Merging Grid into Clustering-based Routing Protocol for Wireless Sensor Networks

Ying-Hong Wang, Yu-Wei Lin, Yu-Yu Lin, Hang-Ming Chang
Computer Science and Information Engineering
Tamkang University
Tamsui Taipei, Taiwan, R.O.C.
inhon@mail.tku.edu.tw, harry040@hotmail.com, jerry198926@hotmail.com, chmcicel7473@gmail.com

Abstract- **Wireless sensors have a finite amount of energy and cannot be recharged after deployment. Therefore, the efficiency of the algorithm is an important consideration. An effective algorithm can reduce energy consumption and prolong network lifetime. In this paper, we proposed a cluster-based routing protocol. First, this algorithm divides networks into several units and those units would be regarded as clusters. Second, according to the remaining energy of the nodes, the nodes will execute cluster-head selection. Finally, routing tables are created for routing within clusters and between clusters. We then compare the method we proposed with others by simulation, and the conclusion proves the method we proposed saves more power resulting in a longer network lifetime.**

Keywords - Cluster; Power saving; Routing protocol; WSNs.

I. INTRODUCTION

In recent years, wireless application and wireless communication markets have become more popular due to the rapid development of wireless communications technology. Moreover, the advances in micro technology has led to wide adoption of multiple wireless network technologies consisting mainly of wireless sensor networks [1], Wireless Local Area Network (WLAN), Worldwide Interoperability for Microwave Access (WiMAX), Ad Hoc Networks, Bluetooth Wireless Personal Area Network(WPAN), etc. In wireless sensor networks, micro-manufacturing technology continues to increase capabilities in environmental sensing, information processing, wireless communications, computing ability, and storage capacity. In order to take full advantage of these advances, reducing energy consumption to extend network lifetime of wireless sensors is critical, and thus an important topic for research.

The characteristics of wireless sensor design calls for small footprint, low cost, power saving, and accurate sensing ability. Not only should the hardware experience a breakthrough in growth, but so should the accompanying software The current areas of research can be divided into the following several categories: routing protocol, target tracking, locating, data aggregation, fault tolerance, sensor node deployment and energy management. Each sensor node has data processing, communication, and data sensing responsibilities--all consuming a limited energy resource. In this premise, a wireless sensor node achieves the greatest benefit from an increase in energy consumption efficiency. Therefore, how to design an effective routing protocol is a very important topic. In our proposal, the major issues is studying reduce energy consumption and routing protocol. In the wireless sensor network applications using the environment as a static target and more under consideration, we hope that the node does not have too strong computing power and other additional equipment in order to achieve as much as possible to reduce energy consumption and cost effectiveness. To achieve this goal we propose a cluster- based routing protocol for wireless sensor networks, that is Merging Grid into Clustering-based Routing Protocol (MGCRP) for Wireless Sensor Networks. Through our proposed method we will show improved energy consumption efficiency resulting in extended network lifetime.

The rest of this paper is organized as follows: Section II presents the related work. Section III elaborates the protocol. Simulation results are discussed in Section IV and we conclude our paper in Section V.

II. RELATED WORK

There are numerous papers on using routing protocols to make wireless sensor networks stable, effective and power saving. Al-Karaki and Kamal [2] and Qiangfeng and Manivannan [3] introduced the concept of using routing protocols in wireless networks.

Recently, there are three leading ways of routing. They are chain-based, cluster-based and tree-based; our paper will focus on cluster-based. In cluster-based routing nodes are divided into clusters and the cluster head will send the data collected from normal nodes to sink. Our research consists of two parts: How to effectively cluster nodes, and how to determine the optimal routing path.

Low Energy Adaptive Clustering Hierarchy (LEACH) is proposed by Heinzelman [4]. This routing protocol divides nodes into several clusters by their location, and the nodes can only communicate with in the same cluster.

A special node will be elected as the cluster head. It will collect data from other normal nodes and then send to the sink. Transmission is the source of large energy consumption, so to ensure equal expenditure of energy by the nodes in the network, another cluster head is chosen after the transmission finishes. However, the cluster-head is chosen at random, so it is hard to determine whether the cluster heads are distributed evenly in the network. Also, in this algorithm, distance between the cluster head and the sink is not considered leading to a potential waste of energy if the distance to the sink is exceedingly far from the cluster head which may be further exacerbated if the randomly chosen cluster head belongs to a high node density cluster.

