presence of a failure. However, this mechanism is specifically designed for heterogeneous sensor networks, where cluster heads are powerful gateways which are responsible for fault detection and recovery.

III. DEPENDABLE CLUSTERING PROTOCOL

A. Problem Statement

We consider the dependable clustering problem, which is to construct a cluster hierarchy with each cluster member covered by two different cluster heads. This problem is similar to the domatic partition problem in graph theory [2]. A domatic partition is a partition of vertices in which each part is a dominating set. With the domatic partition, each vertex in the graph is either in the dominating set or has a neighbor in the set. In a clustered network, each sensor node is either a cluster head or a cluster member. Therefore, all cluster heads actually constructs a dominating set. The domatic partition can generate several different dominating sets. Correspondingly, a solution to the domatic partition problem can also partition the network such that each cluster member can be covered by several different sets of cluster heads. The domatic partition problem is a well known NPcomplete problem. A direct solution to this problem is to greedily select small dominating sets and iteratively remove the selected dominating sets from the graph until the remainder is no longer dominating . Based on this greedy algorithm, we propose a dependable clustering protocol to construct a cluster hierarchy with each cluster member covered by two different cluster heads, which is described in the next section.

B. Dependable Clustering Protocol

a. Features

The dependable clustering protocol is based on the greedy algorithm in for solving the domatic partition problem and is built on top of the minimum-cost clustering protocol (MCCP) [4] for achieving energy efficiency. To select two cluster heads for each cluster member, it first selects a set of primary cluster heads (*i.e.*, a dominating set) to cover the whole network. Then it removes the selected cluster heads and selects another set of cluster heads from the remaining sensor nodes as backup cluster heads. The proposed protocol has two major differences from the greedy algorithm. First, it does not need to select a small dominating set or a small number of cluster heads to cover the whole network. Instead, a set of cluster heads with higher energy are selected. The purpose of selecting small dominating sets in the greedy algorithm is to guarantee that multiple different sets of cluster heads can be found so that each cluster member can be associated with multiple backup cluster heads. With this constraint, however, energy efficiency cannot be considered in clustering. For UWSNs, a sensor node is usually designed for extreme environment and has higher reliability.

To reduce computational complexity and control overhead, one backup cluster head is usually acceptable. For this reason, it is unnecessary to select small dominating sets during clustering, which makes it possible to consider energy efficiency without affecting the selection of backup cluster heads. Second, the greedy algorithm needs to find a new dominating set or a set of backup cluster heads which not only cover the remaining cluster members but also cover the primary cluster heads. In contrast, the proposed protocol only requires the selected backup cluster heads to cover the remaining cluster members because the backup cluster heads are only activated when the primary cluster heads fail.

B. Procedures

The protocol procedures can be divided into three phases: initialization phase, clustering phase, and finalization phase.

a) Initialization Phase

Each node could be in one of the following three states: cluster head (denoted by head), cluster member (denoted by memb) and cluster head candidate (denoted by cand). Initially, every node is a cluster-head candidate, and thus is in the cand state. To begin with, each node performs local topology discovery to find out its one-hop neighbors and maintains an uncovered neighbor set, which contains its one-hop neighbors still in the cand state. A candidate can potentially generate a number of different clusters by combining different nodes in the uncovered neighbor set. If a candidate v has an uncovered neighbor set Uv, it can generate 2Uv potential clusters. Among all potential clusters, a candidate selects a cluster as its qualified cluster with minimum average cost

$$\mathrm{cost(A)} = \sum_{\mathrm{u \in B}} \mathrm{C(u,v)f1(Eu)} + \mathrm{C(v,s)f2(A)}\,/\mathrm{f1(Ev)}$$

where *Ev* (or *Eu*) is the residual energy of node *v* (or *u*) normalized by the initial energy of the node and *Eth* (0<*Eth*<1) is a threshold for the residual energy. When the normalized residual energy of a node is larger than this threshold (e.g., *x*≥*Eth*), the node is called in a high energy state. Otherwise, it is in a low energy state (e.g., *x*<*Eth*). *f*[*Eu*(*t*)] and *f*[*Ev*(*t*)] are energy functions of a node *u* and its corresponding cluster-head candidate *v*, respectively.

