Computer Communications 75 (2016) 71–80

Contents lists available at ScienceDirect

Computer Communications

journal homepage: www.elsevier.com/locate/comcom

On multipacket reception based neighbor discovery in low-duty-cycle wireless sensor networks

Fan Wu [a,*], Guobao Sun [a,b], Guihai Chen [a]

[a] Department of Computer Science and Engineering, Shanghai Jiao Tong University, Shanghai 200240, China
[b] Department of Computer Science and Engineering, University of Minnesota, Twin Cities, Minneapolis, MN 55455, USA

ARTICLE INFO

Article history:
Received 7 September 2014
Revised 25 May 2015
Accepted 12 August 2015
Available online 18 August 2015

Keywords:
Wireless sensor network
Neighbor discovery
Multipacket reception

ABSTRACT

Neighbor Discovery (ND) plays an important role in the initialization phase of wireless sensor networks. In real deployments, sensor nodes may not always be awake due to limited power supply, which forms low-duty-cycle networks. Existing researches on the problem of ND in low-duty-cycle networks are all based on the assumption that a receiver can receive only one packet successfully at a time. k-Multipacket Reception (MPR) techniques (i.e., $k\,(k \geq 2)$ packets can be successfully received at a time) have shown their significance in improving packet transmission. However, how MPR can benefit the problem of ND is still unknown. In this paper, we are the first to discuss the problem of ND in low-duty-cycle networks with MPR. Specifically, we first present a novel ALOHA-like protocol, and show that the expected time to discover all $n-1$ neighbors is $O(\frac{n \log n \log\log n}{k})$ by reducing the problem to a generalized form of the classic *K Coupon Collector's Problem*. Second, we show that when there is a feedback mechanism to inform a node whether its transmission is successful or not, ND can be finished in time $O(\frac{n \log\log n}{k})$. Third, we point out that lacking of knowledge of n results in a factor of two slowdown in the two protocols proposed. We also discuss some extensions related to the protocol's design and different MPR models. Finally, we evaluate the ND protocols introduced in this paper, and compare their performance with the analysis results.

1. Introduction

Wireless Sensor Networks (WSNs) have drawn a lot of researchers' interests because of their wide range of applications. In many cases, sensor nodes are deployed without the support of pre-existing base infrastructures, and they need to form a network through their own cooperation. *Neighbor Discovery* (ND) is a family of protocols designed to find nodes' one-hop neighbors, and is the first step in the initialization of WSNs. The information acquired through neighbor discovery protocols is extremely useful for further operations such as media access and routing.

Existing protocols for ND can be classified into three categories: deterministic protocols [1], multi-user detection-based protocols [2–4], and randomized protocols [5–12]. Deterministic protocols usually use leaders to schedule all nodes' transmissions, and multi-user detection-based protocols identify neighbors by their pre-defined signatures. Compared with the first two categories, randomized protocols are more commonly used to conduct ND. In randomized protocols, the nodes broadcast discovery messages in randomly chosen time slots to reduce the possibility of the collision from the other nodes.

Usually the problem of ND is discussed in a synchronous system, e.g., birthday protocols [5]. In birthday protocols each node independently chooses to transmit during each slot with probability p and to receive with probability $1-p$. By reducing the analysis of birthday protocols to the classical *Coupon Collector's Problem*, Vasudevan et al. [7] discussed the time complexity of birthday protocols. Many subsequent protocols are based on birthday protocols [7–9,12]. For example, due to the development of Code Division Multiple Access (CDMA) and Multiple-Input and Multiple-Output (MIMO), several protocols adopt the fact that nodes can receive more than one packet simultaneously, i.e., Multipacket Reception (MPR), instead of the traditional assumption of Single Packet Reception (SPR) [8,12]. Fig. 1 gives an example about how the MPR technique can help to accelerate the process of ND.

Furthermore, we notice that many existing ND protocols are based on the assumption that nodes are always awake during the ND process. This is unrealistic in WSNs due to the limited power supply. In WSNs, nodes are typically working with a certain duty-cycle (transmitting, receiving, and dormancy) to reduce the energy consumption. A few works focus on this problem and analyze the ND in low duty cycle networks. You et al. [11] discussed the issue of ND process with

[*] Corresponding author. Tel.: +86-21-34206002.
E-mail addresses: fwu@cs.sjtu.edu.cn (F. Wu), gsun@cs.umn.edu (G. Sun), gchen@cs.sjtu.edu.cn (G. Chen).

http://dx.doi.org/10.1016/j.comcom.2015.08.009
0140-3664/© 2015 Elsevier B.V. All rights reserved.

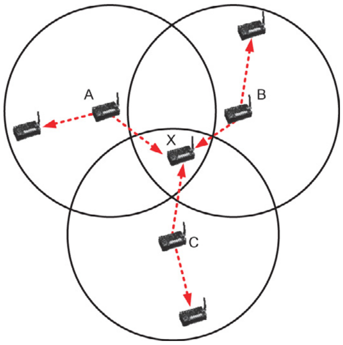

Fig. 1. An example of how the ND is conducted in MPR networks. Node *A*, *B*, and *C* are broadcasting their discovery messages simultaneously. Node *X* is in the coverage of all the three nodes. If they are in a SPR network, collision will occur at *X*. However if they are in a 3-MPR network, *X* will successfully receive three nodes' discovery messages simultaneously.

low-duty-cycle nodes and derived an upper bound on the expected time of ND under the SPR model. Jeon and Ephremides [13] discussed the issue of physical-layer signal processing to achieve MPR but the low-duty-cycle scenario was not covered.

Importing MPR technology into the process of neighbor discovery is of great benefit to wireless sensor nodes in terms of reducing the time needed to finish ND [8,12]. Since the neighbor discover process can last up to weeks [5], it is still unrealistic to use a protocol which keeps all nodes awake for weeks (batteries will still be used up quickly). Even if we omit this extreme case, it is still necessary to deploy a duty-cycled work manner for nodes. In some cases, all nodes cannot be deployed in one single batch. Hence, it is totally possible that nodes in the first batch already wasted a lot of energy before the last batch of nodes is deployed. On the other hand, using normal ND protocols by forcing all nodes to be awake in the initial stage is also not a realistic solution. The reason is that switching from duty-cycled manner to all-awake manner, and then switching from all-awake manner to duty-cycled manner can be very hard due to the overall coordination problem in a large network. (Similar issue has also been discussed in [11].) However, if we deploy MPR techniques into these nodes, the process can be significantly reduced while retaining battery life, since nodes are still operated in a low duty-cycled manner. There already appeared literatures discussing how to achieve MPR in sensor nodes at the signal process level (e.g., [13]). Hence, we think now it is time to study this topic in-depth.

The transition from SPR to MPR in low-duty-cycle WSNs is not trivial, because nodes act completely different from the SPR scenario. First, in traditional ALOHA-like protocols (e. g., birthday protocols), the optimal transmission probability can be easily determined to be $1/n$, where n is the clique size [5]. However, it is difficult to derive a closed form for the optimal transmission probability in k-MPR[1] networks. Second, previous researches with SPR model are all based on the assumption that once a node has transmitted its discovery message without collision, it will certainly be discovered by all the other nodes in a clique, which does not hold in low-duty-cycle WSNs. The reason is twofold. On one hand, it is almost impossible for all nodes to be awake at a certain time instant in low-duty-cycle networks because many nodes may be dormant. On the other hand, even if all nodes happen to be awake, it is still not enough for a node A to transmit its discovery message only once to let all other nodes find it, due to the reason that there may be more than one node, say $m(1 < m \leq k)$ nodes (including A), transmitting simultaneously. Since the radios on sensors nodes are half duplex, $n - m$ nodes can discover A successfully , while $m - 1$ nodes cannot, because they are transmitting.

