Adaptive Data Collection Algorithm for Wireless Sensor Networks

M. Vahabi, M. F. A. Rasid, R. S. A. R. Abdullah, and M. H. F. Ghazvini
Universiti Putra Malaysia, Serdang, Selangor, Malaysia

Summary
Periodical Data collection from unreachable remote terrain and then transmit information to a base station is one of the targeted application of sensor networks. The energy restriction of battery powered sensor nodes is a big challenge for this network as it is difficult or in some cases not feasible to change the power supply of motes. Therefore, in order to keep the networks operating for long time, efficient utilization of energy is considered with highest priority. In this paper we propose TA-PDC-MAC protocol - a traffic adaptive periodic data collection MAC which is designed in a TDMA fashion. This design is efficient in the ways that it assigns the time slots for nodes' activity due to their sampling rates in a collision avoidance manner. This ensures minimal consumption of network energy and makes a longer network lifetime, as well as it provides small end-to-end delay and packet loss ratio. Simulation results show that our protocol demonstrates up to 35% better performance than that of most recent protocol that proposed for this kind of application, in respect of energy consumption. Comparative analysis and simulation show that TA-PDC-MAC considerably gives a good compromise between energy efficiency and latency and packet loss rate.

Key words:
Wireless Sensor Networks, Periodic Monitoring, Sampling Rate.

1. Introduction

Wireless sensor networks are defined as low-cost, low-power networks consisting of radio nodes equipped with sensing devices. The low-cost equipment facilitates large deployments, recompensing for the extremely limited transmission range associated with the low transmission power. Wireless sensor networks have a multitude of applications, ranging from environmental to military domains. These applications include contamination tracking, habitat monitoring, health monitoring, traffic monitoring, building surveillance and monitoring, industrial and manufacturing automation, distributed robotics and enemy tracking in the battlefield.

Due to the vast variety of sensor applications, they should meet some special characteristics. Unlike traditional networks, sensor networks do not rely on the pre-existing infrastructure. This allows random deployment in inaccessible terrains or disaster relief operations. Hence, sensor network protocols must possess self-organizing capabilities. On the other hand, Researchers propose that sensor networks should be scaled to the size of the organisms under study, sample data at frequencies equivalent to physical phenomenon changes that organisms encounter, and deployed to capture the full range of the organism's exposure. Only sensor networks with these properties can provide the appropriate fine-grain information needed for accurate modeling and prediction [7, 8].

In light of common characteristic of many WSN application scenarios, that is monitoring some phenomena and relaying data toward the sink, two categories of data collection can be distinguished; *event-based reporting* of outliers and *periodic data collection* of key parameters [1]. In event-based data collection, the sensors are responsible for detecting and reporting events, Examples of event-based reporting include the localization of a sniper based on analyzing the sound of a gunshot [5], and the "A Line in the Sand" intrusion detection system [6]. This kind of data collection is less demanding in terms of the amount of wireless communication, since local filtering is performed at the sensor nodes, and only events are propagated to the base node.

In the second category of data collection, periodic updates are sent to the base node from the sensor network, based on the most recent information sensed from the physical parameter. Example of the periodic monitoring class include the observation of nesting patterns of storm petrels at Great Duck Island [2], measuring light intensities at various heights in a redwood tree [3] and logging temperature and humidity in the canopy of potato plants for precision agriculture [4]. This approach is also further classified into two; *query-based* data collection which also called continuous queries [9], and periodic reporting data collection. The first class is used to express user or application specific information interest and support aggregate queries, such as minimum, average, and maximum. These types of queries result in periodically generating an aggregate of the recent samples of all nodes. Similar to event-based data collection, the raw data is not extracted from the network and complex data analysis that requires integration of samples from various nodes at various times, cannot be performed in the destination, because of in-network aggregation.

The most comprehensive way of data collection is to extract raw samples from the network through periodic

Manuscript received June 5, 2008.
Manuscript revised June 20, 2008.

reporting of each sampled value from every sensor node. This scheme enables arbitrary data analysis at a sensor stream processing center once the data is collected. Such increased flexibility in data analysis comes at the cost of high energy consumption due to excessive communication and consequently decreases the network lifetime. On way of tackling this problem is to use data compression to reduce the total size of the data transmitted on the wireless channel. But as aforementioned, it will cause inflexibility on data analysis and also introducing delays which is undesirable for real-time applications [10]. In this study we focus on periodic reporting data collection and present a novel node scheduling scheme with respect to their traffic on an energy efficient manner.

