Autonomous Clustering and Message Passing Protocol for Energy Efficiency in Wireless Sensor Networks

JENHUI CHEN
Department of Computer Science and Information Engineering,
Chang Gung University, Kweishan, Taoyuan, Taiwan 333, R.O.C.
Email: jhchen@mail.cgu.edu.tw
CHIEN-CHUN JOE CHOU
Sonics, Inc., Mountain View, CA, USA.
Email: joechou@sonicsinc.com

Received: July 15 2005; revised: October 30 2005

***Abstract*— Wireless sensor networks consist of a large number of nodes with limited battery power and sensing components, which can be used for sensing specified events and gather wanted or interesting information via wireless links. It will enable the reliable monitoring of a variety of environments for both civil and military applications. There is a need of energy-efficient message collection and power management methods to prolong the lifetime of the sensor network. Many methods, such as clustering algorithm, are investigated for power saving reason, however, they only consider reducing the amount of message deliveries by clustering but not the load balance of the clusters to extend the maximum lifetime of the network. Therefore, in this paper, we propose a fully distributed, randomized, and adaptable clustering mechanism named autonomous clustering and message passing (ACMP) protocol for improving energy efficiency in wireless sensor networks. Sensor nodes, according to ACMP, can cluster themselves autonomously by their remaining energy and dynamically choose a corresponding cluster head (CH) to transfer the collected information. Sensor nodes adjust an appropriate power level to form clusters and use minimum energy to exchange messages. The network topology is changed dynamically depending on the CH's energy. Moreover, by maintaining the remaining energy of each node, the traffic load is distributed to all nodes and thus prolong the network lifetime efficiently. Simulation results show that ACMP can achieve a highly energy saving effect as well as prolong the network lifetime.**

***Index Terms*— autonomous, cluster, distribution, wireless sensor networks.**

I. INTRODUCTION

A wireless sensor network consists of a lot of inexpensive, lower-power, and tiny sensor nodes, which has a wide range of potential applications including environment monitoring, target tracking, security, medical systems, health care, and robotic exploration, etc [5], [9], [13]. These sensor nodes can self-organize to form a network and communicate with each other by wireless interface. These nodes are usually unreliable and inaccurate, but their size and cost enable applications to network hundreds or thousands of these tiny sensor nodes in order to achieve high quality, fault-tolerant sensing systems. Each node has one or more sensors, embedded processors and low-power radios, and is normally battery operated.

Because of the energy restriction of sensor nodes it needs an energy-efficient communication protocol for battery power saving so that the network's lifetime is prolonged. One major task of these sensor nodes is to gather wanted information and send them back to a coordinator called *sink* node for analyzing and monitoring specific matters. This action will consume a lot of energy if there is no efficient communication protocol. One potential solution of saving battery consumption is to reduce the number of messages to the sink node. One simple way to reduce the number of messages is to divide all sensor nodes into several clusters and gathers the information from nodes by cluster head. After accumulating a reasonable amount of messages, cluster heads transfer the aggregated information to the sink node in order to reduce the energy consumption [22]. Fig. 1 depicts an application where sensor nodes periodically transmit information to a remote observer (e.g., a sink node). It shows that the communication overheads can be reduced by separating sensor nodes into several clusters.

Many clustering algorithms have been investigated and proposed in recent years [1], [2], [4], [7], [8], [11], [12], [19], [20]. The Span [4] and geographic adaptive fidelity (GAF) [19] algorithms are geographic topology based clustering protocols that utilize location information to eliminate unnecessary links. However, they may not be feasible since the position of each node is often not provided in practice. The low-energy adaptive clustering hierarchy (LEACH) [11] utilizes randomized rotation of clusterheads (CHs) to evenly distribute the energy load among sensor nodes in the network. In fact, the rotation of CHs is not necessary and may waste more energy if there are few events in some areas. The clustering-based maximum lifetime data aggregation (CMLDA) [8] scheme is a data collection

This work was supported in part by the National Science Council, Taiwan, R.O.C., under contract NSC94-2213-E-182-006.

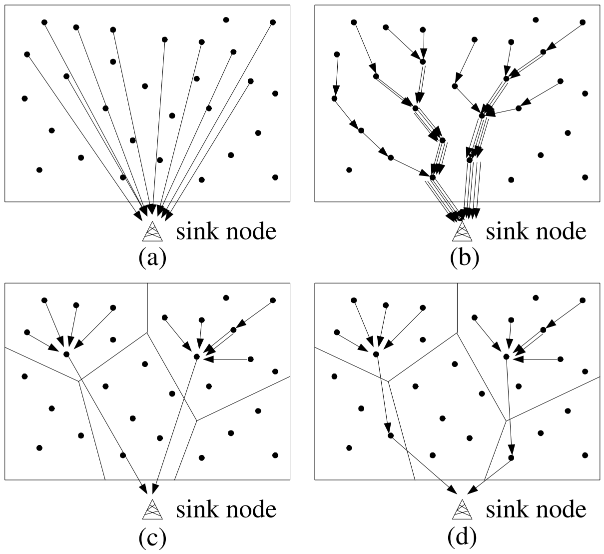

Fig. 1. An illustration of information gathering with and without clustering: (a) Single hop without clustering; (b) Multihop without clustering; (c) Single hop with clustering; and (d) Multihop with clustering.

algorithm that focuses on how to find an efficient manner in which the data should be collected from all sensor nodes and transferred to the sink node, such that the network lifetime is maximized. Nevertheless, CMLDA does not considers the total energy usage and thus not achieve the global solutions. The Max-Min D-Cluster algorithm [1] generates d-hop clusters with a run-time of $O(d)$ rounds. Unfortunately, this algorithm does not ensure that the energy used in communicating information to the sink node is minimized.

In [2] and [7], the authors propose a distributed algorithm for organizing sensors into a hierarchy of clusters with the objective of minimizing the total energy spent on communications of information gathered by sensors to the sink node. However, they do not consider the network lifetime, which is defined as the time from nodes deployment to the time when the first node is run out of function due to energy depletion. The energy consumption is defined as the total energy consumed by all nodes in the sensor network during whole data processing procedures. In [12], authors propose a dynamic cluster-based structure to track movement of boundaries and facilitate the fusion and dissemination of boundary information in a sensor network. It is suitable for tracking special events like fire but is not for tracking one or more individual objects, such as people, animals, and vehicles.

The reactive clustering algorithm decentralized reactive clustering (DRC) [20] protocol where the clustering procedure is initiated only when events are detected. It uses power control technique to minimize energy usage in formatting clusters. Initially, all sensor nodes enter sleeping mode in order to save energy. The cluster forming phase is launched only when events occur. After data aggregation, all nodes will enter sleeping mode again. However, if events occur frequently, the cluster forming phase will cause a lot of overheads and the energy of the CH may run out even rapidly. It may not be suitable for large-scale environment.

To avoid above mentioned drawbacks, this paper proposes an autonomous clustering and message passing (ACMP) protocol for wireless sensor networks. The ACMP has five unique characteristics:

- ACMP is a fully distributed and autonomous sensor communication protocol.
- Each node can join C clusters at most simultaneously.
- The cluster topology is changed dynamically depending on the remaining energy of CHs.
- The load balance of each cluster is considered in this scheme.
- ACMP supports local re-clustering.

By adopting ACMP, sensor nodes will dynamically decide to pass data to a CH which has more remaining energy. The built clusters will be disbanded and rebuilt automatically if the remaining energy of the CHs are low.

