A Non_Ack Routing Protocol in Ad-hoc Wireless Sensor Networks

[1,2] Ching-Mu Chen, [1,3] Tung-Jung Chan and [1] Tair-Rong Chen
[1] Department of Electrical Engineering
National Changhua University of Education
Bao-Shan Campus: No.2, Shi-Da Road, Changhua City
Taiwan, R.O.C.
d95621003@mail.ncue.edu.tw
[2] Department of Computer Science and Information Engineering
Chung Chou Institute of Technology
No. 6, Lane 2, Sec. 3, Shanjiao Rd., Yuanlin Township, Changhua County 510
Taiwan, R.O.C.
kchen@dragon.ccut.edu.tw
[3] Department of Electrical Engineering
Chung Chou Institute of Technology
No. 6, Lane 2, Sec. 3, Shanjiao Rd., Yuanlin Township, Changhua County 510
Taiwan, R.O.C.
jung@dragon.ccut.edu.tw

Abstract: - A wireless sensor network contains many sensor nodes in a certain sensing area. Every sensor node consists of a central processing unit, transmitting unit, receiving unit and power supply unit. In order to have efficient sensed messages, both the base station and sensor nodes have to communicate for each others as well. Therefore, a certain sensing area is divided into many sub-areas and every sensor node transmits its message to cluster head which, normally, will integrate and forward these messages to the base station. At the first step, set-up phase is applied to form clusters and to select cluster heads. During the second phase, the transmission phase, every sensor node received a message from another node has to transmit a confirmed message, acknowledgement (*ACK*), to ensure whether or not the data has been received which means every sensor node will dissipate much energy on transmitting and receiving. However, this paper proposes an adaptive routing protocol called *Non_Ack* in the Ad-hoc wireless sensor networks that is the data for every sensor node is transmitted directly to the cluster head and then forwards back to the base station. Some conditions for these sensor nodes with the sensed data directly transmitted without any acknowledgement may be failed so that this paper also considers as nodes fail to transmit the data back, the base station will be taken over for the network. By the consideration of these failure nodes, the base station will establish a binary tree for fault tolerance. A binary tree in the base station is also a big help for query-based routing protocol. Finally, simulation results reveal the entire network lifetime can be extended well and the performance is much better.

Key-Words: - Energy efficiency, Network lifetime, Ad-hoc wireless sensor networks, Routing protocol, No acknowledgement, Binary tree, Query-based.

1 Introduction

In recent years, high technologies [1-3] come with the better performance solution such as micro-chip design, tiny embedded OS, faster data processing with lower power chip, and longer lifetime battery. With these significant hardware improvements, the entire wireless sensor network lifetime can be extended very well. However, a wireless sensor network consists of many sensor nodes. Every sensor node is mainly composed of the central processing unit, transmitting unit, receiving unit, and power supply unit as shown in Fig. 1.

A transmitting unit also contains an amplifier unit that dissipates the most energy in order to transmit a message. To transmit a message via the amplifier unit consumes the energy depending on how far the data is transmitted. The more distance the sensor node transmits, the more energy it needs. Therefore, in order to save energy via the transmitting unit, the data of every sensor node is better to transmit to the closer sensor nodes. Also, with the proposed *Non_Ack* routing protocol in this paper, the transmitted and received energy of every sensor node can be saved.

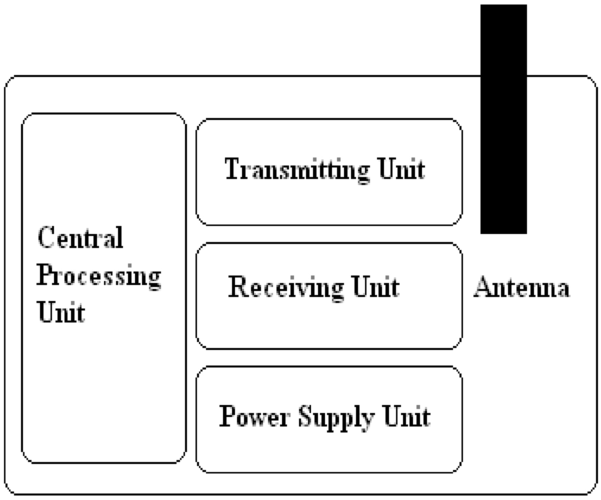

Fig. 1: Architecture of sensor node.

Due to the wireless technique, messages of each sensor node can be delivered to the unattended place as needed. Many applications can be applied via the wireless sensor network such as temperature, hurricane, medical [4], military, chemical and so on. For example, sensor nodes can be randomly deployed into a forest to catch the messages of human temperature, chemical weapon and so on. The problem with a forest is there are many trees out there so that enemies and weapons are difficult to be founded. Therefore, with the wireless sensor networks, it is easy to find enemies and locate the weapons.

Technically, the direct transmission technique [5] [6] is sensor nodes can transmit the data directly to the base station so that the response time to every sensor node is very fast but dissipating too much energy. The problem with the direct transmission technique is the distance from the sensor node to the base station. The more distance for a sensor node to deliver messages, the more energy it needs. Therefore, in order to short the distance for sensor nodes to transmit a long distance message, multi-hop technique is reasonably applied. The multi-hop technique is that every node is hopping in a short distance so that the energy of sensor nodes can be saved because the distance of sending data for every sensor node is shorter. Since sensor nodes communicating for each other will dissipate much energy [7], sensor nodes should spend as little energy [8] as possible in exchanging data.

