PERFORMANCE PREDICTION OF PACKET SCHEDULING ALGORITHM IN WIRELESS SENSOR NETWORKS

[1]C. Jandaeng,[1]W. Suntiamontut, [2]N. Elz

[1]Centre of Excellence in Wireless Sensor Network,
Department of Computer Engineering, Prince of Songkla University
cjundang@gmail.com

[2]Computer System and Network Laboratory,
Department of computer Science, Prince of Songkla University,
Hat Yai, Thailand 90112

ABSTRACT

The main cause of the wasted energy consumption in wireless sensor networks is the packet collision. The packet scheduling algorithm is therefore introduced to solve this problem. This paper presents an analytical model for predicting the performance of packet scheduling algorithms in wireless sensor networks. This model can be used to predict the performance of the packet scheduling algorithm with different node density and CBR in wireless sensor networks. The proposed prediction model can give the packet collision and packet delivery ratios very close to the experimental results. The accuracy of our model is 77% of packet collision model and 81% for packet delivery ratio model.

KEYWORDS

1. INTRODUCTION

A wireless sensor network is a self-configured network containing numerous small sensor nodes. Each node consists of sensing modules, a processing unit, radio frequency components and power sources [1]. They organize and communicate among themselves in an ad-hoc fashion. The wireless sensor networks technology has been deployed in several applications such as health care monitoring systems, home automation and environment monitoring systems [2]. These applications require inexpensive facilities and little manual maintenance. According to the application requirements, each node has been implemented using a low-power microcontroller and radio module. In addition node is supplied with a small battery. Energy usage is the indicator of network lifetime [3].

All sensor nodes share a single channel using a multiple access protocol. The packet transmission may lead to a time overlap of two or more packet receptions, called collisions. The packet collision problem causes the packets loss, packet retransmission, decreasing throughput, increased delay/latency and increased wasted energy consumption. Many research works on MAC protocol have been proposed to solve the packet collision [4] such as Spatial TDMA traffic-adaptive medium access protocol (TRAMA) [5], Sensor MAC (SMAC) [6], and

113

Timeout MAC (TMAC) [7]. MAC protocol based on IEEE 802.15.4 was developed for low-power communication. This protocol uses a random back off in order to reserve and access the channel. A node is authorized to send the packet when the channel is idle. In contrast, random back off is activated when the channel is busy. Unfortunately, this technique will not work properly when used in a large scale wireless sensor network.

Time Division Multiple Access (TDMA) is a solution to reduce the packet collision problem. Total transmission time is divided into frames and each frame is divided into time slots. After that each time slot will be assigned to a sensor node to guarantee that every node is granted to send a packet in its time slot permission guaranteeing collision avoidance. The latency directly varies with frame length. On other hand, the throughput inversely varies with frame length. There have been many approaches presented to minimize the frame length and maximize the throughput which are explained in section 2.

There have been some packet scheduling algorithms [15] for wireless sensor networks proposed recently. Packet Scheduling Algorithm (PSA) has to be considered carefully before it is chosen to deploy in each application. The packet collision ratio and the packet delivery ratio (PDR) are the main parameters which the developer uses to consider how efficient the PSA is. Therefore, this paper proposes a performance prediction model of our PSA to help the developers investigate whether the PSA is suitable for their applications.

The remainder of the paper is organized as follows. First we briefly introduce the packet collision problem and the existing packet scheduling algorithms in Section 2. Our system model and the packet scheduling algorithm are explained in section 3. The performance prediction model is proposed in Section 4. The prediction results are discussed in Section 5. Finally we give conclusions about the performance prediction model in Section 6.

2. PREVIOUS WORKS

Peng *et al* [8] presented the TDMA with a scheduling matrix. The row of the matrix denotes frame length while the column of matrix denotes nodes. The members of the matrix represent transmission authorization. In [8], they proposed to optimize the number of row that refers to the frame length with Tabu search and greedy algorithm. This approach can reduce the average latency and produce a high throughput in a dense area.

Wang, G. and Ansari, N. [9] have proved that the scheduling matrix optimization is a NP-complete problem. They also proposed the approximation method, mean field anneal (MFA) to optimize the schedule matrix. The matrix optimization is divided into two phases: minimize frame length and maximize throughput. After that approximation methods were proposed. Salcedo-Sanz, S *et al* [10] minimized frame length with a neural network (NN) and maximized throughput with a genetic algorithm (GA), whereas Yeo, J. *et al* [11] applied the sequence vertex coloring (SVC) in both phases. Moreover, Haixiang, S. and W. Lipo [12] proposed a hybrid algorithm which combined back tracking sequential coloring (BSC) and noisy chaotic neural network (NCNN) to optimize the scheduling matrix. BSC-NCNN gives the most minimal average time delay, while the NN-GA provides higher throughput.

