TKN Telecommunication Networks Group

Technische Universität Berlin

Telecommunication Networks Group

Optimized Asynchronous Passive Multi-Channel Discovery of Beacon-Enabled Networks

Niels Karowski, Konstantin Miller

{karowski, miller}@tkn.tu-berlin.de

Berlin, March 2015

TKN Technical Report TKN-15-002

TKN Technical Reports Series

Editor: Prof. Dr.-Ing. Adam Wolisz

Abstract

Neighbor discovery is a fundamental task for wireless networks deployment. It is essential for setup and maintenance of networks and is typically a precondition for further communication. In this work we focus on passive discovery of networks operating in multi-channel environments, performed by listening for periodically transmitted beaconing messages. It is well-known that performance of such discovery approaches strongly depends on the structure of the adopted Beacon Interval (BI) set, that is, set of intervals between individual beaconing messages. However, although imposing constraints on this set has the potential to make the discovery process more efficient, there is demand for high-performance discovery strategies for BI sets that are as general as possible. They would allow to cover a broad range of wireless technologies and deployment scenarios, and enable network operators to select BI's that are best suited for the targeted application and/or device characteristics.

In the present work, we introduce a family of novel low-complexity discovery algorithms that minimize both the Expected Mean Discovery Time (EMDT) and the makespan, for a quite general family of BI sets. Notably, this family of BI sets completely includes BI's supported by IEEE 802.15.4 and a large part of BI's supported by IEEE 802.11. Furthermore, we present another novel discovery algorithm, based on an Integer Linear Program (ILP), that minimizes EMDT for arbitrary BI sets.

In addition to analytically proving optimality, we numerically evaluate the proposed algorithms using different families of BI sets and compare their performance with the passive scan of IEEE 802.15.4 w.r.t. various performance metrics, such as makespan, EMDT, energy usage, etc.

Contents

Copyright at Technische Universität Berlin.
All rights reserved.

Chapter 1

Introduction

We are currently at the brink of a new era of computing driven by rapid augmentation of physical objects around us with computational and wireless communication capabilities. The resulting network of smart objects that interact and exchange information without direct human intervention, the so-called Internet of Things (IoT), can offer deep real-time awareness and control of the physical environment and serve as foundation for novel applications in a wide range of domains. It is estimated that 25 [1] to 50 [4] billion devices and objects will be connected to the Internet by 2020. Since the vast majority of these devices will communicate wirelessly, this development raise the demand for efficient neighbor discovery approaches. Discovering neighbor networks in a wireless network environment is typically the first step of communication initialization. In addition, if performed periodically, it helps avoiding conflicts by, e.g., selecting a less occupied channel. Note that since most devices will be battery powered, neighbor discovery must be performed in an energy-efficient way.

This paper focuses on the problem of discovering neighbors in a fast and energy-efficient manner by passively listening to periodically transmitted beaconing messages of neighbors that are agnostic to the discovery process. Neighbors may operate on different channels using various BI's based on their targeted application, used hardware, and operational state (e.g., dynamically adapting BI's to current battery level). Since devices are typically not synchronized and have no a priori knowledge about their environment, designing schedules determining when to listen, on which channel, and for how long, that allow for a fast discovery of all neighbors, is a challenging task.

State-of-the-art beacon-enabled wireless network technologies include, e.g., IEEE 802.11 [3] and IEEE 802.15.4 [2]. With these technologies, beacon messages are transmitted periodically for management and time synchronization purposes. The idea of the presented discovery approaches is to use periodic transmission of beacon frames for the discovery process. In contrast, discovery approaches requiring control over the transmission of discovery messages might be in conflict with the deployed Media Access Control (MAC) protocol, while reusing beacon messages allows for more flexibility in working with a broad range of technologies.

It is well-known that performance of discovery approaches strongly depends on the structure of the BI set. Although restricting the latter facilitates the discovery process, it is necessary to develop discovery strategies that are as nonrestrictive as possible, in order to cover a broad range of existing wireless technologies, and to allow network operators to select BI's that are best suited for the targeted application and/or device characteristics.

Our previous work [6, 7] focused on discovery of IEEE 802.15.4 networks, where BI's are restricted to the form $\tau \cdot 2^{BO}$, where τ is the duration of a superframe, and Beacon Order (BO) is a parameter taking values between 0 and 14. In contrast, in the present work, we focus

Copyright at Technische Universität Berlin.
All rights reserved.

on developing efficient algorithms for broader families of BI sets, including arbitrary BI sets that do not have any restrictions.

In particular, we develop a family of novel low-complexity algorithms that "greedily" maximize the number of discovered networks independently in each subsequent time slot. We identify an important family of BI sets, where each BI is an integer multiple of all smaller BI's, and prove that for BI sets from this family, greedy algorithms are optimal w.r.t. the EMDT and the makespan. Notably, the identified family completely includes BI sets supported by IEEE 802.15.4 and a large part of BI sets supported by IEEE 802.11.

In addition, we present a novel discovery approach, based on an ILP, that minimizes EMDT for arbitrary BI sets. Although quite attractive due to the broad range of supported BI's, this approach has high computational complexity and memory consumption, such that its usage is restricted to offline computations, and to network environments with a limited number of channels and small BI's.

In addition to analytically proving optimality of the proposed discovery algorithms, we numerically evaluated them using different families of BI sets, and compare their performance with the passive scan, specified as part of the IEEE 802.15.4 standard, w.r.t. various performance metrics, such as makespan, EMDT, energy usage, etc.

Finally, we discuss the strong impact the structure of permitted BI sets has on the performance of discovery approaches, and provide recommendations for BI selection that supports efficient neighbor discovery. This recommendation might be useful, on the one hand, for the development of novel wireless communication based technologies using periodic beacon frames for management or synchronization purposes, and, on the other hand, for the deployment of existing technologies that support a wide range of BI's, such as, e.g., IEEE 802.11.

In summary, our main contributions are as follows.

- A family of low-complexity discovery algorithms that minimize EMDT **and** makespan for the broad family of BI sets where each BI is an integer multiple of all smaller BI's. Notably, this family completely includes BI's supported by the standard IEEE 802.15.4 and a large part of BI sets supported by IEEE 802.11.
- An ILP-based approach to computing listening schedules minimizing EMDT for arbitrary sets of BI's.
- A recommendation on the selection of BI sets to support the discovery process.

The rest of this paper is structured as follows. Section 2 presents our system model, accompanied by basic definitions and preliminary results. In Section 3 we present the families of BI sets studied in the present work. Section 4 introduces the notion of EMDT and establishes an upper bound on the time required to minimize it. Section 5 introduces the developed discovery strategies, accompanied by analytical results on their optimality. Section 6 presents evaluation settings and results. In Section 7, we provide recommendations on the selection of BI's supporting the discovery process. Finally, Section 8 concludes this paper and outlines future work.

Copyright at Technische Universität Berlin.
All rights reserved.

Chapter 2

System Model and Basic Definitions

We consider a set of networks N, operating on a set of channels C. Each network $\nu \in N$ is assigned a fixed channel $c_\nu \in C$. Each network announces its presence using beacon signaling messages (beacons, in the following) that, e.g., are sent by a selected device acting as network coordinator. We assume that beacons are sent periodically with a fixed interval of $b_\nu \cdot \tau$ seconds, $b_\nu \in B$, where $B \subset \mathbb{N}^+$ is a finite set of beacon intervals, and τ is a technology dependent parameter. The set of BI's B might be determined by, e.g., the choice of communication technology or by specifications provided by network operators. A characterization of BI sets studied in the present work will be provided in Chapter 3.

We divide time into slots of length τ, and number them starting with 1, such that i-th time slot contains the time period $[(i-1)\tau, i\tau]$. The parameter τ is typically determined by the choice of technology. For IEEE 802.15.4, τ might be the duration of a base superframe, which is approximately 15.36 ms (960 symbols) when operating in the 2.4 GHz frequency band. In case of IEEE 802.11, τ might be the duration of a Time Unit (TU), which equals 1024 μs. We assume that the maximum beacon transmission time (time required to send one beacon) is smaller than τ. We denote the offset, that is, the first time slot, in which network ν sends a beacon, by $\delta_\nu \leq b_\nu$. We denote the set of beaconing time slots of a network ν by $\mathcal{T}_\nu = \{\delta_\nu + i \cdot b_\nu\}_{i \geq 0}$. We call the resulting set of (channel, time slot) pairs, $\mathcal{B}_\nu = \{c_\nu\} \times \mathcal{T}_\nu$, the beacon schedule of network ν.

With the given notation, each network ν is assigned a tuple $(c_\nu, b_\nu, \delta_\nu)$, which we call a network configuration. Note that assigning multiple networks the same network configuration does not necessarily lead to beacon collisions due to the assumption that the maximum beacon transmission time is shorter than τ. With IEEE 802.15.4, e.g., default beacon size when operating in the 2.4 GHz frequency band is 38 symbols, as compared to the time slot duration $\tau = 960$ symbols.

We denote the set of possible network configurations for a given BI set B and a given set of channels C by $K_{BC} = \{(c, b, \delta) \mid c \in C,\, b \in B,\, \delta \in \{1, \ldots, b\}\}$. We define a network environment to be a function $E : N \mapsto K_{BC}$ assigning each network a network configuration. Slightly abusing notation, in the following, we use index ν to refer to a particular network, and index κ to refer to a particular network configuration. Thus, we denote a network configuration $\kappa \in K_{BC}$ by $\kappa = (c_\kappa, b_\kappa, \delta_\kappa)$. Analogously, we define $\mathcal{T}_\kappa$ and $\mathcal{B}_\kappa$ as the set of beaconing time slots and the beacon schedule of any network using network configuration κ.

For a time slot t and a BI b, $\delta_b(t) = (t \bmod b) + 1$ shall denote the unique offset such that a network with configuration $(c, b, \delta_b(t))$, $c \in C$, transmits its beacon in time slot t. Observe that $\delta_b(t)$ has periodicity b, that is, $\delta_b(t) = \delta_b(t + b)$ for each t, b. Consequently, vector function $\delta(t) = (\delta_b(t),\, b \in B)$ has periodicity $LCM(B)$, that is, $\delta(t) = \delta\left(t + LCM(B)\right)$ for

Copyright at Technische Universität Berlin.
All rights reserved.

each t.

We assume that initially each device knows its own network configuration, the set of available BI's B and the set of channels C. In addition, it might know probabilities $(p_\kappa,\ \kappa \in K_{BC})$ that a neighbor network might use configuration κ. Equipped with this knowledge, in order to detect neighbor networks, a device performs neighbor discovery by selecting time slots during which it listens on particular channels in order to overhear beacons transmitted by neighbors, starting with time slot 1. We call the resulting set of (channel, time slot) pairs a listening schedule, denoted by $\mathcal{L} \subset C \times \mathbb{N}$. Since we assume that devices cannot simultaneously listen on multiple channels, we demand $c \neq c' \Rightarrow t \neq t'$ for all $(c,\ t)\,,(c',\ t') \in \mathcal{L}$.

In a typical network environment, not all configurations will be used, while, at the same time, some configurations might be used by more than one network. We call an environment complete, if each configuration is used by exactly one network.

Definition 1 (Complete environment)**.** *For a BI set $B \subset \mathbb{N}^+$ and a set of channels C, an environment $E : N \mapsto K_{BC}$ is called complete if and only if E is bijective (for each $\kappa \in K_{BC}$ there exists exactly one $\nu \in N$ with $\kappa = (c_\nu, b_\nu, \delta_\nu)$).*

Under ideal conditions, when beacons are never lost and devices are not mobile, if a schedule $\mathcal{L}$ contains at least one element from the beacon schedule $\mathcal{B}_\kappa$ for each configuration $\kappa \in K_{BC}$, it allows to discover all neighbor networks in a complete environment. We call such a schedule *complete.*

Definition 2 (Complete schedule)**.** *For a BI set $B \subset \mathbb{N}^+$ and a set of channels C, a schedule $\mathcal{L} \subset C \times \mathbb{N}$ is called complete if and only if $\mathcal{L} \cap \mathcal{B}_\kappa \neq \emptyset,\ \forall\, \kappa \in K_{BC}$.*

For a complete schedule $\mathcal{L}$, we denote by $T_\nu(\mathcal{L}) = \min\{t \in \mathcal{T}_\nu \mid (c_\nu,\ t) \in \mathcal{L}\}$ the discovery time of network ν. Similarly, we denote by $T_\kappa(\mathcal{L}) = \min\{t \in \mathcal{T}_\kappa \mid (c_\kappa,\ t) \in \mathcal{L}\}$ the discovery time of all networks operating with configuration κ. Whenever the considered schedule is clear from the context, we might simply write T_ν or T_κ. An important performance metric for a listening schedule is the time it requires to detect all neighbor networks. We call this time the makespan of a schedule.