In this paper, we study the problem of ND in low-duty-cycle WSNs with k-MPR radios, and conduct in-depth performance analysis on ALOHA-like ND protocols with various extensions. The contributions of this paper are listed as follows:

- First, to the best of our knowledge, we are the first to consider the problem of ND using MPR radios in low-duty-cycle WSNs. We show that MPR can significantly accelerate the ND process, and thus the duration of ND in low-duty-cycle networks can be tremendously shortened. We study the ALOHA-like protocol in k-MPR networks and prove that the expected time needed is $O(\frac{n \log n \log \log n}{k})$, where n is the clique size, by reducing the problem to a generalized form of *K Coupon Collector's Problem* [14].
- Furthermore, when a feedback mechanism is introduced into the system, we prove that it provides a $\log n$ improvement over the ALOHA-like protocol, i.e., the complexity can be reduced to $O(\frac{n \log \log n}{k})$.
- We extend our protocols to the case where the clique size n is unknown and show that it results in a factor of two slowdown.
- We discuss the performance of the protocol in an ideal MPR model, in which nodes can receive arbitrary packets simultaneously.
- In comparison with the normal *multi-antenna* MPR model, we also discuss how the protocol works in a *multi-channel* MPR model.

The rest of this paper is organized as follows. In Section 2, we present related works. In Section 3, we describe the model and analyze the performance of ALOHA-like protocol in low-duty-cycle WSNs. In Section 4, the case when a feedback mechanism is introduced into the system is discussed. In Section 5, we discuss some related issues. In Section 6, we validate the theoretical results by simulation. The paper concludes with our future work in Section 7.

2. Related works

Many works have focused on the problem of ND and various protocols have been proposed and analyzed to adapt to different situations and assumptions. Basically, protocols of ND can be classified into three classes: deterministic protocols [1], multi-user detection-based protocols [2–4], and randomized protocols [5–12]. Deterministic protocols usually need a leader, which is aware of the whole topology of the network and schedule the transmitting and receiving beforehand to total avoid collisions. This kind of scheduling costs a lot of time and it is hard to implement it in a large scale distributed system. The multi-user detection-based protocols need complicated signal processing techniques and require that each node keeps all other nodes' signal signatures, which is unrealistic in many scenarios. Compared with the previous two kinds of protocols, randomized protocols are widely deployed due to their effectiveness and low cost.

The milestone of the randomized protocols of ND is the Birthday Protocol proposed in [5] by McGlynn and Borbash, who consider the randomized strategy in a synchronous system to avoid collisions in a clique. In birthday protocol, each node transmits its discovery message with probability p and receives other nodes' messages with probability $1 - p$ in a slot. Furthermore, the authors proved that the optimal transmission probability $p = 1/n$, where n is the size of the clique.

Based on the birthday protocol, Vasudevan et al. [6] proposed a similar randomized strategy when directional antennas are used instead of omnidirectional antennas. However the authors only provided numerical results, instead of analyzing the expected time theoretically in this paper. Later in [7], the authors first theoretically analyzed the time upper bound of the birthday protocol by reducing the ND problem to the classical *Coupon Collector's Problem*. When there are n nodes in the clique in a synchronous system, the expected time needed to discover all nodes is given by neH_n where H_n is the nth Harmonic number. In [7], the authors also proposed methods to

[1] k-MPR means that a receiver can successfully receive at most $k(k \geq 2)$ packets simultaneously.

handle more realistic situations where n is unknown beforehand, the system is asynchronous [10] and a feedback mechanism is introduced into the system [9]. Basically, not knowing n beforehand and the asynchronous system leads to no more than a factor of two slowdown respectively, and there will be a $\ln n$ improvement if a feedback mechanism is brought in. In addition, the author also proposed a method to determine when to terminate the ND process when n is unknown.

Zeng et al. first extended the results of [5,7] to the k-MPR situation. In contrast to previous works that are all based on the assumption that there is a collision if two or more nodes transmit simultaneously in a clique, k-MPR allows at most $k\,(k \geq 2)$ nodes in a clique transmit simultaneously. The authors proved that the expected time needed to discover all nodes is $\Theta(n\ln n/k)$. Ideally, if $k \geq n$, the expected time is shortened to $\Theta(\ln n)$. Similarly, the lack of knowledge of n, the asynchronous system and the import of feedback mechanisms result in the same factors of slowdown or speedup as they are in [7]. However, in idealized MPR model, not knowing n leads to a factor of $\log_2 n$ slowdown.

You et al. [12] considered a different MPR model in comparison with [8]. In [12], there are k channels. At each slot each node can transmit on one of the k channels or receive on all channels simultaneously. As a result a node can receive at most k packets successfully if k nodes choose mutually exclusive channels to transmit their messages (note that there is only one channel in [8].). The authors got the same time complexity $\Theta\ (n\ln\ \ n/k)$.

Sun et al. [15,16] proposed a protocol based on ALOHA-like protocols, but significantly mitigated the vital drawback that lies in traditional ALOHA-like protocols. In traditional ALOHA-like protocols, nodes independently choose how to act in a slot randomly, and the optimal value of transmission probability is proven to be $1/n$ where n is the size of the clique. However this leads to extremely low channel utility, and the waste of time slots. By introducing a pre-handshaking strategy, [15,16] tremendously improved the channel utility, and increased the successful probability in a time slot. In [17], they proposed the time complexity analysis for ALOHA-like protocols in low-duty-cycle wireless networks with MPR.

There are also literatures focusing on ND in mobile scenarios ([18,19], etc.). In this scenario, deterministic methods are most commonly used, and they will ensure that two encountering nodes will discover each other in a given time upper bound [18–21]. Usually, mobile devices have to operate in a duty-cycled manner to preserve energy, but since they are more targeted at the case when there are only two devices, we still need to investigate issues when many nodes are simultaneously densely deployed in an area.

Recently many works have focused on the proper ND protocols for WSNs. Many sensor nodes work in a duty cycle because of the shortage of power supply. Hence, the protocols which assume that a node has been discovered by all other nodes if it has transmitted successfully only once no longer work in the low-duty-cycle WSNs, since some nodes may be dormant and cannot receive anything at some time instants.

You et al. [11] extended the discussion of [7] to the low-duty-cycle case. In this paper, the authors introduced a kind of random duty cycle schema, in which nodes can choose to be dormant with a certain probability, which forms the low-duty-cycle sensor network. By reducing the problem to the *K Coupon Collector's Problem* [22], the authors proved that when the duty cycle is 1/2, the upper bound is $ne(\log_2 n + (3\log_2 n - 1)\log_2\log_2 n + c)$ with a constant c.

Besides these works which are aimed at accelerating the process of ND, there are also many other researches that discuss other problems about ND. For example, the authors in [23,24] discussed the formal definition of secure ND and defined the adversary actions. Energy consumption of ND has also been extensively analyzed and discussed in [25] and other literatures. Due to the reason that we focus on accelerating the ND process, we will not introduce them in detail.

3. ALOHA-like protocol

In this section, we present the ALOHA-like protocol and analyze its performance based on the model and assumptions described in Section 3.1. Recall that there are n nodes and they are in a clique. Furthermore, the case where n is unknown to nodes is discussed in Section 3.4. Since the main idea of the adaptive ALOHA-like protocol is still based on the ALOHA-like protocol, we will introduce it in next section.

3.1. Network model and assumptions

First we present our network model and assumptions under which we discuss the issue of ND. They are widely adopted by many previous works [5,7,8,11,12]. These assumptions are as follows:

- Each node has a locally unique identifier (e.g. the MAC address, the location).
- Time is identically slotted and nodes are synchronized on slot boundaries.
- n nodes are deployed in a clique. Every two nodes in the clique of size n can communicate with each other, i.e., all nodes are within other nodes' communication range. For simplicity, we label them as $\{1, 2, \ldots, n\}$.
- We assume that n is known to all nodes in the clique. This will be relaxed in later sections.
- All nodes have the same transmission range and use omnidirectional antennas.
- Nodes are in a k-MPR ($k \geq 2$) network, which indicates that there is a collision in the clique if and only if there are more than k nodes transmitting simultaneously in a slot. This capability can be achieved by MIMO or CDMA. To simplify our discussion, we neglect some real implementation issues of MPR, and assume that all sending nodes' packets can be successfully decoded at receivers' side as long as the number of transmitters is no more than k. This is also adopted by other literatures (e.g., [8]). We will discuss more realistic scenarios in later sections.
- Nodes are half duplex, i.e., nodes can either transmit or listen in a slot but not both at the same time.