Moreover, the low energy adaptive clustering hierarchy [9] shows better performance because of the random clustering. It uses probability and randomly cluster head choosing method to extend its networking lifetime. Papers [10-12] also compared to the LEACH protocol. The method of clustering is that a certain sensing area is divided into many sub-areas whereas every sub-area delivers its message to another sub-area or directly to the base station. However, every cluster has a cluster head to transmit its message. If a cluster head keeps receiving and transmitting a message to the base station, it will run out its energy quickly. Therefore, a cluster head needs to be re-selected so that every node can be dissipating its energy evenly. Therefore, there are two phases in the architecture of LEACH. One is the setup phase that is to select one of sensor nodes as a cluster head for each cluster in each round. During the setup phase, one of the sensor nodes in each cluster will be randomly chosen as a cluster head with pre-determined parameter *p.* Sensor nodes closer to the cluster head will be a cluster. Another step is the transmission phase that is the data of non-cluster head node is transmitted to the cluster head and then forwards them directly to the Base Station. Therefore, the architecture of LEACH uses the method of random and probability to save the energy consumption. The main purpose for every sensor node is to collect useful data back to the base station. During the data transmitted and received, sensor nodes need much energy in order to have longer surveillance. In order to have the sensed data back to base station, wireless and networking technique are reasonable applied especially in routing protocol scheme [13]. Therefore, energy consumption [14-16] becomes more and more important to be discussed.

Moreover, sensor nodes directly send the data back to base station and its energy will be run out quickly. The multi-hop; however, can have better energy performance compared to the direct transmission technique. However, these schemes can use the acknowledgement (*ACK*) to ensure that data can be transmitted back to the base station during the transmission phase. However, sending data back to the base station between acknowledgement (*ACK*) and no acknowledgement (*Non_Ack*) will be discussed in this paper. By the proposed scheme, *Non_Ack*, inactive node stops the data sending back to the base station. However, in order to avoid this problem, *CheckID* of sensor node and cluster head will be checked by the base station. By the *CheckID* of sensor node and cluster head, the base station can easily trace the data as needed. Finally, with the proposed *Non_Ack* routing protocol, not only it can extend the network lifetime, it also can be used for both querying data and failure nodes' checking.

2 Network Model

As shown in Fig. 1, every sensor node consists of four units that are a central processing unit, transmitting unit, receiving unit and power supply unit. The most dissipated energy is from the central processing unit, transmitting unit and receiving unit. In transmitting unit, it also needs an amplifier to transmit the message of sensor nodes. The more distance a sensor node transmits, the more energy a sensor node dissipates. Moreover, [17-20] the dissipating energy of receiving unit depends on the total amounts of message that sensor nodes receive.

The transmitted energy consumption, E_{TX}, from one node to another node is given by

$$E_{TX} = l \cdot E_{elec} + l \cdot \varepsilon_{fs} d_{toCH}^2 \ , \tag{1}$$

where l represents a message (bits) between two sensor nodes' communication. Here, assumption of free space d^2 is applied. ε_{fs} represents an amplified transmitting energy in free space.

The received energy consumption, E_{RX}, from one node to another node is given by

$$E_{RX} = l \cdot E_{elec} . \tag{2}$$

Parameters used in this paper are shown in the table 1.

Table 1

Notation	Description
E_o = 0.5J/bit	Initial energy for every node
E_{elec} = 50nJ/bit	Per bit energy consumption
E_{DA} = 5nJ/bit	Energy consumption of data aggregation
ε_{fs} = 10pJ/bit/m^2	Amplified transmitting energy in a free space
d_{toCH}^4	Energy consumption of the distance from a node to the cluster head in a multi-path environment
d_{toCH}^2	Energy consumption of the distance from a node to the cluster head in free space environment

A sensor node transmits a message to another sensor node which will send a confirmed message back to ensure the message has been delivered already. The dissipated energy of a sensor node to transmit the energy is E_{TX}. Similarly, the dissipated energy of a sensor node to receive the energy is E_{RX}. Therefore, as a sensor node transmits a message, it will also receive a confirmed message. However, the total amount of dissipated energy of a sensor node is given by

$$\begin{aligned} E_{Ack\_Node} &= E_{TX} + E_{RX} \\ &= l \cdot E_{elec} + l \cdot \varepsilon_{fs} d_{toCH}^2 + l \cdot E_{elec} \\ &= 2 * l \cdot E_{elec} + l \cdot \varepsilon_{fs} d_{toCH}^2 . \end{aligned} \tag{3}$$

Equation (3) is assumed that the amounts of transmitted messages from one sensor node to another sensor node are equal to the confirmed message from one sensor node to another sensor node. The reason why the energy needs to be counted as twice is the message sent to another sensor node consumes the acknowledgement message (*ACK*) to ensure that the message can be sent out. In reality, the amount of *ACK* message is much less than the original transmitted message. Therefore, $\frac{1}{10}$ of the amount of *ACK* messages are assumed and expressed by

$$\begin{aligned} E_{Ack\_Node} &= E_{TX} + E_{RX} \\ &= l \cdot E_{elec} + l \cdot \varepsilon_{fs} d_{toCH}^2 + \frac{1}{10} \cdot l \cdot E_{elec} \\ &= 1.1 \cdot l \cdot E_{elec} + l \cdot \varepsilon_{fs} d_{toCH}^2 . \end{aligned} \tag{4}$$

However, suppose one of sensor nodes is failed or inactive, other sensor nodes sending the message to this sensor node will keep sending message until sensor nodes run out of their energies. Therefore, sensor nodes can transmit their messages directly to other sensor nodes with no acknowledgement, *Non_Ack*. With the *Non_Ack* scheme, every sensor node can directly transmit the message to the cluster head and then forwards the message directly back to the base station. This will not consume the *ACK* energy and, therefore, the network lifetime can be extended. The consumed energy with *Non_Ack* is given by

$$E_{NACK\_Node} = l \cdot E_{elec} + l \cdot \varepsilon_{fs} d_{toCH}^2 . \tag{5}$$