Ahmad, I. *et al* proposed an idea to avoid packet collision. The network topology is represented by a finite state machine (FSM) [13]. The set of nodes are grouped with the maximal compatibles and incompatibles concept [14]. This idea started from setting up groups that equal the number of nodes. After that, combine groups together under the condition that all nodes in the same group are not neighbor nodes. Finally, all sensor nodes are grouped in many groups

and they can send packet in the same time without collision. The number of groups is frame delay while the summation of number of node in all groups is throughput. This idea leads to minimize latency and maximize throughput.

3. SYSTEM MODEL AND PROBLEM STATEMENT

3.1 System Model

We assume that wireless sensor networks have n nodes using the same communication range with a sink node collecting the data from the other nodes. Therefore, we can represent the wireless sensor network as undirected graph $G=(V, E)$, where V is a set of nodes $(V_1, V_2, …, V_n)$ and E is a set of edges for all communication links. In the case of $(u, v) \in E$, node u and v are one hop apart. Further, if u and v are not one hop apart and have an intermediate node k which $(u, k) \in E$ and $(k, v) \in E$, node u and v are two hops apart. The network topology as shown in Figure 1 network topology represented with graph theory can be represented by graph theory $V=\{1, 2, 3, 4, 5\}$ and $E=\{(1, 2), (2, 1), (2, 3), (3, 2), (3, 4), (4, 3), (3, 5), (5, 3), (4, 5), (5, 4)\}$.
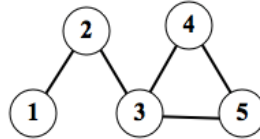

Figure 1 network topology represented with graph theory

There are two types of packet collisions: direct collision and hidden collision. Let $u, v, k \in V$ that are one hop apart. If u transmission time overlaps with v transmission time, a direct collision will occur. On the other hand, a hidden collision will occur when both u and v transmit the data to k at overlapping times, and u and v are not one hop apart. For example, there is a direct collision between node 4 and 5 because of sharing the same channel with node 3 as shown in Figure1. The hidden collision will occur between node 2 and 4 when both of them send the packet to the same destination (node 3) at the same time.

3.2 The Scheduling Matrix

In order to avoid packet collision, the scheduling with TDMA approach is introduced. The network topology can be represented by the scheduling matrix as shown in Figure 2.

Figure 2 scheduling matrix

Figure 2 shows the number of node, N, and the number of row, L. The number of row refers to frame length and refers to delay. The matrix elements assigned with 1 refers to throughput. In addition, if frames 1 and 5 are duplicated, then frame 5 can be deleted to reduce frame length.

The frame length is decreased while channel utilization is increased. The channel utilization is ratio between throughput and channel availability. There have been several methods proposed to find and remove unnecessary frames to reduce delays and increase channel utilization as explained before.

3.3 Packet Scheduling Algorithm

The Packet Scheduling Algorithm (PSA) [15] is the algorithm that schedules all packets from application layer and network layer in order to reduce network congestion in the data link layer to avoid the packet collision. When the PSA is implemented, packet collisions will be minimized and increasing of throughput as a by product. A PSA that based on greedy algorithm is a simple algorithm and easily implemented in a sensor node. The basic assumptions of the PSA are defined as follows. All packets communicate via IEEE 802.15.4 standard that avoids packet collision with a simple CSMA/CA mechanism. All sensor nodes must know the information of at least 2-hops neighbor nodes. Finally, time synchronization, neighbor discovery, and routing protocols are not considered in this work.

The color of each node represents node status. Two functions, *combine()* and *match()*, are used to reduce the frame length. The node statuses and functions are defined below:

Definition 1. *Node status* is represented by a node color for each time slot. A black node can send any packet with a guarantee of no collision. If a white node requests to send a packet, its packet may collide. A gray node is the initial status with no guarantees for packet collision. Finally, a gray node can change status to other colors by *combine()* and *match()*.

Definition 2. *combine()* is used to reduce frame length by combining two frames. The two frames must be tested with *match()* before combination. They will be merged in case the *match()* function returns valid. In the combination process, the status of a node will be changed to another color as defined below. Let *A, B* and *R* denote frame and A_i, B_i, and R_i be node status i^{th} time slot of *A, B,* and *R; V* denotes the set of nodes and *R=combine(A,B)*.

$$\forall_{i: i \in V}, R_i = \begin{cases} A_i : if \ B_i = GRAY \\ B_i : \ otherwise. \end{cases}$$

Definition 3. The *match()* is used to validate two frames before combination. All node status must follow the condition below:

$$match(A, B) \leftrightarrow \forall_{i: i \in N} A_i = GRAY$$
$$\vee (A_i = BLACK \wedge B_i = GRAY)$$
$$\vee (A_i = WHITE \wedge B_i \neq BLACK)$$

The algorithm consists of two phases: frame length minimization and throughput maximization. In the frame length minimization phase, frames are sorted by the number of gray nodes. Sensor nodes find the two frames that have the most gray nodes and are valid with *match()*. After that both frame are merged with *combine()* and added into scheduling matrix. All frames are sorted again. All steps are repeated until we cannot find two frames that are valid with *match()*. In the throughput maximization phase, a sensor node scans the scheduling matrix in column order to

find the gray node. If column i^{th} has a gray node in frame j^{th}, the frame i^{th} from this scheduling matrix and j^{th} from original scheduling matrix are merged with *combine()* when they are matched. The details and performance analysis of this algorithm are compared with previous work and explained in [15].