b) Clustering Phase

If the candidate itself has minimum average cost, it becomes a cluster head and advertises an INVITE message to all the nodes in its qualified cluster to invite them to become its cluster members. If an INVITE message is received by a candidate and the destination of this message is the candidate, the candidate will first change its candidate state to the memb state. The JOIN message will acknowledge the receipt of the INVITE message and at the same time notify the other candidates within the cluster diameter that the candidate has become a cluster member of some cluster head. If no INVITE message is received or some INVITE messages for other nodes are received, the candidate will stay in its cand state and reselect its qualified cluster because some elements in its uncovered neighbor set might have been covered by some cluster heads or have become cluster heads. Meanwhile, after a node becomes a cluster member, it will keep monitoring the INVITE messages for other nodes and add the senders's IDs of these messages into its CH list. The above procedures are performed by all candidates until each of them becomes either a cluster head or a cluster member. At the end, no candidate is left in the network and each cluster member is associated with a primary cluster head, which is contained in the CH list. Meanwhile, the CH list may also contain one or more cluster heads whose coverage areas intersect with that of the primary cluster head.

c) Finalization Phase

The CH list generated in the clustering phase may contain two or more cluster heads, which however is not guaranteed. As a result, each cluster member is covered by a new primary cluster head or becomes a cluster head itself. Thus, a CH list containing at least one element is generated for each cluster member and by combining the two CH lists the final CH list contains at least two cluster heads. For each cluster member, the cluster head in its backup CH list with the minimum distance to the cluster member is selected as the backup cluster head, where the backup CH list is a set excluding the primary cluster head from the CH list.

IV. ALGORITHM

A INITIALISE

```
(a)status level {head,cand,memb}
(b)Begin
  Each node maintains an
  uncovered neighbor set,s still in the cand state
(c) For each potential clusters, a candidate selects a cluster
  as its qualified cluster with minimum average cost
(d) Calculate cost of cluster A
(e) if( residual energy >threshold),
    X←high energy
. Otherwise, it is in a low energy state ,( x<Eth).
```

B CLUSTERING PHASE

```
(a)If((CH←{CH is cluter head})≠ ϕ)
(b)While v receves INVITE(u,$X_u$) do
   CH(v) ←{v}    CH(v)
(c)Join_cluster(cluster_head_ID,NodeID)
(d)   Else cluster_head_msg(NodeID,cost)
(e)Compare avg cluster cost and individual cluster cost
```

C FINALIZATION PHASE

```
(a)if status is cluster member then
(b)for each CH list have two elements
     CH_SET←CH(v) ∪CH_old(v)
(c)Final CH have at least two cluster head
(d)Calculate minimum distance
(e)Backup cluster head←(min distance node)
```

In the above pseudo codes, *CH*(*v*) is the CH list of candidate *v*. *Av* is the qualified cluster of candidate *v*. *Xv* is a set containing the cluster members of *Av* . *head* is a flag indicating a cluster head. cand is a flag indicating a candidate. memb is a flag indicating a cluster member. INVITE(v, Xv) is a message inviting the nodes in set Xv to become the cluster members of candidate v, JOIN(v,u) is a message acknowledging that node v has received the INVITE message sent by candidate u and joined the cluster as a cluster member of candidate u.

C. Failure Detection

Failure detection is a prerequisite for the cluster members to switch over to a cluster with the backup cluster head in the event of the failure of a primary cluster head. Each cluster member can independently detect the failure of its cluster head by checking the heartbeats periodically sent by the cluster head . Due to the channel uncertainty or signal interference in the harsh underwater environment, however, a sensor node may mistakenly detect the failure of a cluster head, which would unnecessarily trigger a fault recovery process and thus consume extra energy. To avoid unnecessary energy consumption, it is important to accurately detect the failure of a cluster head. Moreover, a detection mechanism should not interrupt normal data transmissions.

To support accurate and TDMA compatible failure detection, we employ a high-accuracy detection mechanism proposed specifically for underwater sensor networks [8]. The detection mechanism is based on a TDMA MAC protocol and runs concurrently with normal network operation by periodically performing a distributed detection process at each cluster member. It combines independent failure detection with a distributed agreement protocol to reach an agreement on the fault status of the cluster head so that the detection accuracy can be largely enhanced.

