ELSEVIER

Available online at www.sciencedirect.com

SciVerse ScienceDirect

Procedia Engineering 15 (2011) 2042 – 2046

Procedia Engineering

www.elsevier.com/locate/procedia

Advanced in Control Engineeringand Information Science

Based on Cross-layer Adaptive Contention Window MAC Protocol for Middle and High Rate Sensor Networks

SHEN Jian-Fang*[1,a] □CHENG Liang-Lun[2,b]

[1]Faculty of Computer Guangdong University of Technology, Guangzhou, China

[2]Faculty of Automation Guangdong University of Technology, Guangzhou, China

[a]E-mail: tysjf@163.com

[b]E-mail: llcheng@gdut.edu.cn

Abstract

In middle and high rate sensor network (MHWSN) usually contains a large quantity of self-organizing distributed nodes. It can be widely applied in emergency searching, disaster salvation and military applications etc. As middle and high rate sensor network node energy is limited, so energy efficient is the primary issue for Sensor network Medium Access Control (MAC) protocol design. BEB back-off algorithm cause more collision when node is in middle or high rate. A new back-off algorithm named ACW (Adaptive Contention Window) is proposed in this paper, which adjusts the CW based on the number of historical collision. In ACW, the adjustment of CW can properly reflect the state of medium contention, which results in improvements of throughput and reducement delay in MHWSN.

Selection and/or peer-review under responsibility of [CEIS 2011]

Keywords:SMAC, Middle and High Rate Sensor Network, Contention Window

1. Introduction

SMAC is initiated based upon 802.11 protocols, is designed to fulfill the energy efficient demand of sensor network. Periodical Sleep mechanism, Adaptive Snooping mechanism, Crosstalk Avoidance

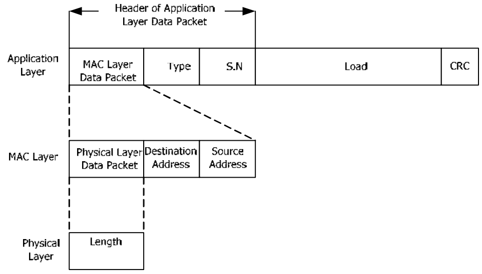
* Corresponding author. Tel.:+8613570594369
E-mail address: tysjf@163.com

doi:10.1016/j.proeng.2011.08.381

mechanism and Message Transmission mechanism make the SMAC pretty excellent in network energy-consuming and latency performances when node is in low rate, and its major design target is to reduce energy-consuming so as to provide good expansibility. but when node is in middle and high rate. BEB back-off algorithm cause more collision.

2. Critical Technology for Realizing SMAC Protocol

(1) Structure of Data Packet



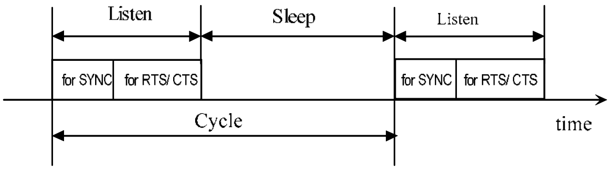


Fig.1. Data Packet Format of SMAC Protocol Fig.2.Periodic Listen and Sleep Mechanism

Data packet format of SMAC protocol is shown as Fig.1. In SMAC protocol, the upper layer packet contains the information of lower layer packet. Each layer is only assumed responsibility to deal with the information belong to it when a packet is received.

(2) Periodic listen and sleep

In SMAC protocol stack, a period includes listen period and sleep period. In listen period node changes synchronization with neighbors and receive packets from neighbors. In sleep period node turns off radio. The periodic listen and sleep schedule is shown in Fig.2.When MAC Layer receives a packet from upper layer, it will start carrier sensing. If the result shows the MAC layer is idle, it will send data to Physical Layer; if MAC layer is busy, it will enter sleep state, waiting until the next idle time, and the data will be resent by then. When MAC Layer receives a packet from Physical Layer and finds there isn't any error after Cyclic Redundancy Check (CRC), MAC Layer will send this packet to upper layer.

(3) Select and Maintain Scheduling Table

Before starting periodical listen and sleep, each node needs to select sleep scheduling mechanism and must be consistent with that of neighbor nodes. How to select and keep the consistence? There are three conditions, as follows:

(a) In listen time, if a node does not discover the sleep scheduling mechanisms of other nodes, it will select a sleep scheduling mechanism immediately;(b) If a node receives a sleep mechanism the neighbor nodes broadcast before selecting and declaring its own scheduling mechanism, it will apply the sleep scheduling mechanism of neighbor's; (c) After selecting and broadcasting its own sleep scheduling mechanism, if a node receives several different sleep mechanisms, this case should be considered in two ways: if one node does not have a neighbor node, it will discard current sleep scheduling mechanism of its own and adopt the latest received one; if this node have one or more neighbor nodes, it will adopt several different sleep scheduling mechanisms simultaneously.

