An Adaptive Traffic Load Based Scheduling Protocol for Wireless Sensor Networks

Prasan Kumar Sahoo
Department of Computer Science
and Information Engineering,
Chang Gung University,
Taoyuan, 33302, Taiwan
Email: pksahoo@mail.cgu.edu.tw

I-Shyan Hwang, Chun-Yuan Cheng
Dept. of Information Communication,
Dept. of Computer Science and Engineering,
Yuan-Ze University,
Taoyuan, 32026, Taiwan
Email: ishwang@saturn.yzu.edu.tw

***Abstract*—Wireless sensor network is used for several applications starting from surveillance to health monitoring. Nodes are usually deployed randomly and densely over the monitored region and are supposed to monitor it time to time. The nodes in wireless sensor networks are battery powered and therefore, it is crucial to manage the power consumption of the nodes efficiently. Though most of the existing power-saving protocols minimize the power consumption by periodic sleep and wake up schedules, they fail to adjust a sensor node's sleep duration based on its traffic load. In this paper an adaptive traffic load based node scheduling protocol is proposed to decide the active and sleep schedules of the nodes. The whole network is divided into finite number of virtual zones and a routing path algorithm is designed based on residual energy of the next hop nodes. Simulation results of our protocol shows that the control packet overhead and energy consumption are reduced considerably as compared to similar quorum-based medium access control protocols.**

I. Introduction

In Wireless Sensor Networks (WSNs), normally nodes are distributed randomly to form the network without help of any infrastructure. It is used for several applications such as military, environmental, health monitoring, and mobile object tracking. The nodes in WSNs are inexpensive, small sized, battery-powered sensing devices with ability of sensing, processing, and are expected to transmit data to a remote base station. Each sensor node is capable of aggregating data, storing, computing, and transmitting to complete its own mission. After the mission is completed, sensor nodes always coordinate to transmit the requested information to the sink hop-by-hop. In general, sensor nodes are battery powered and sometimes it is impossible to recharge them. When a large number of sensors are dead due to power consumption, the nodes must be recycled and redeployed. This operation increases considerable cost. For this reason, it is very important to design efficient protocols to save power and let the network lifetime be extended.

There have been many approaches to prolong the network lifetime, such as energy-efficient MAC protocols [1], [2], [3], [4], [5], routing protocols [6], and deployment algorithms [7]. On the one hand, since idle listening has been identified as a major reason of energy waste, papers in [1], [2] have proposed to reduce the time of a sensor node spent in idle listening. On the other hand, as proposed in [3], the listen period can be adjusted by the sensors with a predefined active/sleep mode based on the traffic load. However, these protocols have to check whether the data needs to be received or transmitted at beginning of each time frame or not. Besides, designing the fixed active/sleep schedule will waste unnecessary power when the network traffic is light, and it also causes additional delay when the network traffic is heavy.

In this paper, we focus on designing energy-efficient MAC protocols to prolong network lifetime. An adaptive traffic load-based scheduling protocol is proposed to improve the above-mentioned disadvantages. The main idea is to use a traffic load model to combine with grid-based quorum. By the traffic load model, each sensor will calculate its own total traffic load by the density of its communication range and the hop counts from the sink node. Therefore, it can set traffic load dynamically and know how many time frames must wake up. In $N \times N$ grid-based quorum, each sensor has the ratio of $\frac{(2N-1)}{N^2}$ wake-up time slots. Hence, the grid size would influence the ratio of the wake-up time and the sleeping time and the larger grid size has the lesser wake-up time. Accordingly, the grid size can be adjusted dynamically based on the current traffic load and quorum, which implies that the active/sleep scheduling can be controlled dynamically. Based on this mechanism, a sensor network can cope with unexpected traffic loads to handle the events easily, which is more suitable for real situations.

The rest of the paper is organized as follows. Related work is discussed in Section II. System model is given in Section III and proposed algorithms are designed in Section IV. Performance evaluation of the proposed protocols are done in Section V and concluding remarks are made in Section VI.

