# Two-way Communication with Wait-slot Scheme for Neighbor Discovery Process in Dense Bluetooth Low Energy Networks

Ting-Ting Yang, Hsueh-Wen Tseng, Member, IEEE
Department of Computer Science and Engineering
National Chung-Hsing University, Taiwan, R.O.C.
Email: d103056005@mail.nchu.edu.tw, hwtseng@nchu.edu.tw

***Abstract*—Bluetooth Low Energy (BLE) Beacon technology is well on the way to becoming the future of business due to its inexpensive and low-power properties. All communications in BLE networks must involve neighbor discovery process (NDP) in the first place since a BLE device needs to create a connection or exchange information with its neighbors. Thus, the performance of the discovery latency is a challenging issue to be addressed for integrating BLE into the Beacon application development as the number of BLE devices increases. In this paper, we propose a two-way communication with wait-slot scheme (TCWS) to minimize the probability of collision occurring on the response frames of BLE devices and improve the latency of NDP. We formulate the state transition diagram for analyzing the performance of our proposed scheme. The results show that TCWS provides much better performance in terms of the probability of collision and the discovery latency in dense BLE networks.**

***Index Terms*—Bluetooth Low Energy, Neighbor Discovery Process, Discovery Latency.**

## I. INTRODUCTION

In [1], several experts shared their insights about Beacon technology. We'll start to see the beginning of the revolution of Beacon technology [2], which will eventually change the way we live our everyday lives. ABI Research predicts that over 10 billion Bluetooth-enabled devices (i.e., Beacons) will be on the market by 2018 [3]. In retail setting, Beacons will be used to optimize flow of service and staffing and also deliver coupons to a consumer's mobile device as they're passing a specific item of interest. According to InReality [4], 75% of shoppers are using their mobile devices to make a purchase in stores. When customers pass stores with installed Beacons, their mobile device can receive the push notifications of specials and discount coupons via the APP.

From the viewpoint of the application, brick and mortar retail stores use BLE Beacons for mobile commerce, offering customers special deals through mobile marketing. Mobile phones used by the customers must rapidly and efficiency detect the broadcast signal from the BLE Beacon so that the retailer can push the relevant information immediately to the customers. All communications in BLE networks must involve neighbor discovery process (NDP) in the first place since a BLE device needs to create a connection or exchange information with its neighbors [5]. Therefore, a fast and energy efficient NDP is an important issue and the performance of the discovery latency is a challenge to be addressed for integrating BLE into the Beacon application development.

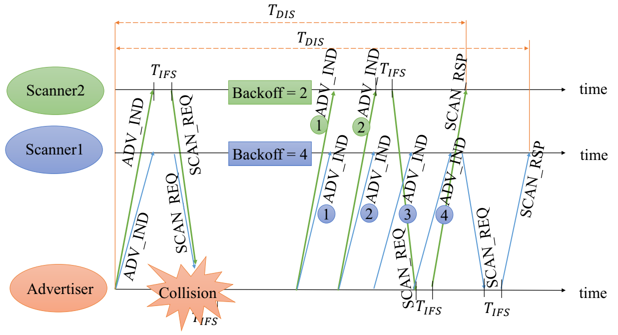


Fig. 1. The backoff procedure of the standard BLE in NDP.

As BLE Beacon technology becomes more pervasive, the number of BLE devices increases to raise the probability of collision among advertisers (i.e., Beacons) and scanners (i.e., BLE devices). In the standard BLE, a single advertiser broadcasts its signal uniformly to all directions. There may be ten or ten thousand scanners, or maybe none at all listening to its broadcast. At this time, if one or more scanners receive the signal on the same channel at the same time, they simultaneously send back the respond frame to the advertiser. As a result, scanners cause a collision phenomenon to degrade the performance significantly.

BLE has 40 channels with 2 MHz bandwidth. Three out of these 40 channels, with channel indexes 37, 38, and 39, are used for advertising, and the rest are data channels. The advertiser continuously and sequentially sends Packet Data Units, called ADV_IND PDUs, to each of the three advertising channels during an advertising interval. Then, the scanner sends back a SCAN_REQ PDU on the same channel. As shown in Fig. 1, the standard BLE runs backoff procedure to minimize collisions of SCAN_REQ PDUs from scanners. The backoff count is decreased by one in the scanners when they successfully receive an ADV_IND PDU. The scanner sends SCAN_REQ PDU only until the backoff count reaches the value of zero.

978-3-901882-98-2

Although it reduces the probability of the collision occurring on SCAN_REQ PDUs, it results in a larger discovery latency ($T_{DIS}$).