The remainder of this paper is organized as follows. Section II describes the system model of ACMP. Section III discribes the detail of ACMP. In Section IV, shows ACMP effectiveness via simulations and compares it to other clustering techniques. we perform a series of simulations to evaluate the performance of ACMP. Finally, we give some conclusions in Section V.

II. System Model

ACMP uses carrier sense multiple access (CSMA) medium access control (MAC) protocol to form clusters. ACMP is a cluster-based protocol. Therefore, once clusters are created, CHs will coordinate all messages from their members. Each CH, moreover, creates a time schedule based on time division multiple access (TDMA) protocol [6], [14], [17] to tell its members when they will wake up to transmit or receive. The radio of each member can be turned off until the node's allocated transmission time and thus minimize its energy consumption. The CH must keep its receiver on to receive data from its members. When all the data has been received, the CH can compress the data into a single message and transmit the aggregated information to the sink node.

A typical sensor node consists mainly of a sensing circuit for signal conditioning and conversion, a digital signal processor, and radio links [3], [10], [15]. The energy consumption model [11], [16] for each sensor are given as below.

A. Communication Energy Dissipation

The key energy parameters for communication in this model are the energy/bit consumed by the transmitter electronics (α_t), energy dissipated in the transmit op-amp (α_a), and energy/bit consumed by the receiver electronics (α_r). Taking Fig. 2, assume a d^2 energy loss due to channel transmission. Thus, to transmit a r-bit message a distance d using the radio model, the radio expends:

- $E_{Tx}(r,d) = \alpha_t r + \alpha_a r d^2$, where E_{Tx} is the energy consumed to send a r-bit message.
- $E_{Rx}(r,d) = \alpha_r r$, where E_{Rx} is the energy consumed to receive a r-bit message.
- α_t, α_r, energy dissipated in transmitter and receiver electronics per bit (Taken to be 50 nJ/bit).

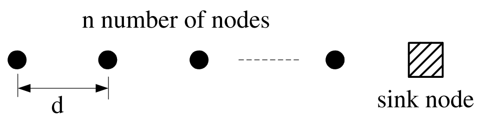

Fig. 2. A simple linear network.

Type	Destination	Source	CH	Remaining Energy	TTL	Hop_Count

Fig. 3. The ACMP control message format.

- α_a, energy dissipated in transmitter amplifier (Taken to be 100 pJ/bit/m^2).
- r, number of bits in the message.
- d, message transmission distance.

We make the assumption that the radio channel is symmetric such that the energy required to transmit a message from node A to node B is the same as the energy required to transmit a message from node B to node A for a given signal-to-noise ratio (SNR).

B. Computation Energy Dissipation

We assume the current leakage model of [15], [18]. The model depends on the total capacitance switched and the number of cycles in the program.

III. Autonomous Clustering and Message Passing

Consider a sensor network consisting of hundreds or thousands of sensor devices, which are fairly distributed in an area, with a same hardware specification. Each sensor node has k kinds of power level $E = \{e_1, e_2, \ldots, e_k\}$ and its corresponding transmission distances are $d_1, d_2, \ldots, d_k$. Assume $e_1 < e_2 < \ldots < e_k$ then we have $d_1 < d_2 < \ldots < d_k$. Taking Fig. 2, in minimum-transmission-energy (MTE), each node sends a message to the closest node on the way to sink node. The node located at distance nd from sink node would require n transmits a distance r and $n-1$ receives. From the literature [11], it shows that the direct communication to sink node requires less total energy than MTE routing protocol if:

$$\frac{\alpha_t}{\alpha_a} > \frac{d^2 n}{2}. \tag{1}$$

According to this criterion, ACMP can choose a minimum energy consumption route to the CH to form the cluster. In the following, we will describe ACMP in detail.

A. Clustering and Power Control

Assume each node has a same probability P to become a CH in the network. Initially, each node decides itself whether to serve as a CH or not according to P. A node will advertise a control message via broadcast to its neighbors within its radio range with lowest power e_1 once it becomes a CH. The control message format is shown in Fig. 3 and described below:

- The "TYPE" field indicates the type of the message which represents REQUEST, REPLY, REFRESH, or RESET, respectively. The REQUEST message is sent by an unclustered node which wishes to form a cluster; the REPLY message is used to reply to the REQUEST; the REFRESH message is used by CHs to announce its members to update their CH table. The RESET message is sent by CHs to announce its members to exit their cluster and delete the records of CHs that announce the message.
- The "Destination" is the destination address.
- The "Source" is the sender's address.
- The "CH" field indicates the message belongs to which CH.
- The "Remaining Energy" is the remaining energy of CH.

INPUT: P, E
BEGIN
Generates a random probability p_i
IF $p_i \geq P$ **THEN**
  Takes $B(E_r)$ for the backoff countdown
  IF no REQUEST from other nodes during $B(E_r)$ **THEN**
    Node i becomes the CH and sends REQUEST with power e_1
  ELSE
    Node i becomes a cluster member
    IF REQUEST's TTL > 1 **THEN**
      Forward the REQUEST
ELSE
  Waits for a specified time duration
  IF receives a REQUEST during the duration **THEN**
    Node i becomes a cluster member
    IF REQUEST's TTL > 1 **THEN**
      Forward the REQUEST
FOREVER until becomes a CH or cluster member
END

Fig. 4. The clustering algorithm.

A node will send a REQUEST message if it wants to form a cluster. Before sending the REQUEST message, nodes should execute a backoff procedure to avoid more than one node sending this message at the same time. Let $B(E_r)$ be the backoff function and represented as

$$B(E_r) = \frac{2m}{E_r} \tag{2}$$

where m is the number of sender's neighbors and E_r is the remaining energy of sender. This strategy is to ensure that the node with a higher remaining energy will become a CH first.

Nodes which receive the REQUEST message will become a cluster member automatically and check the time-to-life (TTL) field to determine whether forward this message or not. When the TTL is bigger than 1, it subtracts 1 from TTL and forwards this message with minimum transmit energy e_1 via broadcast to its neighbors. The forwarding process will be terminated until the value of TTL reaches 1. The Hop_count field will be increased by 1 when forward is performed. This field is provided for sensor nodes to estimate themself how far they are from the CH. We note that each node can join C clusters at most simultaneously. A detailed description of the clustering algorithm is shown in Fig. 4.

Fig. 5 illustrates an example of the cluster forming process. Initially, node A wishes to be a CH and broadcasts a REQUEST message to its 1-hop neighbors with power level e_1 and TTL = 2. If it's 2-hop neighbors do not join any cluster or the number of joined clusters less than C, it will join the cluster. Assume node A first finishes its backoff countdown

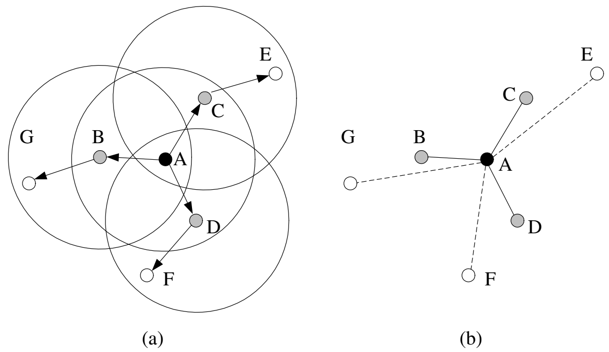

Fig. 5. An illustration of initially clustering process: (a) node A uses e_1 to broadcast the REQUEST message to form the cluster; (b) the formed cluster where solid and dished lines represent communication powers of e_1 and e_2, respectively.

and broadcasts the REQUEST message to form the cluster. Nodes B, C, and D, after receiving the REQUEST message, will become the cluster members and check the TTL whether is less than 1. In this case, TTL equals 2, nodes B, C, and D will forward the REQUEST with TTL = 1 and Hop_count = 2. Meanwhile, nodes B, C, and D will estimate an appropriate power level to connect to node A according to (1) and send REPLY messages for joining the cluster.