4. PERFORMANCE PREDICTION MODEL

4.1 Simulation

The IEEE 802.15.4 module for wireless sensor networks has been proposed by Jianliang Zhen and declared to be a standard module in Network Simulator (NS2). The details of the IEEE 802.15.4 standard are described in [16]. Wireless sensor networks have mostly been deployed in two types of applications, surveillance and tracking systems. The data communication in these applications is periodic as Constant Bit Rate (CBR) in NS2. The User Datagram Protocol (UDP) is selected as the transport protocol in order to disable the packet retransmission mechanism. Wireless sensor networks are similar to ad hoc networks. Thus we choose the AODV routing protocol to use in the network simulation. The data link layer and physical layer rely on non-beacon IEEE 802.15.4 standard that manages packet collision with simple CSMA/CA [17].

Three independent variables: density of node, CBR rate and quantum time are relevant to this performance model. From the density of node perspective, the sensor nodes in transmission range must receive and process incoming packets. Some packets must be forwarded which will increase network congestion and also lead to high packet collision. All sensor nodes are randomly placed in a field sized 40x40 m^2. Node density (d) is varied with $0.01 <= d <= 0.1$.

The CBR rate indicates the number of packets sent from a node. When the CBR data is passed to the lower layer, the overhead such as routing packet and MAC packet are added. In order to evaluate the effects of the traffic rate from the application layer, the CBR rates (t) are varied from 1 to 250 kbps ($1 <= t <= 250$) where the bandwidth of IEEE 802.15.4 standard is 250 kbps. All sensor nodes send the CBR packets with the size 40 bytes to the base station that is placed in the middle of the sensor field.

The quantum time is the time interval between frame i^{th} and $(i+1)^{th}$. If the quantum time is long, the sensor node has to wait longer than usual for the next frame. This will directly affect the packet delivery ratio. In AODV protocol [18], a node waits for the Route Reply (RREP) after broadcasting a Route Request (RREQ). If a route is not responded within 2,800 milliseconds, the node may discover a route by broadcasting another RREQ. Data from the upper layer protocol are buffered in a queue while waiting for the route. If the routing process takes a time longer than the transmission time of the upper layer protocol, the FIFO queue will become full and the data will be dropped. In the simulation, the quantum times (q) are varied from 0.1 ms to 1.0 ms ($0.1 <= q <= 1$) in order to study the relation of the quantum time, packet collision rate and packet delivery ratio.

4.2 Performance metrics

The performance of the PSA is evaluated in terms of two aspects, packet collision rate and packet delivery ratio. The packet collision rate dominates the wasted energy in the network and the packet delivery ratio gives the throughput of the PSA. All parameters used in the simulation are described as follows:

- *Energy consumption*: The energy consumption of the radio transceiver can be calculated using the transmitting current (I_{TX}), the receiving current (I_{RX}) and the supply voltage (*V*) with the data rate of IEEE 802.15.4 standard (250kbps), one byte data will use *31.25 μs* for transmission. Then we can calculate the energy usage in the transceiver for a *k* bytes transmission using *E(k)=kTV(I_{TX} + I_{RX})*. For example, CC2420 module has an average current of 19.4mA and 17.4mA for data transmitting and receiving, respectively [19]. Therefore the total energy consumption of the CC2420 is *k2.74 uJ* for *k* bytes transmission when the supply voltage is 2.4 V. The main purpose of the PSA algorithm is to minimize the packet collision that leads to reduce wasted energy. Thus we can conclude that the energy dissipation depends directly on the packet collision rate.

- *Packet collision rate* (PCR) is the amount of dropped data in one second. All sensor nodes share a single channel using a multiple access protocol. Packet collision then occurs when a packet transmission leads to a time overlap of two or more packet receptions. High node density has higher possibility of collision than low node density. The CBR rate increases the number of control packets in the MAC layer increasing packet congestion.

- *Packet delivery ratio* (PDR) is the ratio of the total number of packets received by the nodes to the total number of packet transmitted. A PDR close to 1.0 means that the network has the high performance.