Energy-Balanced Chain-cluster Routing Protocol (EBCRP) [5] is proposed by Bao Xi-Rong. It is a cluster-based distributed algorithm that builds a path of chains through the uses of a ladder algorithm. EBCRP can be divided into three parts, chain-cluster formation, cluster-head selection and steady state. In chain-cluster formation stage, it divides the network into several rectangular blocks and use a ladder algorithm to build a chained path. The next hop in a ladder algorithm is the next increment along the y-axis.

In the cluster head selection stage, a few nodes will be selected to communicate with the sink. In each round, the node closest to the sink with the most residual energy will be the cluster head. In the steady-state stage, the cluster head will collect data and send it to the sink. Each round the cluster head will change to reduce the load of the cluster head. In this algorithm, every node besides the cluster-head transmits data to their neighboring node resulting in a duplication of data communication between 90% of nodes.

III. MERGING GRID INTO CLUSTERING-BASED ROUTING PROTOCOL

Our proposed routing protocol is divided into two phases: Clustering Phase and Routing Phase. We will add Cluster-Head Rotation Mechanism to maintain routing persistence in the Routing Phase. The Clustering Phase starts after the deployment of sensor nodes. In this phase, the sink through the use of the location mechanism, determines the location of each sensor node by using user defined N value of grid length to divide the network into several grids of the same size and, then calculates the each center of grid and the number of nodes within each grid. Moreover, each grid as a cluster, then merge these valid clusters. After clustering the network, a cluster head is chosen for each cluster, and sends it to all cluster heads in network so that each cluster will have information of other cluster head. After all the cluster heads have been selected, the next step is the transmission stage. But before data transfer can proceed, efficient routes needs to be determined. In this routing phase, we initially only used Bellman-Ford shortest path algorithm by D. Bertsekas and R. Gallager [6] to build the inner-clusters and outer-clusters initial routes. However, this operation consumed too much energy during the sensing and transmit stages. To alleviate this issue, during the data transmission stages we added the cluster head rotation mechanism, to avoid overloading any single node.

A. *Network Environment and Assumption*

We assume the wireless sensor network is composed of a sink and a large number of static sensor nodes randomly deployed in the target area.

a. *System Environment*

We assume n sensor nodes randomly distributed in the area to be monitored are continuously sensing and reporting events. These sensor nodes are static. We use S_i to indicate the i-th node, sensor nodes set S = $\{S_1, S_2, \dots, S_n\}$, and the number of S is n. We make the following assumptions about the sensor nodes and the network module.

i) The sink is deployed in a region away from the sensors and we assume that the energy of the sink is infinite.
ii) Sensor nodes will be assigned a unique identifier before deployed in the sensing area.
iii) All nodes have the same computing, storage and energy capabilities.
iv) The sensor node's transmission power can be changed according to the distance from the receiver.
v) All sensor nodes are static. In addition, each sensor node knows its own location and the sink knows their location though the use of the location mechanism.

b. *Energy Consumption Module*

Our paper is using formula from W. B. Heinzelman, A. P. Chandrakasan, and H. Balakrishnan [7] for the communication energy consumption module, and following the formula:

$$E_T(k,d) = E_{Tx}k + E_{amp}(d)k \quad (1)$$

$$E_R(k) = E_{Rx}k \quad (2)$$

$$E_{fuse}(k) = E_{fuse}k \quad (3)$$

Formula (1) means the sensor nodes have an energy cost when transmitting data, (2) means the sensor nodes have an energy cost when receiving data, and (3) means the sensor nodes have an energy cost when the data fuses. In these three functions, where k is data packet size, E_{Tx} is energy cost for transmitting one unit data of the sensor node, E_{Rx} represents the energy cost when node receives one unit data, E_{fuse} is energy cost of the fusing the data. When the sensor nodes transmit amplification is required, so transmitting nodes have an additional $E_{amp}(d)k$ energy cost. The value of $E_{amp}(d)k$ can be determined by formula (4)

$$E_{amp}(d)k = \varepsilon_{FS}d^2 \quad (4)$$

where *d* is the distance between two nodes, ε_{FS} represents the amplified electric power energy cost.

