WIRELESS COMMUNICATIONS AND MOBILE COMPUTING
Wirel. Commun. Mob. Comput. (2011)
Published online in Wiley Online Library (wileyonlinelibrary.com). DOI: 10.1002/wcm.1128

RESEARCH ARTICLE

PND: a p-persistent neighbor discovery protocol in wireless networks

Kyunghwi Kim[1], Heejun Roh[1], Wonjun Lee[1]*, Sinjae Lee[1] and Ding-Zhu Du[2]

[1] Department of Computer Science and Engineering, Korea University, Seoul, Korea
[2] Department of Computer Science, University of Texas at Dallas, Richardson, TX, U.S.A.

ABSTRACT

In wireless communications research, a number of literature assume that every node knows all of its neighbor nodes. To this end, neighbor discovery research has been conducted, but it still has room for improvement in terms of discovery delay. Furthermore, prior work has overlooked energy efficiency, which is considered as the critical factor in wireless devices or appliances. For better performance with respect to the discovery delay and energy efficiency, we proposed a novel p-persistent-based neighbor discovery protocol and devised a simple and light algorithm estimating the number of neighbor nodes to support the proposed protocol. Our protocol requires a lower delay and a smaller number of messages for the discovery process than the existing protocols. For extensive performance evaluation, we adopted extra comparison targets from other research areas within the same context. Copyright © 2011 John Wiley & Sons, Ltd.

KEYWORDS
neighbor discovery; p-persistent; new estimation algorithm

***Correspondence**
Wonjun Lee, Department of Computer Science and Engineering, Korea University, Seoul, Korea.
E-mail: wlee@korea.ac.kr

1. INTRODUCTION

The number of commercial off-the-shelf wireless devices has rapidly increased with the growth in capabilities, and electronic appliances are connected by wireless interfaces to enable them to be more intelligently operated. With these changes, *ad hoc* mode communication is receiving more interest in terms of delay and resource efficiency via direct traffic exchange compared with communication through specific centers or servers. Accordingly, neighbor discovery (or device discovery) plays a more important role in wireless communications. However, many researchers suggest protocols or methods based on the assumption that every node knows its neighboring nodes even in the topology control study. We can obtain the information of one-hop neighbor nodes through neighbor discovery, and this is used for the entire wireless systems such as Media Access Control (MAC) protocols, routing algorithms and topology control mechanisms. As a result, discovery delay has critical effects on the performance of systems using the above-mentioned protocols, algorithms and/or mechanisms. The degree of impact is more critical in mobile wireless networks because of frequent changes of neighbor nodes.

Several studies with respect to neighbor node discovery have been conducted [1–8], and most of them apply ALOHA-like schemes to minimize the expected time for identifying neighbor nodes as in [1]. To this end, researchers try to reduce the number of collisions, and neighbor discovery protocols can be thought of as a kind of collision avoidance research considering delay minimization and energy efficiency. For an extensive survey, we have expanded the research area to radio frequency identification (RFID) anti-collision algorithms [9–14]. Our simulation study includes algorithms as comparison targets.

Existing protocols mainly focus on the ALOHA [10–12] or tree-based schemes [13,14]. The transmission probabilities of the ALOHA-based and tree-based schemes are restricted to the number of slots and branches, respectively. On the other hand, p-persistent-based protocols can use correct and detailed values for transmission probabilities but have been only used to maximize network throughput or minimize average packet delay [15]. In this paper, we propose a p-persistent neighbor discovery (PND) protocol that utilizes the optimized transmission probability for minimizing the expected discovery delay and improving the energy efficiency of nodes. Further, we devise a transmission probability control (TPC) algorithm to allow the proposed protocol to be operated without information on the number of neighbor nodes or any density information on the network. Our two main contributions are as follows:

- The PND protocol adaptively adjusts the transmission probability with respect to the number of undiscovered nodes, and it can achieve fast and energy-efficient neighbor discovery with a small number of messages.
- The TPC algorithm enables the proposed protocol to be used without any prior knowledge of the number of neighbors. The algorithm was derived from slot probability analysis of p-persistent operations. Unlike existing protocols using the number of collision slots, we also utilized the number of idle slots for estimating undiscovered neighbors.

p-Persistent neighbor discovery shows better performance with an increasing number of neighbors than existing protocols, and it can guarantee the discovery of all neighbors, even in a multi-hop environment with a simple check process under a fixed frame size as in [1]. Moreover, the TPC algorithm enables the proposed protocol to be efficiently operated with an unknown number of neighbors through a fast approach of the transmission probability to the corresponding number of neighbors.

The remainder of this paper is organized as follows. Section 2 presents the system model, and the detailed description of the PND protocol is explained in Section 3. The analysis is described in Section 4, and Section 5 shows the extensive simulation results compared with the existing discovery protocols and the evaluation of the proposed control algorithm. The related work is introduced in Section 6. Concluding remarks are summarized in Section 7.

2. SYSTEM MODEL

In this paper, we consider a synchronous system in which time is divided into slots, and nodes are synchronized on slot boundaries. The operation of the proposed protocol is described in a clique that has size N for ease of understanding, and the protocol is applicable in a clique and even in a multi-hop environment. The PND uses a frame (which means the sum of several slots) and a period (i.e., the time between two consecutive success slots) in a similar manner to a phase and an epoch in [1], respectively.

Two network environments are dealt with in this paper: (i) only identical nodes are distributed, and (ii) a coordinator exists and triggers the discovery process to find its neighbors. The network environments are illustrated in Figure 1, and the gray-colored node in Figure 1(b) is the coordinator. In the first environment, there are N nodes in a clique, whereas the second one uses N neighbors within range of the coordinator. Even though the first environment is more general, we include the second one to show the practical results using the system parameters in the IEEE 802.11 standards [16], and the first one is used in multi-hop tests.

In each slot, a node in a system transmits its ID to be discovered by its neighbors until the termination condition is satisfied. Note that the condition is determined according to applied policies for feedback or message types. For example, the node stops sending its ID after receiving an acknowledgement message in the case using the explicit feedback policy. In a system, all nodes are operated with the same slot sizes, and the transmission probabilities only depend on the ratio of the collision slot size to idle slot size (see the following section). Therefore, the nodes have the same transmission probability as a function of the number of remaining (undiscovered) neighbors, derived in the analysis section (i.e., Equation (14)).

The PND can be applied with any message types or policies of feedback such as an explicit response message, busy tone or implicit feedback (a guard gap). The length of each slot type is determined according to what types or policies are used. Hence, PND may have different lengths for different slot types and can be operated under two slot size configurations, *homogeneous and heterogeneous*.

Based on the binomial distribution, the following equations are used to analyze the expected discovery delay for the discovery protocols.

$$\begin{cases} P_{\mathrm{I}}(n,p) = (1-p)^{n} \\ P_{\mathrm{S}}(n,p) = np(1-p)^{n-1} \\ P_{\mathrm{C}}(n,p) = 1-(1-p+np)(1-p)^{n-1} \end{cases} \tag{1}$$

As defined in Table I, p is the transmission probability, which is the most critical factor governing the performance of the discovery process. The other parameters and terms used in this paper are presented in Table I. These equations were formulated by using the probability mass function of the binomial distribution.