Hence, in this paper, we propose an improved scheme to reduce the probability of collision occurring on SCAN_REQ PDUs in the advertising channels and improve the discovery latency of NDP. The rest of the paper is organized as follows. Section II shortly reviews the related works of the BLE NDP. After that, we describe the standard BLE transmission scheme and explain our method, two-way communication with wait-slot scheme (TCWS), to enhance the performance of NDP in Section III. In Section IV, we formulate the state transition diagram for analyzing the performance of our proposed scheme and derive performance measures as closed form. The performance comparison is given in Section V. Finally, Section VI concludes the paper.

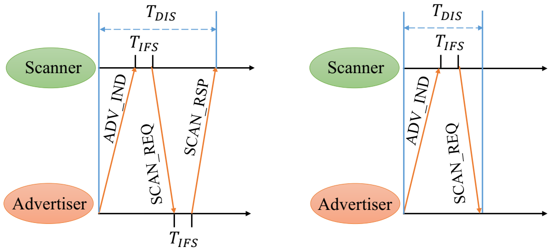
## II. RELATED WORKS

In the last couple of years, several works focused on developing the analytical model of NDP, which is applicable for any parameter settings specified in the standard BLE and then validated via extensive simulations and experiments. The authors developed an analytical model for the discovery latency of NDP based on the listening interval of scanners [6] [7] [8] [9] and the length of advertising event [10] [11] [12] [13].

The pioneering works on the mathematical analysis of NDP are [6] and [7]. These models focus on the pair BLE devices and assume there is no collision among homogeneous BLE devices. The scheme [8] provides an adaptive mechanism to learn the network contention and adjusts their parameters (the interval length of scanning and advertising) accordingly. In [9], the scanner executes a random backoff procedure to determine the transmission time of an advertising PDU (i.e., SCAN_REQ PDU). However, they do not consider the changing of different advertising channels.

The analytical models [10] and [11] considered multiple NDP device pairs (i.e., advertiser and scanner). However, the authors assume constrains to simplify the problem but the constrains affect the accuracy of the analytical model. In [12], the work concentrated on NDP for one pair of advertiser and scanner, as well as considering transmission collision with nearby advertisers. Moreover, an enhanced discovery mechanism based on carrier sensing for BLE devices to avoid collisions during advertisement process has been proposed in [13]. In their scheme, advertisers must listen to advertising channels before advertisement to reduce collisions among them. However, this scheme is applicable only multiple advertisers and a single scanner.

In a word, much of the literature previously developed analytical models to study BLE NDP, and then they used various parameters into the models to show the performance of the discovery latency. In terms of scanners, the discovery latency of NDP depends on the length of the scan interval. Thus, it has the limitation on the length of the scan interval to prevent long discovery latency. In addition, these papers do not consider that scanners sequentially listen on the different advertising channels as well as advertisers send the advertising information on the different advertising channels periodically. In this case, it affects the probability of successful discovery device. In terms of advertisers, the discovery latency of NDP between pair BLE devices focuses on multiple advertisers and a single scanner. The enhanced scheme decreases the probability of the collision among advertisers to improve the discovery latency, but it is not applicable for scenarios with BLE Beacon applications. Therefore, in Section III, we propose an enhanced scheme to improve the performance of NDP for the BLE advertising application with multiple scanners.



(a)Three-way handshake (b)Two-way communication

Fig. 2. Two different neighbor discovery processes.

## III. TWO-WAY COMMUNICATION WITH WAIT-SLOT SCHEME

In the BLE advertising application, advertisers (Beacons) transmit advertising PDUs to BLE-enabled devices (smartphones) within broadcast range. When the advertiser sends an ADV_IND PDU, it then triggers a three-way handshake. The scanner waits a $T_{IFS}$ and then responds a scan request (SCAN_REQ PDU) as it receives an ADV_IND PDU successfully. Then, the advertiser sends a scan response (SCAN_RSP PDU) after a $T_{IFS}$ to the scanner, as shown in Fig. 2(a). This is a neighbor discovery process (NDP) in the standard BLE. $T_{IFS}$ is the gap between two successive transmitted frames. However, if the number of scanners increases, multiple scanners attempt to send SCAN_REQ PDU to the advertiser to increase the probability of collision substantially. Even though the scanner uses a backoff count to decide when to reply SCAN_REQ PDU, it results in a considerable discovery latency, as shown in Fig. 1. Thus, in this paper, we propose a two-way communication with wait-slot scheme (TCWS) to improve the performance of NDP.

