Heterogeneous Multi-channel Neighbor Discovery for Mobile Sensing Applications: Theoretical Foundation and Protocol Design

Lin Chen
LRI-CNRS UMR 8623
Univ. of Paris-Sud
chen@lri.fr

Kaigui Bian
Inst. Netw. Comp. & Inf. Sys.,
School of EECS, Peking Univ.
bkg@pku.edu.cn

Meng Zheng
Key Lab. Netw. Contr. Sys.
Shengyang Inst. Automation
Chinese Academy of Sciences
zhengmeng_6@sia.cn

ABSTRACT

Neighbor discovery is of paramount importance in mobile sensing applications that rely heavily on data timely collected and shared among nearby users. Guaranteed discovery with bounded latency and supporting heterogenous duty cycles to provide fine-grained control of energy conservation levels are among the most crucial requirements in the design of efficient neighbor discovery protocols. While simultaneously satisfying these two requirements is non-trivial, the situation is exacerbated if the operating frequencies of mobile devices span multiple channels and discovery occurs only if nodes switch to the same channel. In this paper, we formulate this problem as *heterogeneous multi-channel neighbor discovery* problem and establish a theoretical framework of the problem, under which we derive the performance bound of any neighbor discovery protocol. Based on the theoretical results, we then develop Mc-Dis (Multi-channel Discovery), a novel multi-channel discovery protocol that (1) achieves guaranteed discovery with order-minimal worst-case discovery delay and (2) supports almost all duty cycles to provide fine-grained control of energy conservation levels.

Categories and Subject Descriptors

C.2.1 [**Computer-Communications Networks**]: Network Architecture and Design—*Wireless Communication*; F.2.2 [**Analysis of Algorithms and Problem Complexity**]: Nonnumerical Algorithms and Performance

Keywords

Neighbor Discovery; Multi-channel; Energy Efficiency; Wireless networks

1. INTRODUCTION

The ever-growing deployment of millions of personal mobile devices, e.g., smart-phones and tablets, provides a fertile ground for numerous mobile sensing applications ranging

from mobile social networking [11], proximity-based gaming [1] to participatory and crowd sensing [5]. The success of such applications, where mobile devices equipped with different types of sensors interact with each other upon encounters, relies heavily on data timely collected and shared among the nearby users in an opportunistic fashion.

The supporting primitive that identifies all the neighbors in a mobile device's communication range is referred to as *neighbor discovery* protocols, one of the bootstrapping primitives supporting many basic network functionalities, such as topology control, clustering, medium access control and routing. Ideally, nodes should discover their neighbors as quickly as possible for other protocols to start execution.

However, designing efficient neighbor discovery protocols for mobile sensing applications is particularly challenging due to the stringent energy constraint of mobile devices. Specifically, these devices, usually battery-constrained, alternate between active and sleeping modes by turning their radios on only periodically. This energy-saving technique is called *duty cycling*. Despite its effectiveness in energy conservation, the duty cycling technique significantly challenges the neighbor discovery protocol design in the quest of limiting discovery latency with low power consumption. Specifically, the two important design objectives, saving energy through a duty-cycle based scheduling and limiting the neighbor discovery latency, are at odd with each other.

Moreover, the operating frequencies of mobile devices typically span a swath of spectrum subdivided into multiple orthogonal channels. Such multi-channel characteristic brings an additional dimension to the neighbor discovery problem, as each pair of neighbors not only need to wake up at the same time slot, but also should switch to the same channel in order to discover each other. Wireless channels are notoriously unstable and the channel conditions may vary in both time and space domains. Any two nodes may have different channel perceptions due to their locations, traffic patterns, interference, noises, etc. Consequently, to achieve maximal discovery robustness, an effective neighbor discovery protocol needs to ensure discovery between any pair of neighbors on every common channel they can access.

We coin the term *heterogeneous multi-channel neighbor discovery* problem to denote the following problem in the context described above: *How can neighbor nodes with heterogeneous duty cycles, operating on different channels, without clock synchronization, discover each other over every common channel within a bounded delay?* Particularly, the following requirements should be satisfied:

- Maximal discovery diversity/robustness with bounded (and minimal) worst-case discovery delay;
- Support for heterogenous and arbitrary duty cycles to provide fine-grained control of energy saving levels.

It is the combination of the above two design requirements that makes the neighbor discovery protocol design far from trivial and should be handled holistically. As reviewed in Sec. 2, no existing work, to the best of our knowledge, can satisfy both requirements simultaneously. To address the heterogeneous multi-channel neighbor discovery problem, we present the design and evaluation of a multi-channel neighbor discovery protocol, termed as Mc-Dis (Multi-channel Discovery), that can guarantee discovery between any pair of neighbor nodes over *every* common channel within *bounded* delay while supporting almost all duty cycles. The main contributions of the paper are articulated as follows.

- *Theoretical foundation:* We establish a theoretical framework to study the heterogeneous multi-channel neighbor discovery problem, under which we derive the performance bound of any neighbor discovery protocol.
- *Protocol design:* We design Mc-Dis, a novel multi-channel discovery protocol that achieves guaranteed discovery with order-minimal worst-case discovery delay and supports almost all duty cycles.

By applying theoretic tools in number theory and graph theory (more specifically, the Chinese Remainder Theory [14] and the Maximum independent Set problem [6]), the proposed neighbor discovery protocol, Mc-Dis, has the following noteworthy properties which make it especially suitable for mobile sensing applications.

- *Fine-grained control of duty cycle*: In contrast to existing solutions using prime numbers or power-multiples of minimal duty cycles, Mc-Dis can practically support almost all duty cycles, thus providing much more fine-grained control of energy conservation levels.
- *Bounded worst-case discovery delay*: Mc-Dis achieves bounded discovery delay even between nodes with heterogeneous duty cycles.
- *Full discovery diversity*: Mc-Dis guarantees discovery over *each* channel, thus minimising the probability of discovery failures caused by various channel problems.
- *Robustness against asymmetrical channel perceptions*: Mc-Dis achieves the same discovery performance even if nodes have asymmetrical channel perceptions, either on the accessible channel set or on the channel index.
- *Robustness against clock drift*: Mc-Dis achieves the same performance even if clocks of any two nodes drift away from each other by an *arbitrary* amount of time.

The rest of the paper is organized as follows. Sec. 2 summarises the related work on neighbor discovery. Sec. 3 describes the system model and formulates the optimal heterogeneous multi-channel neighbor discovery problem. Sec. 4 establishes the theoretical performance bound. Sec. 5 presents the design of Mc-Dis in the single-channel case and performs a theoretical analysis on its performance. Sec. 6 further presents the design of Mc-Dis in the mult-channel case and investigates its performance there. Sec. 7 presents the simulation results. Sec. 8 concludes the paper.

2. RELATED WORK

Existing work on neighbor discovery in duty-cycled wireless networks can be categorized into *probabilistic* and *deterministic* protocols.

Probabilistic protocols (e.g. [10,16,17,19,23]) adopt probabilistic strategies at each node. A representative one is the birthday protocol [10] where nodes transmit/receive or sleep with different probabilities. The work in [19] and [23] further addresses the case with multi-packet reception and directional antennas. Probabilistic protocols have the advantages of being memoryless and stationary and thus are especially robust and adapted in ad hoc environments where no a priori knowledge or coordination is available. The main drawback is the lack of performance guarantee in terms of discovery delay.