II. Related Work

In the past few years, several medium access control protocols are proposed to minimize the energy consumption of the sensor networks. The Sensor-MAC (SMAC) [1] avoids idle listening by sending sensor nodes to sleep state periodically, if a node is not involved in any communication. The concept

This work is supported by the National Science Council of Taiwan partly under the grant NSC100-2221-E-182-069.

is similar to IEEE 802.11 power saving mode, in which each node wakes up at the beginning of each beacon interval to check if it needs to remain active or not. By keeping the duty cycle low, SMAC reduces the power consumption of each sensor node. However, SMAC still has some disadvantages. First, its low duty cycle may result in long transmission latency, and second, it fails to adapt to network traffic well because of its fixed duty cycle. If the duty cycle is determined according to the heavy-loaded node, a lot of energy will be wasted for light-loaded nodes. On the contrary, if the duty cycle is determined based on a light-loaded node, higher transmission latency is expected.

The Timeout-MAC (TMAC) [2], an improvement of SMAC, uses an adaptive duty cycle that a sensor node in listen mode will not go to sleep until there is no activity for a certain time T_A. TMAC finds a way to determine the active duration of the nodes. However, it still suffers from long transmission latency. For both SMAC and TMAC, all sensor nodes still have to wake up at every time frame and therefore they are not energy efficient as light-loaded nodes may remain idle in most cycles. In the Pattern-MAC (PMAC) [3], authors propose a method in which listen period of each sensor can be adjusted based on their own traffic load. Each sensor determines the predefined active or sleep mode based on its own traffic load. If the traffic load is heavy, the node will keep the state active for a long time, and to check whether the data needs to be transmitted at every time frame. In contrast, if the traffic load is light, the node will sleep for a long time and it does not have to check whether the data needs to transmit at every time frame. After sensors decide their predefined active/sleep modes, they will exchange this information to their neighbors and the active/sleep scheduling will be coordinated by the PMAC.

Quorum-based MAC (QMAC) [4], [5], [6] is different from the SMAC, TMAC and PMAC, which avoids waking up at every time frame and uses the concept of quorum. The quorum theory has been widely used in distributed systems, which provides mutual exclusion guarantees, fault tolerance, agreement, and voting. Here, this paper uses the quorum set to identify the time frame during which sensors must wake up. There are some kinds of quorum, such as grid-based [8], majority-based [9], and tree-based [10]. Without loss of generality, it uses the grid-based quorum to implement the protocol. In a grid-based quorum, one row and one column are picked from an $N \times N$ grid, while according to the property of quorum, it is guaranteed that any two sensors will meet at some time frames. As shown in Fig. 1, there are two intersections between sensors A and B. Out of those two intersections, one for R_a and C_b and another for C_a and R_b. Sensor A picks row R_a and column C_a as its quorum, while sensor B picks row R_b and column C_b.

As mentioned above, each sensor only needs to wake up at set time frames, and it can not only keeps the connectivity with its neighbors, but also reduces the times of the idle listening to achieve prolonging the network lifetime. According to the hop count between the sensor nodes with the sink, it can be classified to the corresponding corona, and the sensing data will be transmitted from the sensor node in the outside corona to the inside one. But, there are also some disadvantages in QMAC, as it must know the total traffic load of the network in advance, by using this traffic load information to set up the active/sleep scheduling of sensors, and sensors in the same corona will be set the same active/sleep scheduling and the same grid size of quorum. In this method, the most important problem is that the scheduling and grid size cannot be adjusted after the set. However, in real circumstances, the traffic will be changed dynamically by different events. Therefore, the traffic load of the sensors is not always the same in the same corona. Because of the QMAC sets, the static scheduling and grid size, initially which is not adapted to the real circumstance will result in increasing the energy consumption and latency of data transmission. For this reason, it is essential to design an energy efficient protocol based on dynamic traffic load, which is described in the next few sections of our work.