Definition 3 (Makespan of a schedule)**.** *For a BI set $B \subset \mathbb{N}^+$, and a set of channels C, we call the time slot of a complete schedule $\mathcal{L}$ during which the last configuration is detected the makespan of $\mathcal{L}$ and denote it by $T_\mathcal{L}$. That is, $T_\mathcal{L} = \max_{\kappa \in K_{BC}} T_\kappa(\mathcal{L})$.*

The following proposition and corollary establish a lower bound on the makespan.

Proposition 4. *For a BI $b \in B \subset \mathbb{N}^+$, an offset $\delta \in \{1, \ldots, b\}$, a set of channels C, and a complete schedule $\mathcal{L}$, the earliest time slot until which all configurations $\{(c, b, \delta) \mid c \in C\}$ can be discovered is $b(|C| - 1) + \delta$.*

Proof. Note that $|\{(c, b, \delta) \mid c \in C\}| = |C|$. Since each configuration in this set has its beacons on a different channel, the number of time slots that have to be scanned cannot be smaller than $|C|$. Observe that the earliest time slot when this number of corresponding time slots can be reached is $b(|C| - 1) + \delta$, proving the claim. □

Copyright at Technische Universität Berlin.
All rights reserved.

Corollary 5. *For an arbitrary set of BI's $B \subset \mathbb{N}^+$, a set of channels C, and a complete schedule $\mathcal{L}$, it holds $T_\mathcal{L} \geq \max(B) \cdot |C|$. We call complete schedules with makespan $\max(B) \cdot |C|$ makespan-optimal.*

Proof. The claim is a direct consequence of Proposition 4, with $b = \delta = \max(B)$. □

In addition to minimizing makespan, it is often desirable to minimize the Mean Discovery Time (MDT) of a schedule, defined in the following.

Definition 6 (MDT of a schedule)**.** *For a set of networks N, a BI set $B \subset \mathbb{N}^+$, a set of channels C, and a network environment E, the MDT of a complete schedule $\mathcal{L}$ is given by*

$$\frac{1}{|N|} \sum_{\nu \in N} T_\nu (\mathcal{L}) \ .$$

For a complete environment, MDT can also be computed as

$$\frac{1}{|N|} \sum_{\nu \in N} T_\nu (\mathcal{L}) = \frac{1}{|K_{BC}|} \sum_{\kappa \in K_{BC}} T_\kappa (\mathcal{L}) = \frac{1}{|C||B| \sum_{b \in B} b} \sum_{\kappa \in K_{BC}} T_\kappa (\mathcal{L}) \ .$$

Typically, however, a device does not know its network environment (otherwise, it would not have to perform a discovery) so that MDT cannot be computed and optimized a priori. Nevertheless, a device still can minimize the expected value of the MDT given probabilities that a neighbor network is using configuration $\kappa \in K_{BC}$. We will address this question in Chapter 4.

For complete environments, we define a special type of schedules that we call recursive and that we will use to prove optimality of the proposed discovery strategies in the following chapters.

Definition 7 (Recursive schedule)**.** *In a complete environment with BI set $B \subset \mathbb{N}^+$, and a set of channels C, a schedule $\mathcal{L} \subset C \times \mathbb{N}$ is called recursive if and only if*

- *for any $t \in [1, \max(B) \cdot |C|]$ there exists a $c \in C$ such that $(c, t) \in \mathcal{L}$ (no idle slots)*
- *for any $b \in B$, and for any $t, t' \in [1, b \cdot |C|]$ with $\delta_b(t) = \delta_b(t')$, if (c, t), $(c', t') \in \mathcal{L}$ then $c \neq c'$ (no redundant scans)*

The following proposition provides a characterizing property of recursive schedules.

Proposition 8. *In a complete environment with a BI set $B = \{b_1, \ldots, b_n\} \subset \mathbb{N}^+$, $b_i < b_j$ for $i < j$, and a set of channels C, a schedule $\mathcal{L} \subset C \times \mathbb{N}$ is called recursive if and only if each scan $(c, t) \in \mathcal{L}$, with $t \in [1, b \cdot |C|]$, results in the discovery of network configuration $(c, b, \delta_b(t))$. Alternatively, a schedule is recursive if and only if in each time slot $t \in [1, b_i \cdot |C|]$, for a $b_i \in B$, it discovers at least $n - i + 1$ configurations.*

Proof. Follows directly from the definition of a recursive schedule. □

Recursive schedules have a compelling property that they are always complete, makespan-optimal, and MDT-optimal, as stated in the following Corollary. In the following chapters, we will define families of BI sets where recursive schedules always exist, and present algorithms for their efficient computation.

Copyright at Technische Universität Berlin.
All rights reserved.

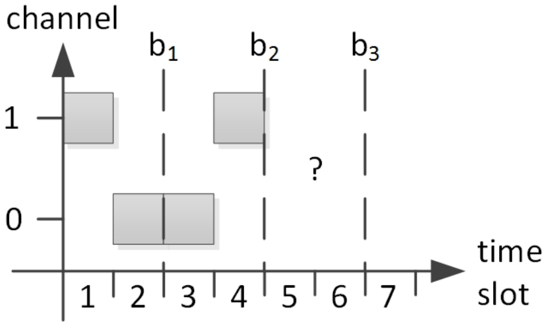


Figure 2.1: Example of the non-existence of a structured schedule showing the beginning of the listening schedule depicted by gray boxes for $B = \{1, 2, 3\}$ and $|C| = 2$.

Corollary 9. *In a complete environment with a BI set $B \subset \mathbb{N}^+$, a recursive schedule is complete, makespan-optimal, MDT-optimal, and maximizes the number of configurations detected until each time slot, for each $B' \subset B$.*

Proof. Follows from Proposition 8. □

It is worth noting that a recursive schedule does not always exist. Consider the example illustrated in Figure 2.1, with $B = \{1, 2, 3\}$ and $|C| = 2$. Vertical dashed lines indicate the time slots until all configurations (c, b_i, δ) for a BI b_i have to be completely discovery on each channel $c \in C$ in order for the schedule to be recursive. Consequently, gray boxes indicate channels that have to be scanned during the first 4 time slots (uniquely determined up to swapping channel 0 and 1). Observe that a recursive listening schedule for this example does not exist due to the fact that it would have to discover all remaining network configurations using BI b_3 during only two time slots 5 and 6. However, there are three remaining configurations: $(0, 3, 1)$, $(1, 3, 2)$, and $(1, 3, 3)$.

Finally, we would like to remark that the analytical optimality results in the following are obtained under three idealizing assumptions.

- Switching between channels is performed instantaneously (switching time is 0).
- Beacon transmission/reception time is 0.
- There are no beacon losses.

It is ongoing work to evaluate the performance of the proposed algorithms by means of simulations and experiments in real network environments.

Table 2.1 provides a summary of all defined parameters.

Copyright at Technische Universität Berlin.
All rights reserved.

Table 2.1: Summary of the defined parameters

Name	Description
General	
τ	Time slot duration
$C \subset \mathbb{N}^+$	Set of channels
$B \subset \mathbb{N}^+$	Set of BI's
$LCM(B)$	Least common multiple of a set B
Network	
N	Set of networks
$\nu \in N$	Network ν
$c_\nu \in C$	Fixed operating channel of network ν
$b_\nu \in B$	BI of network ν
$\delta_\nu \in \{1, \ldots, b_\nu\}$	Beacon offset of network ν
$(c_\nu, b_\nu, \delta_\nu)$	Configuration of network ν
$\mathcal{T}_\nu = \{\delta_\nu + i \cdot b_\nu\}_{i \geq 0}$	Set of beaconing time slots of network ν
$\mathcal{B}_\nu = \{c_\nu\} \times \mathcal{T}_\nu$	Beacon schedule of network ν
$T_\nu(\mathcal{L}) = \min\{t \in \mathcal{T}_\nu \mid (c_\nu, t) \in \mathcal{L}\}$	Discovery time of network ν, given listening schedule $\mathcal{L}$
Network configuration	
$K_{BC} = \{(c, b, \delta) \mid c \in C, b \in B, \delta \in \{1, \ldots, b\}\}$	Set of possible network configurations for a given BI set B and a given set of channels C
$\kappa = (c_\kappa, b_\kappa, \delta_\kappa) \in K_{BC}$	Network configuration κ using BI b_κ, channel c_κ and offset δ_κ
$\mathcal{T}_\kappa = \{\delta_\kappa + i \cdot b_\kappa\}_{i \geq 0}$	Set of beaconing time slots of networks operating with configuration κ
$\mathcal{B}_\kappa = \{c_\kappa\} \times \mathcal{T}_\kappa$	Beacon schedule of networks operating with configuration κ
$T_\kappa(\mathcal{L}) = \min\{t \in \mathcal{T}_\kappa \mid (c_\kappa, t) \in \mathcal{L}\}$	Discovery time of all networks operating with configuration κ
Listening schedule	
$\mathcal{L} \subset C \times \mathbb{N}$	Listening schedule consisting of a sequence of (channel, time slot) pairs
$T_\mathcal{L} = \max_{\kappa \in K_{BC}} T_\kappa(\mathcal{L})$	Makespan of a listening schedule $\mathcal{L}$

Copyright at Technische Universität Berlin.
All rights reserved.

Chapter 3

Families of Beacon Interval Sets

In this section we define several families of BI sets, studied in the present work. They allow us to formulate properties of developed discovery strategies as functions of BI sets. The defined families are characterized in the following table and illustrated in Figure 3.1.

$\mathbb{F}_1$	This is the most general family of BI sets that contains any finite subset of $\mathbb{N}^+$.
$\mathbb{F}_2$	Family $\mathbb{F}_2 \subset \mathbb{F}_1$ includes all BI sets, for which the maximum element equals the LCM of the set: $\max(B) = LCM(B)$, that is, the maximum beacon interval is an integer multiple of all other beacon intervals in the set.
$\mathbb{F}_3$	For any BI set $B \in \mathbb{F}_3 \subset \mathbb{F}_2$, we demand that $\max(B') = LCM(B')$ holds for any subset $B' \subseteq B$. An equivalent formulation is that for any $b, b' \in B \in \mathbb{F}_3$ with $b < b'$ there exists an $\alpha \in \mathbb{N}^+$ such that $b' = \alpha \cdot b$. That is, any beacon interval is an integer multiple of any smaller beacon interval in the set.
$\mathbb{F}_4$	Family $\mathbb{F}_4 \subset \mathbb{F}_3$ includes all sets whose elements are powers of the same base, potentially multiplied by an common coefficient. That is, for a $B = \{b_1, \ldots, b_n\} \in \mathbb{F}_4$ we demand that there exist $k, c \in \mathbb{N}^+$ and $e_1, \ldots, e_n \in \mathbb{N}$ such that $b_i = kc^{e_i}$, $\forall\, i \in \{1, \ldots, n\}$.
$\mathbb{F}_{\text{IEEE 802.15.4}}$	This family contains all sets defined by the IEEE 802.15.4 standard. All elements $b \in B \in \mathbb{F}_{\text{IEEE 802.15.4}}$ must have the form $b = 2^{BO}$, where BO is a network parameter called the beacon order that can be assigned a value between 0 and 14. Possible intervals lie in the range between approx. 15.36 ms and 252 s.
$\mathbb{F}_{\text{IEEE 802.11}}$	The family of BI sets allowed by the IEEE 802.11 standard contains arbitrary sets such that each element $b \in B \in \mathbb{F}_{\text{IEEE 802.11}}$ can be represented by a 16 bit field, that is, $b \in \left[1,\, 2^{16} - 1\right]$.

An important property of the family of BI sets $\mathbb{F}_3$ is that in complete network environments using BI sets from $\mathbb{F}_3$ there always exists a recursive schedule, as shown by the following proposition.

Proposition 10. *In a complete environment with a BI set $B \in \mathbb{F}_3$, and a set of channels C, a recursive schedule always exists.*

Copyright at Technische Universität Berlin.
All rights reserved.