### A. *Two-way Communication*

In general, a three-way handshake can support BLE-enabled devices for some applications that need specific data. Under such communication procedure, the scanner obtains related information for applications so that the scanner does not have to create a full data connection to the advertiser. However, from the aforementioned descriptions, the scanner only fetches data from the cloud datacenter once via the APP after receiving ADV_IND PDU. Thus, the scanner obtains related information about retailers. In this case, replying SCAN_RSP PDU to the scanner is resource wasting for the advertiser. Hence,

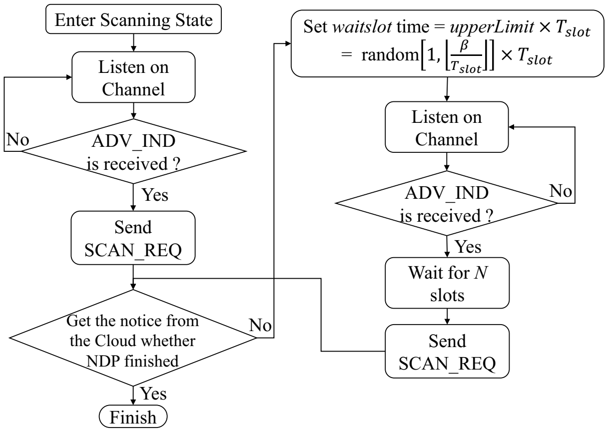


Fig. 3. The operation of the wait-slot scheme.

we simplify the three-way handshake and propose a two-way communication scheme to finish NDP, as shown in Fig. 2(b).

The time for transmitting ADV_IND PDU and SCAN_REQ PDU are denoted as $T_{ADV\_IND}$ and $T_{SCAN\_REQ}$. When the advertiser and the scanner exchange ADV_IND PDU and SCAN_REQ PDU successfully, the discovery process is finished and $T_{DIS}$ is used to represent the total time for NDP. Thus, the discovery latency ($T_{DIS}$) is defined as the interval for the advertiser from entering into the first advertising event by sending an ADV_IND PDU until it successfully receives a SCAN_REQ PDU from the scanner. That is $T_{DIS} = T_{ADV\_IND} + T_{SCAN\_REQ} + T_{IFS}$.

In the BLE specification, we can set the advertising filter policy that prohibits processing SCAN_REQ PDUs in the advertiser [5]. Consequently, the advertiser shall either move to the next advertising channel to send another ADV_IND PDU, or close the advertising event. Thus, we use the two-way communication scheme which does not reply SCAN_RSP PDU in the advertiser to reduce resources wasting and improve the latency of NDP. Note that we have described the basic idea of backoff procedure in the standard BLE. The scanners implement a backoff mechanism when they do not receive SCAN_RSP PDU from the advertiser. In general, several scanners are close around the scanner with receiving ADV_IND PDU. They have to use backoff mechanism to decrease the probability of collision of transmitting SCAN_REQ PDUs. However, the advertiser does not reply SCAN_RSP PDU to the scanner in the two-way communication scheme. In the standard BLE, if the scanner does not receive SCAN_RSP PDU, it does not decide the backoff count to retransmit SCAN_REQ PDU. Thus, the scanner has to reply SCAN_REQ PDU smartly to finish NDP in the two-way communication scheme. In consequence, we propose a wait-slot scheme to enhance the two-way communication in Subsection III-B.

### B. Wait-slot Scheme

In the two-way communication scheme, the advertiser periodically broadcasts advertising PDUs on the three advertising channels in sequence when it enters the advertising event. Then, the advertiser listens on the same channel. If more scanners hear the advertising PDU on the same advertising channel at the same time, they simultaneously respond SCAN_REQ PDU to the advertiser. As a result, collisions occur on the advertiser. Thus, we need to separate the responding packets in different time to avoid the collision. Furthermore, all the responding packets from scanners have to send back to the advertiser at different time before the advertiser switches to the next channel. Sequentially, the advertiser receives all responding packets from scanners to finish NDP. For this reason, we propose a wait-slot scheme to distribute SCAN_REQ PDUs in different time when the scanners respond SCAN_REQ PDU upon receiving an ADV_IND PDU from the advertiser. In the standard BLE, the scanner shall run a backoff procedure to minimize collisions of SCAN_REQ PDUs from multiple scanners. In the backoff procedure, it uses two parameters, *backoffCount* and *upperLimit* to restrict the number of SCAN_REQ PDUs sent when collisions occur [5].