Moreover, suppose there are n nodes in a cluster. Every sensor node sends its message to the cluster head and then the cluster head consumes the energy is given by

$$E_{Total\_CH} = n \cdot l \cdot E_{elec} . \tag{6}$$

Furthermore, as the cluster head receives the data from each node, the *ACK* message is no need to send it back right away. The optimal consumed energy from the cluster head to all its member nodes is given by

$$E_{OptCHMember} = l \cdot E_{elec} + l \cdot \varepsilon_{fs} d^2_{Max\_toCH} . \tag{7}$$

3 A Non_Ack Routing Protocol Model

Normally, it has two phases in the Ad-hoc wireless sensor networks that are set-up and transmission phase. During the setup phase, clusters are formed and cluster heads are selected. During the transmission phase, non cluster heads transmit the data to the cluster head and then cluster head will forward the data back to the base station [21][22] as shown in Fig. 2.

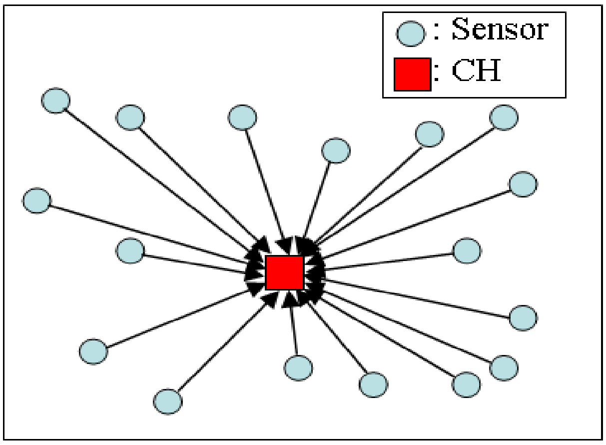

Fig. 2: Sensor nodes transmit the data to the cluster head.

Fig. 3 shows sensor nodes are randomly deployed in a cluster, for example. *N1*, *N2*, *N3* and so on are sensor nodes in a cluster. For example, *N1* needs to transmit data (*T1*) to the cluster head as shown in Fig. 4. The cluster head will send a confirmed message (*Ack1*) back. However, there are 17 sensor nodes in a cluster. Suppose every sensor node will transmit a data to the cluster head. There will be totally 17 confirmed messages from the cluster head to all its member nodes.

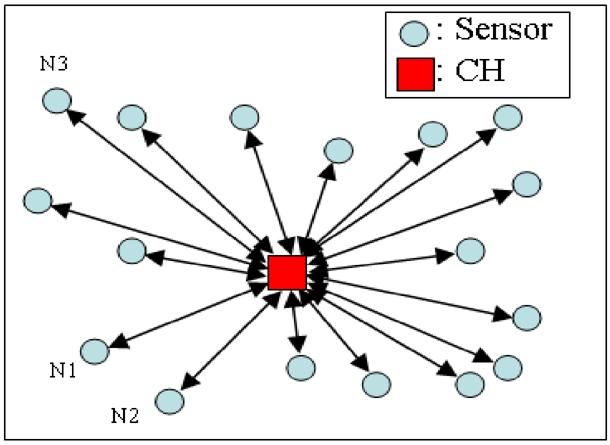

Fig. 3: A cluster head with sensor nodes in a cluster.

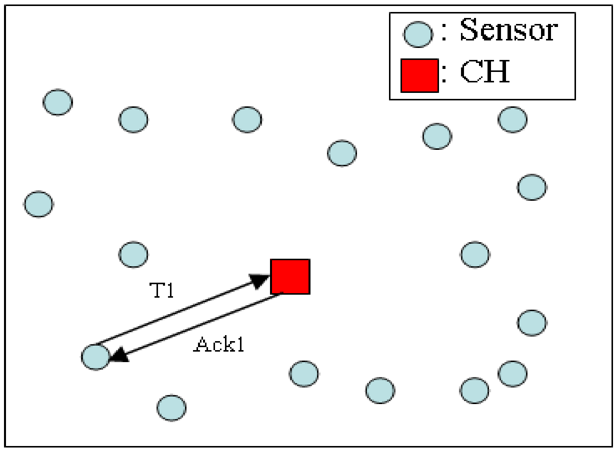

Fig. 4: A sensor node transmits data to cluster head which will send a confirmed message back.

In order to save energy, the cluster head can only transmit a single message which is longest message as shown in Fig. 5. Once the cluster head sends a message (*Ack_L*) which is longest distance from the cluster head to all sensor nodes and then all sensor nodes will be covered by this signal (*Ack_L*).

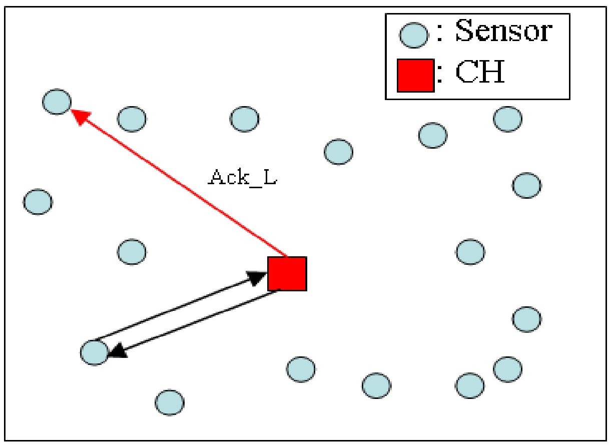

Fig. 5: *Ack_L* is the longest distance for the cluster head to all sensor nodes.