We present an energy efficient traffic adaptive MAC protocol for environmental monitoring application for sensor networks with different packet intervals.

2. Related Works

Due to the specific energy constrained environment, MAC design for sensor networks generally has to take energy consumption as one of its primary concerns. There have been several MAC protocols specially designed for sensor networks. S-MAC [11] is the most renowned protocol proposed by Ye *et al.*. In S-MAC locally managed synchronization and periodic sleep/listen schedules based on these synchronizations form the basic idea behind it. Neighboring nodes form *virtual clusters* based on common sleep schedule to reduce control overhead and enable traffic-adaptive wake-up. S-MAC also includes the concept of message passing, in which long messages are divided into frames and sent in a burst to reduce contention latency.

The other protocol design for WSNs is D-MAC [12]. It is an energy efficient and low latency MAC protocol for tree-based data gathering in wireless sensor networks. The major traffic in WSNs is from sensor nodes to a sink which construct a data gathering tree. D-MAC utilizes this data gathering tree structure specific to sensor network applications to achieve both energy efficiency and low packet delivery latency. D-MAC staggers the sleep/listen schedule of the nodes in the data gathering tree according to its depth in the tree to allow continuous packet forwarding flow in which all nodes on the multi-hop path can be notified of the data delivery in progress and duty cycle adjustment command. Collision avoidance methods are not utilized in this protocol; when a number of nodes that have the same schedule (the same level in the tree) try to send to the same node, collision will occur.

T-MAC [13] is proposed to enhance the poor results of the S-MAC protocol under variable traffic loads. In T-MAC, the listen period ends when no activation event has occurred for a time threshold TA. The decision for TA is presented along with some solutions to the early sleeping problems. The activation time events include reception of any data, sensing of communication on the radio, the end-of-transmission of a node's own data packet or acknowledgement, etc.

TRAMA [14] is a TDMA-based algorithm proposed to increase the utilization of classical TDMA in an energy efficient manner. It is similar to Node Activation Multiple Access (NAMA) [5], in which for each time slot a distributed election algorithm is used to select one transmitter within each two-hop neighborhood. This kind of election eliminates the hidden terminal problem and hence ensures that all nodes in one-hop neighborhood of the transmitter will receive data without any collision. In this protocol higher percentage of sleep time and less collision probability are achieved, as compared to CSMA based protocols.

The most relevant work to this paper is the protocol designed by Erazo et al. [16]. It is a medium access control protocol for wireless sensor networks specialized for periodic data collection (PDC) applications. It focuses on reducing energy consumption in these applications. Specifically, it reduces the duty cycle of nodes lowering drastically idle listening in a TDMA manner. It also proposed a very simple method for synchronization of the nodes to follow their own sleep/listen schedules. In its next version [17] the route partitioning method was introduced to develop this idea for a large scale networks.

3. Design Overview

The study of PDC medium access control [16] shows its strong energy efficient feature in MAC protocol design for environmental monitoring applications, while, it supports just one sampling rate for all nodes in the network. This protocol cannot provide a fair data gathering system for the networks with different packet generation rates. Generally, time sampling rate of sensor nodes are not same, furthermore in all environmental monitoring applications, different sampling intervals due to the needs are required.

We present traffic adaptive PDC (TA-PDC) MAC to support networks with high and low sampling rates, which is still as energy efficient as the previous model. In this section, a brief outline of A-PDC-MAC protocol is given first, followed by the problem encountered in the existing protocol [17]. Then discuss about our proposed model to overcome the shortage.

TA-PDC-MAC includes approaches to reduce energy consumption by lowering the duty cycle as introduced in [11, 12, 13]. It also uses staggered time schedule for nodes to keep small delay in the entire network. The route partitioning method will nullify the hidden terminal problem without using the RTS/CTS packet exchange or the virtual carrier sense in similar way as done in [17].

In our protocol, the sink computes the time schedule of all nodes in the network with respect to their data generation rate. A node in the network can be involved in more than one route and should be capable of following different time schedule and duty cycles. In environmental monitoring applications it is desired to observe some physical variables more frequently than others. We propose a method to support sensor nodes with different sampling rates while keeping the energy efficiency as well.