In our two-way communication scheme, the advertiser does not response SCAN_RSP PDU. Thus, we use the *upperLimit* to express the *waitslot* time of each scanner. The scanner has to wait for a little time according to the value of *waitslot*, and then sends SCAN_REQ PDU to the advertiser. In order to guarantee that SCAN_REQ PDU is sent during advertising period per channel (i.e., max allowable listening time for SCAN_REQ PDU after sending ADV_IND PDU on each channel), which is denoted as $\beta$. That is, the value of *waitslot* is restricted to the tolerable time that is the advertiser to wait for SCAN_REQ PDU of the scanner after sending an ADV_IND PDU. It can be expressed as

$$waitslot = upperLimit \times T_{slot} = random[1, \lfloor\frac{\beta}{T_{slot}}\rfloor] \times T_{slot}, \quad (1)$$

where $T_{slot}$ is the transmission time of a SCAN_REQ PDU and we use it to represent the unit slot time. The length of a slot is about 0.176 $ms$. Furthermore, $\lfloor\frac{\beta}{T_{slot}}\rfloor$ is the largest integer equal to or smaller than $\frac{\beta}{T_{slot}}$. Then, the duration of $\beta$ is quantized into several discrete components and the unit of a discrete component is $T_{slot}$. Thus, we can obtain the different values of *waitslot* from the product of a random number and $T_{slot}$. In this case, each scanner can send SCAN_REQ PDU to the advertiser successfully at different time.

In the wait-slot scheme, the scanner sends a SCAN_REQ PDU upon receiving an ADV_IND PDU and then obtains a random value of the *upperLimit* to produce *waitslot* time according to Eq. 1. The operation of the wait-slot scheme is shown in Fig. 3. Initially, the advertiser broadcasts an ADV_IND PDU on one of the three advertising channels. If the scanner listens on the same channel, it replies a SCAN_REQ PDU to the advertiser to finish NDP. In this case, if the advertiser successfully receives SCAN_REQ PDU from the scanner, NDP is finished and the advertiser records the scanner's address. In the situation, using the two-way communication scheme can effectively shorten the latency of NDP. Besides, the cloud datacenter can collect all the scanners that had finished NDP

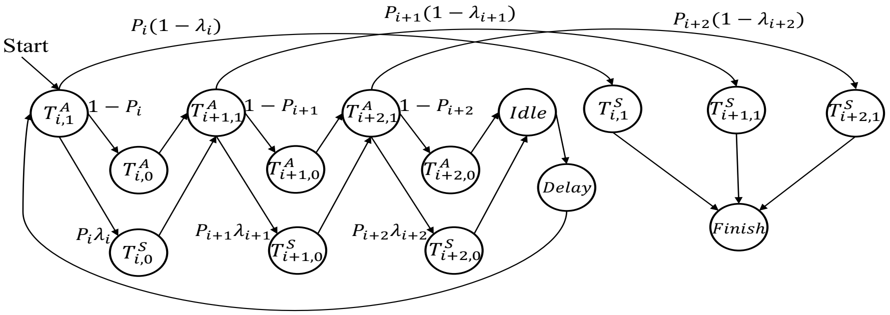


Fig. 4. The state transition diagram for BLE NDP.

and can send the notice to those scanners. Hence, if these scanners receive the same ADV_IND PDU, they do not need to reply SCAN_REQ PDU anymore.

Contrarily, once there is a collision phenomenon among SCAN_REQ PDUs, the advertiser cannot successfully receive the reply of scanners. Since the advertiser broadcasts ADV_IND PDUs continuously, the scanner can detect ADV_IND PDU at a second time and waits for the *waitslot* time to send back SCAN_REQ PDU on the same channel. In this way, it reduces the probability of collision on advertising channels and improves the probability of successful discovery.

## IV. ANALYTIC MODELING

In this section, we propose an analytical model of TCWS to evaluate the average discovery latency of NDP. In order to evaluate the performance of NDP, we quantized the time unit into a discrete component, call a time slot (i.e., $T_{slot}$). In other words, the model is assumed to be time slotted, and all time durations are normalized by the time slot.

There are three possible cases about NDP: inaudibility, success, and collision. The advertiser periodically sends an ADV_IND PDU on each channel. If the scanner is sleeping outside the *scanWindow* or is listening on the different channels, the scanner cannot hear this advertising information. On the other hand, the scanner can successfully discover the advertiser if it is awake within the current *scanWindow* and is listening on the same advertising channel as the advertiser. The scanner should be able to exchange ADV_IND PDU and SCAN_REQ PDU with the advertiser. At this time, if other scanners also send SCAN_REQ PDU simultaneously, the collision occurs on the advertiser.

