PDMAC-SIC: Priority-based Distributed Low Delay MAC with Successive Interference Cancellation for Industrial Wireless Networks

Yida Xu [1, 2], Qi Wang [1], Jianmin Liu [1, 2], Chentao He [1, 2], Boyu Diao [1], Yongjun Xu [1]
[1] *Institute of Computing Technology Chinese Acadamy of Sciences,* Beijing, China
[2] *University of Chinese Academy of Sciences,* Beijing, China
xuyida@ict.ac.cn, wangqi08@ict.ac.cn

***Abstract*—Communications in industrial applications like wireless factory automation demands different timing requirements. Providing timely medium access of the critical traffic and its prioritization over regular traffic is a significant challenge in industrial wireless networks. Successive Interference Cancellation (SIC) technique is an effective way to decrease access delay by allowing multiple transmissions concurrently. A series of novel Medium Access Control (MAC) protocols are proposed to differentiate access delay for various traffic types or only exploit SIC for unique traffic type. However, to the best of our knowledge, this work is the first priority-based distributed MAC protocol that employs SIC (PDMAC-SIC) to provide low delay and accommodate different types of traffic for industrial wireless networks. There are two major contributions of our work: first, an extra power contention procedure other than traditional RTS/CTS contention in CSMA/CA is introduced in our PDMAC-SIC. This power contention procedure allows multiple transmitters to access the same channel simultaneously and thus access delay is decreased. Second, PDMAC-SIC is modeled by Markov chain and then the access delay is minimized by optimizing the size of power contention window. Our analytical model is verified through simulation. Results reveal that PDMAC-SIC performs better on access delay and packet loss rate than the existing good performing priority based CSMA/CA.**

Keywords—medium access control, successive interference cancellation, medium access priority, access delay, Markov chain

I. INTRODUCTION

Due to the lower cost and easier deployment, wireless networks for monitoring and control solutions have been applied widely in industry. In these industrial wireless networks, data traffic is divided into different categories: emergency traffic and regular traffic [1]. The category of traffic is assigned by industrial applications. Emergency traffic, e.g. control signal and accidental event signal, contains emergent information and thus needs to be transmitted faster than regular traffic. Therefore, priority system is essential to distinguish emergency traffic from regular traffic. High priority packets that contain emergency data demand low access delay and packet loss rate. The Medium Access Control (MAC) protocols that manage the access to the shared wireless channel are important to improve delay performance of emergency traffic. MAC protocols based on Carrier Sense Multiple Access / Collision Avoidance (CSMA/CA) gain lots of attention, because CSMA/CA can provide low delay and is adapted for burst traffic in industrial wireless networks [2].

Successive Interference Cancellation (SIC) is a promising technique to improve the performance of wireless networks by increasing spectrum efficiency [3]. The idea of SIC is to decode multiple signals sequentially by subtracting interference due to the decoded signals before decoding other signals. SIC relies on imbalance of the received powers from different signals that come from different transmitters. Quite a few works have studied exploiting SIC on distributed random accessing MAC algorithms [4-7]. [4] finds out that traditional CSMA rarely effectively exploits the new transmission opportunities that SIC provides. [5] formulates the problem of determining the optimal probability distribution associating with power levels required by SIC. [6] presents a novel protocol based on multi-user information theory called CSMA k-SIC, which shows significant improvement compared to CSMA. [7] proposes a cross-layer optimization, involving physical, MAC and network layers to study a throughput maximization problem. However, how to differentiate delay performance between regular traffic and emergency traffic remains an open problem for exploiting SIC in MAC protocols.

As a widely used algorithm, performance including access delay of CSMA/CA has been thoroughly studied [8]. The Enhanced Distributed Channel Access (EDCA) [10] was proposed to provide traffic differentiation for CSMA/CA by enabling high priority packets to access channel more frequently. In recent years, several works that studied distributed MAC with traffic differentiation have been published [10-12]. However, algorithms in these works allow only one transmitter to access a given wireless channel at any instance, which limit the performance of the network.