3.2. Protocol description

In the ALOHA-like protocol, we assume that each node independently chooses how to act in a time slot. The detailed protocol is shown in Fig. 2.

In total, there are three states for a node in each slot: transmitting, receiving, and dormancy. The corresponding probabilities are $p_w p_t$, $p_w p_l$ and $1 - p_w$, respectively.

If we assume a node transmits with probability p in a slot, we can also know that it is in receiving state with probability $p_w - p$ and is dormant with probability $1 - p_w$.

1. Each node chooses to be awake with probability p_w and to be dormant with probability $1 - p_w$. Dormant nodes do not transmit or receive.
2. Awake nodes will independently choose to transmit with probability p_t and receive with probability p_l.

Fig. 2. ALOHA-like protocol.

We can determine the probability of a *successful slot*, i.e., there is at least one node transmitting in the slot, and no collision occurs. A lemma is as follows:

Lemma 1. *For a given slot in k-MPR networks, the probability that no collision occurs in the slot is given by*

$$p_s = \sum_{i=1}^{k} \binom{n}{i} p^i (1-p)^{n-i}. \tag{1}$$

Proof. In k-MPR networks, a collision occurs if and only if more than k nodes transmit simultaneously. Since all nodes independently choose to transmit with probability p, the probability that i nodes transmit in a slot is

$$p_i = \binom{n}{i} p^i (1-p)^{n-i}.$$

Therefore, the lemma holds. □

3.3. Performance analysis

In this subsection, before the analysis of the ALOHA-like protocol in k-MPR low-duty-cycle WSNs, we first introduce two lemmas. One is about the generalized form of *Coupon Collector's Problem*, and the other is about the estimation of p_s.

Lemma 2. *There are n different coupons and a collector randomly chooses a(a < n) distinct coupons (with replacement) in a run. Denote the number of expected runs to get all n coupons picked out and each coupon is picked out at least m times as $D^a_{m,n}$. Then*

$$D^a_{m,n} = O\left(\frac{nm \log\log n + n \log n}{a}\right). \tag{2}$$

Due to the space limitation we omit the analysis and proof of this theorem and it can be found in [14]. Note that the problem definition is a little bit different from the definition here. In [14], only one coupon out of those a coupons is kept. However, this does not affect this result because if we keep all a coupons in a round, the result should be no worse than the result presented in [14].

In the MPR network, it is difficult to theoretically give an optimal value for the transmission probability p, here we use a probability that helps us to establish a firm bound in later sections, and it will not affect the asymptotic results.

Lemma 3. *Let $p = \frac{k-1}{n}$ and the following inequality holds:*

$$p_s > \frac{1}{2} - \frac{1}{e}.$$

Proof. Define X_i^t as a binary indicator random variable of the event "node i transmits in slot t". Then the expression of p_s in time slot t can be rewritten as follows

$$p_s = Pr\left[1 \le \sum_i X_i^t \le k\right].$$

It is obvious to see that $\sum_i X_i^t$ follows a Binomial distribution, and its mean is $np = k-1$. Due to the reason that the mean and the median are at most $\ln 2$ apart [26], the median is in $[k-1-\ln 2, k-1+\ln 2]$. Since $k-1+\ln 2 < k$, we can see that

$$Pr\left[\sum_i X_i^t \le k\right] > \frac{1}{2}.$$

Hence,

$$p_s = Pr\left[\sum_i X_i^t \le k\right] - Pr\left[\sum_i X_i^t = 0\right]$$

$$> \frac{1}{2} - (1-p)^n > \frac{1}{2} - \left(1 - \frac{1}{n}\right)^n$$
$$> \frac{1}{2} - \frac{1}{e}, \tag{3}$$

where the last inequality comes from the known inequality $(1-\frac{1}{n})^n < \frac{1}{e}$ for $\forall n \in \mathbb{N}$. □

Recall that p is the probability for each node to transmit, and p_s is the probability that no collisions occur in this time slot (when there is at least one transmitter).

In Section 1, we have mentioned that it is not enough for a node to transmit successfully only once. The following theorem shows how many successful transmissions are needed for a node.

Theorem 1. *If a node A transmits its discovery message $3\log_L n$ times without collisions where $L = 1/(1-p_w+p)$, then A is discovered by its $n-1$ neighbors with high probability, asymptotically.*

Proof. Since in these $3\log_L n$ slots no collision occurs, a node B will discover A if and only if B is in receiving state in at least one of these $3\log_L n$ slots. Hence, the probability that B does not discover A successfully is given by

$$p_B = (1-p_w+p)^{3\log_L n}.$$

Hence,

$$p_B = \left(\frac{1}{L}\right)^{3\log_L n} = \frac{1}{n^3}.$$

We denote the event "there is at least one node that does not discover A after $3\log_L n$ slots" as ϵ, and we can determine $P(\epsilon)$ according to *Union Bound*

$$P(\epsilon) \le np_B = \frac{1}{n^2}.$$

We can see that $P(\epsilon) \to 0$ as $n \to \infty$. □

In fact, we can see that $(1+\alpha)\log_L n (\alpha > 0)$ times is enough for a node. To ensure the high discovery probability, here we choose $\alpha = 2$. Also, since n is always pre-determined, $3\log_L n$ should be more regarded as a constant, instead of a variable.

We are now ready to analyze the performance of the ALOHA-like protocol and point out its time complexity.

Theorem 2. *Let T be the time needed to discover all n nodes by using the ALOHA-like protocol. The expected value of T is given by*

$$E[T] = O\left(\frac{n \log n \log\log n}{k}\right). \tag{4}$$

Proof. In each time slot, there are $O(k)$ nodes intending to transmit and each node need to transmit $O(\log n)$ times without collisions to let all other nodes discover it. According to Lemma 2, we know that if there is no collision in every slot, the expected time needed is given by

$$W = O\left(\frac{n \log n \log\log n}{k}\right).$$

Note that T is a Pascal random variable with the parameter p_s and W, therefore we get

$$E[T] = \frac{W}{p_s} < \frac{W}{\frac{1}{2} - \frac{1}{e}} = O\left(\frac{n \log n \log\log n}{k}\right).$$

□

We note that the above analysis results on the time complexity of the ALOHA-like protocol is a generalization of previous works (i.e., [5,7,8,11]). By using lemmas and theorems in this paper, we can derive many results on the performance of the ALOHA-like protocol in various scenarios.

- In [5,7], SPR model is used and nodes are always awake, which implies that in each run we can only choose at most one coupon, and it only needs to be picked out once. By setting $k = 1$ and $m = 1$, we get $O(n \log n)$ which is the time complexity of birthday protocols [5,7].
- In [8], MPR model is introduced and nodes are always awake. Hence, in each run we can choose at most k coupons. In [8], Zeng et al. pointed out that it is enough for a node to successfully transmit three times in order to make all the other nodes discover it. Therefore by setting $m = 3$, we get $O(\frac{n \log n}{k})$, which is proven in [8,13].
- In [11], SPR model is adopted and nodes have a duty cycle of $p_w = 1/2$. In each run we can only choose at most one coupon. In [11], You et al. proved that $3 \log n$ successful transmissions are enough for a node to make all the other nodes discover it. By setting $k = 1$ and $m = 3 \log n$, we get $O\left(n \log n \log\log n\right)$.

3.4. Unknown number of neighbors

In previous discussion, we all assume that the clique size n is known to all nodes in the clique. In this subsection, we will discuss how the protocol works if n is unknown to all nodes.