The design goals of our modified protocol are less packet delay, lower energy consumption, and less data collision. It also benefits a simple unique centralized synchronization method. The sink is responsible among starting and maintaining the SYNC packets while other nodes only broadcast these packets in the area.

4. Implementation Method

Route partitioning has two states: initialization and maintenance. At the beginning of the initialization state sensor nodes dispatch route discovery packets (RD) to the sink. As RD packet is traversed through the network, the address and the desired sampling rate of each sensor node will be added to it. So these RD packets contain the complete route from each node to the destination together with data generation rate of that node.

When sink receives all RD packets from the network, it will run the proposed algorithm to separate the routes into different groups due to node's data generation rate. Then it will check the times in which these sampling rates overlap each other, if there is any time then nodes in that route should follow their longer duty cycle (DC) to achieve more energy efficiency. Table 1 illustrates the algorithm of this method.

Figure 1. illustrates an example to make a better sense of the proposed algorithm. Here are two nodes with higher sampling rates than other nodes in the network. These nodes don't have same DCs. After receiving all RD packets sink performs its route partitioning procedure and makes the "main route" matrix object for future use. As the sink is aware of all nodes' sampling rates, it will make other routes' objects due to the existing DCs. Each path that has been established in the "high traffic route" will be originated by the sink node and ended by the high traffic generation node. So it will be a sub-path of the "main route" object. Then, sink checks the overlap time between higher DCs and the lower one in the network. If there is any, the route with lower DC will active during that time. In this example, routes 6 and 7 include higher DC nodes. As these two routes are sub-paths of main route, they don't need to be active concurrently with main routes' activation time schedule.

TABLE I
Traffic adaption algorithm

```
Receive (RD - packets)
Calculate - no - of - DC - in - network
create main - routes        / / object for all main routes in network
for i = 2 to No.of.DC {
   create routes              // object for saving routes with same DC
   eliminate - Subpath - in - all - routes
}
for j = 2 to No.of.DC {
     p = (main - route.DC)/ (roue[j].D C)
     if ( p ∈ N )     // time which two different DC route overlap
         route[j]. Inactive Time = p
   }
}
```

There are three main communication patterns in sensor network applications. The first involves local data exchange and aggregation purely among nearby nodes. The second involves the dispatch of control packets and interest packets from the sink to sensor nodes. The last and the most significant traffic pattern in WSN is data gathering from sensor nodes to the sink. The proposed wakeup schedule is considered in the third type communication pattern, and the control packets have their own distinct slot times.

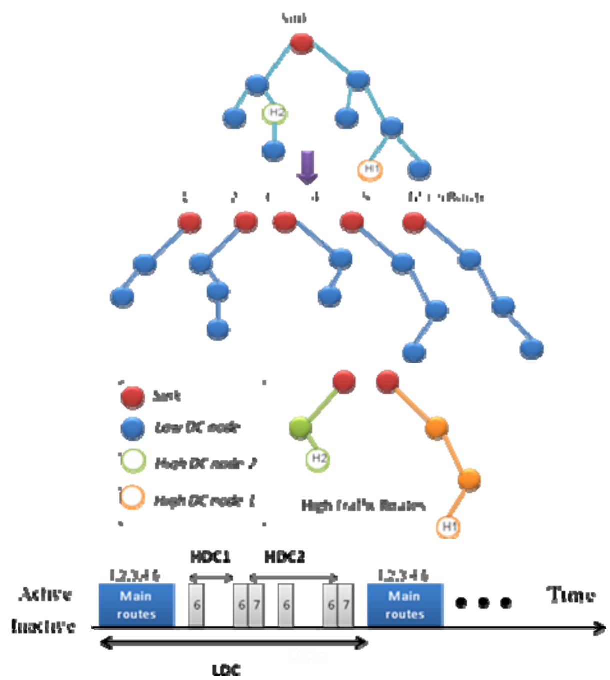

Fig. 1 An example of route partitioning and traffic adaptation.

