An Energy-Efficient, Application-Oriented Control Algorithm for MAC Protocols in WSN

Deliang Li, Fei Peng, and Depei Qian

Beihang University,
Haidian. 37, 100191 Beijing, China
busyterm@yahoo.com, pengfeius@gmail.com, depeiq@buaa.edu.cn

Abstract. Energy efficiency has been a main concern in wireless sensor networks where Medium Access Control (MAC) protocol plays an important role. However, current MAC protocols designed for energy saving have seldom considered multiple applications coexisting in WSN with variation of traffic load dynamics and different QoS requirements. In this paper, we propose an adaptive control algorithm at MAC layer to promote energy efficiency. We focus on the tradeoff relation between collisions and control overhead as a reflection of traffic load and propose to balance the tradeoff under the constraints of QoS options. We integrate the algorithm into S-MAC and verify it through NS-2 platform. The results demonstrate the algorithm achieves observable improvement in energy performance while meeting QoS requirement for different coexisting applications in comparison with S-MAC.

Keywords: MAC, tradeoff, collisions, control packets, multi-applications, traffic load, QoS, packet loss.

1 Introduction

In Wireless Sensor Network (WSN), reducing energy consumption of each sensor node is one of the important issues to prolong network lifetime due to the limited amount of energy operate in sensor node. The energy consumption comes from three aspects: communication, computation, and sensing, among which energy consumed on the communication part is a critical portion. As one of the major communication layer protocols, Medium Access Control (MAC) is therefore a primary concern on designing an energy-efficient MAC protocol in WSN.

It has been identified in [1] that there are four major sources of energy wastes at MAC layer for WSN: idle listening, collision, overhearing and control overhead, which have been the current research focus proposed for the power-saving MAC protocols. However, the investigation on the inherent tradeoff between collisions and control overhead is usually ignored. It is easy to see that the energy waste due to control overhead and collision highly depend on each other. Traditional collision avoidance prevents energy waste due to collisions, but leads to high control packet overhead; whereas using less robust mechanisms for collision handling typically reduces control packet overhead, but will lead to more energy waste due to an

J. Zheng et al. (Eds.): ADHOCNETS 2009, LNICST 28, pp. 805–817, 2010.

increased number of collisions. For instance, in contention-based MAC protocols, CSMA/CA employs RTS/CTS/DATA/ACK handshake mechanism for reducing collisions among competing nodes; In TDMA-based MAC protocols, to setup a collision-free schedule, two-hop information should be exchanged at the beginning of each frame.

With the evolvement of the WSN applications, the applications take on diversity and integration. Each type of applications shows different traffic characteristics and requires for different QoS performance. Nowadays in WSN, multiple applications co-exist and the traffic load integrating traffic characteristics of various applications in the network presents spatio-temporal variation. However, most of MAC protocols only focused on one type of applications and proposed energy-saving techniques only suitable for specific traffic load condition.

It is reasonable that application-specific QoS requirements and traffic load have great impact on energy saving. First, traffic load has a close relation with the collision and control overhead tradeoff. Collisions become rare under light traffic load, but significant under heavy traffic load. Hence, light traffic condition takes role in tradeoff more important than heavy traffic condition. In this way, power can be saved by expanding energy on control packets only when needed under a variety of traffic load. Second, applications QoS requirements also have great impact on the tradeoff. Applications with loose QoS constrains (i.e. packet losses) could provide larger design space for adopting aggressive energy saving technologies such as minimizing/eliminating MAC layer handshake than applications with stringent QoS limits. Therefore, MAC protocol should be optimized for many applications to balance collision and control overhead trade-offs. Energy may be adaptively conserved with the permission of the application-specific QoS and the network load.

This motivates us to propose an adaptive control algorithm based on the collision and control overhead tradeoff, with the adjustment of the required QoS and the variation of traffic load to enhance MAC energy efficiency. In this paper, we mainly focus on the design of contention-based MAC protocols based on CSMA/CA. Since the collision and control overhead tradeoff also exists in TDMA-based MAC, our scheme could be extended to other types of MAC, which is a subject of our future work.