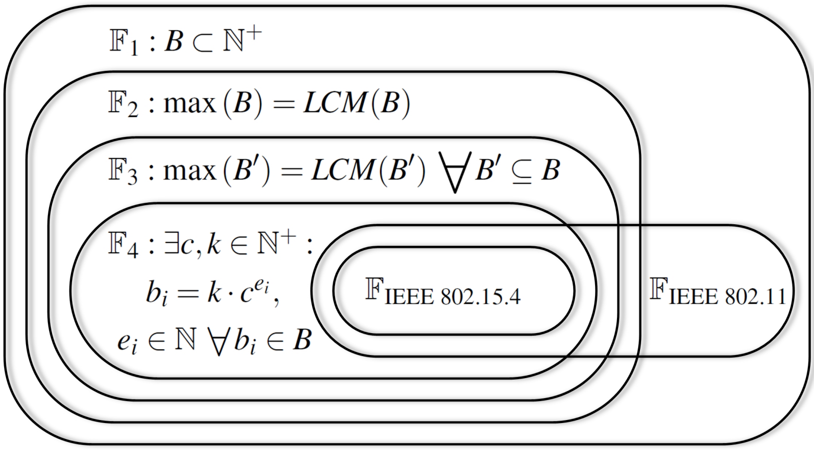


Figure 3.1: Studied families of BI sets.

Proof. To prove the claim we will use the characteristic property of recursive schedules from Proposition 8. Assume w.l.o.g. $B = \{b_1, \ldots, b_n\}$ with $b_i < b_j$ for $i < j$. Observe that it is possible to discover one configuration $(c, b_1, \delta_{b_1}(t))$ in each time slot $t \in [1,\, b_1 \cdot |C|]$, since for each offset $\delta \in \{1, \ldots, b_1\}$ there are $|C|$ time slots in the interval $[1,\, b_1 \cdot |C|]$ where this offset occurs, and since the sets of time slots for individual offsets to not intersect.

Further, observe that in $\mathbb{F}_3$, each discovery of a configuration $(c, b_i, \delta_{b_i}(t))$ results in a discovery of $\left(c, b_j, \delta_{b_j}(t)\right)$ for each $j \in \{i+1, \ldots, n\}$. Consequently, by induction, for each $b_i \in B$, one configuration (c, b_i, δ) can be discovered in each time slot $t \in [1,\, b_{i-1} \cdot |C|]$, while, analogously to the argumentation above, one of the remaining configurations can be discovered in each of the time slots $t \in [b_{i-1} \cdot |C| + 1,\, b_i \cdot |C|]$. $\square$

In the following, w.l.o.g., we only consider BI sets whose Greatest Common Divisor (GCD) is 1. Indeed, please observe that a listening schedule for a BI set with GCD $d \neq 1$ is equivalent to a listening schedule for the transformed BI set where each element is divided by d, and τ is substituted by $\tau' = \tau \cdot d$. This transformation allows to reduce the computational complexity, especially for ILP-based approaches.

Copyright at Technische Universität Berlin.
All rights reserved.

Chapter 4

Preliminaries

A desired goal for a discovery strategy is to minimize the Mean Discovery Time (MDT), where mean is taken over all networks in a particular environment. Since, however, a device does not a priori know which network configurations are present in its environment, this problem is not solvable. Nevertheless, it is possible to minimize the expected value for MDT, called Expected Mean Discovery Time (EMDT) in the following, given the assumption that certain network configurations are picked by individual networks with certain probabilities. In the following, we assume uniform probabilities for channels, BI's, and BI offsets. In the next two sections we establish a formula for EMDT under the assumption of a uniform distribution. We then proceed to establishing an upper bound on the makespan of EMDT-optimal schedules, which will help us to evaluate performance of discovery algorithms proposed in Chapter 5.

4.1 Expected Mean Discovery Time (EMDT)

Since a device does not a priori know which network configurations are used by its neighbors, it is not possible to design a listening schedule that minimizes MDT. However, in the absence of this information, a reasonable assumption for a device performing a discovery is that of a uniform distribution of configurations. It can then follow a discovery strategy that allows to minimize the expected value for MDT, the EMDT. Note that while mean value and expected value are synonyms, to improve readability, we use the term Mean Discovery Time (MDT) to refer to the mean value taken over neighbor networks in a particular environment (see Definition 6), while we use the term Expected Mean Discovery Time (EMDT) to refer to the expected value of MDT's taken over instances of network environments.

To be more precise, we assume probabilities $P = (p_\kappa, \kappa \in K_{BC})$ that a particular combination of channel, BI, and offset are used by a neighbor network, where p_κ are defined as follows. We assume that each channel and each BI have equal probability to be selected by a neighboring network, and that all offsets feasible for a particular BI have equal probability to be selected by a network using this BI. Thus, a network configuration $\kappa \in K_{BC}$, $\kappa = (c_\kappa, b_\kappa, \delta_\kappa)$, has probability $p_\kappa = \frac{1}{b_\kappa |B||C|}$ to be used by a neighbor network $\nu \in N$.

In the following proposition, we compute EMDT and show that for uniform probabilities, EMDT equals MDT given a complete environment.

Proposition 11. *For a set of networks N, a set of BI's $B \in \mathbb{F}_1$, a set of channels C, a complete listening schedule $\mathcal{L}$, and probabilities $p_\kappa = \frac{1}{b_\kappa |B||C|}$ that a network configuration*

Copyright at Technische Universität Berlin.
All rights reserved.

$\kappa \in K_{BC}$ is used by a network $\nu \in N$, EMDT is given by

$$E\left[\frac{1}{|N|}\sum_{\nu\in N} T_\nu\left(\mathcal{L}\right)\right] = \frac{1}{|B|\,|C|}\sum_{\kappa\in K_{BC}}\frac{1}{b_\kappa}T_\kappa\left(\mathcal{L}\right)\ .$$

Note that EMDT equals MDT for a complete environment

Proof. First, assume that the number of networks $|N|$ is fixed. Let $N_\kappa \subseteq N$ be the subset of networks using configuration $\kappa \in K_{BC}$. Observe that

$$E\left[\frac{1}{|N|}\sum_{\nu\in N} T_\nu\left(\mathcal{L}\right)\right] = E\left[\sum_{\kappa\in K_{BC}}\frac{|N_\kappa|}{|N|}T_\kappa\left(\mathcal{L}\right)\right] = \sum_{\kappa\in K_{BC}}\frac{E\left[|N_\kappa|\right]}{|N|}T_\kappa\left(\mathcal{L}\right)\ .$$

In order to calculate $E\left[|N_\kappa|\right]$, we have to calculate $P\left[|N_\kappa| = n\right]$. Since we assumed $p_\kappa = \frac{1}{b_\kappa|B||C|}$, we have

$$P\left[|N_\kappa| = n\right] = \binom{|N|}{n}\frac{1}{\left(b_\kappa\,|B|\,|C|\right)^n}\cdot\left(1-\frac{1}{b_\kappa\,|B|\,|C|}\right)^{|N|-n} = \binom{|N|}{n}\frac{\left(b_\kappa\,|B|\,|C|-1\right)^{|N|-n}}{\left(b_\kappa\,|B|\,|C|\right)^{|N|}}\ .$$

Consequently,

$$\begin{aligned}
\frac{E\left[|N_\kappa|\right]}{|N|} &= \frac{1}{|N|}\sum_{n=0}^{|N|} n\cdot P\left[|N_\kappa| = n\right]\\
&= \frac{1}{|N|}\sum_{n=0}^{|N|} n\binom{|N|}{n}\frac{\left(b_\kappa\,|B|\,|C|-1\right)^{|N|-n}}{\left(b_\kappa\,|B|\,|C|\right)^{|N|}}\\
&= \frac{1}{|N|}\left(\frac{b_\kappa\,|B|\,|C|-1}{b_\kappa\,|B|\,|C|}\right)^{|N|}\sum_{n=0}^{|N|} n\binom{|N|}{n}\frac{1}{\left(b_\kappa\,|B|\,|C|-1\right)^n}\\
&= \frac{1}{|N|}\left(\frac{b_\kappa\,|B|\,|C|-1}{b_\kappa\,|B|\,|C|}\right)^{|N|}|N|\,\frac{\left(b_\kappa\,|B|\,|C|\right)^{|N|-1}}{\left(b_\kappa\,|B|\,|C|-1\right)^{|N|}}\\
&= \frac{1}{b_\kappa\,|B|\,|C|}\ .
\end{aligned}$$

Since the resulting expression does not depend on $|N|$, the assumption of fixed $|N|$ can be relaxed, and thus, the claim is proved. □

In the following section, we will use this proposition to study the number of time slots required to minimize EMDT.

4.2 Upper Bound on Makespan of EMDT-Optimal Schedules

In this section, we will show that for an arbitrary set of BI's, EMDT can be minimized within $LCM(B)\cdot|C|$ time slots, and that, consequently, for BI sets in $\mathbb{F}_2$, EMDT can be

Copyright at Technische Universität Berlin.
All rights reserved.

minimized within $\max(B) \cdot |C|$ time slots, implying that in $\mathbb{F}_2$, EMDT-optimal schedules are also makespan-optimal. The following proposition establishes an upper bound on the number of time slots required to minimize an arbitrary strictly increasing function of discovery times. The idea for the proof is to show that any schedule that results in a network being detected after time slot $LCM(B) \cdot |C|$ can be modified such that the network in question is detected before time slot $LCM(B) \cdot |C|$, without increasing the discovery times of other networks.

Proposition 12. *For an arbitrary set of BI's $B \in \mathbb{F}_1$, a set of channels C, and a function $f : \mathbb{N}^{|K_{BC}|} \rightarrow \mathbb{R}$, which is strictly increasing in each argument, complete schedules $\mathcal{L}^*$ that minimize $f\left(\left(T_\kappa\left(\mathcal{L}\right)\right)_{\kappa \in K_{BC}}\right)$ have a makespan $T_{\mathcal{L}^*} \leq LCM(B) \cdot |C|$.*

Proof. Assume schedule $\mathcal{L}$ minimizes f and $T_\mathcal{L} > LCM(B) \cdot |C|$. Consequently, there is at least one configuration $\kappa = (c, b, \delta)$ with discovery time $T_\kappa\left(\mathcal{L}\right) = T_\mathcal{L} > LCM(B) \cdot |C|$. Consider time slots $\tilde{\mathcal{T}}_\kappa = \{\delta + i \cdot LCM(B)\}_{i \in \{0,\dots,|C|-1\}}$. Observe that $\left(\{c\} \times \tilde{\mathcal{T}}_\kappa\right) \cap \mathcal{L} = \emptyset$ since otherwise κ would have been detected during one of the time slots in $\tilde{\mathcal{T}}_\kappa$. Consequently, there either exists an idle time slot $\tilde{t} \in \tilde{\mathcal{T}}_\kappa$, or, since $|\tilde{\mathcal{T}}_\kappa| = |C|$, there exist time slots $t', t'' \in \tilde{\mathcal{T}}_\kappa$ and a channel $c' \neq c$ such that $\left(c', t'\right), \left(c', t''\right) \in \tilde{\mathcal{T}}_\kappa$.

In the first case, we construct a new schedule $\mathcal{L}' = \mathcal{L} \setminus \{(c, T_\mathcal{L})\} \cup \left\{\left(c, \tilde{t}\right)\right\}$, such that κ is detected during $\tilde{t}$ and none of the discovery times of other network configurations are increased.

In the second case, we construct a new schedule $\mathcal{L}' = \mathcal{L} \setminus \{(c, T_\mathcal{L}), (c', t'')\} \cup \{(c, t'')\}$. With the new schedule, configuration κ is detected during time slot t''. In order to show that the discovery times of other networks do not increase, consider the function $\delta(t)$, defined in Chapter 2, providing for each time slot t a vector of offsets that can be detected in t. Since periodicity of $\delta(t)$ is $LCM(B)$, we conclude that $\delta(t') = \delta(t'')$, and thus no discoveries are performed during time slot t'' with the schedule $\mathcal{L}$. Consequently, none of the discovery times are increased with the new schedule.

Repeating the above procedure for each κ with discovery time $T_\kappa\left(\mathcal{L}\right) > LCM(B) \cdot |C|$ results in a schedule $\mathcal{L}^*$ with makespan $T_{\mathcal{L}^*} \leq LCM(B) \cdot |C|$ with $f\left(\left(T_\kappa\left(\mathcal{L}^*\right)\right)_{\kappa \in K_{BC}}\right) < f\left(\left(T_\kappa\left(\mathcal{L}\right)\right)_{\kappa \in K_{BC}}\right)$, proving the claim. ☐

The following Corollary presents a notable consequence from Proposition 12 for BI sets from $\mathbb{F}_2$.