3. p-PERSISTENT NEIGHBOR DISCOVERY PROTOCOL

The proposed protocol was operated with feedback of neighboring nodes in a similar manner to protocols applying collision detection by neighbor nodes' responses in [1]. Without the feedback, the expected discovery delay is significantly increased because of duplicate discoveries of the same nodes, and we will show this performance degradation in the simulation results. In this section, the PND protocol is explained for the cases of known and unknown N values.

3.1. p-Persistent neighbor discovery protocol with known number of neighbor nodes

We described the detailed procedure of PND under the assumption that the number of neighbors is known. This assumption is subsequently relaxed in the following

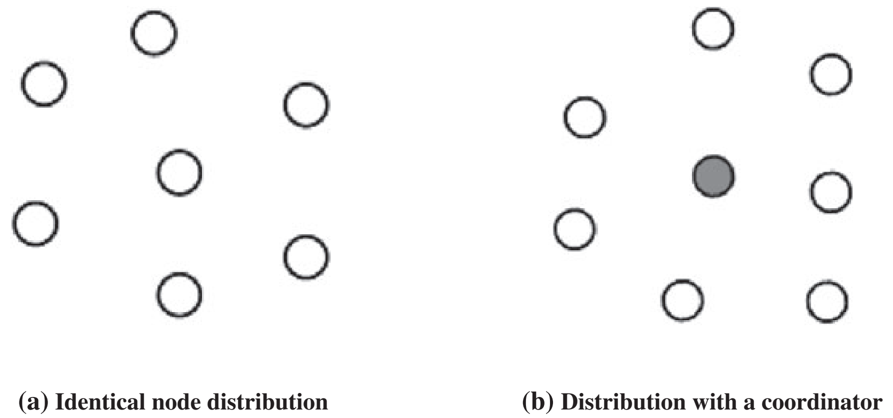


Figure 1. Network environments. (a) Identical node distribution. (b) Distribution with a coordinator.

Table I. Parameters and terms.

Parameter/term	Definition
p	Transmission probability
N	Clique size
k	Number of discovered nodes ($k \leq N$)
n	Number of undiscovered nodes ($n + k = N$)
$P_C/P_I/P_S$	Slot probabilities for each slot type (collision/idle/success)
$T_C/T_I/T_S$	Slot time for each slot type including delay of all its required procedures (collision/idle/success)
γ	Ratio of collision slot time to idle slot time
ρ	Ratio of success slot time to idle slot time

sections. A node that has succeeded in the transmission of its ID turns to the listen mode and does not send any more messages for discovery. The advantage of feedback is to prevent the rapid growth of the discovery delay, where the number of discovered nodes (k) is close to the clique size (N).

The simplest way for applying feedback is to use the response messages for all slot types (homogeneous slot size), which utilizes the longest slot size for the configuration. On the other hand, for the heterogeneous slot size configuration, we applied a constant duration for an idle slot, which is shorter than the transmission time of any other messages used in the discovery process.

The discovery process consists of several periods. A period has only one success slot and zero or more collision and idle slots. It is similar to the renewal period in the renewal stochastic process because the period sizes are independent and identically distributed random variables with non-negative values. This period structure allows us to analyze the discovery delay more easily. The expected average discovery delay of the PND protocol is formulated by the sum of the expected delay of each period:

$$E\left[T_N\right] = \sum_{k=1}^{N} E\left[T_{\text{Period}}(k)\right] \tag{2}$$

Without loss of generality, the total expected delay is described by

$$\begin{aligned} E[T_{\text{Period}}(k)] &= T_I \cdot E\left[N_I(k)\right] + T_C \cdot E\left[N_C(k)\right] + T_S \\ &= T_I \left(E\left[N_I(k)\right] + \gamma \cdot E\left[N_C(k)\right] + \rho\right) \\ &= T_I \left[\frac{(1-p)+\gamma\left((1-p)^{-n+1}-1+p\right)}{np} + (\rho-\gamma)\right] \end{aligned} \tag{3}$$

where $E\left[N_I(k)\right] = P_I/P_S$ and $E\left[N_C(k)\right] = P_C/P_S$.

The used terms are listed in Table I. To minimize Equation (3), the following equation is obtained by differentiation:

$$\frac{\partial E\left[T_{\text{Period}}(k)\right]}{\partial p} = T_I \frac{(1-p)^{-n}\left(\gamma\left(np + (1-p)^{-n} - 1\right) - (1-p)^n\right)}{np^2} \tag{4}$$

From the above equation, we can infer that the value of p depends sensitively on the number of competing nodes (undiscovered nodes) as well as γ. The total expected discovery delay can be minimized by choosing the optimal p value. We derived the value of p for each n with respect to γ via numerical analysis, and the results will be shown in Section 5. The approximated result is also derived in the following section.

The optimal p value changes with decreasing n during run time; thus, it should be adjusted with an increase of k to obtain the optimal discovery delay. In other words, the transmission probability increases with the number of discovered nodes.

3.2. p-Persistent neighbor discovery protocol with unknown number of neighbor nodes

The PND protocol can achieve very efficient performance by optimizing the delay of neighbor discovery. For practical applications, a transmission probability control algorithm should be applied with the estimation algorithm of the number of neighbor nodes. Unless the algorithm provides appropriate control of p, the performance of the proposed protocol will be inferior. In this section, we propose a simple and efficient control algorithm developed for the PND protocol.

3.2.1. Transmission probability control algorithm

The proposed algorithm is based on the analysis of the slot probability, and it adjusts the transmission probability with respect to the number of neighbors. To this end, the following condition is required (see the next section for the proof of the condition).

Condition. On an interval (0, 1) of p, the expected numbers of the collision and idle slots have to be either increasing or decreasing functions of p and n.

The occurrence probability of the collision slot is an increasing function, whereas that of the idle slot is a decreasing function, and these features enabled us to make transmission probability control algorithms. For the given value of n and the corresponding optimal transmission probability p, the collision and idle slot probabilities are shown within specific ranges. Hence, we can set thresholds for these slot probabilities, and the TPC algorithm operates by increasing or decreasing the transmission probability based on the thresholds. The operation of nodes in the PND protocol with the TPC algorithm is as follows:

(1) In the first frame, the nodes use the initial transmission probability, and then they update it for each frame.

(2) For each slot, each node decides either to send its ID (**SEND** mode) or to listen to the medium (**LISTEN** mode) by comparing a generated random number to the current transmission probability.

   (2-1) **SEND** mode: After the node sends its ID message, the following occurs:

      i. Feedback message reception for success: It finishes all procedures for neighbor discovery and enters to the listen mode until the end of the discovery process.

      ii. Feedback message reception for collision: It increases the number of collision slots and returns to step 1.

   (2-2) **LISTEN** mode: If the node does not try to send its ID (idle), it has to determine the state of the current slot via the feedback message and return to step 1.

