

MAC Throughput Improvement Using Adaptive Contention Window

Chun-Liang Lin, Wei-Ting Chang, Min-Huei Lu

Department of Electrical Engineering, National Chung Hsing University, Taichung, Taiwan
Email: chunlin@dragon.nchu.edu.tw

Received 2 January 2015; accepted 19 January 2015; published 23 January 2015

Copyright © 2015 by authors and Scientific Research Publishing Inc.

This work is licensed under the Creative Commons Attribution International License (CC BY).

<http://creativecommons.org/licenses/by/4.0/>



Open Access

Abstract

HomePlug AV (HPAV) is a standard developed by HomePlug Powerline Alliance (HPA) for power line communication. In HomePlug AV, it uses a technology named Carrier Sense Multiple Access with Collision Avoidance (CSMA/CA) to reduce collision happened in network. However, when network nodes increase, the contention window number may not be wide enough. It will cause collision probability to increase. In this paper, we introduce a new idea of adaptive contention window which will produce suitable contention window under actual network environment. Our method only requires the information of CSMA/CA parameters. It means that one doesn't need to correct the original CSMA/CA procedure but substitutes old parameters by the new ones. Simulation experiments conducted in the network simulator NS3 show that compared with HomePlug AV, our method promotes throughput significantly when the node number increases.

Keywords

CSMA/CA, Contention Window, Power Line Communication

1. Introduction

Power-Line Communication (PLC) is developed rapidly in recent years. Because it doesn't need to build additional transmit channel and exists almost everywhere even in the backward areas, it is mostly recognized as the solution of "last mile" to the current network communication. Since power line was not originally built for information transmission, there are several problems. One of these is the data collision. When channel transmits multiple packets at the same time, packets will collide naturally causing packets to be destroyed. The station has to retransmit packets which will cause throughput decrease in the end. Thus, HomePlug AV [1] uses CSMA/CA [2] to avoid collision happening. The principle is as follows: each time the station tries to send package, it will detect channel's status. If the channel is idle, the station sends the packets. Otherwise, it goes into the contention

mode. This method can avoid too many stations transmitting at a time hence to effectively reduce the chance of data collision.

A PLC architecture consists of head end (HE) and user modem. HE erects at the transformer side and one side is connected to the transmission backbone, like Ethernet. Another side is connected to the power line to receive request from PLC users. The user modem connects to the communication equipment to communicate through the power line. If a PLC modem accesses the Internet, it will receive packets from the head end through the power line.

Although HomePlug AV standard can guarantee high throughput when the number of nodes is very limited, when the node number increases, the value of contention window setting in HomePlug AV will be big which leads throughput to decrease accordingly. In [3], information theory and data mining technique were applied for network traffic profiling. Some research tasks focused on enhancing throughput via the revised MAC layers [3]-[8]. Among which, the authors of [3] proposed an adaptive contention window mechanism for HomePlug AV and verified their designs through experiments. They have conducted an experiment to reach the best successful transmission number in a beacon period. That is, if the number of successful transmits is less than the optimal one, it means that the current contention window size is inappropriate. It will then change it in the next beacon period. In 2011, the authors of [4] found the optimal contention window value for the situation of different nodes. Both of them show that throughput can be improved in an efficient manner if the size of contention window is optimized.

Even though previous research can achieve the desired effect, most of them require information of all stations. That means that they need another bit to transmit that information to each node. Or, they have to modify the protocol. In this research, we propose a new strategy of an adaptive contention window mechanism that doesn't change the original CSMA/CA procedure or require station's information as a basis of modifying the contention window size. Our correction factor includes the CSMA/CA parameters. Soft experiments at the network simulator, NS3, show the improved effect in throughput.

2. Principles of CSMA/CA

For PLC, when a station tries to send a packet, it first checks the channel condition. If the channel is in idle, it transmits data accordingly. Otherwise, it enters the contention mode. The contention period is a region to contend the authority of using channel by other stations.

Before going into the contention period, the station will allocate each packet priority. Packets with higher priority use the channel first. If there are several packets possessing the same priority, it will become the contention period. In priority resolution, HPAV define four parameters, CA0, CA1, CA2 and CA3. CA3 present the highest priority, and CA0 is the lowest, etc. Each station will be defined the packet's priority in two priority regions denoted PRS0 and PRS1 respectively. At each PRSi, the station chooses to send or not to send signals. According to the PRS, we can determine the packet's priority level. Different priority level corresponds to different parameter settings. **Table 1** shows the role of it.

After implementing the setting of priority, each station equips with a specific packet priority level. Higher priority level wins the contention and possesses more chances to send the packet. If there are several stations with the same priority, it will proceed to the next stage-random backoff procedure.

For the random backoff procedure, we first introduce contention window and three counters, backoff counter, deferral counter and backoff procedure counter.

1) Contention window (CW): Contention window is a fixed number defined in **Table 2**. It is used to determine the value of backoff counters.

2) Backoff counter (BC): Backoff counter's value chooses from a random value of contention window. It means that CW is the maximal value of BC could be. BC will be decreased by one, each time when it senses a busy status in the channel until BC equals to zero. The station will transmit the packet.

3) Deferral counter (DC): When the number of users in the local area network is large, using only BC to delay the transmission time is not enough. Deferral counter is used to avoid collision further. At the beginning, DC is fixed. Different CAs have different initial values of DC corresponding to. If the channel status sensed is idle, DC remains unchanged. On the contrary, if it is busy, DC will be decreased by one. When DC reaches to zero, the time contention is declared "fail". It will restart contention again.

4) Backoff procedure counter (BPC): Backoff procedure counter is set to be zero initially. When the station

Table 1. Relationship between PRS and CA.

	Low priority		High priority	
	CA0	CA1	CA2	CA3
PRS0	0	0	1	1
PRS1	0	1	0	1

Table 2. Parameter setting.