We use a standard method [7,8,12] to handle this situation. The basic idea is to divide the whole ND process into phases. In phase i, each node runs the protocol as if there are 2^i neighbors to discover. This phase lasts $O(\frac{2^i \log 2^i \log\log 2^i}{k})$ slots. Consequently, in the $\lceil \log_2 n \rceil$th phase, each node will run the protocol as if there are n neighbors and this phase will last $O(\frac{n \log n \log\log n}{k})$ slots. This is just what we have derived before and the ND process can be terminated after this phase.

Now the expected time needed is

$$\sum_{i=1}^{\lceil \log_2 n \rceil} m \frac{2^i \log 2^i \log\log 2^i}{k}$$
$$< \sum_{i=1}^{\lceil \log_2 n \rceil} m \frac{2^i \log n \log\log n}{k}$$
$$= m \frac{2n \log n \log\log n}{k},$$

where m stands for the constant of the time complexity. Hence, we observe that the lack of knowledge of n results in a factor of two slowdown.

4. Adaptive ALOHA-like protocol

In this section, we discuss the problem of ND in low-duty-cycle k-MPR WSNs when a transmitting node knows whether its transmission is successful or not. The feasibility and design of such kind of feedback mechanisms has been discussed in [7,9].

In contrast to Section 3, we divide a time slot into two sub-slots. Nodes independently choose to transmit or receive in the first sub-slot. For a receiving node it will check if there is a collision in the first sub-slot, i.e., more than k nodes are simultaneously transmitting [9], and broadcast a signal in the second sub-slot if a collision occurs. As a transmitting node, it keeps listening in the second sub-slot. If it hears a signal it knows that its transmission in this slot was failed; otherwise, it knows that the transmission was successful. We note that when all nodes are transmitting in the same slot, all of them will think their transmissions are successful, because they cannot receive feedback signals, which nullifies the feedback mechanism. Fortunately, the probability of this event is $(\frac{k-1}{n})^n$, which tends to be 0 as $n \to \infty$. Therefore it is reasonable to ignore it.

The main idea of our design with the feedback mechanism follows from [8]. Fig. 3 gives a complete description of the adaptive ALOHA-like protocol.

We refer to a node that only receives and sleeps as *passive* node, otherwise *active* node.

1. At the beginning of the ND process all nodes are set to be active.
2. Time is divided into phases. In phase i there are n_i nodes to be discovered; p is set to be $(k-1)/n_i$ and this phase lasts $W_i = \Theta\left(\frac{n_i \log\log n_i}{k}\right)$ slots.
3. At the end of a phase, all nodes that have successfully transmitted their discovery messages at least $3 \log_L n_i$ times will turn passive.
4. This process continues until there are at most $n / \ln n$ nodes that are active. Then the ALOHA-like protocol without feedback mechanisms will be used as shown in Section 3.

Fig. 3. Adaptive ALOHA-like protocol.

We will prove that with a proper W_i, at least half of the nodes will turn passive at the end of each phase before $n/\ln n$ nodes are remained.

According to the scheme described above, the following inequality holds:

$$n_i \le \frac{n}{2^{i-1}},$$

where n_i is the number of active nodes in phase i. Hence, the total time needed is given by

$$\sum_{i=1}^{\log_2 \ln n} O\left(\frac{n \log\log \frac{n}{2^{i-1}}}{k 2^{i-1}}\right) + O\left(\frac{\frac{n}{\ln n} \log \frac{n}{\ln n} \log\log \frac{n}{\ln n}}{k}\right).$$

We can get from the equation above that the total time needed is $O(\frac{n \log\log n}{k})$, which has a factor of $\log n$ speedup compared with the one in Section 3.

Because every phase runs independently and identically except that n_i and p_i are different, we will consider only one phase and prove that in the first phase, with proper n and p, at least $n/2$ nodes will turn passive at the end of the phase.

Theorem 3 will use the first phase as an example, and all remaining phases will be exactly identical except different n_i and p_i values. In the first phase, the number of active nodes will be the total number of nodes in the clique, i.e. n. When applying the theorem to other phases, the corresponding n_i and p_i values should be used, instead of n and p.

Theorem 3. *Let S denote the set of nodes that turn from active to passive at the end of this phase. Let* $p = (k-1)/n$ *and* $W = \frac{\eta n \log\log n}{k-1}$ *where* η *satisfies the condition*

$$\frac{(\eta \log\log n - 3 \log_L n + \frac{1}{2})^2}{32\eta} > \frac{k}{k-1}.$$

Then for $\forall k \ge 2$

$$Pr\left[|S| < \frac{n}{2}\right] < e^{-\frac{n}{k \log\log n}}. \tag{5}$$

Proof. Define the variable Y_i^t for node i in slot t as follows:

$$Y_i^t = \begin{cases} X_i^t & \text{if } t \text{ is a successful slot,} \\ 0 & \text{otherwise.} \end{cases}$$

By Lemma 3 and the definition of Y_i^t we can get

$$Pr[Y_i^t = 1] = Pr[X_i^t = 1] \cdot Pr\left[\sum_i X_i^t \le k\right] \ge \frac{p}{2}.$$

Let $K_i^t = \min\{3\log_L n, \sum_{t'<t} Y_i^{t'}\}$ and $S^t = \{i | K_i^t = 3\log_L n\}$. We have

$$K^t \triangleq \sum_i K_i^t \le |S^t| \cdot (3\log_L n) + (n - |S^t|) \cdot (3\log_L n - 1).$$

If $|S^t| < n/2$ we get

$$K^t < n\left(3\log_L n - \frac{1}{2}\right).$$

Hence,

$$Pr\left[|S^W| < \frac{n}{2}\right] \le Pr\left[K^W < n\left(3\log_L n - \frac{1}{2}\right)\right] \tag{6}$$

We define $Z^t \triangleq K^t - K^{t-1}$ if $|S^{t-1}| \le n/2$. (If $|S^{t-1}| > n/2$ the conclusion holds obviously.) Note that $Z \triangleq \sum_{t=1}^{W} Z^t = K^W$. Therefore according to Eq. (6),

$$Pr\left[|S^W| < \frac{n}{2}\right] \le Pr\left[Z < n\left(3\log_L n - \frac{1}{2}\right)\right].$$

On the other hand, we have

$$\begin{aligned} E[Z^t] &= \sum_{i \notin S^{t-1}} Y_i^t \ge \left(n - |S^{t-1}|\right) \cdot \frac{p}{2} \\ &\ge \frac{n}{2} \cdot \frac{k-1}{2n} = \frac{k-1}{4}. \end{aligned}$$

Define $\widetilde{Z}^t = E[Z | X_i^m, m \le t]$ and the following equation holds:

$$E[\widetilde{Z}^t | \widetilde{Z}^{t-1}] = \widetilde{Z}^{t-1},$$

which indicates that the sequence forms a martingale. Then we apply *Azuma's inequality* [27] to $\widetilde{Z}^t$ and we have

$$\begin{aligned} Pr\left[|S^W| < \frac{n}{2}\right] &\le Pr\left[Z < n\left(3\log_L n - \frac{1}{2}\right)\right] \\ &= Pr\left[\widetilde{Z}^W < n\left(3\log_L n - \frac{1}{2}\right)\right]. \end{aligned}$$

Since $E[\widetilde{Z}^W] = E[Z] \ge \frac{(k-1)W}{4} = \frac{\eta n \log\log n}{4}$, we get

$$\begin{aligned} &Pr\left[\widetilde{Z}^W < n\left(3\log_L n - \frac{1}{2}\right)\right] \\ &\quad \le Pr\left[\widetilde{Z}^W < E\left[\widetilde{Z}^W\right] - \frac{nM}{4}\right] \\ &\quad \le \exp\left(-\frac{n^2 M^2 (k-1)}{32\eta k^2 n \log\log n}\right), \end{aligned}$$

where $M = \eta \log\log n - 3\log_L n + \frac{1}{2}$.

