DOI: 10.1109/IOLTS.2019.8854406

Power-aware Reliable Communication for the IoT

Philipp H. Kindt
Chair of Real-Time Computer Systems (RCS)
Technical University of Munich (TUM)
Munich, Germany
kindt@rcs.ei.tum.de

Samarjit Chakraborty
Chair of Real-Time Computer Systems (RCS)
Technical University of Munich (TUM)
Munich, Germany
samarjit@tum.de

***Abstract*—Wireless devices are becoming increasingly pervasive in our everyday life and hence, robust wireless connectivity is a crucial requirement for a host of applications. In particular, with the IoT becoming a reality, an increasing number of wireless devices are frequently brought into their range of reception simultaneously, which need to interact with each other in an ad-hoc fashion. While effective solutions for reliable connected communication exist, establishing a first contact between devices remains the Achilles' heel of mobile wireless devices. For a reliable operation, every device needs to discover all devices in its range within bounded time. Thus, wireless protocols that bound the time within which devices are discovered need to be designed. However, even in such protocols, it is inevitable that a certain fraction of packets sent by multiple devices discovering each other simultaneously collide. As a result, some discoveries fail and a dependable operation can only be realized if the fraction of failed discoveries is kept negligibly low, even in worst-case scenarios. In addition, most devices are battery-powered and rely on a low-power operation for achieving reasonably long battery lifetimes. In this paper, we discuss how wireless protocols can be designed to realize a highly reliable and dependable connection setup, while at the same time meeting the energy-constraints of mobile IoT devices. Towards this, we analyze the challenges in protocol design and discuss which properties of the wireless hardware impact the robustness of the connection setup procedure of IoT devices.**

***Index Terms*—Wireless Networks, Neighbor Discovery, Bluetooth**

I. Introduction

Within the last decades, our environment has become increasingly wireless. While wireless connectivity has been initially used mostly at home for connecting laptops to the internet, more and more *mobile* applications are becoming widespread. For example, a host of gadgets like smartwatches, etc., are connected to smart phones. Battery-powered location beacons are deployed within buildings to enable indoor navigation services. Further, the industry increasingly makes use of wireless applications, for example by attaching transmitters to containers in warehouses. All of these applications have in common that the devices run on batteries and are therefore highly power-constrained. Further, their mobile nature requires that all devices frequently discover the communication partners within their surrounding in a reliable manner. The robustness of this procedure, which is called *neighbor discovery* (ND), is therefore an important prerequisite for the proper functioning of mobile wireless systems, especially in IoT scenarios. At the same time, this procedure needs to be carried out in an energy-efficient manner, since draining the batteries too quickly would prevent the realization of many applications.

To achieve robust and reliable discovery, multiple *deterministic* ND protocols have been proposed [1]–[5], which guarantee discovery within bounded time. In such protocols, after two devices come into their range of reception, the time until discovery occurs is limited. Whereas many of these protocols have been studied thoroughly in academic research, the class of *periodic interval (PI)-based* protocols, such as e.g., Bluetooth Low Energy (BLE), has become widely used in practice. In such protocols, every device transmits beacons periodically with a period T_a, called the *advertising interval*. Similarly, every device listens to the channel for incoming packets for d_s time-units every T_s time-units. The duration d_s is called the *scan window* and the period T_s is called the *scan interval*. T_a, d_s and T_s are universally defined and drawn over continuous time.

The dependability of the discovery procedure strongly depends on the values of T_a, T_s and d_s. In particular, for some valuations, discovery is not guaranteed at all, whereas some other parametrizations can guarantee discovery, but are not beneficial in terms of latency and energy consumption. The BLE specification [8] does not suggest how these parameters need to be chosen to achieve a good performance and a high dependability. Hence, how PI-based protocols need to be configured to guarantee short discovery latencies while at the same time preserving the battery charge has not been clear until recently.

In this paper, we outline how such protocols can be configured to guarantee discovery within a limited amount of time. Towards this, we first present a mechanism for bounding latencies and analyzing the discovery latencies of such protocols, which has been proposed in [6]. Next, we briefly introduce a parametrization scheme that has been proposed in [4]. The resulting parametrizations are proven to be optimal [7], which means that for a given energy consumption, no ND protocol can guarantee shorter discovery latencies.