4.3 Multiple regression models

Regression analysis [20] is a statistical method to show the relation between dependent variables and independent variables. This method is used for a prediction based on the information collected in the past and the description of the relationship between dependent and independent variables. The proposed regression model is defined as:

$$y = f(x_1, \ldots, x_p) + \varepsilon, \varepsilon \in N(0, \delta^2)$$

$f(x_1, \ldots, x_p)$ is a regression function that represents the raw data. It may be a linear or non-linear function. ε is a random variable that represents the error and fits the normal distribution. In some cases, the relationship between dependent and independent variables is a non-linear function. The dependent variables are transformed to be linear data. The least square method (LSM) is a method to find the parameters ($\beta_0, \ldots, \beta_p$) and can produce ε^2 which is the smallest value. The linear regression function is presented follows:

$$f(x_1, \ldots, x_p) = \beta_0 + \beta_1 x_1 + \ldots + \beta_p x_p$$

The R statistic package is used to obtain statistic value from the raw data. The correlation explains the relationship between each variable. The *F-test* indicates that the proposed model represents the observed data. In order to indicate the model accuracy, the goodness-of-fit and standard error are measured.

4.4 Performance model validation

To validate the performance model, NS2 has been used to simulate the simple CSMA/CA and CSMA/CA with PSA. The packet collision rate and packet delivery ratio results of the relevant

parameters such as CBR rate (kbps), node density and quantum times have been analyzed the correlation hypothesis with Pearson's product-moment. Then the regression model is generated with the R package.

The multiple regression planes in tables 1 and 2 represent the simulation results. We can conclude that all regression models are represented in the simulation results with the strong evidence (*p-value < 0.05*). There are few errors as can be noticed from the independent variables (R^2 and R^2_{adj} are closed). This information is very useful to assess the overall accuracy of the model.

Table 1 Packet Collision Rate of CSMA/CA

	Correlation		Regression		
	r	*p-value*	β	*t*	*p-value*
log(t)	0.1025	0.000	0.24	7.216	0.000
d^2	0.9436	0.000	5177	58.426	0.000
$d^2 log(t)$	0.9352	0.000	322.90	16.489	0.000
$SE_{est} = \pm 5.389$					
$R^2 = 0.9762$; $R^2_{adj} = 0.9762$; $F=40920$; $p\text{-}value= 0.000$					

From table 1, we found that the packet collision rate is a weak correlation with CBR rate. In contrast, with the node density, the packet collision has a strong correlation. In simple CSMA/CA, the packet collision is related with CBR rate, density of node, and interaction between CBR rate and density of node with coefficient 0.24, 5177, and 322.90, respectively. The prediction model represents packet collision rate to 97.62% with the significant 0.05. The standard error of estimation is $\pm$ 5.389 kbps. When *t* denotes the CBR rate and *d* denotes the density of nodes, the prediction model of packet collision rate for simple CSMA/CA, *PCR(t,d)* can be shown below:

$$PCR(t,d) = 0.24 \log(t) + 5177 d^2 + 322.90 d^2 \log(t)$$

subject to:

$$1 \le t \le 250 kbps$$
$$0.01 \le d \le 0.10$$

Table 2 Packet Delivery Ratio of CSMA/CA

	Correlation		Regression		
	r	*p-value*	β	*t*	*p-value*
log(t)	0.1773	0.000	-0.94	-6.778	0.000
log(d)	-0.7645	0.000	-16.36	-84.688	0.000
log(d)log(t)	0.4970	0.000	2.57	42.982	0.000
$SE_{est} = \pm 7.659$					
$R^2 = 0.7821$; $R^2_{adj} = 0.7819$; $F=3578$; $p\text{-}value= 0.000$					

The CBR rate correlates with the packet delivery ratio less than density of node as packet collision. On the other hand, the density of nodes inversely varies with packet delivery ratio and the interaction between two parameters is average. The packet delivery ratio depends on density of nodes. The relationship between packet delivery ratio, density of node, and interaction of

CBR rate and density of nodes are –0.94, -16.36 and 2.57, respectively. The prediction model represents packet delivery ratio to 78.21% with the significant 0.05. The standard error of estimation is $\pm$ 7.659%. When t denotes CBR rate and d denotes the density of nodes, the prediction model of packet delivery ratio for a simple CSMA/CA, $PDR(t,d)$ can be shown below:

$$PDR(t,d) = -0.94\log(t) - 16.36\log(d) + 2.57\log(t)\log(d)$$

subject to:

$$1 \leq t \leq 250 kbps$$
$$0.01 \leq d \leq 0.10$$

The CBR rate and density of nodes affects the packet collision rate and packet delivery ratio. The density of node affects directly and growth in same trend line with packet collision rate and inversely growth with packet delivery ratio. The CBR rate has almost no correlation with either indicator.

When the PSA algorithm is applied together with the simple CSMA/CA, the results are similar to the simple CSMA/CA. However, the quantum time (q) is determined in this experiment. All parameters are plotted and tested with the R packages. After that the results are shown in Tables 3 and 4.