In this paper, we propose a priority-based distributed low delay MAC scheme that adopts SIC (PDMAC-SIC) for industrial wireless networks. SIC technique enables a node to receive signals from multiple transmitters simultaneously and thus improves the delay performance of network. In PDMAC-SIC, high priority packets are allowed to transmit using either high power or low power, while low priority packets are only allowed to transmit using low power. Thus, high priority packets have more opportunities to access media than low priority packets. PDMAC-SIC consists of two phases. The first phase is named RTS contention phase, which is a CSMA/CA like contention phase for transmitters to transmit Request To Send (RTS) packets. When an RTS packet reaches its destination, the second phase called power contention phase is triggered. Power contention phase gives transmitters that fail RTS contention an extra chance to transmit data. Winners of RTS contention phase and power contention phase are allowed to transmit data simultaneously. Because the transmitting power is regulated in the contention process, receiver can decode the mixed signals. PDMAC-SIC is modeled by Markov chain and then the access delay is optimized. In performance evaluation, we show that PDMAC-SIC performs better on access delay and packet loss rate than the existing good performing priority assisted CSMA/CA.

The contributions of this paper are as follows: (1) PDMAC-SIC is the first priority based distributed MAC employing SIC, which improves access delay for industrial wireless networks. (2) Performance of PDMAC-SIC is modeled by Markov chain and then access delay is minimized. The analytical model is verified by simulations and the results show that PDMAC-SIC performs better than the existing priority assisted CSMA/CA.

The rest of this paper is organized as follows. Section 2 introduces the network model and PDMAC-SIC. Performance of PDMAC-SIC is analyzed in Section 3. Evaluation results are shown in Section 4, and conclusions are summarized in the last section.

II. Network Model and Algorithm Description

We suppose that the network is constituted by a group of wireless nodes that share the same transmitting and receiving capabilities in this paper. Data traffic is divided into two categories: regular traffic and emergency traffic. Priority of data packets is assigned according to the traffic type, i.e. low priority packets for regular traffic and high priority packet for emergency traffic. Transmitters are able to switch their transmitting power between two levels. If a pair of nodes can communicate with each other by using low transmitting power, these two nodes are neighbors. Meanwhile, we assume that the transmitting rate is stable when the transmitting power is changed.

First of all, interference from non-neighbor nodes needs to be addressed. To simplify calculation, we use Signal to Interference plus Noise Ratio (SINR) model described in equation (2.1) to decide whether the signal can be decoded successfully.

$$SINR = \frac{P_r}{n_0 + P_I} \geq \gamma \tag{2.1}$$

where γ is the threshold for correct reception of signals and the symbols of P_r, n_0 and P_I represent power of received signal, noise and interference, respectively.

In this network, SIC technique is implemented on receivers, so mixed signals can be decoded by the receivers if SINR of the signals meets the equation (2.2).

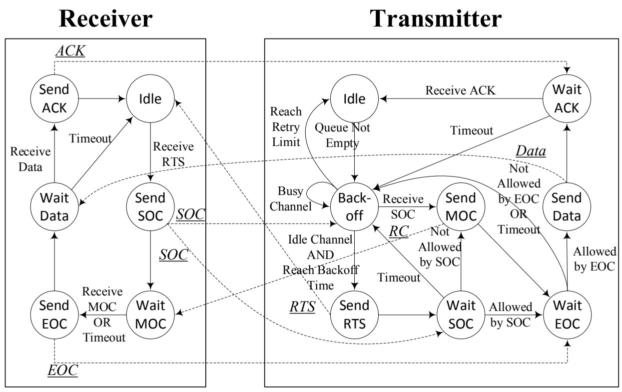


Fig. 1. Graph of state transition of PDMAC-SIC.

