Neighbor Discovery in Wireless Networks with Multipacket Reception

Alexander Russell, Sudarshan Vasudevan, *Member, IEEE*, Bing Wang, *Member, IEEE*, Wei Zeng, *Member, IEEE*, Xian Chen, *Member, IEEE*, and Wei Wei, *Member, IEEE*

Abstract—Neighbor discovery is one of the first steps in configuring and managing a wireless network. Most existing studies on neighbor discovery assume a single-packet reception model where only a single packet can be received successfully at a receiver. In this paper, motivated by the increasing prevalence of multipacket reception (MPR) technologies such as CDMA and MIMO, we study neighbor discovery in MPR networks that allow packets from multiple simultaneous transmitters to be received successfully at a receiver. Starting with a clique of n nodes, we first analyze a simple Aloha-like algorithm and show that it takes $\Theta(\frac{n \ln n}{k})$ time to discover all neighbors with high probability when allowing up to k simultaneous transmissions. We then design two adaptive neighbor discovery algorithms that dynamically adjust the transmission probability for each node. We show that the adaptive algorithms yield a $\Theta(\ln n)$ improvement over the Aloha-like scheme for a clique with n nodes and are thus order-optimal. Finally, we analyze our algorithms in a general multi-hop network setting. We show an upper bound of $O(\frac{\Delta \ln n}{k})$ for the Aloha-like algorithm when the maximum node degree is Δ, which is at most a factor $\ln n$ worse than the optimal. In addition, when Δ is large, we show that the adaptive algorithms are order-optimal, i.e., have a running time of $O(\frac{\Delta}{k})$ which matches the lower bound for the problem.

Index Terms—Wireless networks, ad hoc networks, multipacket reception, network management, neighbor discovery, randomized algorithms

1 Introduction

Neighbor discovery is one of the first steps in configuring and managing a wireless network. The information obtained from neighbor discovery, viz. the set of nodes that a wireless node can directly communicate with, is needed to support basic functionalities such as medium access and routing. Furthermore, this information is needed by topology control and clustering algorithms to improve network performance [13], [20]. Due to its critical importance, neighbor discovery has received significant attention, and a number of studies have been devoted to this topic (e.g., [10], [18], [27], [28], [30]). Most studies, however, assume a single packet reception (SPR) model, i.e., a transmission is successful if and only if there are no other simultaneous transmissions.

In contrast to prior literature, we study neighbor discovery in multipacket reception (MPR) networks where packets from multiple simultaneous transmitters can be received successfully at a receiver. This is motivated by the increasing prevalence of MPR technologies in wireless networks. For instance, code division multiple access (CDMA) and multiple-input and multiple-output (MIMO), two widely used technologies, both support multipacket reception. Neighbor discovery in MPR networks differs fundamentally from that in SPR networks in the following manner. In a SPR network, a node is discovered by each of its neighbors if it is the only node that transmits at a given time instant; while in an MPR network, a node can transmit simultaneously with several other neighbors, and each of these nodes may be discovered simultaneously by the receiving nodes.

We focus on randomized algorithms throughout, as (i.) randomization is a powerful tool for avoiding centralized control, especially in settings with little a priori knowledge of network structure and (ii.) randomization offers extremely simple and efficient algorithms for homogeneous devices to carry out fundamental tasks like symmetry breaking.

We first consider clique topologies where all the nodes are the neighbors of each other and, subsequently, generalize our algorithms and analysis to the multi-hop network setting. For each algorithm presented in this paper, we analyze its performance in terms of *neighbor discovery time*, i.e., the time until each node discovers its respective neighbors. This is a critical performance metric since faster neighbor discovery leads to shorter delays to commence other network operations.

Our main contributions are as follows:

- We first consider a clique of n nodes in which node transmissions are synchronous and the number of nodes, n, is known. Specifically, we analyze an Aloha-like neighbor discovery algorithm, and show that the neighbor discovery time is $\Theta(\ln n)$ in an idealized MPR network that allows an arbitrary number of nodes to transmit simultaneously, and the neighbor discovery time is $\Theta(\frac{n \ln n}{k})$ when allowing up to k nodes to transmit simultaneously. We next propose two adaptive neighbor discovery algorithms, one being collision-detection based, and the other being

- *A. Russell, B. Wang, and W. Wei are with the Computer Science & Engineering Department at the University of Connecticut.*
- *S. Vasudevan is with Palo Alto Networks, Inc.*
- *W. Zeng is with Connecticut Transportation Safety Research Center.*
- *X. Chen is with Microsoft Corporation.*

Manuscript received 30 Oct. 2013; revised 19 Apr. 2014; accepted 20 Apr. 2014. Date of publication 29 June 2014; date of current version 5 June 2015. Recommended for acceptance by M. Kandemir.
For information on obtaining reprints of this article, please send e-mail to: reprints@ieee.org, and reference the Digital Object Identifier below.
Digital Object Identifier no. 10.1109/TPDS.2014.2321157

ID based. In both algorithms, a node becomes inactive once it is discovered by its neighbors, allowing the remaining active nodes to increase their transmission probability. We show that these adaptive algorithms are order-optimal, i.e., they achieve a running time of $\Theta(\frac{n}{k})$ and are thus a $\Theta(\ln n)$ factor faster than the Aloha-like algorithm.

- We extend our algorithms to the cases where the number of neighbors is not known beforehand or nodes transmit asynchronously, and show that these generalizations result in at most a constant or $\Theta(\ln n)$ factor slowdown in algorithm performance.
- For general network topologies, we first analyze the performance of the Aloha-like neighbor discovery algorithm, and show an upper bound of $O(\frac{\Delta \ln n}{k})$ in a network with n nodes and the maximum node degree is Δ. In addition, when Δ is large, we show that the adaptive algorithms are order-optimal, i.e., the neighbor discovery time for these algorithms is $O(\frac{\Delta}{k})$, which matches the lower bound for the problem.

The rest of the paper is organized as follows. Section 2 reviews related work. Section 3 presents the problem setting. Section 4 describes an Aloha-like neighbor discovery algorithm and its analysis. Section 5 describes two adaptive neighbor discovery algorithms, and shows that they improve upon the Aloha-like scheme by a $\ln n$ factor. Section 6 extends our results to scenarios where a node has no estimate of the number of its neighbors and node transmissions are asynchronous. Section 7 presents analytical results for general network topologies. Last, Section 8 concludes the paper and presents future directions.

2 Related Work

Our work is inspired by [27] that designs and analyzes several randomized neighbor discovery algorithms in SPR networks. As we will see in this paper, the design and analysis of neighbor discovery algorithms is substantially more challenging in the case of MPR networks as compared to the SPR networks studied in [27]. Furthermore, our study generalizes the study of [27], viz., our results are for the case where we allow up to k (≥ 1) simultaneous transmissions, and reduce to those in [27] by simply letting $k = 1$.

Several other studies develop randomized/deterministic neighbor discovery algorithms in SPR networks. McGlynn and Borbash [23] propose birthday-like randomized neighbor discovery algorithms that require synchronization among nodes. Tseng et al. [26] propose three power-saving protocols to schedule asynchronous node wake-up times in IEEE 802.11-based multi-hop ad hoc networks, and describe deterministic neighbor discovery schemes in each of the three protocols. Zheng et al. [33] provide a more systematic treatment of the asynchronous wakeup problem and propose a neighbor discovery protocol on top of the optimal wakeup schedule that they derive. Borbash et al. [5] propose asynchronous probabilistic neighbor discovery schemes for large-scale networks. Keshavarzian et al. [16] propose a deterministic neighbor discovery algorithm. More recently, Dutta and Culler [10] propose an asynchronous neighbor discovery and rendezvous protocol between a pair of low duty cycling nodes. Khalili et al. [17] propose feedback based neighbor discovery schemes that operate in fading channels. A recent study [6] considers the problem of continuous neighbor discovery where each node has partial knowledge of its neighborhood. Our study differs from each of the abovementioned works in that we consider neighbor discovery in MPR instead of SPR networks.

The algorithms proposed in [2], [3], [21], [22] use a multiuser-detection based approach for neighbor discovery. They require each node to possess a signature as well as know the signatures of all the other nodes in the network. Further, nodes are assumed to operate in a synchronous manner. When a node receives transmission from multiple neighbors, it determines which nodes are the transmitters based on the received signal (or energy) and the prior knowledge of the node signatures in the network. Although these studies allow multiple transmitters to transmit simultaneously, their focus is on using coherent/noncoherent detection [2], [3], [21] or group testing [22] to identify neighbors with a high detection ratio and low false positive ratio, and do not provide analytical insights on the time complexity of their schemes. In contrast, our study aims to understand the efficiency of different neighbor discovery algorithms by deriving analytical results on their time complexity. Further, from a practical viewpoint, our approach does not require node signatures and can operate in asynchronous systems.

There are numerous studies on neighbor discovery when nodes have directional antennas (e.g., [15], [25], [28], [30]). The focus in these works is on antenna scanning strategies for efficient neighbor discovery. There have been several recent proposals on neighbor discovery in cognitive radio networks (e.g., [4], [19]). They determine the set of neighbors for a node as well as the channels that can be used to communicate among neighbors. In contrast, we assume omni-directional antennas (or antenna arrays) and multipacket reception capabilities at each node.

3 Problem Setting

Consider a static network with n nodes indexed from 1 to n. Each node has a unique ID (e.g., its MAC address or geographic location). Each node embeds its ID in the messages it transmits to its neighbors. A node, x, is discovered by another node, y, if and only if y successfully receives a message from x. Each node has an omni-directional antenna (or an antenna array). The radio at each node is assumed to be half-duplex, i.e., a node can either transmit or receive packets, but not both at the same time. We assume that all nodes have multipacket reception capabilities. That is, a node can correctly receive packets from multiple transmitters simultaneously. This MPR capability can be provided through smart antenna array techniques such as MIMO, or coding techniques such as CDMA.

Similar to [11], [31], [32], we use a *reception matrix* to model the MPR capabilities of nodes. Specifically, let $\epsilon_{i,j}$ represent the probability that j packets are received successfully given that i packets are transmitted simultaneously. In our context, since at most n packets can be transmitted simultaneously at one point of time, the reception matrix is of dimension $n \times (n+1)$, and is represented as

$$\begin{pmatrix} \epsilon_{1,0} & \epsilon_{1,1} & & & \\ \epsilon_{2,0} & \epsilon_{2,1} & \epsilon_{2,2} & & \\ \vdots & \vdots & \vdots & & \\ \epsilon_{n,0} & \epsilon_{n,1} & \epsilon_{n,2} & \cdots & \epsilon_{n,n} \end{pmatrix},$$

where all elements in the upper triangle (i.e., $\epsilon_{i,j}, \forall j > i$) are zero by definition, and hence are omitted for clarity.