Corollary 13. *For a BI set $B \in \mathbb{F}_2$, a set of channels C, and a function $f : \mathbb{N}^{|K_{BC}|} \rightarrow \mathbb{R}$, which is strictly increasing in each argument, complete schedules $\mathcal{L}^*$ that minimize $f\left(\left(T_\kappa\left(\mathcal{L}\right)\right)_{\kappa \in K_{BC}}\right)$ are makespan-optimal.*

Proof. The claim follows directly from Corollary 5, Proposition 12, and the defining property of $B \in \mathbb{F}_2$ that $LCM(B) = \max(B)$. ☐

Please observe that Proposition 11 implies that the upper bounds established in Proposition 12 and Corollary 13 also apply to schedules minimizing EMDT.

Corollary 14. *For an arbitrary set of BI's $B \in \mathbb{F}_1$, and a set of channels C, EMDT-optimal listening schedules $\mathcal{L}^*$ have a makespan $T_{\mathcal{L}^*} \leq LCM(B) \cdot |C|$.*

Copyright at Technische Universität Berlin.
All rights reserved.

Proof. From Proposition 11 we obtain an expression for EMDT, which is strictly increasing in each configuration detection time T_κ. Applying Proposition 12 we obtain the claim. □

Corollary 15. *For a set of BI's $B \in \mathbb{F}_2$, and a set of channels C, EMDT-optimal listening schedules are also makespan-optimal.*

Proof. From Proposition 11 we obtain an expression for EMDT, which is strictly increasing in each configuration detection time T_κ. Applying Corollary 13 we obtain the claim. □

In the following Chapter we will propose efficient approaches to computing listening schedules that are both EMDT-optimal and makespan-optimal.

Copyright at Technische Universität Berlin.
All rights reserved.

Chapter 5

Discovery Strategies

In this section we present several novel discovery algorithms for multichannel environments, that are Expected Mean Discovery Time (EMDT)-optimal and makespan-optimal for the family of BI sets $\mathbb{F}_3$ (see Chapter 3 for a definition). Note that $\mathbb{F}_3$ contains a broad spectrum of BI sets, completely including BI sets defined by the IEEE 802.15.4 standard, and a large fraction of BI sets defined by the IEEE 802.11 standard. In addition, we develop an ILP-based approach, denoted GENOPT, that computes EMDT-optimal discovery schedules for arbitrary BI sets in $\mathbb{F}_1$.

Note that all presented strategies operate by passively listening to periodically transmitted beacon messages of neighbors that are agnostic to the discovery process. Moreover, due to their optimality for quite general sets of BI's, such as $\mathbb{F}_1$, $\mathbb{F}_2$, or $\mathbb{F}_3$, they can be deployed without conflicting with existing MAC layer technologies, allowing network operators to select BI sets suited best for the targeted application and/or device characteristics, such as, e.g., energy constraints. In particular, our results apply to BI sets used by technologies such as IEEE 802.11 and IEEE 802.15.4.

The rest of this chapter is structured as follows. In Section 5.1, we briefly present discovery strategies specified by the IEEE 802.15.4 standard as well as our previous work on optimized IEEE 802.15.4 discovery. In Section 5.2, we define the family of algorithms GREEDY, prove their optimality, and study their complexity. In Section 5.3, we describe a strategy named CHAN TRAIN, which is a modification of the GREEDY approach attempting at reducing the number of channel switches. An ILP-based approach to computing EMDT-optimal listening schedules for arbitrary BI sets is presented in Section 5.4. Finally, in Section 5.5 we describe a discovery strategy for the special case of BI sets with two elements.

5.1 Discovery Strategies for IEEE 802.15.4 Networks

In the following we will describe discovery strategies for IEEE 802.15.4 networks. First, we will explain the different scanning techniques offered in the IEEE 802.15.4 standard and then we will give a brief overview of our previous work on optimized discovery strategies for IEEE 802.15.4 networks.

5.1.1 PSV - Passive Scan in IEEE 802.15.4

The IEEE 802.15.4 standard supports BI's that are multiple of powers of 2. We denote the corresponding family of BI sets by $\mathbb{F}_{IEEE\ 802.15.4}$ (see also Chapter 3). The IEEE 802.15.4 standard defines four types of scanning techniques [2]. When performing an active or an

Copyright at Technische Universität Berlin.
All rights reserved.

orphan scan, devices transmit beacon requests or orphan notifications on each selected channel. With the passive scan and the energy detection scan, a device only listens on channels without any transmission. Frames are only decoded in the passive scan while the result of an energy scan is the peak energy per channel.

In our work we focus on passive discovery techniques and therefore compare our strategies to the passive scan of IEEE 802.15.4, denoted by PSV. PSV proceeds by sequentially listening on each channel $c \in C$ for $\max(B)$ time slots. Note that listening schedules generated by PSV are optimal w.r.t. the makespan and the number of channel switches for arbitrary BI sets $B \in \mathbb{F}_1$. Still, they fail to minimize EMDT, meaning that although the time when the last network is detected is minimized, the time of discovery of other network can be significantly worse than its optimal value, as shown in the following example. Consider a setting with $B = \{1, 2\}$ and $|C| = 2$. The schedule generated by PSV has an EMDT of 2.25, while minimum EMDT is 2 (scanning channel 1 during time slots 1 and 4, and channel 2 during time slots 2 and 3). The optimality gap further increases for larger scenarios.

5.1.2 (SW)OPT - Previous Work on Optimized IEEE 802.15.4 Discovery Strategies

In our previous work [6, 7] we developed discovery strategies for IEEE 802.15.4 networks, the OPTimzed (OPT) and the SWitched OPTimized (SWOPT) strategy. Both strategies are based on solving ILP minimizing the EMDT and showed significant improvement as compared to PSV as well as the SWEEP strategy [8]. OPT and SWOPT use the same ILP, however, SWOPT additionally performs the preprocessing on the BI set described in Section 3 (dividing all BI's by their GCD), resulting in listening schedules with fewer channel switches. Note that OPT and SWOPT generate recursive listening schedules.

Even though (SW)OPT was developed for BI sets from $\mathbb{F}_{\text{IEEE 802.15.4}}$, it can be adapted to support BI sets from $\mathbb{F}_3$. However, due to a grouping of time slots in the formulation of the ILP that allows to compute the discovery probability of individual network configurations, (SW)OPT cannot be extended to support BI sets from $\mathbb{F}_2$.

5.2 GREEDY Discovery Algorithms

In the following, we show that for the family of BI sets $\mathbb{F}_3$, schedules that are both EMDT-optimal and makespan-optimal can be computed in a very efficient way by greedily maximizing the number of discovered configurations in each time slot. In particular, we will show that for $B \in \mathbb{F}_3$ the computational complexity required by a straightforward implementation is $\mathcal{O}\left(|C|^2 |B| \max(B)\right)$, while memory consumption is $\mathcal{O}\left(|C| |B| \sum_{b \in B} b\right)$. We start by formally defining the family of algorithms GREEDY.

Definition 16 (GREEDY)**.** *The family of algorithms GREEDY contains all algorithms that greedily maximize the number of detected configurations in each time slot. More precisely, for an arbitrary BI set $B \in \mathbb{F}_1$ and channel set C, in every time slot t, an algorithm $A \in GREEDY$ scans channel $c \in C$ that maximizes the number of configurations from K_{BC} discovered in time slot t, given the configurations discovered in previous time slots $\{1, \ldots, t-1\}$.*

Copyright at Technische Universität Berlin.
All rights reserved.

In the following, we will say "GREEDY listening schedule" to refer to a schedule generated by GREEDY algorithm.

Note that for arbitrary BI sets, a GREEDY algorithm does not necessarily generate EMDT-optimal or makespan-optimal schedules. Consider an example using $B = \{1, 2, 3, 5\} \in \mathbb{F}_1 \setminus \mathbb{F}_2$ and $|C| = 3$. Figure 5.1a shows a GREEDY listening schedule, which is neither makespan-optimal nor EMDT-optimal, as can be seen from a comparison with the listening schedule in Figure 5.1b, which is both makespan-optimal and EMDT-optimal. The number in each square represents the number of configurations that can be discovered by scanning a particular channel during a particular time slot. Gray squares represent channels scanned by the illustrated listening schedule. In particular, optimal value for the makespan in this example constitutes 15 time slots, while optimal value for EMDT is 4.875 time slots, in contrast to 5.125 time slots achieved by the GREEDY schedule. Another example using $B = \{2, 3, 4, 6, 12\} \in \mathbb{F}_2 \setminus \mathbb{F}_3$ and $|C| = 2$ is depicted in Figure 5.2. The GREEDY listening schedule achieves a EMDT of 6.3 time slots in comparison with the optimal value of 6.1 time slots.

In the following, however, we will show that for the family of BI sets $\mathbb{F}_3$, GREEDY algorithms always result in EMDT-optimal and makespan-optimal schedules. We prove this claim by showing that GREEDY algorithms generate recursive schedules (see Definition 7). Afterwards, we will discuss computational complexity of GREEDY algorithms.

Proposition 17. *In network environments using BI sets from $\mathbb{F}_3$, a schedule is GREEDY if and only if it is recursive.*

Proof. W.l.o.g. we assume $B = \{b_1, \ldots, b_n\}$ with $b_i < b_j$ for $i < j$. From Proposition 8 we know that a schedule is recursive if and only if in each time slot $t \in [1, b_i \cdot |C|]$, for a $b_i \in B$, it discovers at least $n - i + 1$ configurations, which is the maximum number of discoverable configurations in each time slot. From Proposition 10 we know that in a complete environment with a BI set $B \in \mathbb{F}_3$ a recursive schedule exists. Consequently, since by definition, GREEDY schedules maximize the number of configurations discovered in each time slot, GREEDY schedules are recursive. The claim that a recursive schedule is GREEDY follows directly from Corollary 9. □

From this result we are able to derive the following conclusions.

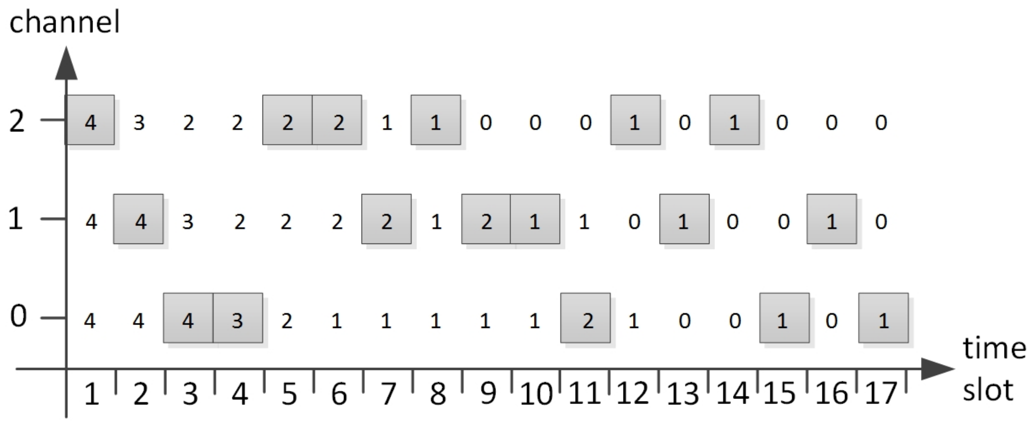
Corollary 18. *In a network environment with a BI set $B \in \mathbb{F}_3$, GREEDY schedules are complete, makespan-optimal, EMDT-optimal, and maximize the number of configurations detected until each time slot, for each $B' \subset B$.*

Proof. This is a direct consequence of Corollary 9, and Propositions 11 and 17. □

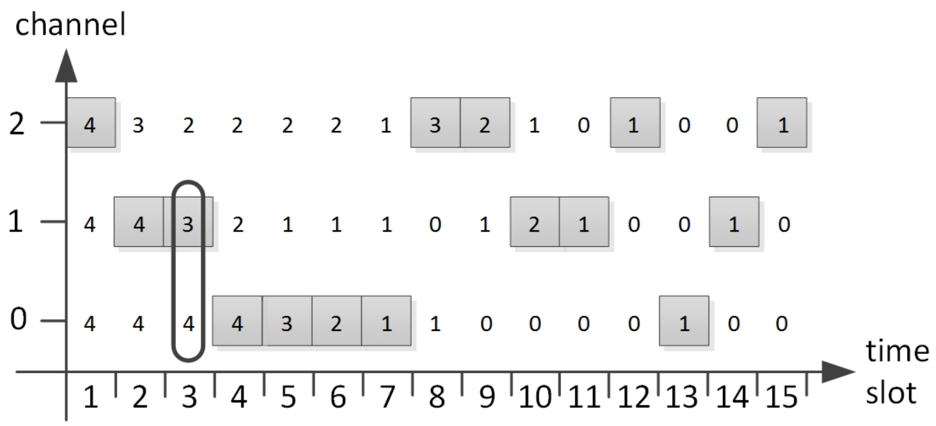
As shown in the example above, the assumption of a BI set from $\mathbb{F}_3$ is crucial to prove EMDT-optimality of GREEDY schedules. However, relaxing the assumption on the used BI set, we still can show that for the family of BI sets $\mathbb{F}_2$, GREEDY schedules are makespan-optimal, as stated in the following Proposition.