Taking the condition given in the theorem we can get

$$Pr\left[|S^W| < \frac{n}{2}\right] < e^{-\frac{n}{k\log\log n}}.$$

□

Then by using the similar insight from [12], we can prove that our algorithm can give the correct result with high probability, like all other randomized protocols.

Theorem 4. *The adaptive ALOHA-like protocol can make all nodes discovered by all others with high probability, i.e., $P \to 1$ as $n \to \infty$.*

Proof. According to Theorem 3, at least half of current active nodes will turn into passive modes with high probability in each stage, until there are $\frac{n}{\ln n}$ active nodes remaining. As a result, the probability that the result is correct is given by

$$P = \prod_{i=1}^{m} \left(1 - e^{-\frac{n_i}{k\log\log n_i}}\right),$$

where $m \le \log\ln n$. Since $n_i \ge \frac{n}{\ln n}$ holds for all phases i, we can then get

$$P \ge \left(1 - e^{-\frac{n}{k\ln n\log\log n}}\right)^{\log\ln n}.$$

According to this equation, we can get

$$\log P \ge \log\ln n \log\left(1 - e^{-\frac{n}{k\ln n\log\log n}}\right).$$

Because

$$\lim_{n\to\infty} \log\ln n \log\left(1 - e^{-\frac{n}{k\ln n\log\log n}}\right) = 0,$$

we can know that $P \to 1$. □

We note that previous mentioned strategy to handle the case when n is unknown in Section 3.4 can also be used in the protocol with a feedback mechanism. Similarly we can observe a factor of two slowdown in this case.

5. Discussion

In this section, we discuss some issues related to the neighbor discovery in low-duty-cycle wireless sensor networks, and their corresponding analysis results.

5.1. Idealized MPR

In our previous discussion, we assume a k-MPR model where $k < n$. However, it is of importance for us to explore the scenario when arbitrary packets can be successfully decoded simultaneously. Although it seems unrealistic, it is of practical importance in scenarios when $k \ge n$, i.e., the number of nodes is less than or equal k.

To analyze the performance of ALOHA-like protocol when $k \ge n$, we use a link-centric method to analyze the probability that one node discovers another node in a time slot. Consider two nodes i and j, the probability that i is discovered by j in a slot is given by

$$p_s = p_w p_t \cdot p_w(1 - p_t) = p_w^2 p_t (1 - p_t),$$

where $p_w p_t$ is the probability that i transmits, and $p_w(1 - p_t)$ is the probability that j receives. Since $k \ge n$ now, we can just simply ignore how other nodes act in this expression.

In normal settings, because p_w is a constant, we can trivially get the value of p_t to be 1/2 when maximizing p_s. As a result,

$$p_s = \frac{1}{4} p_w^2.$$

Then we can get the conclusion that the time complexity of the ALOHA-like protocol is $\Theta(\ln n)$, by using the analysis method in [8]. We will also evaluate this result in Section 6.

The method in Section 3.4 for handling the case when n is unknown can still be utilized here. After the $\lceil \log_2 n \rceil$th phase, all nodes can be discovered with high probability, and the total time is given by

$$\sum_{i=1}^{\lceil \log_2 n \rceil} c \ln n = \Theta(\log_2 n \ln n).$$

Although normally lacking knowledge of n leads to a factor of two slowdown, here we can see that under idealized MPR model, a factor of $\log_2 n$ slowdown is introduced.

5.2. Multi-channel MPR

In our previous protocols and theoretical analysis, we focus on *multi-antenna* MPR model, in which there is an assumption that for k-MPR, if the number of transmitters is less than or equal to k, these packets can all be successfully received by receivers. Another kind of MPR model is known as *multi-channel* MPR model [12,28]. In multi-channel MPR model, the bandwidth is divided into k channels. Each

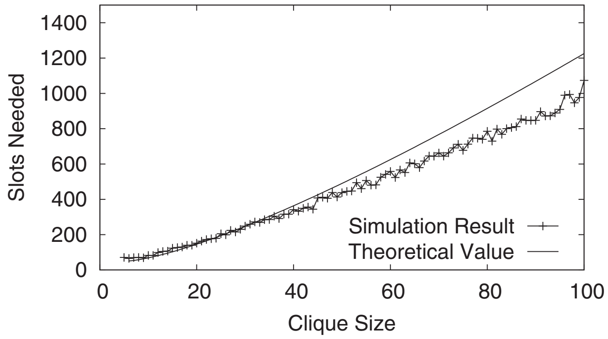

Fig. 4. Validation of the ALOHA-like protocol.

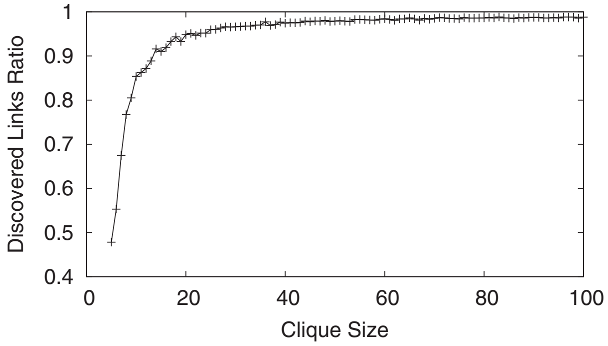

Fig. 5. The discovery ratio of the adaptive ALOHA-like protocol.

node can choose to transmit in one of these k channels, but can receive packets from all channels at the same time.

Under this MPR model assumption, the ALOHA-like neighbor discovery protocol can be designed as follows: in each time slot, nodes choose to be awake with probability p_w. If the node chooses to be awake, it will then choose to transmit in a certain channel with probability p_t. Since there are k available channels, the probability of receiving is then given by $p_w(1-kp_t)$.

To analyze its performance, we first analyze the probability of two nodes' discovering process. Again we consider two nodes i and j. In a slot, the probability that i is discovered by j is given by

$$p_s = k \cdot p_w p_t \cdot p_w(1-kp_t) \cdot (1-p_t)^{n-2}.$$

In this expression, $p_w p_t$ is the probability that i transmits, and $p_w(1-kp_t)$ is the probability that j receives. $(1-p_t)^{n-2}$ means that all the other $n-2$ nodes cannot choose the same channel to transmit. Recall that i and j have k different channels to choose from. Then we can use similar methods like [8,12] to get its time complexity, i.e. $O(\frac{n\log n}{k})$, by using *Union Bound*. The evaluation can be found in [12] and we will omit them here.

Here we point out that although it seems like there is a discrepancy because the time complexity of the ALOHA-like protocol in multi-antenna MPR is $O(\frac{n\log n\log\log n}{k})$, while the complexity in multi-channel MPR is $O(\frac{n\log n}{k})$, in fact the definitions of k in these two models are significantly different. In multi-antenna MPR model, as long as the number of simultaneous transmitters is not larger than k, there will be no collision. However, in multi-channel MPR model, there are k channels and nodes choose one of these channels to transmit. In this model, even if the number of simultaneous transmitters is larger than k, some packets can still be successfully received as long as there is only one transmitter in some channels.

6. Performance evaluation

In this section, we validate our theoretical results by simulations. In our simulations we assume that nodes are all in a clique, and the size of the clique n is known beforehand. The cases when n is unknown and nodes are in multi-hop networks are well discussed and verified in [7,8,11,12], thus we omit them. Each data plot in the figures stands for an average result over 20 runs for accuracy. The clique size is set to a range from 10 to 50, unless stated separately in the figure.

6.1. Validation of theoretical results

Figs. 4 and 5 are the simulation results for the ALOHA-like protocol and the adaptive ALOHA-like protocol, respectively. In both figures the parameter $p_w = 0.5$ and $k = 3$. The clique size ranges from 5 to 100. Fig. 4 shows the trend of the number of time slots needed to discover all nodes with increasing size of the clique. We can see that the simulation results well fit with the corresponding theoretical values. The deviation is due to the reason that the closed form of the generalized *K Coupon Collector's Problem*'s expected time is nontrivial to be derived, thus we can only give the asymptotic results, but it is still able to prove the correctness of our derivation. Since the choice of constant does not affect the trend of the theoretical figure, we use it to prove the correctness of our asymptotic derivation.