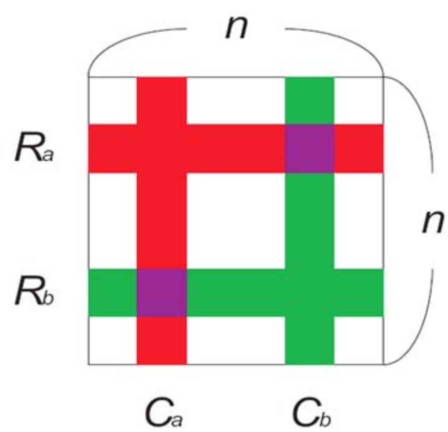


Fig. 1. The concept of quorum.

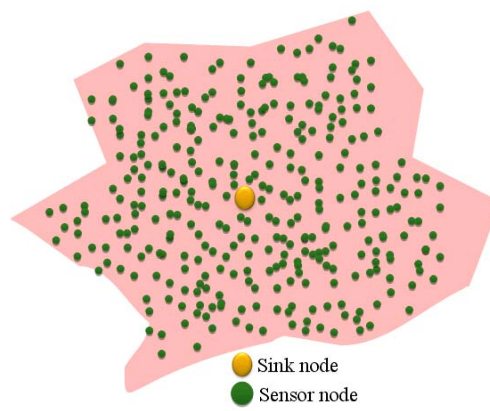


Fig. 2. Deployment of sensors over an irregular monitored region.

III. System Model

Let us consider a wireless sensor networks, in which nodes are deployed randomly and densely around a sink over an irregular monitored region as shown in Fig. 2. Sensor nodes are assumed to be static after deployment and have same communication, and sensing range. That is, throughout the work, it is considered that the communication range of the nodes is equal to its sensing range. Each node has the same initial energy and transmits data to the sink hop-by-hop. The total cycle time is divided into a series of time frames and all sensor nodes have the same cycle time. The data collected by the sensors within the same communication range are identical. A *Zone Dividing Algorithm* as proposed in this paper is used to divide the sensors into different zones. It starts from the sink node and divides the zones according to their communication range. After the *Zone Dividing Algorithm* is executed, the Routing Path Selection algorithm is also proposed. The path selection rule is designed based on the remaining energy of the next hop neighbors of a node.

Another important point in this work is designing an efficient scheduling protocol. It is assumed that all sensor nodes are time synchronized and have the same cycle time. In the proposed scheduling protocol, the cycle time is divided into a series of time frames to match the quorum and uses a simple conversion to calculate the size of the quorum. In real circumstances, the traffic load of sensors may be changed in different conditions. Hence, it is important to design a dynamic scheduling protocol. For this reason, the traffic load model is used to calculate the traffic load within an effective common communication range, where the data in that range can be regarded as identical and is transmitted to the inner zone. Finally, the traffic load is transferred to the suitable quorum size.

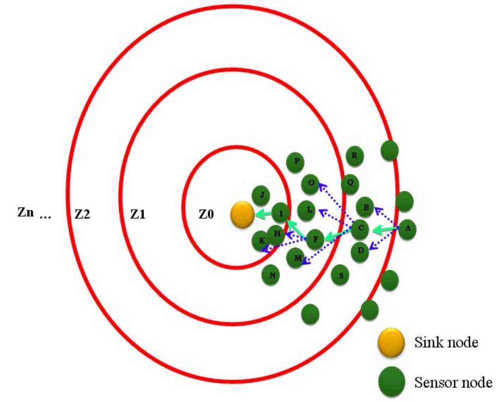


Fig. 3. Example of division of zones among the nodes of the whole networks.

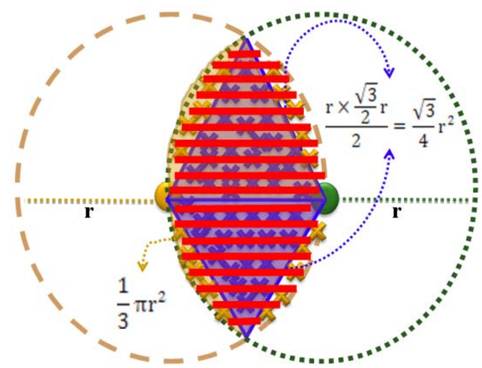


Fig. 4. The area of effective common communication range.