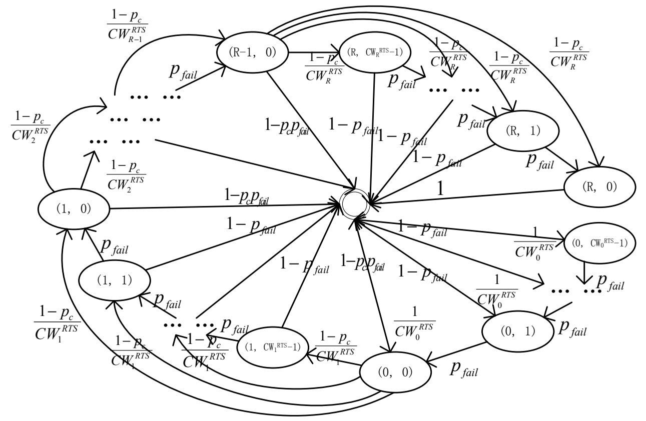


Fig. 2. Graph of state transition of PDMAC-SIC

$$\begin{aligned} &\frac{P_r^H}{n_0 + P_I' + P_r^L} \geq \gamma, \\ &\frac{P_r^L}{n_0 + P_I'} \geq \gamma, \end{aligned} \tag{2.2}$$

where P_r^H and P_r^L refer to power of received signal that are transmitted using high transmitting power and low transmitting power, respectively. The symbol P_I' represents power of interference generated by other transmitting signals when SIC is employed in the network.

Only interference from non-neighbor nodes needs to consider since MAC algorithm addresses interference from neighbor nodes. Because all the nodes in this network transmit in the same channel, the power of interference signal is the summation of power from all non-neighbor nodes, as equation (2.3):

$$P_I = \sum_{i \in \bar{N}} P_r(i) \tag{2.3}$$

where $\bar{N}$ denotes the set of non-neighbor nodes. In PDMAC-SIC process, only high priority packets have the opportunity to be transmitted using high power. Therefore, the proportion of high power signals in total signals is no more than the proportion of high priority packets in total packets. Let α denote the Proportion of High Priority Packets (PHPP). P_I' is formulated as follows:

$$P_I' \leq l \cdot \sum_{i \in \bar{N}} P_r(i) + \alpha \cdot k \cdot l \cdot \sum_{i \in \bar{N}} P_r(i) \tag{2.4}$$

where

$$k = \frac{P_r^H}{P_r^L} \tag{2.5}$$

denotes the lower bound of ratio of high received power to low received power, and

$$l = \frac{P_r^L}{P_r} \tag{2.6}$$

denotes the lower bound of ratio of low received power to received power without SIC.

Consequently, l is derived by P_r, n_0, γ according to equations (2.1)-(2.6). For an instance, given $P_r = 2$, $n_0 = 1$, $\gamma = 1$, then $P_l = 1$, $l = \frac{2}{2-\alpha}$ are derived. The result means that if PHPP $\alpha = 10\%$, the low received power when SIC is enabled is only 5% higher than the received power when SIC is disabled. With the propagation model of free-space:

$$P_r = d^{-\lambda} P_t \quad (2.7)$$

transmitting power is a linear function of receiving power, so the low transmitting power available for SIC is also only 5% more than transmitting power without SIC. When P_r, n_0, γ are given, l can be obtained and then the lower bound of P_r^H can also be obtained according to equation (2.2) and (2.6). With acquired P_r^L and P_r^H, interference from non-neighbor nodes is limited as the same as in the situation without SIC.

In this paper, PDMAC-SIC is proposed to decrease access delay. All the control packets in PDMAC-SIC are transmitted using low transmitting power. Process of PDMAC-SIC consists with two phases. In the first phase, transmitters act as CSMA/CA: listening to the channel and contending channel by sending RTS. The first phase is named as "RTS contention phase". When the receiver receives an RTS, the second phase called "power contention phase" begins. An RTS for high priority packet triggers low-power contention that allows an access from both high priority and low priority packets, while an RTS for low priority packet triggers high-power contention that only allows an access from high priority packets. High-power contention and low-power contention experience the same process except different contention window size and priority of packet allowed participating.

Table 1. Power Allocation Table

Priority of packet wins RTS contention	Priority of packet wins power contention	Power for packet wins RTS contention	Power for packet wins power contention
High priority	None	Low power	None
High priority	High priority	Low power	High power
High priority	Low priority	High power	Low power
Low priority	None	Low power	None
Low priority	High priority	Low power	High power

The process of PDMAC is outlined as bellow, while Fig. 1 gives a visual process of PDMAC-SIC where solid arrows show state transitions and dash arrows show packet transmissions.