The transmission probability update was conducted in a distributed manner and does not require any additional signal exchanges. Therefore, the TPC algorithm is simple and light for nodes. The number of undiscovered nodes is estimated as follows:

$$n_{i+1} = n_i - s_i + c_i - i_i + IDLE_{\text{th}} - COLL_{\text{th}} \quad (5)$$

The value n of the next frame was determined using the number of success (s), collision (c) and idle (i) slots in the current frame and the thresholds of idle ($IDLE_{\text{th}}$) and collision ($COLL_{\text{th}}$) slots. The thresholds were determined by multiplying the slot probabilities by the length of the frame. With the optimal transmission probability, the numbers of idle and collision slots should approach to the thresholds, respectively. Hence, exploiting the differences, ($IDLE_{\text{th}} - i_i$) and ($COLL_{\text{th}} - c_i$), makes the estimation more correct after each frame. In Equation (5), the differences of idle and collision slots are applied with a plus and minus sign, respectively, because idle and collision slot probabilities have opposite properties; the idle slot probability is a decreasing function, whereas the collision slot probability is an increasing function. This feature will be explained in detail in the following section.

If the value n of the next frame is smaller than 1, it is changed to 1. After estimating the number of undiscovered nodes, the TPC selects the value p corresponding to the number of n_{i+1} based on the result of the next section (i.e., Equation (14)). Where only the last node exists, the transmission probability has to be 1 for fast identification of it. However, the estimation of this value is not always perfect, and thus, we set the probability to 0.5 for the one-neighbor node case in the performance evaluation.

4. ANALYSIS OF TRANSMISSION PROBABILITY CONTROL ALGORITHM

4.1. Feature of per-slot probability

In this section, partial differentiation for the collision and idle probabilities per slot is applied. The optimal transmission probability has the largest value when the clique size N is 2. The result is 0.5 for a homogeneous slot size configuration and less than 0.5 in a heterogeneous one. Hence, the following conditions are obtained: $0 < p \leq 0.5$, $\ln(1-p) < -0.69$, $1 < (1 - p + Np)$ and $2 \leq n$. Using these results, we can derive the following results:

Idle probability:

$$\frac{\partial P_{\mathrm{I}}(n,p)}{\partial p} = -n(1-p)^{n-1} < 0 \tag{6}$$

$$\frac{\partial P_{\mathrm{I}}(n,p)}{\partial n} = (1-p)^{n}\ln(1-p) < 0 \tag{7}$$

Therefore, the idle slot probability is a decreasing function of n and p.

Collision probability:

$$\frac{\partial P_{\mathrm{C}}(n,p)}{\partial p} = (n-1)\,np\,(1-p)^{n-2} > 0 \tag{8}$$

$$\frac{\partial P_{\mathrm{C}}(n,p)}{\partial n} = -(1-p)^{n-1}\left[p + (1-p+np)\ln(1-p)\right] > 0 \tag{9}$$

Therefore, the collision slot probability is an increasing function of n and p.

4.2. Approximated slot probabilities to calculate thresholds for transmission probability control algorithm

The expected number of slots follows the binomial distribution, and we can apply the Poisson approximation. Therefore, Equation (1) is re-formulated as follows:

$$\begin{cases} P_{\mathrm{I}}(n,p) = (1-p)^{n} \approx \mathrm{e}^{-np} \\ P_{\mathrm{S}}(n,p) = np(1-p)^{n-1} \approx np\cdot\mathrm{e}^{-np} \\ P_{\mathrm{C}}(n,p) \approx 1-(1+np)\,\mathrm{e}^{-np} \end{cases} \tag{10}$$

The expected delay of each period is calculated with Equation (10) as follows:

$$\begin{aligned} E\left[T_{\mathrm{Period}}\right] &= T_{\mathrm{I}}\left(E\left[N_{\mathrm{I}}\right] + \gamma\cdot E\left[N_{\mathrm{C}}\right] + \rho\right) \\ &= T_{\mathrm{I}}\left[\frac{1-\gamma+\gamma\cdot\mathrm{e}^{np}}{np} + (\rho-\gamma)\right] \end{aligned} \tag{11}$$

By differentiation, we can derive the approximated value of p ($\hat{p}^{*}$).

$$\mathrm{e}^{n\hat{p}^{*}}\left(1-n\hat{p}^{*}\right) = \frac{\gamma-1}{\gamma} \tag{12}$$

Using the Taylor series approximation, we can simplify Equation (12) and derive the approximated value of p as follows:

$$\left(1+n\hat{p}^{*}\right)\left(1-n\hat{p}^{*}\right) = \frac{\gamma-1}{\gamma} \tag{13}$$

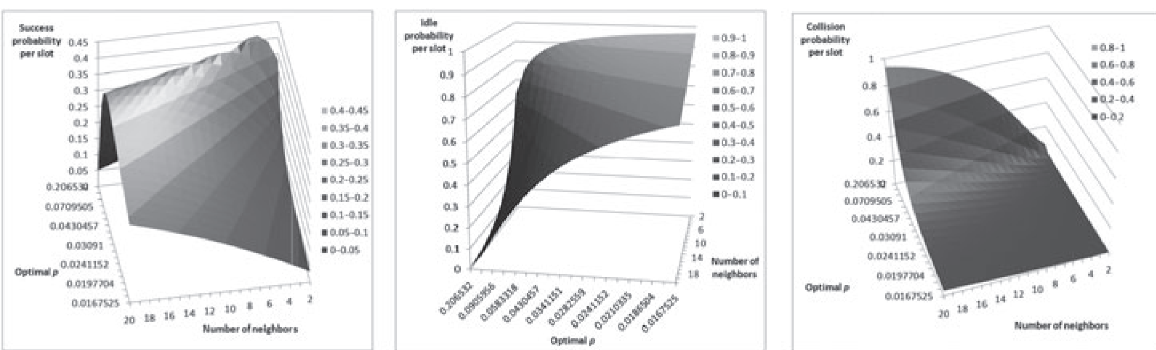
$$\hat{p}^{*} = \frac{\left(\sqrt{1/\gamma}\right)}{n} \tag{14}$$

With the approximated value of p, the expected slot probabilities are given by

$$\begin{cases} P_{\mathrm{I}}(n,p) \approx \mathrm{e}^{-\left(\sqrt{1/\gamma}\right)} \\ P_{\mathrm{S}}(n,p) \approx \left(\sqrt{1/\gamma}\right)\mathrm{e}^{-\left(\sqrt{1/\gamma}\right)} \\ P_{\mathrm{C}}(n,p) \approx 1-\left(1+\left(\sqrt{1/\gamma}\right)\right)\mathrm{e}^{-\left(\sqrt{1/\gamma}\right)} \end{cases} \tag{15}$$

We can calculate the values of the expected slot probabilities for heterogeneous ($\gamma = 14.76$) and homogeneous ($\gamma = 1$) slot size configurations. The derivation of γ for the heterogeneous configuration will be explained in the next section. The optimal results were obtained from numerical analysis, whereas the approximated ones were calculated from Equation (15).