As an example, consider a CDMA system in which a packet is transmitted with a randomly generated code and is successfully received only if there are no more than two simultaneous transmissions. Then for $i = 1, 2$, we have $\epsilon_{i,i} = 1$ and $\epsilon_{i,j} = 0, \forall j \neq i$; for $i > 2$, we have $\epsilon_{i,0} = 1$ and $\epsilon_{i,j} = 0, \forall j > 0$ since no packets can be received successfully when there are more than two simultaneous transmissions.

In this paper, we consider an MPR model (henceforth, called the MPR-k model), in which up to k simultaneous packets can be decoded successfully at a receiver. The value of k is fixed and is known beforehand. In practice, it is determined by the number of orthogonal codes when using CDMA [31], or by the number of antennas in the case of MIMO systems. For a given k, $\epsilon_{i,i} = 1$, when $1 \leq i \leq k$ and $\epsilon_{i,j} = 0, \forall j \neq i$. When $i > k$, $\epsilon_{i,0} = 1$ and $\epsilon_{i,j} = 0, \forall j > 0$. The reception matrix is

$$\begin{pmatrix} 0 & 1 & & & & \\ 0 & 0 & 1 & & & \\ \vdots & \vdots & \vdots & & & \\ 0 & 0 & 0 & \cdots & 1 & \\ 1 & 0 & 0 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & & & \\ 1 & 0 & 0 & \cdots & 0 & 0 \end{pmatrix}.$$

Note that SPR is a special case of MPR-k (i.e., when $k = 1$). Another special case of MPR-k is when $k = n - 1$, referred to as the *idealized MPR model*. It is of practical interest in scenarios where the number of neighbors is close to the amount of diversity provided by MPR technologies (i.e., the number of orthogonal codes in CDMA or the number of antennas in MIMO).

We remark that the MPR-k model studied in this paper is a simple generalization of the well-known collision channel model studied in the case of SPR networks. In our model, collisions are the only source of packet errors. In practice, background noise contribute to errors as well (which can be modeled by letting $\epsilon_{i,0} > 0$, when $1 \leq i \leq k$ in our reception matrix). However, the main emphasis of our work is on providing useful insights for designing algorithms and understanding (to a first-order) their performance when deployed in the real world. We therefore choose a model which is analytically tractable while ignoring some real-world aspects of wireless channels such as channel noise and fading. We emphasize, however, that the correctness of the algorithms proposed in this paper is independent of the chosen model, and should therefore be applicable in real-world MPR settings.

4 Aloha-Like Neighbor Discovery Algorithm

In this section, we consider a simple Aloha-like neighbor discovery algorithm and analyze it for the case of a clique. We start with the simplifying assumptions that all nodes know the clique size, n. Furthermore, we assume that time is divided into slots, and that nodes are synchronized on slot boundaries. These assumptions will be relaxed later, in Sections 6.1 and 6.2.

The algorithm operates as follows. Each node transmits with probability p and listens with probability $1 - p$ in each slot, where p is a parameter, the optimal value of which will soon be determined. The transmission probability does not change over time. The case where the transmission probability is allowed to change will be studied in Section 5.

In the following, we first determine the optimal transmission probability and then present an asymptotic analysis of the Aloha-like neighbor discovery algorithm.

4.1 Optimal Transmission Probability

In an SPR wireless network, it is well-known that the optimal value of p is $1/n$. However, as we will see next, deriving the optimal value of p in the MPR case is non-trivial.

Consider two arbitrary nodes, x and y. Let p_s denote the probability that x is discovered by y in a given time slot. Thus, p_s is the probability that x transmits, y listens, and there are at most $k - 1$ other nodes transmitting. Therefore,

$$p_s = p(1-p)\sum_{i=0}^{k-1}\binom{n-2}{i}p^i(1-p)^{n-2-i}. \tag{1}$$

To minimize the time to discover all neighbors, we need to choose a p that maximizes p_s. Let p^* denote this optimal transmission probability, and let p_s^* denote the corresponding value of p_s. When $k = n - 1$ (i.e., the idealized MPR model), $p_s = p(1 - p)$. It is easy to see that $p^* = 1/2$ and $p_s^* = 1/4$. Deriving the optimal transmission probability is more challenging for general k. The following theorem provides analytical results for the optimal transmission probability.

Theorem 1. *Consider a clique of n nodes executing the Aloha-like algorithm, where n is known. Under the MPR-k model, the optimal transmission probability $p^* = \alpha k/n$, where $\alpha = 1$ for $k = 1$, and*

$$\alpha \in \begin{cases} (0.07, 6.38), & \text{if } k = 2, \\ (0.11, 4.17), & \text{if } k = 3, \\ (0.09, 3.55), & \text{if } k \geq 4. \end{cases}$$

Proof. For $k = 1$, i.e., the SPR case, it is clear $\alpha = 1$ since $p^* = 1/n$. We next consider the three cases $k = 2$, $k = 3$, and $k \geq 4$, respectively. For each case, we first derive a lower bound on the optimal p_s^*, and then derive the constants stated in the theorem. Note that since we are considering the MPR-k case, we have $n \geq k + 2$.

Let $B = B_1 + \cdots + B_{n-2}$, where $B_x = 1$ when node x transmits, and $B_x = 0$ otherwise. Then B follows a Binomial distribution, and (1) can be rewritten as

$$p_s = p(1-p)\Pr(B < k). \tag{2}$$

Note that

$$\begin{aligned} p_s^* &= p^*(1-p^*)\Pr(B < k) \\ &< p^*\Pr(B < k) = \frac{\alpha k}{n}\Pr(B < k). \end{aligned}$$

Using the tail bound for binomial distribution in [1, Theorem A.1.13], we obtain

$$\begin{aligned}\Pr(B < k) &< e^{-((n-2)p^*-k)^2/2(n-2)p^*}\\ &\approx e^{-(np^*-k)^2/2np^*}\\ &= e^{-\frac{(\alpha k-k)^2}{2\alpha k}} = e^{-\frac{k(\alpha-1)^2}{2\alpha}}.\end{aligned}$$

Hence,

$$p_s^* < \frac{\alpha k}{n} e^{-\frac{k(\alpha-1)^2}{2\alpha}}. \quad (3)$$

When $k = 2$, $p_s = p(1-p)^{n-1} + (n-2)p^2(1-p)^{n-2}$. Taking $p = 1/(n-2)$ yields

$$\begin{aligned}p_s &= \frac{1}{n-2}\left(1-\frac{1}{n-2}\right)^{n-2}\left(1-\frac{1}{n-2}+1\right)\\ &\geq \frac{1}{e^2(n-2)}\left(1-\frac{1}{n-2}+1\right)\\ &\geq \frac{1}{e^2(n-2)}.\end{aligned}$$

Since p_s^* denotes the optimal value of p_s, we have

$$p_s^* \geq \frac{1}{e^2(n-2)}. \quad (4)$$

Since $p_s^* < p^* = \alpha k/n = 2\alpha/n$, it follows from (4) that

$$\alpha > \frac{n}{n-2}\frac{1}{2e^2} > \frac{1}{2e^2} \approx 0.07.$$

On the other hand, a simple numerical calculation from (3) reveals that when $\alpha > 6.38$, $p_s^* < \frac{1}{e^2(n-2)}$, thus contradicting (4). Hence, $\alpha \in (0.07, 6.38)$ when $k = 2$.

When $k = 3$, $p_s = p(1-p)^{n-1} + (n-2)p^2(1-p)^{n-2} + \binom{n-2}{2}p^3(1-p)^{n-3}$. Taking $p = 1/(n-3)$ yields

$$\begin{aligned}p_s &= p(1-p)^{n-3}\left[(1-p)^2 + (n-2)p(1-p) + \frac{n-2}{2}p\right]\\ &\geq \frac{1}{e^2(n-3)}\left[(1-p)^2 + (n-2)p(1-p) + \frac{n-2}{2}p\right]\\ &= \frac{5n-18}{2e^2(n-3)^2}.\end{aligned}$$

Since p_s^* denotes the optimal value of p_s, we have

$$p_s^* \geq \frac{5n-18}{2e^2(n-3)^2}. \quad (5)$$

Since $p_s^* < p^* = \alpha k/n = 3\alpha/n$, it follows from (5) that

$$\alpha > \frac{n(5n-18)}{6e^2(n-3)^2} \geq \frac{5}{6e^2} \approx 0.11.$$

On the other hand, a simple numerical calculation from (3) reveals that when $\alpha > 4.17$, $p_s^* < \frac{5n-18}{2e^2(n-3)^2}$, thus contradicting (5). Hence, $\alpha \in (0.11, 4.17)$ when $k = 3$.

When $k \geq 4$. We first derive a lower bound on p_s^* by letting $p = (k-3)/(n-2)$. When $p = (k-3)/(n-2)$, the mean of the Binomial random variable, B, is $(n-2)p = k-3$. Since the mean and the median are at most $\ln 2$ apart [12], the median is in $[k-3-\ln 2, k-3+\ln 2]$. Since $k-1 > k-3+\ln 2$, we have

$$\Pr(X < k) \geq 1/2.$$

Since $n \geq k+2$, we have

$$p = \frac{k-3}{n-2} \leq \frac{k-3}{k} = 1 - \frac{3}{k}.$$

Therefore,

$$1 - p \geq \frac{3}{k}.$$

Summarizing the above, we obtain

$$p_s^* \geq \frac{k-3}{n-2}\cdot\frac{3}{k}\cdot\frac{1}{2} = \frac{3(k-3)}{2k(n-2)}. \quad (6)$$

Since $p_s^* < p^* = \alpha k/n$, it follows from (6) that

$$\alpha > \frac{3(k-3)n}{2k(n-2)k} \geq \frac{3(k-3)}{2k^2} \geq \frac{3}{32} \approx 0.09.$$

We next derive a condition on the value of α that leads to contradiction between (6) and (3), namely leads to

$$\frac{\alpha k}{n} e^{-\frac{k(\alpha-1)^2}{2\alpha}} \leq \frac{3(k-3)}{2k(n-2)}.$$

Simplifying the above, we are interested in an α that leads to

$$\alpha e^{-\frac{k(\alpha-1)^2}{2\alpha}} \leq \frac{3(k-3)n}{2k^2(n-2)}. \quad (7)$$

In (7), the left hand side is a decreasing function of k, and hence the maximum value is $\alpha e^{-\frac{4(\alpha-1)^2}{2\alpha}}$. On the other hand, as described earlier, the right hand side is larger than 0.09. When $\alpha > 3.55$, we have $\alpha e^{-\frac{4(\alpha-1)^2}{2\alpha}} < 0.09$, and hence a contradiction. Therefore, we need $\alpha < 3.55$. In summary, we have shown that $\alpha \in (0.09, 3.55)$ when $k \geq 4$. □

We remark that even though Theorem 1 does not yield the exact value of α (and consequently, that of p^*), it nonetheless yields a characterization of the optimal transmission probability p^*, which will be useful in the asymptotic analysis of the running time of the Aloha-like algorithm (see Section 4.2.2).