There is more than one duty cycle in the network. Our proposed traffic adaption method divides routes into two groups, “main route” and “high traffic routes”. The time schedule of each node participated in these two groups will determine by proposed algorithm shown in table 3. After receiving the SD packet, each node finds its contribution in each group of routes. SD packet includes the DC of the route, the addresses of nodes in that route, inactive time, route size, route type and the time when the route should be activated (TS). So each node can find its position in the current route. For the case that a node is located at the main route, the time schedule of transmission and reception period will be calculated by the following formulas as in [17]. When nodes in the main route get closer to the sink, their receiving and transmitting periods get larger. This happens because the closer the node is to the sink, the more data it receives from other node and the more packets will be relayed. Table 2 summarizes the terms used in the algorithm and the given equations.

$$t_{TX,i,j} = t_B L_{Data}(|R_j| - i + 1) \quad (1)$$

$$t_{RX,i,j} = t_B L_{Data}(|R_j| - i) \quad (2)$$

TABLE II
List of symbols

Symbol	Quantity
$t_{TX,i,j}$	Duration of transmitting slot of node i within route j
$t_{RX,i,j}$	Duration of receiving slot of node i within route j
$\|R_j\|$	Size of route j
t_B	Time to TX/RX a byte
L_{Data}	Data packet length

If a node is located in any of the high traffic routes, it is responsible to forward just one data packet with higher DC. In fact it should forward the packet that has been generated by higher data generation node. So there is no need to consider any growth in transmission and reception periods of the nodes nearby the sink. Figure 2 shows this simple staggered time schedule for a chain including four nodes.

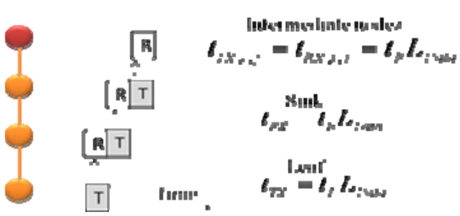

Fig. 2 Time schedule for nodes in high traffic route

TABLE III
Time schedule algorithm

Receive (SD - packet)
If(packet.destination_address == myAddress)
i = 1
While(Rj[i] ≠ myAddress)
i + +
$myPositionfromSink = i$
samplingperiod = packet.DC
if (packet.mainroute == 1){
$\Phi_{myPositionfromSink,j} =$
$compute\_my\_offset\_in\_this\_route(myPositionfromSink, sizeof(R_j[]))$
$t_{TX,myAddress,j} = t_B L_{Data}(sizeof(R_j[]) - myPositionfromSink + 1)$
$t_{RX,myAddress,j} = t_B L_{Data}(sizeof(R_j[]) - myPositionfromSink)$}
else{
if $(sizeof(R_j[]) == myPositionfromSink)$
$t_{RX,myAddress,j} = 0$
else
$t_{RX,myAddress,j} = t_B L_{Data}$
$t_{TX,myAddress,j} = t_B L_{Data}$
if $(myPositionfromSink \geq ([sizeof(R_j[])] - 1)$
$\Phi_{myPositionfromSink,j} = 0$
$\Phi_{myPositionfromSink,j} = (sizeof(R_j[]) - myPositionfromSink - 1) \times t_B L_{Data}$
inactivePeriod = packet.InactiveTime
}
ActivationTime = packet.TS - rxDelay + $\Phi_{myPositionfromSink,j}$

5. Energy Analysis

We consider a network with j paths in main route and N nodes in which M number of nodes are high DC. The expected energy consumption on a node is sum of energies that a node spends in each listen/sleep state. Table 4 summarizes the terms which we have used in the equations and the rest of symbols are as same as what has been defined in table II. As there are different duty cycles in network, we derive the expected energy spent during T_o period as follow:

$$E_T = \left[\frac{T_o}{T_{LDC}}\left(\sum_{j=1}^{N}\sum_{i=0}^{|R_j|} E_{i,j}\right)\right] + \left[\frac{T_o}{T_{HDC}}\left(\sum_{j=1}^{M}\sum_{i=0}^{|R'_j|} E'_{i,j}\right)\right] + X \quad (3)$$

Where X is defined as follow:

$$X = \begin{cases} -\sum_{j=1}^{M}\sum_{i=0}^{|R'_j|} E'_{i,j} & \frac{T_{LDC}}{T_{HDC}} \in N \\ 0 & \frac{T_{LDC}}{T_{HDC}} \notin N \quad \text{in overlap time} \end{cases}$$

If there is no overlap time between sampling rates the “X” value will be zero, otherwise it has other value. We focus on data gathering part of communication, and don't consider the energy that spends in other parts of initialization and maintenance phase.