In this paper, our contributions are:

1) We focus on the multi-applications WSN, in which multiple types of applications co-exist.
2) To promote the energy efficiency of MAC protocols, we propose a scheme that employs application-specific QoS requirements and traffic load to balance the tradeoff between collisions and control overhead, which has received little attention.
3) We build an energy model to acquire a balance point for the collision-control overhead tradeoff applicable for each application. The balance point is expressed by the collision loss rate as a reflection to the variation of traffic load as for per application-based. We define the control threshold as the minimum value between the balance point and QoS requirement so that the control packets should be adaptively transmitted according to the control threshold for the purpose of power saving as well as QoS guarantee. Each application can have its packet loss rate upon collision measured in the simulation easily and get its balance point derived by the loss rate due to collision accordingly.

4) We also do simulation experiments in which we integrate the algorithm into S-MAC named as improved S-MAC (ISMAC) and compare it with S-MAC on energy performance. The results demonstrate that the algorithm could help promote the energy efficiency and maintain QoS performance required by different applications coexist.

The remainder is organized as follows: Section II introduces recent works on MAC protocols in WSN. Section III discusses the adaptive algorithm for control packets. In Section IV we presents the simulation and performance comparison. Section V concludes the paper.

2 Related Work

According to channel access policies, the MAC protocols in WSN are generally categorized into three classes: contention-based, TDMA-based, and hybrid MAC layer protocols. Contention-based MAC is based on the idea that when one node needs to send data it will compete for wireless channel. It is therefore simple to implement and has good scalability. TDMA-based MAC protocols [8] have a natural advantage of collision-free medium access. However, they suffer heavily from problems like clock synchronization, channel under-utilization and fixed time-slot assignments. Therefore, TDMA-based MAC has poor performance at varying traffic loads. Hybrid MAC [9] combines the advantages of contention-based MAC with that of TDMA-based MAC thus it has good adaptability to traffic variation. However, hybrid MAC protocols are usually very complex and need complicated scheduling algorithms which waste much energy. Therefore, in this paper, we mainly focus on contention MAC protocols.

Contention-based MAC is generally based on CSMA/CA or CSMA to compete for the channel before transmitting data. It could be classified into synchronous and asynchronous modes. Typical synchronized contention-based MAC protocols include [2]-[4]. S-MAC [2] adopts a periodic listen-sleep cycle to reduce idle listening and employs RTS/CTS/DATA/ACK handshake mechanism similar to IEEE 802.11 to avoid collisions and overhearing. However, the duration of the active and sleep period is fixed so that it increases idle listening under low traffic load. T-MAC [3] enhances the poor performance produced by S-MAC under variable traffic load by introducing an adaptive duty cycle. However, due to the periodical listen-sleep cycles adopted in [2] and [3], both SMAC and TMAC are applicable for periodical data gathering applications. Different from [2] and [3], [4]-[7] are asynchronous mechanisms. PS-MAC [4] was proposed in which each node determines ‘listen’ or ‘sleep’ pseudo-randomly based on its own pre-wakeup probability and pre-wakeup probabilities of its neighbor nodes in each time slot. However, the optimal pre-wakeup probability of each node based on the congestion level of each node and the network topology has not been studied. B-MAC [5] uses long-length preambles to reduce duty cycle and minimize idle listening time, which results in the energy waste in sender and receiver. And overhearing in neighbor nodes of the sender will arise. Due to long preambles and asynchrization, B-MAC is applicable for non-real time application. WiseMAC [6] dynamically adjusts the length of preambles by piggybacking the wake-up time in latest ACK packet which shortens the length of preambles. However, WiseMAC uses non-persistent CSMA to reduce idle listening so the hidden terminal problem still exists. It is suitable for query-driven application. X-MAC [7] further shortens the length of preambles by using strobed preambles. However, when the traffic load varies with time, energy waste on idle listening, collisions and overhearing is still exists. It is usually adopted in event-driven application.

At present each MAC protocol is generally suitable for one specific type of applications and they don't take the multiple applications co-existing in the network into consideration. In addition, these MAC protocols mainly focused on idle listening, collisions, overhearing, and control overhead while paying little attention to the tradeoff between collisions and control overhead. Until now the researches on the collision-control tradeoff are rare. In [10] the researchers proposed a method of varying the control packets exchanged according to traffic load to balance the collision-control energy tradeoff. However, the method couldn't adaptively be adjusted with traffic variations and must be estimated through a lot of off-line experiments. In addition, it doesn't take application-specific QoS requirements into consideration. From the measurement of traffic load in [8] and [9], they still have evident disadvantages, such as too much information exchanged or offline. Hence, it is essential to seek novel methods to measure traffic load condition and balance the tradeoff between collisions and control overhead.