In order to ensure the data can be transmitted to the cluster head, sensor nodes need to communicate with cluster head so that it dissipates energy very much. In order to save energy and extend the network lifetime, an adaptive routing protocol with no acknowledgement (*Non_Ack*) is proposed in this paper that is sensor nodes transmit the data to the cluster head without the confirmed message from the cluster head in the transmission phase as shown in Fig. 6. The confirmed message, *Ack1* is become *Non_Ack*, will be no more needed that means all sensor nodes transmit the data directly to the cluster head and then forward it directly back to the base station.

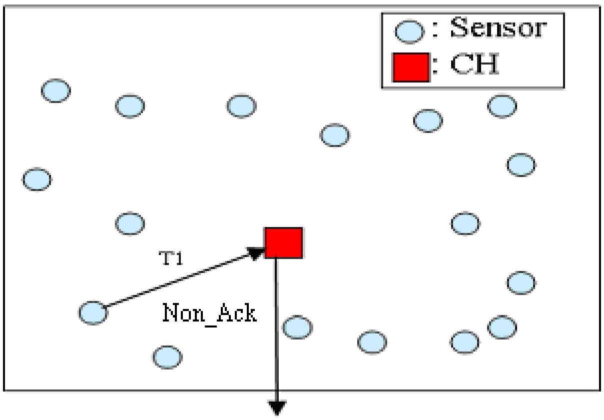

Fig. 6: Sensor node sends a message to the base station with no acknowledgement (*Non_Ack*).

Moreover, sensor nodes fail to send data back can be avoided from the network. The following steps are to avoid this problem of the routing protocol with no acknowledgement.

1. Sensor nodes periodically send the data back.
2. Every packet contains a *CheckID* of every node.
3. Establish a binary tree (Fig. 7) in the base station that a node with an odd-number *CheckID* (Fig. 8) is at the left side of the root node or parent node and a node with an even-number *CheckID* (Fig. 9) is at the right side of the root or parent node.
4. If some of the *CheckID*s are missing, the base station records it for at most 3 times and then updates the node information as inactive.
5. If the base station receives the missing *CheckID* again, the node information with inactive will be updated as active.

Since sensor nodes transmit the data to the cluster head, some of them may not be working properly or may become inactive sensor nodes. Therefore, the base station will be responsible to pick up one of the non cluster head nodes as a new cluster head to continue the entire process.

Furthermore, every packet of a sensor node contains a *CheckID*; therefore, *i*-bit is needed for transmission phase expressed by

$$i(bits) = \log_2 n\,, \tag{8}$$

where *i* represents the bit-number needed to transmit of a *CheckID* and *n* represents numbers of sensor nodes are randomly deployed in wireless sensor networks.

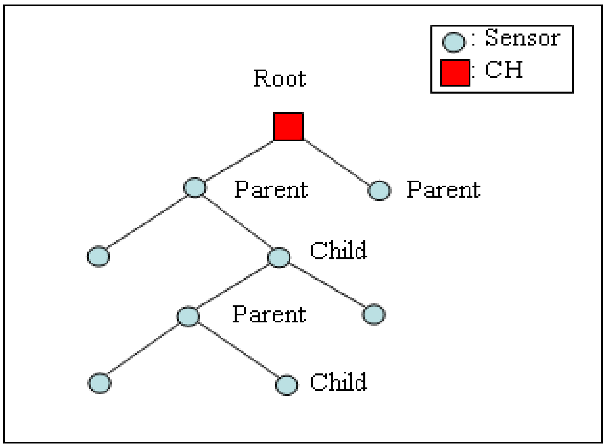

Fig. 7: Binary tree in the base station.

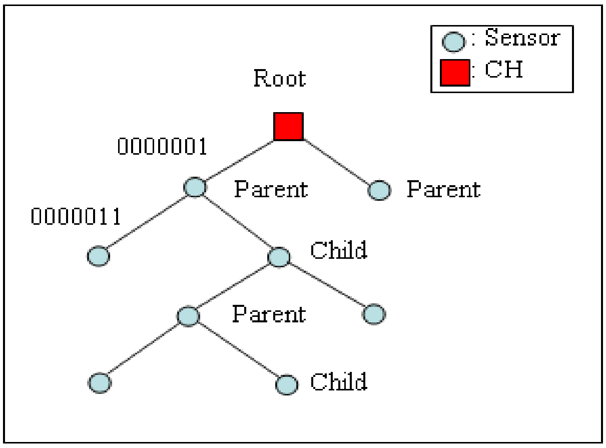

Fig. 8: A node with an odd-number *CheckID*.

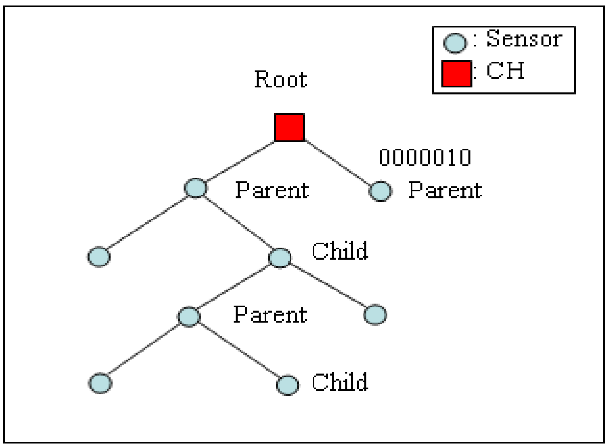

Fig. 9: A node with an even-number *CheckID*.