Table 3 Packet Collision Rate of CSMA/CA with PSA

	Correlation		Regression		
	r	p-value	β	t	p-value
$log(t)$	-0.0489	0.000	-0.08	-5.305	0.000
d^2	0.9576	0.000	5769.55	147.430	0.000
$log(q)$	0.0126	0.000	0.64	3.316	0.000
$d^2log(t)$	0.9246	0.000	307.96	35.209	0.000
$log(t)log(q)$	-0.0422	0.000	-0.17	-3.838	0.000
$d^2log(q)$	-0.9486	0.000	-362.99	-7.704	0.000
$d^2log(q)log(t)$	0.9153	0.000	105.23	10.244	0.000
$SE_{est} = \pm 5.197$					
$R^2 = 0.9792; R^2_{adj} = 0.9792; F = 201000; p\text{-}value = 0.000$					

Table 3 shows the statistic value of packet collision rate. The density of nodes affects the packet collision rate. All parameters that are concerned with the density of nodes are strongly correlated with packet collisions. Although the CBR rate and quantum time are weakly correlated with packet collision rate, these two parameters are strongly correlated with packet collision rate when they interact with density of nodes. The coefficient between packet collision rate implementing PSA algorithm, density of node, CBR rate, quantum time, and all reaction of all parameters are -0.08, 5769.55, 0.64, 307.96, -0.17, -362.99, and 105.23, respectively. The prediction model represents the packet collision rate to 97.92% with the significant 0.05. The standard error of estimation is $\pm$ 5.197 kbps. The prediction model of packet collision rate for simple CSMA/CA with PSA, $PCR(t,d,q)$, is shown below:

$$PCR(t,d,q) = -0.08\log(t) + 5769.55d^2 + 0.64\log(q)$$
$$+307.96d^2\log(t) - 0.17\log(t)\log(q) - 362.99d^2\log(q)$$
$$+105.23d^2\log(q)\log(t)$$

subject to:

$$1 \le t \le 250kbps$$
$$0.01 \le d \le 0.10$$
$$0.1 \le q \le 1ms$$

The CBR rate directly affects the packet delivery ratio but in the inverse direction. The density of nodes has medium affect and packet delivery ratio and the quantum time has less effect on packet delivery ratio. The packet delivery ratio of CSMA/CA with PSA algorithm is shown in table 4. Moreover, the other parameters interact with high significance. The prediction model represents the packet collision rate to 72.16% with the significant 0.05. The standard error of estimation is ± 7.68%. When t denotes CBR rate and d denotes the density of nodes, the prediction model of packet collision rate for simple CSMA/CA with PSA, *PDR(t,d,q),* is shown below:

Table 4 packet delivery ratio of CSMA/CA with PSA

	Correlation		Regression		
	r	*p-value*	*β*	*t*	*p-value*
log(t)	-0.6523	0.000	-0.68	-10.08	0.000
log(d)	-0.2728	0.000	-15.08	-158.01	0.000
log(q)	0.0012	0.000	19.27	22.60	0.000
log(d)log(t)	0.3605	0.000	2.42	83.16	0.000
log(q)log(t)	-0.6136	0.000	-3.70	-19.54	0.000
log(q)log(d)	0.2494	0.000	4.51	16.34	0.000
log(t)log(q)log(d)	0.5476	0.000	-0.87	-14.20	0.000
$SE_{est} = \pm 7.868$					
$R^2 = 0.7216; R^2_{adj} = 0.7216; F=11110; p\text{-}value = 0.000$					

$$PDR(t,d,q) = -0.68\log(t) - 15.08\log(d) + 19.27\log(q)$$
$$+2.42\log(d)\log(t) - 3.70\log(q)\log(t) + 4.51\log(q)\log(d)$$
$$-0.87\log(t)\log(q)\log(d)$$

subject to:

$$1 \le t \le 250kbps$$
$$0.01 \le d \le 0.10$$
$$0.1 \le q \le 1ms$$

5. RESULTS AND DISCUSSION

5.1 Prediction results

The main cause of packet collision is the large number of transmitted in the same interval time. This study noted that the data transfer rate in application layer (CBR rate) and the density of

nodes cause the number of packets to increase in the MAC layer. After the simple CSMA/CA is applied with PSA the quantum time is determined.