3 The Algorithm

In this section, we present our algorithm to adaptively transmit control packets in accordance with the traffic load status and application-specific QoS requirements. We define an important parameter: colliding loss rate as a mapping parameter to indicate the load of traffic applied in our algorithm. Based on the critical traffic load condition, we build an energy model to obtain the balance point to tradeoff between the collision and control overhead under which our algorithm could save energy compared to the normal control mechanism. Meanwhile, we define QoS threshold to express the application-specific QoS requirements. Combining the balance point and QoS threshold, we further define "Control threshold" as the minimum value of the both to reflect the conditions of traffic load and QoS constrains comprehensively. The energy could be saved without the influence of QoS performance when we adaptively adjust control packets' transmission according to the control threshold. We will discuss our algorithm in detail in the following.

3.1 Mapping Parameter for Traffic Load

To reflect the variation of traffic load, we need to choose a proper parameter to map the dynamics of traffic load. Here we define "packet colliding loss rate" as the mapping parameter. The colliding loss rate equals the ratio between the number of colliding packets and the number of the sent packets in the node within a period of time. Here we include two types of packets: control packets and data packets. Given a type of applications that are characterized by their traffic load and QoS features, the colliding loss rate P_{col} for each application in the node can be expressed as:

$$P_{\mathrm{col}} = n_{\mathrm{drop}}/n. \tag{1}$$

Defining the collision packet loss rate has several advantages: first, employing the collision packet loss rate as the measurement of traffic load can be used to estimate the balance point to tradeoff the control overhead to avoid collision. Second, computing the collision packet loss rate doesn't have to exchange information among neighbor nodes so that it could save much energy as compared with the measurement method in [8]. Third, in contrast to the methods in Z-MAC [9], the collision packet loss rate in our method could be estimated online and give good feedback to the nodes as the input parameter to adjust the transmission of control packets based on the tradeoff balance point described in below.

3.2 The Balance Point

Aiming at the collision-control tradeoff we build an energy model and acquire a balance point for each application. When local packet loss rate is below the balance point, data packets could be transmitted without the help of control packets so as to save energy. This is called adaptive control mechanism compared to the normal control mechanism where the handshake of control packets is always required before transmitting data packets.

First, we make some assumptions to simplify the model. In operations of a WSN, a node may stay in six states: idle, successful data transmitting, successful control packets transmission, collisions, reception and sleep. We focus on the collision-control energy tradeoff that mainly happens during the transmission periods, the idle and sleep states out of the periods are ignored where they take no effect on the tradeoff balance. The total energy consumption then includes four parts: reception, successful control packets transmission, successful data transmission and collisions. Denoting energy consumption of these four parts as $E_{\text{reception}}$, E_{control}, E_{data} and $E_{\text{collision}}$ respectively, the total energy consumed in the node during T can be expressed as:

$$E= E_{\text{reception}}+ E_{\text{control}} + E_{\text{data}} + E_{\text{collision}}. \tag{2}$$

We choose a duration time of T, E represents the total energy consumption in a node for a QoS specific application during T. We define P_{col} and $P_{col}^{'}$ as the collision packet loss rate under two conditions: adaptive control mechanism and normal control mechanism. P_{col} and $P_{col}^{'}$ are variables influenced by traffic load of different applications. E_{t} and E_{r} represent the energy expended for transmitting one byte and receiving one byte respectively. Here, we introduce two types of packets: control packets and data packets. At MAC layer, R and $R^{'}$ denotes the transmission rate, also S and $S^{'}$ denotes the receiving rate of the data packets and the control packets respectively. Under adaptive control mechanism, control packets are adaptively transmitted when necessary, therefore $R^{'}\leq R$ and $S'\leqslant S$. In the same node, we assume that data transmitting rate R should exceed the receiving rate S, that is, $R\geqslant S$, so that no buffer overflow would happen, otherwise the traffic load would be heavy and there is no advantages to use adaptive control algorithm for $R^{'}\leq R$. Each application would have its own average length of control packets and data packets as denoted by L_0 and L respectively. Unless otherwise stated, the following deductions refer to each application.