(4) Clock Synchronization

In SMAC protocol, node and neighbor nodes should keep clock synchronization for listen and sleep simultaneously. SMAC protocol adopts relative but not absolute timestamp, and meanwhile let listen time much greater than clock error and floating so as to reduce synchronizing errors. Besides, node will update its own clock in accordance with the packet received from neighbor nodes to keep clock synchronization with them.

(5) Carrier Sense Multiple Access/Collision Avoidance (CSMA/CA)

Basic mechanism of Carrier Sense Multiple Access/Collision Avoidance (CSMA/CA) is establishing a handshake mechanism to transfer data, this handshake mechanism is: Firstly a sender sends a request-to-send (RTS) packet to its receiver, receiver reply a clear-to-send (CTS) packet to sender. After receiving this CTS packet, sender begins to send packets to receiver. The handshake between RTS and CTS is to ensure the neighbor nodes know that sender and receiver are transmitting data, and then the transmission collision can be greatly reduced.

(6) Network Allocation Vector

In SMAC protocol, each node maintains a Network Allocation Vector (NAV) to indicate the active time of neighbor nodes, and each packet of SMAC protocol contains a continuous time duration indicator, which indicates the time duration for current communication. When a neighbor node receives a packet, no matter this packet is sent by sender or the receiver sent to other nodes, the node will know how long it should sleep, that is, to update NAV value according to the time duration in the packet. When NAV value is not zero, node should enter sleep state to avoid crosstalk. As soon as the NAV value is zero, it will wake up and get ready for communication.

3. Performance Analysis of BEB in SMAC Protocol

In SMAC protocol adopts Binary Exponential Back-off (BEB) algorithm. Node select a random back-off time, The back-off timer is calculated as below: Back-off time = Random(•) * aSlotTime , Random (•) is randomly selected from 0 to CW with equal probability, where CW represents contention window size. Initially, contention window is $CW_{\min}$. CW is doubled for the next transmission attempt every time the packet is involved in a collision while CW is reset to the initial value $CW_{\min}$ after every successful transmission. However, the contention window size can not exceed its $CW_{\max}$.

If transmission is in a collision,

CW $\leftarrow$ min(CW * 2, $CW_{\max}$)

If successful transmission,

CW $\leftarrow CW_{\min}$

BEB is efficiency when node is in low rate. But with increment of load of the network, shows two shortcomings:

A. Unfairness is getting more and more notable. In a short period, because CW is reset to the initial value $CW_{\min}$ after every successful transmission while double CW for the packet is involved in a collision. In the next contention the latest successful node would content successfully in a larger probability as compared with those fail node in the latest contention. All nodes are unfair to share the restricted medium resource. This is short-period unfairness.

B. There is drastic jitter of delay. With the increment of load of the network, CW changes drastically. The CW size of some nodes may be much bigger which result in bigger delay. But once the nodes access the medium, the CW would be reset to the least value as $CW_{\min}$, and then the delay would reduce much more. These would certainly lead bigger range delay jitter, which result in bad influence on some real-time event.

Hence, the contention window resetting mechanism causes a very large variation of the contention window size, and degrades the performance of a network when it is heavily loaded.

4. Adaptive Contention Window (ACW)

To resolve these problems, Adaptive contention window algorithm called ACW was proposed. ACW adjusts the *CW* size on the base of the number of collision times of the waiting data packet. So CW size precisely response the state of the medium. Variable $count$ is the number of collision times of the waiting data packet. The algorithm is described as follow:

$CW_{\mathrm{i}} \leftarrow CW_{\min}$ (i=0);

$$CW_{\mathrm{i}} \leftarrow \left[\prod_{n=0}^{i-1} \frac{threshold - n}{threshold} + 1 \right] * CW_{\min} \quad \text{(i>0)};$$

When data packet in a collision,

IF $count < threshold$

$count \leftarrow count + 1$;

$CW \leftarrow CW_{count}$;

IF $count = threshold$

$count \leftarrow 0$;

$CW \leftarrow CW_0$;

When transmission id successful,

$count \leftarrow [count / 2]$;

$CW \leftarrow CW_{count}$;Where [] represents getting integer, threshold is a adjustable integer which is decided by the value of $CW_{\min}$ and $CW_{\max}$. In other words, threshold is the biggest integer which satisfy $CW_{threshold} < CW_{\max}$. For example, when $CW_{\min}$=16, $CW_{\max}$=1024, threshold = 9.The figure of state transformation of CW size is described as figure 3.