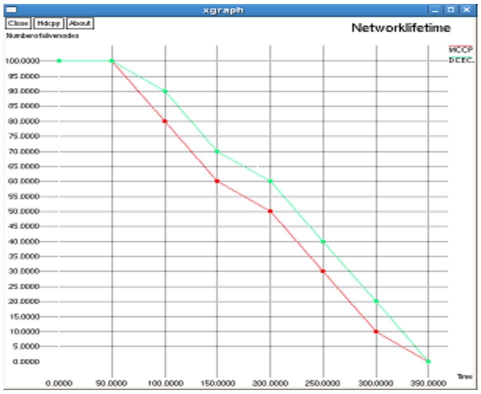

Fig.1. Variation of alive nodes over time period

V. PERFORMANCE EVALUATION

In this section, we evaluate the performance of the proposed clustering protocol through simulation experiments. We used ns-2 to perform the simulation and extended it to include the underwater propagation loss and the physical layer characteristics of underwater transceivers. In the simulation, we considered the sensor nodes uniformly deployed in a 100m×100m sensing region unless otherwise specified. We assume that at the beginning of each round, each node has an initial battery power of *2J*.Figure 1 indicates the variation in number of live nodes over the time period is plotted.

VI. CONCLUSION

In this paper, we proposed a dependable clustering protocol to provide a survivable cluster hierarchy against cluster-head failures in UWSNs. The proposed clustering protocol attempts to select a primary cluster head and a backup cluster head during clustering so that the cluster members associated with the failed cluster head can quickly switch over to the backup cluster head in the event of a cluster-head failure. Meanwhile, it attempts to select a set of clusters with minimum total cost so that network lifetime can be prolonged to ensure long-term underwater environmental monitoring. The simulation results have shown that the protocol can effectively enhance network survivability and improve network capacity in the event of cluster-head failures.

VII. References

[1] I. Akyildiz, D. Pompili, And T. Melodia, ‘‘Underwater Acoustic Sensor Networks: Research Challenges,’’ *Elsevier’s Journal Of Ad Hoc Networks*, Vol. 3, No. 3, Mar. 2005, Pp. 257-281.

[2] J. Heidemann, W. Ye, J. Wills, A. Syed, And Y. Li, ‘‘Research Challenges And Applications For Underwater Sensor Networking,’’ In Proc. Of Ieee Wireless Communications And Networking Conference, Las Vegas, Nv, Apr. 2006.

[3] M. G. Jun-Hong Cui, Jiejun Kong and S. Zhou, ‘‘Challenges: Building scalable mobile underwater wireless sensor networks for aquatic applications,’’ IEEE Network, vol. 20, no.3, May. 2006, pp.12-18.

[4] P. Wang, C. Li, and J. Zheng, ‘‘Distributed minimum-cost clustering protocol for underwater sensor networks (UWSNs)," in Proc. of IEEE ICC’07, Glasgow, Scotland, Jun.2007.

[5] V. Mhatre and C. Rosenberg, ‘‘Homogeneous vs heterogeneous clustered networks: A comparative study,’’ in Proc. of IEEE ICC’04, vol. 6, Paris, France, Jun. 2004, pp. 3646-3651.

[6] W. B. Heinzelman, A. P. Chandrakasan, and H. Balakrishnan, ‘‘An application-specific protocol architecture for wireless microsensor networks,’’ Wireless Communications, IEEE Transactions on, vol. 1, no. 4, Oct. 2002, pp. 660-670.

[7] LIU Pin, Huang Ting-lei, ZHOU Xian-yan, WU Gong-xing, “An Improved Energy Efficient Unequal Clustering Algorithm of wireless sensor network”, IEEE International conference on ontelligent computing and integrated syatems, pp. 930-933, 2010.

[8] Xu Lu , Lianglun Cheng, Ning Yang, “A Data-aggregated Unequal Clustering Routing Protocol for Wireless Sensor Networks” , 2nd international workshop on Intelligent system and applications, pp. 1-4, may 22-23 ,2010.