This process will be performed continually until TTL reaches 1. As a result, nodes E, F, and G join the cluster after receiving the forwarded REQUEST message. In this example, we assume the transmit range of e_1 and e_2 are $d_1 = 10$m and $d_2 = 20$m, respectively. After forming the cluster, nodes B, C, and D use e_1 and nodes E, F, and G use e_2 to connect to the CH. The result of clustering is shown in Fig. 5(b).

Every node should maintain a CH table, which records the joined CH's address, the related transmit power level to the CH, and the remaining energy of the CH. The CH maintains a participation table records the information of the participating nodes and the transmit power levels to its members.

B. The analysis model

Assume all sensor nodes are distributed uniformly in a Lm$\times$ Lm square area and the diameter of the cluster is represented as h-hop. The total energy consumption of sensor networks is the energy consumed by all member nodes sending data to their CHs and all CHs sending aggregate data to the sink node. Let N, N_C, and N_M represent the total number of sensor nodes, CHs, and members in a sensor network, respectively, and $N = N_C + N_M$. The $N_M(i)$ is denoted as the number of members within the i-hop distance from CHs.

Let Δ_1 be the total energy spent by all sensors communicating r bits of data to their respective CHs. Denote N_M^{ij} being the number of members, which is i hops distance from its corresponding CH, belong to the CH j. Thus, the total energy spent by all sensors is given by

$$\Delta_1 = \sum_{i=1}^{h}\sum_{j=1}^{N_C} N_M^{ij}\big(E_{Tx}(r, d_i) + E_{Rx}(r, d_i)\big), \qquad (3)$$

where d_i represents the distance of i hops to the CH. Since sensor nodes are distributed uniformly in the Lm $\times$ Lm square area and the sink node is placed in the center of the area, then the average distance $\overline{D}$ from CHs to sink node will be

$$\begin{aligned}\overline{D} &= \sum_{x=0}^{L}\sum_{y=0}^{L} P_{xy} D_{xy} \\ &= \sum_{x=0}^{L}\sum_{y=0}^{L} \frac{1}{(L+1)^2}\sqrt{(\frac{L}{2}-x)^2 + (\frac{L}{2}-y)^2},\end{aligned} \qquad (4)$$

where P_{xy} is the probability of CHs distributed in the location (x, y), and D_{xy} is the distance from CHs at (x, y) to the sink node.

Let Δ_2 be the total energy spent by all CHs communicating r bits of data to the sink node. From (1), the energy consumption can benefit from direct transmission than multihop transmission if the transmission distance d satisfies

$$d < \sqrt{\frac{2\alpha_t}{n\alpha_a}}. \qquad (5)$$

Since each transmission includes the energy consumptions of both the transmitter and receiver, thus the more intermediate nodes is involved in forwarding the more energy is consumed. Therefore, the minimum energy consumption is achieved when $n = 2$ in a given distance. Thus the maximum transmission distance $d_{\max}$ of each CH that will benefit from using direct communication than MTE routing protocol if and only if $d_{\max} \leqslant nd$. From (5), we have

$$d_{\max} \leqslant \sqrt{\frac{2n\alpha_t}{\alpha_a}}. \qquad (6)$$

For example, in the energy consumption model [11], [16], if $n = 2$, $\alpha_t = 50$ nJ/bit, and $\alpha_a = 100$ pJ/bit/m^2, the $d_{\max} \approx 44.72$m.

In the worst-case, CHs use $d_{\max}$ transmission range to transmit their aggregate data to sink by one hop or multiple hops. Thus, the average number of hops $\overline{h}_s$ from CHs to the sink node is equal to $\overline{D}/d_{\max}$. The total energy spent by all CHs communicating r bits of data to the sink node can be obtained by

$$\Delta_2 = N_C\Big[E_{Tx}(r, d_{\max})\overline{h}_s + E_{Rx}(r, d_{\max})(\overline{h}_s - 1)\Big]. \qquad (7)$$

Assume each CH's cover area can be divided into h concentric circles and the width of each section is d_1. Then the area of the i-th concentric circle A_i can be calculated as $(d_i)^2\pi$. Thus the area of the i-th section denoted as S_i is given by

$$\begin{aligned}S_i &= A_i - A_{i-1} \\ &= d_i^2\pi - d_{i-1}^2\pi \\ &= i^2\pi d_1^2 - (i-1)^2\pi d_1^2 \\ &= (2i-1)\pi d_1^2.\end{aligned} \qquad (8)$$

For example, the area of $S_2 = (2\times 2-1)\pi d_1^2 = 3\pi d_1^2$ and $S_3 = 5\pi d_1^2$. Thus, the ratio of the number of nodes in the i-th section to the overall number of nodes in the h-hop cluster is equal to $S_i/\pi(hd_1)^2 = (2i-1)/h^2$. From (3) and (7), the

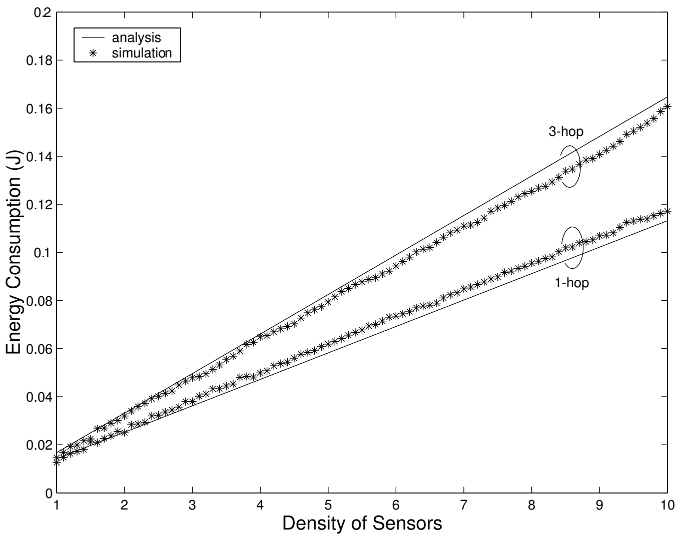

Fig. 6. The results of analysis and simulation of ACMP when $h = 1$ and $h = 3$.

total energy consumption of the overall network will be

$$\begin{aligned}\Delta_1 + \Delta_2 &= \sum_{i=1}^{h}\sum_{j=1}^{N_C} N_M^{ij}\big(E_{Tx}(r,d_i) + E_{Rx}(r,d_i)\big)\\ &\quad + N_C\Big[E_{Tx}(r,d_{\max})\overline{h}_s + E_{Rx}(r,d_{\max})(\overline{h}_s - 1)\Big]\\ &= \sum_{i=1}^{h}\Big[(N-N_C)(\frac{2i-1}{h^2})(E_{Tx}(r,d_i)+E_{Rx}(r,d_i))\Big]\\ &\quad + N_C\Big[E_{Tx}(r,d_{\max})\overline{h}_s + E_{Rx}(r,d_{\max})(\overline{h}_s - 1)\Big].\end{aligned} \tag{9}$$

Now we have to solve the value of N_C. Assume the radius of a cluster is h-hop and denoted as d_h ($d_h = hd_1$), the minimum number of CHs $N_{C,\min}$ that can cover a Lm $\times$ Lm square area can be calculated by

$$N_{C,\min} = \frac{L^2}{\pi {d_h}^2} = \frac{L^2}{\pi h {d_1}^2}. \tag{10}$$

Fig. 6 shows the analysis and simulation results of ACMP in detail. All nodes are distributed uniformly in 100m × 100m where $h = 1$ and $h = 3$. We vary the density of sensors from 1 to 10 (100 to 1000 nodes) to investigate the energy consumptions in the result of (9) and ACMP. Assume every node transmit 1 bit of data to their CHs and after all CHs aggregate all of the data from member nodes, they send the data to the sink node. We can see that the simulation results are close to our analysis.