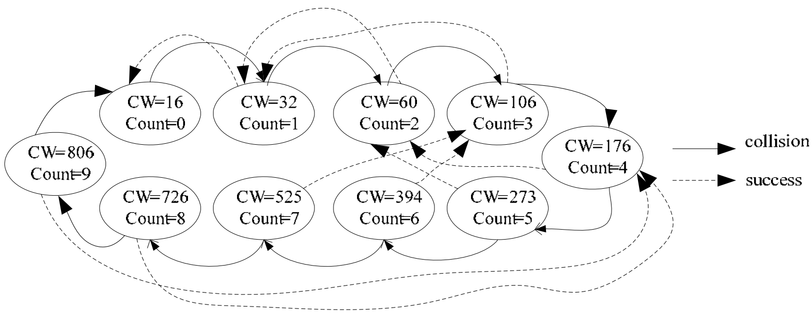
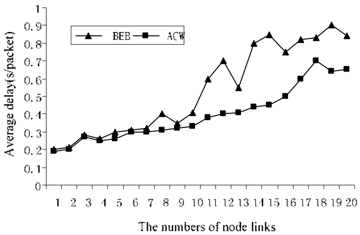


Fig.3. $CW_{\min}$=16, $CW_{\max}$=1024, State Transformation of *CW* size

Compared with BEB, CW of ACW changes more smoothly. ACW adjusts the CW size on the base of the times of collision of the waiting data packet. CW can precisely reflect the state of the medium. So ACW improves the short-period unfairness among nodes. Otherwise, once a node failed enough times, and then its CW is equal to CWthreshold, after a new collision, its CW would be reset to $CW_{\min}$. So in order to enhance its capacity of accessing medium in the next contention on the purpose of preventing from not accessing medium in a long time for too big CW. After successful transmission, a slower back-off mechanism is adopted, on the first hand it can reduce the delay jitter, on the other hand it can prevent more new collision as a result of resetting CW as $CW_{\min}$.

5. Simulat i on

Compare SMAC with ACW, and get simulation for performance in NS2, distributed N a node in rectangle. Each node is within the covered scope of other nodes. Number of adjacent the odd number and the even number established TCP connection. Node 0 and node 1 established connection, nodes 2 and nodes 3 established a connection. Every 50s start a node，The simulation time is 20s, $CW_{\min}$ =16, $CW_{\max}$ =1024, the RATE of date is 1.5k, every data packet is 1024bytes, the transmission power is 0.26w, Bandwidth=20k.



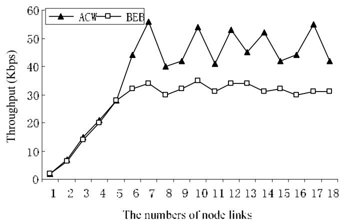


Fig. 4 Average delay of packets comparison of BEB and ACW Fig. 5 Throughput comparison of BEB and ACW

It is well-known wireless medium is badly limited resource in middle and high rate sensor network. The CW size of SMAC adopts BEB changes too drastic, as in result the performance of throughput and delay drastically degrade with the increment of load. This paper proposed a algorithm called ACW (Adaptive Contention Window) based on collision number. The simulation results show that ACW can effectively reduce collisions and improve throughput and reduce delay in MHWSN.

References

[1] Yu Haibin, Zeng Peng, Liang Wei. Intelligent wireless sensor networks System [M]. Beijing: Science Press, 2006:68-102.

[2] Zhang Ye, He Chen, Jiang Lingge. Modeling the S-MAC Protocol in Single-Hop Wireless Sensor Networks [C]//Proc. of IEEE ICC08. Beijing, China: IEEE Press, 2008: 317-321.

[3]Ye Wei, Heidemann J, Estrin D. Medium Access Control with Coordinated Adaptive Sleeping for Wireless Sensor Networks[J]. IEEE Trans. on Networking, 2004, 12(3): 493-506

[4]Navrati Saxena, Abhishek Roy, Jitae Shin.Dynamic duty cycle and adaptive contention window based QoS-MAC protocol for wireless multimedia sensor networks[J].Computer Networks.2008,52(13):2532-2542

[5] Ye W□Heidemann J□Estrin D□An energy efficient MAC protocol for wireless sensor networks[A]□Proceedings of the IEEE lnfocon[C]□New York□USA□IEEE□2002□1567□1576□