4.2 Asymptotic Analysis

Let T be a random variable that denotes the neighbor discovery time, i.e., the time until all n nodes have discovered their respective neighbors. We next present an asymptotic analysis of the neighbor discovery time in MPR networks. In particular, we show that $T = \Theta(\ln n)$ under the idealized MPR model, and $T = \Theta(\frac{n \ln n}{k})$ under the MPR-k model.

4.2.1 *Idealized MPR*

Recall that the idealized MPR model is a specific instance of the MPR-k model where $k = n-1$. Under this model, $p^* = 1/2$ and $p_s^* = 1/4$. We show that there exist two constants, $c_1 > c_2 > 0$, such that $c_2 \ln n < T < c_1 \ln n$ with high probability (abbreviated as w.h.p. henceforth), thus implying that $T = \Theta(\ln n)$.

We first show that there exists a constant $c_1 > 0$ such that $T < c_1 \ln n$ w.h.p. Let T_{xy} denote the time until node x discovers a given neighbor y:

$$\Pr(T_{xy} \le t) = 1 - \left(1 - p_s^*\right)^t = 1 - \left(\frac{3}{4}\right)^t .$$

That is,

$$\Pr(T_{xy} > t) = \left(1 - p_s^*\right)^t = \left(\frac{3}{4}\right)^t . \tag{8}$$

For $c_1 = -3/\ln(3/4)$, we have

$$\Pr(T_{xy} > c_1 \ln n) = \left(\frac{3}{4}\right)^{c_1 \ln n} = \frac{1}{n^3} . \tag{9}$$

Let T_x denote the time until x discovers all its neighbors. Then,

$$\Pr(T_x > t) = \Pr\Big(\max_y T_{xy} > t\Big) \le \sum_{y=1, y\neq x}^{n} \Pr(T_{xy} > t) . \tag{10}$$

Combining (9) and (10) yields

$$\Pr(T_x > c_1 \ln n) \le n \Pr(T_{xy} > c_1 \ln n) \le \frac{1}{n^2} . \tag{11}$$

Recalling that T is the time until all n nodes have discovered their respective neighbors, it follows that

$$\Pr(T > t) = \Pr\big(\max_x T_x > t\big) \le \sum_{x=1}^{n} \Pr(T_x > t) . \tag{12}$$

Combining (11) and (12), we obtain

$$\Pr(T > c_1 \ln n) \le n \Pr(T_x > c_1 \ln n) \le \frac{1}{n}, \tag{13}$$

where the right hand side approaches 0 as $n \to \infty$.

We now prove that we can find another positive constant c_2 such that $T > c_2 \ln n$ w.h.p. Note that to finish node discovery by time t, each node must transmit at least once by time t. Let T' be a random variable that denotes the time when all the nodes have transmitted at least once. It is easy to see that $T \ge T'$:

$$\Pr(T' \le t) = (1 - (1 - p^*)^t)^n = \left(1 - \left(\frac{1}{2}\right)^t\right)^n .$$

For $c_2 = 0.5/\ln 2$, we obtain

$$\left(1 - \left(\frac{1}{2}\right)^{c_2 \ln n}\right)^n = (1 - n^{-0.5})^n \le e^{-n^{0.5}},$$

where the right hand side approaches zero as $n \to \infty$. Therefore, $\Pr(T' \ge c_2 \ln n) \to 1$ as $n \to \infty$. Since $T \ge T'$, it follows that $\Pr(T \ge c_2 \ln n) \to 1$ as $n \to \infty$.

4.2.2 MPR-k

Recall from Theorem 1 that under the MPR-k model, the optimal transmission probability $p^* = \alpha k/n$, where α is a constant. We define $a \sim b$ if $\lim_{n\to\infty} a/b = 1$. Let $f(p)$ represent the binomial sum term in (1), i.e.,

$$f(p) = \sum_{i=0}^{k-1} \binom{n-2}{i} p^i (1-p)^{n-2-i} .$$

Using the Poisson approximation for binomial distribution [24] yields

$$f(p) \sim e^{-\lambda(p)} \sum_{i=0}^{k} \frac{\lambda(p)^i}{i!},$$

where $\lambda(p) = (n-2)p$. When $p = p^*$, $\lambda(p^*) = \frac{\alpha k(n-2)}{n} \sim \alpha k$ i.e., $\lambda(p^*)$ is independent of n. Therefore, for a given value of k, $f(p^*)$ can be considered to be a constant (i.e., independent of n). For convenience, let $\phi = f(p^*)$. Therefore,

$$p_s^* = \frac{\alpha k}{n}\left(1 - \frac{\alpha k}{n}\right)\phi .$$

Since $1 - \frac{\alpha k}{n} \sim 1$, we have

$$p_s^* \sim \frac{\gamma k}{n}, \tag{14}$$

where $\gamma = \alpha\phi$ is a constant. We next show that we can find constants $d_1 > d_2$, so that $\frac{d_2 n \ln n}{k} < T < \frac{d_1 n \ln n}{k}$ w.h.p., thus implying $T = \Theta(\frac{n \ln n}{k})$. From (8) and (14), we obtain

$$\Pr(T_{xy} > t) = \left(1 - \frac{\gamma k}{n}\right)^t .$$

Let $d_1 = 3/\gamma$. Then,

$$\Pr\left(T_{xy} > \frac{d_1 n \ln n}{k}\right) = \left(1 - \frac{\gamma k}{n}\right)^{\frac{d_1 n \ln n}{k}} \le \frac{1}{n^3} .$$

An analysis similar to the one for the idealized MPR case yields

$$\Pr\left(T > \frac{d_1 n \ln n}{k}\right) \le \frac{1}{n}, \tag{15}$$

where the right hand approaches 0 as $n \to \infty$.

We now prove that we can find a constant d_2 such that $T > \frac{d_2 n \ln n}{k}$ w.h.p. As in the idealized MPR setting, let T' represent the time when all the nodes have transmitted at least once. Thus, $T \ge T'$:

$$\Pr(T' \le t) = \left(1 - (1 - p^*)^t\right)^n = \left(1 - \left(1 - \frac{\alpha k}{n}\right)^t\right)^n .$$

Let $d_2 = 0.5/\alpha$. Then,

$$\left(1 - \left(1 - \frac{\alpha k}{n}\right)^{d_2 n \ln n/k}\right)^n = (1 - n^{-0.5})^n \le e^{-n^{0.5}},$$

where the right hand side approaches zero as $n \to \infty$. Therefore, $\Pr(T' \ge \frac{d_2 n \ln n}{k}) \to 1$ as $n \to \infty$. Since $T \ge T'$, we have $\Pr(T \ge \frac{d_2 n \ln n}{k}) \to 1$ as $n \to \infty$.

From the above, we conclude that $T = \Theta(\frac{n \ln n}{k})$ w.h.p. For $k = 1$, i.e., the SPR case, we recover the $\Theta(n \ln n)$ result derived in [27]. For values of k approaching n, we recover the $\Theta(\ln n)$ result derived under the idealized MPR setting.

5 Adaptive Neighbor Discovery

We next design two adaptive neighbor discovery schemes that improve upon the Aloha-like scheme described in the previous section. Both schemes utilize feedback information from nodes to achieve faster discovery. One of the schemes requires collision detection at nodes, i.e., the ability to distinguish between a collision and an idle slot, while the other scheme only requires each node to transmit the IDs of the discovered neighbors as feedback to other nodes. We will show that both schemes achieve a factor $\ln n$ improvement over the Aloha-like scheme in a clique setting.

Throughout this section, we assume a clique of n nodes. Furthermore, we assume that n is known to each node and that nodes transmit in synchronized slots. Both these assumptions will be relaxed in subsequent sections.

5.1 Main Idea

The main idea behind our adaptive neighbor discovery schemes is to provide feedback to the transmitting nodes allowing them to stop transmitting once they have been discovered by their neighbors. This in turn reduces channel contention resulting in faster neighbor discovery. As we will see, the use of feedback results in a $\ln n$ factor improvement in running time over the Aloha-like algorithm.

In an SPR network, a successful transmission by a node is received by all other nodes in the clique. The recipient nodes signal the reception status to the transmitting node, thus allowing it to drop out of neighbor discovery, i.e., stop transmitting in the future. In contrast, since MPR capability allows successful reception even in the presence of multiple simultaneous transmissions, a node may be discovered by some subset of its neighbors in the clique, while not being discovered by the remaining subset of neighbors. This occurs for instance under the MPR-k model, when two or more (but fewer than k) nodes transmit simultaneously. Each of the transmitting nodes is discovered by its neighbors, but the transmitting nodes do not discover each other (since nodes have half-duplex radios). We therefore require each node to have m ($m \geq 1$) successful transmissions before dropping out of the neighbor discovery process. We next determine what the appropriate value of m should be.

Lemma 1. *Consider a clique of n nodes under the MPR-k model. A node that has transmitted successfully for m times is discovered by all its neighbors with probability at least $1-(k-1)^m/n^{m-1}$.*

Proof. Consider an arbitrary node x. Suppose that it has transmitted successfully m times. Let $\mathcal{E}_x^t$ denote the event that node x transmits in slot t and the transmission is successful. Given that the event $\mathcal{E}_x^t$ occurs, the number of transmitters in the tth slot is at most k under the MPR-k model. Therefore,

$$\Pr\left[\mathcal{E}_y^t \mid \mathcal{E}_x^t\right] \leq \frac{k-1}{n}.$$

Let the m time slots in which node x transmits successfully be denoted $t_1, \ldots, t_m$. Since the transmissions in different slots are independent, it follows that

$$\Pr\left[\mathcal{E}_y^{t_1} \wedge \cdots \wedge \mathcal{E}_y^{t_m} \mid \mathcal{E}_x^{t_1} \wedge \cdots \wedge \mathcal{E}_x^{t_m}\right] \leq \frac{(k-1)^m}{n^m}. \quad (16)$$

In other words, the probability that a node does not discover x after x has transmitted successfully m times is no more than $(k-1)^m/n^m$. For ease of notation, define $\mathcal{E}_x = \mathcal{E}_x^{t_1} \wedge \cdots \wedge \mathcal{E}_x^{t_m}$. Then (16) can be rewritten as

$$\Pr[\mathcal{E}_y \mid \mathcal{E}_x] \leq \frac{(k-1)^m}{n^m}. \quad (17)$$

Therefore

$$\begin{aligned}\Pr[\mathcal{E}_1 \vee \cdots \vee \mathcal{E}_{n-1} \mid \mathcal{E}_x] &\\ \leq \sum_{r=1}^{n-1} \Pr[\mathcal{E}_r \mid \mathcal{E}_x] &\\ \leq \frac{(n-1)(k-1)^m}{n^m} < \frac{(k-1)^m}{n^{m-1}}.&\end{aligned}$$

Therefore,

$$\Pr[\neg(\mathcal{E}_1 \vee \cdots \vee \mathcal{E}_{n-1}) \mid \mathcal{E}_x] > 1 - \frac{(k-1)^m}{n^{m-1}}.$$

In other words, the probability that node x is discovered by all the neighbors after m successful transmissions is at least $1-\frac{(k-1)^m}{n^{m-1}}$.