While such parametrizations perform well in environments with small numbers of devices carrying out the ND procedure simultaneously, they lack robustness in busy environments with many devices discovering each other simultaneously.

This work has been partially supported by the German Research Foundation (DFG) under grant number CH918/5-1 - "Slotless Neighbor Discovery"

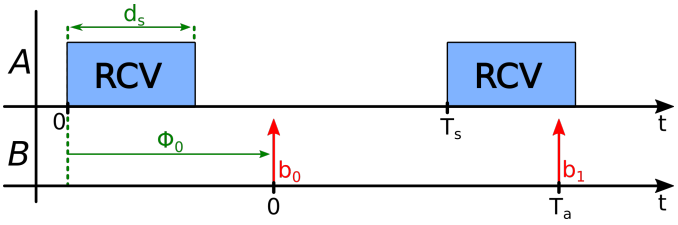

Fig. 1. Principle of a ND protocol. Device *A* listens to the channel periodically for d_s time-units with an interval T_s. Device *B* transmits beacons periodically every T_a time-units.

Due to collisions, only a certain fraction of the discovery attempts are successful within the worst-case latency. We therefore outline methods to increase the robustness of such ND solutions, and also study the hardware-properties that impact the dependability.

II. Deterministic Neighbor Discovery

In this section, we present how PI-based protocols can be configured such that the discovery latencies become bounded.

A. Overview

ND protocols work according to the following principle. For establishing a first contact to other wireless devices in range, every device repeatedly sends beacons and scans the channel for incoming packets from other devices. A device *A* discovers another device *B*, as soon as a beacon of *B* falls into a reception window of *A*. As already mentioned, we in this paper consider periodic interval (PI)-based protocols, in which beacons are sent periodically with a period T_a and reception windows of length d_s are repeated with an interval of T_s, as depicted in Figure 1. In this figure, the arrows depict beacons, whereas the rectangles depict reception windows. We first consider only two devices *A* and *B*, of which *A* only receives without transmitting, whereas *B* only transmits beacons without receiving. We generalize this later by discussing the additional problems that arise when multiple devices both receive and transmit.

Let beacon b_0 of device *B* be the first one that is sent after both devices have been brought into their range of reception. Let us consider the most recent scan window of device *A* that begins before this beacon transmission. Beacon b_0 of device *B* is sent by Φ_0 time-units after the beginning of this scan window (cf. Figure 1). Since the clocks of both devices are unsynchronized, the value of the offset Φ_0 is random. Which properties do the parameter values T_a, d_s and T_s need to fulfill for guaranteeing that a beacon will be sent within a reception window for all possible initial offsets Φ_0?

B. Mechanism for Determinism

Let us consider differences γ, which are formed by

$$\gamma = |i \cdot T_s - j \cdot T_a|\,, \tag{1}$$

with i and j being integer-values. A beacon overlaps with a scan window within bounded time only if there exist a tuple

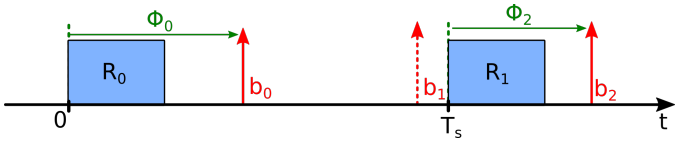

Fig. 2. Beacons of one device and reception windows of another device. The parametrization shown fulfills $\Phi_0 - \Phi_2 < d_s$ and hence guarantees deterministic discovery.

(i, j), for which $0 < \gamma < d_s$ [6]. This intuitively becomes clear from Figure 2: Let the first beacon b_0 that is sent after two devices come into range have a temporal distance of Φ_0 time-units from its neighboring scan window that is temporally on the left. Then, the temporal distance between a scan window that is $i \cdot T_s$ time-units later and a beacon that is sent $j \cdot T_a$ time-units later is $\Phi_0 - \gamma$ time-units. In Figure 2, $i = 1, j = 2$ and hence $\Phi_2 = \Phi_0 - \gamma$. Here, we label the offset Φ_j by the index of the beacon b_j. Clearly after $k = 1, 2, 3, 4...$ steps of $i \cdot T_s$ and $j \cdot T_a$ time-units, the temporal distance between the corresponding pair of scan window and beacon becomes $\Phi_0 - k \cdot \gamma$, $k = 1, 2, 3, 4, ...$ time-units. If now $0 < \gamma < d_s$, there must be one k for which the distance between a beacon and a scan window becomes smaller than d_s, regardless of the initial offset Φ_0. Hence, a beacon will lie within a scan window for every initial offset Φ_0 and hence, discovery is guaranteed to take place within bounded time. Such sequences of beacons and scan windows are called γ-*sequences* [6]. It is worth mentioning that the distance Φ_j can also grow for increasing values of j. Here, the distance to the scan window that neighbors b_0 temporally on the right shrinks for increasing values of k and hence, this case forms no exception.