In order to avoid the collision, the Time Division Multiple Access (TDMA) is also applied. Each cluster has one cluster head that also contains a *CheckID*. Because of the cluster head has its own *CheckID* and each cluster is separated from others. Therefore, during the data transmitted, every cluster can be working at the same time as shown in Fig. 9 and the table 2 is about the schedule of sensor nodes.

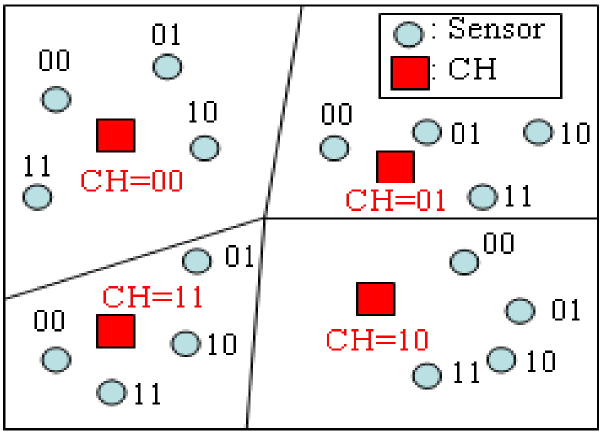

Fig. 9: *CheckID* for Both sensor nodes and CH.

Table 2: Schedule of sensor nodes and cluster heads.

Nodes / CH	Node 1	Node 2	Node 3	Node 4
00	00	01	10	11
01	00	01	10	11
10	00	01	10	11
11	00	01	10	11

By the *Non_Ack* routing protocol, every sensor node can directly transmit the data to the cluster head which will forward the integrated data back to the base station. Those failed sensor nodes will be taken over by the base station with the *CheckID*. With the *CheckID*, the base station also can query the data as needed. Since the base station establishes a binary tree and the *CheckID* with odd-number of sensor node is stored at the left-side of the binary tree whereas *CheckID* with even-number of sensor node is stored at the right-side of the binary tree.

Therefore, it is easier to have the event-driven with the query-based from the base station. To query the odd-number with *CheckID* is shown in Fig.10. To query the even-number with *CheckID* is shown in Fig. 11.

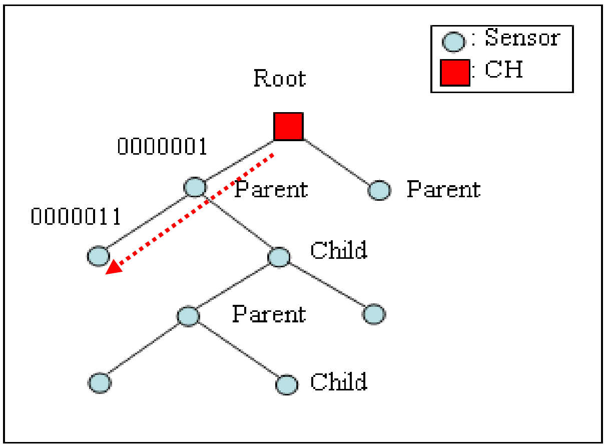

Fig. 10: Query of the odd-number with *CheckID*.

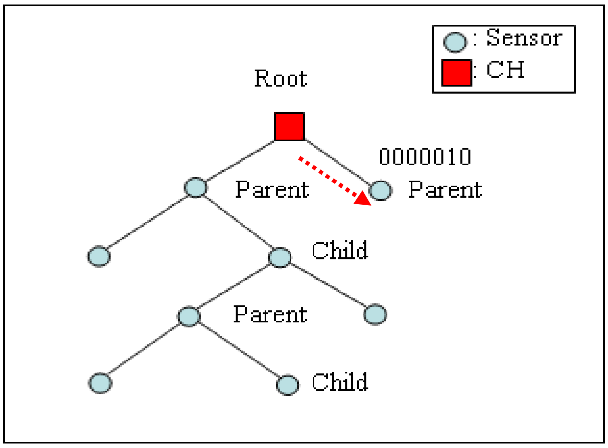

Fig. 11: Query of the even-number with *CheckID*.

4 Simulation results

In this paper, 100 sensor nodes are randomly deployed into 100 m x 100 m square as shown in Fig. 12. and the simulation tool, Matlab, is used as a simulator. Fig. 13 and Fig. 14 show the formation of clusters and how many sensor nodes are still alive there. Here, "o" denotes a sensor node is still alive where as ". " denotes a sensor node is no longer alive and "* " denotes a cluster head. Also, sensor node's colour in Fig. 14 represents which node belongs to which cluster. It is obvious that Fig. 13 shows the energy usage for sensor nodes far away from the base station is much less than those sensor nodes closer to the base station. However, Fig. 14 shows more sensor nodes are still alive compared to the Fig. 13 that means a *Non_Ack* routing protocol is much better.