c. *Sensor Node and Cluster Information*

Table 1 shows the sensor node information, which is used to record information about itself. Next we will introduce each field of the table. Node_ID is the identification of the node. Res_Energy is the residual energy of the node. Head_ID is the identification of the cluster head in its own cluster, if the Head_ID and Node_ID are the same, the node itself is the cluster head. Cluster_ID is the cluster number the sensor node belongs to. Next_Hop is the next sensor node to forward data to. Table 2 is the cluster table, it records information of every cluster member and including the following fields Node_ID, Res_Energy, and Cost. Node_ID is the identification of member of the node in the cluster.

TABLE I. SENSOR NODE INFORMATION TABLE

Node_ID	Res_Energy	Head_ID	Cluster_ID	Next_Hop

TABLE II. CLUSTER_TABLE

Node_ID	Res_Energy	Cost

TABLE III. HEAD_LIST

Node_ID	Cluster_ID	Cost

Res_Energy is the residual energy of the node in the cluster. Cost is transmission cost between two nodes. Table 3 is used to record information one neighboring cluster heads. The fields include Node_ID, Cluster_ID and Cost. Node_ID is the identification of the neighboring cluster head. Cluster_ID is the cluster number of the neighboring of cluster head. Cost is transmission cost between two cluster heads.

B. Clustering Phase

a. Clustering

In this paper, MGCRP merge neighboring nodes as far as possible in the same cluster. Before discussing the clustering step, first, we must are define the routing protocol parameters and variables.

- Rectangle Unit Block (Block)**:** This is a rectangular block. The user defined value of N divides the network into several blocks which are the same size and are not overlapping.
- Center of Block (BC)**:** After dividing the network into several grids, we will calculate the center coordinates of the grids resulting in a vector of coordinates. We assume Block_Center i (BC_i) is the i-th center coordinates of the block.
- Cluster**:** Cluster can be regarded as a set C which is includes several sensor nodes. We can represent a set C as $C = \{S_j\}, S_j \in S, j=1,2,\ldots,n$. Where j is the number of sensor nodes. We assume the Cluster_ID is i which represent the cluster number of the grid. In any cluster (C_i), if any member of the sensor nodes are not in a cluster, it is an invalid cluster, otherwise, it is valid cluster.
- Distribution**:** We define a new parameter in a valid cluster Distribution it is used to evaluate the distribution of nodes in a valid cluster. The number of nodes within a valid cluster must be closer to BC. The formula (5) is used to calculate the distribution of the cluster. Where $d(S_m, BC_i)$ is the distance between the member of the sensor nodes in cluster and the BC of the cluster and where $N(C_i)$ is the number of sensor nodes in the cluster.

After defining these parameters and variables, the following details the description of each step.

Step 1: Network Gridding

After the deployment of the sensor nodes, we will make **a grid of the** network. In this paper, we assume that the sensor nodes in the network can be arranged to an M*M area, and assume every length of block is N. The network will be divided into $(\frac{M}{N})^2$ same size blocks. Where the user defined the N value and M value is the length of the sensor network.

Step 2: Calculate Center of Grid

Formula (6) calculates the center of the grid. BC_i is the two-dimensional coordinate vector, where $i=1,2,\ldots,(\frac{M}{N})^2$, this is used to indicate the number of the grid, and also is also the Cluster_ID. The numbering starts from the (0,0) position along the X axis towards the right, Sequenced 1, 2, …, until numbered to the right- border of the sensor network, then back to left-border of the sensor network. In this moment, shift the Y-axis direction one unit block down, then repeat the sequencing step until the grid is complete. Then use the number of grids and formula (6) to get each center of grid.