For every given T_a, T_s and d_s, a set of γ-sequences can be identified using which the worst-case discovery latency can be computed (see [6] for details on this computation). Using this theory, one can show that except for a finite number of singularities, the discovery latencies are bounded for all parameter values. Figure 3 depicts the worst-case latencies for $T_s = 3\,\text{s}$, $d_s = 0.5\,\text{s}$ and sweeping values of T_a. As can be seen, the resulting function is highly irregular with multiple minima and maxima. This rises the following question: Which parameter values need to be chosen to achieve a good trade-off

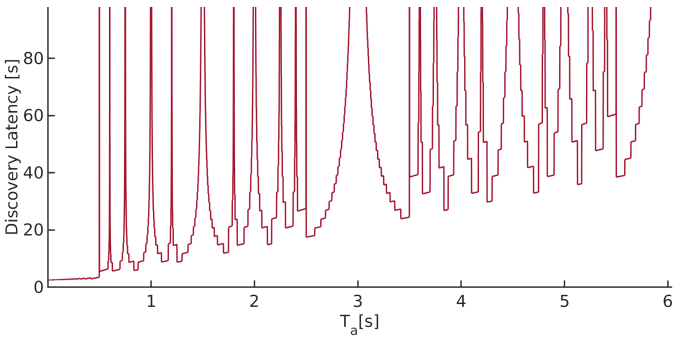

Fig. 3. Computed discovery latencies for $T_s = 3\,\text{s}$, $d_s = 0.5\,\text{s}$ and sweeping values of T_a.

between latency and energy-consumption?

C. Energy-Efficiency

The energy consumption of a ND protocol is usually expressed by the duty-cycle of the device running the protocol. In many scenarios, both devices send beacons and scan for the channel. Hence, a generic definition of the duty-cycle is given by

$$\eta = \frac{d_s + o_s}{T_s} + \alpha \frac{d_a + o_a}{T_a}. \quad (2)$$

Here, d_a is the transmission duration of a packet, o_a is the overhead for switching from sleep mode to transmission and vice-versa and o_s the overhead for switching from sleep mode to reception and vice versa. The factor α is the fraction of power the radio spends during transmission over the power it spends during reception.

As already mentioned, the goal is to identify parametrizations that optimize the trade-off between the duty-cycle (and hence the energy-consumption) and the discovery latency. Due to the large design space with three degrees of freedom (i.e., T_s, T_a and d_s), exhaustive searches are not feasible. In [4], a parametrization scheme to translate any given duty-cycle (and hence, energy budget) into optimized parameter values has been proposed. In a comparison with multiple popular ND protocols [4], none of the other neighbor discovery protocol could guarantee a lower worst-case latency than a PI-based protocol configured accordingly. However, as we describe next, these values are optimized for one-way discovery between one transmitting and one receiving device. We will describe the issues faced in the more general case of multiple radios both receiving and transmitting, next.

III. Reliable Operation

A. Limitations of Optimal Parametrizations

The parametrizations obtained by known parametrization schemes, e.g., [4], rely on the following properties:

1) For every possible initial offset Φ_0, exactly one beacon overlaps with a scan window per worst-case latency L (i.e., no redundancy).
2) The fraction between the duty-cycle spend for advertising (i.e., the rate using which beacons are sent) and for scanning (i.e., the fraction of scanning time per scan interval) is optimized.

However, the first of these properties implies that if one packet is lost, the worst-case latency will be exceeded, since only a later scan window can potentially lead to a successful discovery. In other words, such parametrizations enable a robust and energy-efficient connection setup procedure only in setups with few devices carrying out the discovery procedure simultaneously, in which collisions only seldom occur. In more busy environments, with e.g., more than 10 devices discovering each other, packet collisions prevent a reliable operation. To increase the robustness, countermeasures against the effects of collisions need to be taken, which are described next.