Deterministic protocols, on the other hand, are proposed to provide strict bound on the discovery delay [3, 4, 7, 24]. In deterministic protocols, each node wakes up according to certain schedule carefully tuned to ensure that each pair of two wake-up schedules overlap in at least one active slot. Based on the design of wake-up schedule, deterministic protocols can further be divided into three classes. The first class of them, based on Quorum [9,15], construct the wake-up schedule by assigning a column and a row of an $m \times m$ array to each node such that no matter which row and column are selected, any two nodes have at least two overlapping awaken slots. The main drawback of the Quorum-based approaches is the support of only symmetrical duty cycles [15]. Although enhanced solutions have been proposed to support asymmetric duty cycles, only two different duty cycles can be supported [9]. The second class of deterministic protocols overcome this limitation by using prime numbers to guarantee bounded discovery delay even for asymmetrical duty cycles. A typical one in this class is Disco [4], in which each node selects two prime numbers, based on which its wake-up schedule is configured. A more recent proposition, U-Connect [7], uses a single prime number per node and has a shorter discovery delay, given the same duty cycle. The third class, proposed in [3], employs two kinds of wake-up slots, termed as anchor slots and probe slots, to achieve both lower worst-case and average discovery delay. In [21] and [22], two protocols are proposed to be implemented on top of deterministic protocols to achieve further performance gain, either in energy conservation or discovery delay. One drawback of existing deterministic protocols is the failure to support all duty cycles due to their limited choice on either prime numbers or power-multiples of the smallest duty cycles, and consequently only a limited choices of energy conservation levels can be supported.

There are some multi-channel neighbor discovery protocols proposed for non duty-cycled networks. Mittal *et al.* proposed a suite of probabilistic and deterministic neighbor discovery protocols [8,12,13,20] for cognitive radio networks (CRNs). Arachchige *et al.* developed a leader election protocol to setup a CRN in which a leader is selected based on node IDs and then performs neighbor discovery by periodically transmitting beacons [2]. Karowski *et al.* developed neighbor discovery protocols for IEEE 802.15 networks to minimise the expected discovery delay in that context.

Despite extensive research efforts devoted to neighbor discovery, none of them can solve the heterogeneous multi-channel neighbor discovery problem by achieving bounded discovery delay for nodes operating on heterogenous duty cycles. In this regard, our work provides a systematic formulation and analysis on the heterogeneous multi-channel neighbor discovery problem and the design of a functional

neighbor discovery protocol that satisfies the design requirements posed in Sec. 1.

3. HETEROGENEOUS MULTI-CHANNEL NEIGHBOR DISCOVERY

3.1 System Model

We consider a time-slotted (but not necessarily synchronized) energy-constraint wireless network operating on a set $\mathcal{N}$ of N channels. To discover its neighbors in the multi-channel environment, each node wakes up periodically and switches across different channels. The main design challenges we need to address are summarised as follows:

- *Lack of clock synchronization*: Due to the resource constraint, it is extremely difficult to maintain tight synchronization among the local clocks of different nodes, and thus the clocks of any two nodes may drift away from each other by an arbitrary amount of time, which may lead to the discovery failure.
- *Asymmetrical duty cycle lengths*: The duty cycle lengths of two network nodes are typically asymmetrical, depending on their independent energy constraint and the applications running on them. Neighbor discovery protocols should ensure that any two nodes can wake up in a same slot on the same channel regardless of their asymmetrical duty cycle lengths.
- *Asymmetrical channel perceptions*: Wireless channels are notoriously unstable and the channel conditions may vary in both time and space domains. Consequently, any two nodes may have different channel perceptions due to their locations, traffic patterns, interference, noises, etc. Formally, each node u has its own perception on $\mathcal{N}$, denoted as $\mathcal{N}_u$.

In the following, we formally define the *neighbor discovery schedule* that characterises the wake-up and channel hopping pattern of a node.

DEFINITION 1 (NEIGHBOR DISCOVERY SCHEDULE). *The neighbor discovery schedule of a node u is defined as a sequences $\mathbf{x}_u \triangleq \{x_u^t\}_{1 \leq t \leq T_u}$, where T_u is the period of the sequence[1], and*

$$x_u^t = \begin{cases} 0 & u \text{ sleeps in slot } t \\ n \in \mathcal{N}_u & u \text{ wakes up on channel } n \end{cases}.$$

Consider two nodes a and b with their neighbor discovery schedules being $\mathbf{x}_a$ and $\mathbf{x}_b$ whose periods are T_a and T_b. Given the periodicity of $\mathbf{x}_a$ and $\mathbf{x}_b$, it suffices to consider consecutive $T_a T_b$ slots, i.e, $1 \leq t \leq T_a T_b$. If $\exists t \in [1, T_a T_b]$ and $h \in \mathcal{N}$ such that $x_a^t = x_b^t = h$, we say that a and b can discover each other in slot t on channel h. Slot t is called the discovery slot and channel h is called the discovery channel between a and b. Example 1 illustrates the above definition.

EXAMPLE 1. *Consider a network of two channels and two nodes a, b whose neighbor discovery schedules are $\mathbf{x}_a = \{0, 0, 1\}$ and $\mathbf{x}_b = \{0, 1, 0, 2\}$ with $T_a = 3$ and $T_b = 4$. The duty cycles of a and b are $d_a = 3$ and $d_b = 2$. The neighbor discovery schedules of a and b are repeated each 12 slots, as illustrated in Fig. 1 for one period. We can observe that a and b can discover each other on slots 6 on channel 1.*

Slot index	1	2	3	4	5	6	7	8	9	10	11	12	...
Node a :	0	0	1	0	0	1	0	0	1	0	0	1	...
Node b :	0	1	0	2	0	1	0	2	0	1	0	2	...

Figure 1: Neighbor discovery schedule example.

To model the situation where the clocks of different nodes are not synchronised[2], we apply the concept of *cyclic rotation* to neighbor discovery schedules. Specifically, given a neighbor discovery schedule $\mathbf{x}_a$, we denote $\mathbf{x}_a(k)$ a cyclic rotation of $\mathbf{x}_a$ by k slots where k is called the cyclic rotation phase. In Example 1, we have $\mathbf{x}_a(1) \triangleq \{1, 0, 0\}$ and $\mathbf{x}_b(2) = \{0, 2, 0, 1\}$.

3.2 Optimal Neighbor Discovery Problem

Performance Metric 1: Maximal Time to Discovery For the neighbor discovery problem, the primary performance metric is the maximal time to discovery (MTTD), i.e., the worst-case discovery delay. Given two nodes a and b, the MTTD between them is defined as the upper-bound of the latency (in number of slots) before successful mutual discovery for all possible clock drift between them. Reconsider Example 1, we can observe that the MTTD is 11, achieved between $\mathbf{x}_a(6)$ and $\mathbf{x}_b(6)$.

Performance Metric 2: Discovery Diversity The other metric, particularly pertinent for the multi-channel environment, is the *discovery diversity*, which characterizes the capability of a neighbor discovery protocol of discovering a neighbor regardless of its operational channel. We say that a neighbor discovery protocol achieves *full* discovery diversity if the discovery of any pair of nodes is guaranteed on *every* common channel they can access. It can be checked that the neighbor discover schedule in Example 1 cannot achieve full discovery diversity as a and b can never discover each other on channel 2.

Performance Metric 3: Maximal Time to Full Discovery Diversity When full discovery diversity can be achieved, we further define the third metric *maximal time to full discovery diversity* (MTTFDD) as the worst-case delay to achieve full discovery diversity. MTTFDD can be regarded as a generalisation of MTTD in multi-channel networks. MTTFDD degenerates to MTTD in single-channel networks. Throughout the paper, we analyse MTTD in single-channel case and MTTFDD in multi-channel case.

We conclude this section by formulating the optimal heterogeneous multi-channel neighbor discovery problem.

PROBLEM 1. *The optimal heterogeneous multi-channel neighbor discovery problem is defined as follows:*
minimise $\quad T$,
subject to $\quad \forall t_a^0 \in [1, T_a], t_b^0 \in [1, T_b], \forall d_a, d_b, \exists t \leq T$
$\qquad\qquad$ *such that* $x_a^t(t_a^0) = x_b^t(t_b^0) = h, \forall h \in \mathcal{N}_a \bigcap \mathcal{N}_b.$
That is, devising neighbor discover schedules to minimize T, the worst-case discovery delay while achieving full discovery diversity between any pair of nodes a and b for any duty cycle pair (d_a, d_b), any initial time offset t_a^0 and t_b^0 and any channel perception $\mathcal{N}_a$ and $\mathcal{N}_b$.

In what follows, we first establish a theoretical performance bound of any neighbor discovery protocol. We then

---

[1]A random neighbor discovery schedule is a special case where $T_u \to \infty$.