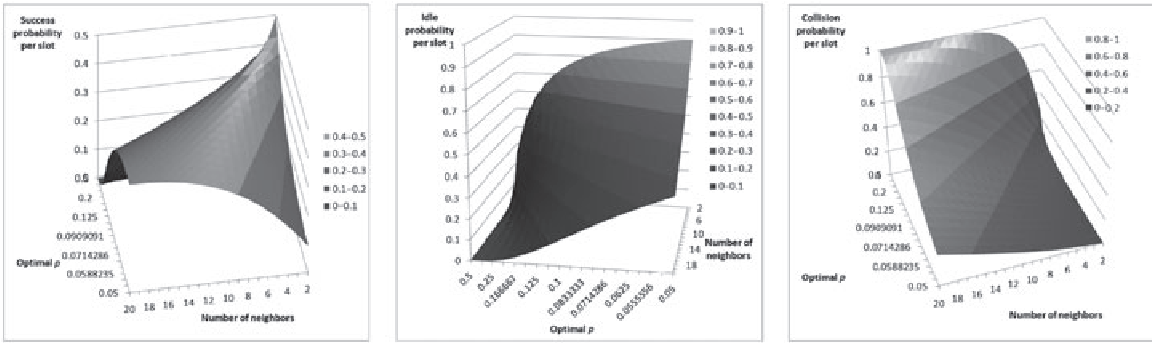
As shown in Figures 2(a) and 3(a), the graph of the success slot is neither increasing nor decreasing, and there are two possibilities for the number of neighbors with a given transmission probability; thus, the number of success slots cannot be used as a criterion to control the transmission probability. However, as shown in Figures 2(b), (c) and 3(b), (c), the idle and collision probabilities are decreasing and increasing functions, respectively, which



(a) (b) (c)

Figure 2. Slot probabilities (heterogeneous): (a) success slot, (b) idle slot and (c) collision slot.

Wirel. Commun. Mob. Comput. (2011) © 2011 John Wiley & Sons, Ltd.
DOI: 10.1002/wcm



(a) **(b)** **(c)**

Figure 3. Slot probabilities (homogeneous): (a) success slot, (b) idle slot and (c) collision slot.

are within a narrow range with respect to each n value and corresponding p value, as illustrated in Table II.

The collision slot probability has a lower variance than the idle one, but it is low in the heterogeneous configuration; thus, we utilized both parameters for estimating the number of undiscovered nodes, unlike traditional ALOHA-based protocols. The difference between the analyzed and optimal values is caused by the approximation process. However, the approximation error does not greatly affect the setup of thresholds. This will be explained in the next section.

5. PERFORMANCE EVALUATION

We used a 1-Mbps channel bit rate to send a message, and some parameters of the IEEE 802.11 standard were applied such as that in [17]. The length of the ID was set to 32 bits, and the management frame format was utilized to send the ID information and the feedback message. Hence, the feedback message size including the 2-bit notification for slot types was 226 bits. The transmission delay of each slot type is as follows, and it can be changed according to the system requirements.

- Success and collision slot time including feedback message:

$$T_S = T_C = 2\text{PHY\_HDR} + 2\text{MAC\_HDR} + \frac{L_{ID} + 2}{1 \times 10^6} = 738\,\mu\text{s}$$

- Idle slot time:

$$T_I = aSlotTime = 50\,\mu\text{s}$$

From the above results, γ is 14.76 in the heterogeneous configuration. For the non-feedback neighbor discovery protocol (NF in graphs), we used 384 µs for the success and collision slot delay without feedback message ($\gamma = 7.68$), and *aSlotTime* was used for the idle slot time in the heterogeneous configuration. On the other hand, the homogeneous configuration for NF applies 384 µs for all slot types.

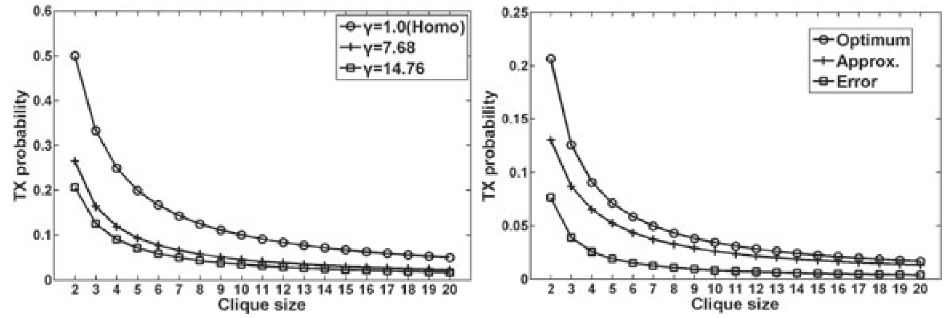
5.1. Transmission probability

Figure 4(a) shows the optimal probability p with respect to the number of neighbors for two slot size configurations of PND and NF. The values for the homogeneous configuration are higher than those for the heterogeneous one; the difference is greatest when the number of neighbors is 2 ($0.5 - 0.20653 = 0.293468$), but it is decreased with increasing n. It means that the effect of γ is reduced according to the increase of the number of neighbors. On the other hand, the ratio of the transmission probabilities for 2 to 20 neighbors in each configuration increases 10-fold. This also means that a transmission probability control algorithm must be carefully designed to consider the number of undiscovered neighbors.

In Figure 4(b), we also illustrate the optimal and approximated probabilities in the heterogeneous configuration. (There is no difference in the homogeneous one, and NF has a similar graph to PND in the heterogeneous configuration.) Because the approximated probability is derived under the assumption that n is large, the difference between the optimal and approximated values decreases with increasing n; the error is lower than 0.1 with two

Table II. Comparison of slot probabilities with respect to configurations.

	Heterogeneous ($\gamma = 14.76$)		Homogeneous ($\gamma = 1$)	
	P_I	P_C	P_I	P_C
Optimum	0.63–0.71	0.04	0.25–0.36	0.25–0.26
Approximation	0.77	0.03	0.37	0.26



(a) **Optimal *p* value for two configurations of PND and NF** **(b)** **Optimal vs. approximated *p* values ($\gamma = 14.76$)**

Figure 4. Optimal and approximated *p* values. (a) Optimal *p* value for two configurations of p-persistent neighbor discovery and non-feedback protocols. (b) Optimal versus approximated *p* values ($\gamma = 14.76$).

neighbors and below 0.01 with eight or more neighbors. The effect of the approximation error is described in the next section.

5.2. Comparison with known number of neighbor nodes

In the following simulation tests, each data point is averaged over 1000 runs. Figure 5 shows the effect of the approximation error of p in the heterogeneous configuration for PND and NF. As mentioned earlier, the homogeneous configuration has no error, and we do not include its graph. In both protocols, the approximation error has little influence on the discovery delay for any clique size N. Hence, we can conclude that the approximated p value is very suitable for the neighbor discovery process.