Proposition 19. *In a network environment with a BI set $B \in \mathbb{F}_2$, GREEDY schedules are makespan-optimal.*

Copyright at Technische Universität Berlin.
All rights reserved.



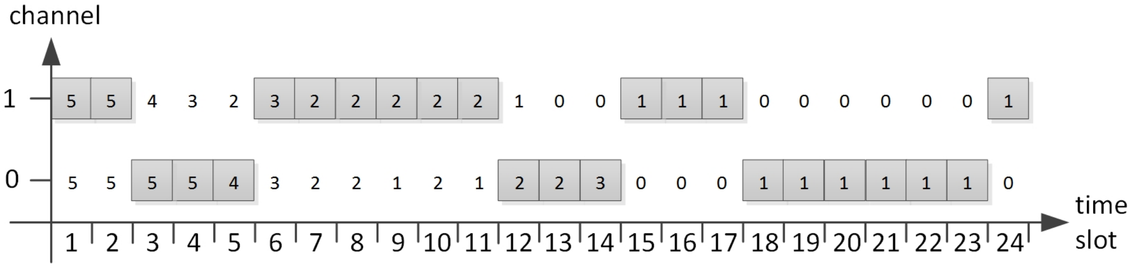
(a) GREEDY



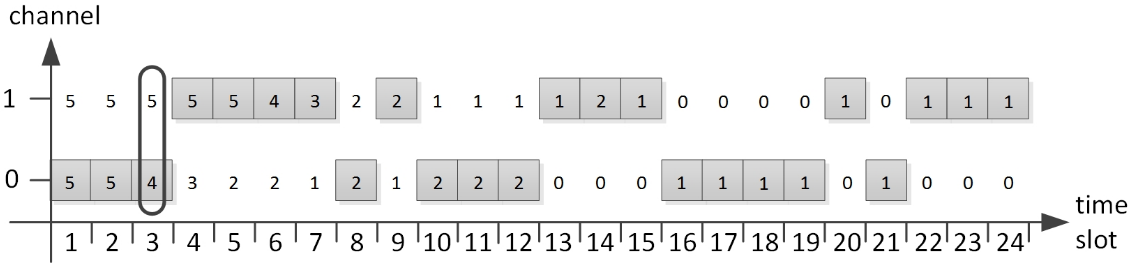
(b) EMDT-optimal schedule

Figure 5.1: Example for a GREEDY listening schedule depicted by gray boxes that is neither EMDT nor makespan optimal using BI set $B = \{1, 2, 3, 5\} \notin \mathbb{F}_2$ and $|C| = 3$.

Copyright at Technische Universität Berlin.
All rights reserved.



(a) GREEDY



(b) EMDT-optimal schedule

Figure 5.2: Example for a GREEDY listening schedule depicted by gray boxes that is not EMDT optimal using BI set $B = \{2, 3, 4, 6, 12\} \in \mathbb{F}_2 \setminus \mathbb{F}_3$ and $|C| = 2$.

Proof. Consider BI set $B = \{b_1, \ldots, b_n\} \in \mathbb{F}_2$, with $b_i < b_j$ for $i < j$, and a channel set C. Consider a GREEDY listening schedule $\mathcal{L}$. We first show that in each time slot $t \leq \max(B) \cdot |C|$, $\mathcal{L}$ detects exactly one configuration $(c, b_n, \delta_{b_n}(t))$, for a $c \in C$. Assume that this is not the case, that is, at time slot t, $\mathcal{L}$ scans channel c such that $(c, b_n, \delta_{b_n}(t))$ have already been detected earlier in a time slot $t - \ell \cdot b_n$ for an $\ell > 0$. Then, however, all other configurations $\{(c, b_i, \delta_{b_i}(t))\}_{i \in \{1,\ldots,n\}}$ has been detected earlier as well, since in $\mathbb{F}_2$, periodicity of $\delta_{b_n}(t)$ is an integer multiple of periodicity of $\delta_{b_i}(t)$ for $i \in \{1, \ldots, n-1\}$. Thus, $\mathcal{L}$ would scan a time slot resulting in 0 detected configurations or this time slot would be idle. On the other hand, however, we know from Corollary 5 that for each time slot $t \leq b_n \cdot |C|$ there exists a channel $c \in C$ such that a configuration $(c, b_n, \delta_{b_n}(t))$ can be detected. This contradicts the assumption that $\mathcal{L}$ is GREEDY, proving the claim. □

Note that GREEDY algorithms have a compelling property of low complexity. A straightforward example implementation proceeds as follows. It iterates over time slots until all configurations are discovered. For each configuration it stores a binary variable indicating if it has been discovered or not, resulting in $|C|\,|B| \sum_{b \in B} b$ bits of required memory space. At each time slot t, it iterates over all channels $c \in C$, computing for each channel, which of configurations $\{(c, b, \delta_b(t)) \mid b \in B\}$ are not yet discovered. Finally, it selects a channel, for which this number is highest. Consequently, computational complexity at each time slot is $\mathcal{O}(|C|\,|B|)$. The overall computational complexity depends on the number of time slots required to discover all configurations. Since for complete environments with $B \in \mathbb{F}_2$ GREEDY

Copyright at Technische Universität Berlin.
All rights reserved.

algorithms are makespan-optimal, the number of time slots is $\max(B)\,|C|$, resulting in total complexity of $\mathcal{O}\left(|C|^2\,|B|\max(B)\right)$. In $\mathbb{F}_1$, we only know that GREEDY algorithms terminate at latest in time slot $LCM(B)|C|$. (Proof is similar to proof of Proposition 19.) That is, an upper bound for the computational complexity over $\mathbb{F}_1$ is $\mathcal{O}\left(|C|^2\,|B|\,LCM(B)\right)$.

Please note that if the assumption made in our system model in Section 2 that the GCD of considered BI sets is 1 does not hold, the complexity can be further reduced by replacing B by $B' = \left\{\frac{b}{GCD(B)}\right\}_{b\in B}$. This preprocessing step allows to reduce computational complexity over $\mathbb{F}_2$ to $\mathcal{O}\left(|C|^2\,|B|\,\frac{\max(B)}{GCD(B)}\right)$, over $\mathbb{F}_1$ the upper bound becomes $\mathcal{O}\left(|C|^2\,|B|\,\frac{LCM(B)}{GCD(B)}\right)$.

Note that in general, in each time slot, there might exist several channels whose selection maximizes the number of discovered configurations. Therefore, GREEDY is a family of algorithms and not a single algorithm. The difference between them is the tiebreaking rule that, in each time slot, selects one channel to be scanned from the set of channels maximizing the number of discovered configurations. In the following we describe two deterministic and two probabilistic tiebreaking rules.

GREEDY RND randomly selects a channel among the channels maximizing the number of discovered configurations.

GREEDY DTR selects the channel with the highest identifier among the channels maximizing the number of discovered configurations.

GREEDY RND-SWT tests if the channel scanned in the previous time slot is within the set of channels maximizing the number of discovered configurations. If yes, it is selected. If no, it proceeds as GREEDY RND. By prioritizing the most recently selected channel, GREEDY RND-SWT tries to reduce the number of channel switches by creating a channel train.

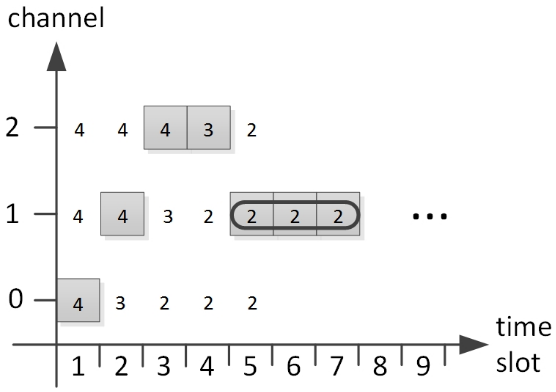
GREEDY DTR-SWT is similar to GREEDY RND-SWT but without a random component. It tests if the channel scanned in the previous time slot is within the set of channels maximizing the number of discovered configurations. If yes, it is selected. If no, it proceeds as GREEDY DTR.

5.3 CHAN TRAIN - Reducing the Number of Channel Switches

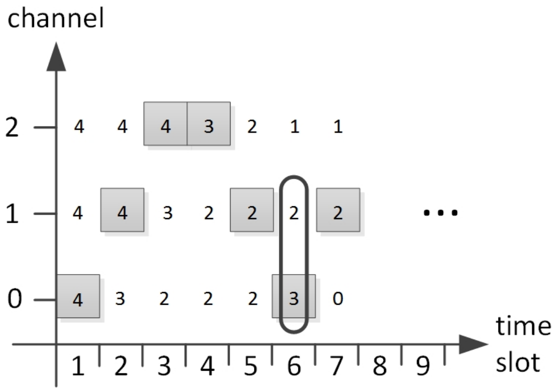
In this section, we propose an algorithm, which, similar to the GREEDY SWT algorithms in the previous section, in addition to minimizing EMDT, tries to minimize the number of channel switches. While GREEDY RND-SWT and GREEDY DTR-SWT do that by taking into account the most recently scanned channel, the CHAN TRAIN strategy goes one step further and also takes into account channels that will be scanned in future time slots.

To be more precise, in time slot $t = 1$, CHAN TRAIN computes the set of channels maximizing the number of detected configurations in t. Out of those, it than selects the channel maximizing the sum of consecutive future time slots t' where at least the same number of configurations can be detected and of the number of previous consecutive time slots allocated on this channel. In case multiple channels achieve the same maximum sum, it selects the channel with the lowest identifier. It then jumps to $t + t'$ and repeats the procedure.

Copyright at Technische Universität Berlin.
All rights reserved.



(a) CHAN TRAIN



(b) GREEDY

Figure 5.3: Example showing the non-greedy behavior of the CHAN TRAIN strategy for the BI set $B = \{1, 2, 3, 6\} \in \mathbb{F}_2 \setminus \mathbb{F}_3$ and $|C| = 3$

Proposition 20. *In an environment with a BI set $B \in \mathbb{F}_3$, CHAN TRAIN is GREEDY.*

Proof. From Proposition 8 and the definition of CHAN TRAIN we conclude that CHAN TRAIN generates recursive schedules for BI sets from $\mathbb{F}_3$. Propositions 11 and 17 then implies the claim. □

Note that for a BI set $B \notin \mathbb{F}_3$ Proposition 20 is no longer true, as illustrated by the example in Figure 5.3.

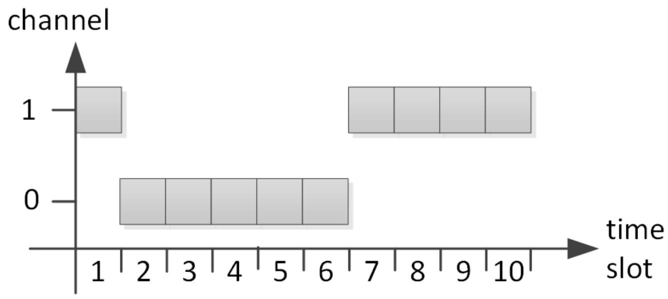
5.4 GENOPT - Minimizing Mean Discovery Time for Arbitrary Beacon Interval Sets

In the previous sections, we presented a family of low-complexity algorithms that are EMDT-optimal for the broad family of BI sets $\mathbb{F}_3$. Still, those algorithms might fail to achieve optimality for BI sets in $\mathbb{F}_1 \setminus \mathbb{F}_3$, as shown in examples in Figures 5.1 and 5.2.