We compare the packet collision and packet delivery ratio between the experimental results and the prediction model as shown in figures 3 to 6. The scatter plot of packet collision rate and packet delivery ratio from the simulation results and the prediction model are much close that concern with the R^2 of prediction model. The R^2 is close to 1.0 means that the *PCR(t,d)* and *PCR(t,d,q)* from the prediction model is close to the simulation results. Moreover, the packet collision rate varied directly with the density of nodes and CBR rate as a parabolic function and a logarithmic function. Whereas, the packet delivery ratio varies inversely with the density of node as a logarithmic function and varies directly with the interaction between the density of nodes and CBR rate as a logarithmic function. From these observations, we found that these relationships are very useful when the prediction of the packet collision rate and packet delivery ratio is required.
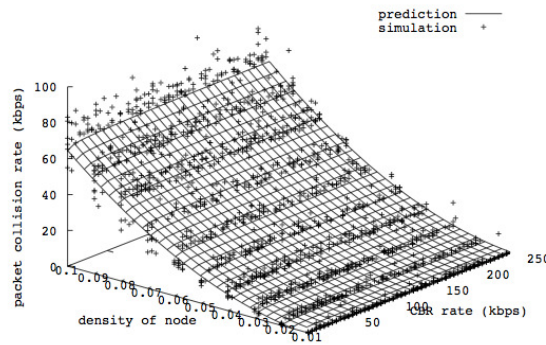

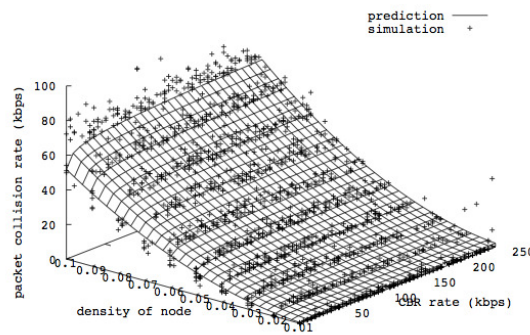Figure 3 comparison of packet collision rate

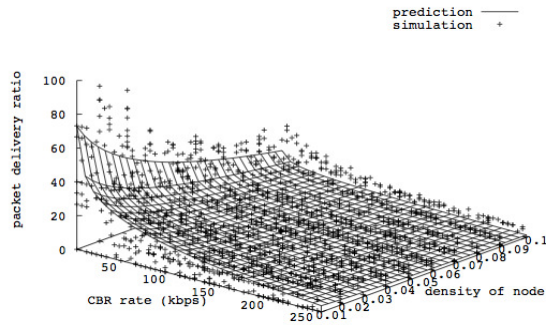Figure 4 comparison of packet collision rate (with PSA)
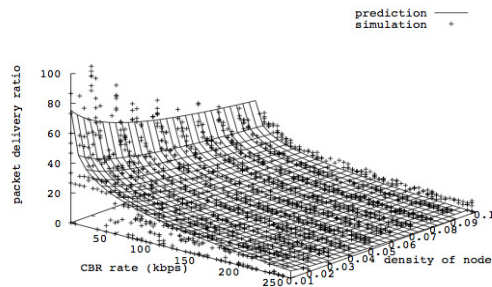
Figure 5 comparison of packet delivery ratio

Figure 6 comparison of packet delivery ratio (with PSA)

5.2 Performance of PSA

This section compares the performance between a simple CSMA/CA and PSA. Both performance metrics are transformed to *Rpcr(t,d,q)* and *Rpdr(t,d,q)* functions that are the ratio between PSA and CSMA/CA. The *f(t,d) = 1.0* is threshold of packet collision rate and packet delivery ratio. Figure 5 shows the comparison between *Rpcr(t,d,q)* represented with solid surface and *f(t,d)* represented with dash surface. For packet collision rate, the solid surface being lower than the dash surface means that the packet collision of PSA is less than the standard. On the other hand, the solid surface of packet delivery ratio being higher than the dash surface mean that the CSMA/CA with PSA is better than the original CSMA/CA.

a) packet collision rate
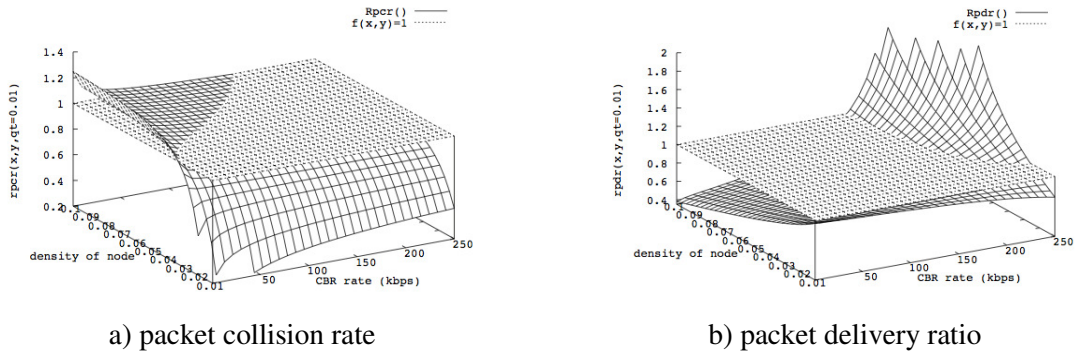

b) packet delivery ratio

Figure 5 performance comparison

Figure 5a shows that the PSA decreases packet collision rate if the density of nodes is less than 0.06 for low CBR rare. It decreases packet collision rate 20% for the low density of nodes (d > 0.03) while packet collision rate is decreased up to 50% while the density of nodes is 0.02 and more than 50% when density of node closes to 0.01. However, this algorithm increases packet collision rate more than simple CSMA/CA only 5% when density of node closes to 0.1 and CBR rate is less than 75 kbps. Moreover, packet collision rate are increase up to 8% when the CBR rate closes to 1.0 kbps.