[2]Here we assume that clocks of different nodes are asynchronous but their slot boundaries are aligned. The situation of unaligned slot boundaries is analysed in Sec. 6.4.

present the baseline design and optimisation of Mc-Dis in the single-channel case, before proceeding to the multi-channel case with symmetrical channel perception (i.e., $\mathcal{N}_a = \mathcal{N}_b$). We complete our analysis by addressing the generic case with asymmetrical channel perceptions and arbitrary clock drift to iron out a version of Mc-Dis that works in practice.

4. PROTOCOL-INDEPENDENT DISCOVERY DELAY BOUND

Armed with the theoretical framework established previously, this section derives the performance bound of any multi-channel neighbor discovery protocol achieving full discovery channel diversity. The result derived in this section establishes a lower-bound of the solution of Problem 1.

THEOREM 1. *(Protocol-independent Bound of MTTFDD) For any neighbor discovery protocol achieving full discovery channel diversity, the MTTFDD between any pair of nodes a and b, denoted by L, is lower-bounded by $N^2 d_a d_b$, where d_a and d_b denote the duty cycles of a and b.*

PROOF. Let T_a and T_b denote the period of $\mathbf{x_a}$ and $\mathbf{x_b}$, i.e., the neighbor discovery schedules of a and b. It can be noted that regardless of the clock drift, the neighbor discovery schedules of a and b repeats every $T_a T_b$ time slots. Hence, if they can discovery each other with full discovery diversity regardless of the clock drift, the worst-case discovery delay until full diversity L is upper-bounded by $T_a T_b$.

Without loss of generality, we fix $\mathbf{x_a}$ and cyclically rotate $\mathbf{x_b}$ by l slots, denoted as $\mathbf{x_b}(l)$, where $l = 0, 1, \ldots, T_a T_b - 1$. Now consider $\mathbf{x_a}$ and $\mathbf{x_b}(l)$. Recall that the maximal time to full discovery diversity is the worst-case discovery delay until full diversity among all initial clock phases of a and b, there must be at least N discovery slots each L slots where both a and b wakes up in the slot, resulting a minimal number of discovery slots $N T_a T_b / L$ within consecutive $T_a T_b$ slots. Let S denote the total number of accumulated discovery slots within consecutive $T_a T_b$ slots between $\mathbf{x_a}$ and $\mathbf{x_b}(l)$ as l is incremented from 0 to $T_a T_b - 1$, we have

$$S \geq \frac{N(T_a T_b)^2}{L}. \tag{1}$$

On the other hand, let i_a^h (i_b^h, respectively) denote the number of time slots in $\mathbf{x_a}$ ($\mathbf{x_b}$, respectively) in which a (b) wakes up on channel h within consecutive $T_a T_b$ slots. We can express the duty cycles of a and b as

$$d_a = \frac{T_a T_b}{\sum_{h \in \mathcal{N}} i_a^h}, \quad d_b = \frac{T_a T_b}{\sum_{h \in \mathcal{N}} i_b^h}.$$

After some algebraic operations, we obtain

$$T_a T_b = \sum_{h \in \mathcal{N}} d_a i_a^h = \sum_{h \in \mathcal{N}} d_b i_b^h = \sum_{h \in \mathcal{N}} \frac{d_a i_a^h + d_b i_b^h}{2}. \tag{2}$$

Since $\mathbf{x}_a$ and $\mathbf{x}_b(l)$ achieve full discovery diversity, for any channel h, the total accumulated number of discoveries between $\mathbf{x}_a$ and $\mathbf{x}_b(l)$, as l is incremented from 0 to $T_a T_b - 1$, in which the discovery channel is h, is $i_a^h \cdot i_b^h$.

Hence the total number of accumulated discoveries, as l is incremented from 0 to $T_a T_b - 1$, is $S = \sum_{h \in \mathcal{N}} i_a^h \cdot i_b^h$.

Noticing $d_a i_a^h d_b i_b^h \leq \left(\frac{d_a i_a^h + d_b i_b^h}{2} \right)^2$, it follows from (2) that

$$S = \sum_{h \in \mathcal{N}} i_a^h \cdot i_b^h = \frac{\sum_{h \in \mathcal{N}} d_a i_a^h \cdot d_b i_b^h}{d_a d_b} \leq \frac{(T_a T_b)^2}{d_a d_b N}$$

It then follows from (1) that $\frac{N(T_a T_b)^2}{L} \leq \frac{(T_a T_b)^2}{d_a d_b N}$, which leads to $L \geq N^2 d_a d_b$. $\square$

Theorem 1 derives the performance limit of any neighbor discovery protocol. We can further generalise Theorem 1 on the pair-wise neighbor discovery to the network-wise neighbor discovery, as stated in the following corollary.

COROLLARY 1. *For any network where the largest two duty cycles of nodes are d_1 and d_2, the MTTFDD between any pair of nodes in the network is lower-bounded by $N^2 d_1 d_2$ for any neighbor discovery protocol. Asymptotically, when $d_1 \simeq d_2 \simeq O(d)$, $L \simeq O(N^2 d^2)$.*

Corollary 1 can also be viewed from another angle: to achieve a target MTTFDD L, the duty cycle of nodes should be upper-bounded by $O\left(\frac{\sqrt{L}}{N}\right)$. Consequently, the energy consumption cannot be lower than $O\left(\frac{\sqrt{L}}{N}\right)$.

In what follows, we develop a multi-channel neighbor discovery protocol termed as Mc-Dis that approaches the derived performance bound. To streamline our presentation, we first develop Mc-Dis for the single-channel case and then proceed to the multi-channel case. After specifying the protocol and deriving its performance, we investigate the case with asymmetrical channel perceptions and arbitrary clock drift to iron out a version of Mc-Dis that works in practice.

5. MC-DIS: SINGLE-CHANNEL CASE

5.1 Motivation and Protocol Design

In the single-channel case, the neighbor discovery schedule $\mathbf{x}_u$ for each node u degenerates to a binary sequence where

$$x_u^t = \begin{cases} 1 & u \text{ wakes up in slot } t, \\ 0 & u \text{ sleeps in slot } t. \end{cases}$$

Each node wakes up periodically to discover its neighbors. The wake-up period is determined by its duty cycle. Specifically, we consider two neighboring nodes a and b with duty cycles d_a and d_b. To discover each other, nodes a and b wake up every d_a and d_b slots, i.e., $x_a(t) = 1$ for $t = k d_a$ and $x_b(t) = 1$ for $t = k d_b + \delta_{ab}$ where δ_{ab} is the clock offset between a and b, $k = 1, 2, \cdots$. It follows from the Chinese Remainder Theorem [14] that if d_a and d_b are co-prime to each other, the two nodes are ensured to discover each other regardless of δ, i.e., there exists t_d such that $x_a(t_d) = x_b(t_d) = 1, \forall \delta_{ab}$.

However, assigning co-prime numbers to each node in a distributed way is far from trivial. A commonly adopted solution is to use only prime numbers because two distinct prime numbers are by definition co-prime to each other, as in Disco [4] and U-Connect [7]. However, limiting the choices to prime numbers fail to support all the duty cycles due to the limited number of prime numbers. Note that among duty cycles smaller than 1000, only $\frac{1}{6}$ are prime numbers.

Motivated by the above analysis, Mc-Dis adopts the following neighbor discovery schedule. For each node u with duty cycle d_u,

$$x_u(t) = \begin{cases} 1 & t \text{ is divisible by either } 2d_u - 1 \text{ or } 2d_u + 1, \\ 0 & \text{otherwise.} \end{cases}$$

EXAMPLE 2. *Consider two nodes a and b with $d_a = 3$, $d_b = 5$ with a clock offset $\delta_{ab} = 1$. Under Mc-Dis, using the time of a as reference, a wakes up in slots $5k$ and $7k$, i.e., $5, 7, 10, 14, 15, 20, 21, \cdots$, b wakes up in slots $9k+1$ and $11k+1$, i.e., $10, 12, 19, 23, \cdots$, as illustrated in Fig. 2. The discovery happens in slot 10.*

The period of $\mathbf{x}_u$ in Mc-Dis is $(2d_u - 1)(2d_u + 1)$, in which there are $4d_u - 1$[3] active slots. Hence, the actual average duty cycle, denoted as $\widehat{d}_u$, is $\frac{(2d_u - 1)(2d_u + 1)}{4d_u - 1}$ which approaches to the required duty cycle d_u when d_u is large. Generally, the relative error between $\widehat{d}_u$ and d_u is upper-bounded by 7.1%, as established in the following lemma.