C. Load Balance of CHs

Clustering enables the network scalability to large number of sensors, reduce the communication overhead and extends the network life. CHs are responsible for coordination among the nodes within their clusters and collection of data information (inter-cluster communication) and sent these data to the sink node. However, the CH energy will run out quickly if many events occur frequently in its dominated area or it has to coordinate many members in its cluster. We, therefore, propose a load distribution mechanism for load balance of CHs in ACMP. Assume each CH has j thresholds of remaining energy $R = \{r_1, r_2, \ldots, r_j\}$ and $r_1 > r_2 > \ldots > r_j$.

When the remaining energy of the CH is less than one of thresholds, it will select the maximum transmit power level recorded in the participation table to broadcast the REFRESH message to announce all of its members. The member nodes will update their CH table after they receive the REFRESH message. Each cluster member chooses a CH with a maximum remaining energy according to the CH table to report data. If there are more than one candidate CHs, i.e., their remaining energy are equal, the cluster members choose the closest CH to report data.

D. Local Re-clustering

When the remaining energy of CHs reaches the lowest energy threshold r_j, it will choose the maximum transmit power level recorded in the participation table to broadcast the RESET message to all of it cluster members. Each member will join another cluster immediately after receives the message. If nodes do not have any alternative CH for join and its remaining energy is higher than r_j, it will form a new cluster with probability P. Otherwise, it will serve as a slaver and join its previous CH again. After local re-clustering, every cluster member will update its CH table.

E. An Example

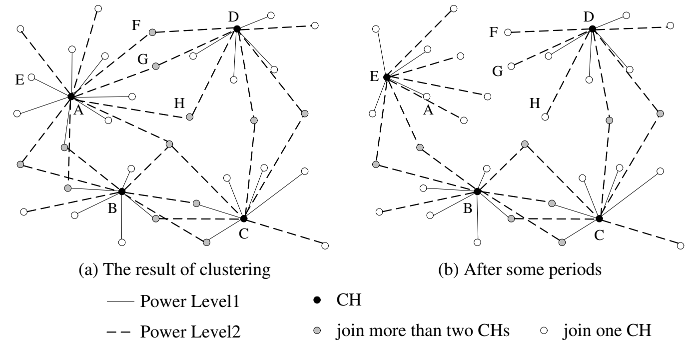

Fig. 7. An example of reclustering process of ACMP where cluster size is 20m, $h_1 = 10$m and $h_2 = 20$m. (a) The initial clustering topology. (b) The result of the local reclustering by node A.

Fig. 7 illustrates an example of clustering process in a sensor network by using ACMP. Initially, all nodes do not join any cluster and will form clusters with a given probability P to become a CH in the network. Assume each node can join 2 clusters at most simultaneously, and the maximum number of hops of each cluster is 2 hops. Fig. 7(a) shows the initial result of clustering, nodes A, B, C, and D become CHs, each member uses appropriate power level to communicates with their CHs according to the Hop_count. After some periods, A's remaining energy lower than r_1, it then broadcasts a RESET message to reset the cluster. After performing this recluster process, the topology is reorganized and shown in Fig. 7(b). This result shows that ACMP can reorganize the network to prolong the network lifetime automatically.

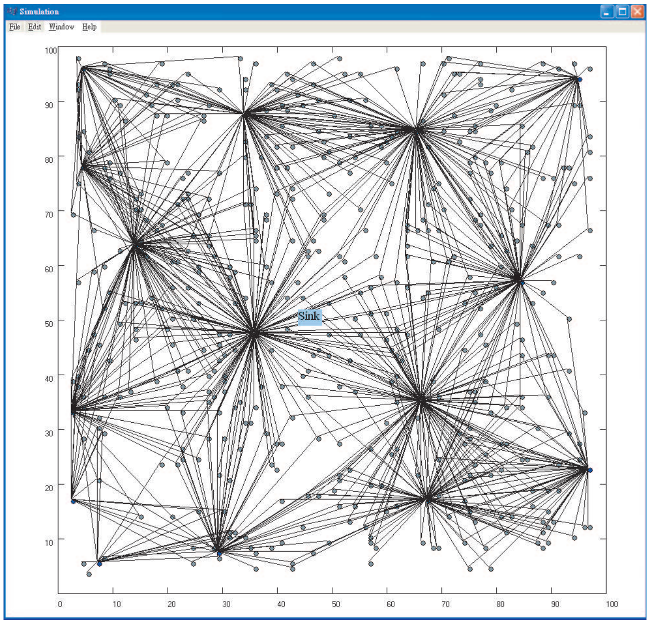

Fig. 8. The snapshot of the simulated sensor network topology when $N =$ 500 and TTL = 3.

IV. Simulation Model and Results

In the simulation model, different numbers of sensor nodes N are uniformly distributed in a 100m×100m square, which are represented as different network densities (D_e). Table I shows the six network densities considered in our simulation. For example, $D_e = 5$ represents 500 sensor nodes, which are uniformly distributed in the network. The size of each cluster is measured by d_1 (the distance of minimum transmission power) and represented as hops (TTL). Notice that the term, for example, “ACMP with 3-hop” implies that the radius of the cluster is 3 hops long and all members within this cluster will use “one hop” to transmit messages to the CH, i.e., direct transmission. Each sensor node can join different number of clusters C simultaneously. This implies that a sensor node can join C clusters around its neighboring nodes at most if any. Fig. 8 shows an example of the network topology performed in the simulations when $N = 500$, TTL $= 3$, and $C = 3$. The sink node is placed in the center of the square to collect all information from sensor nodes in the network.

TABLE I

System Parameters in Simulations

Number of Sensors (N)	Density (D_e)	Probability (P)	Maximum Number of Hops (TTL)
500	5	0.1012	5
1000	10	0.0792	4
1500	15	0.0688	3
2000	20	0.0622	3
2500	25	0.0576	3
3000	30	0.0541	3

The event occurring model in all simulations is generated by a given probability P_e. The simulated area is divided into many 2m×2m squares and each of them has the same event occurring probability P_e. The event is triggered every one round and the round is defined as a specific time unit. Once an event occurs, the sensor nodes around the event will generate messages to transfer to the corresponding CHs. Each sensor node has ten kinds of power levels e_1, e_2,..., and e_{10}, and their corresponding transmission distances are 10m, 20m,..., and 100m, respectively. The sensing range of each sensor is set 2m long and the initial energy of each node is one Joule. There are nine threshold levels of remaining energy $R = \{9/10, 8/10, \ldots, 1/10\}$. When the remaining energy of the CH reaches any threshold of R, the CH will announce the status to its members. The member after receiving this message will choose a CH, which has a largest remaining energy among its CHs, for passing sensed data if any. The CH performs the re-clustering procedure locally only when the remaining energy reaches $1/10$.