Step 3: Calculate Distribution (C_i) of Valid Cluster

In this step, we will calculate the Distribution of the valid cluster. First, we give a set VC that includes all valid clusters in the network. N(VC) expresses the number of valid clusters. We will only calculate the Distribution of clusters in the VC set. After each Distribution in each cluster has been calculated, we will start the cluster merging process.

$$Distribution(C_i) = \frac{\left[\sum_{S_m \in S} d(S_m, BC_i)\right]}{N(C_i)} \quad (5)$$

$$BC_i = \left(\left[(i-1)\%\left(\frac{M}{N}\right) + \frac{1}{2}\right] * \frac{M}{N}, \left[\left|(i-1)/\left(\frac{M}{N}\right)\right| + \frac{1}{2}\right] * \frac{M}{N}\right) \quad (6)$$

Step 4: Merge Valid Cluster

First, we choose the fewest number of nodes and the cluster with the largest Distribution value from the VC set. Assume a cluster from the VC set that meets the above conditions is C_A, where $C_A \in VC$, $A=1,2,\ldots,(\frac{M}{N})^2$, then we will start the merge. Let the distance between S_a and S_b be minimal, where $S_a \in C_A$, $S_b \in C_B$, $B=1,2,\ldots,(\frac{M}{N})^2$, $C_B \in VC$ and $C_B \neq C_A$. Then we add all the sensor nodes to C_B from C_A. In other words, let all the Cluster_IDs of the sensor nodes from C_A change to C_B, and remove from the VC C_A, resulting in one less N(VC) .

Step 5: Clustering Finish

Assume the variable K is the user set up number of clusters in the network. The value of K will affect the efficiency of network, so we must decide the variable K according to the network size and number of nodes. The operation of clustering in step 4 will be repeated until N(VC)=K. After clustering finishes, the sink will send related information to the sensor node for an update.

b. Cluster Head Selection

The main task of the cluster head is to fuse data that sensor nodes sensed within a cluster, receive other cluster heads' sensed data, and, send to sink, after clustering finishes and, cluster head must be selected from each cluster. To do so, the sink will broadcast a Head_Elect Message packet to every sensor node in each cluster in the network. When a sensor node gets this packet, it will generate a random variable P between 0 and 1, where *P* is used to differentiate between the same residual energy from other sensor nodes. After sensor node got a random variable *P*, then immediately to calculate itself residual energy. The residual energy is then calculated.

The member nodes of the same cluster compare each of their residual energies according to the transmission power to obtain cost between them. Sensor nodes with the most residual energy will be selected as the cluster head. If more than one sensor nodes have the same residual energy in the same cluster then the sensor node with the larger *P* value will be selected as cluster head. After each of the cluster heads of cluster has been selected, each cluster head will send a Head_Confirm packet to

 ISBN: 978-1-61208-203-5

TABLE IV . HEAD_CONFIRM PACKET

Header	Node_ID	Cluster_ID

the sink. The packet format is shown in Table 4. Each Head_Confirm packet contains three fields, they are Header, Node_ID and Cluster_ID. The Header records the name of packet, Node_ID expresses the Node_ID of the sensor node that is the cluster head, Cluster_ID expresses the Cluster_ID of the cluster to where the cluster head belongs. After the sink receives all the Head_Confirm packets, it will consolidate the information and forward it to each cluster head allowing them to, update their Head_List table.

C. Route

After the clustering stage and the cluster head selection stage, the cluster structure has been established and is complete. The sensor nodes will start sensing and continuously monitor and then through a routing path, start to transmit data. In our paper, the transmission route can be divided into two parts: inner-cluster transmission route and outer-cluster transmission route. Inner-cluster transmission route refers to the path between cluster head and sensor nodes within the same cluster. Outer-cluster transmission route refers to the path between cluster heads.

Our proposed path selection method is mainly based on transmission cost between the sensor nodes. So we use Bellman-Ford shortest path algorithm for the route selection method. We arrange the network as a graph, and assume the sensor nodes in the network are the vertexes of graph and the transmission cost between nodes are edges of the graph. Through the Bellman-Ford algorithm we can calculate the lowest cost of each sensor node to the other.