When using control packets adaptively, the total number of data packets and control packets sent can be expressed as $R\times T$ and $R^{'}\times T$ individually. Among these control packets, $R^{'}(1-P_{col})T$ of them are successfully transmitted. Therefore the energy consumption of successfully transmitted control packets could be computed as:

$$E_{control}=R^{'}(1-P_{col})T\times L_0\times E_t. \tag{3}$$

According to (3), the number of data packets successfully transmitted together with control packets is $R^{'}(1-P_{col})T$. Since control packets are adaptive transmitted with the average rate of $R^{'}$ less than data rate ($R^{'}\leqslant R$), the rest of data successfully sent without the aid of control packets is $[R-R^{'}(1-P_{col})](1-P_{col})T$, therefore we could compute the energy consumption of data packets for successful transmission as:

$$E_{data}=\{[R-R^{'}(1-P_{col})]\times E_t+R^{'}\times E_t\}\times L(1-P_{col})T \tag{4}$$

Next, considering collisions may happen in both receiver and sender sides, we transfer the collision packets from the receivers to the senders since the receiver side collision could be caused by the sender side transmission. Since we compute the collision energy consumption from the sender perspective, the number of data packets and control packets collided during transmission can be calculated as $[R-R^{'}(1-P_{col})]T\times P_{col}$ and $R^{'}\times T\times P_{col}$ respectively. Therefore we could get the energy consumption due to collisions including both data and control packets as:

$$E_{collision}=\{[R-R^{'}(1-P_{col})]\times L+R^{'}\times L_0\}\times T\times P_{col}\times E_t. \tag{5}$$

Since collisions happen at the receivers have been included into the sender side, we could easily acquire the energy consumption for receiving data packets and control packets during T as $S\times T(1-P_{col})L\times E_r$ and $S^{'}\times T(1-P_{col})L_0\times E_r$ respectively. The energy cost at the receiver side can be expressed as the energy consumed by the sum of data and control packets receptions,

$$E_{reception}=[S\times L+S^{'}\times L_0]T(1-P_{col})E_r. \tag{6}$$

According to [11], the receive power E_r and the transmit power E_t of the node transceiver are determined by product firms and has a close relation to be expressed as $E_r=kE_t$. In most cases, k could be taken as 1 or 0.5. Here we take k as 1 for the simplicity. Replace E_r in (6) with E_t, we get $E_{reception}$ in below:

$$E_{reception}=[S\times L+S^{'}\times L_0]\times T(1-P_{col})E_t. \tag{7}$$

From (2) (3) (4) (5) and (7), the total energy consumption with control packets adaptively transmitted could be calculated as:

$$\begin{aligned} E&=E_{reception}+E_{control}+E_{data}+E_{collision} \\ &=T\times E_t[S(1-P_{col})L+S^{'}(1-P_{col})L_0+R^{'}\times L_0+R\times L]. \end{aligned} \tag{8}$$

Compare to normal control packet mechanism, each data packet should be transmitted after the handshake of control packets, therefore, control packet are transmitted at the larger rate than data packet, so we get $R^{'}\geqslant R$, under which the receiving rate for control packets and data would be equal ($S^{'}=S$). $R^{'}>R$ means that there are collision occurs for control packets due to the high traffic load, where little space is left for the adaption of control packet. To get a useful result for the balance point, we take critical state at $R^{'}=R$. Replacing $R^{'}$ by R and $S^{'}$ by S in equation (8), we could obtain the energy consumption of $E^{'}$ at this case.

$$\begin{aligned} E^{'} &= E_{\text{reception}} + E_{\text{control}} + E_{\text{data}} + E_{\text{collision}} \\ &= T \times E_{\text{t}}[S(1-P_{col}^{'})L + S(1-P_{col}^{'})L_0 + R \times L_0 + R \times L]. \end{aligned} \tag{9}$$

We use $\triangle E$ to represent the energy saving when control packets are adaptively sent compared to the situation that the control packets are transmitted at least at the same rate as data packets. Hence,

$$\triangle E = E^{'} - E = T \times E_{\text{t}} [S(1-P_{col}^{'})L + S(1-P_{col}^{'})L_0 - S(1-P_{\text{col}})L + S^{'}(1-P_{\text{col}})L_0] + (R-R^{'})T \times L_0 \times E_{\text{t}}. \tag{10}$$