1) When a transmitter has a data and sensors an idle channel, it begins RTS contention. The trransmitter sends RTS packet after waiting random backoff time-slot. Different to CSMA/CA, RTS in PDMAC-SIC contains the priority of data packet.

2) When the receiver receives an RTS, it broadcasts Start Of Contention (SOC) packet to its neighbors, which means that the power contention begins. If the receiver receives an RTS for high priority packet, a low-power contention begins. Otherwise, a high-power contention begins. Transmitters that have data packets with proper priority and have not sent their RTS are permitted to attend the power contention. Transmitters with high priority packets are allowed to attend both high-power contention and low-power contention, while transmitters with low priority packets are only allowed to attend low-power contention.

3) Transmitters that gain the permission from step 2 wait random time-slots selected in $\left[0, CW^{power}\right]$ and then send Message Of Contention (MOC) packet to the receiver. CW^{power} is the general name for CW^{power}_{High} and CW^{power}_{Low} that are contention window sizes for high-power contention and low-power contention, respectively.

4) The receiver broadcasts End Of Contention (EOC) packet to its neighbors after receiving MOC. In the MOC, transmitting power is allocated as Table 1.

5) When the data is received successfully, the receiver sends ACKnowledgement (ACK) packet to the corresponding transmitters.

III. Analytical Model of PDMAC-SIC

In PDMAC-SIC, power contention window size CW^{power}_{High} and CW^{power}_{Low} are key parameters that affect network performance. Because CW^{power}_{High} and CW^{power}_{Low} are calculated in the same way, CW^{power} is used as the general symbol for them to simplify description. In this paper, access delay represents the duration from the time that the data packet reaches the head of transmitting queue to the time that ACK of the data packet is received.

Let $r(t)$ and $b(t)$ be the stochastic processes that describe the number of retries of attempting to send RTS and the backoff timer for a given transmitter at slot time t, respectively. Markov chain that used to represent the process $\{r(t), b(t)\}$ is shown in Fig. 2. Let p^{RTS}_{fail} denote the probability that a transmitter sends an RTS and a collision occurs in a randomly given time slot during the RTS contention phase. Let p^{power}_{fail} denote the probability that a transmitter fails in the power contention. $p_{fail} = p^{RTS}_{fail} \cdot p^{power}_{fail}$ is the probability that a transmitter fails in both RTS contention phase and power contention phase.

Let $CW^{RTS}_r = 2^r \cdot CW^{RTS}_0$ denote size of the RTS contention window for a transmitter that has retried to send RTS for r times. The circle in the center of Fig. 2 reprsents the initial state of the transmitter, and state $s_{r,b}$ is the state that the transmitter has retried to send RTS for r times and its RTS contention backoff timer counts to b. To simplify calculation, we only consider saturated traffic network in this paper. The stationary distribution of the Markov chain is:

$$p\left(s^*_{r,b}\right) = \lim_{t\to\infty} P\{r(t) = r, b(t) = b\} \quad (3.1)$$

where $p\left(s^*_{r,b}\right)$ is derived from the matrix of state transition. Let p_{RTS} denote the probability that a transmitter sends a RTS packet in a time slot in the RTS contention phase. p_{RTS} is given by:

$$p_{RTS} = \sum_{r=0}^{R} p\left(s_{r,0}^{*}\right) \quad (3.2)$$

When all transmitters transmit RTS with the same probability, the probability of RTS collision is:

$$p_{fail}^{RTS} = 1 - \left(1 - p_{RTS}\right)^{n-1} \quad (3.3)$$

where n denotes the number of transmitters in the contention.

Let p_{tr}^{RTS} denote the probability that any RTS is transmitted in a random time slot, and p_{suc}^{RTS} denote the probability that an RTS is successfully transmitted in a random time slot and thus power contention is triggered. p_{tr}^{RTS} and p_{suc}^{RTS} are given by:

$$p_{tr}^{RTS} = 1 - \left(1 - p_{RTS}\right)^{n} \quad (3.4)$$

$$p_{suc}^{RTS} = \frac{n \cdot p_{RTS} \cdot \left(1 - p_{RTS}\right)^{n-1}}{1 - \left(1 - p_{RTS}\right)^{n}} \quad (3.5)$$