IV. Proposed Scheduling Protocol

The proposed scheduling protocol is divided into three parts. The first one is the *Zone Dividing* and *Routing Path Selection Algorithm*. Second one is the *Traffic Load Model*. The last one is the *Traffic load-based Scheduling Algorithm*. Different parts of the proposed algorithms are explained as follows.

A. Zone Dividing and Routing Path Selection Algorithm

In this subsection, it is assumed that the process of zone division starts from the sink. All sensors sensed within the communication range R_c of the sink will be assigned to Zone 0. The whole network is divided into N zones such that the distance between a sensor and the nodes in its next $(N-1)$-th zone is less than or equal to R_c as shown in Fig. 3. This algorithm has two parts. They are the zone division algorithm and routing path selection algorithm. In the zone division algorithm, the whole network can be regarded as concentric circles that are composed of coronas. The communication between the sensors and the sink is considered to be unidirectional as sensors are supposed to sense the region and should transmit data to the sink. The sensors that are belonged to zone N, only transmit data to the sensors with maximum residual energy in zone $(N-1)$. However, sensors in zone 0, transmit data to the sink directly as shown in Fig. 3.

For example, sensor A located in zone 3, should transmit data to the nodes of zone 2 having maximum residual energy. Accordingly, sensor A has to check the current energy level of sensors B, C, and D located in zone 2 and finally selects sensor C, which has the highest residual energy. By the same way, sensor I located in zone 0 will be selected to receive data from node F located in zone 1, which ultimately transmits data to the sink directly. Since, the proposed scheduling protocol is based on the traffic load, zone division is necessary to differentiate the nodes with variable data in different zones. Each sensor may have different neighbors and density of the nodes in different zones. The routing path is decided based on the existing residual energy of the next hop neighbors of a node.

B. Traffic Load Model

In this subsection a traffic load model is designed to calculate the amount of traffic generated by the sensors of different zones. Let us consider the distance ρ with hop counts HC_i between the sink and the nodes within effective common communication range as shown in Fig. 4. HC_i can be found out from the respective zone number of the nodes as the zones are numbered by their communication range.

$$\beta = \frac{\lambda}{D(HC_i)}\left(1+\frac{1}{2\pi(HC_i)r(n)}\int_{HC_i}^{1}\frac{2\pi(HC_i)}{D(HC_i)}D(HC_i)d(HC_i)\right)$$
$$= \frac{\lambda}{D(HC_i)}\left(1 + \frac{1}{(HC_i)r(n)}\int_{HC_i}^{1}HC_i d(HC_i)\right)$$

$$= \frac{\lambda}{D(HC_i)}(1 + \frac{1}{(HC_i)r(n)}(\frac{1}{2} - \frac{1}{2}(HC_i)^2))$$

$$= \frac{\lambda}{D(HC_i)}(1 + \frac{1}{2(HC_i)r(n)}(1 - (HC_i)^2)) \qquad (1)$$

Where, HC_i is the distance from the sink node to the sensor, and length of one hop is $0 < hop < R_c$. $D(HC_i)$ is the density of the nodes at $HC_i(nodes/m^2)$, and $\lambda/D(HC_i)$ is the data rate transmitted by each node. $r(n)$ is the thickness of the hypothetical corona. $\lambda/D(\rho)$ is the traffic generated by each sensor and β is the total traffic load.

C. Adaptive Traffic Load-based MAC (AT-MAC)

As discussed in our related work, most papers consider quorum-based MAC protocol based on the simulation results. They assume the possible traffic load without any theoretical analysis based on the node density. Then, they use the simulation based traffic load to decide the quorum based grid size and design the wake up or sleep schedule of the nodes. In real situations, the traffic load of each sensor could be changed. Hence, in this part we propose an Adaptive Traffic Load-based scheduling algorithm (AT-MAC), as given in Table I. In our protocol, the traffic load of the nodes is combined with the quorum theory, which can adaptively adjust the quorum grid size.