Denoting d and $d^{'}$ as the number of packets dropped due to collision under adaptive control mechanism and normal control mechanism during T respectively. Intuitively $d^{'} \leqslant d$. According to the definition of collision packet loss rate in (1), we could get:

$$P_{\text{col}} = d / [\mathrm{R} \times \mathrm{T} + R^{'} \times \mathrm{T}] \tag{11}$$

$$P_{col}^{'} = d' / 2\mathrm{R} \times \mathrm{T} \tag{12}$$

The purpose of our adaptive control algorithm is to save energy while keep the performance here is packet losses lower enough at accept level constrained by the normal case. To acquire the balance point, we take the critical state as $d^{'}=d$ and combine equations (10), (11), and (12) together, where we find that if the adaptive control algorithm could save energy, □$E \geq 0$ should be satisfied, that is,

$$P_{\text{col}} \leqslant 2(S - S^{'}) L_0 / [S(L+L_0)(R+R^{'}) - 2S \times L - 2S^{'} \times L_0] \text{ and } S^{'} \leqslant S \text{ and } R^{'} \leqslant R \tag{13}$$

Generally, control packet size is smaller than the data packet size and $R^{'} \leqslant R$ as assumed, We get important conclusion that the control packets can be transmitted according to the condition (13), under which energy can be saved without introducing control packet overheads while keep the contention at low level for data transmission.

From (13) we obtain the balance point Φ be equal to,

$$\Phi = 2(S - S^{'}) L_0 / [S(L+L_0)(R+R^{'}) - 2S \times L - 2S^{'} \times L_0] \text{ and } S^{'} \leqslant S \text{ and } R^{'} \leqslant R \tag{14}$$

In the above, we acquire a balance point of the collision-control tradeoff model for valid for each application. According to (14), small Φ requires low packet loss rate so that control packets have to be sent more often to avoid the collision.

3.3 QoS Threshold

From our analysis we could find that application QoS requirements have significant impact on the tradeoff between collisions and control overhead. Our scheme constitutes a step towards this goal where we focus primarily on one main parameter of application QoS requirements—packet loss. Certainly, it could be extended to other QoS parameters and is of our future research.

Different applications have different packet loss requirements. To differentiate the diverse application requirements on packet loss, we define an important parameter-"QoS threshold" which reflects one of the QoS characteristics - packet loss limit allowed by different applications. The high QoS threshold means the application can accept high packet loss rate whereas the low QoS threshold corresponds to low packet loss rate the application can endure.

3.4 Control Threshold

We have defined collision packet loss rate to indicate the traffic variation and deduced the balance point reflecting the tradeoff between collision and control overhead. In section 3.4, QoS threshold are set as a map to QoS requirements. To adaptively adjust control packets' transmission, we define another control parameter-"Control threshold" for a type of applications through using two values, the balance point and QoS threshold. Each application has its own Control threshold to be used for adjusting control packets transmission while meeting various QoS requirement of different applications as below,

$$\textit{Control threshold}=\textit{min}[\text{balance point, QoS threshold}]. \tag{16}$$

If the collision packet loss rate is less than the control threshold defined in (16), the control packet would not necessarily be adopted where the Control threshold could not only guarantee the energy saving, but also assure the application QoS requirements.

3.5 Algorithm Description

The adaptive algorithm for control packets focuses on multiple applications in WSN and each class of applications has its own packet loss requirement and traffic characteristics. The Control threshold set for each application takes both the application QoS requirements and traffic characteristics into consideration and acts as a constraint to save energy. The variation of traffic load closely relates to the balance point expressed by the "packet colliding loss rate". Each application in a node adjusts its control packets' transmission through computing its own local packet loss rate due to collision periodically and comparing it with Control threshold C_l (l represents the application that current packet belongs to). We now explain the algorithm in a step-by-step manner:

1) Every T second, each application in the node records the number of packets sent and dropped as n and d respectively, both include control packets and data packets. The packet collision rate P_{col} can be expressed as d/n. To compute R and R' in section 3.2, we also measure the number of sent control packets as $n_{control}$, so that $R=n - n_{\text{control}} /T$ and $R' = n_{\text{control}} /T$. Additionally, we need to record the number of packets received and sent as n_r and n_r' respectively to calculate $S= n_r /T$ and $S'= n_r' /T$ in section 3.2. The above measurements are used to calculate the balance point in (15) and the control threshold in (16). If no packets sent, the packet loss rate caused by collision can be set as 0. Otherwise we compute the packet colliding loss rate P_{col} of each application according to equation (1). During T duration, if there are data to be sent, go to step 2. Or else return to step 1.