The average access delay is given by:

$$E\left[delay\right] = E\left[c_{slot}\right] \cdot E\left[d_{slot}\right] \quad (3.6)$$

where c_{slot} denotes the number of time slots in RTS contention phase required to transmit a packet successfully, and d_{slot} denotes the duration of one time slot in RTS contention phase. If no signals listened by the transmitter, its backoff timer is not interrupted. Therefore, duration of an undisturbed time slot d_{slot}^{u} is just as the preset duration of time slot, i.e. $d_{slot}^{u} = T_{slot}$. Let d_{slot}^{con} denote duration of time slot when power contention is triggered, which is given by:

$$\begin{aligned} E\left[d_{slot}^{con}\right] &= T_{DIFS} + T_{RTS} + T_{SIFS} + T_{SOC} + E\left[d_{con}\right] \\ &+ T_{SIFS} + T_{EOC} + T_{SIFS} + T_{Data} + T_{SIFS} + T_{ACK} + T_{slot} \end{aligned} \quad (3.7)$$

where $E\left[d_{con}\right]$ is the average duration of power contention phase. $T_{DIFS}, T_{SIFS}, T_{RTS}, T_{SOC}, T_{EOC}, T_{Data}$ and T_{ACK} are time needed for DIFS, SIFS, transmitting RTS, transmitting SOC, transmitting data and transmitting ACK, respectively. $E\left[d_{con}\right]$ is given by:

$$\begin{aligned} E\left[d_{con}\right] &= \sum_{i=1}^{R} p_{suc}^{power}\left(1 - p_{suc}^{power}\right)^{i-1} \\ &\cdot\left(\left(i-1\right)\left(CW^{power} + T_{DIFS} + T_{RTS} + T_{SIFS} + T_{SOC}\right) + \frac{CW^{power}+1}{2}\right) \end{aligned} \quad (3.8)$$

where the probability that any transmitter wins in the power contention is:

$$\begin{aligned} p_{suc}^{power} &= \sum_{i=1}^{n-1} \frac{\left(-1\right)^{i-1} i! \binom{n}{i} \binom{CW^{power}}{i} \left(CW^{power} - i\right)^{n-i}}{\left(CW^{power}\right)^{n}} \\ &+ \frac{\left(-1\right)^{n-1} n! \binom{CW^{power}}{n}}{\left(CW^{power}\right)^{n}} \end{aligned} \quad (3.9)$$

Let d_{slot}^{fail} denote duration of time slot when no RTS is transmitted successfully and power contention is not triggered, which is given by:

$$E\left[d_{slot}^{fail}\right] = T_{DIFS} + T_{RTS} + T_{SIFS} + T_{slot} \quad (3.10)$$

The average duration of a time slot is:

$$\begin{aligned} E\left[d_{slot}\right] &= \left(1 - p_{tr}^{RTS}\right) \cdot d_{slot}^{u} + p_{tr}^{RTS} \cdot p_{suc}^{RTS} \cdot E\left[d_{slot}^{con}\right] \\ &+ p_{tr}^{RTS} \cdot \left(1 - p_{suc}^{RTS}\right) \cdot E\left[d_{slot}^{fail}\right] \end{aligned} \quad (3.11)$$

The average number of time slot experienced for a successfully transmitted packet is:

$$E\left[c_{slot}\right] = \sum_{r=0}^{R} E\left[c_{slot} \mid r\right] \quad (3.12)$$

where $E\left[c_{slot} \mid r\right]$ denotes the average number of time slots experienced if the packet is successfully transmitted just after retry r times. $E\left[c_{slot} \mid r\right]$ is given as:

$$E\left[c_{slot} \mid 0\right] = \frac{\sum_{b=0}^{CW_0^{RTS}} b \cdot p\left(s_{0,b}^{*}\right)}{\sum_{b=0}^{CW_0^{RTS}} p\left(s_{0,b}^{*}\right)} \quad (3.13)$$

$$E\left[c_{slot} \mid r\right] = \sum_{i=0}^{r-1} E\left[c_{slot} \mid i\right] + \frac{\sum_{b=0}^{CW_r^{RTS}} b \cdot p\left(s_{r,b}^{*}\right)}{\sum_{b=0}^{CW_r^{RTS}} p\left(s_{r,b}^{*}\right)}, r > 0 \quad (3.14)$$

The relation between $E\left[delay\right]$ and $p\left(s_{r,b}^{*}\right)$ is derived from equations (3.6) – (3.14). Then, from equations (3.1), the optimized CW^{power} to get the minimum access delay is calculated.