Since, each sensor has the same cycle time T and α as the maximum data rate, each sensor can know the maximum traffic αT that it can transmit in T. Besides, the data rate may not be always high and therefore the sensor is not required to wake up during all the cycle time as scheduling of the nodes can be adjusted dynamically in AT-MAC. Since, each sensor uses the relationship between $(\alpha.T)$ and β, the ratio γ can be found. Taking the $N \times N$ quorum grid size, each sensor has the ratio of $\frac{(2N-1)}{N^2}$ wake up time slots to receive and transmit data within T. It is to be noted that initially the ratio γ can be exchanged to $\frac{(2N-1)}{N^2}$ and derive an approximation N to set the quorum grid size. Finally, by the traffic load model, the parameter β may be changed, and each sensor will adjust quorum grid size dynamically by which energy consumption will be efficient to cope with different conditions of β.

As shown in Fig. 5(a), N is assumed to be 3 and the second row and first column are selected randomly. The corresponding cycle time T is shown in Fig. 5(b).

Similarly, as shown in Fig. 6(a), an example of different quorum size is taken, where sensors can communicate in different frames. As shown in Fig. 6(b), the first, fifth and seventh frame of Quorum *A* can communicate with Quorum *B* at its first, second, sixth and fourteenth frame. Quorum *B* can communicate with Quorum *C* except of the tenth frame.

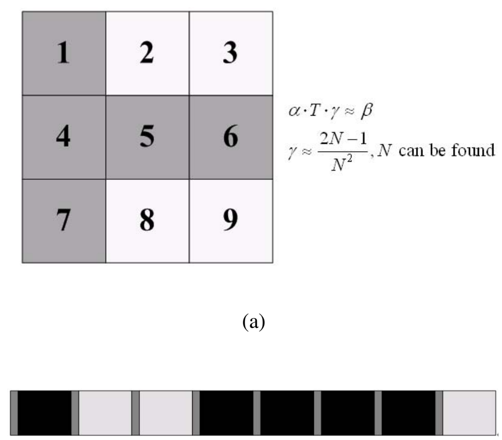
V. PERFORMANCE EVALUATION

In this section, we evaluate performance of our zone based AT-MAC algorithm as compared to AQEC [4] and QMAC_LR [5] algorithms through simulation. The detail description of the simulation setups and results are given as follows.

TABLE I
AT-MAC ALGORITHM

Notations:
$N \times N$: Quorum grid size, where N is a positive integer,
α: Maximum data rate of each sensor,
β: Traffic load of each sensor based on the traffic load model,
T: Cycle time of each sensor, which is same for all sensors,
γ: A parameter that represents the ratio,

STEP 1:
Initially, each sensor sets its own quorum grid size based on equation 1;
Total traffic load of each sensor is calculated as $\beta = \alpha.T.\gamma$;
STEP 2:
Ratio γ is calculated to find out quorum size N, i.e. $\gamma = (2N-1)/N^2$;
STEP 3:
The size N is decided to choose one row and one column;
That N represents the wake up time frame;
STEP 4:
Each sensor repeats this algorithm to adjust its grid size, if β is changed.



(b)

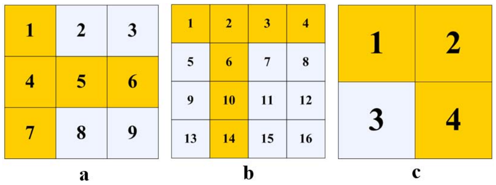
Fig. 5. Example showing execution of AT-MAC with N = 3.

A. Simulation Setups

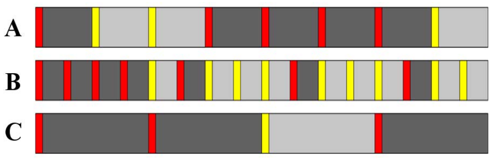
In our simulation test bed, a monitored area of size $300{\times}300m^2$ is setup. The numbers of deployed nodes over the monitored region are considered to be 200∼1000. Each node can generate 100bytes packet in every 5seconds and the channel capacity is considered to be 10Kbps. Initial energy of each sensor is taken to be 100Joule, and the energy consumption for each transmission is 0.5Joule.