BPC	CA	
	CA3, CA2 High priority	CA1, CA0 Low priority
BPC = 0	DC = 0 CW = 7	DC = 0 CW = 7
BPC = 1	DC = 1 CW = 15	DC = 1 CW = 15
BPC = 2	DC = 3 CW = 15	DC = 3 CW = 31
BPC ≥ 3	DC = 15 CW = 31	DC = 15 CW = 63

fails to transmit packets (happened collision). Station will attempt to contend again and BPC will be increased by one. According to different BPC, there are different DC and CW settings to respond.

We express the behavior through Markov chain [4] in **Figure 1**. Each time we begin the backoff procedure by setting the same probability to obtain the value of BC from zero to CW ($=W$). Therefore, each probability at the top statement is $1/W$. At each time slot (defined in HPAV is 35.85 ms), the station will detect the communication channel. The probability is P to sense an “busy”, and it is $1 - P$ to sense a “idle”. When DC or BC is equal to zero but the channel status sensed is still in busy. BPC will be increased by one and BC will be rechosen. That means that there are too many users occupying the channel. Then, DC and BC need to be increased for longer waiting time for collision avoidance. When BC reaches to zero and the channel status sensed is idle, the station can then transmit packets. The complete flow chart is illustrated in **Figure 2**.

The time of successful transmission can be expressed as in **Figure 3**. PRS0 and PRS1 are regions to decide packets priority, a contention region refer to the waiting time of the contended channel before transmitting packets, data transmission is the time of packets sent in the PHY layer; RIFS refers to the time before sending acknowledgement, ACK is the time of transmitting acknowledgement, and CIFS is time before starting the next packet. Among which, PRS0, PRS1, RIFS, ACKS, and CIFS are fixed numbers in HomePlug AV, see **Table 3**.

3. Slot Utilization and Relation with Throughput

Referring to **Figure 3**, we define $r_s(t)$ and $r_c(t)$ respectively, the time required for a successful data transmission and the time for a collision happened during data transmission. The two terms can be expressed by

$$r_s(t) = T_s + \sigma_{\text{all}}, \quad r_c(t) = T_c + \sigma_{\text{all}}$$

where σ_{all} is the total slot time in the contention region and

$$T_s = \text{PRS0} + \text{PRS1} + T_{\text{frame}} + \text{RIFS} + \text{ACK} + \text{CIFS} \quad (1)$$

$$T_c = \text{PRS0} + \text{PRS1} + T_{\text{frame}} + \text{CIFS} \quad (2)$$

If we define P_b as the probability of slot sensed in busy. From [2], P_b can be expressed by

$$P_b = 1 - (1 - \tau)^n \quad (3)$$

where τ is the probability of the station transmitted in a slot and n is the number of stations. Since P_b reflects

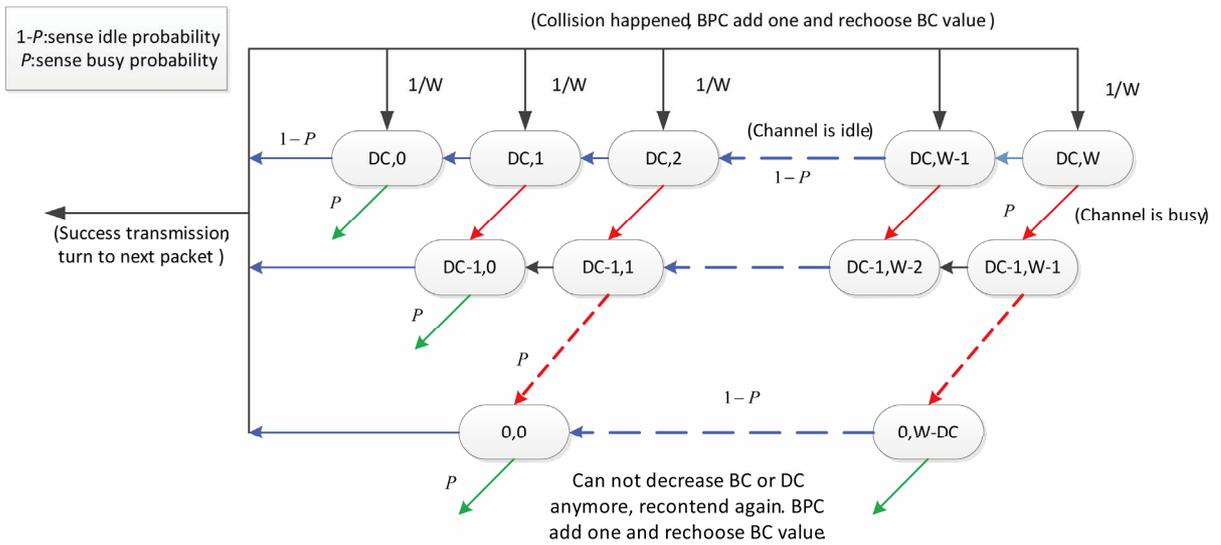


Figure 1. Markov chain of CSMA/CA.