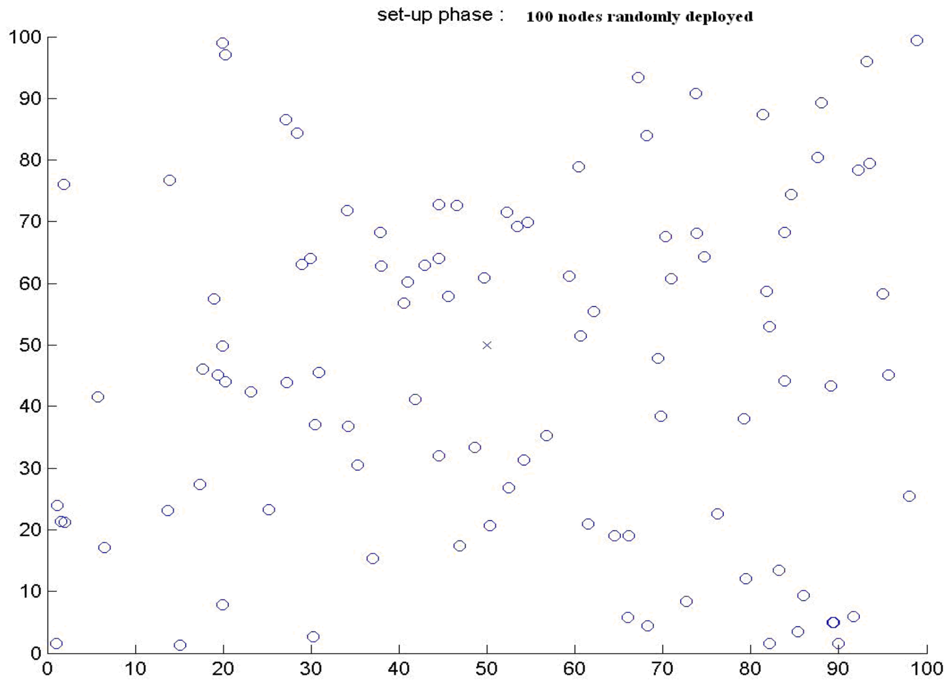

Fig. 12: 100 nodes randomly deployed.

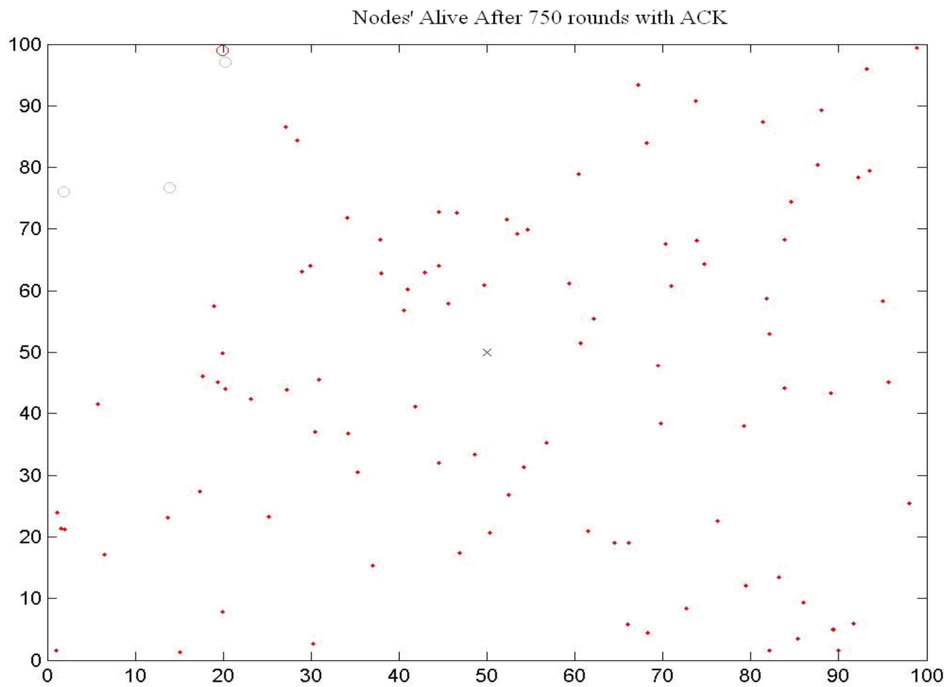

Fig. 13: Numbers of node still alive after 750 rounds in *AC*K routing protocol.

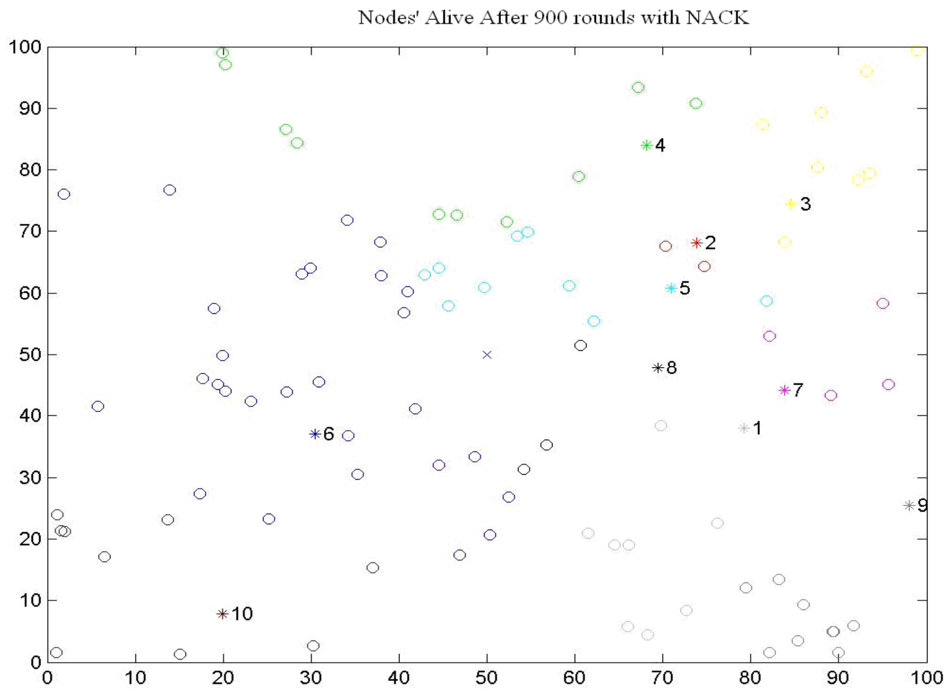

Fig. 14: Numbers of node still alive after 900 rounds in *Non_Ack* routing protocol.