B. Simulation Results

The simulation results are divided into two parts. The first part calculates the control packet overhead of zone dividing algorithm in different situations. In this case, sensors set the zone with different number of nodes and vice versa. The second part simulates the total bandwidth consumption when each sensor generates the packet and transmits it from the last zone to the sink. It also discusses the simulation result



(a)



(b)

Fig. 6. Example showing execution of AT-MAC, when N = 2,3 and 4.

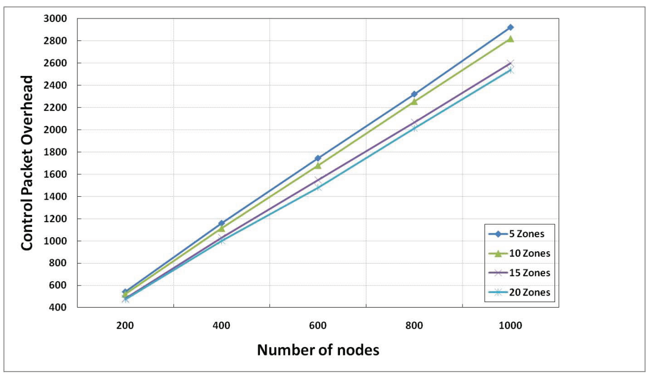


Fig. 7. Control packet overhead for setting different number of zones.

in terms of Constant Bit Rate (CBR) and Variable Bit Rate (VBR), which are compared with AQEC and QMAC_LR.

As shown in Fig. 7, control packet overhead is analyzed for setting of different number of zones with different number of nodes. In this simulation, control packet overhead increases when number of nodes increase. In the other hand, it increases with different number of nodes in the same zone. It is to be noted that each sensor sets zone and routing path hop-by-hop and transmits packet zone-by-zone in our zone dividing algorithm, which implies that the same number of nodes with fewer zones must communicate more times either in the same zone or in the next zone. For this reason, control packet overhead is inversely proportional to the number of zones. Fig. 8 shows the energy consumption for setting of zones with different number of nodes and zones. The energy consumption is related to the control packet overhead. It is observed that higher the control packet overhead is, more amount of energy is consumed. When the number of nodes is more than or equal to 800, the ratio of energy consumption is approximately equal to the same.

The simulation results of bandwidth consumption from the

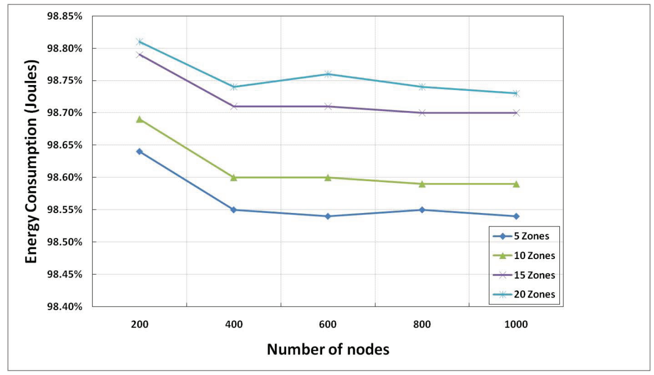


Fig. 8. Energy consumption for setting different zones.

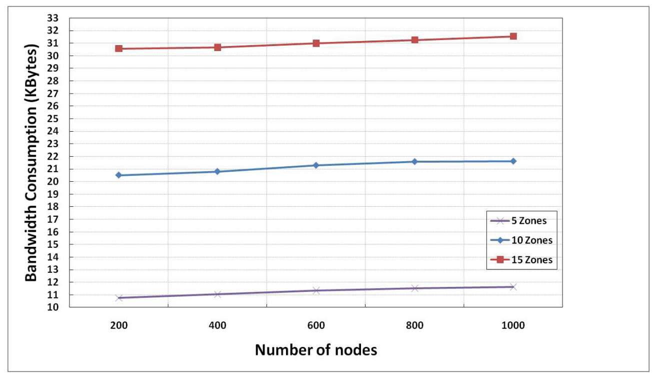


Fig. 9. Bandwidth consumption for transmitting data from the last zone to the sink with CBR.