Fig. 5 provides a link-based view of the ND process. In the link-based view we regard the connection between any two nodes in the clique as a link, and it is easy to see there are $n(n-1)$ links in a clique of n nodes. To measure how many links have been discovered, we use the discovered links ratio, which is the number of discovered links divided by the number of total links. We present the ratio of discovered links in the given time with different sizes of cliques, where the given time is determined by the adaptive ALOHA-like protocol (We set $\eta = 5$ in this figure.). In this figure, the theoretical value should be constantly 1. We can see from the figure that the discovery ratio is very close to 1 when the clique size is relatively large. Nevertheless the ratio is not that acceptable when n is small. It is reasonable because our results are all asymptotic results and the results match our derivation well when n is large.

Note that for the ALOHA-like protocol and the adaptive ALOHA-like protocol, we are using two different metrics to evaluate them respectively. For the ALOHA-like protocol, we use the time slots needed to measure. For the adaptive ALOHA-like protocol, we use the discovered links ratio. The reason of using two metrics is that, in the ALOHA-like protocol scenario, the algorithm will keep running until all nodes are discovered (i.e., the discovered links ratio will always be 1). In the adaptive ALOHA-like protocol scenario, the algorithm is designed to have a fixed running time for given n value. Since it is a probabilistic algorithm, probably not all nodes can be discovered in the given running time. As a result, to measure how many nodes have been discovered, we decided to use discovered link ratio in the latter scenario. There are in total $n(n-1)$ links for n nodes in a clique, and the ratio is defined as the number of discovered links divided by $n(n-1)$. For the adaptive ALOHA-like protocol, it is not realistic to measure the precise time slots needed to achieve discovered links ratio 1, since the algorithm itself is designed to have a fixed running time for a given n value.

6.2. Different settings for ALOHA-like protocol

We now analyze the performance of ALOHA-like protocol when different duty cycles and k-MPR are deployed in the clique. In this simulation, when comparing different duty cycles we set $k = 3$. When comparing different k we set the duty cycle $p_w = 0.8$.

Fig. 6 shows the comparison among three different settings of duty cycles when $k = 3$. Fig. 8 shows the trend of the time slots needed with increasing duty cycles when $n = 50$. When the duty cycle increases, the total time needed to discover all nodes decreases. It is predictable because low duty cycle means many nodes may be

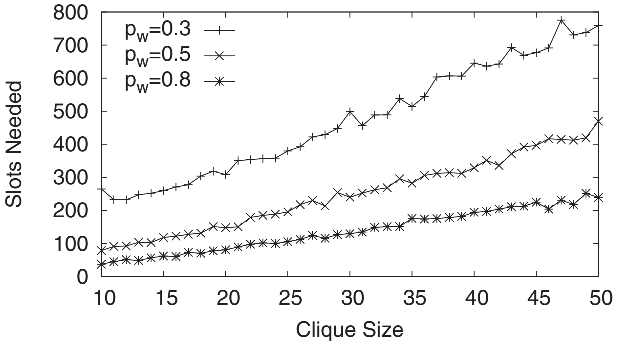

Fig. 6. Comparison of different duty cycles (ALOHA-like).

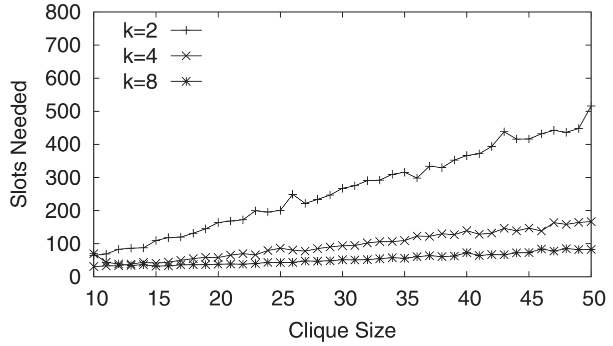

Fig. 7. Comparison of different k-MPR (ALOHA-like).

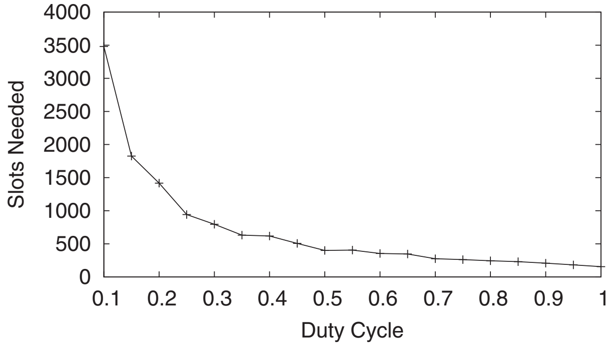

Fig. 8. Comparison of different duty cycles (ALOHA-like, $n = 50$).

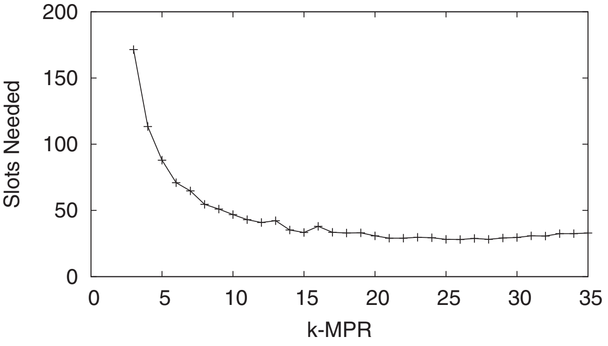

Fig. 9. Comparison of different k-MPR (ALOHA-like, $n = 50$).

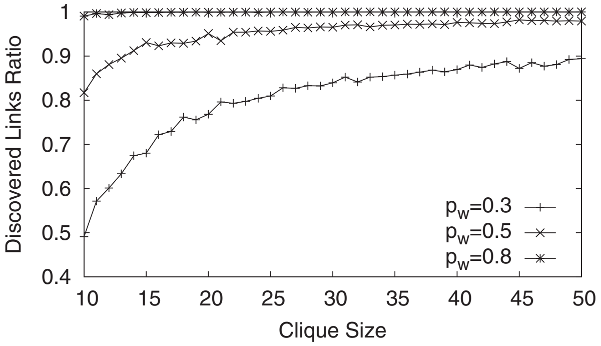

Fig. 10. Comparison of different duty cycles (adaptive ALOHA-like).

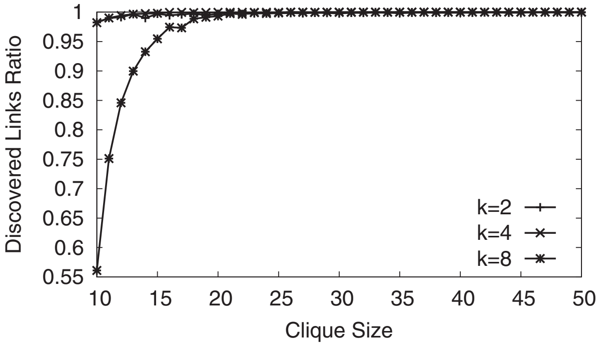

Fig. 11. Comparison of different k-MPR (adaptive ALOHA-like).

dormant at a single slot and the transmitting nodes' discovery messages cannot be received by most nodes, indicating that transmitters need more transmissions to ensure that the discovery messages have been received by all nodes at least once.

The comparison among different settings of k-MPR is shown in Fig. 7 (Note that now $p_w = 0.8$). Fig. 9 shows the trend of the time slots needed with increasing k when $n = 50$. When k increases, the total time needed decreases. Furthermore, when k doubles, the speed of ND is about twice faster. This coincides with our theoretical result and thus proves the correctness of our theorems.