IV. Performance Evaluation

The industrial wireless network is simulated using the discrete event-driven network simulator WSNet [13]. The network is composed of several uniformly distributed transmitters and one receiver as shown in Fig. 3, which is a typical topology in industrial wireless networks. Traffic of the network is saturated. Category of traffic is generated randomly with certain probability, which means proportion of emergency traffic is fixed in each simulation. Other system parameters in the simulation are shown in Table 2.

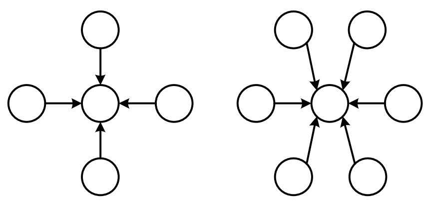
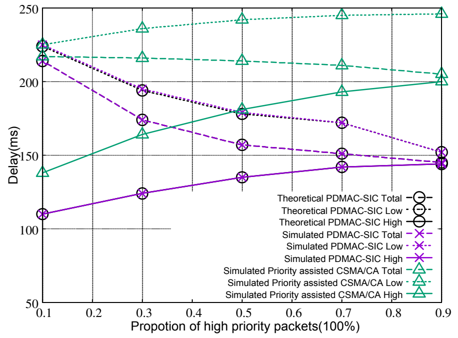
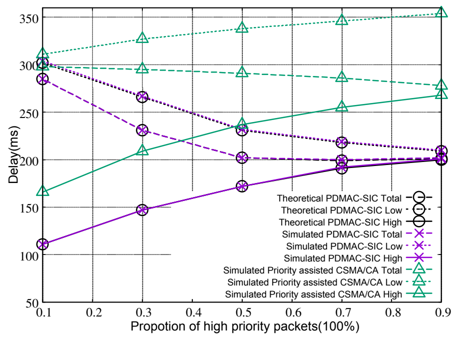


Fig. 3. Topology of network in the simulation



(a) 4 transmitters



(b) 6 transmitters

Fig. 4. Comparison of access delay

Fig. 4, Fig. 5 and Fig. 6 respectively plot access delay, packet loss rate and energy efficiency against Proportion of High Priority Packets (PHPP) from PDMAC-SIC compared to results from priority assisted CSMA/CA [10]. Each simulation runs for 100 times.

"Best effort" and "Voice" are selected as traffic types for low priority and high priority packets. Contention window size for different traffic types in the priority assisted CSMA/CA is listed in [10]. Other parameters remain the same as shown in Table 2.

Table 2. System Parameters.

Parameter	Value	Parameter	Value
Area size	2km x 2km	Low transmitting power	21.5mW
Number of transmitters	4, 6	High transmitting power	43mW
Radio propagation model	Free space	Receiving power	7mW
Battery model	Linear	CW_0^{RTS}	8
Number of date packets	1000	R	7
Size of data packets	10000bit	T_{SIFS}	10 µs
Size of control packets	168bit	T_{DIFS}	50 µs
Bit rate	120000bit/s	T_{slot}	20 µs