In Figure 6, we compare two configurations with clique size 20 for the PND and NF protocols. The common feature of NF with both configurations is that the delay graphs begin with a lower delay than that of PND, but the slope of the graphs is significantly increased when the number of discovered nodes is close to 100%. The cause of this feature is the duplicate success slots. Therefore, the use of feedback is more efficient for the neighbor discovery process that wants to identify all neighbors. For the other setup of N, a similar trend has occurred on the delay graphs, and the increase of slope is higher for a larger N. Figure 7 also shows that the protocols in the heterogeneous configuration are better than the corresponding protocols in the homogeneous one because of the smaller size of the idle slot. That is, the heterogeneous configuration is less restricted in terms of the slot size setup and provides faster neighbor discovery.

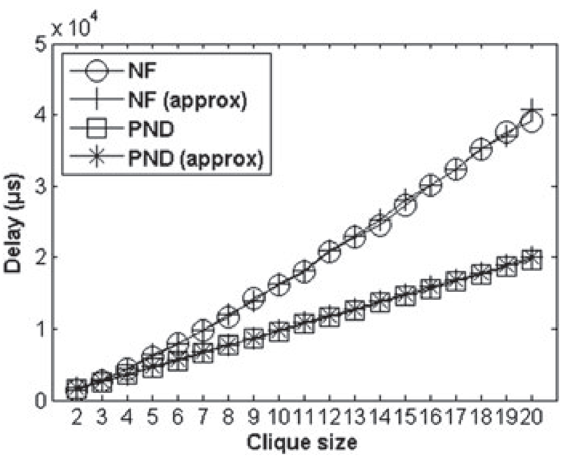


Figure 5. Effect of approximation error of *p* on expected delay.

5.3. Comparison with unknown number of neighbor nodes

We chose a protocol based on dynamic framed slotted ALOHA by using 2.3922 for the estimation factor in [11] and another one by using the collision detection method in [1]. For convenience, we call the former dynamic framed length ALOHA (DFLA) and the latter ALOHA-like neighbor discovery (AND) in this paper. DFLA is the representative collision avoidance algorithm that can reduce the expected delay, and its estimation factor is well known for high accuracy. AND is the state-of-the-art protocol in neighbor discovery research, which can be operated efficiently in the general multi-hop environment. DFLA and AND are operated in the homogeneous configuration, but we included their modified versions for the heterogeneous configuration in the simulation tests. As mentioned previously, we used a value of 0.5 for the transmission probability for one undiscovered node, because estimation of n is not always perfect.

We used a value of 10 for the initial frame size of DFLA and the fixed frame size of PND. The threshold for the number of collision and idle slots was set to 1 and 7,

Wirel. Commun. Mob. Comput. (2011) © 2011 John Wiley & Sons, Ltd.
DOI: 10.1002/wcm

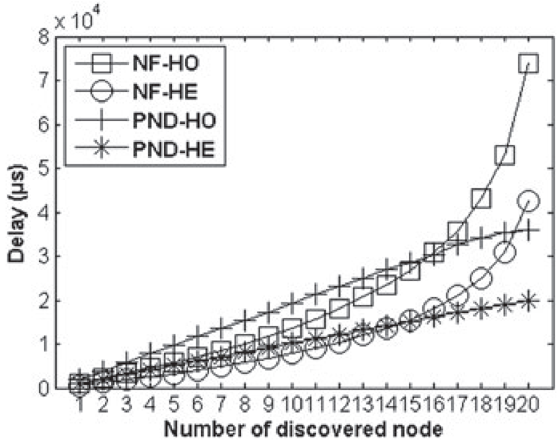
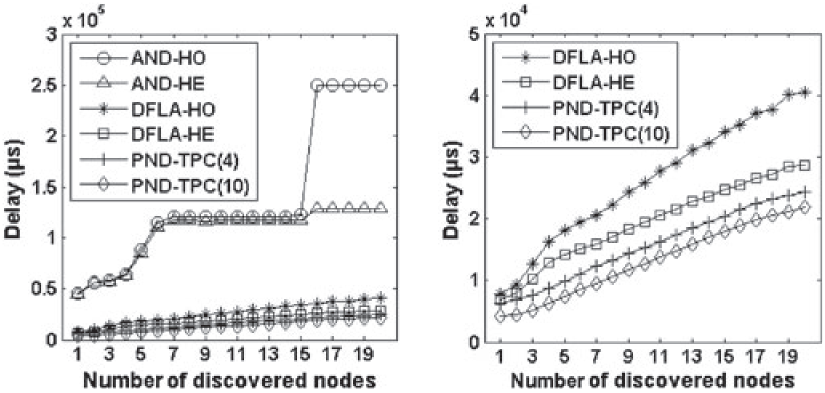


Figure 6. Performance of p-persistent neighbor discovery and non-feedback protocols with 20 neighbors (HO, homogeneous; HE, heterogeneous).



(a) 20 neighbors **(b)** Selected protocols

Figure 7. Average discovery delay with fixed N. (a) Twenty neighbors and (b) selected protocols.

respectively, based on the results of analysis. This threshold setup is changed for tests in a multi-hop environment, because the homogeneous configuration is used there. We applied 4 and 10 for the initial value n of p_n. When n of p_n is 4, the initial transmission probability of PND is about 0.09, which is close to that of DFLA at the first frame; thus, we use 4 for the initial value of n. The reason that we applied 10 for the value n of p_n is to show the effect of the setup.

The simulation setup for AND is as follows:

- Frame size at each level (m) of phase

$$F_{\text{AND}} = 2^{m+1}e$$

- Transmission probability for each frame:

$$P_{\text{AND}} = 1/\left(2^m - i\right)$$

where i is the number of discovered neighbors. These control formulas are introduced and explained in [1].

Figure 7 illustrates the average discovery delay of neighbor discovery protocols with fixed clique sizes ($N = 20$). The graph abbreviations HO and HE denote homogeneous and heterogeneous configurations, respectively, and the numbers in parentheses are the initial values n of p_n for the TPC algorithm. In this simulation test, the protocols are not finished immediately after the last success but after the frame containing the last success slot, because they do not know the total number of neighbors.

In Figure 7(a), AND dramatically increases at some points, whereas the others rise smoothly according to the increase of the number of discovered nodes. This feature of the AND graph is caused by the policy for adjusting the frame length, which extends the length by a factor of 2. Based on the results of AND, from the difference between the graphs of the two configurations, we also found that most of the idle slots have occurred in the last phase.

The results of DFLA and PND are obtained and shown in Figure 7(b). We can know that PND has a smaller number of idle slots than DFLA, because the difference of the two configurations is lower for PND. DFLA only considers the number of collision slots, and PND deals with the number of collision and idle slots, which is why our protocol has better performance for discovery delay than DFLA.