2) Check which application the current data packet belongs to. Then get the QoS threshold of the current application.
3) Compute the balance point Φ for the application according to equation (15).
4) Compute the Control threshold C_l with both 2) and 3) according to equation (16).
5) If collision packet loss rate P_{co^1} is less than C_l, data will be transmitted without the involvement of control packets but the QoS requirements could still be satisfied. Or else the control packets should be transmitted before transmitting data packets to avoid data collisions.

4 Performance Evaluation

4.1 Simulation Setup

In order to evaluate the performance of the algorithm, we implemented our experiments on the NS2 simulator platform. We integrated the algorithm into S-MAC and compare the energy consumption between the improved S-MAC (denoted as ISMAC) and S-MAC.

We design the nodes position in a medium-scale network topology, where 50 stationary nodes are distributed in a region of 1400×400 meters and the sink is fixed near the center of the topology. All nodes share a single frequency band of 2.4 GHz and each node uses an omni-directional antenna. The radio transmission power is fixed to 15.0dbm which denotes approximately 250meters transmission range. The main parameters are shown in Table.1. We run simulations under packet interval from 1s to 100s. From 1s to 10s, the difference between successive packet intervals is 1s, and from 10s to 100s the successive packet interval is 10s. At each point of packet interval we run 10 times of simulations and obtain the average results. Each simulation runs are lasted for 1000 seconds.

Table 1. The Energy Parameters Value

Parameters	Value
Bandwidth(kbps)	20
Initial energy/node(J)	1000
txPower (mw)	386.0
rxPower (mw)	368.2
idlePower(mw)	344.2
sleepPower (mw)	0.05
Duty Cycle (%)	10
Length of control packet(bytes)	10
Length of data gathering packet(bytes)	50
Length of query packet(bytes)	100
The packet loss requirement of data gathering	30%
The packet loss requirement of query	10%
Control duration T (second)	10

In the experiments, we mainly simulate two types of applications including data gathering and query. As given in the Table 1, data gathering application is tolerant to packet loss rate to some extent of 30% while query application should guarantee the reliability and impose stringent constraints on packet loss of 10%. To evaluate the energy performance, we define energy consumption efficiency η as the total energy expenses in the network divided by the packets bytes successfully received by the sink node over the duration of a simulation run. We change the packet generation interval in order to observe the energy performance under different traffic loads. In SMAC, there is a collision function used to handle packet losses happened upon collision, so we employ this function in ISMAC to measure the number of colliding packets including control packets and data packets.

4.2 The Energy Consumption Efficiency of Single Application in the Network

Figure.1 and Figure.2 demonstrate the energy performance of data gathering and query applications individually. Both figures point out that ISMAC obtain decreased energy consumption efficiency obviously comparing with the original S-MAC. In addition, we could see that the energy consumption efficiency rises progressively with increased packet generation interval. From the model we built before, the number of data packets successfully transmitted is $R(1\text{-}P_{col})T$ and the total energy consumption is E, based on the definition of energy consume efficiency in simulation, we could calculate η from our analysis as below,

$$\eta = \mathrm{E}/[\, S(1\text{-}P_{col})T \times L\,]$$

$$= T \times E_t[\,(1\text{-}P_{col})L + S'(1\text{-}P_{col})L_0/S + R' \times L_0/S + (R\text{-}R')L/S + L(1-P_{col})R'/S]/L(1\text{-}P_{col}). \quad (17)$$

According to the equation (17), with the packets generation interval increasing, R decreases that causes η increase. This gives a good explanation for the simulation results where the energy efficiency is raised with the increased interval between data transmission shown in Figs.1-2.