Unfortunately, PSA can reduce PDR. The ratio of PDR is shown in figure 5b. The PSA decreases PDR for in cases: 1) density of nodes less than 0.05 for all CBR rate and 2) density of nodes more than 0.05 if CBR rate is less than 150 kbps. The PSA increases packet delivery ratio less than 15%. On the hand, the PSA increases PDR up to 50% when CBR rate is more than 150 kbps and density of nodes more than 0.05. The PDR is 200% increased for high density of nodes and CBR rate. The density of nodes effects PDR more than CBR rate because it corresponds with packet collision. PDR from PSA is better than simple CSMA/CA with low density of node.

To evaluate the accuracy of prediction model, the 5000 samples from simple CSMA/CA and 50,000 samples from CSMA/CA with PSA are divided into two parts. The 60% of samples are used for prediction model generation whereas the 40% of samples are used for testing the prediction model. The PCR model and PDR model of simple CSMA/CA are 76.43% and 81.11% respectively. The prediction mode of standard with PSA is 77.64% for PCR and 80.05% for PDR.

The results in figure 5 show that the density of nodes is the main affect upon performance while CBR rate and quantum time have less impact upon performance. The next section is the factor analysis that affects the network performance.

5.4 Effects of parameters

This performance prediction model consists of three parameters: density of nodes, CBR rate, and quantum time. The density of nodes directly affects packet collision rate because packet collision rate grows in a parabolic function when the density of nodes is increased. The density of node is increased by 10% and other parameters are arbitrary values, the packet collision rate is increased in the interval of 20% to 50%. The density of node affects packet collision rate because route discovery packets (RREQ) are broadcasted in MAC layer and the number of RREQ packet at destination depends on the density of nodes. Therefore, it results in the packet having high congestion and high probability of packet collision.

The CBR rate affects packet collision for two reasons: controlled packet increasing and CBR packet in MAC layer. The controlled packet is high in both establishment phase and maintenance phase. When controlled packets are dropped, the routing protocol determines that topology is changed or cannot find a route and it results in the route discovery mechanism is restarted.

The quantum time has less affect upon packet collision rate because the packet collision rate is increased only 0.2% when quantum time is increased 10%. In contrast, the quantum time is the major impact upon packet delivery ratio, more than any other parameter. When quantum time is increased by 10% and other parameters are arbitrary values, the packet delivery ratio is decreased up to 21%.

We conclude that the density of nodes affects packet collision while other parameters have little affects. The quantum time and CBR rate affect packet delivery ratio more than density of nodes. However, the changing of parameters has almost no effect on the packet delivery ratio but density of nodes has a direct impact on packet delivery ratio.

6. CONCLUSION

The packet scheduling algorithm is proposed to reduce packet collision in wireless sensor networks. All packets in application and network layers are scheduled before forwarding to the data link layer. This can reduce the wasted packets. The proposed performance prediction model is able to help the developers to obtain the packet collision rate and the packet delivery ratio when they employ the packet scheduling algorithms in their applications.

The prediction models predict the packet collision rate and packet delivery ratio with accuracy 97.92% and 72.16%, respectively. The PSA reduces the packet collision 5% for the high density of node and up to 60-70% for the low density of nodes. On the other hand, the PSA decreases the packet delivery ratio 15% for the low density of nodes. On the hand, the PSA increases PDR up to 50% when CBR rate is more than 150 kbps and density of nodes more than 0.05. The PDR is 200% increased for high density of node and CBR rate.

The accuracy of PCR model and PDR model of simple CSMA/CA are 76.43% and 81.11% respectively. The prediction model is accurate to 77.64% and 80.05% for PSA and PDR, respectively.

ACKNOWLEDGEMENT

REFERENCES

[1]   F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci. A survey on sensor networks. IEEE Communications Magazine, 40(8):102–105, 2002.

[2]   J. Yick, B. Mukherjee, and D. Ghosal. Wireless sensor network survey. Computer Networks, 52:2292–2330, August 2008.

[3]   L. Zhaohua and G. Mingjun. Survey on network lifetime research for wireless sensor networks. In Broadband Network Multimedia Technology, 2009. IC-BNMT '09. 2nd IEEE International Conference on, p. 899 –902, October. 2009.