TABLE IV
Symbols used in energy analysis

Symbol	Quantity
$E_{i.j}$	Energy consumed in node i within path j of main route.
$E'_{i.j}$	Energy consumed in node i within path j of high traffic route.
E_T	Total energy consumption
T_o	Period of observation
P_{RX}	Power spent in receive mode
P_{TX}	Power spent in transmit mode
P_{SP}	Power spent in sleep mode
T_{LDC}	Data inter-arrival time of low DC nodes
T_{HDC}	Data inter-arrival time of high DC nodes
$t_{sp,i,j}$	Duration of sleeping slot of node i within route j

The expected energy consumption on each node is sum of the energy spends in transmitting, receiving and sleeping mode. So,

$$E_{i,j} = E_{TX,i,j} + E_{RX,i,j} + E_{SP,i,j}$$
$$= P_{TX}.t_{TX,i,j} + P_{RX}.t_{RX,i,j} + P_{SP}.t_{SP,i,j} \quad (4)$$

$$t_{sp,i,j} = T_X - t_{RX,i,j} - t_{TX,i,j} \quad (5)$$

Where T_X is equal to T_{LDC} for low DC nodes, and is equal to T_{HDC} for high DC nodes. Sleeping time is not constant and depends on the location of node in a path, route type, and number of routes that the node is involved. Since P_{SP} is significantly less than P_{TX} or P_{RX}, we neglect the energy spent on sleep state. By replacing equations (1) and (2) in (4) and ignoring the sleep part we can determine the energy consumption on intermediate nodes in a route j of main route paths

$$E_{i,j} = P_{TX}.t_B.L_{Data}.(|R_j| - i + 1) + P_{RX}.t_B.L_{Data}.(|R_j| - i) \quad (6)$$

The energy consumed in sink and the last leaf on the path can be calculated as follow:

$$E_{Sink} = P_{RX}.t_B.L_{Data}.|R_j| \quad (7)$$

$$E_{|R_j|} = P_{TX}.t_B.L_{Data} \quad (8)$$

By adding equations (6), (7) and (8) the first term of equation (3) will be equal to:

$$\frac{T_o}{T_{LDC}} \sum_{j=1}^{N} \left(\left(\sum_{i=1}^{|R_j|-1} E_{i,j} \right) + E_{\sin k} + E_{|R_j|} \right)$$

(9)

The expected energy consumption of this term is equal to:

$$\frac{T_o N}{2T_{LDC}}.t_B.L_{Data}.\left(\delta^2_{|R_j|} + \left(\overline{|R_j|}\right)^2 + \overline{|R_j|} \right)(P_{TX} + P_{RX}) \quad (10)$$

Where, $\overline{|R_j|}$ is the mean value of all route lengths in the network with the variance value of $\delta^2_{|R_j|}$.

The second term of equation (3) shows the energy spending on data gathering in high traffic route. As mentioned before, in this part, all intermediate nodes are responsible for forwarding high DC nodes' data and act like a router. So each timeslot of transmission or reception just lasts for the duration of sending or receiving one data packet. Then it can be simply defined that expected total energy consumption in this part is equal to:

$$\frac{T_o M}{T_{HDC}}.t_B.L_{Data}.(P_{TX} + P_{RX}).|R_j| \quad (11)$$

So, the average value of energy consumption in the network should be as below:

$$E[E_T] = t_B.L_{Data}.(P_{TX} + P_{RX}) \times$$
$$\left[\frac{T_o N}{2T_{LDC}} \times \left(\delta^2_{|R_j|} + \left(\overline{|R_j|}\right)^2 + \overline{|R_j|} \right) + \frac{T_o M}{T_{HDC}} \times \overline{|R_j|} \right] + \overline{X} \quad (12)$$

As indicated before, "X" is a factor to eliminate the second part of equation (3) if there is an overlap time between sampling rates.

6. Simulation and Results

We implemented our prototype design with ns-2 network simulator [18]. Two different scenarios have been studied. Both of them include 10 nodes of sensors. The first scenario has one high data generation rate node while the second one has two nodes located in different routes. We consider just one sink node that all data should be gathered in it. Nodes are distributed randomly in a 500×500 m^2 area. Figure 3. shows our network simulation configuration.

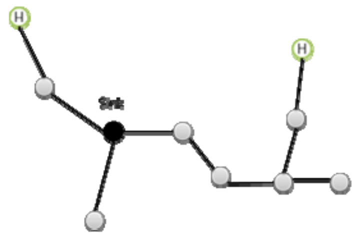

Fig. 3 Network topology

In this section, the maximum number of high DC nodes that can be supported by the proposed model is discussed. We have run the simulation for two different scenarios with same network configuration as shown in figure 2. In first scenario we assume that all nodes in the network are working with 100 seconds inter-arrival time and high DC nodes are working at 30 seconds inter-arrival time, whereas, in the second scenario the inter-arrival time for high DC nodes is equal to 60 seconds. The number of high DC nodes is incremented by one in each turn of running the simulation to observe the network behavior.