last zone to the sink for constant bit rates (CBR) and for variable bit rates (VBR) are shown in Fig. 9 and Fig. 10, respectively. The sensor node in the last zone generates data packets and transmits them to the inner zone hop-by-hop in each 100 seconds of simulation time. These figures show the average of total bandwidth consumption in each zone except for the sink. From Fig. 9, it is found that the constant bit rate can adjust the quorum size by the fixed bit rate when the first round is terminated. As shown in Fig. 10, the bandwidth consumption is comparatively less for variable bit rates. Consumption of higher bandwidth occurs if number of zones are increased. This is due to an additional burden on bandwidth for diving the zones.

As shown in Fig. 11, our protocol is compared with AQEC and QMAC_LR in terms of control packet overhead. In this simulation, number of zones are fixed to be 15. It is observed that AQEC needs largest number of control packets as each sensor broadcasts the packets and receives them regularly except when they are not within the communication range. Since, QMAC_LR uses next hop group member to set which nodes can receive the packet in the inner zone, its control packet overhead is comparatively less. In our algorithm, the zone setting is done in a hop-by-hop manner, and each node chooses the next hop sensor based on its current residual energy. Hence, the control packet overhead in our protocol is comparatively less. As shown in Fig. 12, the simulation

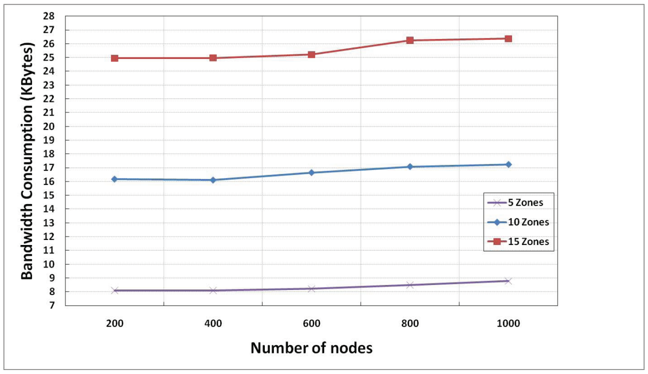


Fig. 10. Bandwidth consumption for transmitting data from the last zone to the sink with VBR.

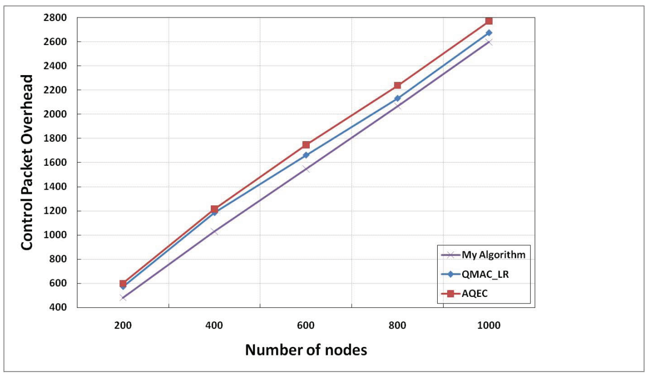


Fig. 11. Comparisons of control packet overhead for fixed number of zones.

of bandwidth consumption is compared with AQEC and QMAC_LR. In our proposed protocol, it is effective to adjust the quorum size for suitable load expect for the large number of nodes. AQEC and QMAC_LR use the fixed quorum size, which is decided by different corona. Therefore, when the traffic is light, a node can have large quorum size and has to wake up, which wastes a lot of energy. On the other hand, It will cause a lot of delay when the small quorum size confronts the heavy load.