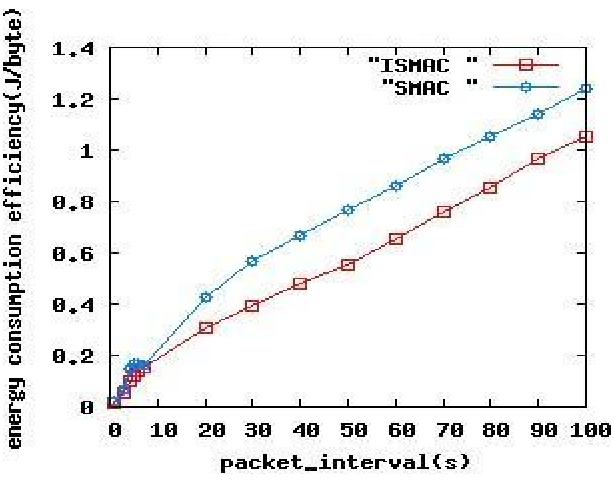

Fig. 1. The comparison for data gathering on energy consumption efficiency

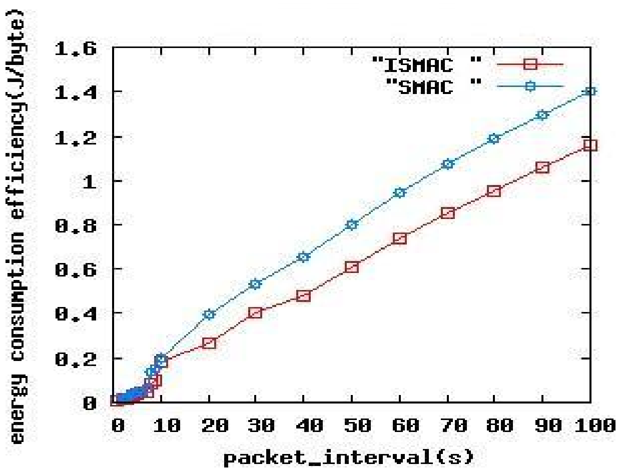

Fig. 2. The comparison for query on energy consumption efficiency

4.3 The Energy Performance of Multiple Applications

In this part, we conduct an experiment in which two applications, that is, data gathering and query co-exit in the network. Figure.3 presents the energy consumption of the network with the two applications which shows that the ISMAC obtains better energy performance than the original SMAC. We also illustrate the packet loss performance of the two applications in Figure.4. The line with square mark represents the packet loss rate of query application and the line with circle mark demonstrates the packet loss rate

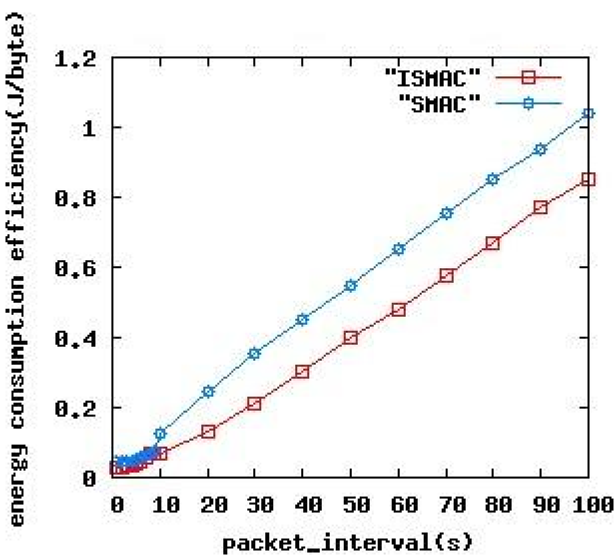

Fig. 3. The comparison of energy consumption efficiency under multi-applications

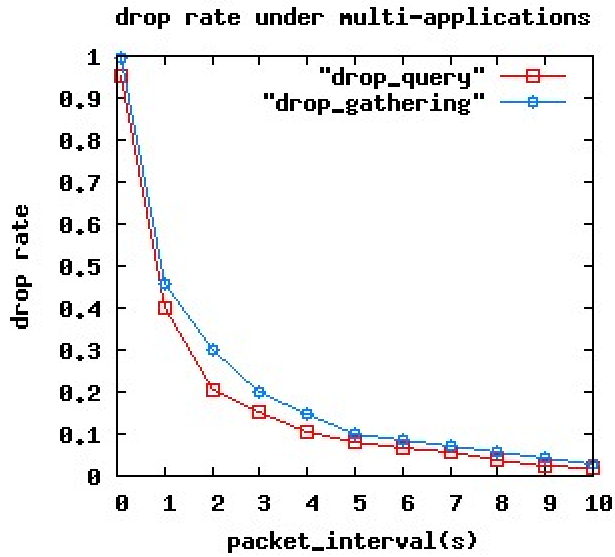

Fig. 4. The drop rate of two applications

of data gathering application. We could find from the simulation that both applications have their packet loss requirement satisfied when employing the algorithm. As shown in Figure 4, when packet generation interval is greater than 3s, the packet loss rate of data gathering application drops below 20%. When the packet generation interval is greater than 5s, the packet loss rate of query application reduces below 10%, which meet the QoS packet loss requirements predetermined by both applications given in Table 1 (30% and 10% allowable drop rate for data gathering and query applications respectively). This demonstrates that our algorithm could not only reduce energy consumption, but also comply with the packet loss requirements of various applications.