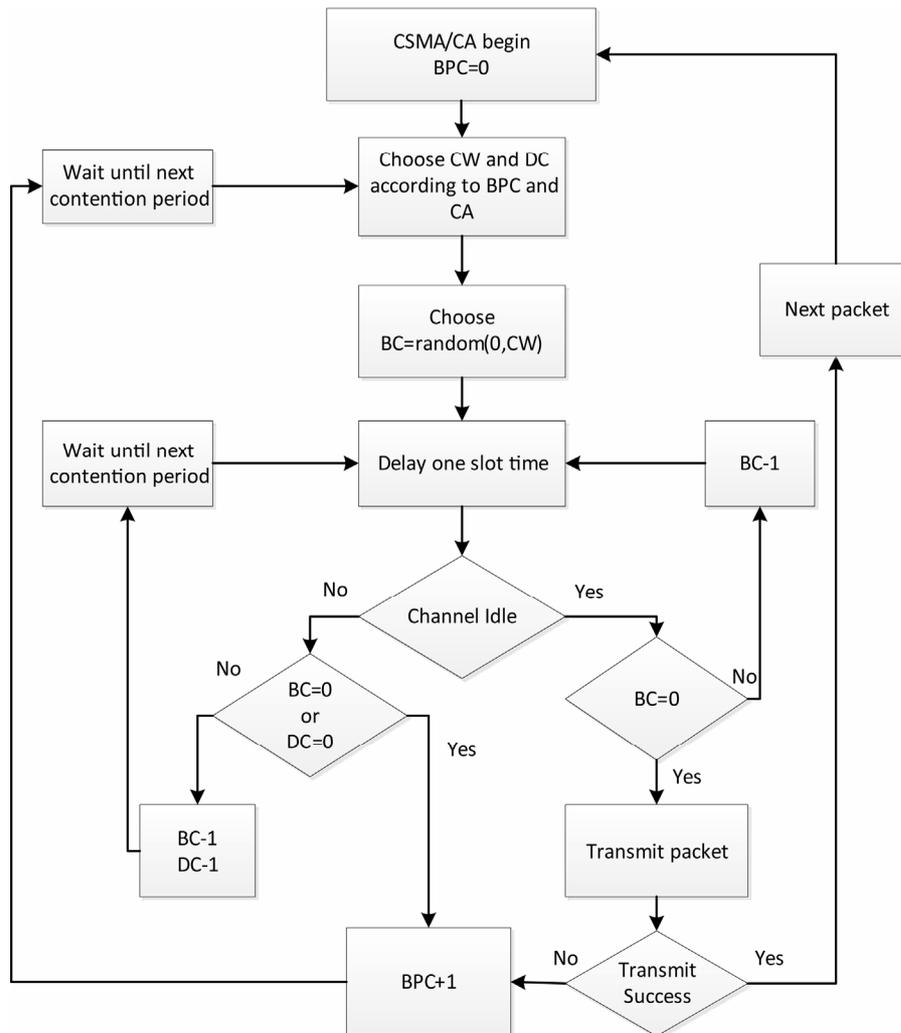


Figure 2. Operational flow chart of the CSMA/CA procedure.



Figure 3. The process for a successful transmission.

Table 3. HPAV system parameters.

Parameters	Time
Beacon period	33.33 ms
CIFS	100 μ s
RIFS	30.72 μ s
PRS0	35.84 μ s
PRS1	35.84 μ s
Slot time	35.84 μ s
MAX_FL	2501.12 μ s
Response tineout	140.48 μ s

the probability of channel in busy status, it can also be defined as

$$\tilde{P}_b = \frac{\text{busy slot number}}{\text{busy slot number} + \text{idle slot number}} \quad (4)$$

This is also referred as “slot utilization” [9].

We conduct several extensive simulation experiments on the Network Simulator 3 (NS3) [10]. NS3 is an event driver developed to mimic network environment. Compared with its predecessor NS2, NS3 is completely developed by C++ and the system architecture is simpler than that of NS2. We record slot utilization every 30 seconds at various node arrangement. There are four CW cases in the simulation. The results are shown in **Figure 4(a)** where T_s defined in Equation (1) is set to be 3500 μ s and the slot time is similar to that defined in **Table 3**.

In **Figure 4(a)**, red line refers to HomePlug AV standard which let $CW = [7 \ 15 \ 31 \ 63]$. Other lines are the situation with different CW values. When CW is set to be larger, the probability for obtaining a larger BC is higher. Correspondingly, its slot utilization is comparatively smaller. However, even the CW setting is as such, the slot utilization increases continuously. This fact inspires us to consider whether one can use slot utilization to relate the congestion level. Another simulation been conducted is to observe the variation of throughput at the same setting, illustrated as in **Figure 4(b)**.

In **Figure 4(b)**, HomePlug AV standard has the highest throughput when the number of contention nodes is small. However, when the node number increases, the throughput will decrease because of the higher collision probability. On the contrary, $CW = [63 \ 127 \ 255 \ 511]$ setting possesses the worst throughput at the beginning. That’s because it wastes too much time in the idle slot. However, it will cause low collision probability when the node number becomes larger. This why it possesses highest throughput when the node number is more than 35.

Therefore, if one can assign the system a low CW setting when the node number is few and a high CW when the node number is large then better throughput could be expected. This is the basic idea behind the adaptive contention window mechanism proposed in this research.

To compare **Figure 4(a)** and **Figure 4(b)**, from the node number 5 to 12, $CW = [15 \ 31 \ 63 \ 127]$ has the highest throughput. From 13 to 35 nodes, $CW = [31 \ 63 \ 127 \ 255]$ possesses the highest throughput. After 35, $CW = [63 \ 127 \ 255 \ 511]$ has the highest throughput. Most of their slot utilization are in 0.1 ~ 0.2 when they possess the highest throughput. One can thus infer that the value of slot utilization in 0.1 ~ 0.2 would be the best setting in this case.