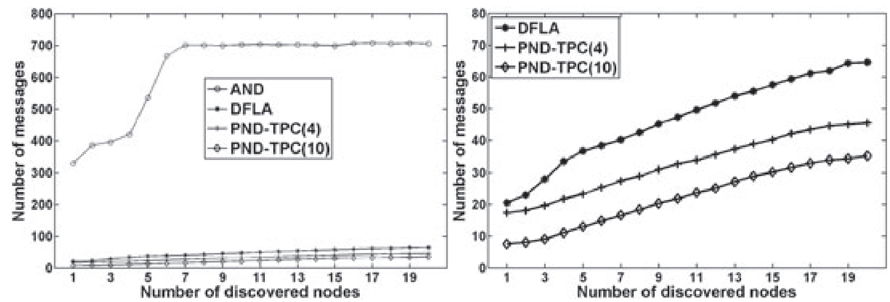
In Figure 8, we illustrate the number of transmitted messages for each protocol. The number of received messages is not recorded, because it can be calculated by multiplying the number of transmitted messages and $N-1$. The configuration type does not affect the number of messages. As shown in Figure 8(a), the graphs of AND maintain a low slope after certain points, and we can infer that the number of discovered nodes is decreasing. From the graphs in Figure 8(b), we find that PND-TPC(10) uses fewer messages, because it begins with an n value closer to N than PND-TPC(4). The difference between DFLA and PND is greater when the number of discovered nodes is closer to that of the total neighbor nodes. As a result, DFLA uses twice as many messages as PND-TPC(10), and the total energy consumption including message reception is most efficient in the PND protocol.

We show the average discovery delay for each protocol with respect to the clique size N in Figure 9. The graphs of AND rise abruptly at certain points (5, 9 and 17), as shown in Figure 9(a), and this is caused by the policy for controlling the frame length in AND. The large difference between the two configurations in AND is due to the method of regulating the transmission probability. In other words, AND generates many idle slots. On the contrary, the results of PND and DFLA increase smoothly. As shown in Figure 9(b), PND is also more energy efficient than DFLA, because the optimal transmission probability is applied.

The discovery delay of PND is normalized by the result obtained in the case of a known number of neighbors, and it is shown in Figure 10 with respect to the two values of the initial n of p_n. We obtained the PND delay immediately after the last success (not the last frame) in order to show the effect of TPC precisely. The graph of PND-TPC(4) is stable and has low values at points around 4. It is smaller than PND-TPC(10) with fewer than eight nodes, whereas PND-TPC(10) has better performance at $N > 7$. Hence, the setup of n for p_n must consider the average node degree of the network. Except for low density environments, the discovery delay of PND-TPC(10) is increased by less than 10% compared with the case of a known number of neighbors. The results show that the TPC algorithm is effective.

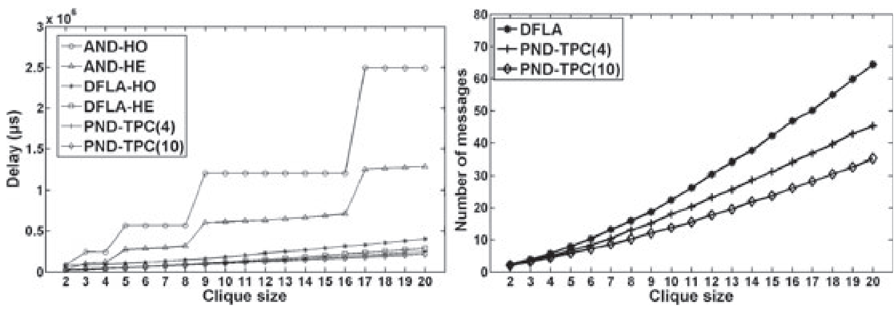
5.4. Multi-hop tests

Two protocols can be considered as comparison targets: AND and DFLA. In multi-hop environments, PND applies



(a) 20 neighbors (b) Selected protocols

Figure 8. Number of transmitted messages with fixed *N*. (a) Twenty neighbors and (b) selected protocols.



(a) Average discovery delay (b) # of transmitted messages

Figure 9. Comparison of delay and energy with *N* values ranging from 2 to 20. (a) Average discovery delay and (b) number of transmitted messages.

Wirel. Commun. Mob. Comput. (2011) © 2011 John Wiley & Sons, Ltd.
DOI: 10.1002/wcm

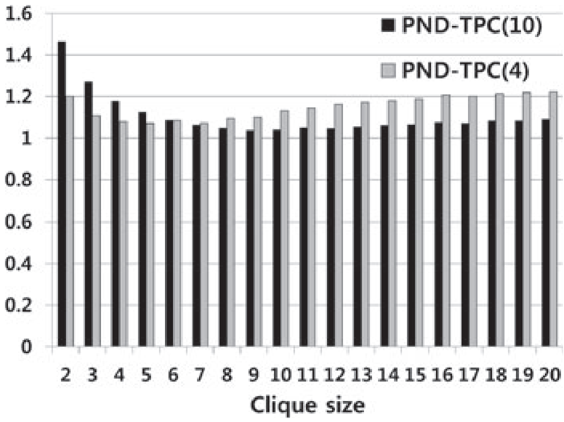


Figure 10. Normalized delay.

a homogeneous slot size configuration because using a heterogeneous setup resolves some problems such as partial collision, difficult separation of collision slots and loss of transmission chance. Terminating the discovery process is also a very hard problem, whereas it is easy for a homogeneous configuration, because each frame has the same length for all nodes in the PND protocol. Because AND applied a similar environment in [1] and uses a fixed size for each frame, it is an appropriate comparison target. On the other hand, DFLA is not suitable for a multi-hop simulation. Each node based on DFLA calculates its next frame size, and thus, every node may have a different frame length, which generates a different ending time for each frame. In that case, DFLA requires some terminating conditions that are not easy to guarantee 100% neighbor discovery; thus, we excluded it from the comparison targets. The other protocols based on a non-feedback procedure are not considered because of the need for excluding DFLA.

For utilizing the homogeneous configuration, we set the thresholds of the collision and idle slots to 3 based on the results of analysis. Because PND and AND can use the same policy for feedback (or collision detection in [1]), we compared the number of slots and messages for the discovery process. We performed a test with 200 to 1800 nodes that are uniformly distributed, and the interval was set to 200. The test area was 3 km × 3 km, and the transmission range of each node was 150 m.

As shown in Figures 11 and 12, the *x*-axes do not represent the number of neighbors but rather the average number of neighbors, representing a variety of distributions for each value. We obtained each data point in the graphs by averaging over 100 runs with a distribution of the given number of neighbors. For each value, we conducted tests with a five-node distribution. Therefore, five adjacent points were based on the same value. Figure 11 illustrates that the delay of PND increases smoothly, whereas that of AND rises abruptly at certain positions, because AND extends the length of frame size by a factor of 2. The number of AND messages also rises more quickly for both transmission and reception, as shown in Figure 12.

We summarize the reasons why PND can achieve better performance than AND.

- p-Persistent neighbor discovery adaptively approaches the number of neighbors, whereas AND increases the phase in a strict sequential manner; thus, it has a longer delay and much more messages than PND.
- The transmission probability control of AND restricts the number of discovered nodes for each phase, for example, one node for the first phase, three nodes for the second phase, and seven nodes for the third phase. On the other hand, PND uses a value of 0.5 for p considering the errors in estimation of undiscovered nodes of the TPC algorithm.
- ALOHA-like neighbor discovery does not reflect the number of discovered nodes in frame size control. As mentioned previously, AND has many idle slots in the last phase, but PND utilizes the number of success nodes for controlling the transmission probability.