In simulations, ACMP is compared with the energy efficient hierarchical clustering (EEHC) algorithm [7] and LEACH [11] scheme, to evaluate the performance of power consumption and network lifetime. We refer to the optimal energy minimization parameters of the EEHC algorithm in [7], which is shown in Table I. For comparison, ACMP adopts the same parameters as in EEHC, excepting the parameters TTL $= 3$ and TTL $= 1$. The data length of each sensed information is represented as 2000 bits long and reports to the CH per each event. Each simulation run lasts 50,000 rounds and each simulation result is obtained by averaging the results from ten independent simulation runs.

A. Simulation Results

In the following experiments, we investigate two major metrics as the performance of the protocols:

- Network Lifetime: The time from nodes deployment to the time when the first node is run out of function due to energy depletion. It is measured in rounds.
- Energy Consumption: The total energy consumed by all nodes in the sensor network during the whole data processing procedure.

The first experiment evaluates the network lifetime of ACMP by varying the parameter C. The experiment is terminated immediately when any node runs out of its energy. From the experiment results, shown in Fig. 9, the network lifetime will be longer than that members can only join one CH. This is because that ACMP can benefit from two main mechanisms: (i) autonomous clustering (dynamic load balance of CHs) and (ii) autonomous message passing, and, hence, extends the network lifetime efficiently. The first mechanism enables CHs to re-cluster itself automatically as their remaining energy is lower than each threshold of R. This mechanism can prevent the CH from running out of its energy quickly by taking turns to be the CH with its neighboring nodes. The second mechanism is a dynamic load balance scheme to alleviate the message forwarding load of CHs. This mechanism is achieved by each sensor node dynamically choosing one of its neighboring CHs, which has the largest remaining energy, to pass the message if the sensor node can join more than one CH simultaneously.

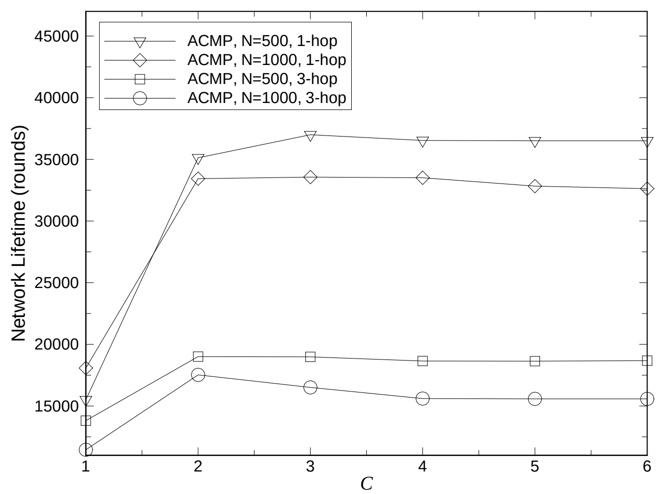

Fig. 9. The network lifetime vs. Max. joinable CHs (C) when $P_e = 0.1$.

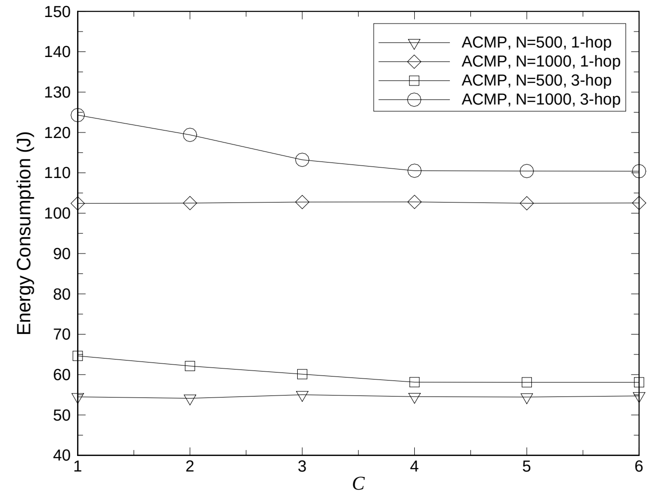

Fig. 10. The energy consumption vs. Max. joinable CHs (C) when $P_e = 0.1$ and the simulation time is 15000 rounds.

Fig. 9 shows that the network lifetime of $C = 1$ is lower than that can join more than one CH both in 1-hop (TTL $= 1$) and 3-hop cases. From this result, we can know that the network lifetime can be prolonged efficiently when the sensor nodes can choose more than one CH. Moreover, from the results, we can see another interesting remarkable results that the network lifetime will be shorter when the cluster size is larger. This is because that the CHs' energy will be run out rapidly due to the cluster size is larger, the CHs must coordinate more sensors as well as lead to more overheads of the CHs. This strategy will waste more energies and degrade the network lifetime. One the other hand, when the cluster size is smaller, there are fewer sensors in the clusters and the overheads of the CHs is lower. Thereby if the cluster size is smaller, the network lifetime is longer than the cluster size is larger.

Following above experiment, Fig. 10 shows the energy consumption of the sensor nodes under different C. We can see that the energy consumption of $N = 1000$ is higher than that of $N = 500$. But the energy consumption of 1-hop cases ($N = 500$ and $N = 1000$) in different C are quite equal. This is because that no matter how the sensor node chooses the CH for passing messages, the energy consumption is same since the sensor node sends messages to the CH with power level e_1. However, in the case of 3-hop, the total energy consumption will decrease when the C increases. Under the case of $C = 1$, if the distance between the member and the CH is far, it has to use higher power level to send messages. This will cost a lot of energy consumptions. On the contrary, when $C > 1$, the members can send message to their CHs according to the remaining energies of CHs alternately. If the remaining energies of CHs are equal, the member will randomly choose one CH to transmit. Because the traffic load is dispersed to each sensor nodes, the network lifetime is prolonged efficiently.

In Fig. 11 and Fig. 12, we vary the D_e from 5 to 30 to investigate the energy consumption and network lifetime of ACMP, EEHC, and LEACH. The event occurring probability is set as $P_e = 0.1$. We observe that, from Fig. 11, EEHC consumes more energies than ACMP 1-hop and 3-hop since ACMP computes a minimum energy consumption route by (1) to report sensed information to the CH. However, EEHC always uses the minimum power to report data to the CH and involves many intermediate nodes for data forwarding. Meanwhile, LEACH also consumes more energies than ACMP 1-hop and 3-hop since LEACH utilizes randomized rotation of CHs to evenly distribute the energy consumptions among sensor nodes in the network. Since the rotation of CHs is periodically performed by EEHC, it will cause more energy consumptions when P_e is low. We also note that the gap of energy consumption among ACMP, EEHC, and LEACH gets bigger as D_e increases since, in higher density network, more sensor nodes will be involved to forward data and thus consume more energies.