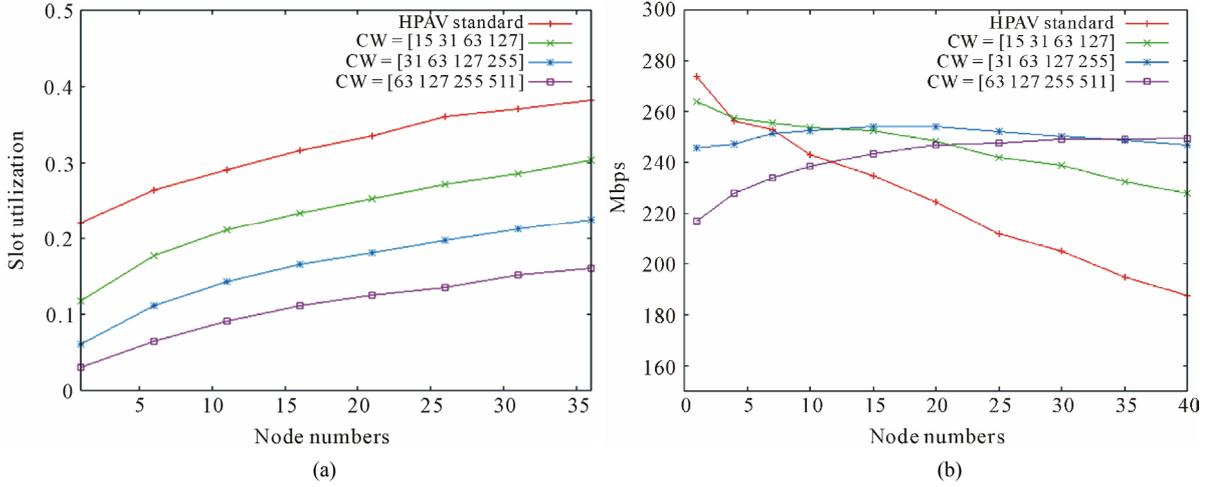


Figure 4. Simulation on different CW setting: (a) Slot utilization; (b) Throughput.

In summary, if one can control slot utilization in an appropriate region then better throughput could be expected.

4. Adaptive Contention Window Mechanism

From [11], it was known that when τ defined in Equation (3) as the following way, throughput of the network will exhibit the best performance:

$$\tau = \frac{\sqrt{[n + 2(n-1)(T_{c,avg} - 1)]/n - 1}}{(n-1)(T_{c,avg} - 1)} \approx \frac{1}{n\sqrt{T_{c,avg}/2}} \quad (5)$$

where $T_{c,avg}$ is the average time of collision transmission T_c in the slot time σ as

$$T_{c,avg} = \frac{T_c}{\sigma} \quad (6)$$

Substituting Equation (6) into Equation (5) yields the new τ as

$$\tau = \frac{1}{n\sqrt{\frac{T_c}{2\sigma}}} \quad (7)$$

Substituting Equation (7) into Equation (3) gives the optimal slot utilization which will lead to the optimal throughput. We set this optimal slot utilization as the standard which is expressed as $P_{b,opt}$:

$$P_{b,opt} = 1 - (1 - \tau)^n = 1 - \left(1 - \frac{\gamma}{n}\right)^n \quad (8)$$

where $\sqrt{\frac{2\sigma}{T_c}} = \gamma$.

After recognized data received, every station records the slot utilization by using Equation (4) for the last successful transmission packet. If slot utilization of the last packet is greater than $P_{b,opt}$, it means that the current contention window is not big enough which may cause the congestion. It will assign the system a larger contention window. Otherwise, it will adjust the contention window for reduction. Our adaptive contention window architecture can be illustrated as in **Figure 5**.

In this figure, ρ is expressed as slot utilization of the last packet. By the revised function- $f(\rho)$, it converts the slot utilization to a revising factor to multiply the original contention window (CW = [7 15 31 63]). That is, the contention window of the next packet will be changed to $f(\rho)CW$.

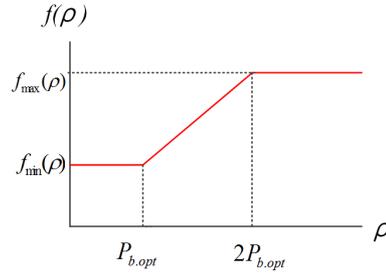


Figure 6. Correction factor of the contention window $f(\rho)$.

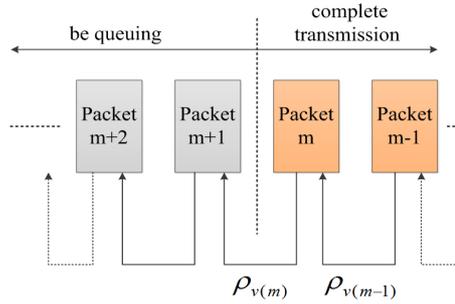


Figure 7. Sketch of slot utilization ρ_m and virtual slot utilization $\rho_{v(m)}$.

$$\rho_{v(m)} = \begin{cases} \rho_{v(m-1)} + 0.2(2P_{b,opt} - \rho_{v(m-1)}), & \rho_m > P_{b,opt} \\ \rho_{v(m-1)} - 0.2(\rho_{v(m-1)} - P_{b,opt}), & \rho_m < P_{b,opt} \\ \rho_{v(m-1)}, & \rho_m = P_{b,opt} \end{cases} \quad (11)$$

By this process, every station CW will reach to a region which the probability of every station transmitting packets will be close. The revising factor $f(\rho)$ will be increasing continuously when $\rho \geq P_{b,opt}$ as well.

While the previous method works in an efficient way, there is a weakness, *i.e.* the contention window mechanism shows lower throughput when the node number is few. That's because if we modify the contention window at BPC = 0 by the previous method, it tends to prompt a larger revising factor when the node number is few. Thus, it still exhibits a large contention window causing lower throughput. It is expected that, at each time instant, ρ could really reflect the previous congestion level. Thus, when n is large, we don't want the station transmits at the first contention period time. It means that one still needs a larger CW at BPC = 0 when the node number is large. Therefore, one has to define an updated revising factor for the state BPC = 0.

Unlike the revising factor characterized in **Figure 5** using a simple multiplication to modify the contention window, the revising factor at BPC = 0 should be updated in the way of addition. Comparing to the multiplication operation, the amount of addition should be smaller. It will be more suitable for the case of BPC = 0 since its deferral counter is 0.