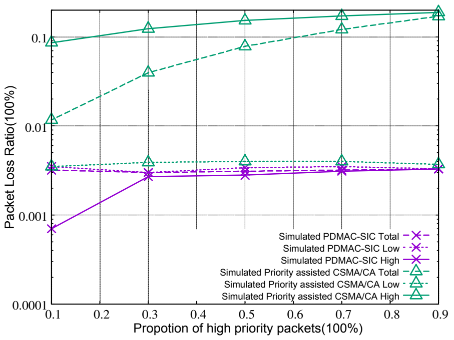
Fig. 4 indicates that PDMAC-SIC reduces access delay of high priority packets effectively. Access delay of high priority packets in PDMAC-SIC is about 30% shorter than in priority assisted CSMA/CA. An almost exact match between analytical and simulation results validates our analytic model of access delay. In both the case of PDMAC-SIC and conventional priority assisted CSMA/CA, high priority packets experience shorter delay than low priority packets. The conventional priority assisted CSMA/CA assigns small contention windows to high priority packets in order to achieve short access delay. When PHPP are relative low, e.g. 10%, priority assisted CSMA/CA acquires decent access delay for both high priority and low priority packets. In this situation, PDMAC-SIC achieves rather shorter access delay for high priority packets. In PDMAC-SIC, high priority packets and low priority packets contend fairly in RTS contention phase. Thus, when the most of packets are low priority packets, most RTS contentions are won by low priority packets. RTS for low priority packet triggers high-power contention that only allows high priority packets to contend. Since high priority packets are relative few, a high priority packet probably wins the first accessed high-power contention and thus achieves quite low access delay. When PHPP increases, contentions become more intense for conventional CSMA/CA due to the smaller contention window size for high priority packets. More intense conventions decrease the probability for packets to win the channel, and thus lengthen access delay for both low priority and high priority packets. When PHPP increases from 10% to 90%, access delay of high priority packets increases by about 60ms and access delay of low priority packets also increases by about 30ms.

Contrarily, in PDMAC-SIC, increase of PHPP causes more low-power contentions triggered, which give an extra chance to both low priority and high priority packets for parallel transmission. When PHPP increases from 10% to 90%, access delay of high priority packets increases by about 50ms but access delay of low priority packets also decreases by about 100ms.

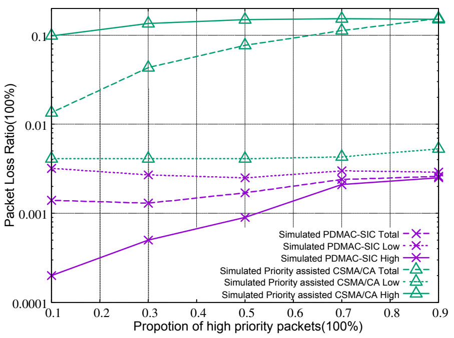
Fig. 5 plots different Packet Loss Rate (PLR) against PHPP. PDMAC-SIC contains the similar RTS contention phase as conventional CSMA/CA, which guarantees that PLR of PDMAC-SIC is not worse than PLR of conventional CSMA/CA. In conventional CSMA/CA, smaller contention window for high priority packets results in higher PLR. The result shows that PLR of high priority packets is 10 times more than PRL of low priority packets in conventional CSMA/CA. In PDMAC-SIC, low priority and high priority packets act equally in RTS contention phase. In addition, high priority packets that are able to participate in high-power contentions have better chance to be transmitted before reaching the limit of retry. Compared to the priority assisted CSMA/CA, PDMAC-SIC improves access delay without deteriorating PLR for high priority packets. The reason is that multiple packets can be received parallel with SIC technique and thus the utility of wireless channel increases.

Fig. 6 shows that power consumption in PDMAC-SIC for transmitting a packet increases as PHPP increases. The reason is that SIC technique requests more power for transmitting signals. When 90% of the packets are high priority packets, PDMAC-SIC costs about 30% more energy compared to conventional CSMA/CA. Meanwhile, when 10% of the packets are high priority packets, PDMAC-SIC costs only 10% more energy.

The results indicate that PDMAC-SIC exploits SIC technique effectively. Compared to conventional priority assisted CSMA/CA, PDMAC-SIC performs much better on both access delay and PLR at the expense of a little more energy consumed.