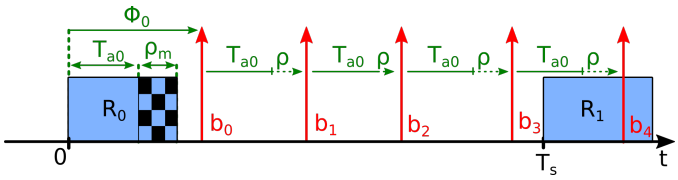

Fig. 4. Deterministic, decorrelated neighbor discovery protocol. Beacons are sent with T_{a0} plus a random delay $\rho \in [0, \rho_m]$. For guaranteeing discovery, d_s has to be extended by ρ_m time-units beyond its optimal value.

B. Decorrelation

Each radio transmits packets with the same interval T_a. Hence, if a pair of packets from two different devices collide, all later pairs of packets will also collide (since the temporal distance between any two packets is always T_a time-units). This means that in PI-based protocols without countermeasures, a certain fraction of discovery attempts, which is identical to the packet collision rate, permanently fails.

Therefore, the collision probabilities of the individual packets of one device need to be decorrelated from each other. This is usually achieved by randomizing the points in time packets are being sent at. For example, BLE adds a random delay to each instance of T_a, i.e., $T_a = T_{a0} + \rho$. Here, T_{a0} is the static part of T_a, whereas ρ is a random amount of time between $0\,\text{ms}$ and $\rho_m = 10\,\text{ms}$. With this, the probability of a second collision after two packets have collided is reduced. However, since packets are no longer sent at their optimal points in time, this deteriorates the energy-efficiency. Hence, a trade-off between the amount of random delay and energy-efficiency needs to be made. Figure 4 shows a protocol in which discovery is guaranteed within T_s time-units, since $\max(T_a) \leq d_s$. Since $T_a = T_{a0} + rand(0...\rho_m)$, d_s needs to be extended by ρ_m beyond the value needed for transmitting without the random delay, i.e., $T_a = T_{a,0}$. This additional listening time per scan interval increases the energy-consumption. Th optimal amount of random delay and hence which sacrifice on the energy-efficiency is appropriate in which scenario (e.g., expected number of devices discovering each other, etc.) has not been sufficiently studied in the literature, yet. This leads to multiple open research questions:

- Given a considered range of number of devices and the desired duty-cycle, how much random delay should be added? Small random delays increase the energy efficiency but deter the performance for larger numbers of devices.
- Since the duty-cycle is altered compared to the one based on which the parameter values have been computed, the resulting protocol will be less efficient. What are optimal parametrizations that account for the additional reception-duty-cycle that is induced by compensating for the randomness?
- How should the random delay be realized (e.g., distribution of random values)?

To the best of our knowledge, no solutions to these issues

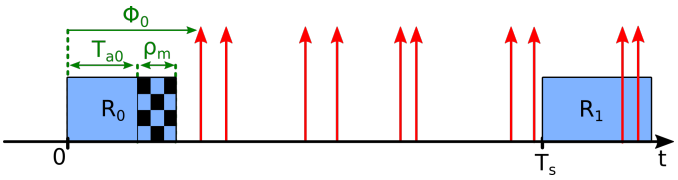

Fig. 5. Redundant ND protocol. For almost every initial offset Φ_0, 2 beacons lie within the reception window.

have been studied in the literature. The BLE specification [8] suggests a random delay between 0 and 10 ms per advertising interval, independent of the duty-cycle and the estimated number of devices. Clearly, this value cannot be optimal for all situations.

C. Redundancy

The random delay decorrelates the collision probabilities of multiple subsequent beacons. Hence, if a beacon that overlaps with a reception window collides with another beacon, the next overlapping beacon is successfully received with a high probability. However, in such cases, the theoretical worst-case latency is exceeded significantly. In the example from Figure 4, the worst-case is approximately $T_s - d_s$ time-units. If the beacon that overlaps with the scan window collides, the discovery latency will be nearly doubled to $2 \cdot T_s - d_s$, given that the next overlapping beacon does not also collide. A method to reduce this discovery latency is increasing the number of beacons sent, such that for every initial offset Φ_0, multiple beacons lie within every reception window. An extension to the situation shown in Figure 4, in which two beacons lie within a reception window for almost every initial offset, is shown in Figure 5. Here, if one beacon collides, an additional beacon falls into the reception window, which is spaced from the collided one by a random amount of time. Hence, the worst-case latency is only marginally increased in the case of one beacon colliding. It needs to be mentioned that the worst-case latency is degraded to a purely theoretic construct in such scenarios, since the second beacon could also collide and hence no discovery can be guaranteed. The theoretical worst-case latency therefore needs to be regarded as the time within which a very high fraction of discovery attempts are successful, e.g., 99 %.