The modification is to introduce a correction factor α , say

$$\alpha = \left\lceil \frac{4}{M} f(\rho_{v(m)}) \right\rceil \quad (12)$$

where the operator $\lceil \cdot \rceil = \min \{n \in Z | n \geq x\}$. By this way, we obtain an integer α , with its values in between 1 to 4, to indicate the congestion level.

Next, we record the counter of the backoff procedure of the last packet, denoted BPC_m . To proceed, set the maximum value of BPC_m to be 3 and define

$$\beta = \text{BPC}_m + 1$$

Therefore, the value of β will be from 1 to 4. Combining α and β forms a congestion level indicator as

$$\omega = \alpha^\beta$$

where [Table 4](#) lists all possible values of ω .

We choose ω as our revising factor when $\text{BPC} = 0$. By this process, the station will have less probability to transmit at the first round in the backoff region when the node number is large. It doesn't influence throughput when the node number is fewer.

The complete design process of our adaptive contention window process can be summarized as follows:

Step 1: Based on the given slot time σ , collision transmission time T_c and node number n obtain $P_{b,opt}$ as

$$P_{b,opt} = 1 - \left(1 - \frac{1}{n} \sqrt{\frac{2\sigma}{T_c}} \right)$$

Step 2: Define the revising factor $f(\rho)$ by $P_{b,opt}$:

$$f(\rho) = \begin{cases} 1, & \rho < P_{b,opt} \\ \frac{M-1}{P_{b,opt}} \rho + 2 - M, & P_{b,opt} \leq \rho \leq 2P_{b,opt} \\ M, & \rho > 2P_{b,opt} \end{cases}$$

Step 3: Count the idle and busy slot numbers of the last transmission packet to obtain the virtual slot utilization:

$$\rho = \frac{\text{busy_slot_numbers}}{\text{all_slot_numbers}}$$

$$\rho_{v(m)} = \begin{cases} \rho_{v(m-1)} + 0.2(2P_{b,opt} - \rho_{v(m-1)}), & \rho > P_{b,opt} \\ \rho_{v(m-1)} - 0.2(\rho_{v(m-1)} - P_{b,opt}), & \rho < P_{b,opt} \\ \rho_{v(m-1)}, & \rho = P_{b,opt} \end{cases}$$

Step 4: Specify the revising factor ω when $\text{BPC} = 0$ in [Table 2](#):

$$\alpha = \left[\frac{4}{M} f(\rho_{v(m)}) \right]$$

$$\beta = \text{BPC}_m + 1$$

$$\omega = \alpha^\beta$$

Step 5: Update the contention window via:

1) Backoff procedure is at $\text{BPC} = 0$:

$$\text{CW}^* = \text{CW} + \omega$$

2) Backoff procedure is at $\text{BPC} > 0$:

$$\text{CW}^* = \text{CW} f(\rho_{v(m)})$$

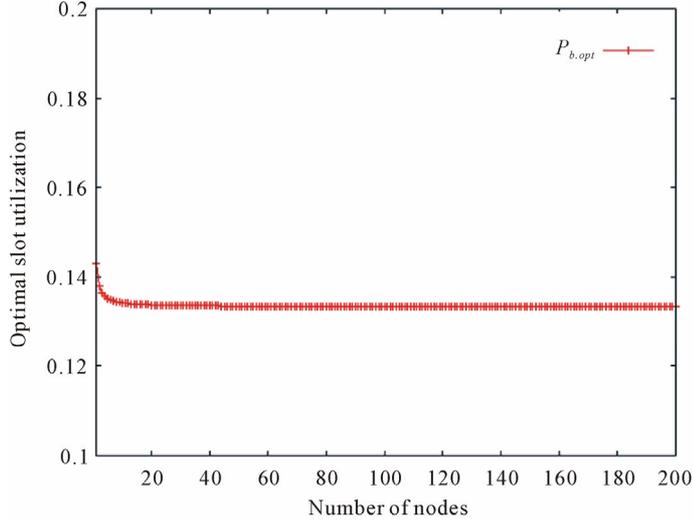
So far, our focus is only in own station. However, the station number in Equation (8) is not fixed which is unknown in general. To resolve this problem, we calculate $P_{b,opt}$ ($T_c = 3500 \mu\text{s}$) with different n , see [Figure 8](#).

When the node number is in $1 \sim 200$, its value will not significantly affect $P_{b,opt}$. To prove the node numbers will not influence $P_{b,opt}$ further, we calculate the sensitivity when n is larger than 200.

Define $\gamma = \sqrt{\frac{2\sigma}{T_c}}$. It follows that the sensitivity function of $P_{b,opt}$ with respect to the node number n is given by

Table 4. List of the values of ω .

ω	$\beta=1$	$\beta=2$	$\beta=3$	$\beta=4$
$\alpha=1$	1	1	1	1
$\alpha=2$	2	4	8	16
$\alpha=3$	3	9	27	81
$\alpha=4$	4	16	64	256

**Figure 8.** Variation of $P_{b,opt}$ on different node numbers.

$$S_n^{P_{b,opt}} = \frac{\partial P_{b,opt}}{\partial n} \cdot \frac{n}{P_{b,opt}} = -\left(1 - \frac{\gamma}{n}\right)^n \left[\ln\left(1 - \frac{\gamma}{n}\right) + \frac{\gamma}{n\left(1 - \frac{\gamma}{n}\right)} \right] \frac{n}{1 - \left(1 - \frac{\gamma}{n}\right)^n}$$

When $n > 200$, the terms $\left(1 - \frac{\gamma}{n}\right)^n$ and $\frac{n}{1 - \left(1 - \frac{\gamma}{n}\right)^n}$ will be smaller than 1. However,

$$\ln\left(1 - \frac{\gamma}{n}\right) + \frac{\gamma}{n\left(1 - \frac{\gamma}{n}\right)} \ll 1. \text{ Thus, } S \approx 0.$$

In the following study, the network node number n is set to be 10. Furthermore, if we consider $CW = [127 \ 255 \ 511 \ 1023]$ as the largest contention window, which will be approximately 15 times to the case of $CW = [7 \ 15 \ 31 \ 63]$, then we set the maximum revising function M to be 15.