LEMMA 1. *The relative error between the duty cycle of the neighbor discovery schedule $\widehat{d}_u$ and the required duty cycle d_u is upper-bounded by 7.1%*

PROOF. Denote the relative error between $\widehat{d}_u$ and d_u as ϵ, we have:

$$\epsilon = \left[\frac{(2d_u - 1)(2d_u + 1)}{4d_u - 1} - d_u \right] / d_u = \frac{d_u - 1}{d_u(4d_u - 1)}.$$

It can be checked that for all $d_u \geq 1$, ϵ is upper-bounded by 7.1%, with the upper-bound achieved when $d_u = 2$. $\square$

In practical applications, d_u is usually large, ϵ is thus much smaller than the upper-bound 7.1%: e.g., ϵ drops below 2.3% when $d_u \geq 10$. Asymptotically, $\epsilon \simeq O\left(\frac{1}{4d_u}\right)$.

5.2 Mc-Dis Core Idea: Regular Duty Cycles

Following the Chinese Remainder Theorem, the mutual discovery of two neighbor nodes a and b in Mc-Dis, regardless of their clock drift, requires at least one of $2d_a \pm 1$ to be co-prime with at least one of $2d_b \pm 1$. In the vast majority of cases, this requirement can be satisfied. To illustrate this, if we allow the maximal duty cycle D to be 100, then all duty cycles except 17 and 38 can be supported by Mc-Dis; if we allow the maximal duty cycle D to be 1000, only 43 duty cycles cannot be supported, i.e., Mc-Dis can support nearly 96% of all duty cycles.

In this subsection, we conduct a formal analysis on the design idea of Mc-Dis. We start by formulating the definition of *regular duty cycles* that are natively supported by Mc-Dis.

DEFINITION 2 (REGULAR DUTY CYCLE). *Given the duty cycle upper-bound D, we call a duty cycle d a regular duty cycle if for any $2 \leq d' \leq D$, at least one number from $2d \pm 1$ is co-prime with at least one number from $2d' \pm 1$.*

For two nodes a and b, if at least one of their duty cycles d_a and d_b is regular, a and b can discover each other. Reconsider Example 2 with $D = 1000$, it can be checked that the duty cycles $d_a = 3$ and $d_b = 5$ are both regular. Evidently the two nodes can discover each other, as illustrated in Fig. 2.

We conclude this subsection by stating the following properties of the regular duty cycles:

- *The vast majority of duty cycles are regular.* As illustrated in the beginning of the subsection, more than 96% duty cycles are regular. In contrast, in existing solutions based on prime numbers, only a small portion of duty cycles can be supported due to the limited choice of prime numbers.

[3]Note that $(2d_u - 1)(2d_u + 1)$ is divisible by both $2d_u - 1$ and $2d_u + 1$.

- *There are no three consecutive non-regular duty cycles.* In other words, if d is non-regular, at least one from $d \pm 1$ is regular. This implies that if the required duty cycle d happens to be a non-regular one, the node can operate on $d + 1$ or $d - 1$. If we take the case of using $d - 1$, the effective duty cycle is $\frac{[2(d-1)-1][2(d-1)+1]}{4(d-1)-1}$ The relative error to the required duty cycle can be computed as

$$\epsilon = \left[\frac{[2(d-1)-1][2(d-1)+1]}{4(d-1)-1} - d \right] / d = \frac{-3d_u + 3}{d(4d-5)},$$

which is decreasing in d. In the case of $D = 1000$, the smallest non-regular duty cycle being 17, ϵ is upper-bounded by 4.5% for all non-regular duty cycles.

The analysis in this subsection demonstrates that Mc-Dis can support most duty cycles and even in the cases where the duty cycles cannot be directed supported, Mc-Dis can use neighboring duty cycles with almost negligible errors.

Concerning the implementation aspect of Mc-Dis, we would like to emphasis that the regular duty cycles can be pre-calculated off-line and stored in a look-up table.

5.3 Discovery Delay Upper-bound

In single-channel case, only the first performance metric (maximal time to discovery, MTTD) is applicable. In Theorem 2, we derive the MTTD of Mc-Dis between two nodes a and b if at least one of d_a and d_b is regular.[4]

THEOREM 2 (DISCOVERY DELAY UPPER-BOUND). *Given any two nodes a and b, if at least one of their duty cycles d_a and d_b is regular, they are ensured to discover each other within at most $(2d_a + 1)(2d_b + 1)$ slots.*

PROOF. Recall the definition of the regular duty cycle, at least one of $2d_a \pm 1$ is co-prime with at least one of $2d_b \pm 1$. It follows from The Chinese Remainder Theorem that a and b can discover each other within at most $(2d_a + 1)(2d_b + 1)$ slots, regardless of their clock offset. $\square$

5.4 Mc-Dis Optimisation: Supporting More Duty Cycles

From previous analysis, we can see that using non-regular duty cycles may result in discovery failure. For example, if the duty cycle upper-bound is $D = 100$, node a with duty cycle $d_a = 17$ may never discover node b with duty cycle $d_b = 38$ because neither $2d_a \pm 1$ (33 and 35) is co-prime to either $2d_b - 1$ (75) or $2d_b + 1$ (77). A direct solution is to remove 17 and 38 from the usable duty cycle set and use 16 and 37 instead. However, since one of d_a and d_b being regular is sufficient to ensure discovery, we can do better by removing only 38 from the set of usable duty cycles. Now 17 becomes usable as for any duty cycles other than 38, discovery is guaranteed with 17. In this subsection, we explore the natural question of constructing the usable duty cycle set with the maximal number of elements, formalised as follows.

PROBLEM 2. *Let U denote the usable duty cycle set with the duty cycle upper-bound D,*

[4]Throughout our analysis, we focus on the pair-wise discovery between any pair of neighbor nodes a and b. The obtained results can be readily generated to the network level where each node should discover all its neighbor nodes by following the same way as Theorem 1 and Corollary 1.

Figure 2: Mc-Dis in single-channel case: $d_a = 3$, $d_b = 5$.

Figure 3: Duty cycle graph (part).

maximise $|U|$
subject to $\forall d_1, d_2 \in U$, $d_1 \neq d_2$, at least one of $2d_1 \pm 1$ is co-prime with at least one of $2d_2 \pm 1$.

Solving Problem 2: A Graph-based Approach

We address Problem 2 by casting it into a problem on a graph. Specifically, we construct a graph in which each vertex represents a duty cycle and there exists an edge between two vertexes if the duty cycles represented by the two vertexes may fail to discover each other (mathematically, neither $2d_a \pm 1$ is co-prime to either $2d_b - 1$ or $2d_b + 1$). We observe that the duty cycle graph typically consists of a number of non-connected clusters. Fig. 3 illustrates one of such cluster for $D = 1000$. We seek to remove the minimal number of vertexes (and the edges connected to them) such that each of the remaining vertex is isolated, meaning that the remaining vertexes represent the usable duty cycles. It can be checked that in the cluster of Fig. 3, we need to remove at least two nodes, e.g., 38 and 137. Unfortunately, Problem 2 can not be solved within polynomial time, as proved in the following lemma.

LEMMA 2. *Problem 2 is NP-hard.*

PROOF. We establish the NP-hardness of Problem 2 by casting it into the problem of finding maximum independent sets (MIS) [6] which is NP-hard. An independent set (IS) of a graph is a set of vertices, no two of which are adjacent. That is, it is a set $\mathcal{I}$ of vertices such that for any two vertices in $\mathcal{I}$, there is no edge connecting them. Equivalently, each edge in the graph has at most one endpoint in $\mathcal{I}$. An MIS is an IS with maximum cardinality, i.e., contains the maximal number of vertices. Consider Fig. 3, an MIS is $\{17, 423\}$.