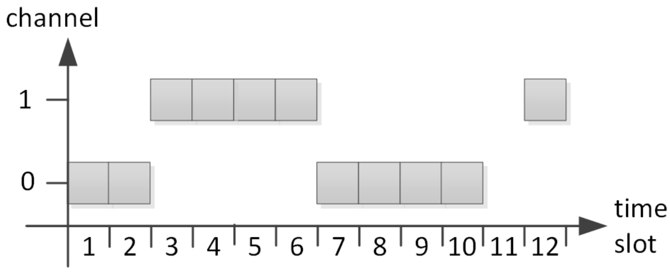
In this section, we formulate a set of linear constraints describing a complete listening schedule for arbitrary BI sets $B \in \mathbb{F}_1$. We use this set to develop a discovery strategy GENOPT that minimizes EMDT, by complementing it with an appropriate linear objective function. Note that this set of constraints might be used to generate schedules optimized w.r.t. other metrics. Since GENOPT involves solving an ILP, it suffers from high computational complexity and memory consumption, and should only be performed offline and for network environments that are limited in size.

Note that listening schedules generated by GENOPT might not be makespan-optimal as shown in following example. Figure 5.4 depicts an example using $B = \{1, 2, 4, 5\}$ and $|C| = 2$ for which an EMDT-optimal listening schedule cannot be constructed within $\max(B) \cdot |C|$ time slots. Figure 5.4a shows an EMDT-optimal listening schedule with the additional constraint

Copyright at Technische Universität Berlin.
All rights reserved.



(a) Makespan constrained by the optimum value $\max(B) \cdot |C|$



(b) Makespan constrained by $LCM(B) \cdot |C|$

Figure 5.4: Example for a listening schedule generated by GENOPT, using BI set $B = \{1, 2, 4, 5\} \in \mathbb{F}_1 \setminus \mathbb{F}_2$ and $|C| = 2$. Observe that imposing an upper bound on the makespan of the generated schedule increases EMDT. Therefore, in this example, no schedule exists which is both EMDT-optimal and makespan-optimal.

of using at most $\max(B) \cdot |C|$ time slots. Observe that its EMDT is 3.875 time slots as compared to the optimal value of 3.75 time slots of the schedule shown in Figure 5.4b.

To formulate GENOPT, we define the following variables.

$$x_{c,t,b} = \begin{cases} 1 \text{ , if configuration } (c, b, \delta_t(b)) \text{ is detected during scan of channel } c \text{ in time slot } t \\ 0 \text{ , otherwise} \end{cases}$$

$$h_{c,t} = \begin{cases} 1 \text{ , if a scan is performed on channel } c \text{ at time slot } t \\ 0 \text{ , otherwise} \end{cases}$$

GENOPT can be formulated as follows.

Copyright at Technische Universität Berlin.
All rights reserved.

$$
\begin{aligned}
\min \quad & \frac{1}{|C|\,|B|}\sum_{c\in C}\sum_{b\in B}\sum_{t=1}^{LCM(B)|C|} x_{c,t,b}\cdot t\cdot\frac{1}{b} & & \\
\text{s.t.} \quad & \sum_{m=0}^{\frac{LCM(B)\cdot|C|}{b}-1} x_{c,mb+\delta,b} = 1 & \text{for all } c\in C,\ b\in B,\ \delta\in\{1,\ldots,b\} & \quad \text{(C1)}\\
& x_{c,b,t} \le h_{c,t} & \text{for all } c\in C,\ b\in B,\ t\in\{1,\ldots,LCM(B)\,|C|\} & \quad \text{(C2)}\\
& \sum_{c\in C} h_{c,t} \le 1 & \text{for all } t\in\{1,\ldots,LCM(B)\,|C|\}\ . & \quad \text{(C3)}
\end{aligned}
$$

In this formulation, constraint (C1) ensures that each configuration is detected, (C2) ensures that a configuration $(c, b, \delta_t(b))$ can only be detected if channel c is scanned during time slot t, (C3) makes sure that only one channel is scanned during a time slot.

5.5 OPT_{B2} - Special case: $|B| = 2$

In this section, we describe the discovery strategy OPT_{B2} for the special case where the BI set contains two elements, $B = \{b_1, b_2\}$, $b_1 < b_2$, which is EMDT-optimal for any $B \in \mathbb{F}_1$ with $|B| = 2$. The number of channels can be arbitrarily large.

Definition 21 (OPT_{B2})**.** *We are given a BI set $B = \{b_1, b_2\}$, with $b_1 < b_2$, and a set of channels $C = \{c_1, \ldots, c_m\}$. Discovery strategy OPT_{B2} generates a listening schedule, where channel $j \in \{1, \ldots, m\}$ is scanned during time slots $[(j-1)b_1 + 1,\ jb_1]$ and $[mb_1 + (m-j)(b_2 - b_1) + 1,\ mb_1 + (m-j+1)(b_2 - b_1)]$.*

Proposition 22. *In network environments with BI sets $B = \{b_1, b_2\}$, OPT_{B2} is EMDT-optimal and makespan-optimal.*

Proof. Observe that OPT_{B2} generates recursive schedules. The claim then follows from Corollary 9 and Proposition 11. □

5.6 Results Overview

Table 5.1 provides an overview of optimality results for studied discovery strategies w.r.t. EMDT, makespan and number of channel switches, for different BI families, accompanied by computational complexity. Note that if a discovery strategy is makespan-optimal the generated listening schedules are also energy-optimal, since they do not contain idle time slots or redundant scans. The performance metrics we use for evaluation are described in detail in Section 6.2.

Copyright at Technische Universität Berlin.
All rights reserved.

Strategy	Makespan	EMDT	Channel Switches	Complexity
PSV	$\mathbb{F}_1$		$\mathbb{F}_1$	$\mathcal{O}(C)$
(SW)OPT	$\mathbb{F}_3$	$\mathbb{F}_3$		High (ILP)
GREEDY	$\mathbb{F}_2$	$\mathbb{F}_3$		$\mathcal{O}\left(\lvert C\rvert^2 \lvert B\rvert\, LCM(B)\right)$
GENOPT	$\mathbb{F}_2$	$\mathbb{F}_1$		High (ILP)
OPT_{B2} (only $\lvert B\rvert = 2$)	$\mathbb{F}_1$	$\mathbb{F}_1$		$\mathcal{O}(C)$

Table 5.1: Overview of discovery strategies and their optimality w.r.t. the makespan, EMDT, and number of channel switches, for different families of BI sets.

Copyright at Technische Universität Berlin.
All rights reserved.

Chapter 6

Evaluation

In the following we present a numerical evaluation of the proposed discovery strategies. Complementing optimality results w.r.t. EMDT and makespan for BI sets from $\mathbb{F}_3$, presented in Chapter 5, we evaluate the proposed algorithms over BI sets for which EMDT-optimality and/or makespan-optimality is not guaranteed, also considering performance metrics such as number of channel switches, and energy consumption. We will show that the performance of GREEDY algorithms is very close to the optimum in most settings. Given their low complexity, they thus represent an attractive solution to the problem of neighbor discovery for a broad range of deployment scenarios.

We will describe the evaluation setting in Section 6.1, the performance metrics in Section 6.2, and the results of the evaluation in Section 6.3.

6.1 Evaluation Setting

In the following, we evaluate and compare GREEDY discovery algorithms with PSV, CHAN TRAIN, and GENOPT strategies. In order to evaluate the proposed algorithms over BI sets from $\mathbb{F}_1$ and $\mathbb{F}_2$ we draw random samples from these families of BI sets until the confidence intervals for the studied performance metrics are sufficiently small. Note, however, that in order to include GENOPT, which is based on solving an ILP, into the evaluation, we have to restrict the size of the studied scenarios, to keep the computational effort feasible. The number of decision variables of GENOPT depends mainly on the $LCM(B)$ and the number of channels $|C|$. Consequently, we have to restrict the maximum number of channels, as well as the number of elements in studied BI sets, and their size. In particular, we vary the number of channels between 2 and 12. The minimum size of BI sets is set to $|B| = 3$, since the special case $|B| = 2$ can be solved in an EMDT-optimal and makespan-optimal way for arbitrary BI sets from $\mathbb{F}_1$, as show in Section 5.5. Please note that due to the fact that a listening schedule for a BI set $B = \{b_1, \ldots, b_n\}$ can be transformed into a schedule for a BI set $B' = \{c \cdot b_1, \ldots, c \cdot b_n\}$, substituting $\tau' = c \cdot \tau$, we only consider BI sets with $GCD(B) = 1$. Still, the results of the evaluation are valid for a much broader range of BI sets, including all sets obtained by multiplying individual BI's with a constant factor.

In the following we describe the procedure used to sample families of BI sets $\mathbb{F}_1$ and $\mathbb{F}_2$ as well as the approach to computing ILP-based listening schedule for the GENOPT strategy.

Copyright at Technische Universität Berlin.
All rights reserved.

6.1.1 Sampling $\mathbb{F}_1$

We draw random samples $B \in \mathbb{F}_1$ as follows. We first draw $|B|$ from a uniform distribution over $[3, 6]$. We then draw individual BI's from a uniform distribution over $[1, 10]$. The selected BI's are then divided by their GCD. If the resulting BI set is new, it is used for evaluation. In total, this approach results in 775 unique BI sets.

6.1.2 Sampling $\mathbb{F}_2$

We draw random samples $B \in \mathbb{F}_2$ as follows. For each number from $[1, 256]$ we first compute the power set of its factors. From the computed power sets, we select subsets whose cardinality is between 3 and 8, that contain the number itself, and whose GCD is one. In total, we obtain 259286 unique BI sets.

6.1.3 Computing GENOPT

The computational complexity of GENOPT strongly depends on the maximum number of time slots allowed for the solution. Corollary 14 states that the makespan of EMDT-optimal schedules is bounded from above by $LCM(B) \cdot |C|$. Note that $LCM(B)$ is $\mathcal{O}(\max(B)!)$ and thefore computing solutions over such a higher number of time slots can be extremely time consuming. However, we observed that in the vast majority of cases, optimal solutions can be found within $2 \cdot \max(B)$ time slots. Therefore, in order to reduce solving time, we execute GENOPT iteratively, increasing the upper bound on the maximum time slots. More precisely, we initialize the Gurobi solver [5] with the best solution found by executing the GREEDY RND strategy multiple times (1000 times for $B \in \mathbb{F}_1$, 50 times for $B \in \mathbb{F}_2$, because of higher complexity due to larger considered $\max(B)$) and restrict the maximum number of time slots to the number of time slots required by this solution. In the second step, the number of slots is doubled, while in the final third step, it is set to $LCM(B) \cdot |C|$. For the first and second iteration the time limit for the optimization is set to 1800 seconds, while for the final iteration it is set to 7200 seconds. If, however, after the time limit is exceeded the optimality gap is still greater than 3% for $B \in \mathbb{F}_1$ and 1% for $B \in \mathbb{F}_2$, the optimization is resumed until the optimality gap is sufficiently small.

6.2 Performance Metrics

The discovery strategies have been compared w.r.t. the performance metrics described in the following.

EMDT The EMDT is the expected mean detection time of all neighbor networks, given that all channels, BI's, and offsets have an equal probability to be selected by a network (see Chapter 4 for more details). Here expected value is over all possible network environments for a given set of channels and set of BI's, while mean is over the neighbor networks in a particular environment (see Section 4.1 for more details). In the following, EMDT results are depicted normalized by their optimal value, or, to be more precise, by the objective boundary obtained while generating the listening schedule for GENOPT (which is within 1% to 3% from the optimum, see Section 6.1.3).

Copyright at Technische Universität Berlin.
All rights reserved.

The EMDT is an important metric in scenarios in which a device has to discover a subset of its neighbors, e.g. in Delay Tolerant Networks (DTN's) in which devices have to detect suitable forwarders.

Number of channel switches This metric refers to the number of times a device has to change the listening channel when executing a complete listening schedule. In the following, it is depicted normalized by the minimum number of switches for a complete schedule, which is $(|C| - 1)$. When devices perform a channel switch they are in a deaf period in which they are not able to receive any frames, which might lead to losing beacons. A low number of channel switches is especially important if the ratio of the channel switching time to the duration of the time slot τ is large. In [6, 7] it has been shown that listening schedules with higher number of channel switches result in higher discovery times when being executed in a realistic environment on IEEE 802.15.4 devices, even though under ideal conditions, when the duration of a channel switch is assumed to be 0, they exhibit identical performance.

Makespan The makespan is the total number of time slots required by a listening schedule to detect all networks that are using any configuration $\kappa \in K_{BC}$ (see Definition 3). In other words, it is the number of time slots required by the schedule to detect all neighbors in a complete environment. Note that in environments that are not complete, makespan does not equal the detection time of the last neighbor. Still, the device performing the discovery cannot know when all neighbors are detected and, therefore, has to execute the schedule until all potential neighbors using a configuration $\kappa \in K_{BC}$ are detected. In the following, makespan is depicted normalized to the optimum makespan $\max(B) \cdot |C|$, established by Corollary 5.