Parametrization schemes that optimize the parameter values in such redundant scenarios are yet to be found. In particular, what is the ideal rate of beacon transmissions given a duty cycle, a desired success rate and and certain number of devices discovering each other simultaneously? How should they be spaced and, since the system should perform well in different environments, what is the best trade-off both for sparse and dense environments?

D. Hardware Aspects

So far, we have considered protocol design aspects of robust ND. However, also the hardware has an important impact on the robustness. An obvious requirement is a negligible low bit error rate, which modern radios do fulfill [9]. However, there is one key property that has received little attention until now: The radio's turnaround times not only play an important role in the energy efficiency, but also heavily impact the robustness of any bi-directional discovery procedure because of the following reason [7]. Let a device *A* both transmit beacons and listen to the channel periodically. Let a second device *B* have the same pattern of scan windows. Since the sequence of beacons of device *A* is designed such that at least one beacon lies within a scan window of device *B*, and since *A* has the same pattern of scan windows as *B*, it is inevitable that device *A* sends at least one beacon within one of its own scan windows. Therefore, a scan window needs to be interrupted and hence, the radio switches from reception to transmission and then back to reception. Both turnaround times lie within the scan window, during which the radio cannot receive any incoming packets. This results into an increased rate of failed discoveries. Hence, if the turnaround times could be reduced in future radios, the robustness of the ND procedure could be further increased.

IV. Concluding Remarks

In this paper, we have presented the main mechanism of setting up connections between IoT devices in a reliable fashion by bounding the discovery latency and energy consumption. We have outlined which mechanisms future protocols could use to increase the robustness of bi-directional discovery also in busy environments. In particular, we have presented methods to decorrelate the collision probabilities of multiple subsequent beacons on a device, and we have proposed redundantly probing techniques. Future research needs to optimize these techniques towards the best possible trade-off.

References

[1] P. Dutta and D. Culler, "Practical asynchronous neighbor discovery and rendezvous for mobile sensing applications," in *ACM Conference on Embedded Network Sensor Systems (SenSys)*, 2008, pp. 71–84.

[2] A. Kandhalu, K. Lakshmanan, and R. Rajkumar, "U-connect: A low-latency energy-efficient asynchronous neighbor discovery protocol," in *International Conference on Information Processing in Sensor Networks (IPSN)*, 2010, pp. 350–361.

[3] M. Bakht, M. Trower, and R. Kravets, "Searchlight: Won't you be my neighbor?" in *Annual International Conference on Mobile Computing and Networking (MOBICOM)*, 2012, pp. 185–196.

[4] P. Kindt, M. Saur, and S. Chakraborty, "Slotless protocols for fast and energy-efficient neighbor discovery," *CoRR*, vol. abs/1605.05614, 2016.

[5] P. Kindt, D. Yunge, G. Reinerth, and S. Chakraborty, "Griassdi: Mutually assisted slotless neighbor discovery," in *ACM/IEEE International Conference on Information Processing in Sensor Networks (IPSN)*, 2017, pp. 93–104.

[6] P. Kindt, M. Saur, and S. Chakraborty, "Neighbor discovery latency in BLE-like protocols," *IEEE Transactions on Mobile Computing (TMC)*, vol. 17, no. 3, pp. 617–631, 2018.

[7] P. Kindt and S. Chakraborty, "On optimal neighbor discovery," in *Conditionally Accepted to ACM SIGCOMM*, 2019.

[8] Bluetooth SIG, "Specification of the Bluetooth system 5.0," December 2016, volume 0, available via bluetooth.org.

[9] E. Tsimbalo, X. Fafoutis, E. Mellios, B. Haghighi, M.and Tan, G. Hilton, R. Piechocki, and I. Craddock, "Mitigating packet loss in connectionless bluetooth low energy," in *IEEE World Forum on Internet of Things (WF-IoT)*, 2015.