As it is shows in figure 4, the maximum number of high DC nodes that can be supported by TA-PDC-MAC is equal to 8. If all nodes work with high DC inter-arrival time, the proposed model is not suited, because of some extra overhead introduced by our model. The trend of increasing energy consumption between two consecutive point in the graph depends on the size of route that connects next high DC node to the sink, $|R_j|$. It is apparent that for large amount of $|R_j|$, the energy usage is high, because more nodes are involved in the route.

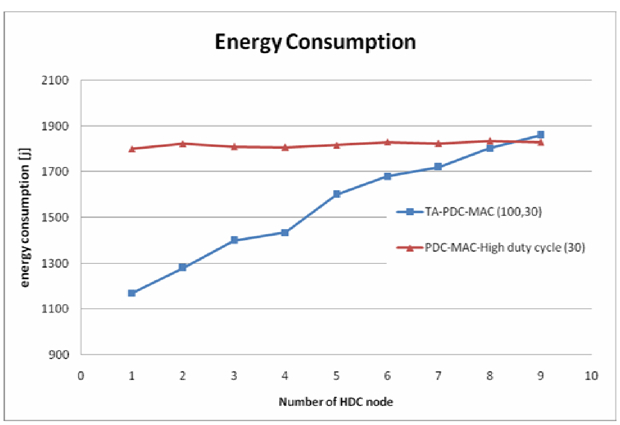

Fig. 4 Energy consumption with respect to number of high DC nodes, with 30 and 100 seconds inter-arrival times.

In the second scenario, we increase the number of high DC nodes by one. Figure 5 shows that the maximum number of high DC nodes that can be supported, is equal to 4. The locations of high DC nodes affect the number of them. For selecting the next high DC node, we consider the average route length ($\overline{|R_j|}$) of them equal to 3 for both scenarios. On the other hand, it can be revealed that difference of high and low duty cycles has a huge impact on the optimum number of high DC nodes that can be supported by the proposed protocol. For larger difference between the existing high and low DCs in the network, the number of high DC nodes that can be supported will be more. With the same mean path length, in the first scenario, 90% of sources can work with high duty cycle, whereas in second one up to 40% can be supported. The maximum value of energy consumption improvement for the first scenario is around 35%, and this value is equal to 15% for the second one.

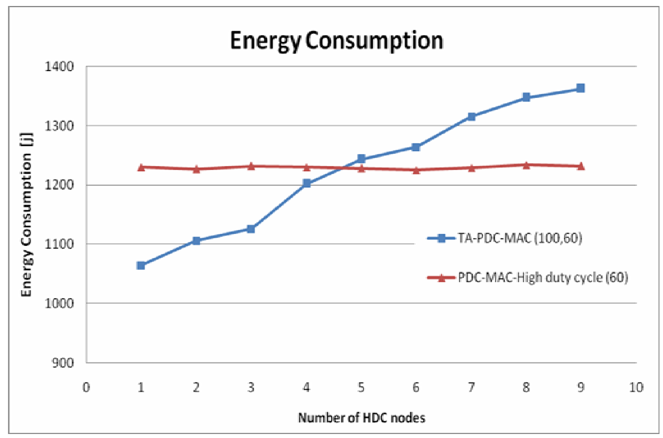

Fig. 5 Energy consumption with respect to number of high DC nodes, with 60 and 100 seconds inter-arrival times.

By considering first term of equation (3) of part D in section III, and replacing T_{HDC} instead of T_{LDC}, that give us the energy consumption of main route, and let it be equal to equation (12); the number of high DC nodes can be found. The energy consumed by PDC-MAC that works with high duty cycle will be:

$$\frac{T_o N}{2T_{HDC}}.t_B.L_{Data}.\left(\delta^2_{|R_j|} + \left(\overline{|R_j|}\right)^2 + \overline{|R_j|}\right)(P_{TX} + P_{RX}) \qquad (13)$$

By making this equation equal to the last introduced formula in (12), the number of high DC nodes will be as follow:

$$M = \frac{N.\left(\delta^2_{|R_j|} + \left(\overline{|R_j|}\right)^2 + \overline{|R_j|}\right)}{2\overline{|R'_j|}}.\left(\frac{T_{LDC} - T_{HDC}}{T_{LDC}}\right) \qquad (14)$$

Where, $\overline{|R'_j|}$ is the mean route length of high traffic routes and depends on the location of high DC nodes in the network.