Number of active time slots The number of active slots of a schedule is the number of time slots during which a device is actually listening on any channel. It is proportional to the amount of energy required to execute a schedule. The number of active time slots is always less or equal than the makespan due to the fact that a listening schedule might contain idle time slots during which no scan is allocated on any channel. In particular, makespan-optimal schedules are also optimal w.r.t. the number of active time slots. In the following, the number of active time slots is depicted normalized to its optimum value $\max(B) \cdot |C|$.

6.3 Evaluation Results

In the following, we extend the analytical optimality results established in Chapter 5 by numerical evaluations over sets of BI's for which optimality cannot be proven. Moreover, in addition to EMDT and makespan considered in Chapter 5, we extend the set of considered performance metrics by the number of channel switches and number of active time slots. As described in more details in Section 6.1, we randomly sample the families of BI's $\mathbb{F}_1$ and $\mathbb{F}_2$, and vary the number of channels $|C|$ between 2 and 12. For each $|C| \in [2, 12]$ we repeat the evaluation for 150 different BI sets. The values depicted in the following are mean values computed over 150 iterations with different BI sets, accompanied by the confidence intervals for the confidence level 95%.

Copyright at Technische Universität Berlin.
All rights reserved.

6.3.1 Family of BI Sets $\mathbb{F}_2$

In Chapter 5 we showed that GREEDY algorithms are makespan-optimal for BI sets from $\mathbb{F}_2$. Moreover, the results of the numerical evaluation revealed that CHAN TRAIN is makespan-optimal over the evaluated BI sets from $\mathbb{F}_2$. Therefore, in the following, we only show results for the EMDT and the number of channel switches. (Note that the number of active time slots is also optimal for makespan-optimal schedules.)

Figure 6.1a depicts the normalized EMDT of the evaluated discovery strategies. We observe that GREEDY algorithms are within 2% of the optimum and approximate the optimum even further when the number of channels increases. CHAN TRAIN has a slightly higher EMDT, which is, however, still within 3% of the optimum. In contrast, PSV has a significantly larger EMDT, reaching 400% of the optimum, and diverging when the number of channels increases. Note that the 4 studied GREEDY algorithms are almost indistinguishable w.r.t. to their EMDT at the given confidence level.

Figure 6.1 shows the number of channel switches normalized to the minimum value $|C|-1$. We can observe that the design of CHAN TRAIN aiming at reducing the number channel switches is successful at achieving its goal. It results in the second lowest number of channel switches with about 30 times the number of switches as compared to optimum achieved by PSV. Furthermore the distance to the optimum number is constant with increasing number of channels. Further we observe that GREEDY algorithms GREEDY RND-SWT and GREEDY DTR-SWT, that prioritize the channel allocated on the previous time slot if possible, allow to reduce the number of channel switches w.r.t. their versions that do not do the prioritization. We also observe that GENOPT and GREEDY RND have relatively poor performance, requiring up to a factor of 200 more switches than the optimum value achieved by PSV.

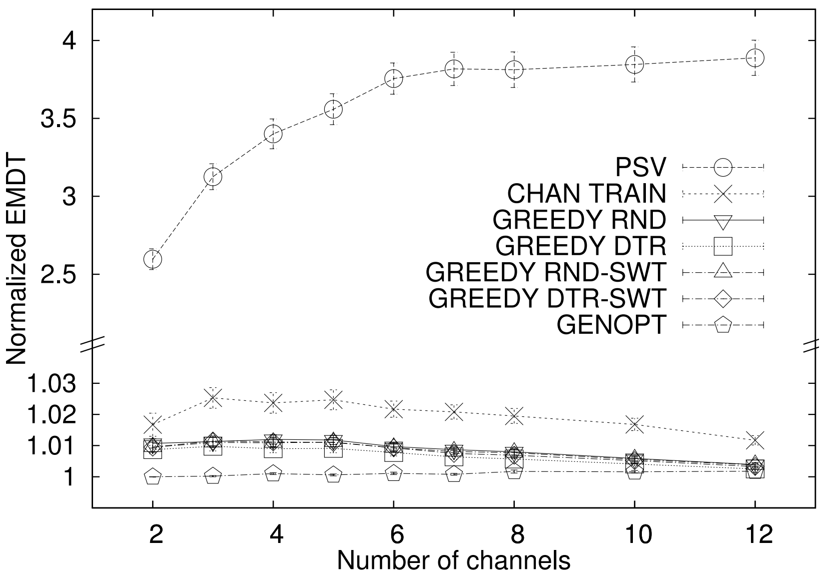
6.3.2 Family of BI Sets $\mathbb{F}_1$

Figure 6.2a shows the normalized EMDT for the family of BI sets $\mathbb{F}_1$. We observe that the performance of the individual discovery algorithms, realative to each other, did not significantly change. We also observe that their normalized EMDT slightly improved. On the one hand, a potential explanation for this is the less regular structure of BI sets in $\mathbb{F}_1$. As shown in Corollary 14, the upper bound for the makespan of EMDT-optimal shedules is $LCM(B) \cdot |C|$, while in $\mathbb{F}_2$ it is $\max(B) \cdot |C|$ (as shown in Corollary 15). On the other hand, the evaluation for BI sets from $\mathbb{F}_1$ was performed with considerably smaller BI's, in order to make the evaluation of GENOPT computationally feasible. Still, while EMDT of GREEDY strategies are within 1% of the optimum, EMDT of PSV reaches 160% of the optimum is is increasing with the number of channels.

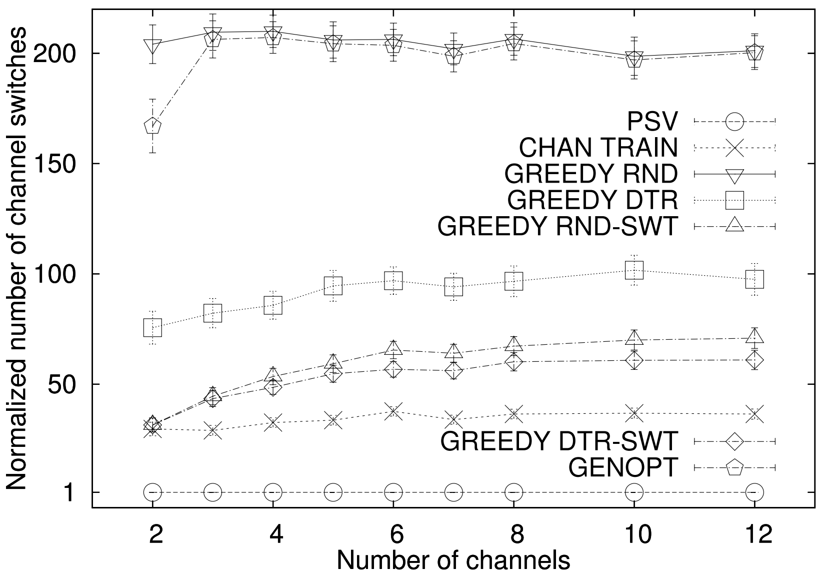
Figure 6.2b displays the number of channel switches. Similar to EMDT, we observe that the relative performance of the discovery strategies remained similar as for $\mathbb{F}_2$. Also, the distance to the optimum has decreased for all approaches, which, again, might be due to the smaller BI's used for evaluation.

Figure 6.3a shows the normalized makespan for the family of BI sets $\mathbb{F}_1$. For all strategies the normalized makespan improves with the increasing number of channels. E.g., for GENOPT the gap reduces from about 15% for two channels to about 1% for 12 channels.

Copyright at Technische Universität Berlin.
All rights reserved.



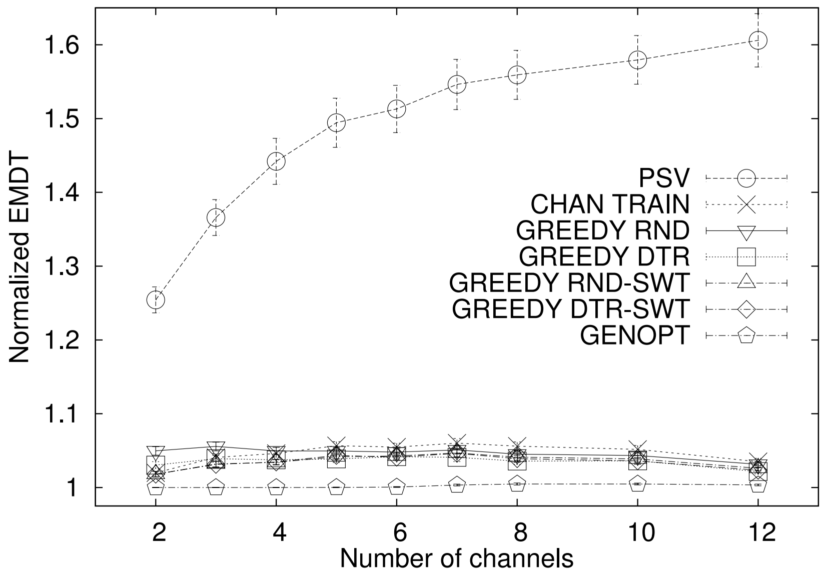
(a) EMDT



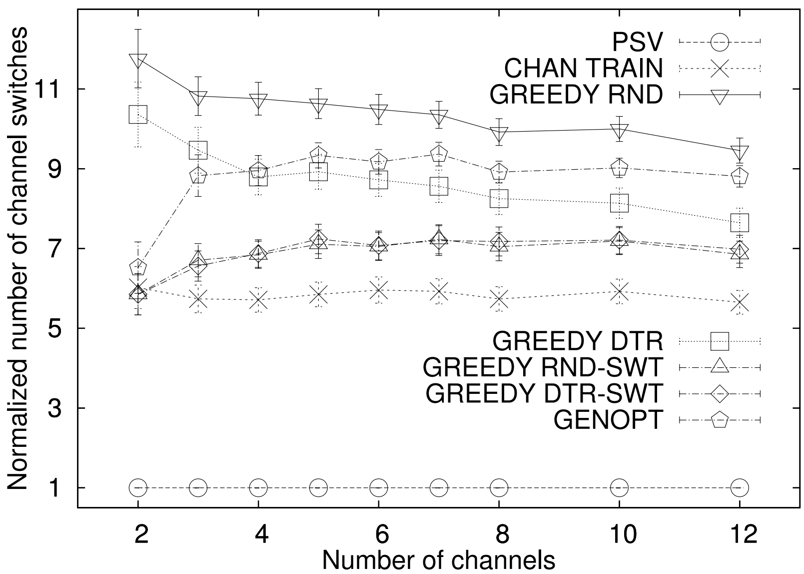
(b) Number of Channel Switches

Figure 6.1: Evaluation results for the family of BI sets $\mathbb{F}_2$ (see Section 6.3.1 for details).

Copyright at Technische Universität Berlin.
All rights reserved.



(a) EMDT



(b) Number of Channel Switches

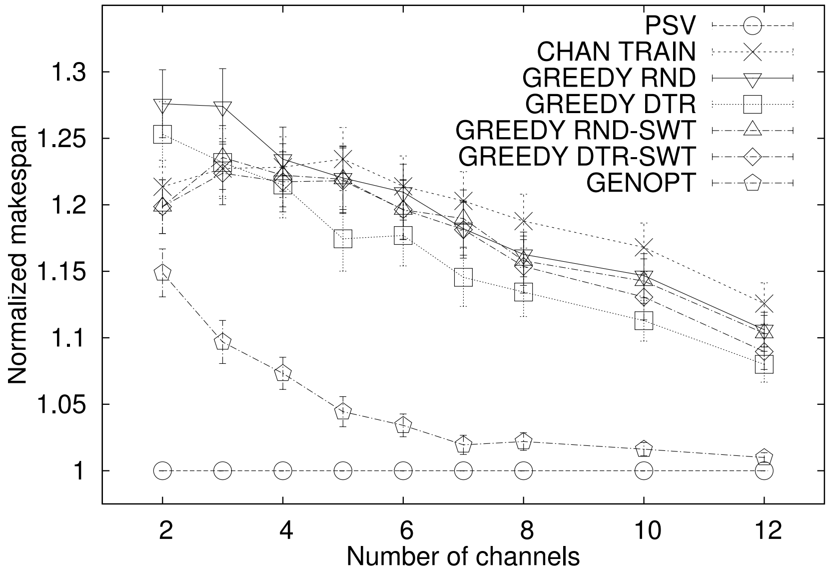
Figure 6.2: Evaluation results for the family of BI sets $\mathbb{F}_1$ (see Section 6.3.2 for details).