We can easily extend the above analysis to the case where node x has transmitted successfully for more than m times, and show that the above result still holds. □

Lemma 1 confirms that when k is small compared to n and m is sufficiently large, a node that has transmitted successfully for m times is discovered by all its neighbors w.h.p. Setting $m=3$ in Lemma 1, we observe that the probability of a successful node being discovered by all its neighbors after m successful transmissions is at least $1-(k-1)^3/n^2$, which is close to 1 for small k and large n. For ease of presentation, we assume $m=3$ throughout our subsequent analysis. Of course, we can employ a larger m, and all our results can be extended in a straightforward manner.

Our adaptive neighbor discovery schemes proceed as follows. We refer to a node that has dropped out of neighbor discovery (i.e., remains in the listen mode) as *passive*. Otherwise, the node is *active*. In the beginning, all nodes are active. We divide time into *phases*. Phase r contains $w_r = \Theta(n_r/k)$ slots, where n_r is the number of active nodes at the beginning of the rth phase. Each node executes the Aloha-like scheme in the rth phase transmitting with probability p_r, a parameter that will be determined in Theorem 2. At the end of a phase, all the nodes that have transmitted successfully m times in the phase become passive. We show in Theorem 2 that when w_r and p_r are chosen appropriately, at least half of the active nodes in a phase will become passive at the end of the phase w.h.p. Therefore, the $(r+1)$-st phase has at most half as many active nodes as in the rth phase, each transmitting with a higher probability than in the rth phase. After $\log \ln n$ phases, we show that at most $n/\ln n$ active nodes are left. Thereafter, the remaining active nodes run the Aloha-like scheme until termination.

The following theorem describes how w_r and p_r are chosen to ensure at least half of the active nodes in the rth phase become passive at the end of the phase w.h.p.

Theorem 2. *Let*

$$[w_r, p_r] = \begin{cases} \left[\frac{\eta n_r}{2}, \frac{1}{n_r}\right], & \text{if } k = 2, \\ \left[\frac{\eta n_r}{k-2}, \frac{k-2}{n_r}\right], & \text{if } k \geq 3, \end{cases}$$

where η is a constant chosen such that

$$\eta \geq \begin{cases} 85, & \text{if } k = 2, \\ 115, & \text{if } k \geq 3 \end{cases}$$

Then,

$$\Pr[n_{r+1} < n_r/2] < e^{-n_r/k}, \quad \forall k \geq 2\,.$$

The proof of Theorem 2 is based on defining a *martingale sequence* [7] and then using the Azuma's inequality [24, Theorem 4.16] to establish the result.

Theorem 3 (Azuma's inequality). *Let the random variables $\tilde{Z}_0, \ldots, \tilde{Z}_n$ form a* martingale *sequence i.e., $E[\tilde{Z}_i \mid \tilde{Z}_0, \ldots, \tilde{Z}_{i-1}] = \tilde{Z}_{i-1}$ for all $i = 1, \ldots, n$. Further, let $|\tilde{Z}_i - \tilde{Z}_{i-1}| \leq k$ for all $i = 1, \ldots, n$. Then,*

$$\Pr[\tilde{Z}_n - E[\tilde{Z}_n] \geq \lambda] \leq \exp\Big(-\frac{\lambda^2}{2nk^2}\Big)$$

and

$$\Pr[\tilde{Z}_n - E[\tilde{Z}_n] \leq -\lambda] \leq \exp\Big(-\frac{\lambda^2}{2nk^2}\Big)\,.$$

We next present the proof of Theorem 2. Since all the phases are similar, it is sufficient to prove the above for phase 1; the proof for the rest of the phases is similar. For ease of exposition, in the following, we drop the subscript that represents the index of a phase, and simply prove the following: Let

$$[w, p] = \begin{cases} \left[\frac{\eta n}{2}, \frac{1}{n}\right], & \text{if } k = 2, \\ \left[\frac{\eta n}{k-2}, \frac{k-2}{n}\right], & \text{if } k \geq 3, \end{cases}$$

where η is a constant as defined earlier (in Theorem 2). Then,

$$\Pr[|S| < n/2] < e^{-n/k}, \quad \forall k \geq 2,$$

where S denotes the set of successful nodes at the end of phase 1.

Proof. We first consider the case of $k \geq 3$. Let random variable B_x^t represents whether node x transmits or not in slot t. Specifically, $B_x^t = 1$ when node x transmits in slot t and $B_x^t = 0$ otherwise. Then $E[B_x^t] = p$. Setting $p = (k-2)/n$, note that for each t, $E[\sum_x B_x^t] = k-2$. The random variable $\sum_x B_x^t$ is binomially distributed; it follows that the mean and median are separated by no more than $\ln(2) \approx 0.69 < 1$ (see [12]) and thus, for each t,

$$\Pr\left[\sum_x B_x^t \geq k-1\right] \leq 1/2\,.$$

Let random variable C_x^t denote whether node i transmits *successfully* or not in slot t. Specifically, $C_x^t = 1$ when node i transmits successfully in slot t and $C_x^t = 0$ otherwise. Then

$$\Pr\big[C_x^t = 1\big] = \Pr\big[B_x^t = 1\big] \cdot \Pr\left[\sum_{y \neq x} B_y^t \leq k-1\right] \geq \frac{p}{2}\,. \tag{18}$$

In other words, when $p = (k-2)/n$, a node can transmit successfully with probability $p/2$. We now want to show that at least half of the nodes can be successful at the end of the phase w.h.p. Define that a node is successful if it transmits successfully for at least three times.[1] That is, $\sum_{t=1}^{w} C_x^t \geq 3$ for at least half of the i.

For convenience, for each i, t define

$$P_i^t = \min\Bigg(3, \sum_{t' \leq t} C_x^{t'}\Bigg), \quad S^t = \big\{x \mid P_x^t = 3\big\}\,.$$

From the definition of S, it follows that $S = S^w$. Reformulating our goal above, we wish to show that for sufficiently large w, $\Pr[|S^w| < n/2] \leq e^{-n/k}$. Note that P_x^t represents node i's "progress" towards being a successful node (i.e., having three successful transmissions) up to time t. Define $P^t = \sum_i P_x^t$, $P^0 = 0$. Then P^t represents the "progress" over all the nodes up to time t. Observe that $P^t \geq \frac{5}{2}n \Rightarrow |S^t| \geq n/2$. So in the following we will focus on establishing that, for sufficiently large w,

$$\Pr[P^w < 5n/2] \leq e^{-n/k}\,. \tag{19}$$

We prove the above using Azuma's inequality. To that end, define Z_t to represent the "progress" of all the nodes in time slot t. Specifically, define

$$Z_t = \begin{cases} P^t - P^{t-1} & \text{if } |S^{t-1}| \leq n/2, \\ k & \text{if } |S^{t-1}| > n/2. \end{cases}$$

Let $Z = \sum_{t=1}^{w} Z_t$. Hence $Z = P^w - P^0 = P^w$, related to our objective in (19). Observe that

$$E\big[Z_t \mid C_x^s, s < t\big],$$

depends only on $|S^{t-1}|$, the number of "saturated" nodes (i.e., those that have already transmitted three times successfully) up to slot $t-1$. When $|S^{t-1}| \leq n/2$, we have

$$\begin{aligned} E\big[Z_t \mid C_x^s, s < t\big] &= \sum_{x \notin S^{t-1}} C_x^t \\ &\geq (n - |S^{t-1}|) \cdot p/2 \\ &\geq (k-2)/4\,. \end{aligned}$$

When $|S^{t-1}| > n/2$, we have

$$E\big[Z_t \mid C_x^s, s < t\big] = k\,.$$

In any case, $E[Z_t \mid C_x^s, s < t] \geq (k-2)/4$. In preparation for application of Azuma's inequality, define

1. This corresponds to $m = 3$; the proof below can be extended in a straightforward manner for other values of m.

$$\tilde{Z}_t = E\big[Z \,|\, B_x^s, s \le t\big] .$$

Observe that [2]

$$E[\tilde{Z}_t \,|\, \tilde{Z}_{t-1}] = \tilde{Z}_{t-1} \quad \text{and} \quad |\tilde{Z}_t - \tilde{Z}_{t-1}| \le k, \quad \text{for all } t .$$

That is, the sequence of random variables $\tilde{Z}_0, \ldots, \tilde{Z}_w$ is a martingale and satisfies the conditions specified in Azuma's inequality. Considering our objective in (19), we have

$$\Pr[P^w < 5n/2] = \Pr[Z < 5n/2] = \Pr[\tilde{Z}^w < 5n/2] .$$

When $w = \eta n/(k-2)$, we have $E[\tilde{Z}^w] = E[Z] \ge w(k-2)/4 = \eta n/4$. Applying Azuma's inequality, we have

$$\begin{aligned}\Pr[\tilde{Z}^w < 5n/2] &= \Pr\left[\tilde{Z}^w < E[\tilde{Z}^w] - \frac{n(\eta-10)}{4}\right] \\ &\le \exp\left(-\frac{n^2(\eta-10)^2(k-2)}{32\eta n k^2}\right) .\end{aligned}$$

When $k = 3$, the right hand side is no more than $e^{-n/k}$ when $\eta \ge 115$. When $k \ge 4$, we have $k/2 \le (k-2)$ and we obtain

$$\Pr[\tilde{Z}^w < 5n/2] \le \exp\left(-\frac{n(\eta-10)^2}{64\eta k}\right) .$$

When $\eta \ge 83$, it can be easily verified that the right hand side is at most $e^{-n/k}$.

We next prove the theorem for the remaining case of $k = 2$. The proof is similar to the case where $k \ge 3$. However, instead of (18), we have

$$\begin{aligned}\Pr\big[C_x^t = 1\big] &= p\left[(1-p)^{n-1} + \binom{n-1}{1} p(1-p)^{n-2}\right] \\ &= \frac{2}{n}\left(1 - \frac{1}{n}\right)^{n-1} \ge \frac{2}{ne} = \frac{2p}{e} .\end{aligned}$$

Following similar steps as those for $k \ge 3$ and letting $w = \eta n/2$ and $\eta \ge 85$, it follows that

$$\begin{aligned}\Pr[|S^w| < n/2] &\le \Pr[P^w < 5n/2] \\ &\le \Pr[\tilde{Z}^w < 5n/2] \le e^{-n/k} .\end{aligned}$$

□

Theorem 2 states that at least half of the active nodes at the beginning of the rth phase become passive at the end of the phase w.h.p. More importantly, the rth phase is of duration linear in the number of active nodes n_r in th phase, which ensures that the total running time of the feedback-based algorithms is $O(n/k)$, thus yielding a $\ln n$ improvement over the Aloha-like algorithm. To see this, recall that the feedback-based algorithms contain two stages: the first stage uses the adaptive scheme until there are fewer than $\frac{n}{\ln n}$ active nodes; the second stage simply uses the Aloha-like scheme. In the first stage, since each phase reduces the number of active nodes by at least one half the number at the start of the phase, the number of active nodes in the rth phase, $n_r \le n/2^{r-1}$. Furthermore, since the first stage stops when there are fewer than $\frac{n}{\ln n}$ active nodes, it contains at most $\log \ln n$ phases. Therefore, the total run time of the first stage is

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

The total run time of the second stage is the time to discover the remaining $n/\ln n$ nodes using the Aloha-like scheme. From the asymptotic results in Section 4.2.2, it is

$$\Theta\left(\frac{n}{k \ln n} \ln \frac{n}{\ln n}\right) = \Theta\left(\frac{n}{k}\right) .$$

Combining the run time of the two stages, the neighbor discovery time is $O(n/k)$.