Figure 6. illustrates the packet delay for both scenarios. The inter-arrival time is set to 100 seconds for low DC

nodes whereas, this value is equal to 30 and 60 seconds for high DC nodes in first and second scenario respectively. We compare our model with PDC-MAC protocol that works with low duty cycle (DC=100). It is clearly observed that our protocol outperform the PDC-MAC in both scenarios. The packet delay increases linearly with the number of high DC nodes in PDC-MAC design. The reason is that each packet has to wait in sender's buffer for one sleep cycle in each high DC node. It is interesting to note that for larger difference between high and low duty cycles in the network, the slope of increasing packet delay will be higher. This can be explained by the fact that for the large difference, time packets should wait in buffer for next activation cycle is more; since they should wait longer at sender to be collected by the network.

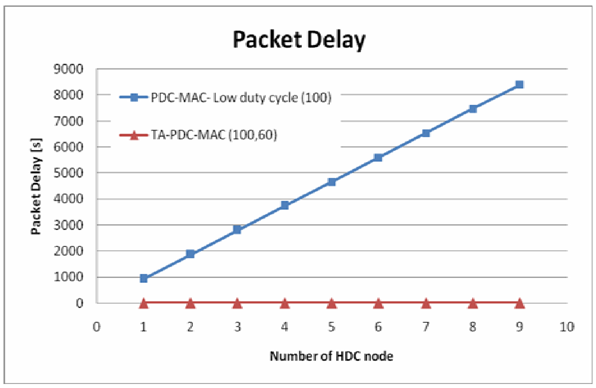

Fig. 6 Packet delay with respect to number of high DC nodes, for scenarios (60,100) and (30,100) seconds inter-arrival times.

Figure 7. shows the packet loss rate of both scenarios. This graph indicates that our model is far suited than PDC-MAC protocol in its low duty cycle mode. By increasing number of high DC nodes in network the amount of packet loss rate will increase drastically in PDC-MAC, whereas, it is kept in the acceptable level by the proposed model. Also it is obvious that for the large difference between high and low duty cycles, the value of packet loss will be increased by the PDC-MAC, where it doesn't have any effect on our model.

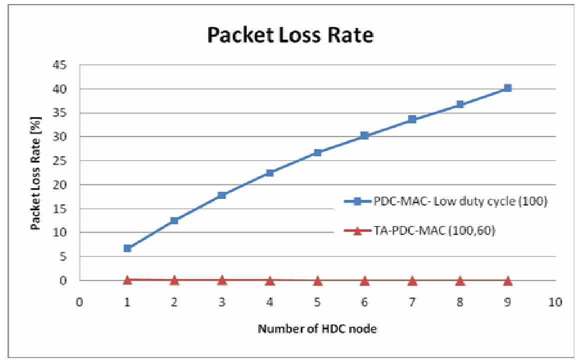

Fig. 7 Packet loss rate with respect to number of high DC nodes, for scenarios (60,100) and (30,100) seconds inter-arrival times.

7. Conclusion

In this paper the Traffic Adaptive PDC-MAC protocol is designed. The proposed model focuses on energy efficiency for environmental monitoring applications with different data generation rate nodes. Reducing packet latency and packet loss rate are the second issues of this work. By eliminating idle listening, considerable amount of energy can be saved. This will be achieved by scheduling the activation time of each node with respect to the generation time of packets. For supporting more than one sampling rate in the network and keeping the desirable amount of energy consumption, we introduced a traffic adaption algorithm. This algorithm makes the protocol to tolerate with different traffic generation rate. Without this method the amount of energy consumption and also packet delay and packet loss rate for networks under different traffic generation rates, will be very high. The simulation results show improvement on packet loss rate and packet delay with reasonable energy consumption level.

References

[1] Preprint of a book chapter in "Medium Access Control in Wireless Networks, Volume II: Practice and Standards" edited by H.Wu and Y. Pan, to be published by Nova Science Publishers in 2007.