5 Conclusion

Most of MAC protocols focused mainly on idle listening, overhearing, collisions and control overhead, the tradeoff between collisions and control overhead is usually ignored. Additionally, the energy efficiency of MAC protocols influenced by multi-applications emerging in WSN is rarely considered. The tradeoff is usually related to traffic load variations characterized by different applications. In the paper, we use colliding packet loss rate as an indication of traffic load dynamics. The control packets are adaptively transmitted according to the balance point derived from our collision-control tradeoff model, as well as the QoS requirement on packet loss at per application basis. The simulation results conducted in NS-2 platform show that our algorithm improve the energy performance greatly compare to original SMAC, while obtain the required QoS performance for various applications. Further work would include other QoS parameters besides packet losses in the paper. The collision-control tradeoff in other types of MAC protocols not just in contention-based MAC is an interest of our future work.

References

1. Demirkol, I., Ersoy, C., Alagoz, F.: MAC protocols for wireless sensor networks: a survey. IEEE Communications Magazine 44(4), 115–121 (2006)
2. Ye, W., Herdemann, J., Estin, D.: An Energy-Efficient MAC Protocol for Wireless Sensor Networks. In: Proc. of the INFOCOM 2002, vol. 3, pp. 1567–1576. IEEE Computer Society, San Francisco (2002)
3. Dam, T., Langendoen, K.: An Adaptive Energy-Efficient MAC Protocol for Wireless Sensor Networks. In: Proc. of the 1st ACM Conf. on Embedded Networked Sensor Systems (SenSys), pp. 171–180. ACM Press, New York (2003)
4. Choi, S.-C., Lee, J.-W., Kim, Y., Chong, H.: An Energy-Efficient MAC Protocol with Random Listen-Sleep Schedule for Wireless Sensor Networks. In: TENCON 2007 - 2007 IEEE Region 10 Conference, Taipei, pp. 1–4 (2007)
5. Polastre, J., Hill, J., Culler, D.: Versatile Low Power Media Access for Wireless Sensor Networks. In: Proc. of the 2nd ACM Conf. on Embedded Networked Sensor Systems (SenSys), pp. 95–107. ACM Press, Baltimore (2004)
6. El-Hoiydi, A., Decotignie, J.D.: WiseMAC: An Ultra Low Power MAC Protocol for Multi-hop Wireless Sensor Networks. In: Proc. of the IEEE symposium on Computers and Communications, pp. 244–251. IEEE Computer Society, Alexandria (2004)
7. Buettner, M., Yee, G., Anderson, E., Han, R.: X-MAC: A Short Preamble MAC Protocol For Duty-Cycled Wireless Networks. In: Proc. of the 4th ACM Int'l Conf. on Embedded Sensor Systems (SenSys), pp. 307–320. ACM Press, New York (2006)
8. Rajendran, V., Obraczka, K., Garcia-Luna-Aceves, J.: Energy-efficient, collision-free medium access control for wireless sensor networks. In: Proc. of the International Conference on Embedded Networked Sensor Systems (SenSys), Los Angeles, pp. 181–192 (2003)
9. Rhee, I., Warrier, A., Aia, M., Min, J.: Z-MAC: a Hybrid MAC for Wireless Sensor Networks. IEEE/ACM Transactions on Networking 16(3), 511–524 (2008)
10. Goel Siddhartha, K., Tamer, E., Mani, S.: Towards Balancing Medium Access Energy Trade-Offs in Wireless Sensor Networks. In: Proc. of the 3rd Annual International Conference on Mobile and Ubiquitous Systems-Workshops, San Jose, pp. 1–8 (2006)
11. Zhou, S., Liu, R.P., Guo, Y.J.: Energy Efficient Networking Protocols for Wireless Sensor Networks. In: Proc. of the 4th IEEE International Conference on Industrial Informatics, Singapore, pp. 1006–1011 (2006)