Copyright at Technische Universität Berlin.
All rights reserved.

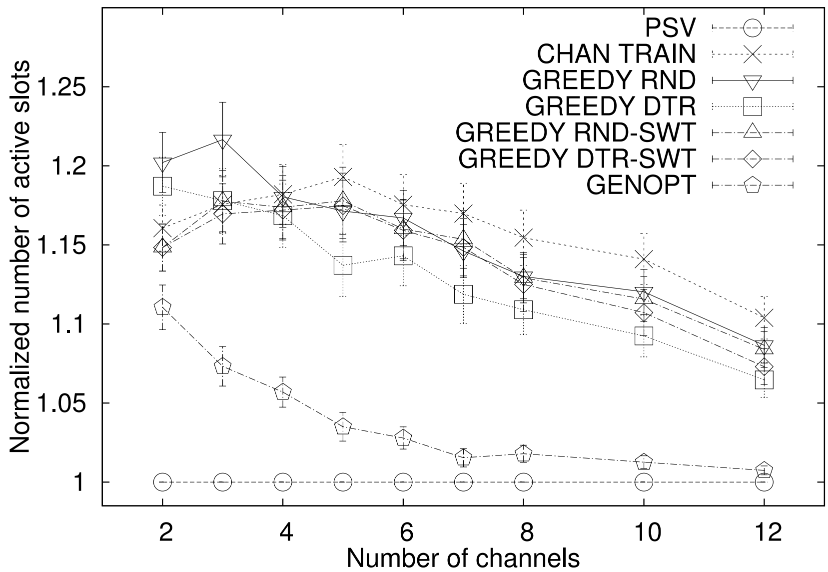
The listening schedules of the considered GREEDY approaches and CHAN TRAIN result in similar makespan at the considered confidence level.

Since makespan is only providing a measure for the last discovery time but not the actual energy usage, Figure 6.3b depicts the normalized number of active slots. The results for all strategies are very similar to those for the makespan, except that they are shifted by about 5% meaning that the schedules of the GREEDY approaches, CHAN TRAIN and GENOPT consist of about 5% idle slots in which no scan is scheduled on any channel due to the fact that no new configurations will be discovered.

Copyright at Technische Universität Berlin.
All rights reserved.



(a) Makespan



(b) Number of Active Slots

Figure 6.3: Evaluation results for the family of BI sets $\mathbb{F}_1$ (see Section 6.3.2 for details).

Copyright at Technische Universität Berlin.
All rights reserved.

Chapter 7

Selection of Beacon Intervals to Support Efficient Neighbor Discovery

In the preceding sections we presented and evaluated discovery strategies that support arbitrary finite BI sets from $\mathbb{F}_1$. However, the low-complexity GREEDY approaches are EMDT-optimal only for BI sets from $\mathbb{F}_3$ and makespan-optimal for BI sets from $\mathbb{F}_2$. Although we also presented an ILP-based approach GENOPT which is EMDT-optimal for BI sets from $\mathbb{F}_1$, it suffers from high computational complexity and can only be applied offline and to network environments limited in size.

In this section we discuss and provide intuition for problems arising when constructing listening schedules for BI sets in $\mathbb{F}_1$ and $\mathbb{F}_2$, and provide a recommendation for the selection of BI sets that support efficient neighbor discovery. This recommendation may be useful for the development of new technologies and communication protocols for wireless communication that use periodic beacon frames for management or synchronization purposes or in case of deploying devices using existing technologies that support a wide range of BI's, such as IEEE 802.11. In particular, we identified two following issues that can arise when constructing listening schedules.

- The makespan of the schedule is greater than $\max(B) \cdot |C|$ time slots (i.e., it is not makespan-optimal).
- The BI's are not discovered in ascending order (i.e., the schedule is not recursive).

The first problem of listening schedules being not makespan-optimal is caused by idle slots and redundant scans. During idle slots no scan is scheduled due to the fact that no new configurations can be discovered on any channel. In contrast, during redundant slots, a part of neighbors that send beacons during this time slot has already been discovered. Redundant scans are caused by the fact that information acquired in previously scanned slots cannot be completely reused for the discovery of other configurations on the same channel.

Figure 7.1a shows an optimized listening schedule generated by GENOPT for $B = \{1, 2, 3\}$ and $|C| = 2$ and is an example for a schedule containing redundant scans. Even though $\max(B) = 3$ there are four time slots scanned on channel c_1. The reason for the redundant scans is a conflict between the attempt to decrease the discovery time for smaller BI's in order to minimize EMDT and the missing possibility of reusing the information acquired by scanning the time slot scheduled for smaller BI's b_1 and b_2 for the discovery of neighbors operating with the larger BI b_3. From time slot t_1 to t_4, the schedule finishes the discovery of neighbors using b_1 and b_2 with a minimum EMDT. However, the scan at t_4 will not contribute

Copyright at Technische Universität Berlin.
All rights reserved.

to the discovery of neighbors operating with b_3 on c_1 since t_1 was already scanned on c_1. Another listening schedule achieving equal EMDT is described by $\mathcal{L} = \{(1,1),(0,3),(1,2)\}$ (see Section 2 for definitions and notation). In total it requires one time slot less but also increases the discovery time for neighbors using b_2 on c_1 while decreasing the discovery time for b_3 on c_0. This schedule suffers from the second identified issue that will be discussed further below.

Figure 7.1b depicts an example for a listening schedule generated by GENOPT for $B = \{2,3,4\}$ and $|C| = 2$ containing an idle slot. The schedule first completes the discovery of neighbors with b_2 on c_1, then b_2 and b_3 on c_0, and then b_3 on c_1 during time slot t_6. Due to the fact that scanning t_6 on c_1 does not result in any new information regarding neighbors operating with b_4, since t_2 has already been scanned on c_1, time slots t_7 and t_8 have to be scanned to finish the discovery of all neighbors operating on c_1. Time slot t_9 is not assigned to channel c_0 because t_5 has been already scanned and therefore the time slot will be idle.

Similar to the previous example illustrating redundant scans, there also exists an alternative listening schedule $\mathcal{L} = \{(1,2),(0,4),(1,3)\}$ for the given parameter set resulting in the same EMDT. This schedule does not contain any idle time slots but, again, completes the discovery of neighbors with larger BI's before detecting all neighbors with lower BI's. In this example the discovery time of neighbors operating with b_3 on c_1 is increased while the discovery of neighbors using b_4 on c_1 is speeded up.

The second identified issue relates to the order in which configurations with specific BI's are discovered. In order to reduce EMDT for a given BI set B and channel set C a listening schedule shall discover neighbors in ascending order of their BI's. We call such schedules recursive (see Definition 7).

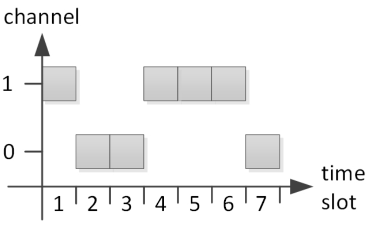
Examples of this issue were already mentioned in the description of the first issue regarding alternative schedules to the solutions shown in Figures 7.1a and 7.1b. Another example is shown in Figure 7.1c. The discovery of neighbors using b_1 or b_2 is completed on channel c_1 even before a time slot has been scanned on channel c_2. Due to the composition of BI's in set B a recursive listening schedule does not exist.

In the following we will give a recommendation on the selection of BI's sets that help avoiding the issues described above. To support efficient neighbor discovery by eliminating idle time slots as well as redundant scans, and to enable generation of recursive listening schedules, we recommend to prefer BI sets from $\mathbb{F}_3$ over those from $\mathbb{F}_1$ and $\mathbb{F}_2$. Note that for the family of BI sets $\mathbb{F}_3$ our low-complexity GREEDY algorithms generate recursive listening schedules that result in optimal EMDT and optimal makespan.

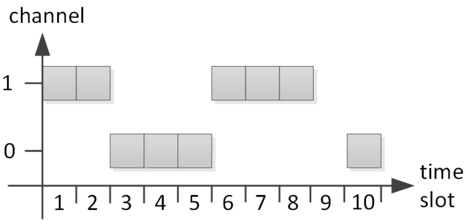
Nevertheless, if a higher diversity of BI's is required, we recommend to select $\mathbb{F}_2$. For this family GREEDY algorithms and CHAN TRAIN generate listening schedules with optimal makespan and a slightly higher EMDT than the optimal value. Depending on whether EMDT or the number of channel switches shall be prioritized, we recommend to use either one of the GREEDY approaches or, alternatively, the CHAN TRAIN strategy.

In case a scenario requires using a BI set from $\mathbb{F}_1$, we suggest to use either a GREEDY algorithm or the CHAN TRAIN strategy. Both options allow to generate listening schedules with an EMDT that is within 5% of the optimal value for the evaluated BI sets. If a dynamic computation of listening schedules directly on the devices themselves is not required and the BI set B and channel set C are static, GENOPT can be used to compute a schedule offline and store it on the devices.

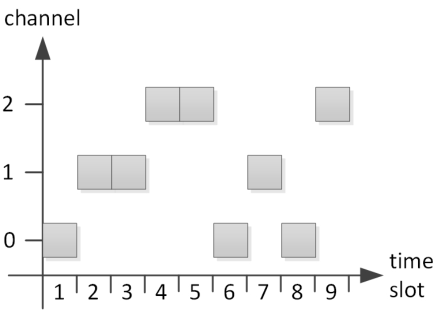
Copyright at Technische Universität Berlin.
All rights reserved.



(a) Redundant scans
$B = \{1, 2, 3\} \quad |C| = 2$



(b) Idle slots
$B = \{2, 3, 4\} \quad |C| = 2$



(c) Order of BI discovery
$B = \{1, 2, 3\} \quad |C| = 3$

Figure 7.1: Issues arising when using arbitrary BI sets.

Copyright at Technische Universität Berlin.
All rights reserved.

Chapter 8

Conclusion

This paper extends our previous work on asynchronous passive multi-channel discovery. It presents novel discovery strategies that are not limited to BI's defined in the IEEE 802.15.4 standard but support a broad range of BI sets. In particular, we characterizes a family of low-complexity algorithms, named GREEDY, minimizing makespan and EMDT for the broad family of BI sets $\mathbb{F}_3$, where each BI is an integer multiple of all smaller BI's. Notably, this family completely includes and significantly extends BI sets supported by the IEEE 802.15.4 standard. In addition, we develop an ILP-based approach minimizing EMDT for arbitrary BI sets that, however, exhibits high computational complexity and memory consumption and is, therefore, only applicable for offline computation and for network environments with limited size.

In addition to analytically proving optimality for the developed approaches for BI sets from $\mathbb{F}_3$, we performed an extensive numerical evaluation on two more general families of BI including the most general ones. We compared our strategies with the passive discovery strategy defined by the IEEE 802.15.4 standard and showed that our approaches exhibit superior performance w.r.t. several performance metrics. Finally, we provide recommendations on the selection of the BI sets that are as nonrestrictive as possible, at the same time allowing for an efficient neighbor discovery.

Our future work will focus on collaboration between devices by sharing gossip information about discovered neighbors in their beacon messages. By including this information into listening schedules computation, devices will be able to speed up the discovery of their neighbors. In addition, we will evaluate the developed approaches in realistic environments using simulations and experiments.

Copyright at Technische Universität Berlin.
All rights reserved.

Bibliography

[1] Gartner Press Release. http://www.gartner.com/newsroom/id/2905717.

[2] IEEE Standard 802.15.4-2006. 2006.

[3] IEEE Standard 802.11-2012. 2012.

[4] D. Evans. The Internet of Things - How the Next Evolution of the Internet Is Changing Everything. Technical report, Cisco Internet Business Solutions Group (IBSG), April 2011.

[5] Inc. Gurobi Optimization. Gurobi optimizer reference manual, 2015.

[6] N. Karowski, A.C. Viana, and A. Wolisz. Optimized Asynchronous Multi-channel Neighbor Discovery. In *Proc. of IEEE INFOCOM*, April 2011.

[7] N. Karowski, A.C. Viana, and A. Wolisz. Optimized Asynchronous Multichannel Discovery of IEEE 802.15.4-Based Wireless Personal Area Networks. *IEEE Transactions on Mobile Computing*, 12(10):1972–1985, October 2013.

[8] A. Willig, N. Karowski, and J.-H. Hauer. Passive Discovery of IEEE 802.15.4-based Body Sensor Networks. *Elsevier Ad Hoc Networks Journal*, 8(7):742–754, 2010.

Copyright at Technische Universität Berlin.
All rights reserved.