[2] A.Mainwaring, J.Polastre, R. Szewczyk, D. Culler, and J. Anderson, *Wireless sensor networks for habitat monitoring*, in first ACM Int. Workshop on Wireless Sensor Networks and Application (WSNA), Atlanta, GA, Sept. 2002, pp. 89-97.

[3] G. Tolle, J. Polastre, R. Szewczyk, D. Culler, N. Turner, K. Tu, S. Burgess, T. Dawson, P. Buonadonna, D. Gay, and W. Hong, *A macroscope in the redwoods*, in 3rd ACM Conf. on Embedded Networked Sensor Systems (SenSys 2005), San Diego, CA, 2005, pp. 51-63.

[4] D. Goense, J. Thelen, and K. Langendoen, *Wireless sensor networks for precise Phytophthora decision support*, in 5th European Conference on Precision Agriculture (5ECPA), Uppsala, Sweden, June 2005.

[5] G. Simon, M. Maroti, A. Ledeczi, G. Balogh, B. Kusy, A. Nadas, G. Pap, J. Sallai, and K. Frampton, *Sensor network-based countersniper system*, in 2nd ACM Conf. on Embedded Networked Sensor Systems (SenSys 2004), Blatimore, MD, Nov. 2004, pp.1-12.

[6] A. Arora, P. Dutta, S. Bapat, V. Kulathumani, H. Zhang, V. Naik, V. Mittal, H. Cao, M. Demirbas, M. Gouda, Y. Choi, T. Herman, S. Kulkarni, U. Arumugam, M. Nesterenko, A. Vora, and M. Miyashita, *A line in the sand: a wireless sensor network for target detection, classification, and tracking*, Computer Networks, 46 (2004), pp. 605-634.

[7] M. Toapanta, J. Funderburk, and D. Chellemi, Development of Frankliniella species (Thysanoptera: Thripidae) in relation to microclimatic temperature in vetch, Entomological Science, 36 (2001), pp. 426-437.

[8] D. Happold. The subalpine climate at smiggin holes,Kosciusko National Park, Australia, and its influence on the biology of small mammals, Arctic & Alpine Research, 30 (1998), pp. 241-251.

[9] L. Liu, C. Pu, and W. Tang, *Continual queries for internet scale event-driven information delivery*, IEEE Knowledge and Dtat Engineering, 11 (1999), pp. 610-628.

[10] T. Arici, B. Gedik, Y. Altunbasak, and L. Liu, *PINCO: A pipelined in-network compression scheme for data collection in wireless sensor networks*, International Conference on Computer Communications and Networks, (2003) pp. 539-544.

[11] W. Ye, J. Heidemann, D. Estrin, "Medium Access Control With Coordinated Adaptive Sleeping for Wireless Sensor Networks", IEEE/ACM Transactions on Networking, Volume: 12, Issue: 3, pp. 493-506, June 2004.

[12] G. Lu, B. Krishnamachari, C. S. Raghavendra, "An Adaptive Energy-Efficient and Low-Latency MAC for Data Gathering in Wireless Sensor Networks", Proceedings of IPDPS'2004, April 2004.

[13] T. van Dam and K. Langendoen, "An adaptive energy-efficient mac protocol for wireless sensor networks", Proceedings of the First ACM Conference on Embedded Networked Sensor Systems, pp. 171–180, Los Angeles, California, USA, November 2003.

[14] V. Rajendran, K. Obraczka, J.J. Garcia-Luna-Aceves, "Energy- Efficient, Collision-Free Medium Access Control for Wireless Sensor Networks", *Proc. ACM SenSys 03*, pp. 181-192, Los Angeles, California, 5-7 November 2003.

[15] L. Bao and J.J. Garcia-Luna-Aceves, "A New Approach To Channel Access Scheduling For Ad Hoc Networks", *Seventh Annual International Conference on Mobile Computing and Networking*, pp. 210–221, 2001.

[16] M. Erazo, Y. Qian, "SEA-MAC: Simple Energy Aware MAC Protocol for Wireless Sensor Networks for Environmental Monitoring", Proceedings of ISWPC'2007, San Juan, PR, February 5-7, 2007.

[17] M. Erazo, Y. Qian, K. Lu, D. Rodriguez "Analysis and Design of a MAC Protocol for Wireless Sensor Networks with Periodic Monitoring Applications", *Milcom 2007*.

[18] The ns-2 simulator: http://www.isi.edu/nsnam/ns/