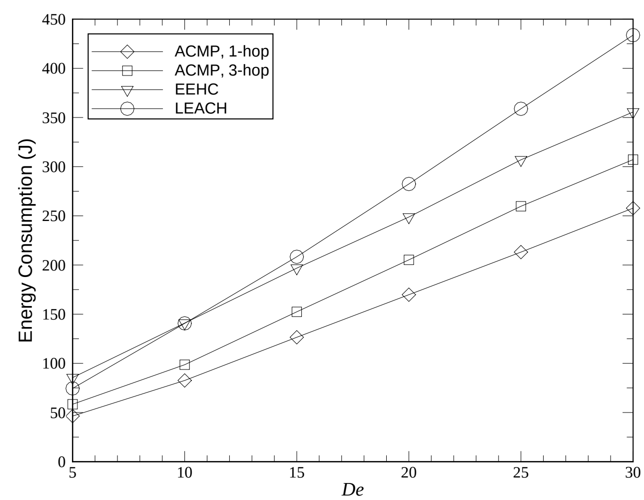

Fig. 11. Density of sensors vs. energy consumption when $P_e = 0.1$ and the simulation time is 15000 rounds.

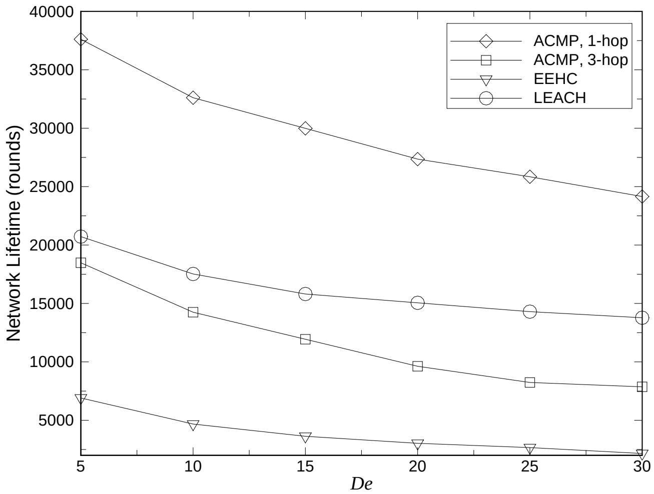

Fig. 12. Density of sensors vs. network lifetime when $P_e = 0.1$.

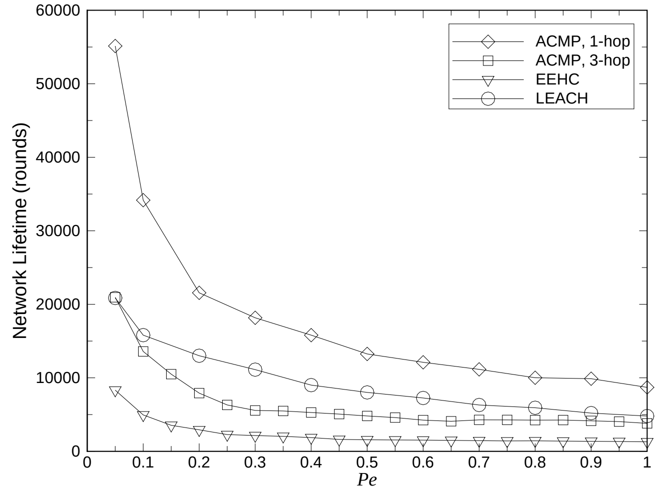

Fig. 14. The P_e vs. network lifetime when N = 1000.

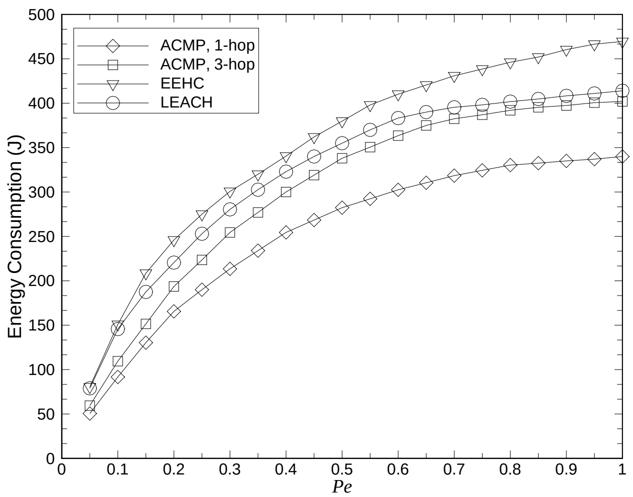

Fig. 13. The P_e vs. energy consumption when N = 1000 and the simulation time is 15000 rounds.

In Fig. 12, the network lifetime of ACMP, EEHC, and LEACH decreases following the increment of the D_e (from 5 to 30). This is because that more nodes will sense events and generate the messages to transfer to the sink node in higher density. This outcome lead to shorter network lifetime. However, ACMP uses the load balance to alleviate the traffic overhead of CHs and expend the lifetime of each node by local re-clustering when its energy is low. Therefore, ACMP can get a longer network lifetime than EEHC. From Fig. 6, we can know ACMP 3-hop spends more energy consumption than ACMP 1-hop, in other words, ACMP 1-hop will have more network lifetime than ACMP 3-hop. The network lifetime of ACMP 1-hop is also longer than LEACH since LEACH utilizes randomly rotation of CHs. This will waste more energies.

To investigate the influence of P_e on energy consumption and network lifetime, we perform a detailed experiment by varying P_e to observe the results obtained by ACMP, EEHC, and LEACH as shown in Fig. 13 and Fig. 14. We can see energy consumptions of ACMP, EEHC, and LEACH increase as P_e increases. ACMP, both in 1-hop and 3-hop, can get lower energy consumption than LEACH and EEHC since either LEACH or EEHC will re-cluster themselves periodically without considering the remaining energy of CHs. On the contrary, ACMP re-clusters depending on the remaining energy of CHs and thus saves more energies than LEACH and EEHC. Notice that the gap of energy consumption between ACMP 1-hop and EEHC gets bigger and bigger when P_e increases since ACMP utilizes autonomous clustering/re-clustering and message passing mechanisms to reduce the probability of one sensor node running out its energy rapidly. This result encourages us to use ACMP especially in the area of frequent event appearance.

Fig. 14 shows the relationship between the network lifetime and P_e. As we can see from Fig. 14, the probability P_e is low while the network lifetime is long. This is because that, ACMP can distribute the traffic load among the CHs and balance the energy consumption well. However, the network lifetime of ACMP, EEHC, and LEACH are getting lower as P_e increases higher. The ACMP 1-hop can obtain a longer network lifetime than EEHC's and LEACH's in higher network density due to local re-clustering scheme to prolong the network lifetime. From these results, we can conclude that a lower energy consumption can be obtained by decreasing the sensor density accordingly.