The operational flow chart of the updated adaptive contention window mechanism is illustrated in **Figure 9**.

5. Experimental Results

The goal of this research is to acquire higher throughput while maintaining slot utilization. The simulation experiments have been conducted at the network simulator NS3. We set σ to be 35.84 μs which remains the same as in HomePlug AV. Simulation study is mainly conducted for different T_s and CW.

Figure 10(a) shows slot utilization for various CW settings. No matter HomePlug AV standard or other CW settings, their slot utilizations exhibit large variations when the number of nodes increases. However, there is

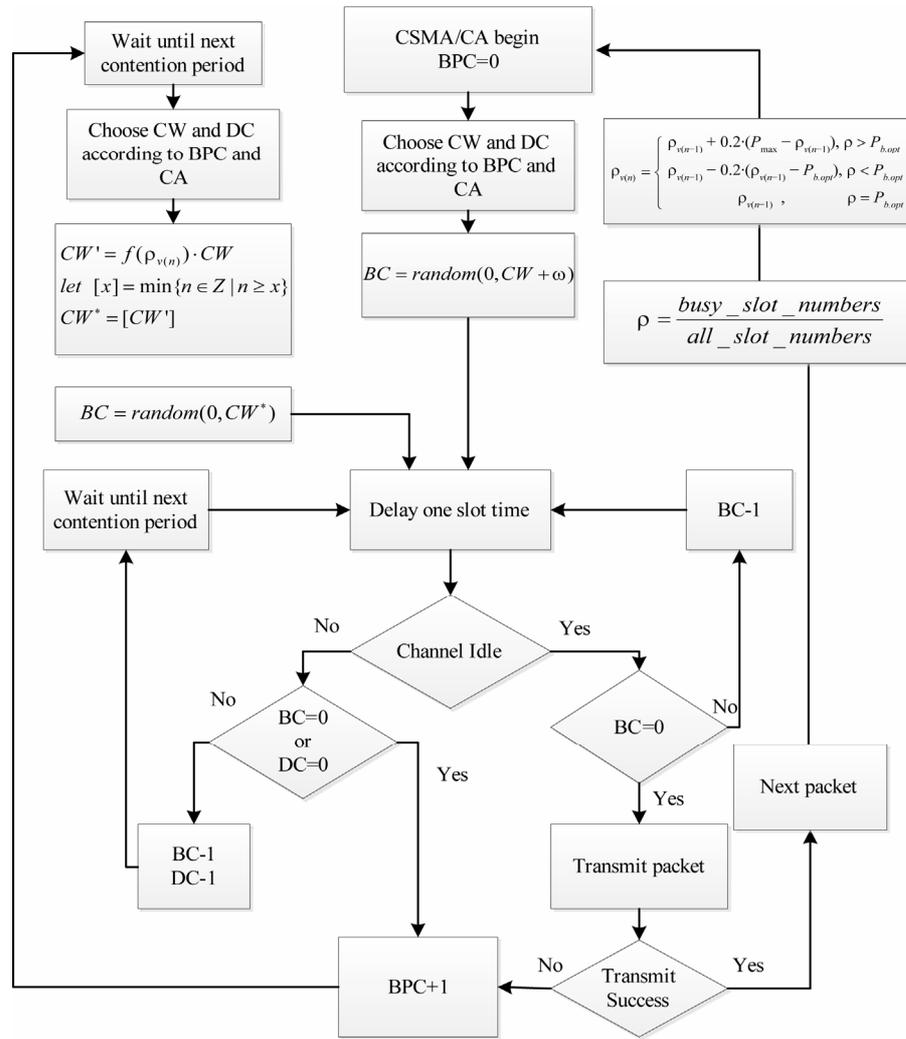


Figure 9. Operational flow chart of the proposed adaptive CW mechanism.

only slight difference among different cases in our proposed approach. This result shows that our method is effectively to control slot utilization. Even for a large scale network, it shows no significant difference.

After verifying slot utilization, the main point of this research is to get better throughput on the large scale network.

We prove in the follows that the proposed approach exhibits the best throughput performance while comparing to other fixed CW settings.

Including the original HomePlug AV (CW = [7 15 31 63]). All packets are supposed to possess the same priority and all stations are on the saturated situation which means that all stations have packets to transmit at all time. We record the average throughput every 30 seconds for each condition.

Figure 10(b) shows throughput of various CW settings with T_s equal to 3500 μ s. HomePlug AV standard exhibits the highest throughput at the beginning. When the node number becomes large, HomePlug AV standard's CW setting is not big enough because its throughput decreases seriously. On the other hand, CW = [63 127 255 511] has the lowest throughput at the beginning because the system wastes too much time on the idle slots. However, it rises subsequently when the node number become large. Our method shows the highest at all time since its CW setting is not fixed. When the node number is relatively few, it uses a small CW. When the node number increases, it automatically increases the CW value accordingly. Thus, when the node number becomes large, it still exhibits low probability in network collision. That's why it can keep high throughput of the network over the whole communication time. This is as expected before. Figure 10(c) shows that when the node