We next describe the feedback mechanism employed by each of the two adaptive neighbor discovery schemes.

5.2 Collision-Detection Based Neighbor Discovery

In this scheme, a node uses the mechanism in [17], [27] to know whether its transmission is successful or not. In particular, we assume that a node can distinguish between a collision and an idle slot. We divide a slot into two sub-slots. Nodes either transmit or listen in the first sub-slot. If a node listens in the first sub-slot and can decode the received packets successfully, then it deterministically sends a signal in the second sub-slot; otherwise, it remains silent. A node that transmits in the first sub-slot knows its transmission is successful if and only if it hears a signal (or senses energy) in the second sub-slot.

The collision-detection based scheme requires each node to differentiate a collision from an idle slot, which may not be feasible on certain hardware. The ID-based scheme described next eliminates such a requirement.

5.3 ID-Based Neighbor Discovery

In the ID-based scheme, we require each node to record the IDs of the nodes that it hears in each slot. When a node transmits, it transmits its ID as well as the IDs of every node from which it successfully received a message in any of the past slots. The key challenge in the ID-based feedback scheme is in devising an efficient scheme to encode node IDs in the messages transmitted by nodes to ensure that the message lengths remain bounded. A naive implementation of the ID-based feedback scheme in which each node uses the binary representation of the IDs, can lead to very long message lengths. In particular, for the rth phase, since the number of slots is w_r, and given that a node can record up to k IDs in a slot each requiring $\log n$ bits, each message is $O(w_r k \log n)$ bits long. Thus, each message in the first phase is $O(nk \log n)$ bits long.

We next propose a novel message encoding scheme that only requires a message length of $O(\log n)$ bits. In this scheme, each node records the IDs of the nodes that it hears in a slot. In particular, since a node can hear up to k IDs in a slot (under the MPR-k model), for

2. The second of these two claims follows from the fact that if two assignments to the B_x^t differ only among the $\{B_x^s\}$ for a particular s, the value of Z can change by no more than k. See [1, Section 7.4] for a detailed treatment.

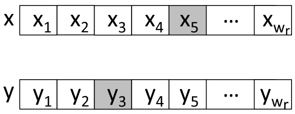


Fig. 1. An example illustrating the ID-based adaptive scheme. Node x records the IDs of the nodes that it hears in slot ℓ as x_ℓ, $\ell = 1, \ldots, w_r$, where w_r is the number of slots in the rth phase. Similarly, node y records the IDs of the nodes that it hears in slot ℓ as y_ℓ, $\ell = 1, \ldots, w_r$. For ease of illustration, a slot is shaded when a node transmits in that slot.

convenience, we require each node to record exactly k IDs in each slot.[3] If a node hears fewer than k IDs, the rest of the IDs are padded as 0. As a special case, if a node transmits in a slot, it records all the k IDs as 0. Fig. 1 shows an example, where x_ℓ denotes the concatenation of the k IDs recorded by node x in the ℓth slot. Note that the received ID sequences at two nodes, x and y are identical in every slot except the ones in which either x or y transmits (but not both). For instance, in the example in Fig. 1, where shaded slots represent the transmission slots, $x_1 = y_1$, $x_2 = y_2$, and $x_4 = y_4$, while $x_3 \neq y_3$ (as y transmits in slot 3) and $x_5 \neq y_5$ (as x transmits in slot 5).

Our encoding scheme takes advantage of the fact that the received ID sequences at different nodes are similar in order to achieve shorter message lengths. The main objective of our encoding scheme is to allow each node x to transmit a short encoded message such that a receiving node y can decode this message to determine the time slots (in the past) in which y's transmissions were successful. We next describe the encoding and decoding procedures in detail.

Consider an arbitrary phase, r. Suppose x transmits and y listens in the tth slot. When x transmits, it transmits its ID along with a coded vector, $\mathbf{h}x = \Sigma_{\ell=1}^{t} \mathbf{v}_\ell x_\ell$, where x_ℓ is the concatenation of the k IDs recorded at node x in the ℓ-th slot, and $\mathbf{v}_\ell$ is a d-dimensional vector that is known to all the nodes beforehand. Specifically, $\mathbf{v}_\ell = (1, a_\ell, \ldots, a_\ell^{d-1})$, where a_ℓ is an element taken from a finite field $\mathbf{F}_q = \{0, 1, \ldots, q-1\}$ of the integers under addition and multiplication modulo $q, \geq q \max(n^k, \max_r w_r)$ is a prime, and d $< q$ is a constant that will be determined in Theorem 4. Intuitively, d is chosen so as to allow a recipient node to solve a system of linear equations to decode messages transmitted by x. When node y receives the message from x, it obtains a vector $\mathbf{h}_x y = \mathbf{h}_x - \mathbf{g}_y$, where $\mathbf{g}_y = \Sigma_{\ell \in C}\ \mathbf{v}_{\ell y \ell}$ and C denotes the set of slots where y listens and does not hear x. The following theorem shows that when d is chosen appropriately, node y can use $\mathbf{h}_{xy}$ to recover the ID sequence recorded at x (and hence determine the slots in which its transmission is successful) w.h.p. In the example in Fig. 1, node x transmits and node y listens in slot 5; y can determine if its transmission in slot 3 is successful after receiving the coded vector from x in slot 5.

Theorem 4. *Let d be a constant chosen such that*

$$d \geq \begin{cases} 170, & \text{if } k = 2, \\ 460, & \text{otherwise.} \end{cases}$$

Then, node y can recover the ID sequence recorded at x upon receiving a coded vector $\mathbf{h}_x$ from node x w.h.p.

Proof. Recall that we consider the scenario where node x transmits and node y listens in slot t. Furthermore, $\mathbf{C}$ denotes the set of slots where y listens and does not hear x. Note that x and y record the same information for any slot in $\mathbf{C}$. That is because, in such a slot, if no more than k nodes transmit, y knows that x must be listening as well, and hence should have recorded the same information for the slot; if there are more than k nodes transmitting, then x and y still record the same information (i.e., a sequence of zeros).

Let $\overline{\mathbf{C}}$ denote the complement of $\mathbf{C}$. That is, $\overline{\mathbf{C}}$ represents the set of slots where either y transmits, or y listens and hears x (i.e., x transmits). Suppose $\overline{\mathbf{C}}$ contains ℓ slots, denoted as $i_1, \ldots, i_\ell$. Recall that $\mathbf{h}_{xy} = \mathbf{h}_x - \mathbf{g}_y$. We next prove that node y can decode $x_{i_1}, \ldots, x_{i_\ell}$ from $\mathbf{h}_{xy}$ w.h.p., and therefore, combining the decoded information for the slots in $\overline{\mathbf{C}}$ and the information for the slots in $\mathbf{C}$ (that y knows), node y can recover the received ID sequence at x.

From the definition of $\mathbf{h}_x$ and $\mathbf{g}_y$, we have

$$\sum_{j=1}^{\ell} \mathbf{v}_{i_j} x_{i_j} = \mathbf{h}_{xy}\,.$$

Writing the above in a matrix form yields

$$\begin{pmatrix} 1 & \ldots & 1 \\ a_{i_1} & \ldots & a_{i_\ell} \\ \vdots & \vdots & \vdots \\ a_{i_1}^{d-1} & \ldots & a_{i_\ell}^{d-1} \end{pmatrix} \begin{pmatrix} x_{i_1} \\ \vdots \\ x_{i_\ell} \end{pmatrix} = \mathbf{h}_{xy}\,. \tag{20}$$

The matrix above is a Vandermonde matrix [14] of dimension $d \times \ell$. There exists a unique solution for $x_{i_1}, \ldots, x_{i_\ell}$ when the rank of the matrix is ℓ. We next show that this condition holds w.h.p. when $d \geq \lceil 4p_r w_r \rceil$, where p_r and w_r are set in accordance with Theorem 2. Let $\mathbf{C}_x$ denote the set of slots in which node x transmits. Similarly, let $\mathbf{C}_y$ denote the set of slots in which node y transmits. Then, $\overline{\mathbf{C}} \subseteq \mathbf{C}_x \cup \mathbf{C}_y$ since $\overline{\mathbf{C}}$ represents the set of slots in which either y transmits, or y listens and hears x (i.e., x transmits). We further let Z be a random variable that denotes the number of slots in which a node transmits, and let μ denote the expectation of Z. Since the transmission probability in the rth phase is p_r, and there are w_r slots in this phase, we have $\mu = p_r w_r$. From Theorem 2, it follows that μ is a constant. Using the tail bound established in [1, Theorem A.1.12], we have

$$\Pr(Z \geq \beta\mu) < (e^{\beta-1}\beta^{-\beta})^{\mu}\,.$$

Letting $\beta = 2$ yields

$$\Pr(Z \geq 2\mu) < (e/4)^{\mu} \approx 0.68^{\mu},$$

which approaches 0. Therefore, we have $|\mathbf{C}_x| \leq 2\mu$ and $|\mathbf{C}_y| \leq 2\mu$ w.h.p. Since

$$|\overline{\mathbf{C}}| \leq |\mathbf{C}_x \cup \mathbf{C}_y| \leq |\mathbf{C}_x| + |\mathbf{C}_y| \leq 4\mu,$$

it therefore follows that when $d \geq \lceil 4p_r w_r \rceil = \lceil 4\mu \rceil$, $d \geq |\overline{\mathbf{C}}| = \ell$ w.h.p. Setting p_r and w_r to the values as defined

3. To ensure that the recorded ID sequences at all the receiving nodes in a slot are the same, we also require the IDs be recorded in the same order (increasing or decreasing).

in Theorem 2, we have $d \geq 2 \times 85 = 170$ when $k = 2$ and, $d \geq 4 \times 115 = 460$ when $k \geq 3$. Furthermore, note that

$$\begin{vmatrix} 1 & \dots & 1 \\ a_{i_1} & \dots & a_{i_\ell} \\ \vdots & \vdots & \vdots \\ a_{i_1}^{\ell-1} & \dots & a_{i_\ell}^{\ell-1} \end{vmatrix} = \Pi_{j>u}(a_{i_j} - a_{i_u}) \neq 0,$$

since the left-hand side is the determinant of a square Vandermonde matrix. Therefore, node y can solve the system of linear equations in (20) to obtain $x_{i_1}, \dots, x_{i_\ell}$. □

The coded vector transmitted by a node contains d elements, where d is a constant defined in Theorem 4. Each element has $O(\max(\log w_r + k \log n,\ 2k \log n)) = O(k \log n)$ bits. Therefore, the message length transmitted by each node is $dO(k \log n) = O(k \log n)$ bits, thus yielding a $O(n)$ reduction over a naive ID-based scheme that transmits the binary representation of all the IDs discovered in each time slot. Finally, we remark that such a system of linear equations can be solved extremely efficiently over finite fields [8], [9]; in particular, Gaussian elimination still applies in this setting.