Clearly, solving Problem 2 is equivalent to constructing MIS in the duty cycle graph that we construct. Since finding an MIS is proved NP-hard, Problem 2 is also NP-hard. □

Given its NP-hardness, we develop a heuristic *polynomial* algorithm (Algorithm 1) to solve Problem 2 based on the observation that the duty cycle graph is only loosely connected and that the maximal degree of the graph is limited (typically no more than 3). The heuristic algorithm consists of iteratively adding the vertex with the smallest degree and removing the edges and vertexes connected to it until when graph becomes empty.

Algorithm 1 Calculate the heuristic usable duty cycle set

Input: Duty cycle graph G
Output: Usable duty cycle set U
   Initialisation: $U \leftarrow \emptyset$
   while G is not empty **do**
      Find a vertex v of minimum degree in G
      $U \leftarrow U \cup \{v\}$
      Remove v and its neighbors from G
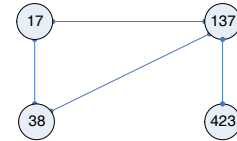   **end while**

THEOREM 3   ($\frac{1}{\Delta+1}$-OPTIMALITY OF ALGO 1). *Algo 1 gives a $\frac{1}{\Delta+1}$-approximation for the maximal usable duty cycle set in a duty cycle graph with the maximal degree Δ.*

PROOF. To prove the theorem, we show that the output of Algorithm 1 U satisfies $|U| \geq \frac{|U^*|}{\Delta+1}$ where U^* is the maximal usable duty cycle set. To this end, we upper-bound the number of vertexes in $V \setminus U$. It follows from Algorithm 1 that a vertex u is in $V \setminus U$ because it is removed as a neighbor of some node $v \in U$ when v is added to U. Since any vertex v has at most Δ neighbors, it holds that $|V \setminus U| \leq \Delta|U|$. Hence, we have $|U| \geq \frac{|V|}{\Delta+1} \geq \frac{|U^*|}{\Delta+1}$ by noticing that U^* is a subset of V. □

We run Algorithm 1 for practical scenarios from $D = 100$ to 2000 and report that it always returns the correct maximal usable duty cycle set. Note that Algorithm 1 can be executed off-line to generate a look-up table that contains all non-supported duty cycles in Mc-Dis. Each node only need to check if its required duty cycle is in the table each time when it needs to set/reset the Mc-Dis parameters.

By Algorithm 1, we can typically reduce the non-supported duty cycles by more than 50%. For example, for $D = 100$, we can support all duty cycles except 38; for $D = 500$, the number of duty cycles that cannot be supported by Mc-Dis reduces from 26 to 12, i.e., less than 2.5% of the total duty cycles; for $D = 1000$, the same number reduces from 43 to only 18, i.e., less than 2% of the total duty cycles.

6. MC-DIS: MULTI-CHANNEL CASE

In this section, we present the Mc-Dis design for the multi-channel case and establish the performance bound.

6.1 Neighbor Discovery Schedule Construction

The neighbor discovery schedule of Mc-Dis for each node in the multi-channel case is constructed based on its globally unique ID such as its MAC address, which can be mathematically expressed as a binary sequence of length l. Using globally unique IDs is a typical method to break the symmetry of any pair of nodes. The neighbor discovery schedule construction process is composed of three steps, summarised here and detailed in the following analysis.

- *Step 1:* Each node u independently generates a padded binary sequence $\mathbf{o}_u$ based on its ID such that the padded binary sequences of any two nodes are *cyclic rotationally distinct* one to the other;
- *Step 2:* Each node i independently generates a sequence $\mathbf{s}_u$ based on $\mathbf{o}_u$ such that for any two nodes a, b and any initial time offset t_a^0 and t_b^0, there always exist four time slots l_{ij} ($i, j \in \{0, 1\}$) such that $s_a^{l_{ij}}(t_a^0) = i$ and $s_b^{l_{ij}}(t_b) = j$. We denote such sequences $\mathbf{s}_u$ as *regular sequences*;
- *Step 3:* Each node i generates its neighbor discovery schedule based on $\mathbf{s}_u$.

Step 1: Constructing cyclic rotationally distinct padded binary sequence

As the first step, each node independently generates a binary sequence based on its ID such that the binary sequences of any two nodes are *cyclic rotationally distinct* one to the other. Note that the sequences resulting from cyclic rotations of a sequence are not considered to be cyclic rotationally distinct with respect to each other and the original sequence. We next show how to construct such cyclic rotationally distinct binary sequences.

Let α denote the ID of a node a and let **1** (**0**) denote a sequence of 1 (0) of length $l' = \lceil \frac{l}{2} \rceil$. We construct the padded ID of Alice as the concatenation of **0**, α and **1**, denoted as **0**$||\alpha||$**1**. By the following lemma, we show that the padded ID sequences generated in such way based on different ID sequences are cyclic rotationally distinct one to another.

LEMMA 3. *Given any two padded ID sequences* **a** *and* **b** *generated from two ID sequences* α *and* β *in the way that* $\mathbf{a} \triangleq \mathbf{0}||\alpha||\mathbf{1}$ *and* $\mathbf{b} \triangleq \mathbf{0}||\beta||\mathbf{1}$, *it holds that*

$$\alpha \neq \beta \implies \mathbf{a} \neq \mathbf{b}(k), \ \forall k \in (0, l+2l'],$$

where $\mathbf{b}(k)$ *is* **b** *with a cyclic rotation of k bits.*

PROOF. We prove the lemma by considering four possible scenarios illustrated in Figure 4, and showing, in each scenario, that a bit in **a** and another bit in $\mathbf{b}(k)$ have different values although the two bits are in the same position within the respective padded ID sequences. This is sufficient to prove that the two padded ID sequences **a** and **b** are cyclic rotationally distinct one to the other.
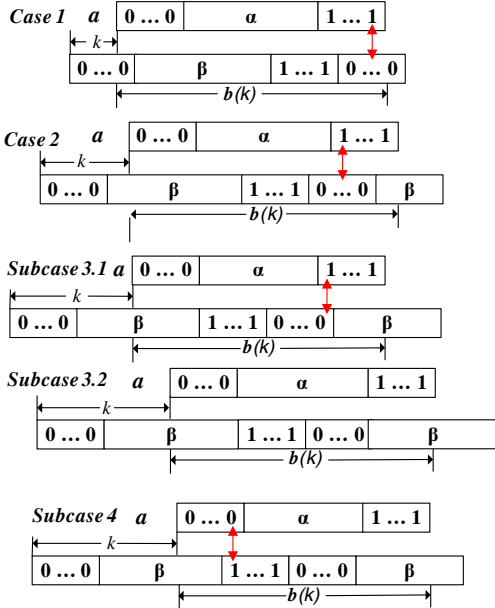
Figure 4: Illustration of the proof of Lemma 3.

Case 1: $k \in (0, l')$. As indicated by the arrow in Case 1 of Figure 4, it holds that $a_L = 1$ and $b_L(k) = 0$ where $L = l + 2l'$.

Case 2: $k \in [l', l+l')$. As indicated by the arrow in Case 2 of Figure 4, it holds that $a_{l+l'} = 1$ and $b_{l+l'}(k) = 0$.

Case 3: $k = l + l'$. We further distinguish two subcases:

- **Subcase 3.1:** l is odd. It holds that $l = 2l' - 1$. As indicated by the arrow in the Subcase 1 of Subcase 3.1 of Figure 4, it holds that $a_{l'} = 0$ and $b_{l'}(k) = 1$.

- **Subcase 3.2:** l is even. It holds that $l = 2l'$. As indicated by the arrow in the Subcase 2 of Subcase 3.2 of Figure 4, since $\mathbf{a} \neq \mathbf{b}$, $\mathbf{a} = \mathbf{1}||\mathbf{0}$ and $\mathbf{b} = \mathbf{1}||\mathbf{0}$ cannot hold simultaneously; There must exists l_0 such that $a_{l_0} \neq b_{l_0}(k)$.