The authors in [1] constructed and analyzed their neighbor discovery protocol effectively with respect to the many conditions and assumptions. However, the protocol, considering unknown neighbors and feedback (collision detection), missed certain points, as summarized.

6. RELATED WORK

The proposed protocol in [2] is used for a large number of wireless sensors dropped several times by an airplane. Because of the limited energy of the used devices in their scenario, the proposed birthday protocol considered not only successful transmissions but also sleep time for energy saving. In the proposed protocol, nodes are operated under three modes in a slotted manner: birthday-listen (BL), birthday-listen-transmit (BLT) and probabilistic round robin (PRR). The authors analyzed the BL and BLT modes; however, the results are limited to the case that the transmission and listen probabilities are the same.

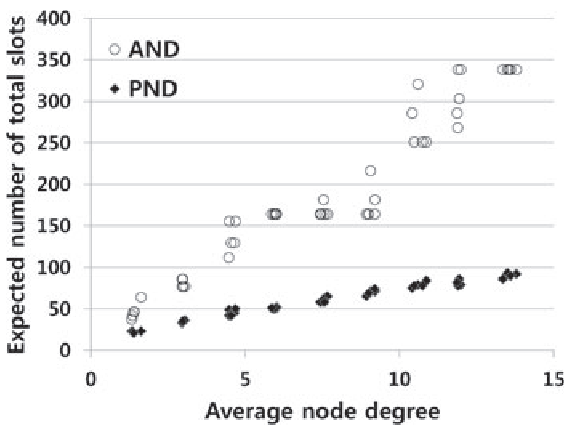


Figure 11. Comparison of the expected number of slots.

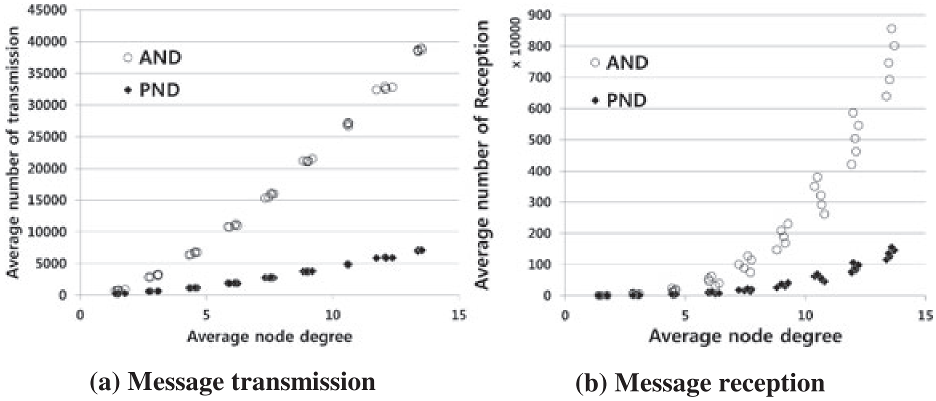


Figure 12. Comparison of the average number of transmitted/received messages. (a) Message transmission and (b) message reception.

The analysis of PRR mode shows the same result with the slotted ALOHA-based protocol. Several kinds of node discovery protocols were proposed and analyzed in [3]. The system model applies the slotted structure, and five protocols are described. Among the proposed protocols, several are ineffective when the clique size is more than 2; the random protocol operates in a manner similar to slotted ALOHA, nodes have to be in listen mode for a slot after transmitting a message with the listen after talking protocol and the sleep protocol uses the backoff approach.

The authors of [1] proposed ALOHA-like neighbor discovery schemes and analyzed the expected time for discovery under diverse conditions and assumptions. Their work includes a collision detection-based scheme using the receiver's status feedback to improve the delay performance through adaptively controlled transmission probability. Moreover, decreasing transmission probability and increasing frame size with changes of phase are applied for the case of an unknown number of neighbors. Because [1] is the state-of-the-art in neighbor discovery research and superior to the protocols in [2,3], one of the protocols in [1], which considers collision detection and an unknown number of neighbors, was used as a comparison target in this paper. Recently, most of the work in the neighbor discovery research area focuses on specific network environments such as cellular communication systems [18] or sensor networks [19]. The proposed scheme of [18] is operated by a based station by using the spatial correlation of wireless channel in a centralized manner, and the neighbor discovery protocol of [19] is presented for the mobile sensors considering low duty cycles. To the best of our knowledge, the neighbor discovery protocols in [1] is the most recent work that can be operated in the general environment, which is directly related to our proposed protocol.

In [11], an estimation method for the number of tags was proposed for dynamic frame length ALOHA. To achieve the maximum throughput, the transmission probability p is set to $1/N$ where N is the number of nodes. The authors calculated the collision rate with p, and they derived the number of collided tags from this result. Thus, the method estimates the number of estimated tags through multiplying 2.3922 by the number of collision slots and determines the length of the next frame. The authors in [12] suggested a dynamic framed slotted ALOHA algorithm, which includes a tag estimation method for estimating the number of tags within a reader's range. For application of the estimation algorithm, the reader has to store the mapping table for the collision ratio and the number of tags. In RFID networks, because the size of each frame is limited by a factor of 2, the required memory size is small. On the other hand, the frame length can be any positive integer in the other wireless networks, and the required memory size is $O(N^2)$. Furthermore, as shown in the simulation results of [12], the performance difference is negligible compared with the method in [11], even with a large number of nodes. We used the method of [11] as another comparison target in this paper. Therefore, the simulation results include comparison targets not only in neighbor discovery studies but also in RFID anti-collision research.

7. CONCLUSION

Neighbor discovery is a fundamental research issue of great importance to wireless networking systems. Moreover, its importance has been rising with the increasing number of wireless devices, in terms of the discovery delay and energy efficiency. We have presented a novel neighbor discovery protocol in this paper, which is based on p-persistent collision avoidance. We also derived approximated values of the transmission probabilities for the PND protocol, and analyses for the proposed protocol and algorithm were presented. Our performance evaluation extensively covered the state-of-the-art work in the given research area and even in other research areas within the same context. Our scheme shows the best performance with respect to the expected discovery delay and energy efficiency in a clique and even in multi-hop environments compared with the existing protocols. The TPC algorithm supports the proposed protocol with a fast approach to the number of undiscovered neighbor nodes. Our protocol and algorithm can be applied in many kinds of wireless systems.

ACKNOWLEDGEMENTS

This research was jointly supported by the second BK 21 Project (grant no. T0801491); the LG Yonam Foundation; MEST, Korea under WCU (R33-2008-000-10044-0); a KOSEF Grant funded by the Korean Government (MEST) (No. R01-2007-000-11203-0); a KRF Grant (KRF-2008-314-D00354) and MKE, Korea under Information Technology Research Center NIPA-2010-(C1090-1021-0008).