6 Practical Considerations

So far, we have assumed that each node knows the number of its neighbors n and that nodes transmit in synchronized slots. In this section, we relax each of these assumptions.

6.1 Unknown Number of Neighbors

We now describe how the neighbor discovery schemes described in the previous sections can be extended to handle the scenario where n is not known a priori. Our schemes are similar in spirit to the algorithm proposed in [27] for the case of SPR networks. The main idea is to keep doubling the estimate for n until a stopping rule is satisfied.

6.1.1 Aloha-Like Scheme

The algorithm runs in *stages*. In the rth stage, each node runs the Aloha-like scheme for a duration of w_r slots with transmission probability $p_r = 1/2^r$. Based on our asymptotic analysis in Section 4.2, we choose $w_r = c_1 \ln 2^r$ under the idealized MPR model and $w_r = d_1 2^r \ln 2^r / k$ under MPR-k, where c_1 and d_1 are constants, defined in Sections 4.2.1 and 4.2.2, respectively. Each node records the number of neighbors that it discovers in the rth stage, denoted as n_r, and decides to terminate the neighbor discovery process in the $r+1$-st stage if the stopping rule

$$n_r \geq 2^{r-1} \wedge n_{r+1} \leq 2^r, \tag{21}$$

is satisfied.

With our choice of w_r, a node correctly terminates at the end of $\lceil \log n \rceil + 1$-st stage and discovers all its neighbors w.h.p. This is because for this choice of w_r, we have $n_{\lceil \log n \rceil} = n \geq 2^{\lceil \log n \rceil - 1}$ and $n_{\lceil \log n \rceil + 1} = n \leq 2^{\lceil \log n \rceil}$ w.h.p., which satisfies the stopping rule at the end of $\lceil \log n \rceil + 1$-st stage.

To verify the correctness of our stopping rule, we simulate the Aloha-like scheme under the idealized MPR and MPR-k (for $k = 2$ and 8) models for $n = 5, 10, 20, 50, 100, 200$. For each setting, we repeat the simulation 100 times. We indeed find that each node terminates only after discovering all its neighbors.

We next briefly describe the total neighbor discovery time of the above scheme. Under idealized MPR, the neighbor discovery time is

$$\sum_{r=1}^{\lceil \log n \rceil} c_1 \ln 2^r = \Theta(\log n \ln n).$$

Under MPR-k, the neighbor discovery time is

$$\sum_{r=1}^{\lceil \log n \rceil} \frac{d_1 2^r \ln 2^r}{k} = \Theta\left(\frac{n \ln n}{k}\right).$$

From the above, we observe that not knowing n leads to at most a $\log n$ factor slowdown under idealized MPR, and a constant factor slowdown under the MPR-k model.

6.1.2 Adaptive Neighbor Discovery

Our extension for the adaptive neighbor discovery schemes is similar to that for the Aloha-like scheme. More specifically, the schemes run in stages doubling the estimate for n after each stage. In the rth stage, we assume there are 2^r neighbors and run the adaptive schemes as described in Section 5. Therefore, each stage contains multiple phases with the total run time of the rth stage being $O(\frac{2^{r+1}}{k})$. We continue the stages until the stopping rule (21) is satisfied. Similar to the extension for the Aloha-like scheme, the neighbor discovery stops in $(\lceil \log n \rceil + 1)$-st stage. We then have a total run time that equals

$$\sum_{r=1}^{\lceil \log n \rceil} O\left(\frac{2^{r+1}}{k}\right) = O\left(\frac{n}{k}\right).$$

Again, we observe that not knowing n leads to a constant factor slowdown.

6.2 Asynchronous Transmissions

Thus far, our discussion assumed a slotted time system and that different nodes are synchronized on slot boundaries. We next relax this assumption. In particular, we consider the following asynchronous model. We assume that time is slotted and each slot is of duration τ. However, the slot boundaries across different nodes are not necessarily aligned.

We consider the asynchronous version of the Aloha-like algorithm where each node transmits with probability p at the beginning of a slot. Consider two arbitrary nodes, x and y. Suppose node x transmits at time t. Let p_s denote the probability that x is discovered by y. Since the slot boundaries of nodes x and y are not aligned, node y cannot transmit in the two adjacent slots that overlap with the interval $[t, t+\tau]$ in order to successfully receive x's transmission. The probability of this event is $(1-p)^2$. Then we may expand p_s as

$$p_s = p(1-p)^2 \cdot \sum_{i=0}^{k-1} \binom{n-2}{i} (1-(1-p)^2)^i (1-p)^{2(n-2-i)}. \tag{22}$$

Under the idealized MPR model, $p_s = (1-p)^2 p$, and hence the optimal transmission probability p^* can be easily obtained as $p^* = 1/3$. The corresponding optimal p_s^* is $4/27$. The results under the MPR-k model are summarized in the following theorem.

Theorem 5. *Consider the Aloha-like algorithm where nodes are assumed to know n and node transmissions are asynchronous. Under the MPR-k model, the optimal transmission probability $p^* = \alpha k/n$, where $\alpha \approx 0.5$ when $k = 1$, and*

$$\alpha \in \begin{cases} (0.03, 14.26), & \text{if } k = 2, \\ (0.06, 8.49), & \text{if } k = 3, \\ (0.05, 6.76), & \text{if } k \geq 4\,. \end{cases}$$

Proof. When $k = 1$, i.e., the SPR model, setting the derivative of p_s to zero, we obtain $p^* = 1/(2n-1) \approx \frac{1}{2n}$, and hence, $\alpha \approx 0.5$. We next derive the results for the other three cases: $k = 2$, $k = 3$, and $k \geq 4$. Similar to the proof of Theorem 1, for each case, we first derive a lower bound on the optimal p_s^*, and then derive the constants stated in the theorem. Note that since we are considering the MPR-k case, we have $n \geq k+2$.

Let $B = B_1 + \cdots + B_{n-2}$, where $B_x = 1$ when node x transmits in one of the two adjacent slots, and $B_x = 0$ otherwise. Then B follows a Binomial distribution, and (22) can be rewritten as

$$p_s = p(1-p)^2 \Pr(B < k)\,. \tag{23}$$

Note that

$$p_s^* = p^*(1-p^*)^2 \Pr(B < k) < \frac{\alpha k}{n} \Pr(B < k)\,.$$

Using the left tail bound for Binomial distribution in [1, Theorem A.1.13], we have

$$\begin{aligned} \Pr(B < k) &< e^{-((n-2)(1-(1-p^*)^2)-k)^2/2(n-2)(1-(1-p^*)^2)} \\ &\approx e^{-(n(1-(1-p^*)^2)-k)^2/2n(1-(1-p^*)^2)} \\ &= e^{-\frac{(np^*(2-p^*)-k)^2}{2p^*(2-p^*)}} = e^{-\frac{(\alpha k(2-p^*)-k)^2}{2\alpha k(2-p^*)}} \\ &= e^{-\frac{k(\alpha(2-p^*)-1)^2}{2\alpha(2-p^*)}} < e^{-\frac{k(\alpha-1)^2}{4\alpha}}, \end{aligned}$$

where the last inequality follows by observing that $1 < 2 - p^* < 2$. Hence,

$$p_s^* < \frac{\alpha k}{n} e^{-\frac{k(\alpha-1)^2}{4\alpha}}\,. \tag{24}$$

When $k = 2$, letting $1-(1-p)^2 = 1/(n-2)$ yields $p \geq 1/2(n-2)$, and hence

$$\begin{aligned} p_s &\geq \frac{1}{2(n-2)}\left(1-\frac{1}{n-2}\right)\left(1-\frac{1}{n-2}\right)^{n-2} \\ &\quad + \frac{1}{2(n-2)}\left(1-\frac{1}{n-2}\right)^{n-2} \\ &\geq \frac{1}{2e^2(n-2)}\left(1-\frac{1}{n-2}+1\right) \\ &\geq \frac{1}{2e^2(n-2)}\,. \end{aligned}$$

Since p_s^* denotes the optimal value of p_s, we have

$$p_s^* \geq \frac{1}{2e^2(n-2)}\,. \tag{25}$$

Since $p_s^* < p^* = \alpha k/n = 2\alpha/n$, it follows from (25) that

$$\alpha > \frac{n}{n-2}\frac{1}{4e^2} \geq \frac{1}{4e^2} \approx 0.03\,.$$

On the other hand, a simple numerical calculation from (24) reveals that when $\alpha > 14.26$, $p_s^* < \frac{1}{2e^2(n-2)}$, contradicting (25). Hence $\alpha \in (0.03, 14.26)$ when $k = 2$.

When $k = 3$, letting $1-(1-p)^2 = 1/(n-3)$ yields $p \geq 1/2(n-3)$, and

$$p_s \geq \frac{5n-18}{4e^2(n-3)^2}\,.$$

Since p_s^* denotes the optimal value of p_s, we have

$$p_s^* \geq \frac{5n-18}{4e^2(n-3)^2}\,. \tag{26}$$

Since $p_s^* < p^* = \alpha k/n = 3\alpha/n$, it follows from (26) that

$$\alpha > \frac{n(5n-18)}{6e^2(n-3)^2} \geq \frac{5}{12e^2} \approx 0.06\,.$$

On the other hand, a simple numerical calculation from (24) reveals that when $\alpha > 8.49$, $p_s^* < \frac{5n-18}{4e^2(n-3)^2}$, contradicting (26). Hence $\alpha \in (0.06, 8.49)$ when $k = 3$.