Fig. 4 shows the transition diagram. The transition states are classified into the transmission of data packet from the advertiser to the scanner $(T^A)$, the transmission of data packet from the scanner to the advertiser $(T^S)$, $Idle$ and $Delay$ states after broadcasting ADV_IND PDUs on three advertising channels (i.e., channel 37, 38, and 39). If the scanner exchanges ADV_IND PDU and SCAN_REQ PDU with the advertiser successfully, it enters $Finish$ state. In Fig. 4, the states $T^A_{i,j}$ and $T^S_{i,j}$ represent the transmission of the data packet of the advertiser and the scanner respectively on the $i$th channel ($i =$ 37, 38, or 39), where $j$ indicates whether the transmission on the current channel is successful ($j = 1$) or failure ($j = 0$).

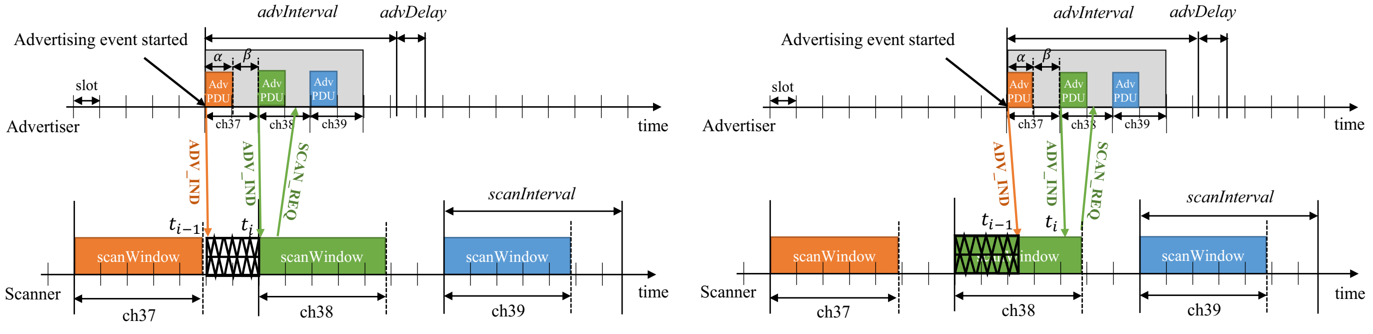
Let us examine the possible transitions from the state $T^A_{i,1}$ when the advertiser starts to broadcast an advertising PDU, as shown in Fig. 4. In inaudibility case, the scanner sleeps outside the *scanWindow* or listens on the different channels. That is, the state transits from $T^A_{i,1}$ to $T^A_{i,0}$ with probability $1 - P_i$, where $P_i$ is the probability that the scanner successfully receives an ADV_IND PDU from the advertiser. When the advertiser recognizes to finish NDP successfully, the state transits from $T^A_{i,1}$ to $T^S_{i,1}$ with probability $P_i(1-\lambda_i)$. Note that the probability that SCAN_REQ PDU of the scanner collides with that of a nearby scanner is $\lambda_i$. Therefore, the scanner receives the advertising PDU but it cannot reply SCAN_REQ PDU to the advertiser successfully. The transition probability from $T^A_{i,1}$ to $T^S_{i,0}$ is $P_i\lambda_i$. Furthermore, when the advertiser and scanner cannot finish NDP with the first advertising PDU within this advertising channel, the state transits from $T^A_{i,0}$ or $T^S_{i,0}$ to the transition state of next PDU, $T^A_{i+1,1}$. Then, the advertiser keeps broadcasting the advertising information until finishing NDP with the scanner.

In this paper, we propose TCWS to minimize the probability of SCAN_REQ PDU collision and improve the performance of NDP for the BLE advertising application with multiple scanners. For one of the scanners, we consider the case that other scanners try to finish NDP with the advertiser at the same time. The probability of such collision case is

$$\lambda_i = P_i \times (\frac{1}{upperLimit}), \tag{2}$$

where the value of the *upperLimit* is $\lfloor \frac{\beta}{T_{slot}} \rfloor$ and $\frac{1}{upperLimit}$ is the probability that the scanner selects a slot time from $[1 ... \lfloor \frac{\beta}{T_{slot}} \rfloor]$ to send the response frame. In addition, $P_i$ is the probability that the scanner successfully receives an ADV_IND PDU from the advertiser. As shown in Fig. 5, we note that the advertiser enters into the advertising event by sending an ADV_IND PDU on channel 37. Thus, the performance of NDP should be evaluated from the first PDU on channel 37.