[4]   A. Bachir, M. Dohler, T. Watteyne, and K.K. Leung. Mac essentials for wireless sensor networks. Communications Surveys Tutorials, IEEE, 12(2):222 –248, quarter 2010.

[5]   A. El-Hoiydi. Spatial tdma and csma with preamble sampling for low power ad hoc wireless sensor networks. In Computers and Communications, 2002. Proceedings. ISCC 2002. Seventh International Symposium on, p. 685 – 692, 2002.

[6]   W. Ye, J. Heidemann, and D. Estrin. Medium access control with coordinated adaptive sleeping for wireless sensor networks. IEEE/ACM Trans. Networks, 12:493–506, June 2004.

[7]   T. van 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, p. 171–180, 2003.

[8]   Y. Peng, B.H. Soong, and L. Wang. Broadcast scheduling in packet radio networks using mixed tabu-greedy algorithm. Electronics Letters, 40(6):375 – 376, March 2004.

[9]   G. Wang and N. Ansari. Optimal broadcast scheduling in packet radio networks using mean field annealing. Selected Areas in Communications, IEEE Journal on, 15(2):250 –260, February 1997.

[10] S. Salcedo-Sanz, C. Bousono-Calzon, and A.R. Figueiras-Vidal. A mixed neural-genetic algorithm for the broadcast scheduling problem. Wireless Communications, IEEE Transactions on, 2(2):277 – 283, March 2003.

[11] J. Yeo, H. Lee, and S. Kim. An efficient broadcast scheduling algorithm for tdma ad-hoc networks. Computers and Operations Research, 29(13):1793 – 1806, 2002.

[12] H. Shi and L. Wang. A hybrid neural network for optimal tdma transmission scheduling in packet radio networks. In Neural Networks, 2005. IJCNN '05. Proceedings. 2005 IEEE International Joint Conference on, volume 5, p. 3210 – 3213 vol. 5, July-4 August. 2005.

[13] I. Ahmad, B. Al-Kazemi, and A. Shoba Das. An efficient algorithm to find broadcast schedule in ad hoc tdma networks. Journal of Computer System, Networks, and Communication, 2008:12:1–12:10, January 2008.

[14] I. Ahmad and A. Shoba Das. A heuristic algorithm for the minimization of incompletely specified finite state machines. Computers & Electrical Engineering, 27(2):159 – 172, 2001.

[15] C. Jandaeng, W. Suntiamontut, and N. Elz. Throughput improvement of collision avoidance in wireless sensor networks. In Wireless Communications Networking and Mobile Computing (WiCOM), 2010 6th International Conference on, pages 1 –5, sept. 2010.

[16] J. Zheng and Myung J. Lee. A comprehensive performance study of ieee 802.15.4. Sensor Network Operations, pages 218–237, 2006.

[17] Ieee standard 802.15.3, wireless medium access controls (mac) and physical layer (phy) specifications for high rate wireless person area networks (wpans), September 2003.

[18] C. Perkins, E. Belding-Royer, and S Das. Ad hoc on-demand distance vector (aodv) routing. RFC 3561, July 2003.

[19] Chipcon as smartrf cc2420 preliminary datasheet (rev 1.2). Website, 2009. http://inst.eecs.berkeley.edu/~cs150/Documents/CC2420.pdf.

[20] S. Chatterjee and A.S Hadi. Regression Analysis by Example. Wiley Interscience, 4th edition, 2006.

Authors

Chaiyut Jandaeng is a lecturer at the School of Informatics of Walailuk University. He completed the B.Sc and M.Sc. in Computer Science, PSU. He is a Ph.D candidate in Computer Engineering at Prince of Songkla University, Thailand. He was a teacher assistance and guest lecturer in information technology after completing masters, before beginning the doctoral research. Areas of interest are computer networks security, and algorithm and programming technique in resource constrained devices.

Wannarat Suntiamorntut received the Ph.D Degrees from the University of Manchester, UK. She was a research assistant at the School of Computer Science, the University of Manchester for 3 year, while she was doing the Ph.D. program. Since 1999, she has been a lecturer at the computer engineering department at Prince of Songkla University, Hat Yai Thailand and is now assistant professor of computer engineering department. In 2008, she became the Director of Collaborative Research Unit in Wireless Sensor Network (CRU-WSN), a joint collaboration between National Electronics and Computer Technology Center and Prince of Songkla University.

Nittida Elz received her B.Sc. degree in Mathematics from Prince of Songkla University, Thailand. She then received her M.Sc. degrees in Computer Science from Chulalongkorn University, Thailand and The University of Melbourne, Australia, followed by a Ph.D. degree in Computer Engineering from La Trobe University, Melbourne, Australia. Her research interest is network management and security.