When $k \geq 4$, let $1-(1-p)^2 = (k-3)/(n-2)$. Then the mean of the Binomial random variable, B, is $(n-2)(1-q) = k-3$. Since the mean and the median are at most $\ln 2$ apart [12], the median is in $[k-3-\ln 2, k-3+\ln 2]$. Since $k-1 > k-3+\ln 2$, we have $\Pr(B < k) \geq 1/2$. Since $n \geq k+2$, it follows that

$$(1-p)^2 = 1 - \frac{k-3}{n-2} \geq 1 - \frac{k-3}{k} = \frac{3}{k}\,.$$

Further simplification yields

$$2p - p^2 = \frac{k-3}{n-2} \Rightarrow p \geq \frac{k-3}{2(n-2)}\,.$$

Summarizing the above and since p_s^* denotes the optimal value of p_s, we have

$$p_s^* \geq p_s \geq \frac{k-3}{2(n-2)} \cdot \frac{3}{k} \cdot \frac{1}{2} = \frac{3(k-3)}{4k(n-2)}\,. \tag{27}$$

Since $p_s^* < p^* = \alpha k/n$, it follows from (27) that

$$\alpha > \frac{3(k-3)n}{4k(n-2)k} \geq \frac{3(k-3)}{4k^2} \geq \frac{3}{64} \approx 0.05\,.$$

We next derive the condition on α which results in

$$\frac{\alpha k}{n} e^{-\frac{k(\alpha-1)^2}{4\alpha}} \leq \frac{3(k-3)}{2k(n-2)},$$

a contradiction stating that an upper bound of p^* is no more than a lower bound of p^* (see (24) and (27)). The above is equivalent to

$$\alpha e^{-\frac{k(\alpha-1)^2}{4\alpha}} \leq \frac{3(k-3)n}{2k^2(n-2)} . \quad (28)$$

In (28), the left hand side is a decreasing function of k, and hence the maximum value is $\alpha e^{-\frac{4(\alpha-1)^2}{4\alpha}}$. On the other hand, as described earlier, the right hand side is larger than 0.05. When $\alpha > 6.76$, we have $\alpha e^{-\frac{4(\alpha-1)^2}{4\alpha}} < 0.05$, leading to the contradiction in (28). Hence, $\alpha \in (0.05, 6.76)$ when $k \geq 4$. □

Following an analysis similar to that in Section IV-B, it is straightforward to conclude that the asynchronous Aloha-like algorithm is at most a constant factor slower than the synchronous version.

Extension of the adaptive neighbor discovery schemes to the asynchronous model is similar and is omitted here to avoid repetition. We can also extend the asynchronous schemes for the case when n is unknown by requiring each node to embed its stage number in the messages, leading to at most a factor of two slowdown compared to the synchronous model.

7 The Multi-Hop Network Case

We next generalize the analysis of our neighbor discovery from a clique setting to that of a multi-hop wireless network. In particular, we first describe our problem formulation, and then present upper bounds on neighbor discovery time for the Aloha-like and adaptive algorithms under the MPR-k model.

7.1 Problem Setting

We consider a multi-hop wireless network $G = (V, E)$, where V denotes the set of nodes and E denotes the set of directed edges, i.e., $(x, y) \in E$ if node y is within x's transmission range. Let $|V| = n$. Further, let Δ be the maximum node degree in G. We say that an edge (x, y) has been *discovered* when node y successfully receives a transmission from x (i.e., in this case, y discovers x). We are interested in determining an upper bound on the time T until all edges in E have been discovered.

7.2 Aloha-Like Neighbor Discovery

The Aloha-like neighbor discovery algorithm in the general network setting operates as follows. In each time slot, a node transmits with probability $\alpha k/\Delta$ and listens with probability $1 - \alpha k/\Delta$.

Consider a pair of nodes x and y such that $(x, y) \in E$. Let n_y denote the number of nodes within y's transmission range. The edge (x, y) is discovered in a given time slot when x transmits, y listens, and no more than k of y's neighbors transmit. From (14), we know that the probability p_{xy} of link (x, y) being discovered in a given time slot is given by

$$p_{xy} \simeq \frac{\gamma k}{n_y},$$

where γ is a constant. Noting that $n_y \leq \Delta$, we obtain

$$p_{xy} \geq \frac{\gamma k}{\Delta} .$$

Now, let $\mathcal{E}_{xy}(t)$ denote the event that the link (x, y) is not discovered after t time slots. Thus,

$$\begin{aligned} P(\mathcal{E}_{xy}(t)) &= (1 - p_{xy})^t \\ &\leq \left(1 - \frac{\gamma k}{\Delta}\right)^t \\ &\leq e^{-\frac{\gamma k t}{\Delta}}, \end{aligned}$$

where the last inequality follows from the well-known fact that $1 - x \leq e^{-x}, \forall x \in \mathbb{R}$.

Let $\mathcal{E}(t)$ denote the event that at least one link in G is not discovered by time t. Therefore,

$$\begin{aligned} P(\mathcal{E}(t)) &\leq P\left(\bigcup_{(x,y)\in E} \mathcal{E}_{xy}(t)\right) \\ &\leq \sum_{(x,y)\in E} P(\mathcal{E}_{xy}(t)) \\ &\leq \sum_{(x,y)\in E} e^{-\frac{\gamma k t}{\Delta}}, \end{aligned}$$

where the second inequality follows from the union bound. Noting that there are at most n^2 directed edges in a graph with n nodes, we obtain

$$P(\mathcal{E}(t)) \leq n^2 e^{-\frac{\gamma k t}{\Delta}} .$$

Letting $t = \frac{3\Delta \ln n}{\gamma k}$, we obtain

$$P(\mathcal{E}(t)) \leq n^2 e^{-3 \ln n} = \frac{1}{n} .$$

Since γ is a constant, we conclude that $T = O(\frac{\Delta \ln n}{k})$ w.h.p.

Since each node can receive at most k successful transmissions in each time slot, it follows immediately that a lower bound on the running time of any neighbor discovery algorithm under the MPR-k model is $\Omega(\frac{\Delta}{k})$. Given the upper bound derived earlier, we conclude that the Aloha-like neighbor discovery algorithm is at most a factor $\ln n$ worse than the optimal.

In the light of the results above, an interesting question that follows is whether we can close the $\ln n$ gap between the lower and upper bounds via adaptive neighbor discovery. We next show that this is indeed true when Δ is large. Analysis of adaptive neighbor discovery for general Δ in a general network setting appears non-trivial and is an interesting future direction.

7.3 Adaptive Neighbor Discovery

We next consider adaptive neighbor discovery in a multi-hop setting when Δ is large. More formally, we restrict our analysis to the case when $\Delta = n/c$, where c is a constant. We show that the neighbor discovery time is $O(n/k) = O(\Delta/k)$, which matches the lower bound for the problem.

Consider an arbitrary node x. We say that x's transmission in a given time slot is *successful*, if it is discovered by all its neighbors that are listening in the slot. A node knows whether its transmission is successful or not based on feedback from other nodes. Specifically, we divide a slot into three sub-slots. In the first sub-slot, nodes either transmit or listen. If a node decodes the received packets successfully, then it deterministically sends a signal in the second sub-slot; otherwise, it deterministically sends a signal in the third sub-slot. A node, x, that transmits in the first sub-slot knows its transmission is successful if and only if it hears a signal (or senses energy) in the second sub-slot and does not hear a signal in the third sub-slot. This is because hearing a signal in the second sub-slot indicates that no more than k of x's neighbors have transmitted in the first sub-slot; and not hearing a signal in the third sub-slot indicates that all of x's neighbors that listened in the first sub-slot have discovered x.

The reason for having the additional third sub-slot when compared to our algorithms for clique topologies is as follows. In a multi-hop network, nodes have different sets (and numbers) of neighbors. One of x's neighbors, y, may not discover x even if it listens and no more than k of x's neighbors transmit, as k or more of y's neighbors could be transmitting in the same slot.

Thus, the probability that a neighbor of x does not discover x when x has a successful transmission is no more than $(k-1)/\Delta$, since there can be no more than $k-1$ other transmitters in x's neighborhood when x's transmission is successful. Specifically, similar to Lemma 1, we have

Lemma 2. *Consider a network with maximum node degree Δ under the MPR-k model. A node that has transmitted successfully for m times is discovered by all its neighbors with probability at least $1-(k-1)^m/\Delta^{m-1}$.*

The above Lemma indicates that when k is small compared to Δ and, m is sufficiently large, a node that has transmitted successfully for m times is discovered by all its neighbors w.h.p.

We then use the same adaptive algorithm as described in Section 5.1. Specifically, we divide time into phases, and classify nodes as active or passive nodes (initially all nodes are active). In a phase, only active nodes transmit and passive nodes listen. A node that has m successful transmissions becomes passive at the end of the phase. From Theorem 2, we conclude that at least half of the active nodes at the beginning of the rth phase become passive by the end of the phase w.h.p. To see why this is the case, we define a stronger notion of successful transmission. We say that a node's transmission is *strongly successful*, if no more than k nodes transmit across the entire network. It is obvious that a strongly successful transmission is also a successful transmission. When phase length and transmit probability are chosen as in Theorem 2, it follows that at least half of the active nodes have at least m strongly successful transmissions, and hence at least m successful transmissions, by the end of a phase. Therefore, at least half of the active nodes will become passive. After $\log \ln n$ phases, we run the Aloha-like scheme until termination. Following the analysis in Section 5.2, we conclude that the neighbor discovery time is $O(n/k)$. Since $\Delta = n/c$, the neighbor discovery time is $O(\Delta/k)$.

8 Conclusions and Future Work

In this paper, we designed and analyzed randomized algorithms for neighbor discovery for both clique and general network topologies under various MPR models. For clique topologies, we started with an Aloha-like algorithm that assumes synchronous node transmissions and a priori knowledge of the number of neighbors n. We showed that the total neighbor discovery time for this algorithm is $\Theta(\ln n)$ under the idealized MPR model, and $\Theta(\frac{n \ln n}{k})$ under the MPR-k model. We further designed adaptive neighbor discovery algorithms for the case when a node knows if its transmission is successful or not, and showed that it provides a factor $\ln n$ improvement over the Aloha-like scheme. We extended our schemes to accommodate a number of practical scenarios such as when the number of neighbors is not known beforehand and the nodes are allowed to transmit asynchronously. We analyzed the performance of our algorithms in each case and demonstrated at most a constant or $\Theta(\ln n)$ factor slowdown in algorithm performance. Finally, we consider the general multi-hop network setting and show that the Aloha-like scheme achieves an upper bound of $O(\frac{\Delta \ln n}{k})$ w.h.p, at most a factor $\ln n$ worse than the optimal, and the adaptive algorithm is order-optimal i.e., it achieves an upper bound of $O(\Delta/k)$ when Δ is large.

We have used neighbor discovery time as the performance metric throughout the paper. Another interesting metric is energy consumption during the neighbor discovery process. Energy consumption of the Aloha-like algorithm can be directly derived from neighbor discovery time. Analyzing energy consumption of the adaptive algorithms in more involved and is left as future work. Another interesting direction of future work is extending our study to more generalized MPR models (e.g., accounting for fading, shadowing and other random errors observed in wireless channels).

Acknowledgments

An earlier version of this paper appeared in MobiHoc 2011 [29]. The authors would like to thank Shengli Zhou (UConn) for helpful discussions on MPR models, and the anonymous reviewers for their insightful comments. This research was supported in part by the National Science Foundation under Grant 0835735 and CAREER Award 0746841, and under ARO W911NF-08-1-0233. Any opinions, findings, and conclusions or recommendations expressed in this paper are those of the authors and do not necessarily reflect the views of the funding agencies.