a. Inner-Cluster Transmission Route Build

Inner-cluster transmission routes are the path between sensor nodes within the same cluster. The sink broadcasts to all the sensors nodes their minimum path cost to the cluster head in their cluster using the Bellman-Ford algorithm according to member cost in the Cluster_Table. In (7), $C(i,j)$ defines the cost between node i and node j, where $P_t(i,j)$ is the transmission power of node i to node j during transmission. After the sensor node receives the minimum cost between the node and the cluster head, it then records the next hop target in the Next_Hop field. When the node wants to transmit data, it sends data to the sensor node based on Next_Hop field. During the transmission, if the sensor node dies or cluster head changes, the Bellman-Ford algorithm is invoked to re-calculate the minimum cost path and the Next_Hop field is updated.

$$C(i,j) = P_t(i,j) \quad (7)$$

b. Outer-Cluster Transmission Route Build

The outer-cluster transmission route and inner-cluster transmission route have the same algorithm, but in the outer-cluster, the send object changes to cluster head to cluster head. The cluster head receives the data that members sent in the cluster and, then it integrates the received data and forwards it to its neighboring cluster head. According to the Cost field of Head_List and through the use of the Bellman-ford algorithm, the cluster head selects the next hop. After the calculation, the cluster head will record the transmission object. If the cluster head has been replaced, then the new cluster head will request a member to re-calculate the minimum cost between cluster heads.

c. *Cluster Head Rotation Mechanism*

The cluster head not only senses the environment but integrates the data from members of the same cluster, and transmits data to other cluster heads. In order to reduce the early death of sensor nodes, we add the cluster head rotation mechanism to distribute the energy consumption. We assume time divided into continuous periods of *T*, in the beginning *T* the sink will send a Cluster_Head Rotation Message to the sensor network. After a normal node receives this message, they will immediately send their residual energy information to the cluster head. Then the cluster head will select the node of with the most residual energy to be the new cluster head. At the same time, the cluster head will broadcast to members within cluster the new identify of the cluster head and update Head_ID of the normal node and send Head_Confirm packet to the sink. The sink will gather all the new cluster head information, consolidate, and send the data to all the cluster heads so they can update Head_List table. During *T*, the sink will repeat the above action to replace the cluster head until the energy of members within cluster is less than the energy defined by the cluster head threshold.

IV. SIMULATION AND ANALYSIS

A. Simulation Environment

This paper uses the Dev C++ simulation environment. The conditions of the sensor network and its related values by W. B. Heinzelman, A. P. Chandrakasan, and H. Balakrishnan [7] are shown on Table 5. Figure 1 shows the relational chart between LEACH, EBCRP and MGCRP, which shows the number of live nodes and number of rounds. Fig.2 is the relational chart between LEACH, EBCRP and MGCRP, which shows the average energy consumption and number of rounds. A round is defined by data that is transmitted to sink safely; a conclusion that is made from the average of 50 kinds of conditions.

B. Simulation Results

We can observe that MGCRP is better than LEACH and EBCRP via Figure 1 and Figure 2. The selection of the cluster head method of LEACH is random, and the clusters transmit collected data directly to the sink. So if there are several clusters which are far away from the sink, the network would die from a large consumption of energy reducing, the number of data transmissions. The selection of cluster head principle of EBCRP is better than LEACH because, it chooses the nodes which are closer to sink to be clusters. The design does not have the transmission distance limitation of clusters in LEACH, but the routing of EBCRP is a chain which is connected by nodes resulting in redundant data transmissions. In this paper, MGCRP combines nodes which are closer to others in a cluster

 ISBN: 978-1-61208-203-5

and when the nodes are distributed unevenly, it shortens the distance between nodes and the cluster head to attain power savings. To not overload any one member, we add cluster head rotation mechanism, to equally to distribute energy consumption to the members in the cluster prolonging the network lifetime.

V. CONCLUSIONS AND FUTURE WORKS

In this paper, we propose a routing protocol which is a clustering-based routing protocol for wireless sensor networks. In this routing protocol, we grid the network and then combine these grids based on a user defined value, There are several advantages of this protocol show below. First, sensors will be allocated by high density in the same cluster no matter what the condition is. Second, we add the cluster head rotation mechanism, and it could allot workload equally to every node. And we choose Bellman-Ford algorithm, spend the minimum of cost to transmit the data to clusters. Through effective clustering, the routing protocol which we proposed could save more energy and prolong the network lifetime.