For example, when we analyze the successful NDP on the channel 38, there are two cases about the probability of the successful ADV_IND PDU transmission. One case is that the advertiser sends the first PDU at time instant $t_{i-1}$, as shown in the Fig. 5(a). At this time, the scanner is sleeping outside the *scanWindow* on the channel 37. Next, when the scanner



(a) (b)

Fig. 5. Illustration of successful transmission of ADV_IND PDU scenario.

switches to the channel 38 and then listens on the channel 38, the scanner can detect the second PDU from the advertiser arriving at time instant $t_i$. Therefore, we can obtain that if the first ADV_IND PDU arriving in the range of grid box, the advertiser can finish ADV_IND PDU transmission. Thus, the successful probability of the leftmost edge of the *scanWindow* on the channel $i$ can be expressed as

$$P_i^L = \frac{1}{3} \times \frac{\alpha+\beta}{\tau_{SI}}, \quad (3)$$

where $\frac{1}{3}$ denotes random selection of a single channel among three channels. $\tau_{SI}$ is the length of the *scanInterval* of the scanner and $\alpha+\beta$ is advertising period per channel. In the Bluetooth specification [5], the max waiting time of the advertiser is 10 $ms$ for receiving SCAN_REQ PDU after sending an ADV_IND PDU per channel. That is, $\alpha+\beta \leq 10\ ms$. In the other case, as shown in the Fig. 5(b), the advertiser sends the first PDU at time instant $t_{i-1}$. Obviously, as the advertiser and the scanner are now operating in different frequency channels, they are unable to hear each other. To make sure that the scanner can receive the second ADV_IND PDU at the channel 38, the first ADV_IND PDU should arrive in the range of grid box, as shown in the Fig. 5(b). Thus, the successful probability of the rightmost edge of the *scanWindow* on the channel $i$ can be expressed as

$$P_i^R = \frac{1}{3} \times \frac{\tau_{SW} - (\alpha+\beta+T_{ADV\_IND}+T_{IFS})}{\tau_{SI}}, \quad (4)$$

where $\tau_{SW}$ is the length of the *scanWindow* of the scanner. $T_{ADV\_IND}$ and $T_{IFS}$ are the transmission time for transmitting an ADV_IND PDU and the inter frame space, respectively.

As a result, we can obtain the probability of successful transmission of ADV_IND PDU on the channel 38 from Eq. 3 and 4. It can be denoted as

$$P_i = P_i^L + P_i^R = \frac{1}{3} \times (\rho - \frac{T_{ADV\_IND} - T_{IFS}}{\tau_{SI}}), \quad (5)$$

where duty cycle ($\rho = \frac{\tau_{SW}}{\tau_{SI}}$) is the frequency of the scanner and is defined as the proportion of time spent on the scanning process by the scanner during a *scanInterval* period. The successful transmission of ADV_IND PDU on the channel 37 or 39 can also be derived as the same way as that of on the channel 38.

In addition, the number of scanners affects the probability of the successful NDP. We need to consider the probability of ADV_IND PDU transmission as well as the probability of SCAN_REQ PDU collision. The collision probability can be extended from Eq. 2 and can be expressed as

$$\lambda_i' = P_i \times (1-(1-\lambda_i)^{N-1}). \quad (6)$$

Thus, we use Eq. 5 and 6 to obtain the probability of successful NDP among $N$ scanners considering the collision. It is denoted as

$$P_i' = P_i \times (1-\lambda_i)^{N-1}. \quad (7)$$

Now, we obtain the expected discovery latency of NDP from the transition state as shown in Fig. 4. The summation is taken to infinity to obtain a mean value over all possible range of $k$. $k$ is the number of advertising event until successful discovery. Using a simple calculus, we have

$$\overline{D} = (\frac{\tau_{AI}+\overline{\tau_d}}{3}) \times [\sum_{k=1}^{\infty}(k \times P_i' \times (1-P_i')^{k-1}) - 1] + [(1,2,3)\times\alpha + (0,1,2)\times\beta + \overline{\tau_{UL}}], \quad (8)$$

where $\tau_{AI}$ is the length of the *advInterval*. $\overline{\tau_d}$ and $\overline{\tau_{UL}}$ are the expected value of the *advDelay* and the *upperLimit*, respectively. Besides, in the $k$th advertising event, the counts of $\alpha$ and $\beta$ are decided by the advertising channel index ($Adv\_idx$) [5] that has finished NDP (i.e., if $Adv\_idx = 38$, $\alpha$ should multiplied by 2 and $\beta$ should multiplied by 1). Finally, the average discovery latency can be derived as

$$\overline{D} = (\frac{\tau_{AI}+5}{3})(\frac{1}{P_i'} - 1) + 15.216. \quad (9)$$