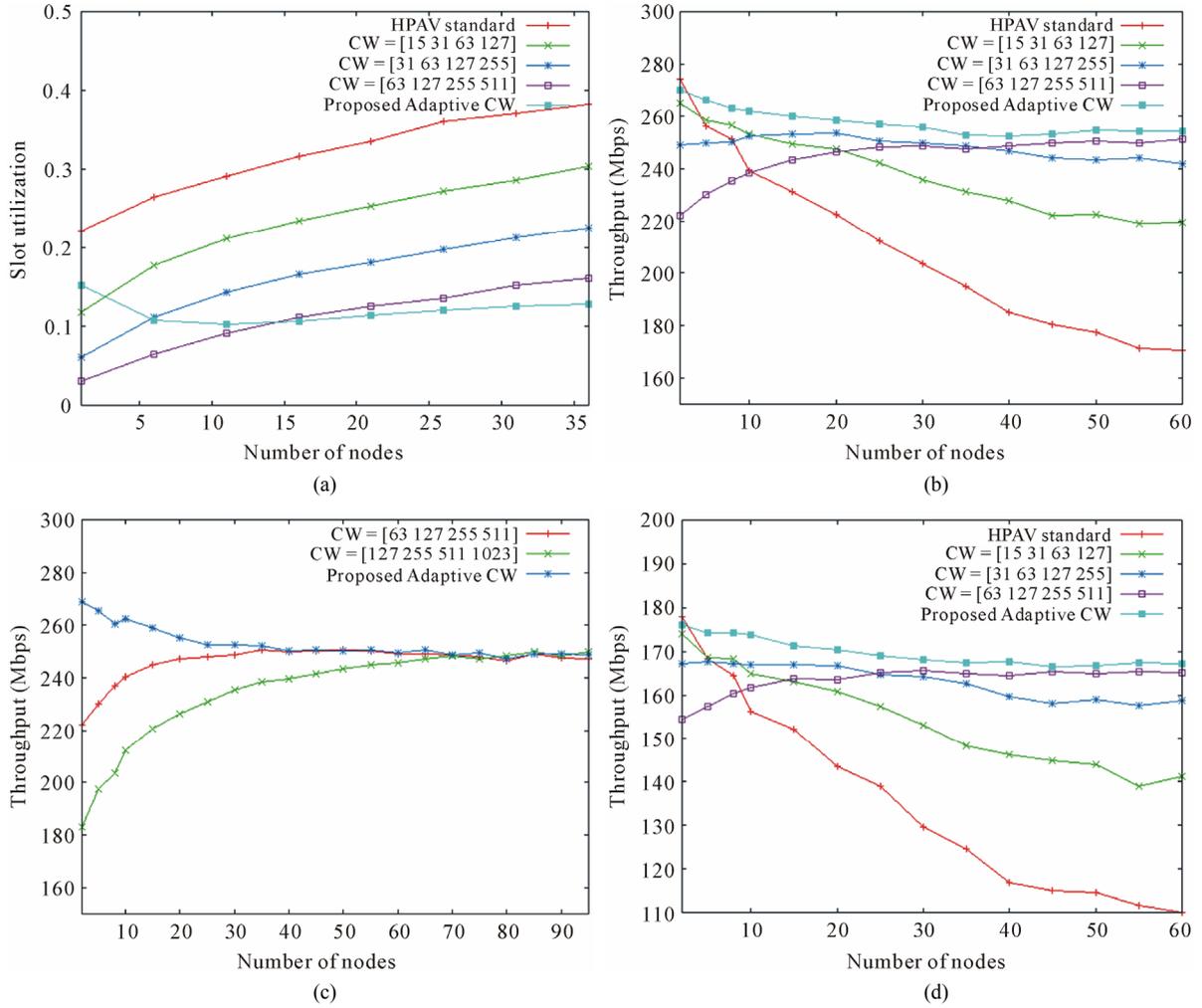


Figure 10. Various CW settings in different simulation cases: (a) Slot utilization; (b) Throughput ($T_s = 3500 \mu s$); (c) Throughput (large node number and large CW); (d) Throughput ($T_s = 5500 \mu s$).

number is increased to 95, it performs as the cases of $CW = [127\ 255\ 511\ 1023]$ in the simulation study. The simulation result shows that when the node number approaches to a large value, throughput will converge to a small region. Thus, setting an extremely large CW doesn't bring positive effect. That's also the reason that we set a limit to $f(\rho)$.

To prove performance robustness of the present mechanism, we conduct experiments for different transmission time. **Figure 10(d)** illustrates that throughput of the successful transmission time equals $5500 \mu s$. This shows the similar result, our method performs the best at all time. From **Figure 10(b)** and **Figure 10(d)**, it can be seen that if transmission time is less, the degree of CW affecting throughput is larger and our adaptive CW mechanism performs better than other cases as well.

We also verify the case when there is an abrupt change in the node number and examine if it still works. Consider the network has 40 nodes at the beginning. At 60 seconds, it reduces to 10 nodes. We compare the situation with 10 nodes at the beginning and record throughput at each second. The simulation result is shown in **Figure 11(a)**. It displays that after 60 seconds, throughput of the situation 1 has been improved and quickly approaches to the line of situation 2. This endorses performance robustness of our proposed mechanism

Finally, the number n in Equation (8) was changed to 10 at the previous simulation test. To prove that it won't significantly alter the result, we conduct extra tests with the number of nodes $n = 30$ and compare it to the case of $n = 10$. See **Figure 11(b)** for the result. It clearly shows that the value of n doesn't significantly affect throughput. That demonstrates robustness of our proposed approach to the network complexity.