Fig. 15 shows nodes' lifetime in rounds. Simulation shows that the first node ran out of its energy at round 750 and the last node died at the round 1050. This shows no matter how far node is, energy usage for every round is almost the same. Fig. 16 shows nodes' lifetime in every round. Simulation show that the first node ran out of its energy at round 900 and the last node died at the round 1695. This shows no matter how far node is, energy usage for every round is almost the same.

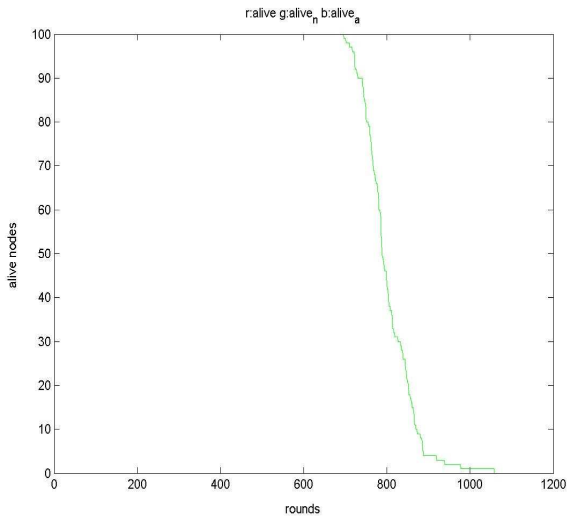

Fig. 15: Network lifetime with *ACK*.

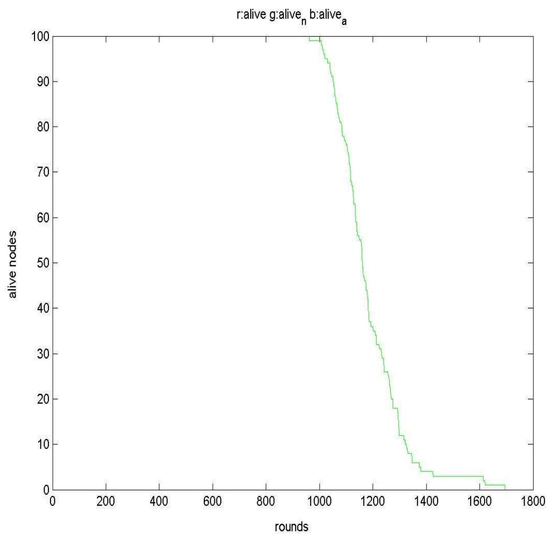

Fig. 16: Network lifetime with *Non_Ack*.

In order to dissipate the energy from every node evenly, probability and random variable are used for this simulation. Fig. 17 shows that before 800 rounds, energy can be dissipated evenly. Moreover, after first node ran out of its energy, the total energy is not much left. In order to dissipate the energy from every node, probability and random variable are used for this simulation. Fig. 18 shows that before 1200 rounds, energy can be dissipated evenly. Moreover, after first node ran out of its energy, the total energy is not much left.

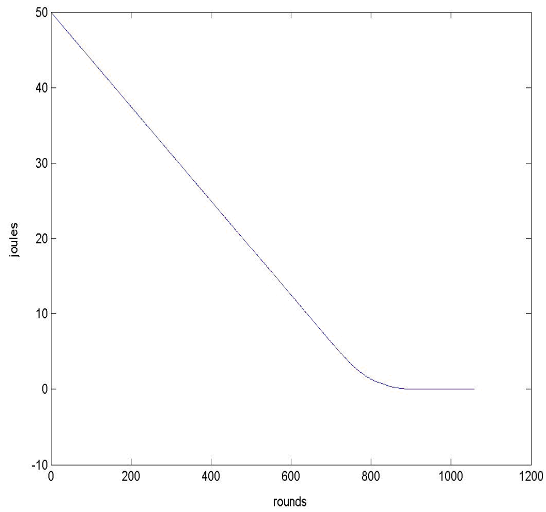

Fig. 17: Total energy changed in rounds with *ACK*.

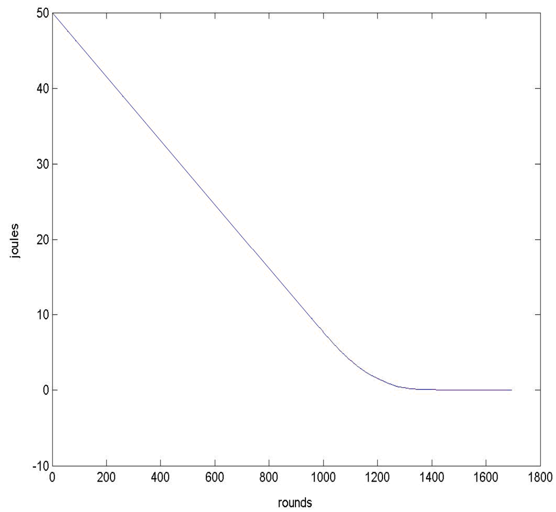

Fig. 18: Total energy changed in rounds with *Non_Ack*.

However, by comparing the Fig. 13 and Fig. 14, the proposed *Non_Ack* routing protocol in this paper is much better. By the Fig. 15 and Fig. 16, we can see there are more than 600 rounds better in the proposed *Non_Ack* routing protocol. Also, Fig. 17 and Fig. 18 show the better performance of the proposed *Non_Ack* routing protocol

4 Conclusion

This paper proposes a *Non_Ack* routing protocol routing protocol. The simulation result reveals its performance is much better than the scheme with *ACK*. The reason why an ad-hoc wireless sensor network needs a routing protocol with *Non_Ack* is it has the characteristic of faster speed to send data back to the base station and it can save much energy. However, in order to make sure whether or not a sensed data can be surely transmitted back to station, the base station can play the role of judging which node is failure to transmit data back. Once one of sensor nodes is failed to transmit the data back, the base station will inform other sensor nodes to continue to transmit the data back. Finally, the base station checks the failure nodes with the *CheckID* which also can be used for the specific data query.