In future, we hope to add redress mechanisms in the transmission of data stage, because when the sensor nodes may be faced with in the time of passing information to a passing objects have been killed, resulting in the data cannot pass and makes the collection of good data must be discarded. When the sensing data is discarded at the same time also means that before passing the sum of data consumed by the power follow the waste, the data must be retransmitted.

ACKNOWLEDGEMENT

Authors appreciate the funding support for the research project from National Science Council, Taiwan.
Project ID NSC 100-2221-E-032-041-

TABLE V . EXPERIMENTAL PARAMETERS

Parameter	Value
Sensing range (m^2)	**(0,0)~(100,100)**
Sink location	**(50,150)**
Sensor node numbers (n)	**100**
Sensor node initial energy (E0)	**0.5 J**
E_{Tx}, E_{Rx}	**50 nJ/bit**
ε_{FS}	**10 pJ/(bit•m^2)**
E_{fuse}	**5 nJ/(bit•single)**
Data packet size	**4000 bits**
Grid length (N)	**10 m**
Cluster number (K)	**5**

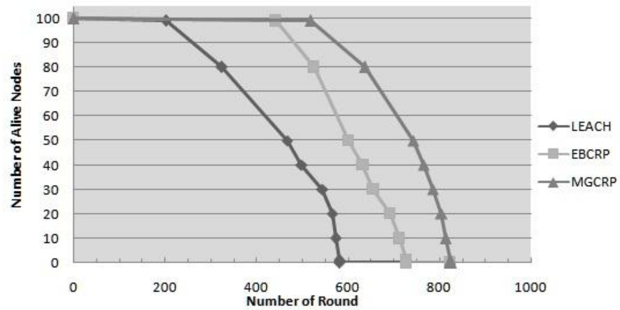

Figure 1. Relation between the number of alive nodes and number of round with different routing protocol

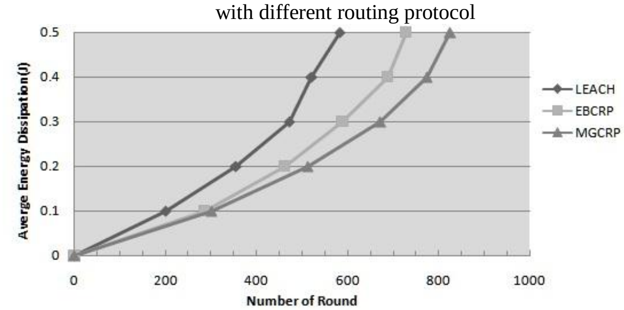

Figure 2. Relation between consumption of average energy and number of round with different routing protocol

REFERENCES

[1] I. F. Akyildiz, S. Weilian, Y. Sankarasubramaniam, and E. Cayirci, "A survey on sensor networks, " *IEEE Communications Magazine*, vol. 40, No 8, pp. 102-114, Aug. 2002.

[2] J. N. Al-Karaki and A. E. Kamal, "Routing techniques in wireless sensor networks: a survey, " *IEEE Wireless Communications*, vol. 11, No 6, pp. 6-28, Dec. 2004.

[3] J. Qiangfeng and D. Manivannan, "Routing protocols for sensor networks," *Proceedings of First IEEE Consumer Communications and Networking Conference*, *2004, CCNC 200* , pp. 93-98, Jan. 2004.

[4] W. R. Heinzelman, A. Chandrakasan, and H. Balakrishnan, "Energy-efficient communication protocol for wireless microsensor networks," *Proceedings of the 33rd Annual Hawaii International Conference on System Sciences*, pp. 1-10 vol. 2, Jan. 2000.

[5] X.-r. Bao, S. Zhang, D.-y. Xue, and Z.-t. Qie, "An Energy-Balanced Chain-Cluster Routing Protocol for Wireless Sensor Networks," *Proceedings of the 2010 2th International Conference on Networks Security Wireless Communications and Trusted Computing (NSWCTC)*, pp. 79-84, Apr. 2010.

[6] D. Bertsekas and R. Gallager, *Data Networks*, 2nd ed. Englewood Cliffs, New Jersey: Prentice–Hall, 1992.

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

 ISBN: 978-1-61208-203-5