Case 4: $k \in (l+l', l+2l')$. As indicated by the arrow in the Case 2 of Figure 4, it holds that $a_{l'} = 0$ and $b_{l'}(k) = 1$.

Noticing that $\alpha \neq \beta \implies \mathbf{a} \neq \mathbf{b}$, we thus conclude that $\mathbf{a} \neq \mathbf{b}(k), \ \forall k \in [0, l+2l')$. □

Step 2: Generating regular sequence

Denote the padded ID sequence as $\mathbf{o}_u$ for player u, the next step for each player is to generate a sequence $\mathbf{s}_u$ based on $\mathbf{o}_u$ such that for any two nodes a, b and any initial time offset t_a^0 and t_b^0, there always exist four time slots l_{ij} ($i, j \in \{0, 1\}$) such that $s_a^{l_{ij}}(t_a^0) = i$ and $s_b^{l_{ij}}(t_b) = j$. We denote such sequences $\mathbf{s}_u$ as *regular sequences*. In the following we develop an algorithm that can generate regular sequences.

Algorithm 2 Construct a regular sequence $\mathbf{s}_u$

Input: ID sequence $\mathbf{o}_u$ of L_o bits
Output: Regular sequence $\mathbf{s}_u$
  for $i = 1$ to L_o **do**
    switch o_u^i **do**
      case 1: expand o_u^i into eight bits 01010101
      case 0: expand o_u^i into eight bits 00110011
    end switch
  end for
  $\mathbf{s}_u \leftarrow$ the expanded sequence of $\mathbf{s}_u$

LEMMA 4. *The sequence generated by Algo 2 is regular.*

Step 3: Generating neighbor discovery schedule

In the last step, the neighbor discovery schedule is constructed as follows. Each node u hops across different channels $h \in \mathcal{N}$ and wakes up based on the following schedule[5]:

$$x_u^t = \begin{cases} h & t - hd_u \text{ is divisible by } 2Nd_u \pm 1, \\ 0 & \text{otherwise,} \end{cases}$$

where $x_u^t = h$ signifies that u wakes up on channel h in slot t while $x_u^t = 0$ indicates that u sleeps in the slot, Nd_u is chosen from the usable duty cycle set as analysed in sec. 5.4.

The above construction of $\mathbf{x}_u$ does not take into account the case where there exist two different channels h^c ($c = 0, 1$) such that $t - h^0 d_u$ is divisible by $2Nd_u - 1$ and $t - h^1 d_u$ by $2Nd_u + 1$. To resolve such conflict, let $t' = t\%L_s$, u operates on channel h^c if $s_u^{t'} = c$. We refer to the slots where u operates on channel h^c in case of conflict as type-c slots.

To intuitively see that the discovery is ensured between any pair of nodes a, b (the detailed proof is presented in the next subsection), note that if Nd_u belongs to the usable duty cycle set derived previously, i.e., at least one of $2Nd_a \pm 1$ is co-prime with at least one of $2Nd_b \pm 1$, discovery can be guaranteed for any initial time offset t_a^0 and t_b^0 because there always exist four time slots l_{ij} ($i, j \in \{0, 1\}$) such that $s_a^{l_{ij}}(t_a^0) = i$ and $s_b^{l_{ij}}(t_b) = j$ following the regularity of $\mathbf{s}_u$.

---
[5]To make the notation concise, we adopt the notation that $t - hd_u$ is divisible by $2Nd_u \pm 1$ denotes that $t - hd_u$ is divisible by $2Nd_u - 1$ or $2Nd_u + 1$ or both.

6.2 Discovery Delay Upper-bound

This subsection studies the theoretical performance of Mc-Dis in the multi-channel environment. In multi-channel case, the second metric on discovery diversity and the third metric on MTTFDD are applicable.

THEOREM 4 (WORST-CASE DISCOVERY DELAY). *If Nd_a and Nd_b belong to the usable duty cycle set, the MTTFDD between two nodes a and b is $O(L_s N^2 \max\{d_a^2, d_b^2\})$, specifically, $O(L_s N^2 d^2)$ if $d_a \simeq d_b \simeq O(d)$.*

PROOF. Without loss of generality, assume that $2Nd_a + 1$ is co-prime with $2Nd_b - 1$. It follows from The Chinese Remainder Theorem that before resolving conflicts (i.e., assume u can operate on two channels simultaneously), for any channel h, there exists $t_0 < (2Nd_a + 1)(2Nd_b - 1)$ such that $x_a^{t_0}(t_a^0) = x_b^{t_0}(t_b^0) = h$ and it holds that on slots $t_k = t_0 + k(2Nd_a + 1)(2Nd_b - 1)$, a can also discovery b before resolving conflicts. However, in realistic settings with conflicts, to ensure discovery, we need to show that there exists k such that a operates in type-1 slot at slot t_k while b operates in type-0 slot at slot t_k. To that end, noticing that $L_s = 4L_o$ is an even number and thus is co-prime with $(2Nd_a + 1)(2Nd_b - 1)$, there must exist $k < L_s$ such that $t_k \% L_s = l_0$, where l_0 denoted the bit index such that in $s_a^{l_0}(t_a^0) = 1$ and $s_b^{l_0}(t_b^0) = 0$. It follows from Lemma 4 that such l_0 exists. It follows from the construction of x_u that a operates in type-1 slot at slot t_k while b operates in type-0 slot at slot t_k, which leads to discovery. It further follows from $k < L_s$ that the MTTFDD is $O(L_s N^2 \max\{d_a^2, d_b^2\})$. □

The capability to achieve discovery on every channel within bounded delay significantly improves neighbor discovery robustness in wireless environment where channel conditions are unpredictable and may vary in both time and space.

6.3 Robustness of Mc-Dis against Asymmetrical Channel Perception

In previous analysis, we implicitly assume that a and b have the same channel perception, i.e., they have symmetrical knowledge on $\mathcal{N}$. In this subsection, we relax this assumption to investigate the scenario where each node u has its own perception on $\mathcal{N}$, denoted by $\mathcal{N}_u$, which is a subset of $\mathcal{N}$. Specifically, the channel perception asymmetry between a and b can be characterised at two levels:

- *Asymmetry on accessible channel set:* They have asymmetrical perceptions on the global channel set $\mathcal{N}$, i.e., $\mathcal{N}_a \neq \mathcal{N}_b$ and $\mathcal{N}_a \bigcap \mathcal{N}_b \neq \emptyset$;
- *Asymmetry on channel index:* They have asymmetrical perceptions on the channel index, i.e., channel $h \in \mathcal{N}$ is indexed h_a by a and h_b by b where $h_a \in \mathcal{N}_a$ and $h_b \in \mathcal{N}_b$ but $h_a \neq h_b$.

The following theorem established the performance of Mc-Dis in such context. The proof sketch is as follows (the detail is omitted due to space limit): Without loss of generality, assume that $2N_a d_a + 1$ is co-prime with $2N_b d_b - 1$. It follows from The Chinese Remainder Theorem that before resolving conflicts (i.e., assume u can operate on two channels), for any channel h indexed as h_a (h_b) by a (b), there exists $t_0 < (2N_a d_a + 1)(2N_b d_b - 1)$ such that $x_a^{t_0}(t_a^0) = h_a$ and $x_b^{t_0}(t_b^0) = h_b$. Then using the similar analysis as the proof of Theorem 4, we can show that the MTTFDD is $O(L_o \max\{N_a^2 d_a^2, N_b^2 d_b^2\})$.

THEOREM 5. *Mc-Dis under asymmetrical channel perceptions achieves the same MTTFDD as under symmetrical channel perceptions, i.e., within at most $O(L_s \max\{N_a^2 d_a^2, N_b^2 d_b^2\})$ (specifically, $O(L_s N^2 d^2)$ if $d_a \simeq d_b \simeq O(d)$ and $N_a \simeq N_b \simeq O(N)$) slots, the discovery between a and b occurs on each channel $h \in \mathcal{N}_a \bigcap \mathcal{N}_b$.*

Theorem 5 shows that Mc-Dis is robust against asymmetrical channel perceptions, either on the channel set or index.