On the other hand we must point out that the time needed is not always decreasing as k increases. Note that the beginning part of the case $k = 8$ is higher than the case $k = 4$. This is mainly because the transmitting probability is based on k and thus this probability is relatively high if k is large. When number of nodes is small, at a slot most nodes are transmitting and few nodes are receiving. Consequently, transmitting nodes need to spend more time letting all other nodes receive their discovery messages. We can observe this from Fig. 9 obviously. When $k > 15$, the total time needed fluctuates instead of keeping decreasing.

6.3. Different settings for adaptive ALOHA-like protocol

In this subsection, we begin to analyze the performance of the adaptive ALOHA-like protocol with different settings of duty cycles and k-MPR. Similarly, when comparing duty cycles we set $k = 3$. When comparing different k we set $p_w = 0.8$.

Fig. 10 shows the discovery ratio of three scenarios with different duty cycles. It is clear that the discovery ratio increases as the duty cycle p_w increases. Again we observe that when the size of the clique is small the ratio turns out to be relatively low because of our asymptotic analysis. When the duty cycle approaches 0.8, the adaptive ALOHA-like scheme ensures that almost all nodes can be discovered in the given time slots as mentioned in Section 6.1. In addition, we can see from the figure that as the size of clique rises, the discovery ratio also rises. This coincides with our aysmptotic analysis and shows that the validity of the adaptive ALOHA-like protocol.

Fig. 11 shows the discovery ratio of three scenarios with different settings of k. In this figure, when the clique size is 10 with $k = 8$, the discovery ratio is only less than 0.6, whereas the ratio is almost 1 when $k = 4$ and $k = 2$. This result may not seem to cater our intuition,

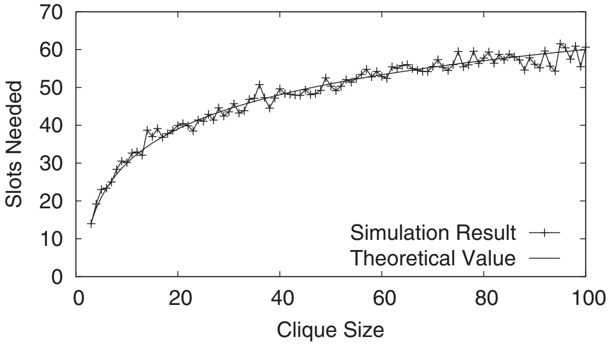

Fig. 12. Validation of the ALOHA-like protocol in idealized MPR.

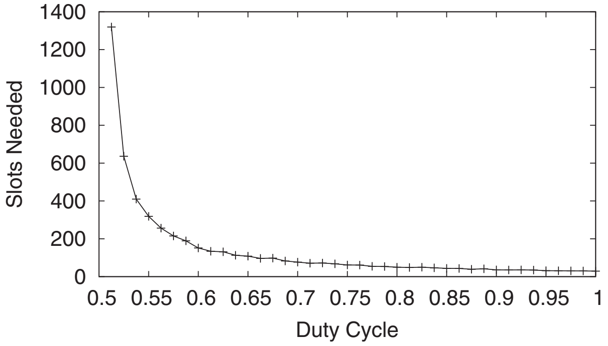

Fig. 14. Comparison of different duty cycles (idealized MPR).

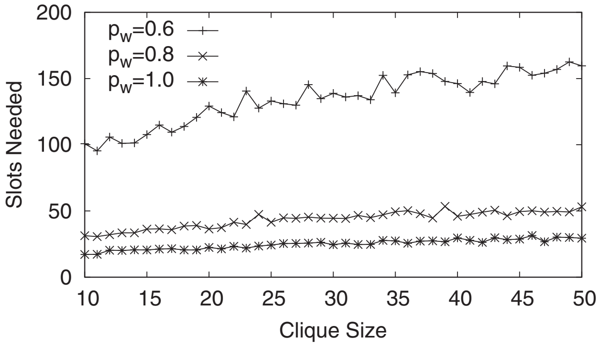

Fig. 13. Comparison of different duty cycles (idealized MPR).

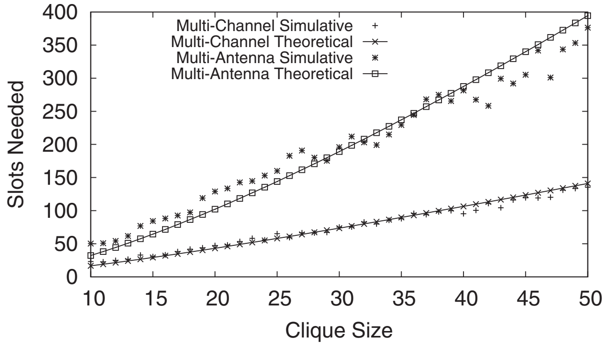

Fig. 15. Comparison between multi-antenna MPR and multi-channel MPR.

but it indeed can justify our assertion again: when the clique size is small, it is not a good idea to use large k. Too many transmitting nodes and too few receiving nodes will bring the side effect which prolongs the process of ND, because a node need to transmit a lot of times to make itself heard by all other nodes.

6.4. Idealized MPR

In this subsection, we present the simulation results on the performance of the ALOHA-like protocol with idealized MPR model, in which nodes have the ability to receive arbitrary number of packets simultaneously.

Fig. 12 shows the comparison between simulation results and the theoretical results. Here the duty cycle is set to be 0.8, and the theoretic curve is $c\ln n$, where c is a constant. The clique size ranges from 5 to 100. We can see from the figure that the simulation results closely fit the theoretic curve, which verifies the time complexity $\Theta(\ln n)$ we pointed out in Section 5.

Fig. 13 shows the time slots consumed under different duty cycle settings and different clique sizes. Again we can see that lower duty cycle requires more time slots needed to finish ND task. We can also observe that the results now still fit the $\Theta(\ln n)$ curve's trend. Fig. 14 shows the trend of time slots with different duty cycles, with the clique size $n = 50$. It is also observed that the time needed decreases with the increasing duty cycle.

6.5. Comparison between multi-antenna MPR and multi-channel MPR

In this subsection we will compare the performance between multi-antenna MPR and multi-channel MPR. As we have discussed in Section 5.2, the settings and assumptions are relatively different between these two. First, there is no duty-cycle setting involved in multi-channel MPR (i.e., nodes are always awake), while we have a duty-cycle setting in our multi-antenna MPR discussion. Second, the definitions of k-MPR are different. In multi-channel k-MPR, there are k channels, and each node can pick one channel out of these k channels. In this scenario, as long as for a given channel, there is only one node using it (even if the number of total transmitters is larger than k), there will be no collision in this channel. Hence, all receiving nodes will receive this node's information. However, in multi-antenna k-MPR, all transmitters can make their voices heard as long as there are less or equal to k transmitters are broadcasting simultaneously.

Based on these concerns, we will compare these two MPR techniques in our experiments by setting parameters carefully. To make these two MPR techniques comparable, we will set the duty-cycle to 1 both.

Fig. 15 shows the experiments for both multi-channel MPR and multi-antenna MPR. The figure shows the number of slots needed to finish ND increases as the clique size increases. The clique size ranges from 10 to 50, and k is set to 5. We have also shown the asymptotic results in the same figure. Specifically speaking, for multi-channel MPR it is $O(\frac{n\log n}{k})$, and for multi-antenna MPR it is $O(\frac{n\log n\log\log n}{k})$. In the figure, we can see that the experiment results are fitting the asymptotical theoretical results. Based on the same k settings, multi-channel MPR performs better than multi-antennfa MPR because the k definitions are different. In multi-channel MPR, if there are more than k transmitters at the same time, some packets may still be received if there are some channels that are only occupied by one transmitter each. However in multi-antenna MPR, if there are more than k transmitters at the same time, all transmitters will not be able to make their voices heard due to collisions.