In ACMP, when the remaining energy of CHs is lower than the threshold of R, it will re-clustering locally. As we discussed early, the re-clustering threshold is set to nine different levels $R = 9/10$, $8/10$,. . . , $1/10$. When the remaining energy of CHs lower than $1/10$, it will re-cluster locally. In Fig. 15 and Fig. 16, we vary the density of sensors from 5 to 30 to investigate the two metrics, energy consumption and network lifetime, in ACMP $1/10$, $2/10$,. . . , $5/10$, respectively. ACMP $i/10$ represents the re-clustering threshold set to $i/10$. We observe that, from Fig. 15, when the D_e increases, it will cost a lot of energy consumptions. The energy consumption increases

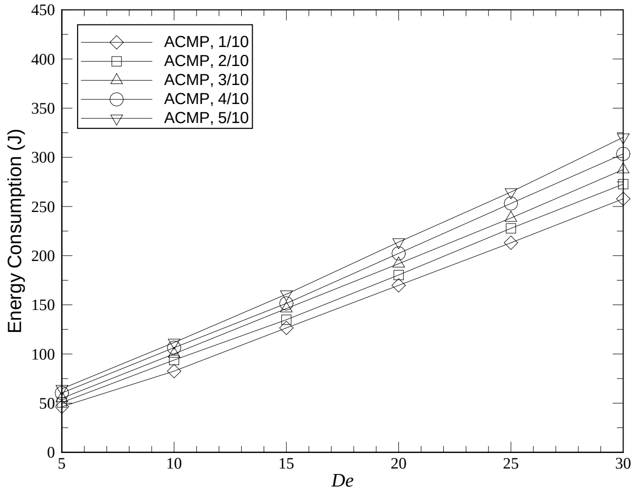

Fig. 15. Density of sensor vs. energy consumption when $p_e = 0.1$ and the simulation time is 15000 rounds. We investigate the relationship of re-clustering threshold $1/10, 2/10, \ldots, 5/10$

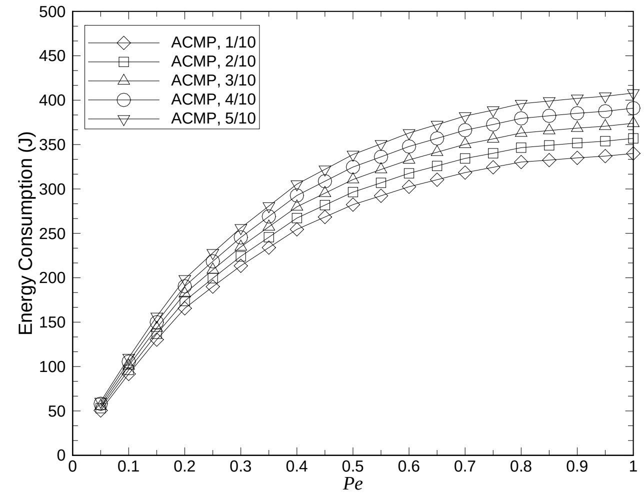

Fig. 17. The p_e vs. energy consumption when N=1000 and the simulation time is 15000 rounds. We investigate the relationship of re-clustering threshold $1/10, 2/10, \ldots, 5/10$

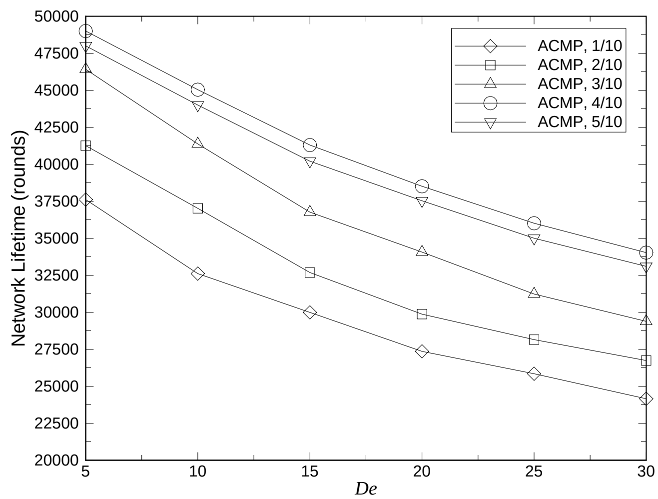

Fig. 16. Density of sensors vs. network lifetime when $p_e = 0.1$. We investigate the relationship of re-clustering threshold $1/10, 2/10, \ldots, 5/10$

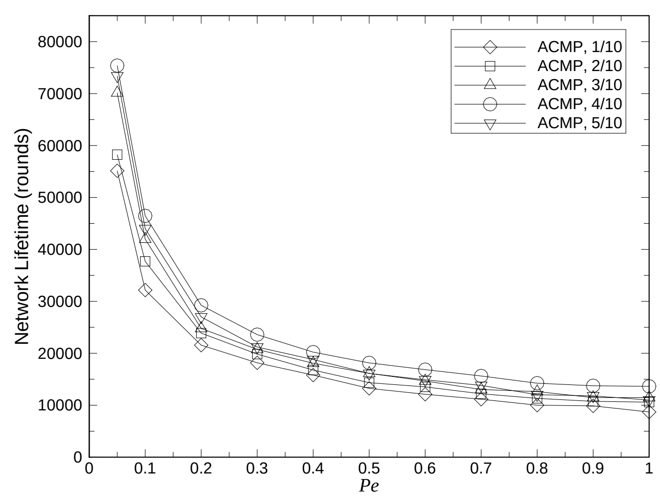

Fig. 18. The P_e vs. network lifetime when N=1000. We investigate the relationship of re-clustering threshold $1/10, 2/10, \ldots, 5/10$

while the re-clustering threshold is increased. It is because that when the re-clustering threshold is high, clusters will re-clustering frequently and it cause a lot of energy consumption.

In Fig. 16, the network lifetime of ACMP $1/10$, $2/10$,…, $5/10$ decreases following the increment of D_e (from 5 to 30). This is because that more nodes will sense events and generate the messages to transfer to the sink node when the D_e is high. This outcome will lead to shorter network lifetime. From Fig. 16, we can know that the network lifetime of ACMP $4/10$ is longer than others. When the re-clustering threshold smaller than $4/10$, the higher re-clustering threshold will obtain more network lifetime. When the re-clustering threshold is $5/10$, the network lifetime is shorter than that of ACMP $4/10$. It is because that higher re-clustering threshold will cause re-cluster frequently and it will cost a lot of energy consumption. In this experiment, we can know that ACMP $4/10$ will get the maximum network lifetime. On the above, we can know when the re-clustering threshold is smaller, the difference of each node's energy consumption is very large, the lower remaining energy sensor nodes will effect the network lifetime. On the other hand, when the re-clustering threshold is larger, the difference of each node's energy consumption is small. But the network re-cluster locally frequently, it cost a lot of energy consumption for the network.

In Fig. 17 and Fig. 18, we increase P_e to investigate the energy consumption of ACMP in different thresholds $1/10$, $2/10$,…, $5/10$. In Fig. 17, the energy consumption increase following the increment of P_e. We can know that ACMP $5/10$ cost maximum energy consumption. This is because that clusters re-clustering frequently and it spends a lot of energy. On the other hand, energy consumption is fewer when re-clustering threshold is smaller.

In Fig. 18, the network lifetime decreases following the increment of P_e. This is because that more nodes will sense events and generate related messages pass to the sink node as D_e is high. This outcome leads to a shorter network lifetime. From Fig. 18, we can know that the ACMP with threshold $4/10$ has the maximum network lifetime. When the re-clustering threshold smaller than $4/10$, the network lifetime increases by the increment of re-clustering threshold. When the re-clustering threshold is larger than $4/10$, the network lifetime does not increase. The reason is similar with previously experiment shown in Fig. 16.

V. CONCLUSION

In this paper, we propose an autonomous clustering and message passing (ACMP) protocol for energy efficiency in wireless sensor networks. The lifetime of the sensor network can be prolonged further by adopting an efficient traffic balance scheme. ACMP provides the load distribution scheme by maintaining the remaining energy to extend the network life time. Meanwhile, ACMP also provides a local re-clustering mechanism to avoid a node runs out of its energy when its remaining energy is low. Besides, ACMP uses the minimum energy consumption route (direct transmission) rather than uses multihop minimum distance route to form the cluster. This strategy enables sensor nodes use energy efficiently to communicate with its CH. Experiment results show that ACMP can achieve a highly energy saving effect as well as prolong the network lifetime.