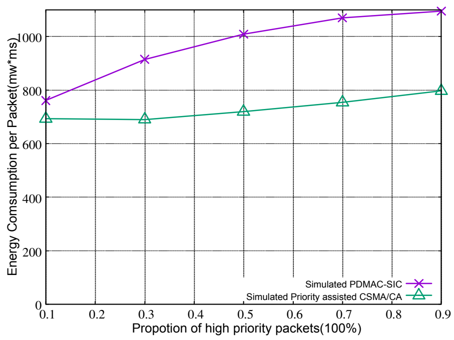


(a) 4 transmitters

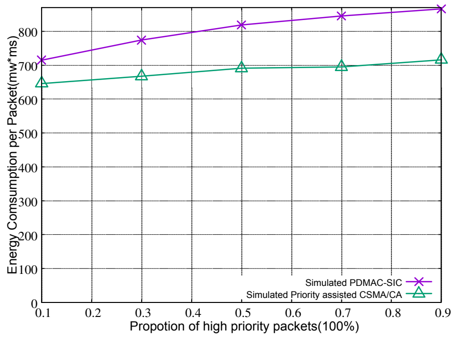


(b) 6 transmitters

Fig. 5. Comparison of packet loss rate



(a) 4 transmitters



(b) 6 transmitters

Fig. 6. Comparison of energy consumption

V. CONCLUSION

In this paper, we propose a priority-based distributed low delay MAC scheme that employs SIC (PDMAC-SIC) for industrial wireless networks. PDMAC-SIC provides opportunities for parallel transmissions by introducing additional contention phase named "power contention phase". In PDMAC-SIC, high priority packets that obtain more opportunities to participate in power contention phase reach lower access delay and packet loss rate than low priority packets. PDMAC-SIC is modeled by Markov chain and the access delay of network is optimized. The analytical model is validated by simulation results. Compared to the existing priority assisted good performing CSMA/CA, PDMAC-SIC performs better in access delay and packet loss rate with the trade-off of limited energy consumption.

ACKNOWLEDGMENT

This work was supported in part by the National Natural Science Foundation of China (NSFC) under Grant No.61602447 and No.61702487

REFERENCES

[1] "ISA100.11a Standard", Wireless Systems for Industrial Automation: Process Control and Related Applications, 2009.

[2] Bharghavan V, Demers A, Shenker S, et al. MACAW: a media access protocol for wireless LAN's[C]. Conference on Communications Architectures. 1994.

[3] Islam S M R , Avazov N , Dobre O A , et al. Power-Domain Non-Orthogonal Multiple Access (NOMA) in 5G Systems: Potentials and Challenges[J]. IEEE Communications Surveys & Tutorials, 2016, PP(99):1-1.

[4] Lv S, Zhuang W, Wang X, et al. A performance study of CSMA in wireless networks with successive interference cancellation[C]. IEEE International Conference on Communications. 2012.

[5] Zou M, Chan S, Hai L V, et al. Throughput Improvement of 802.11 Networks Via Randomization of Transmission Power Levels[J]. IEEE Transactions on Vehicular Technology, 2015, 65(4):1-1.

[6] Sankararaman A, Baccelli F. CSMA k-SIC — A class of distributed MAC protocols and their performance evaluation[C]. Computer Communications. 2015.

[7] Jiang C, Shi Y, Qin X, et al. Cross-Layer Optimization for Multi-Hop Wireless Networks With Successive Interference Cancellation[J]. IEEE Transactions on Wireless Communications, 2016, 15(8):5819-5831.

[8] Chatzimisios P . IEEE 802.11 packet delay-a finite retry limit analysis[C]. IEEE Global Telecommunications Conference. 2009.

[9] IEEE standard for information technology–local and metropolitan area networks–specific requirements–part 11: Wireless LAN medium access control (MAC) and physical layer (PHY) specifications - amendment 8: Medium access control (MAC) quality of service enhancements, IEEE std 802.11e- 2005 (amendment to IEEE std 802.11, 1999 edition (reaff 2003), 2005. 1–212.

[10] Shakir M , Rehman O , Rahim M , et al. Performance Optimization of Priority Assisted CSMA/CA Mechanism of 802.15.6 under Saturation Regime[J]. Sensors, 2016, 16(9).

[11] Syed I , Shin S H , Roh B H , et al. Performance Improvement of QoS-Enabled WLANs Using Adaptive Contention Window Backoff Algorithm[J]. IEEE Systems Journal, 2017, PP(99):1-11.

[12] Sanabria-Russo L , Bellalta B . Traffic Differentiation in Dense Collision-free WLANs using CSMA/ECA[J]. Ad Hoc Networks, 2018: S1570870518300635.

[13] wsnet simulator: wsnet.gforge.inria.fr