7. Conclusion and future work

In this paper, we have analyzed the neighbor discovery problem in low-duty-cycle WSNs, and have derived the time complexity for two protocols respectively. For the ALOHA-like protocol, the expected time to finish ND is $O(\frac{n\log n\log\log n}{k})$ with k-MPR. Furthermore, if a feedback mechanism is introduced into the system, the expected time is $O(\frac{n \log \log n}{k})$. In addition, the lack of knowledge of n results in a factor of two slowdown in comparison with the n-known case. Discussions are presented to solve the issues in real implementations. Furthermore, we have presented another MPR model, i.e., multi-channel MPR model, and pointed out the time complexity of ALOHA-like protocols under this model. Our theoretical results are verified by extensive simulations.

In the future, we would like to evaluate these protocols by doing test-bed experiments. Also we would like to extend the protocols to some more realistic situations, e.g. nodes with different clocks, nodes with different duty cycles and more realistic radio models.

Acknowledgment

This work was supported in part by the State Key Development Program for Basic Research of China (973 project 2012CB316201), in part by China NSF grant 61422208, 61472252, 61272443 and 61133006, in part by CCF-Intel Young Faculty Researcher Program and CCF-Tencent Open Fund, in part by the Scientific Research Foundation for the Returned Overseas Chinese Scholars, and in part by Jiangsu Future Network Research Project No. BY2013095-1-10. The opinions, findings, conclusions, and recommendations expressed in this paper are those of the authors and do not necessarily reflect the views of the funding agencies or the government.

References

[1] A. Keshavarzian, E. Uysal-Biyikoglu, F. Herrmann, A. Manjeshwar, Energy-efficient link assessment in wireless sensor networks, in: Proceedings of INFOCOM 2004. Twenty-third AnnualJoint Conference of the IEEE Computer and Communications Societies, 3, IEEE, 2004, pp. 1751–1761.
[2] D. Angelosante, E. Biglieri, M. Lops, Neighbor discovery in wireless networks: a multiuser-detection approach, Phys. Commun. 3 (1) (2010) 28–36.
[3] J. Luo, D. Guo, Neighbor discovery in wireless ad hoc networks based on group testing, in: Proceedings of the 46th Annual Allerton Conference on Communication, Control, and Computing, 2008, IEEE, 2008, pp. 791–797.
[4] D. Guo, J. Luo, L. Zhang, K. Shen, Compressed neighbor discovery for wireless networks, CoRR, vol. abs/1012.1007 (2010).
[5] M.J. McGlynn, S.A. Borbash, 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, ACM, 2001, pp. 137–145.
[6] S. Vasudevan, J. Kurose, D. Towsley, On neighbor discovery in wireless networks with directional antennas, in: Proceedings of the 24th Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings IEEE, INFOCOM 2005., vol. 4, IEEE, 2005, pp. 2502–2512.
[7] S. Vasudevan, D. Towsley, D. Goeckel, R. Khalili, Neighbor discovery in wireless networks and the coupon collector's problem, in: Proceedings of the 15th annual international conference on Mobile computing and networking, ACM, 2009, pp. 181–192.
[8] W. Zeng, S. Vasudevan, X. Chen, B. Wang, A. Russell, W. Wei, Neighbor discovery in wireless networks with multipacket reception, in: Proceedings of the Twelfth ACM International Symposium on Mobile Ad Hoc Networking and Computing, ACM, 2011, p. 3.
[9] R. Khalili, D.L. Goeckel, D. Towsley, A. Swami, Neighbor discovery with reception status feedback to transmitters, in: Proceedings of IEEE, INFOCOM, 2010, IEEE, 2010, pp. 1–9.
[10] S.A. Borbash, A. Ephremides, M.J. McGlynn, An asynchronous neighbor discovery algorithm for wireless sensor networks, Ad Hoc Networks 5 (7) (2007) 998–1016.
[11] L. You, Z. Yuan, P. Yang, G. Chen, Aloha-like neighbor discovery in low-duty-cycle wireless sensor networks, in: Proceedings of Wireless Communications and Networking Conference (WCNC), 2011 IEEE, IEEE, 2011, pp. 749–754.
[12] L. You, X. Zhu, G. Chen, Neighbor discovery in peer-to-peer wireless networks with multi-channel mpr capability, in: Proceedings of IEEE International Conference on Communications (ICC), 2012, IEEE, 2012, pp. 4975–4979.
[13] J. Jeon, A. Ephremides, Neighbor discovery in a wireless sensor network: multipacket reception capability and physical-layer signal processing, in: Proceedings of the 48th Annual Allerton Conference on Communication, Control, and Computing (Allerton), 2010, IEEE, 2010, pp. 310–317.
[14] W. Xu, A.K. Tang, A generalized coupon collector problem, J. Appl. Probab. 48 (4) (2011) 1081–1094.
[15] G. Sun, F. Wu, X. Gao, G. Chen, Phed: pre-handshaking neighbor discovery protocols in full duplex wireless ad hoc networks, in: Proceedings of IEEE, Global Communications Conference (GLOBECOM), 2012, IEEE, 2012b, pp. 584–590.
[16] G. Sun, F. Wu, X. Gao, G. Chen, W. Wang, Time-efficient protocols for neighbor discovery in wireless ad hoc networks, IEEE Trans. Veh. Technol. 62 (6) (2013) 2780–2791.
[17] G. Sun, F. Wu, G. Chen, Neighbor discovery in low-duty-cycle wireless sensor networks with multipacket reception, in: Proceedings of the 18th International Conference on Parallel and Distributed Systems (ICPADS), 2012 IEEE, IEEE, 2012, pp. 490–497.
[18] P. Dutta, D. Culler, Practical asynchronous neighbor discovery and rendezvous for mobile sensing applications, in: Proceedings of the 6th ACM conference on Embedded network sensor systems, ACM, 2008, pp. 71–84.
[19] D. Zhang, T. He, Y. Liu, Y. Gu, F. Ye, R.K. Ganti, H. Lei, Acc: generic on-demand accelerations for neighbor discovery in mobile applications, in: Proceedings of the 10th ACM Conference on Embedded Network Sensor Systems, ACM, 2012, pp. 169–182.
[20] T. Meng, F. Wu, G. Chen, On designing neighbor discovery protocols: a code-based approach, in: Proceedings of IEEE, INFOCOM, 2014, IEEE, 2014, pp. 1689–1697.
[21] T. Meng, F. Wu, G. Chen, Code-based neighbor discovery protocols in mobile wireless networks, IEEE/ACM Trans. Netw. 2015 (2015).
[22] P. Erdos, A. Renyi, On a classical problem of probability theory, Magyar Tud. Akad. Mat. Kutató Int. Közl 6 (1-2) (1961) 215–220.
[23] M. Poturalski, P. Papadimitratos, J.-P. Hubaux, Secure neighbor discovery in wireless networks: formal investigation of possibility, in: Proceedings of the 2008 ACM symposium on Information, computer and communications security, ACM, 2008, pp. 189–200.
[24] R. Stoleru, H. Wu, H. Chenji, Secure neighbor discovery and wormhole localization in mobile ad hoc networks, Ad Hoc Netw. 10 (7) (2012) 1179–1190.
[25] M. Kohvakka, J. Suhonen, M. Kuorilehto, V. Kaseva, M. Hännikäinen, T.D. Hämäläinen, Energy-efficient neighbor discovery protocol for mobile wireless sensor networks, Ad Hoc Netw. 7 (1) (2009) 24–41.
[26] K. Hamza, The smallest uniform upper bound on the distance between the mean and the median of the binomial and Poisson distributions, Stat. Probab. Lett. 23 (1) (1995) 21–25.
[27] R. Motwani, Randomized Algorithms, Cambridge university press, 1995.
[28] X. Wu, S. Tavildar, S. Shakkottai, T. Richardson, J. Li, R. Laroia, A. Jovicic, Flashlinq: A synchronous distributed scheduler for peer-to-peer ad hoc networks, in: Proceedings of the 48th Annual Allerton Conference on Communication, Control, and Computing (Allerton), 2010, IEEE, 2010, pp. 514–521.