References:

[1] A. Willig, "Recent and Emerging Topics in Wireless Industrial Communications: A Selection", IEEE Transactions on Industrial Informatics, Vol. 4, No. 2, May 2008.

[2] I. F. Akyildiz, W. Su, Y.Sankarasubramaniam, E. Cayirci, "Wireless sensor network: a survey", *Computer Networks*, Vol. 38, pp. 393-422, 2002.

[3] H. Luo, Y. Lin and S. K. Das, "Routing Correlated Data in Wireless Sensor Networks: A Survey", *IEEE Network*, vol. 21, no.6, Nov/Dec. 2007, pp. 40-47.

[4] F. Rahman, N. Shabana, “Wireless Sensor Network based Personal Health Monitoring System”, *WSEAS Transactions on Communications*, Issue V, Vol. 5, May 2006, pp. 966–972.

[5] W.R. Heinzelman, A.P. Chandrakasan, and H. Balakrishnan,"Energy-EfficientCommunication Protocol for Wireless Microsensor Networks", *Proc. 33rd Hawaii Int’l. Conf. Sys. Sci.*, Jan. 2000.

[6] J. Zhu and S. Papavassiliou, "On the energy-efficient organization and the lifetime of multi-hop sensor networks", *IEEE Commun. Letters*, Vol. 7, No. 11, pp. 537-539, Nov. 2003.

[7] D. Culler, D. Estrin and M. Srivastava, "Overview of sensor networks", *IEEE Computer*, Vol. 37, Issue 8, pp. 41- 49, Aug. 2004.

[8] V. Raghunathan, C. Schurgers and S. Park and M. B. Srivastava, "Energy-aware wireless microsensor networks", *IEEE Signal Processing Magazine*, Vol. 19, No. 2, pp. 40-50, March 2002.

[9] W.B. Heinzelman, A.P. Chandrakasan, and H. Balakrishnan,"An Application-Specific Protocol Architecture for Wireless Microsensor Networksv, *IEEE Trans. Wireless Commun.*, vol. 1, no. 4, Oct. 2002, pp. 660–70.

[10] O. Younis and S. Fahmy,"HEED: a hybrid, energy-efficient, distributed clustering approach for ad hoc sensor networks", *IEEE Trans. on Mobile Computing*, pp. 660-669, 2004.

[11] S.D. Muruganathan, D.C.F. Ma, R.I. Bhasin, and A.O. Fapojuwo, "A Centralized Energy-Efficient Routing Protocol for Wireless Sensor Networks", *IEEE Radio Communication,* 2005
[12] Y.-R. Tsai, "Coverage-Preserving Routing Protocols for Randomly Distributed Wireless Sensor Networks", *IEEE Trans. Wireless on Wireless Commun.*, Vol. 6, No. 4, Apr. 2007.
[13] C. Schurgers and M.B. Srivastava, "Energy efficient routing in wireless sensor networks", *IEEE Military Comm. Conf.*, Vol. 1, pp. 357-361, Oct. 2001.
[14] V. Raghunathan *et al.*,"Energy-Aware Wireless Microsensor Networks", *IEEE Sig. Proc. Mag.*, vol. 1, no. 2, Mar. 2002, pp. 40–50.
[15] M. Younis, M. Youssef, K. Arisha, "Energy-aware routing in cluster-based sensor networks", *10th IEEE International Symposium on Modeling, Analysis and Simulation of Computer and Telecommunications Systems*, pp. 129 – 136, 2002.
[16] M. Cardei and J. Wu, "Energy-Efficient Coverage Problems in Wireless Ad Hoc Sensor Networks", *Computer Communications*, vol. 29, no. 4, Feb. 2006, pp. 413-420.
[17] J. Levendovszky, B. Hegyi, "Optimal Statistical Energy Balancing Protocols for Wireless Sensor Networks", *WSEAS Transactions on Communications*, Issue V, Vol. 6, May 2007, pp. 689–694.
[18] Cerpa and D. Estrin, "ASCENT: Adaptive self-configuring sensor networks topologies", IEEE Transactions on Mobile Computing, vol. 03, No. 3, pp. 272-285, July 2004.
[19] M. Veyseh, B. W. Wei, and N. F. Mir, "Clustering and Synchronization Protocol in a Wireless Sensor Networks", WSEAS *Transactions in Communications*, 2006.
[20] K. kalpakis, K. Dasgupta and P. Namjoshi, "Efficient algorithm for maximum lifetime data gathering and aggregation in wireless sensor networks", Computer Networks Volume 42, Issue 6, 21 August 2003, pp.697-716.
[21] Gracanin, D. Eltoweissy, M. Olariu, S. Wadaa, "On modeling wireless sensor networks", A. Parallel and Distributed Processing Symposium, 2004.
[22] R. M. Patrikar and S. G. Akojwar, " Neural Network Based Classification Techniques For Wireless Sensor Network with Cooperative Routing", 12th WSEAS International Conference on Communications.
[23] H. Gupta, Z. Zhou, S. R. Das, and Q. Gu, "Connected sensor cover: self-organization of sensor networks for efficient query execution", *IEEE/ACM Trans. Network.*, 14(1):55–67, 2006.