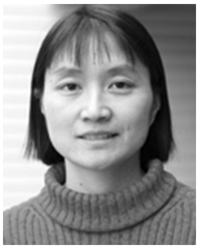
References

[1] N. Alon and J. Spencer, *The Probabilistic Method*. Hoboken, NJ, USA: Wiley, 2008.
[2] D. Angelosante, E. Biglieri, and M. Lops, "Neighbor discovery in wireless networks: A multiuser-detection approach," in *Proc. Inform. Theory Appl. Workshop*, Feb. 2007, pp. 46–53.
[3] D. Angelosante, E. Biglieri, and M. Lops, "A simple algorithm for neighbor discovery in wireless networks," in *Proc. IEEE Int. Conf. Acoustics, Speech Signal Process.*, Apr. 2007, pp. 169–172.
[4] C. L. Arachchige, S. Venkatesan, and N. Mittal, "An asynchronous neighbor discovery algorithm for cognitive radio networks," in *Proc. IEEE Symp. New Frontiers Dyn. Spectr. Access Netw.*, 2008, pp. 1–5.
[5] S. A. Borbash, A. Ephremides, and M. J. McGlynn, "An asynchronous neighbor discovery algorithm for wireless sensor networks," *Ad Hoc Netw.*, vol. 5, no. 7, pp. 998–1016, 2007.
[6] R. Cohen and B. Kapchits, "Continuous neighbor discovery in asynchronous sensor networks," *IEEE/ACM Trans. Netw.*, vol. 19, no. 1, pp. 69–79, Feb. 2011.
[7] J. Davidson, *Stochastic Limit Theory*. London, U.K. Oxford Univ. Press, 1994.

[8] J.-G. Dumas, P. Giorgi, and C. Pernet, (2014). [Online]. Available: http://www-ljk.imag.fr/membres/Jean-Guillaume.Dumas/FFLAS/
[9] J.-G. Dumas, P. Giorgi, and C. Pernet, "FFPACK: Finite field linear algebra package," in *Proc. Int. Symp. Symbolic Algebraic Comput.*, 2004, pp. 119–126.
[10] P. Dutta and D. Culler, "Practical asynchronous neighbor discovery and rendezvous for mobile sensing applications," in *Proc. 6th ACM Conf. Embedded Netw. Sensor Syst.*, 2008, pp. 71–84.
[11] S. Ghez, S. Verdu, and S. C. Schwartz, "Stability properties of slotted Aloha with multipacket reception capability," *IEEE Trans. Autom. Control*, vol. 33, no. 7, pp. 640–649, Jul. 1988.
[12] K. Hamza, "The smallest uniform upper bound on the distance between the mean and the median of the binomial and poisson distributions," *Statist. Probability Lett.*, vol. 23, no. 1, pp. 21–25, Apr. 1995.
[13] W. B. Heinzelman, A. P. Chandrakasan, and H. Balakrishnan, "An application-specific protocol architecture for wireless microsensor networks," *IEEE Trans. Wireless Commun.*, vol. 1, no. 4, pp. 660–670, Oct. 2002.
[14] R. A. Horn and C. R. Johnson, *Topics in Matrix Analysis*. Cambridge, U.K.: Cambridge Univ. Press, 1991.
[15] G. Jakllari, W. Luo, and S. V. Krishnamurthy, "An integrated neighbor discovery and MAC protocol for ad hoc networks using directional antennas," *IEEE Trans. Wireless Commun.*, vol. 6, no. 3, pp. 1114–1024, Mar. 2007.
[16] A. Keshavarzian, E. Uysal-Biyikoglu, F. Herrmann, and A. Manjeshwar, "Energy-efficient link assessment in wireless sensor networks," in *Proc. 23rd Annu. Joint Conf. IEEE Comput. Commun. Soc.*, Mar. 2004, pp. 1751–1761.
[17] R. Khalili, D. Goeckel, D. Towsley, and A. Swami, "Neighbor discovery with reception status feedback to transmitters," in *Proc. IEEE 29th Conf. Inform. Commun.*, Mar. 2010, pp. 2375–2383.
[18] M. Kohvakka, J. Suhonen, M. Kuorilehto, V. Kaseva, M. Hannikainen, and T. D. Hamalainen, "Energy-efficient neighbor discovery protocol for mobile wireless sensor networks," *Ad Hoc Netw.*, vol. 7, no. 24, pp. 24–41, Jan. 2009.
[19] S. Krishnamurthy, N. Mittal, R. Chandrasekaran, and S. Venkatesan, "Neighbor discovery in multi-receiver cognitive radio networks," *Int. J. Comput. Appl.*, vol. 31, no. 1, Jan. 2009.
[20] L. Li, J. Y. Halpern, P. Bahl, Y.-M. Wang, and R. Wattenhofer, "A cone-based distributed topology-control algorithm for wireless multi-hop networks," *IEEE/ACM Trans. Netw.*, vol. 13, no. 1, pp. 147–159, Feb. 2005.
[21] D. D. Lin and T. J. Lim, "Subspace-based active user identification for a collision-free slotted ad hoc network," *IEEE Trans. Commun.*, vol. 52, no. 4, pp. 612–621, Apr. 2004.
[22] J. Luo and D. Guo, "Neighbor discovery in wireless ad hoc networks based on group testing," in *Proc. 46th Annu. Allerton Conf. Commun., Control, Comput.*, Sep. 2008, pp. 791–797.
[23] M. J. McGlynn and S. A. Borbash, "Birthday protocols for low energy deployment and flexible neighbor discovery in ad hoc wireless networks," in *Proc. 2nd ACM Int. Symp. Mobile Ad Hoc Netw. Comput.*, Oct. 2001, pp. 137–145.
[24] R. Motwani and P. Raghavan, *Randomized Algorithms*. Cambridge, U.K. Cambridge Univ. Press, 1995.
[25] R. Ramanathan, J. Redi, C. Santivanez, D. Wiggins, and S. Polit, "Ad hoc networking with directional antennas: A complete system solution," *IEEE J. Select. Areas Commun.*, vol. 23, no. 3, pp. 496–506, Mar. 2005.
[26] Y.-C. Tseng, C.-S. Hsu, and T.-Y. Hsieh, "Power-saving protocols for IEEE 802.11-based multi-hop ad hoc networks," in *Proc. IEEE 21st Annu. Joint Conf. IEEE Comput. Commun. Soc*, 2002, pp. 200–209.
[27] S. Vasudevan, M. Adler, D. Goeckel, and D. Towsley, "Efficient algorithms for neighbor discovery in wireless networks," *IEEE/ACM Trans. Netw.*, vol. 21, no. 1, pp. 69–83, Feb. 2013.
[28] S. Vasudevan, J. Kurose, and D. Towsley, "On neighbor discovery in wireless networks with directional antennas," in *Proc. IEEE 24th Annu. Joint Conf. IEEE Comput. Commun. Soc.*, 2005, pp. 2502–2512.
[29] W. Zeng, X. Chen, A. Russell, S. Vasudevan, B. Wang, and W. Wei, "Neighbor discovery in wireless networks with multipacket reception," in *Proc. 12th ACM Int. Symp. Mobile Ad Hoc Netw. Comput.*, May 2011.
[30] Z. Zhang and B. Li, "Neighbor discovery in mobile ad hoc self-configuring networks with directional antennas algorithms and comparisons," *IEEE Trans. Wireless Commun.*, vol. 7, no. 5, pp. 1540–1549, May 2008.
[31] Q. Zhao and L. Tong, "A multiqueue service room MAC protocol for wireless networks with multipacket reception," *IEEE/ACM Trans. Netw.*, vol. 11, no. 1, pp. 125–137, Feb. 2003.
[32] Q. Zhao and L. Tong, "A dynamic queue protocol for multiaccess wireless networks with multipacket reception," *IEEE Trans. Wireless Commun.*, vol. 3, no. 6, pp. 2221–2231, Nov. 2004.
[33] R. Zheng, J. C. Hou, and L. Sha, "Asynchronous wakeup for ad hoc networks," in *Proc. 4th ACM Int. Symp. Mobile Ad Hoc Netw. Comput.*, 2003, pp. 35–45.

Alexander Russell received the baccalaureate degrees in computer science and mathematics from Cornell University, the MS degree in computer science and electrical engineering, and the PhD degree in mathematics from the Massachusetts Institute of Technology. He is currently a professor of Computer Science and Engineering at the University of Connecticut.

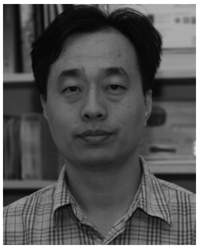
Sudarshan Vasudevan received the MS and PhD degrees in computer science from the University of Massachusetts, Amherst, in 2003 and 2006, respectively. He is currently a senior research engineer at Palo Alto Networks, Inc. in Santa Clara, California. From September 2010 to March 2014, he was a research scientist at Bell Labs, Murray Hill, New Jersey. Previously, he was a research scientist at Adverplex, Inc., a Search Engine Marketing company based in Boston. His research interests include computer networks, modeling, and performance evaluation. He is a member of the IEEE.



Bing Wang received the BS degree in computer science from the Nanjing University of Science & Technology, China, in 1994, and the MS degree in computer engineering from the Institute of Computing Technology, Chinese Academy of Sciences in 1997. She then received the MS degrees in computer science and applied mathematics, and the PhD degree in computer science from the University of Massachusetts, Amherst, in 2000, 2004, and 2005, respectively. She is currently an associate professor of computer science and engineering at the University of Connecticut. Her research interests include computer networks, multimedia, and distributed systems. She received NSF CAREER award in February 2008. She is a member of the IEEE.

Wei Zeng received the BS and MS degrees in computer science and engineering from the South China University of Technology, China, in 2000 and 2003, respectively, and the PhD degree in computer science and engineering from the University of Connecticut in 2012. Currently she is a computer analyst in Connecticut Transportation Safety Research Center at the University of Connecticut. Her research interests include data mining, network diagnosis, network measurement, and network management. She is a member of the IEEE.

Xian Chen received the BS degree in applied mathematics from the Beijing University of Aeronautics and Astronautics, China, in 2003, the MS degree in computer science from Beijing Jiao-Tong University, China, in 2006, and the PhD degree in computer science and engineering from the University of Connecticut in 2013. He currently works at Microsoft. His research interests include the areas of computer networks, fault diagnosis, and performance modeling. He is a member of the IEEE.



Wei Wei received the BS degree in applied mathematics from Beijing University, China, in 1992, and the MS degree in statistics from Texas A & M University in 2000. He then received MS degrees in computer science and applied mathematics, and the PhD degree in computer science from the University of Massachusetts, Amherst, in 2004, 2004, and 2006, respectively. He is currently a research assistant professor in the Computer Science and Engineering Department at the University of Connecticut. His research interests include the areas of computer networks, distributed embedded systems, statistical inference, and performance modeling. He is a member of the IEEE.