VI. Conclusions

In this paper, a Zone Dividing and Routing Path Selection Algorithm is proposed for setting up different zones that decides which sensor node has higher energy to receive the data packet from the last zone to the sink. The proposed Adaptive Traffic load-based MAC (AT-MAC) protocol uses a traffic load model to combine with quorum. Each sensor can calculate its own total traffic load based on the node density of the network, communication range and hop counts from the sink to itself. Hence, a node can know how many time frames it should wake up and sleep. Based on the traffic load and quorum, the grid size can be adjusted dynamically, which implies that the active/sleep scheduling can be controlled efficiently. Based on the this result, the sensor network can cope with unexpected events easily and can schedule a node's active and sleep durations effectively to minimize the energy consumption.

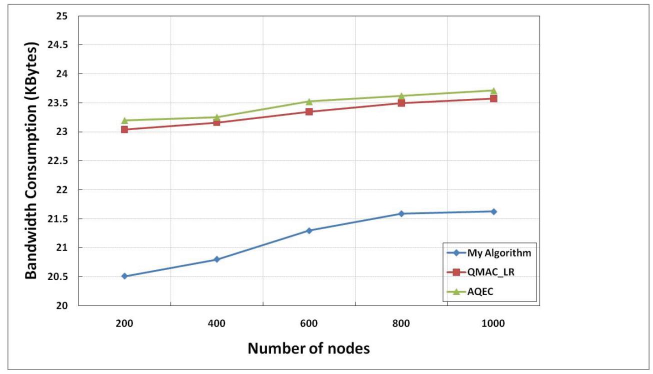


Fig. 12. Comparisons of bandwidth consumption transmitting data packet from the last zone to the Sink with CBR

References

[1] W. Ye, J. Heidemann and D. Estrin, "Medium access control with coordinated adaptive sleeping for wireless sensor networks", IEEE/ACM Transaction on Networking, Vol. 12, Issue 3, pp. 493-506, June. 2004.
[2] T.V. Dam and K. Langendoen, "An adaptive energy-efficient MAC protocol for wireless sensor networks", in Proceedings of the 1st International Conference on Embedded Networked Sensor Systems, Sensys'03, pp. 171-180, Los Angeles, Nov. 2003.
[3] T. Zheng, S. Radhakrishnan and V. Sarangan, "PMAC: An adaptive energy-efficient MAC protocol for wireless sensor networks", in Proceedings of the 19th IEEE International Parallel and Distributed Processing Symposium, pp. 65-72, Denver, April. 2005.
[4] C.M. Chao, J.P. Sheu and I.C. Chou, "An Adaptive Quorum-Based Energy Conserving Protocol for IEEE 802.11 Ad Hoc Networks", IEEE Transaction on Mobile Computing, Vol. 5, Issue 5, pp. 560-570, May. 2006.
[5] C.M. Chao and Y.W. Lee, "A Quorum-Based Energy-Saving MAC Protocol Design for Wireless Sensor Networks", IEEE Transaction on Vehicular Technology, Vol. 59, Issue 2, pp. 813-822, Feb. 2010.
[6] X. Yang; J. Liu; F. Yan, "A Variable grid-quorum pattern MAC protocol for Wireless Sensor Network", in proceedings of the 2010 International Conference on Computer, Mechatronics, Control and Electronic Engineering (CMCE), pp. 387-390, Oct. 2010.
[7] S. Olariu and I. Stojmenovic, "Data-centric protocols for wireless sensor networks", Wiley, 2005.
[8] S.Y. Cheung, M.H. Ammar and M. Ahamad, "The grid protocol: A high performance scheme for maintaining replicated data", IEEE Transaction on Knowledge and Data Engineering, Vol. 4, Issue 6, pp. 582-592, Dec. 1992.
[9] R.H. Thomas, "A majority consensus approach to concurrency control", ACM Transaction on Database Systems, Vol. 4, Issue 2, pp. 180-290, Jun. 1979.
[10] D. Agrawal and A.E. Abbadi, "An efficient and fault-tolerant solution for distributed mutual exclusion", ACM Transaction on Computer Systems, Vol. 9, Issue 1, pp. 1-20, Feb. 1991.