REFERENCES

[1] A. D. Amis, R. Prakash, T. H. P. Vuong, and D. T. Huynh, "Max-Min D-Cluster Formation in Wireless Ad Hoc Networks," in *Proc. IEEE INFOCOM 2000*, vol. 1, pp. 32–41, Tel Avlv, Israel, Mar. 2000.
[2] S. Bandyopadhyay and E. J. Coyle, "Minimizing Communication Costs in Hierarchically Clustered Networks of Wireless Sensors," in *Proc. IEEE WCNC 2003*, vol. 2, pp. 1274–1279, New Orleans, LA, Mar. 2003.
[3] M. Bhardwaj, T. Garnett, and A. P. Chandrakasan, "Upper Bounds on the Lifetime of Sensor Networks," in *Proc. IEEE ICC 2001*, vol. 3, pp. 785–790, Helsinki, Finland, June 2001.
[4] B. Chen, K. Jamieson, H. Balakrishnan, and R. Morris, "Span: An Energy-Efficient Coordination Algorithm for Topology Maintenance in Ad Hoc Wireless Networks," *ACM/Kluwer Wireless Networks*, vol. 8, no. 5, pp. 481–492, Sept. 2002.
[5] J. Chen and L.-R. Li, "Path Planning Protocol for Collaborative Multi-robot Systems," in *Proc. IEEE CIRA'2005*, Espoo, Finland, June 2005.
[6] Z. Chen and A. Khokhar, "Self Organization and Energy Efficient TDMA MAC Protocol by Wake Up For Wireless Sensor Networks," in *Proc. IEEE SECON 2004*, pp. 335–341, Santa Clara, CA, Oct. 2004.
[7] S. Bandyopadhyay and E. J. Coyle, "An Energy Efficient Hierarchical Clustering Algorithm for Wireless Sensor Networks," in *Proc. IEEE INFOCOM 2003*, vol. 3, pp. 1713–1723, San Francisco, CA, Mar./Apr. 2003.
[8] K. Dasgupta, K. Kalpakis, and P. Namjoshi, "An Efficient Clustering-Based Heuristic for Data Gathering and Aggregation in Sensor Networks," in *Proc. IEEE WCNC 2003*, vol. 3, pp. 1948–1953, New Orleans, LA, Mar. 2003.
[9] D. Estrin, R. Govindan, J. Heidemann, and S. Kumar, "Next Century Challenges: Scalable Coordination in Sensor Networks," in *Proc. ACM/IEEE MobiCom'99*, pp. 263–270, Seattle, WA, Aug. 1999.
[10] G. Gupta and M. Younis, "Load-Balanced Clustering of Wireless Sensor Networks," in *Proc. IEEE ICC 2003*, vol. 3, pp. 1848–1852, Anchorage, AK, May 2003.
[11] W. R. Heinzelman, A. Chandrakasan, and H. Balakrishnan, "Energy-Efficient Communication Protocol for Wireless Microsensor Networks," in *Proc. Hawaii Int. Conf. Systems Sciences 2000*, vol. 2, pp. 1–10, Maui, Hawaii, Jan. 2000.
[12] X. Ji, H. Zha, J. J. Metzner, and G. Kesidis, "Dynamic Cluster Structure for Object Detection and Tracking in Wireless Ad-Hoc Sensor Networks," in *Proc. IEEE ICC 2004*, vol. 7, pp. 3807–3811, Paris, France, June 2004.
[13] J. Kulik, W. Rabiner, and H. Balakrishnan, "Adaptive Protocols for Information Dissemination in Wireless Sensor Networks," in *Proc. ACM/IEEE MobiCom'99*, pp. 174–185, Seattle, WA, Aug. 1999.
[14] S. S. Kulkarni, "TDMA Service for Sensor Networks," in *Proc. IEEE DCS Wksp. 2004*, pp. 604–609, Tokyo, Japan, Mar. 2004.
[15] R. Min, et. al, "An Architecture For a Power-Aware Distributed Microsensor Node," in *Proc. IEEE SiPS'00*, pp. 581–590, Lafayette, LA, Oct. 2000.
[16] T. S. Rappaport, *Wireless Communications: Principles and Practice*, Prentice Hall PTR, Upper Saddle River, New Jersey, 1996.
[17] Y. E. Sagduyu and A. Ephremides, "The Problem of Medium Access Control in Wireless Sensor Networks," *IEEE Wireless Commun.*, vol. 11, no. 6, pp. 44–53, Dec. 2004.
[18] A. Sinha and A. P. Chandrakasan, "Energy Aware Software," in *Proc. Int. Conf. VLSI Design 2000*, pp. 50–55, Calcutta, India, Jan. 2000.
[19] Y. Xu, J. Heidemann, and D. Estrin, "Geography-Informed Energy Conservation for Ad Hoc Routing," in *Proc. ACM/IEEE MobiCom 2001*, pp. 70–84, Rome, Italy, July 2001.
[20] Y. Xu and H. Qi, "Decentralized Reactive Clustering for Collaborative Processing in Sensor Networks," in *Proc. IEEE ICPADS 2004*, pp. 54–61, Newport Beach, CA, July 2004.
[21] W. Ye, J. Heidemann, and D. Estrin, "An Energy-Efficient MAC Protocol for Wireless Sensor Networks," in *Proc. IEEE INFOCOM 2002*, vol. 3, pp. 1567–1576, New York, USA, June 2002.
[22] O. Younis and S. Fahmy, "HEED: A Hybrid, Energy-Efficient, Distributed Clustering Approach for Ad Hoc Sensor Networks," *IEEE Trans. Mobile Comput.*, vol. 3, no. 5, pp. 366–379, Oct./Dec. 2004.

Jenhui Chen received his B.S. and Ph.D. degrees in Computer Science and Information Engineering (CSIE) from Tamkang University in July 1998 and January 2003, respectively. In the spring of 2003, he joined the faculty of Computer Science and Information Engineering Department at Chang Gung University and served as the Assistant Professor. Dr. Chen served as the referee for several famous international journals including IEEE Transactions on Fuzzy Systems, IEEE Transactions on Mobile Computing, IEEE Transactions on Wireless Communications, IEEE Transactions on Parallel and Distributed Systems, IEEE Transactions on Computers, and ACM/Kluwer Mobile Networks and Applications (MONET). He served as a Program Committee Member for the 2005 IEEE International Conference on Parallel and Distributed Systems and a referee for the 2005 IEEE/RSJ International Conference on Intelligent Robots and Systems. Dr. Chen was a Local Chair of Mobile Computing Workshop 2005. His main research interests include design, analysis, and implementation of communication and network protocols, wireless networks, robotics, artificial intelligence, and bioinformatics. He is a member of IEEE and IEICE.

Chien-Chun Joe Chou received the M.S. degree and the Ph.D. degree of Computer Science from University of Iowa, Iowa City, USA, by 1992. From 1993 to 1997, Joe was an Associate Professor at the Computer Science Department of Tamkang University, Taipei, Taiwan. Since 1997, Joe has been working as a computer system and performance architect for several Silicon Valley companies including HAL and Intel. Currently, Joe is the Director of Architecture at Sonics, Inc., focusing on System-on-Chip interconnect IP hardware specification, simulation modeling, and performance analysis.