6.4 Robustness of Mc-Dis against Slot Non-alignment and Arbitrary Clock Drift

In this subsection we study the effect of slot non-alignment caused by relative clock drift between the neighbor nodes.

We first briefly introduce the clock model. Each node is equipped with a local clock, which is a time measurement device composed of a hardware oscillator and an accumulator. Mathematically, consider two nodes a and b, we can express the local time at b, denoted as t_b, as a function of the local time of a, denoted as t_a, by the following formula

$$t_b(t_a) = \int_{t_0}^{t_a} \rho_{ab}(\tau) d\tau + t_b(t_0),$$

where $\rho_{ab}(\tau)$ denotes the frequency difference of the oscillator between a and b at time τ, $t_b(t_0)$ is the initial clock offset between them.

If a and b are ideally synchronised, it holds that $\rho_{ab}(\tau) = 1$ and $t_b(t_0) = 0$. In practice, $\rho_{ab}(\tau)$ may drift away from each other, as formalised in the following:

$$\rho_{ab} - \Delta\rho_{max} \leq \rho_{ab}(\tau) \leq \rho_{ab} + \Delta\rho_{max},$$

where $\Delta\rho_{max}$ is bounded by 10^{-6} in practice. Hence we can regard $\rho_{ab}(\tau)$ as a constant ρ_{ab} during the discovery process. Without loss of generality, we assume that the clock of b advances no slower than that of a, i.e., $\rho_{ab} \leq 1$.

When $\rho_{ab} = 1$, i.e., the clock difference between a and b remains $t_b(t_0)$, we distinguish the following two cases (to facilitate presentation, we normalize the slot duration of a):

- Case 1: $t_b(t_0) = k \in \mathbb{Z}$: this is the case with aligned slots addressed in previous analysis;
- Case 2: $t_b(t_0) = k + \delta$ with $k \in \mathbb{Z}$ and $\delta \in (-1/2, 1/2]$: the previous analysis can be directly adapted to this case, the difference being that instead of ensuring entire overlap, a discovery in this case is a partial overlap of time $1 - \delta$.

We now investigate the case where $\rho_{ab} < 1$, meaning that if we regard the slot duration of a as unit time, the slot duration of b is $\rho_{ab} < 1$. The following theorem establishes the discovery performance of Mc-Dis with arbitrary clock drift with $\rho_{ab} < 1$. Due to space limit, the proof is removed.

THEOREM 6. *Regard the slot of a as unit time, a and b can discover each other on each channel h within at most $O(\rho_{ab} L_s N^2 \max\{d_a^2, d_b^2\})$ time.*

The results obtained in this subsection, particularly Theorem 6, demonstrate that the discovery performance established in previous analysis holds even when the clocks of a and b drift away from each other for an arbitrary amount of time. In other words, Mc-Dis is robust against clock drift and slot non-alignment.

7. PERFORMANCE EVALUATION

In this section, we perform a set of simulations to evaluate Mc-Dis in several typical application scenarios, ranging from the synchronized single-channel case to the heterogeneous asynchronous and asymmetrical multi-channel case. We also conduct a comparative study between Mc-Dis and other major existing neighbor discovery protocols in duty-cycled networks by focusing on the capability to support heterogeneous duty cycles and the discovery delay.

7.1 Supported Duty Cycles

The first numerical experiment is a comparative analysis on the supported duty cycles in Disco, U-Connect, Searchlight and Mc-Dis. To that end, for each possible required duty cycle $1 \leq d \leq 100$, we study the relative error in supporting it, denoted as $\epsilon \triangleq |d' - d|/d$ where d' is the the closest duty cycle supported by the simulated protocol w.r.t. d. Noted that a smaller ϵ implies that the protocol can support more energy conservation levels with finer granularity. For Disco, in which the choice of prime numbers depends on the target discovery delay upper-bound, we configure the protocol by aligning the bound to Mc-Dis in order to provide a common baseline. For Searchlight, we set the smallest duty cycle unit as 2 to allow the finest duty cycle granularity.

The results are illustrated in Fig. 5(a). We make the following observations: (1) Searchlight has the worst performance on supporting duty cycles, because it restricts the duty cycles to a power-multiple of the smallest one, i.e., $2, 4, 8$ etc. As a natural consequence, when the required duty cycle goes away from the power-multiples, the related error increases significantly. Such power-multiple-based error trend can be demonstrated by the power-multiple gap between neighboring delay peaks in the figure. (2) Mc-Dis achieves the best performance with ϵ monotonously decreasing in d except for $d = 38$ which is the only duty cycle not supported, meaning that a duty cycle 37 or 39 should be used. The result confirms the design philosophy of Mc-Dis stated in Sec. 5. The highest error is around 7%, which confirms the analysis in Lemma 1. (3) The performance of Disco and U-Connect is between that of Searchlight and Mc-Dis. Compared to Mc-Dis, the performance variations are much more important. The cause of such variations is their limitation to only prime numbers. In Disco, despite the possibility of fine-tuning the choice of prime numbers, since this choice also impacts the discovery delay, it is difficult to strike a balance between the two metrics which may be contradictory to each other.

7.2 Performance in Single-channel Case

We now study the discovery performance of Mc-Dis in the single-channel case by comparing the worst-case discovery delay for the four protocols in three representative scenarios depending on the duty cycles of a and b: (1) both a and b have low duty cycles $d_a = 10$, $d_b = 12$; (2) both of them has high duty cycles $d_a = 50$, $d_b = 60$; (3) a has low duty cycle while b has high duty cycle. For the three scenarios, we simulate both the case where the slots of a and b are aligned (their clocks are not synchronized) and where the slots are not aligned. In the latter case, we adopt the proposition in [3] to let both nodes emit discovery beacons both at the beginning and at the end of each slot to increase the chance of discovery. The results are plotted Fig. 5(b) and 5(c). Throughout our simulations, each point represents the worst-case value of a number of independent simulation runs, with the required number of simulation runs calculated using "independent replications" [18].

From the results, we can see that the worst-case delay of the simulated protocols does not have significant difference, except for the case $(10, 60)$ where the delay of Searchlight outweighs the others. This is because approximating the duty cycle 10 by a power-multiple 16 has a pronounced negative impact on the worst-case discovery delay. Moreover, the performance with non-aligned slots outperforms that with aligned slots, due to the adopted optimisation technique to emit beacons both at the beginning and at the end of each slot. As a result, when the slots are not aligned, the probability of a partial overlap between two active slots is higher.

7.3 Performance in Multi-channel Case

We now evaluate Mc-Dis in the multi-channel case. Note that it is the only protocol supporting multiple channels. Specifically, we simulate the following two scenarios for a system of $N = 10$ channels:

- Both a and b have the same channel perception, i.e., $\mathcal{N}_a = \mathcal{N}_b = \mathcal{N}$. We simulate the sub-scenarios of both aligned and non-aligned slots for different N.
- a and b have asymmetrical channel perceptions and drifted slots. We further simulate three sub-scenarios: (1) There is only one common channel between them and $N_a = N_b = 3$; (2) There are $N_c = N/2$ common channels and $N_a = N_b = 8$; (3) The number of common channels N_c is randomly distributed in $[1, N]$ with random N_a, N_b supporting N_c.

From the simulation results in Fig. 6, we make the following observations: (1) As the system scales in terms of N, the discovery delay also increases. Moreover, we report that the delay increases squarely with the channel numbers, which is in accordance with the analytical results. (2) Discovery is achieved on each channel that both nodes can access, even in the case where a and b have asymmetrical channel perceptions and drifted slots. This property makes Mc-Dis especially adapted in the decentralised mobile applications with heterogeneous wireless nodes.