## V. PERFORMANCE EVALUATION

In this section, we present a comparison of performance results obtained by TCWS, the standard BLE [10] [11], and Backoff scheme in [9], which we named it Backoff-SI. We had noted that the latency of NDP is computed in terms of scanners in [9] and the calculation method is different from others [10] [11]. Therefore, we apply the same Backoff scheme [9] for

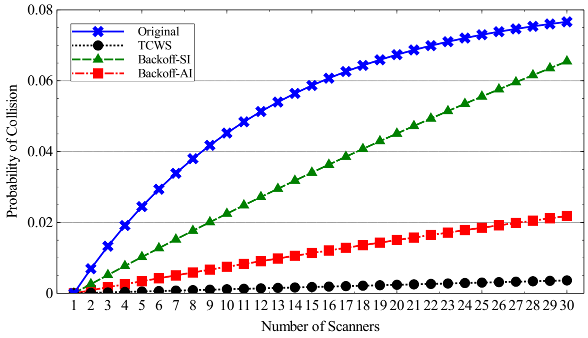


Fig. 6. Probability of collision as the number of scanners increases.

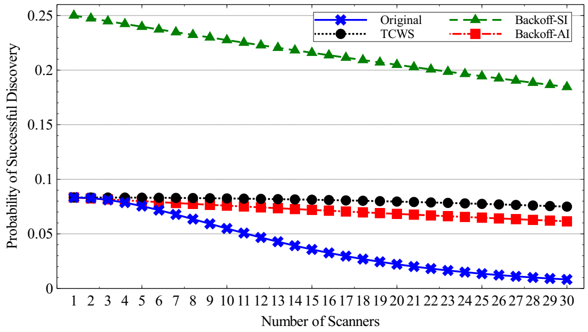


Fig. 7. Probability of successful discovery as the number of scanners increases.

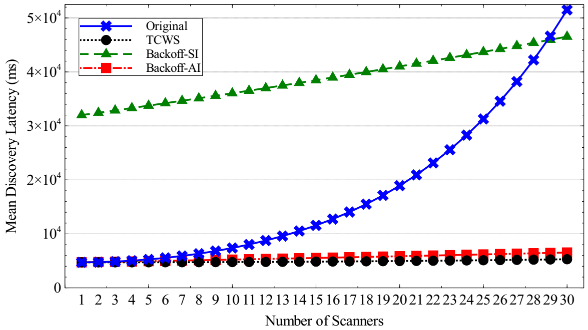


Fig. 8. The mean discovery latency as the number of scanners increases.

BLE NDP but compute the latency in terms of advertisers for performance comparison with TCWS and we named this scheme Backoff-AI. All the parameter settings are referred from [9], [10], and [11]. In order to validate the analytical results, we have considered the performance of discovery by cross-validation of NDP with the standard BLE and NDP with Backoff scheme. The curves are similar to the simulation results in the literature [9], [10], and [11].

Fig. 6 shows the probability of collision among SCAN_REQ PDUs as the number of scanners increases. We use blue-cross line to represent the probability of collision of the original standard BLE [10] [11]. TCWS is represented by black-circle line. Finally, green-triangle and red-rectangle lines are used to represent Backoff scheme [9] in the unit of the *scanInterval* and the *advInterval*, respectively. Since TCWS can partition allowable listening time into appropriate slots for the advertiser to receive SCAN_REQ PDUs on each channel, the scanners can separately send the response frame back to the advertiser to avoid the collision.

Fig. 7 shows the effect of the number of scanners on the probability of successful discovery. Both the advertiser and the scanner have to use the same channel to send and listen the PDU among of three advertising channels to finish NDP during the *scanWindow*. Thus, the probability of successful discovery is still low without considering the collision. The probability of successful discovery for Backoff-SI scheme is three times higher than that of other methods. The reason is that the analytical model is constructed in terms of scanners and they do not consider that the scanner may listen on one of the three different channels in [9]. We also can observe that, for TCWS, the probability of successful discovery slowly degrades as the number of scanners increases. As explained above, it is because TCWS adopts the wait-slot scheme to minimize the probability of collision.