REFERENCES

1. Vasudevan S, Towsley D, Goeckel D, Khalili R. Neighbor discovery in wireless networks and the coupon collector's problem. In *Proceedings of the 15th Annual International Conference on Mobile Computing and Networking (MOBICOM'09),* Beijing. ACM: New York, NY, 2009; 181–192.
2. McGlynn MJ, Borbash SA. Birthday protocols for low energy deployment and flexible neighbor discovery in ad hoc wireless networks. In *Proceedings of the 2nd ACM International Symposium on Mobile Ad Hoc Networking & Computing (MOBIHOC'01),* Long Beach, CA. ACM: New York, NY, 2001; 137–145.
3. Alonso G, Kranakis E, Wattenhofer R, Widmayer P. Probabilistic protocols for node discovery in ad-hoc, single broadcast channel networks. In *Proceedings of the 17th International Parallel and Distributed Processing Symposium (IPDPS'03),* Nice. IEEE Computer Society: Washington, DC, 2003.
4. Kohvakka M, Suhonen J, Kuorilehto M, Kaseva VA, Hännikäinen M, Hämäläinen TD. Energy-efficient neighbor discovery protocol for mobile wireless sensor networks. *Ad Hoc Networks* 2006; **7**: 24 – 41.
5. Vasudevan S, Kurose J, Towsley D. On neighbor discovery in wireless networks with directional antennas. In *Proceedings of the 24th Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM 2005),* Miami, FL. IEEE: Piscataway, NJ, 2005; 2502–2512.
6. Borbash SA, Ephremides A, McGlynn MJ. An asynchronous neighbor discovery algorithm for wireless sensor networks. *Ad Hoc Networks* 2007; **5**: 998 –1016.
7. Keshavarzian A, Uysal-Biyikoglu E, Herrmann F, Manjeshwar A. Energy-efficient link assessment in wireless sensor networks. In *Twenty-third Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM 2004),* Hong Kong. IEEE: Piscataway, NJ, 2004; 1751–1761.
8. Salonidis T, Bhagwar P, Tassiulas L. Proximity awareness and fast connection establishment in bluetooth. In *Proceedings of the First ACM Annual Workshop on Mobile Ad Hoc Networking and Computing (MOBIHOC'00),* Boston, MA. IEEE: Piscataway, NJ, 2000; 141–142.
9. Shih D, Sun PL, Yen DC, Huang SM. Taxonomy and survey of RFID anti-collision protocols. *Computer and Communications* 2006; **29**: 2150 –2166.
10. Vogt H. Efficient object identification with passive RFID tags. In *Proceedings of International Conference on Pervasive Computing*. Springer-Verlag: London, 2002; 98 –113.
11. Schoute FC. Dynamic frame length ALOHA. *IEEE Transactions on Communications* 1983; **4**: 565–568.
12. Cha JR, Kim JH. Dynamic framed slotted ALOHA algorithms using fast tag estimation method for RFID system. In *Proceedings of the 3rd IEEE Consumer Communications and Networking Conference (CCNC'06),* Las Vegas, NV. IEEE: Piscataway, NJ, 2006; 768 –772.
13. Hush DR, Wood C. Analysis of tree algorithms for RFID arbitration. In *Proceedings of the 1998 IEEE International Symposium on Information Theory,* Cambridge, MA. IEEE: Piscataway, NJ, 1998; 107.
14. Myung J, Srivastava J, Shih T, Lee W. Tag-splitting: adaptive collision arbitration protocols for RFID tag identification. *IEEE Transactions on Parallel and Distributed Systems* 2007; **18**: 763 –775.
15. Bruno R, Conti M, Gregori E. Optimization of efficiency and energy consumption in p-persistent CSMA-based wireless LANs. *IEEE Transactions on Mobile Computing* 2002; **1**: 10 –31.
16. IEEE Standard for Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications. *P802.11*, Institute of Electrical and Electronics Engineers, 1997.
17. Bianchi G. Performance analysis of the IEEE 802.11 distributed coordination function. *IEEE Journal on Selected Areas in Communications* 2000; **18**: 535–547.
18. Lee W, Cho D. A new neighbor discovery scheme based on spatial correlation of wireless channel. In *Proceedings of the 69th IEEE Vehicular Technology Conference (VTC Spring 2009),* Barcelona. IEEE: Piscataway, NJ, 2009; 1–5.
19. Dutta P, Culler D. Practical asynchronous neighbor discovery and rendezvous for mobile sensing applications. In *Proceedings of the 6th ACM Conference on Embedded Networked Sensor Systems (SenSys 2008),* Raleigh, NC. ACM: New York, NY, 2008; 71– 84.

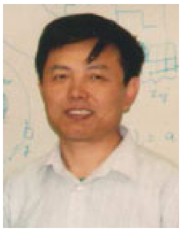
AUTHORS' BIOGRAPHIES

Kyunghwi Kim is a PhD candidate in Computer Science and Engineering from Korea University, Seoul, Korea. He received his BS and MS degrees in Computer Science and Engineering from Korea University, Seoul, Korea, in 2005 and 2007, respectively. His research interests include localization and mobility management in wireless sensor networks and fairness scheduling in wireless mesh networks.

Heejun Roh received his BS degree with honors in Computer Science and Engineering from Korea University, Seoul, Korea in 2009, double majoring in Mathematics. Currently, he is a second year MS student at Korea University. His research interests include mobile wireless communication protocols, wireless mesh network protocols and cooperative communication technology.

Wonjun Lee received his BS and MS degrees in Computer Engineering from Seoul National University, Seoul, Korea in 1989 and 1991, respectively. He received a PhD in Computer Science and Engineering from the University of Minnesota, MN, U.S.A., in 1999. In 2002, he joined the faculty of Korea University, Seoul, Korea, where he is currently a professor in the Department of Computer Science and Engineering. His research interests include mobile wireless communication protocols and Internet architecture technology. He has authored or co-authored over 110 papers in refereed international journals and conferences. He served as a transmission probability control member for the Institute of Electrical and Electronics Engineers International Conference on Computer Communications 2008–2012, Association for Computing Machinery Mobile Ad Hoc Networking and Computing 2008–2009 and over 100 international conferences. He is a senior member of the Institute of Electrical and Electronics Engineers.

Sinjae Lee received his BS, MS and DSc degrees in Computer Science from Towson University, MD, USA in 1999, 2001 and 2008, respectively. He is a research professor in Computer Science and Engineering Department at Korea University, Seoul, Korea. His research interests include mobile wireless network, peer-to-peer reputation systems, network simulation and network security.



Ding-Zhu Du is a Professor in the Department of Computer Science at the University of Texas at Dallas, a World Class University Professor in the World Class University Future Network Optimization Technology Research Center at Korea University, Seoul, Korea, and the Dean of Science at Xi'an Jiaotong University, Xi'an, China. He received a PhD in Mathematics from the University of California, Santa Barbara, in 1985. He co-authored over 250 papers in refereed journals, conferences and books. His research interests include theory of computational complexity and mathematical theory of optimization.