8. CONCLUSION

In this paper, we have investigated the heterogeneous neighbor discovery problem in multi-channel wireless networks. Our developed protocol *Mc-Dis* can achieve mutual discovery at minimal and bounded latency with full discovery diversity, even when the network nodes have asynchronous clocks and asymmetrical channel perceptions. As future work, we envision to integrate Mc-Dis into a multi-channel MAC protocol so as to provide a complete set of MAC primitives including neighbor discovery, neighbor table management and channel access coordination.

9. ACKNOWLEDGEMENTS

10. REFERENCES

[1] Sony ps vita - near. http://us.playstation.com/psvita.
[2] C. Arachchige, S. Venkatesan, and N. Mittal. An asynchronous neighbor discovery algorithm for cognitive radio networks. In *Proc. DySPAN*, 2008.
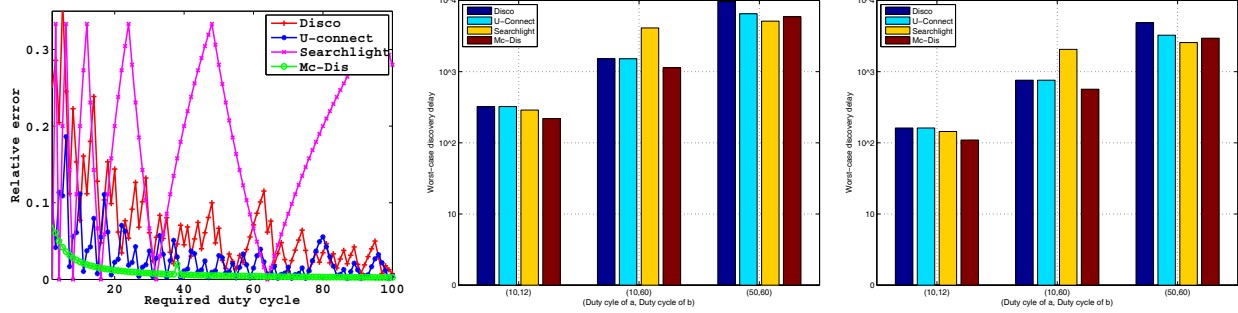
Figure 5: Performance comparison between Mc-Dis and major neighbor discovery protocols, single-channel case: (a) relative error as function of duty cycle; (b) worst-case discovery delay with aligned slots and (c) drifted slots. Mc-Dis achieves comparable delay while supporting significantly more duty cycles.
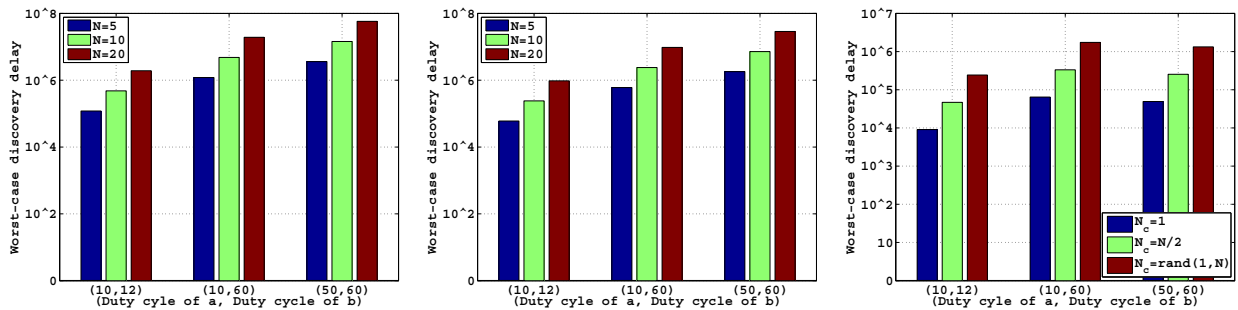

Figure 6: The worst-case discovery delay to full diversity: (a) symmetrical channel perception and aligned slots, (b) symmetrical channel perception and drifted slots, (c) asymmetrical channel perceptions and drifted slots.

[3] M. Bakht, M. Trower, and R. H. Kravets. Searchlight: won't you be my neighbor. In *Proc. Mobicom*, 2012.

[4] P. Dutta and D. Culler. Practical asynchronous neighbor discovery and rendezvous for mobile sensing applications. In *Proc. Sensys*, 2008.

[5] R. Ganti, F. Ye, and H. Lei. Mobile crowdsensing: current state and future challenges. *IEEE Comm. Mag.*, 49(11):32–39, 2011.

[6] C. Godsil and G. Royle. *Algebraic Graph Theory*. Springer, 2001.

[7] A. Kandhalu, K. Lakshmanan, and R. Rajkumar. U-connect: a low-latency energy-efficient asynchronous neighbor discovery protocol. In *Proc. IPSN*, 2010.

[8] S. Krishnamurthy, N. Mittal, R. Chandrasekaran, and S. Venkatesan. Neighbor discovery in multi-receiver cognitive radio networks. *Intl J. Computers and Applications (IJCA)*, 31(1):50–57, 2009.

[9] S. Lai, B. Ravindran, and H. Cho. Heterogenous quorum-based wakeup scheduling in wireless sensor networks. *IEEE Transactions on Computers*, 59(11), 2010.

[10] M. J. McGlynn and S. A. Borbash. Birthday protocols for low energy deployment and flexible neighbor discovery in ad hoc wireless networks. In *Proc. MobiHoc*, 2001.

[11] E. Miluzzo, N. D. Lane, K. Fodor, R. Peterson, H. Lu, M. Musolesi, S. B. Eisenman, X. Zheng, and A. T. Campbell. Sensing meets mobile social networks: the design, implementation and evaluation of the cenceme application. In *Proc. SenSys*, 2008.

[12] N. Mittal, S. Krishnamurthy, R. Chandrasekaran, S. Venkatesan, and Y. Zeng. On neighbor discovery in multi-channel cognitive radio networks. *J. Par. Dist. Comp. (JPDC)*, 69(7):623–637, 2009.

[13] N. Mittal, Y. Zeng, S. Venkatesan, and R. Chandrasekaran. Randomized distributed algorithms for neighbor discovery in multi-hop multi-channel heterogeneous wireless networks. In *Proc. ICDCS*, 2013.

[14] I. Niven, H. S. Zuckerman, and H. L. Mongomery. *An Introduction to the Theory of Numbers*. John Wiley & Sons, 1991.

[15] 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. Infocom*, 2002.

[16] S. Vasudevan, J. Kurose, and D. Towsley. On neighbor discovery in wireless networks with directional antennas. In *Proc. Infocom*, 2005.

[17] S. Vasudevan, D. Towsley, D. Goeckel, and R. Khalili. Neighbor discovery in wireless networks and the coupon collector's problem. In *Proc. MobiCom*, 2009.

[18] W. Whitt. The efficiency of one long run versus independent replications in steady-state simulation. *Management Science*, 37(6):645–666, 1991.

[19] W. Zeng, S. Vasudevan, X. Chen, B. Wang, A. Russell, and W. Wei. Neighbor discovery in wireless networks with multipacket reception. In *Proc. MobiHoc*, 2011.

[20] Y. Zeng, N. Mittal, S. Venkatesan, and R. Chandrasekaran. Fast neighbor discovery with lightweight termination detection in heterogeneous cognitive radio networks. In *Proc. 9th International Symposium on Parallel and Distributed Computing (ISPDC)*, 2010.

[21] D. Zhang, T. He, Y. Liu, Y. Gu, F. Ye, R. K. Ganti, and H. Lei. Acc: generic on-demand accelerations for neighbor discovery in mobile applications. In *Proc. SenSys*, 2012.

[22] D. Zhang, T. He, Y. Liu, F. Ye, R. K. Ganti, and H. Lei. Eqs: Neighbor discovery and rendezvous maintenance with extended quorum system for mobile sensing applications. In *Proc. ICDCS*, 2012.

[23] Z. Zhang and B. Li. Neighbor discovery in mobile ad hoc self-configuring networks with directional antennas algorithms and comparisons. *IEEE Transactions on Wireless Communications*, 7(5), 2008.

[24] R. Zheng, J. C. Hou, and L. Sha. Asynchronous wakeup for ad hoc networks. In *Proc MobiHoc*, 2003.