Fig. 8 shows the effect of the number of scanners on the mean discovery latency. For given parameter settings in [10] and [11], with the longer $\tau_{SI}$, Backoff-SI scheme takes longer time for the advertiser and the scanners to finish NDP since it uses the *ScanInterval* ($\tau_{SI}$) as a unit to compute the discovery latency. On the other hand, Backoff-AI scheme applies the same backoff method as Backoff-SI scheme [9], but there is a lower probability for successful discovery. As a result, Backoff-AI scheme increases the discovery latency higher than that of TCWS. The gap between TCWS and Backoff-AI scheme seems to be more obvious as the number of scanners increases. In brief, TCWS effectively decreases the mean discovery latency to improve NDP performance even though there are many scanners around the advertiser.

## VI. CONCLUSION

In this paper, we propose an enhanced scheme, TCWS, to improve the performance of NDP for the BLE advertising application. Since BLE Beacon technology becomes more pervasive, the number of BLE devices increases to cause significant performance degradation of NDP among an advertiser and scanners. We use two-way communication scheme to reduce resources wasting and improve the latency of NDP. Besides, the wait-slot scheme can distribute the transmissions of SCAN_REQ PDUs in different time to minimize the probability of collision. The results from the proposed model show that with this enhancement, the probability of successful discovery slowly degrades as the number of scanners increases. Thus, TCWS obtains much better performance for the discovery latency in dense BLE networks.

## Acknowledgment

The authors would like to thank the Ministry of Science and Technology, Republic of China (Taiwan), for financially supporting this research under Contract MOST 106-2221-E-005-016.

# REFERENCES

[1] P. Connolly, Principal Analyst at ABI Research and C. Sabin, Director of Business Strategy at Bluetooth SIG, et al., ”Beacon Technology 2017 Forecast: 7 Experts Share Their Thoughts,” [Online]. Available: https://kontakt.io/blog/beacons-2017-forecast-7-experts-share-their-thoughts/.

[2] J. Lindh, Bluetooth Low Energy Beacons, Texas Instruments, p. 2, Jan. 2015.

[3] K. H. Chang, Bluetooth: A viable solution for IoT? IEEE Wireless Commun., vol. 21, no. 6, pp. 6-7, Dec. 2014.

[4] The 2015 Reality of Retail Report [Online]. Available: http://www.inreality.com/resources/report/20150331/.

[5] SIG. Bluetooth, Bluetooth core specification version 4.2 [vol. 6] (2014).

[6] J. Liu, C. Chen, and Y. Ma, Modeling and Performance Analysis of Device Discovery in Bluetooth Low Energy Networks, in Proc. IEEE Global Commun. Conference, pp. 1538-1543, Dec. 2012.

[7] J. Liu, C. Chen, and Y. Ma, Modeling Neighbor Discovery in Bluetooth Low Energy Networks, IEEE Commun. Letters, vol. 16, no. 9, pp. 1439-1441, Sep. 2012.

[8] G. Park, W. Park, M. Hong, K. Cho, W. Cho, J. Seo, and K. Han, An Adaptive Parameter Setting Algorithm to Enhance Performance in Self-organizing Bluetooth Low Energy Networks, Wireless Personal Commun., vol. 87, no. 3, pp. 953-969, April 2016.

[9] J. Kim and K. Han, Backoff Scheme for Crowded Bluetooth Low Energy Networks, IET Commun., vol. 11, no. 4, pp. 548-557, March 2017.

[10] K. Cho, W. Park, M. Hong, G. Park, W. Cho, J. Seo, and K. Han, Analysis of Latency Performance of Bluetooth Low Energy (BLE) Networks, Sensors, vol. 15, no. 1, pp. 59-78, April 2015.

[11] K. Cho, G. Park, W. Cho, J. Seo, and K. Han, Performance Analysis of Device Discovery of Bluetooth Low Energy (BLE) Networks, Computer Commun., vol. 81, pp. 72-85, May 2016.

[12] W. S. Jeon, M. H. Dwijaksara, and D. G. Jeong, Performance Analysis of Neighbor Discovery Process in Bluetooth Low Energy Networks, IEEE Trans. Vehicle Technology, vol. 66, no. 2, pp. 1865-1871, Feb. 2017.

[13] J. Seo, K. Cho, W. Cho, G. Park, and K. Han, A Discovery Scheme Based on Carrier Sensing in Self-organizing Bluetooth Low Energy Networks, Journal of Network and Computer Applications, vol. 65, pp. 72-83, Feb. 2016.