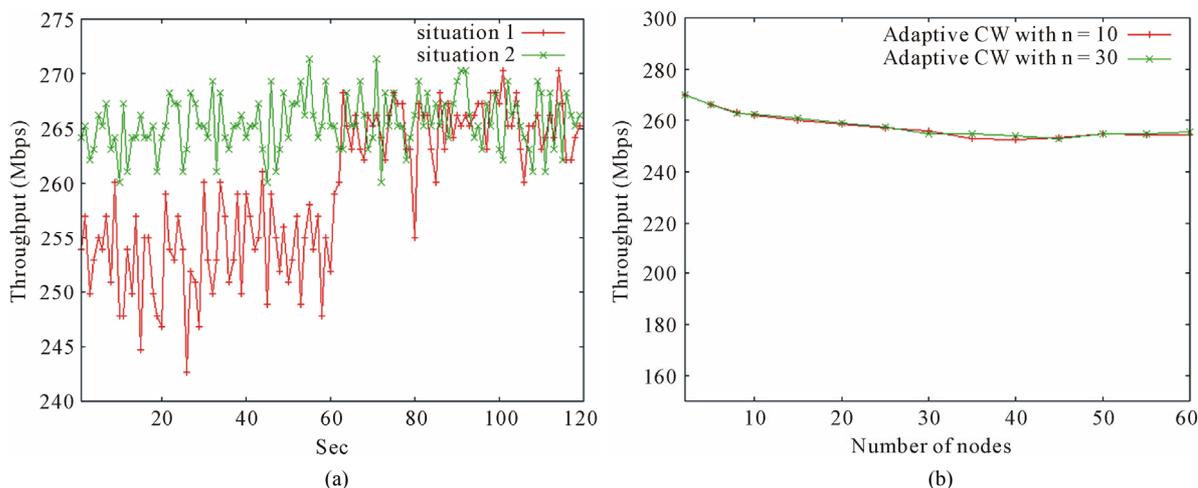


Figure 11. Throughput at different special case: (a) Situation 1: 40 nodes at network beginning. It remains 10 nodes after 10 s. Situation 2: 10 nodes at all time; (b) Various setting of $P_{b,opt}$.

6. Conclusion

PLC is becoming mature in recent years. However, as the usual network communication, the network congestion problem is crucial when considering maintaining communication quality. While HomePlug AV is already a mature protocol in PLC, we propose here an adaptive contention mechanism instead of formulating a new protocol. The adaptive contention window scheme only needs the information from CSMA/CA in HomePlug AV. In addition, because the information needed is self-contained, one does not need to correct the PHY layer setting to acquire extra information from other stations. All needed are to substitute the new CW and BC into the original HomePlug AV mechanism. This makes the approach more practical and presents better feasibility. From the simulation experiments conducted at NS3, it is found that the proposed scheme can effectively improve throughput. We have tested it with a variety of several scenarios; satisfactory results have been observed which show identifiable improvement of our proposed design.

Acknowledgements

This research was sponsored by National Science Council, Taiwan, under the Grant NSC 101-2221-E-005-015-MY3.

References

- [1] HomePlugPowerline Alliance (2007) HomePlug AV Specification, Ver. 1.1.
- [2] Chung, M.Y., Jung, M.H., Lee, T.J. and Lee, Y. (2006) Performance Analysis of HomePlug 1.0 MAC with CSMA/CA. *IEEE Journal on Selected Areas in Communications*, **24**, 1411-1420. <http://dx.doi.org/10.1109/JSAC.2006.874404>
- [3] Velarde-Alvarado, P., Martinez-Pelaez, R., Ruiz-Ibarra, J. and Morales-Rocha, V. (2014) Information Theory and Data-Mining Techniques for Network Traffic Profiling for Intrusion Detection. *Journal of Computer and Communications*, **2**, 24-30. <http://dx.doi.org/10.4236/jcc.2014.211003>
- [4] Yoon, S.G. and Bahk, S. (2011) Adaptive Rate Control and Contention Window-Size Adjustment for Power-Line Communication. *IEEE Transactions on Power Delivery*, **26**, 809-816. <http://dx.doi.org/10.1109/TPWRD.2010.2094629>
- [5] Kriminger, E. and Latchman, H. (2011) Markov Chain Model of HomePlug CSMA MAC for Determining Optimal Fixed Contention Window Size. *Proceedings of IEEE International Symposium on Power Line Communications and Its Applications*, Udine, 3-6 April 2011, 399-404.
- [6] Liu, K.H., Hsieh, D.R., Hsu, J.Y. and Chang, S.Y. (2012) Throughput Improvement for Power Line Communication by Adaptive MAC Protocol. *Proceedings of IEEE International Power Engineering and Optimization Conference*, Melaka, 6-7 June 2012, 135-140.
- [7] Luca, D.B., Alessandro, S.D. and Tonello, A.M. (2013) MAC Enhancements for G3-PLC Home Networks. *Proceed-*

- ings of IEEE International Symposium on Power Line Communications and Its Applications*, Johannesburg, 24-27 March 2013, 155-160.
- [8] Tsokalo, I., Radeke, R. and Lehnert, R. (2013) Enhancement of Backoff Algorithm in CSMA/CA Protocols for Broadband PLC. *Proceedings of IEEE International Symposium on Power Line Communications and Its Applications*, Johannesburg, 24-27 March 2013, 47-52.
 - [9] Wu, H., Long, K. and Heng, S. (2002) IEEE 802.11 Distributed Coordination Function (DCF): Analysis and Enhancement. *Proceedings of IEEE International Conference on Communications*, 28 April-2 May 2002, 605-609.
 - [10] NS3. <https://www.nsnam.org/>
 - [11] Bianchi, G. (2000) Performance Analysis of the IEEE 802.11 Distributed Coordination Function. *IEEE Journal on Selected Areas in Communications*, **18**, 535-547. <http://dx.doi.org/10.1109/49.840210>

Scientific Research Publishing (SCIRP) is one of the largest Open Access journal publishers. It is currently publishing more than 200 open access, online, peer-reviewed journals covering a wide range of academic disciplines. SCIRP serves the worldwide academic communities and contributes to the progress and application of science with its publication.

Other selected journals from SCIRP are listed as below. Submit your manuscript to us via either submit@scirp.org or [Online Submission Portal](#).

